INSTITUTO
UNIVERSITÁRIO
DE LISBOA

DevOps Practices in IT Service Management

João Pedro Carvalho Faustino

PhD in Information Science and Technology

Supervisors:
Doctor Rúben Filipe de Sousa Pereira, Assistant Professor,
ISCTE - Instituto Universitário de Lisboa

Doctor Miguel Leitão Bignolas Mira da Silva, Full Professor,
Universidade de Lisboa - Instituto Superior Técnico

September, 2024

TECNOLOGIAS
E ARQUITETURA

Department of Information Science and Technology

DevOps Practices in IT Service Management

João Pedro Carvalho Faustino

PhD in Information Science and Technology

Jury:
Doctor Bráulio Alexandre Barreira Alturas, Associate Professor with Aggregation,
ISCTE - Instituto Universitário de Lisboa (President)
Doctor Rui Filipe Lima Maranhão de Abreu, Full Professor,
Faculdade de Engenharia da Universidade do Porto
Doctor Virgínia Araújo, Professor,
Universidade Atlântica
Doctor Antonio Trigo, Assistant Professor,
ISCTE - Instituto Universitário de Lisboa
Doctor Rúben Filipe de Sousa Pereira, Assistant Professor,
ISCTE - Instituto Universitário de Lisboa

September, 2024

“Well done is better than well said.”

Benjamin Franklin

Acknowledgments

When I started this PhD thesis, I did not believe it would become such an enriching and transformative journey. It has challenged me in ways that have fostered personal and intellectual growth. Throughout this process, I have learned the value of perseverance, the importance of collaboration, and the power of curiosity.

I am incredibly grateful for the unwavering support, guidance, and encouragement from so many along the way. Without their contributions, this thesis would not have been possible. I acknowledge those who have helped me throughout this journey with sincere appreciation. Therefore, here it goes: my honest and heartfelt thank you.

First, I thank my supervisors, Prof. Dr. Rúben Filipe de Sousa Pereira and Prof. Dr. Miguel Mira da Silva. All the incentives, availability, and guidance were some of the main reasons this thesis reached this stage.

Alexandra, my love, my dear wife, I would like to thank you for all your support and patience in all the critical phases of my life. Thank you for being my cheerleader and encouraging me to be successful in this journey.

To my family, father Armando, mother Catarina, and sister Vanda, thank you for your belief in me, support, and strength throughout this journey.

I would also like to thank my childhood friends Helder, Ivo, Marco, and Paulo for all the support, laughs, and companionship during this stage of my life.

Finally, I want to thank my co-workers, PhD colleagues, organizations, and others who contributed to this thesis's result.

Without you all, this would not have been possible.

Resumo

O departamento de IT é vital para as organizações para enfrentarem as diversas ameaças ao seu negócio. De forma a garantir o alinhamento de expectativas entre os objetivos do IT e das áreas de negócio das organizações, estas têm vindo a adotar *frameworks* de Gestão de serviços de IT como o ITIL. Para evitar que o IT seja uma barreira para responder às necessidades do negócio, as metodologias *Agile* foram criadas para implementar alterações e *feedback* rapidamente. A cultura *DevOps* surge como solução para garantir a comunicação entre *developers* e *operators* possibilitando uma maior frequência de entregas, sem comprometer a estabilidade. Esta investigação vai explorar a integração de *DevOps* em um ambiente ITIL com o foco em três processos: gestão de incidentes, problemas e alterações. Vão ser conduzidos casos de estudo em organizações que utilizam praticas de *DevOps* e um dos processos, com o objetivo de encontrar que atividades dos processos são impactadas pelas práticas de *DevOps*, benefícios e desafios, e melhorias de desempenho no processo. Para validar os resultados destes casos de estudo vão ser realizadas entrevistas a outros indivíduos de outras indústrias e com diferentes experiências profissionais. *Application Monitoring* e *Feedback Loops* aceleraram a deteção de problemas e incidentes mais rapidamente, tal como *Continuous Integration* acelera a resolução de incidentes e problemas. Igualmente, foi possível concluir que *Automated Deployment* acelera a autorização, coordenação e implementação de alterações. Diversas práticas como *Feedback Loops, Stakeholder Participation, Continuous Integration e Delivery* levaram a um aumento de desempenho dos processos.

Palavras Chave: DevOps; ITSM; ITIL; Gestão de Incidentes; Gestão de Problemas; Gestão de Alterações;

Abstract

The IT department plays a vital role in enabling organisations to respond to the various threats to their business. To ensure alignment between IT objectives and business expectations, organisations have increasingly adopted IT service management frameworks such as ITIL. To prevent IT from becoming a barrier to meeting business needs, Agile methodologies were introduced to enable faster implementation of changes and feedback. DevOps culture emerges as a solution to ensure communication between developers and operators, enabling more frequent deliveries without sacrificing operational stability. This research explores the integration of DevOps into an ITIL environment, focusing on three core processes: Incident Management, Problem Management, and Change Management. Case studies were conducted in organisations that apply DevOps practices in at least one of these processes, aiming to identify which process activities are impacted by DevOps practices, the associated benefits and challenges, and performance improvements. To validate the findings, additional interviews were conducted with professionals from various industries and backgrounds. The results show that practices such as Application Monitoring and Feedback Loops help detect problems and incidents more quickly, while Continuous Integration accelerates their resolution. Similarly, it was found that Automated Deployment speeds up the authorization, coordination, and implementation of changes. Several practices such as Feedback Loops, Stakeholder Participation, Continuous Integration and Delivery lead to a process performance improvement .These results were verified by the different data collection methods used in the study.

Keywords: DevOps; ITSM; ITIL; Incident Management; Problem Management; Change Management;

Contents

Resumo ... vii
Abstract ... ix
Contents ... xi
List of Figures ... xiii
List of Tables ... xiii
List of Acronyms ... xv
Chapter 1 ... 1
Introduction ... 1
1.1. Context and Motivation ... 1
1.2. Research Background ... 5
1.2.1. ITIL ... 5
1.2.1.1. IM Process ... 6
1.2.1.2. PM Process ... 7
1.2.1.3. CM Process ... 7
1.2.2. DevOps ... 8
1.3. Publications ... 9
Chapter 2 ... 11
Publication nr. #1 ... 11
Chapter 3 ... 13
Publication nr. #2 ... 13
Chapter 4 ... 15
Publication nr. #3 ... 15
Chapter 5 ... 17
Publication nr. #4 ... 17
Chapter 6 ... 19
Publication nr. #5 ... 19
Chapter 7 ... 21
Publication nr. #6 ... 21
Chapter 8 ... 23
Publication nr. #7 ... 23
Chapter 9 ... 25
Publication nr. #8 ... 25
9.1. Case Study Data Collection Triangulation ... 62
9.1.1. Direct Observation ... 62
9.1.2. Focus Group Exercise ... 67

9.2. Complementing Publication nr. #8 with Extra Interviews ... 72
9.2.1. Extra Interview Analysis on DevOps Impact on Process Activities ... 73
9.2.2. Interview Analysis for DevOps Benefits and Challenges ... 79
9.2.3. Interview Analysis for Process Performance ... 84
9.3. Synthetization of Extra Interviews and Publication nr. #8 ... 86
Chapter 10 ... 91
Conclusions ... 91
10.1. Discussion and Contributions ... 91
10.2. Limitations and Future Work ... 94
Artificial Intelligence Disclaimer ... 95
References ... 95
Appendix A - A Systematic Literature Review on DevOps Capabilities and Areas ... 101
Appendix B – A Maturity model for DevOps ... 123
Appendix C – Agile information technology service management with DevOps: An incident management case study ... 171
Appendix D – An IT Service Management Literature Review: Challenges, Benefits, Opportunities and Implementation ... 223
Appendix E – DevOps benefits: A systematic literature review ... 247
Appendix F - DevOps and Problem Management: A Case Study ... 269
Appendix G – The Influence of DevOps Practices in ITSM Processes ... 289

List of Figures

Figure 1.1 – Research Concept……………………………………………………..………..…..…4
Figure 1.2 – List of Publications….……………………………………….………………...……10
Figure 10.1 – DevOps Practices Impact in ITIL Processes……………………...………..……………93
Figure 10.2 – DevOps Practices Impact on Process Performance……………………...………………94

List of Tables

Table 1.1 - ITSM Challenges and Opportunities.. 3
Table 1.2 - DevOps Benefits 4
Table 1.3 - Research Questions 5
Table 1.4 - List of Publications 9
Table 9.1 – Mapping of ITIL & Organizations' IM lifecycle 63
Table 9.2 – DevOps Practices in Organization IM Process Activities 65
Table 9.3 – Mapping of ITIL & Organizations' PM lifecycle 66
Table 9.4 - DevOps Practices in Org. Problem Management 67
Table 9.5 - Focus Group - Impact of DevOps in IM Process 70
Table 9.6 - Focus Group - Impact of DevOps in PM Process 71
Table 9.7 - Interviewees Detail for Additional Interviews 72
Table 9.8 – Matches Between DevOps Practices and Process Activities (Additional Interviews) 74
Table 9.9 - Insights on DevOps & IM process (Additional Interviews) 75
Table 9.10 - Insights on DevOps & Normal Change process (Additional Interviews) 76
Table 9.11 - Insight on DevOps & Standard Change process (Additional Interviews) 76
Table 9.12 - Insight on DevOps & Emergency Change process (Additional Interviews) 77
Table 9.13 - Insights on DevOps & PM process (Additional Interviews) 77
Table 9.14 - DevOps Benefits (Additional Interviews).. 81
Table 9.15 - DevOps Adoption Challenges (Additional Interviews) 83
Table 9.16 - DevOps Process Improvement (Additional Interviews) 85
Table 9.17 - DevOps Practices and Process Synthetization about Impact 87
Table 9.18 - DevOps Practices and Benefit Synthetization... 88
Table 9.19 - DevOps Practices and Process Improvement Synthetization... 89

List of Acronyms

CM	Change Management
COBIT	Control Objectives for Information and Related Technologies
DORA	DevOps Research and Assessment
EM	Event Management
IaC	Infrastructure as Code
IM	Incident Management
IT	Information Technology
ITIL	Information Technology Infrastructure Library
ITSM	IT Service Management
LR	Literature Review
PM	Problem Management
RM	Release Management
SCM	Service Catalog Management
SDLC	Software Development Lifecycle
SLR	Systematic Literature Review
SRM	Service Request Management

CHAPTER 1

Introduction

This chapter presents all the work performed for this research. It starts by framing the reader about the context, problem, and research questions this research will answer.

This research will be presented as a compilation of publications regarding the main topic. Thus, the following sections show a high-level research background to avoid duplicating definitions and background for each subject.

Another section, at the end of this chapter, shows the publications made and their purpose for this research. This document structure follows an introduction, publications, a conclusion, and references.

1.1. Context and Motivation

Over the decades, organizations have been changing how they manage their businesses to overcome the threats from internal and external environments and their competitors [1], [2]. Nowadays, one of the most essential factors in an organization's ecosystem is the Information Technology (IT) department. The relevance of IT has been growing widely in organizations to achieve their mission and business goals due to the support of business units [3].

The concept of IT services was created to support businesses in the market's fast changes and tendencies [4]. Furthermore, to evaluate how the IT Services are performing towards the business and organizational objectives, a discipline named IT Service Management (ITSM) was created [5]. The objective of ITSM is to manage the IT Services landscape of the organization from the moment of its design until its decommission [6].

ITSM is implemented by using several frameworks such as Information Technology Infrastructure Library (ITIL), Control Objectives for Information and Related Technologies (COBIT), and FitSM, where ITIL is seen as the most implemented framework globally [7], [8]. ITIL evolved until its latest release, ITIL4, was released in 2019. ITIL can be seen as "a set of practices for ITSM that focuses on aligning IT Services with the needs of business." [9].

Traditional ITIL implementations focus on implementing processes to manage IT services. However, ITIL4 is focused on practices, extending these implementations for more than processes [10]. Moving to this practice approach enabled ITIL4 not just to overthink the processes' performance but to combine processes, technology, and people to guarantee the best performance of the IT Service, focusing on value to its customers [11].

However, due to the complexity of the processes, some organizations choose not to implement all ITIL processes, like small-scale organizations [12]. Therefore, organizations usually decide to implement Incident Management (IM), Problem Management (PM), and Change Management (CM) [13]. These three practices are typically seen working together to maintain stability and the regular operation of the IT Services [14], ensuring the correct transition from a service design to its operation [15].

IM is the process where the objective is to minimize the negative impact of a service's unplanned interruption or reduction in the quality of the service by restoring the regular operation as quickly as possible [9]. This will prevent economic losses and user dissatisfaction [16].

For PM, the objective is to reduce the likelihood and impact of incidents by identifying their actual and potential causes [9].

CM can be seen as the last stage in which all the changes that impact IT services are managed [9]. A change can be parameterization, code, or hardware. In this process, many approvals and review activities can be seen as exhaustive, causing delays to the delivery of new fixes (incoming from IM and PM) or new functionalities for the IT Service [17].

However, ITIL can bring many benefits, such as more control of the IT environment and risk and cost reduction, but it also has its challenges. ITIL can be seen as bulky and bureaucratic due to the focus on process performance and roles well-defined roles, not the best option due to the current market speed to satisfy customer demands [18], [19] .

Moreover, a culture named DevOps emphasizes the collaboration between development and operations teams [20]. The main objective of this culture is to improve the delivery of IT Services through the automation of processes and the adoption of the idea of continuous improvement [21]. Due to the collaboration between these two IT teams, DevOps could improve ITIL implementation and the implementation of ITIL processes.

Therefore, this research aims to investigate the relationship between DevOps and ITIL processes to understand the probable impacts of a DevOps implementation on an ITIL environment. Moreover, process terminology will be used since organizations are still adapting to the ITIL practices. But DevOps also focus on combining processes, technology and people as dimensions to achieve its objectives [22], showing that ITIL and DevOps uses the same dimensions to succeed its goals.

During this research, three Systematic Literature Reviews (SLRs) were conducted to identify what could be found in the existing literature about DevOps and ITSM. SLRs are mainly used to perform literature reviews due to the description of the extensive and transparent protocol that is used [23], improving the rigor of the review [24]. The first SLR focused on identifying the challenges, benefits, opportunities, and implementation practices of ITSM implementations [13]. This SLR pointed out several challenges and opportunities organizations face with ITSM, as seen in Table 1.1

The second SLR focused on finding which DevOps benefits are documented in the literature regarding DevOps implementations [25]. Table 1.2 shows these benefits.

DevOps adoption is suggested as an opportunity to improve the ITSM implementation, as seen in Table 1.1, so IT Services can be delivered faster [26]. Moreover, in Tables 1.1 and 1.2, some challenges and opportunities could be matched with some DevOps benefits. For example, in an organization that is facing resistance to ITSM implementation (C1), we can see that DevOps implementations usually benefit from organizational cultural changes (B19) and increase employee motivation (B15).

Table 1.1 - ITSM Challenges and Opportunities

Challenges	
ID	**Description**
C1	Organization resistance
C2	Frameworks complexity
C3	Lack of knowledge/skills
C4	Processes assessment (costly and time-consuming)
C5	Lack of management support
C6	Lack of resources
C7	Difficult on quantifying the benefits
C8	Big investment needed (Implementation and maintaining momentum)
C9	Hard of planning to implement multiple frameworks
C10	Steady lower costs
Opportunities	
ID	**Description**
O1	Lack of guidelines to processes improvement
O2	Processes assessment
O3	Identifying processes interdependencies and their overlap
O4	Maturity Models for the needs of IT management providers
O5	Cloud computing and DevOps

Thus, it could be helpful to adopt a DevOps culture. Another example could be O1 and O2, where DevOps seeks to standardize processes and tools across the organization. Therefore, the process assessment could be simplified and described to describe how to improve.

This motivated the researchers to perform a third SLR based on how DevOps could impact ITSM [27]. As stated before, this research will focus on the ITIL framework; therefore, this third SLR also concerns ITIL processes. The following processes were identified as impactable by implementing DevOps practices: Release Management (RM), IM, PM, and CM. The chosen processes for this research were IM, PM, and CM since these have been more adopted globally as stated previously, and since these were identified as most probably impacted processes according to Faustino et al. [27]. Moreover, these processes are typically seen where developers and operators are more likely to work together, showing how the DevOps practices have impacted the processes.

Table 1.2 - DevOps Benefits

ID	Description
B01	Improvement of Synergy
B02	Faster Time to Market
B03	Faster and Better Feedback
B04	Increase of Code Quality
B05	Increase of Value
B06	Improvement of System Reliability
B07	Less Mean Time to Recover
B08	Increase in Team Performance
B09	Costs Reduction
B10	Processes and Tools Standardization
B11	Maximization of Competences
B12	Decrease of Manual Work
B13	Increase in Customer Satisfaction
B14	Less Failed Changes
B15	Increase in employee motivation
B16	More Innovation
B17	Better Deployment Management
B18	Fewer Security Issues
B19	Organizational Cultural Changes

Based on what is stated before, this study will focus on the question, "How do DevOps practices *impact ITIL processes?*" Figure 1.1 summarizes the context and motivation for this study and how implementing DevOps practices can benefit ITIL processes.

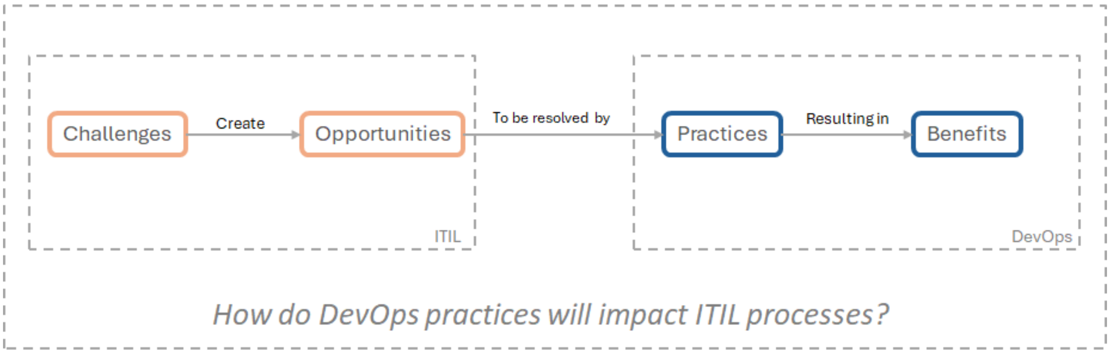


Figure 1.1 - Research Concept

Moreover, since this study does not aim for all ITIL processes, the researchers have split this main question into three research questions, as Table 1.3 shows.

Table 1.3 - Research Questions

ID	Research Question Description
RQ1	How do DevOps practices impact the IM, PM, and CM process?
RQ2	What are the benefits and challenges of implementing DevOps practices in the processes?
RQ3	Has the process's performance improved?

The research methodology used for this is a multi-case study, where the objective is to perform case studies in different organizations, in different ITIL processes, and in which DevOps practices have also been adopted. These case studies show how these two topics coexist in the organization's IT ecosystem. They show how they can relate to this and the benefits and challenges of this DevOps adoption.

The following sections will provide a background for ITIL and its processes and DevOps to help the reader better understand these topics.

1.2. Research Background

Given the objective of this research, this section will explore theoretical concepts about DevOps and ITIL processes to find how these can relate and the probable impacts that could be caused on each of them.

As stated, ITIL is overfocused on process performance and accountability. At the same time, the organization's market requires agility for fast and quality product deliveries to respond to quick market changes. Organizations must change their mindset and culture to incorporate and accept customer feedback. This is one of the main premises of the DevOps culture: adapting the IT landscape to deliver fast and quality software.

1.2.1. ITIL

ITSM can be seen as a holistic approach that is responsible for all the activities of creating, implementing, supporting, and managing IT services. ITSM has been implemented through several frameworks, such as COBIT, ITIL, and FitSM [28]. Moreover, ITIL stands out in terms of implementation compared to the other frameworks [7], [12].

ITSM is a process-oriented approach that uses several processes to improve IT efficiency by measuring IT services' performance. These processes guarantee the quality of IT Services, transparency of IT processes, and the IT changes delivered to users [29].

ITSM is "*a methodical approach to managing IT services – from design, implementation, operation to continual improvement. It not only focuses on the technical aspects of IT but also allows the alignment of services and functions provided by IT within the organization.*" [30, p. 6] .

This study will focus on ITIL since it is one of the most globally implemented ITSM frameworks [8], where a new version, ITIL4, was released in 2019 [31].

Older versions of ITIL, such as ITIL 2011, emphasized IT processes with well-defined inputs, outputs, and roles, while ITIL4 extended the framework to be modernized and flexible [32]Practices are sets of organizational resources designed to perform work or accomplish an objective; they are more flexible and less prescriptive than processes.

ITIL 2011 aimed to manage IT Services between five stages of the IT Service lifecycle: Service Strategy, Service Design, Service Transition, Service Operation, and continuous service Improvement. In ITIL4, the focus is on the value of the IT service. For this, ITIL4 introduces the service value system, a holistic approach where several interconnected activities create and deliver value. This is called a service value chain.

This service value chain is crucial for ITIL4 since it manages the IT Service from its creation until its delivery, constantly monitoring the value it creates to know when to improve.

1.2.1.1. IM Process

IM plays a crucial part in ITSM, being one of the fundamental processes of ITIL [33].

An incident is a non-planned interruption or performance degradation of an IT Service, which also applies to any component that supports the IT service. Since the IT services should support the organization's business processes, it is possible to state that an incident will cause a business interruption [34], [35]. The objective of the IM practice is to manage these incidents so that they minimize service disruptions and outages and have less impact on the organization's business units [36].

An organization's main objective is to create profit, so if the business has some interruption in the IT Services that support the business applications, it will make economic losses, which should be avoided or mitigated as soon as possible [16]. These kinds of business interruptions are not only seen as a negative impact from an economic perspective. However, they could also result in a lack of confidence from the organization’s customers, even internal customers, showing that the IT services are unreliable for the business.

As stated before, the objective of the IM process is to bring the IT Service to its regular operation and performance, so when a complaint or a call reaches the service desk, it will check if it is a known issue and if there is already any solution for the user issue. If not, the service desk will try to find a workaround to solve the incident. If there is no workaround, the incident will be sent to another support group, but if successful, the incident will be closed, and a problem will be raised to solve the root cause. From this moment, PM urges to solve this root cause, where the next section will provide some theoretical background about it.

1.2.1.2. PM Process

In ITIL, a problem is seen as the root cause of the incident. Therefore, the PM practice is said to be used to solve the root cause of one or more incidents, minimizing the impacts that these incidents can have on the organization [4]. The problem manager is responsible for thoroughly analyzing previous incidents, identifying the root cause, and implementing measures to prevent these issues from reoccurring.

PM is about solving issues and preventing them from occurring [37]. This proactive approach, which includes continuous improvement, minimizes the impact on business units and ensures excellent operational stability. This focus on prevention is a mindset highly valued by organizations, reassuring that issues will not reoccur.

As seen in the implementation activity for the PM practice, some solutions require a change, creating a connection to the CM process.

The following section will provide a theoretical background for the CM process.

1.2.1.3. CM Process

CM is not just about integrating changes in IT Services but about fostering a culture of continuous improvement within the organization [9]. By extending the change management process from ITIL 2011, CM can transform the organization's approach to change, engage stakeholders, improve communication, and implement strategies to address potential challenges and resistance to change.

Globally, most organizations use three change management processes: normal change, standard change, and emergency change.

Below is a description of each type of change, adapted from [38]:

- **Standard** - This is used for pre-authorized changes that have a negligible impact on the IT service and follow a well-known process.
- **Emergency** - A change must be implemented immediately to solve a major incident that seriously impacts an IT service. Usually assessed by the Emergency Change Advisory Board
- **Normal** - A change that requires approval from the Change Advisory Board occurs when a significant change can impact an IT service, business processes, or infrastructure.

Since standard changes are pre-authorized changes implemented by following a known procedure, they do not require as many approvals and assessments as normal changes [39]. An example of a standard change could be resetting a user's password. This change needs to be made to the IT Service but is low risk and impact, so the standard change process should be followed.

However, from a high-level perspective, these processes look bulky and bureaucratic, where they have a lot of different people and roles to secure different activities [19], [39].

From nowadays perspective, this does not look realistic due to the constant changes that the market faces. Changes and adaptations to IT services must be fast enough for organizations to adapt to the market changes to face their competition and customer demands [40]. To face this difficulty in delivering new solutions, fixes, and adaptations to IT services, a culture named DevOps has urged. The main premise of this culture is to deliver software quickly with high quality by automating several manual tasks from the Software Development Lifecycle (SDLC) and by joining IT Development and operations to work together to develop and maintain software [22], [41].

1.2.2. DevOps

The Agile Infrastructure Conference first mentioned the DevOps culture [42]. The main goal of DevOps culture is to bridge the Operations and Development IT teams to work together so both teams can help each other from the starting point of software development to high product quality and stable software [43], [44]. DevOps also follows the agile software development practices from the Agile Manifesto [45], focusing on individuals and their skills and embracing the change to correspond to customer feedback, even if this could impact process performance[46]. This way, the developed product will be more valuable to the customer since it will correspond to their needs.

These two IT teams have two different points of view regarding software changes [47]. Developers want to deliver new and improved features, while operations want to maintain the stability of IT services or IT systems; therefore, they do not want developers to deliver new features since they fear they will compromise the stability. This generates a blame game between these two roles where the operators blame the developers for compromising the system stability, while the developers blame the operators for not maintaining the system's operability to receive new features [48], [49].

When these two IT teams collaborate to develop new features, operators can be involved from the beginning of the SDLC and help the developers deliver quality and stable software by providing the necessary infrastructure and tools to ensure the software will run with the desired performance [50].

1.3. Publications

While developing this research, several publications, shown in Table 1.4, were performed and submitted to different scientific journals. The objective was to ensure that the scientific community validated the subject since it would go through a rigorous peer review process, getting the recognition for being published. The rank column refers to the rank at on the year that the research was published.

Table 1.4 - List of Publications

#	Publication Name	Authors	Journal	Rank	Publication	Status	Chapter
1	A Systematic Literature Review on DevOps Capabilities and Areas	Daniel Teixeira; Ruben Pereira; Telmo Antonio Henriques; Miguel Silva; João Faustino	International Journal of Human Capital and Information Technology Professionals	Q2	2020	Published	Introduction Publication nr. #1
2	A maturity model for DevOps	Daniel Teixeira; Rúben Pereira; Telmo Henriques; Miguel Mira da Silva; João Faustino; Miguel Silva	International Journal of Agile Systems and Management	Q2	2020	Published	Introduction Publication nr. #2
3	Agile information technology service management with DevOps: An incident management case study	João Faustino; Ruben Pereira; Braulio Alturas; Miguel Mira da Silva	International Journal of Agile Systems and Management	Q2	2020	Published	Introduction Publication nr. #3
4	An IT Service Management Literature Review: Challenges, Benefits, Opportunities and Implementation	João Serrano; João Faustino; Daniel Adriano; Ruben Pereira; Miguel Mira de Silva	Information	Q2	2021	Published	Introduction Publication nr. #4
5	DevOps benefits: A systematic literature review	João Faustino; Daniel Adriano; Ricardo Amaro; Ruben Pereira; Miguel Mira da Siva	Software: Practice and Experience	Q2	2022	Published	Introduction Publication nr. #5
6	DevOps and Problem Management: A Case Study	Daniel Adriano; João Faustino; Ruben Pereira; Rafael Almeida; Miguel Mira da Silva	SSRN	-	2022	N/A	Introduction Publication nr. #6
7	The influence of DevOps practices in ITSM processes	João Faustino; Ruben Pereira; Miguel Mira da Silva	International Journal of Services and Operations Management	Q3	2023	Published	Introduction Publication nr. #7
8	The Impact of DevOps in IT Service Management – A Multi-Case Study	João Faustino; Ruben Pereira; Miguel Mira da Silva	International Journal of Information Management	Q1	2024	Under Review	Publication nr. #8

Figure 1.2 shows the relationship between all these publications and how these fit into this thesis.

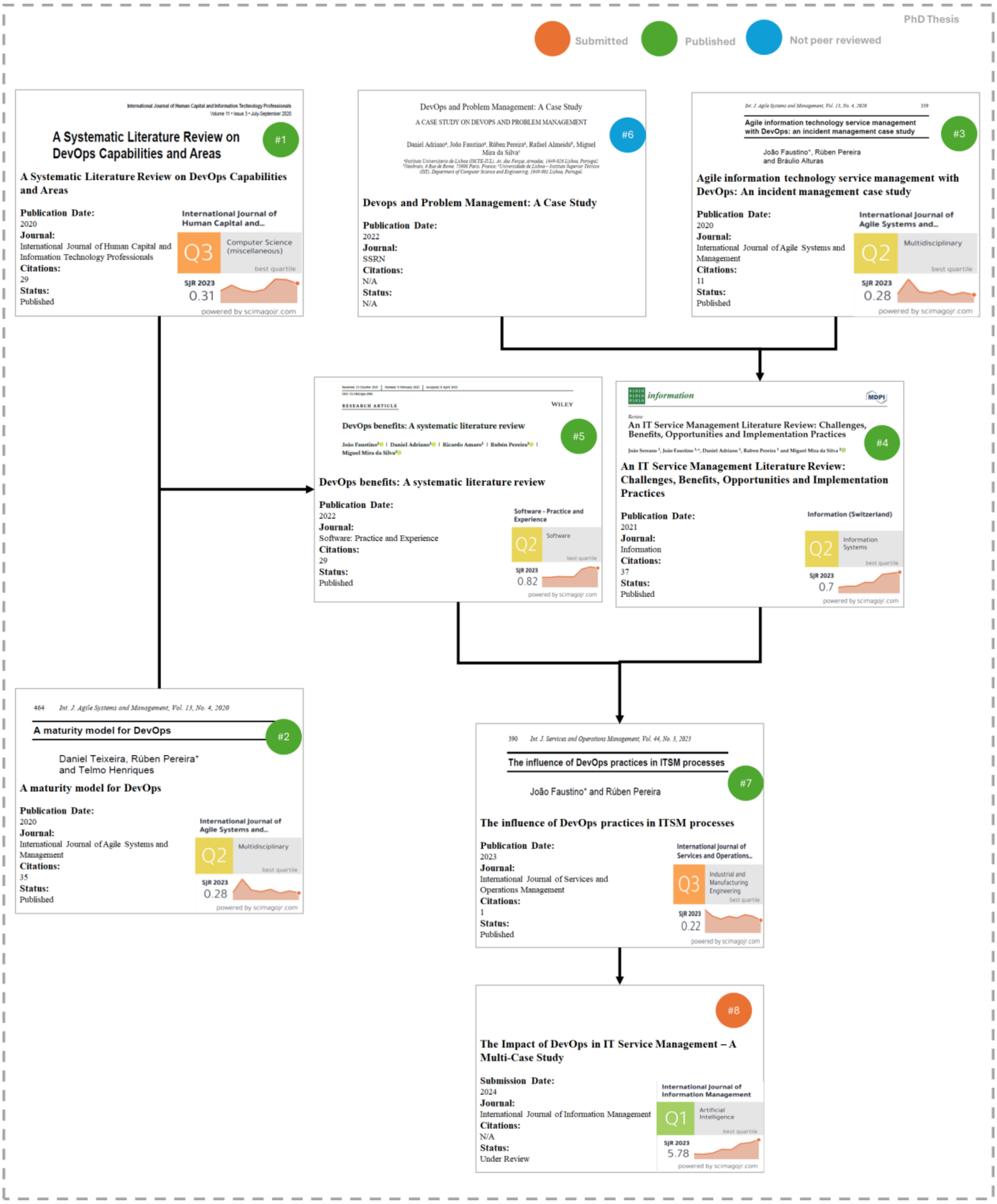


Figure 1.2 - List of Publications

The following chapters will have the contents of each publication made for this thesis.

CHAPTER 2

Publication nr. #1

This is the first publication regarding this thesis, to which the PhD candidate contributed. When performing this research, DevOps was a novel concept, and there was not a common understanding of its definition.

This research aimed to establish determining factors of DevOps implementations, including the main DevOps capabilities and areas with which it evolves. To perform this study, an SLR was used as a research methodology to find guidance and more knowledge about DevOps implementations in organizations. The research focused on the main areas of DevOps and its capabilities, and different digital libraries were used to find the related publications in the literature.

This research was essential to understand how DevOps has been perceived by organizations and the positive evolution that DevOps has had in terms of publications, showing that it is an emerging topic nowadays. Also, this publication was essential in providing a theoretical background about DevOps. It was possible to understand that DevOps is not just employing automation on manual tasks; it is more than just technical practices to improve efficiency. DevOps focuses on a culture of collaboration and sharing, combining people, technology, and processes to align every organization's resource to a common goal.

Nevertheless, this research provided significant input for this thesis by resuming the DevOps practices and how the organizations have progressively adopted these.

The complete publication can be found in Appendix A.

CHAPTER 3

Publication nr. #2

This publication, where the author was also a contributor, focused on DevOps. Due to being a novel topic, adoption or maturity models for DevOps implementation did not exist, and a gap in the existing literature was found. Using the literature review from Publication nr. #1 from Chapter 2, to identify the DevOps practices and areas, this research focused on performing interviews with DevOps experts to build a maturity model based on the professional experience of these experts.

The methodology Design Science Research was used for this study. After identifying the problem and motivation and the objectives and solutions, interviews with DevOps experts started. After several rounds of interviews, the model was demonstrated to teams that work with DevOps practices, and lastly, evaluation interviews were performed.

In total, 28 interviews were conducted with the DevOps experts, and their experience made it possible to assign a specific maturity level for each DevOps practice. After completing all the steps, the proposed maturity model was completed based on the CMMI Maturity model, enabling any organization to assess its maturity level regarding DevOps practices.

The complete publication can be found in Appendix B.

CHAPTER 4

Publication nr. #3

This publication was the first step in motivating this research. It was a publication regarding the master's thesis of the PhD candidate, which first explored how DevOps could impact the IM process, opening a path to exploring other processes.

This publication includes a case study of an application management team that supports an insurance company's core system. This team aims to maintain the core system by solving incidents and enhance the application by delivering new developments to it.

This team used agile methodologies and some DevOps practices to reach its goals successfully. They automated manual steps and improved collaboration between all the interested parties of the core system.

Moreover, this publication shows a positive result on applying these practices on a team responsible for solving incidents and accelerating their resolution, which contributes to a healthier system and business users' satisfaction by not facing issues when performing their tasks. Nevertheless, not only the corrections could be delivered faster, but new developments as well. This allows business users to have new functionalities so the organization can compete within the market.

This publication also gave space for future work where the case study reported that DevOps could impact other processes.

The full publication can be found in Appendix C.

CHAPTER 5

Publication nr. #4

The following publication is also suitable for establishing the motivation for this thesis. This publication was a collaboration between the PhD candidate and another PhD colleague. The objective was to perform a Literature Review (LR) of the existing knowledge about ITSM.

The research methodology used for this publication was the SLR due to the extensive protocol required to add more rigor to the LR. The focus of this publication was to find the benefits and challenges of ITSM adoption and implementation practices.

With this study, the researchers were able to find the gaps and some challenges of the ITSM implementations. Moreover, and quite important, was the ability to find opportunities to improve these ITSM benefits. One of the opportunities found was the implementation of Cloud computing and DevOps. This shows that the scientific community recognizes that ITSM and DevOps could co-exist in the same environment and improve ITSM implementations, supporting this thesis's objective and motivation.

The complete publication can be found in Appendix D.

CHAPTER 6

Publication nr. #5

The objective of Publication nr. #5 is to establish a background of DevOps implementations and their benefits. This research was done by performing two different SLRs: one to find all the described benefits in the existing literature and another to find DevOps implementations that reported benefits. With this study, it is possible to find empirical evidence of the benefits of DevOps.

This research was vital because it sought to find the benefits of DevOps implementations if they could mitigate the ITSM implementation challenges and opportunities reported in Publication nr. #4, corresponding to Chapter 5.

Thus, with these two publications (#4 and #5), it is possible to establish the link that DevOps could positively impact ITSM implementations. However, the reported DevOps implementation literature did not concern ITSM environments, leading to new research opportunities, which can be found in Chapter 9.

The complete publication can be found in Appendix E.

CHAPTER 7

Publication nr. #6

Similar to Publication nr. #3, in Chapter 4, a case study evaluated the DevOps impact on one of the ITIL processes, in this case, the PM process.

This case study was carried out in a manufacturing industry organization, where the DevOps culture is employed to address the working ways of the IT Teams. Several data collection methods, such as Semi-Structured interviews, documental analysis, observation, and a focus group exercise, were possible in this case study.

In this case study, it was possible to see that practices based on continuous planning and collaboration lead to a better PM lifecycle, quick problem identification, higher-quality root cause analysis, and improved resolution times.

Moreover, this publication identified future work, leaving space for an investigation into the relationship between DevOps and other ITIL processes such as Knowledge Management, RM, and Deployment management.

The complete publication can be found in Appendix F.

CHAPTER 8

Publication nr. #7

In this publication, the objective was to find other publications in the literature that report the implementation of DevOps practices in the ITSM processes and establish what related work was done regarding the objective of this thesis.

An SLR was made to find in the existing literature, and after applying the SLR protocol and filters, nine publications were analyzed in total. From the analysis of these nine publications, several processes that could be impacted by DevOps, such as CM, RM, IM, PM, Event Management (EM), Service Request Management (SRM), and Service Catalog Management (SCM), were visible.

However, these publications didn’t show how DevOps impacts processes or which processes' activities. They provided some adaptations that could be done to processes but not the direct impact on process performance or the people who use the process.

This publication shows there is a possible relationship between DevOps and the ITIL (the most implemented ITSM framework [39]) processes; however, it is not possible to evaluate the impacts that could cause, strengthening the objective of this thesis to provide suggestions on how to solve issues within ITIL implementations by having DevOps practices in place in the organization.

The complete publication can be found in Appendix G.

CHAPTER 9

Publication nr. #8

This article is the last step in compiling publications for this thesis. Since the International Journal of Information Management is still reviewing it, it will not follow the same format as the previous publications in the previous chapters.

In this article, the researchers performed three different case studies to study the impact of implementing DevOps practices. Each case study corresponds to one of the ITIL processes: IM, PM, and CM. These case studies were performed in three organizations with different action industries. The primary data collection method was semi-structured interviews with ten people from each organization. The people had different positions in the organization and different levels of experience.

With this study, it was possible to conclude that DevOps practices could impact the processes differently, accelerating them from end to end. The primary key to this impact was the automation and collaboration practices. Automation results in the automation of the SDLC for faster delivery of fixes for problems and incidents. While collaboration also contributes to the faster delivery of fixes, it also enables more confidence in the software quality and changes impact, accelerating the approvals for the changes to move forward.

DevOps practices also contribute to a steady and healthier system due to the monitoring practices allied with the automation practices for a faster reaction to recover from systems failures or performance degradation.

Below is the article submitted to the International Journal of Information Management. However, due to space limitations, some of the research had to be removed from the submission, which will be added in the next section in 9.1 and 9.2. The following pages will follow the format that was used for the journal submission. While all the thesis document as followed the IEEE citation format, the citation format for International Journal of Information Management was APA, which was the format considered for this chapter.

Abstract

In today's complex environment, organisations face constant market changes and customer feedback. As a result, the IT department has become as critical to the organization's success as any other business unit, mainly because of its essential support.

To improve coordination and collaboration between IT and other departments, the IT division is now viewed as a service provider, delivering IT services to business units, which act as its clients. These services are typically managed using IT Service Management frameworks, such as ITIL, to ensure efficiency and structure.

However, organisations must continuously adapt to market shifts and customer demands to stay competitive. Unfortunately, IT is sometimes perceived as a bottleneck in responding to these needs due to the bureaucratic nature of IT service management processes. In response, Agile Software Development Methodologies were created to address the need for rapid change. Still, frequent shifts in the IT landscape can potentially destabilise existing IT services, requiring more collaboration between those who develop and operate IT Services.

To overcome this challenge, the DevOps culture was designed to bridge the gap between development and operations teams. Its goal is to produce high-quality software without compromising the stability of IT services.

This research will explore how DevOps can be integrated into an ITIL-based environment, focusing on the impact on three key IT service management processes: Incident, Problem, and Change Management. Case studies will be conducted on three organisations implementing DevOps practices and utilising at least one of these processes.

This study demonstrates that DevOps' focus on communication, feedback, and automation has the most significant impact on Incident and Problem Management, whereas its automation practices more directly affect Change Management. Enhancing communication between the various affected parties and automating the delivery process can clearly help resolve issues. Automation minimizes human error, ensuring higher delivery quality and fostering greater confidence in successfully implementing changes.

Keywords: DevOps; ITSM; ITIL; Agile; Benefits; Challenges; Incident Management; Problem Management; Change Management;

1. Introduction

Over the decades, organisations have been changing the way they manage their businesses to overcome internal and external threads as well as from their competitors (Kaplan et al., 2018; Wahyudin et al., 2020). Nowadays, one of the most essential factors in an organisation's ecosystem is the Information Technology (IT) department. The relevance of IT has been growing widely in organisations to achieve their mission and business goals due to the support of business units (Alsolamy et al., 2014).

To be able to support businesses in the market's fast changes and tendencies, the concept of IT services was created (Cannon & Wheeldon, 2007). Furthermore, to evaluate how the IT Services are performing towards the business and organisation objectives, a discipline named IT Service Management (ITSM) was created (Yao & Wang, 2010). The objective of ITSM is to manage the IT Services landscape of the organisation from the moment of the IT Service design until it decommission (E Abreu et al., 2010).

ITSM is implemented using frameworks such as the Information Technology Infrastructure Library (ITIL), Control Objectives for Information and Related Technologies (COBIT), and FitSM. ITIL is seen as the most implemented framework globally. ITIL has a crucial role in managing the organisation's IT Services landscape, from the moment of the IT Service design to its decommission, ensuring efficient and effective service delivery. (Aguiar et al., 2018; Galup et al., 2020).

ITIL newest release, ITIL4, was introduced in 2019. This latest version can be defined as "a set of practices for ITSM that focuses on aligning IT Services with the needs of business." (AXELOS, 2019). The traditional ITIL implementations focused on implementing processes to manage the IT services; however, ITIL4 is focused on practices, extending these implementations for more than processes (Hasibović et al., 2023). Moving to this practice approach enabled ITIL4 not just to overthink the processes' performance but to combine processes, technology and people to guarantee the best performance of the IT Service, focusing on value to its customers (Reiter & Miklosik, 2020).

However, due to the complexity of the processes, some organisations choose not to implement all ITIL processes, like small-scale organisations (Yamami, Ahriz, et al., 2017). Therefore, organisations usually choose to implement Incident Management (IM), Problem Management (PM) and Change Management (CM) (Serrano et al., 2021). These three practices are typically seen working together to maintain stability and the regular operation of the IT Services (Lahtela et al., 2010), ensuring the correct transition from a service design to its operation (Galup et al., 2020).

IM is the practice where the objective is to minimise the negative impact of a service unplanned interruption or reduction in the quality of the service to restore the regular operation of the service as quickly as possible (AXELOS, 2019), preventing economic losses and user dissatisfaction (Lou et al., 2013).

For PM, the objective is to reduce the likelihood and impact of incidents by identifying actual and potential causes of incidents (AXELOS, 2019).

At last, CE can be seen as the last stage where all the changes that impact the IT services are managed (AXELOS, 2019). A change can be made in a system parameterisation, code, or hardware. In the process of this practice, there are a lot of approvals and review activities which can be seen as very exhaustive, causing delays to the delivery of new fixes (incoming from IM and PM) or new functionalities for the IT Service (Kim et al., 2016).

However, while ITIL can bring many benefits, such as more control of the IT environment and risk and cost reduction, it also has challenges. The framework can be seen as bulky and bureaucratic due to its focus on process performance and well-defined roles. This can sometimes hinder agility and responsiveness, which may not be the best option in a market that demands quick responses to customer needs. (Ayat et al., 2009; Sharifi et al., 2008).

Moreover, there is a culture named DevOps that emphasises the collaboration between development and operations teams (de Kort, 2016). The main objective of this culture is to significantly improve the delivery of IT Services through process automation and the adoption of continuous improvement, offering a promising future for ITIL processes. (Jabbari et al., 2016). Due to the collaboration of these two IT teams, DevOps could improve the ITIL processes implementation,

Therefore, this research aims to investigate the relationship between DevOps and ITIL processes to understand the probable impacts of a DevOps implementation on an ITIL environment.

The research methodology used for this research is Multi-Case Study, where the objective is to perform case studies in different organisations, in different ITIL processes, where DevOps practices have also been adopted. These case studies can be seen as empirical evidence of how these two topics can coexist in the organisation's IT ecosystem, showing how they can relate, and which are the benefits and challenges of this DevOps adoption.

This research is organised as it follows: the Theoretical Background section describes the main concepts that frame this research and helps readers understand these concepts and the author who contributed to them; in Related Work, the author examined DevOps case studies to confirm that none or few studies exist relating IM and DevOps domains; then, the Research Methodology identifies how the author will design and validate the case study (CS) methodology; subsequently, in the CS Protocol and Conduct section, the author lists all the data that will be needed to conduct the CS; next, the Analyse the CS Evidences section explains how the author will transform collected data for analysis; lastly, the author presents a set of conclusions about all of the findings discovered during the analysis phase as well as explain why this research will be helpful for academics and professionals.

2. Research Background

Given the objective of this research, this section will explore theoretical concepts about DevOps and ITIL processes to find how these can relate and the probable impacts that could be caused on each of them.

As stated, ITIL is overfocused on process performance and accountability. At the same time, the organisation's market requires agility for fast and quality product deliveries to respond to quick market changes. Organisations must change their mindset and culture to incorporate and accept customer feedback. This is one of the main premises of the DevOps culture: adapting the IT landscape to deliver fast and quality software.

ITIL4 has already started adapting ITIL to be more agile by switching from processes to practices, considering not only process performance but also people and tools as part of the entire practice, and also considering DevOps as part of operating the IT Services. However, ITIL is a set of guidelines for managing the organisation's IT Services and does not show how DevOps practices can be used together with ITIL processes. There is a publication that reviews how DevOps has been seen from an ITSM perspective and how organisations can adapt the two concepts to work together (Faustino et al., 2023).

Due to the few publications relating the two concepts, the authors found that this relation is in an early stage, creating the opportunity to build new research. This underscores the importance of further exploration in this area.

To better understand DevOps and ITSM, the following sections will describe these two disciplines' main concepts, practices, and processes, providing enough background to understand how they can be applied together in the same environment.

2.1.ITIL

ITSM can be seen as a holistic approach that is responsible for all the activities of creating, implementing, supporting, and managing IT services. ITSM has been implemented through several frameworks, such as COBIT, ITIL and FitSM (Sarwar et al., 2023). Moreover, there is a framework which stands out, in terms of implementation, regarding the other frameworks, which is ITIL (Yamami et al., 2017).

ITSM is a process-oriented approach that has several processes to improve IT efficiency by measuring IT services' performance through the processes. These processes guarantee the IT Services quality, transparency of the IT processes, and the IT changes delivered to its users (Lema et al., 2015).

ITSM is "a *methodical approach to the management of IT services – from design, implementation, operation to continual improvement. It not only focuses on the technical aspects of IT but also allows the alignment of services and functions provided by IT within the organisation.*" (Marrone & Kolbe, 2011, p. 6).

This study will focus on ITIL since it is one of the most globally implemented ITSM frameworks (Galup et al., 2020), where a new version, ITIL4, was released in 2019 (Guilfoos & Triplett, 2022).

Older versions of ITIL, such as ITIL 2011, had an emphasis on the IT processes with well-defined inputs, outputs and roles, while ITIL4 extends the framework to be modernised and flexible (Pratama & Umaroh, 2023). Practices are sets of organisational resources designed for performing work or accomplishing an objective, being more flexible and less prescriptive than processes.

ITIL 2011 aimed to manage IT Services between five stages of the IT Service lifecycle: Service Strategy, Service Design, Service Transition, Service Operation, and continuous service Improvement. In ITIL4, the main focus is the value of the IT service. For this, ITIL4 introduces the Service Value System (SVS), a holistic approach where several interconnected activities create and deliver value, called a service value chain. This service value chain is crucial for ITIL4 since it manages the IT Service from its creation until its delivery, constantly monitoring the value it creates to know when to improve.

The chosen processes for this research were Incident Management (IM), Problem Management (PM), and Change Management (CM) since these have been more adopted globally as stated previously, and also since these were identified as most probably impacted processes according to Faustino et al. (2023). Moreover, these processes are typically seen by as processes where developers and operators are more likely to work together, showing how the DevOps practices have impacted the processes, which will be detailed in the next section.

2.1.1. IM Process

IM plays a crucial part in ITSM, being one of the fundamental processes of ITIL (Latrache et al., 2015).

An incident is a non-planned interruption or performance degradation of an IT Service, which also applies to any component that supports the IT service. Since the IT services should support the business processes of the organisation, it is possible to state that an incident will cause a business interruption (Bartolini et al., 2006; Richard et al., 2019).The objective of the IM practice is to manage these incidents, so it minimises the service disruptions and outages to cause less impact on the organisation's business units (Steinberg, 2011).

An organisation's main objective is to create profit, so if the business has some interruption on the IT Services that support the business applications, it will create economic losses, which should be avoided or mitigated as soon as possible (Lou et al., 2013). These kinds of business interruptions are not only seen as a negative impact from an economic perspective. However, they could also result in a lack of confidence from the organisation’s customers, even internal customers, showing that the IT services are unreliable for the business.

As stated before, the objective of the IM process is to bring the IT Service to its regular operation and performance, so when a complaint or a call reaches the service desk, it will check if it is a known issue and if there is already any solution for the user issue. If not, the service desk will try to find a workaround to solve the incident. If there is no workaround, the incident will be sent to another support group, but if successful, the incident will be closed, and a problem will be raised to solve the root cause. From this moment, PM urges to solve this root cause, where the next section will provide some theoretical background about it.

2.1.2. PM Process

In ITIL, a problem is seen as the root cause of the incident. Therefore, the PM practice is said to be used to solve the root cause of one or more incidents, minimising the impacts that these incidents can have on the organisation (Cannon & Wheeldon, 2007). The problem manager is responsible for thoroughly analysing previous incidents, identifying the root cause, and implementing measures to prevent these issues from reoccurring.

Problem Management (PM) is not just about solving issues but also about preventing them from occurring (Kush, 2013). This proactive approach, which includes continuous improvement, minimises the impact on business units and ensures excellent operational stability. This focus on prevention is a mindset highly valued by organisations, reassuring that issues will not reoccur.

As seen in the implementation activity for the PM practice, some solutions require a change, creating a connection to the CM process.

The following section will provide a theoretical background for the CM process.

2.1.3. CM Process

CM is not just about integrating changes in IT Services but about fostering a culture of continuous improvement within the organisation (AXELOS, 2019). By extending the change management process from ITIL 2011, CM can transform the organisation's approach to change, engage stakeholders, improve communication, and implement strategies to address potential challenges and resistance to change.

Globally, most organisations use three change management processes: normal change, standard change, and emergency change.

Below is a description of each type of change, adapted from (Rance, 2011):

- **Standard** - This is used for pre-authorised changes that have a negligible impact on the IT service and follow a well-known process.
- **Emergency** - For example, a change needs to be implemented immediately to solve a major incident that seriously impacts an IT service. Usually assessed by the Emergency Change Advisory Board (ECAB)
- **Normal** - A change that requires approval from the Change Advisory Board (CAB) occurs when a significant change can impact an IT service, business processes, or infrastructure.

Since standard changes are pre-authorised changes that are implemented by following a known procedure, it does not require so many approvals and assessments as normal changes (Kaiser, 2018) an example of a standard change could be resetting a user's password. This change needs to be made to the IT Service but is low risk and impact, so the standard change process should be followed.

However, from a high-level perspective, these processes look bulky and bureaucratic, where they have a lot of different people and roles to secure different activities (Kaiser, 2018; Sharifi et al., 2008).

From nowadays perspective, this does not look realistic due to the constant changes that the market faces. Changes and adaptations to IT services must be fast enough for organisations to adapt to the market changes to face their competition and customer demands (Soni, 2016). To face this difficulty in delivering new solutions, fixes, and adaptations to IT services, a culture named DevOps has urged. The main premise of this culture is to deliver software quickly with high quality by automating several manual tasks from the Software Development Lifecycle (SDLC) and by joining IT Development and operations to work together to develop and maintain software (Cuppett, 2016; Sharma & Coyne, 2014).

2.2.DevOps

The Agile Infrastructure Conference first mentioned the DevOps culture (Lwakature, 2017). The main goal of DevOps culture is to bridge the Operations and Development IT teams to work together so both teams can help each other from the starting point of software development to high product quality and stable software (Riungu-Kalliosaari et al., 2016; Silva et al., 2018). DevOps also follows the agile software development practices from the Agile Manifesto (Beck et al., 2001), focusing on individuals and their skills and embracing the change to correspond to customer feedback, even if this could impact process performance. This way, the developed product will be more valuable to the customer since it will correspond to their needs.

These two IT teams have two different points of view towards software changes (Waschke, 2015). Developers want to deliver new and improved features, while operations want to maintain the stability of IT services or IT systems; therefore, they do not want developers to deliver new features since they fear they will compromise the stability. This generates a blame game between these two roles where the operators blame the developers for compromising the system stability, while the developers blame the operators for not maintaining the operability of the system to receive new features (Hussaini, 2015).

When these two IT teams collaborate to develop new features, operators can be involved from the beginning of the SDLC and help the developers deliver quality and stable software by providing the necessary infrastructure and tools to ensure the software will run with the desired performance (Hemon et al., 2020).

To be able to accomplish its objectives, DevOps employs several practices, as it is seen in Table 1, adapted from (Faustino et al., 2018; Jabbari et al., 2016). There were considered the most low-level practices from Jabbari et al. (2016), so the interviews can have more options to choose from the practices. For example, both Automated Deployment and Continuous Deployment are considered a DevOps practice Jabbari et al. (2016), but an organisation can implement automated deployment while not considering continuous deployment. This shows that both organisations can apply DevOps practices but with a different level of maturity (Teixeira et al, 2020).

Table 1 - DevOps Practices

ID	Practice name	Description
P1	Continuous Planning	Continuous planning of products/features in several deliveries to allow the incorporation of customer/business feedback.
P2	Feedback Loops Between Dev and Ops	Frequent checkpoints between Operators and Developers to discuss deliveries and application pain points.
P3	Continuous Monitoring	An approach where an organisation constantly monitors its IT systems and networks to detect security threats, performance issues, or non-compliance problems in an automated manner.
P4	Measure Performance Metrics (in CI, Test & Ops)	Define metrics to measure the system's performance while developing, testing, and operating.
P5	Automated Feedback for Performance Models and Performance Predictions	Creation of performance feedback reports based on models and predictions.
P6	Application Monitoring	Monitoring of applications to detect performance/wrong behaviour from the applications.
P7	Automated Dashboards	Build dashboards to provide developers and operators with information about the applications' status and behaviours so that the correct actions can be taken.
P8	Continuous Integration	It is a practice that encourages developers to check their code as much as possible so the other developers can always work on the latest version. This practice also runs a build and tests the code to ensure the new code will not impact the existing software.
P9	Prototyping Application	Build a prototype of the feature or application to perform demos for the customers or business users.
P10	Continuous Deployment	Once the automatic tests pass the code or artefact, automated deployment will be performed for the following environment.
P11	Automated Deployment	Practice that allows to schedule application deployments to be executed automatically.
P12	Continuous Delivery	This practice ensures that the software is always ready to be deployed. If a package is created after the build and automatic testing, that package is stored as an artefact on a repository and can be deployed.
P13	Continuous Testing	Automatic tests are triggered after each build of the code. It is used together with Continuous Integration.
P14	Automated Testing	Tests that are executed automatically produce a report of the test status.
P15	Process Standardisation	Standardisation of processes is used to unify the processes inside the organisation's IT department. This will allow the same modus operandi between all the IT teams, facilitating the personal rotation between IT teams.
P16	Infrastructure as Code	This practice allows the coding of the infrastructure for the application and the loading of that code to create or update the existing infrastructure. This allows the quick creation of environments and the scale in/out of server resources based on demand.
P17	Stakeholder Participation	Stakeholder participation requires all the key stakeholders to participate in all the software development or operations phases. This allows more engagement from the stakeholders to make decisions on the right timing for the application's developers and operators.

Organisations worldwide have reported the benefits of DevOps implementations, such as faster time to market, faster and better feedback and increased team performance and customer satisfaction (Faustino et al., 2022).

Moreover, ITSM has its challenges (Serrano et al., 2021). One challenge is the time consumption of the process assessment and framework complexity. This shows how bureaucracy can impact the deliveries from the IT teams, as stated before about the ITSM processes. Based on the DevOps benefits and the premises of DevOps by automating manual tasks, DevOps can help improve the ITSM processes' performance.

3. Research Methodology

Since the research in the domain of DevOps application in the ITSM processes is in its very early stages (Faustino et al., 2023), the nature of this research is exploratory due to the limited literature about the subject. Exploratory research is meant to start a study on an observed phenomenon without prior (or few) works on a specific context (Zaidah, 2007). Moreover, a Case Study (CS) is built around a question (Thomas, 2016), which in this case is, “How do DevOps impact ITSM processes?”.

The question is the final objective that the CS needs to answer, but the CS also should be defined by its purpose, approach and process (Thomas, 2016). Moreover, the subject may lead to three different types of CSs: unique or outlier (when the researcher tries to study a phenomenon out of the norm), a key case (when the researcher is studying a phenomenon that happens a lot), and a local knowledge case (where the researcher is investigating something familiar to him) (Thomas, 2016). This CS is classified as a local knowledge case since the researchers are familiar with DevOps practices and ITSM.

Researchers can adopt either a single-case or a multiple-case approach. A single-case approach should be adopted when the event that is supposed to be studied is limited by a single occurrence, or the study will only target a single unit of analysis (Yin, 2009; Zaidah, 2007). Multiple case studies are used on real-life events where numerous, easy-to-replicate sources of evidence exist (Zaidah, 2007). Since for this research, the objective is to perform a study on how DevOps impacts three different processes, the approach will be a multi-case study, where these three case studies will be done on three different organisations where the processes where both the DevOps processes and one of the processes are applied.

. This study will focus on CM, PM and IM, as identified in Faustino et al. (2023) as possible candidates to be implemented with DevOps practices and due to the researchers' experience with these processes. Where each case study will be applied to each process.

Since this research intends to study the possible influence of DevOps practices on the ITSM processes activities grounded on the experience of several organisations, it must be considered a retrospective CS.

Table 2 — Research Questions

Research Question ID	Description
RQ1	Which DevOps Practices can impact each process?
RQ2	What are the benefits and challenges of implementing DevOps practices in the processes?
RQ3	How has DevOps improved the processes?

Researchers should explain or explore a phenomenon that leads to the following purposes: intrinsic, instrumental, evaluative, explanatory, and exploratory (Thomas, 2016). As stated before, this research will be exploratory; where for this type of approach, Thomas also suggests the following: testing a theory, building a theory, drawing a picture, descriptive, interpretative, and experimental (Thomas, 2016). As previously stated, no literature has investigated the relationship between DevOps and the ITIL processes; therefore, this research aims to build a theory.

Some authors provide insights into the structure of a CS (Tellis, 1997). Table 3 shows the approach that will be followed in this research (Yin, 2009).

Table 3 — Case Study Stages

Stage	Stage Description
Design the Case Study Protocol	This stage comprises two minor stages: determining the required skills and developing and reviewing the protocol. The latter involves extensive reading about the topic to create draft questions. Yin suggests that the researchers should be good listeners who can interpret the responses and create draft questions.
Conduct the Case Study	Preparation of the data collection, distribution of the surveys and conducting interviews.
Analyse Case Study Evidence	An analytical strategy should be employed to evaluate the data gathered in the previous stages of the research.
Develop Conclusions	Develop conclusions regarding the data analysis made in the previous stages to establish a bridge between the researcher and the user and explain the benefits or problems found during the research.

Source: Adapted from (Tellis, 1997; Yin, 2009)

To build a CS, Figure 1 describes the different classifications of our research according to Thomas’ framework and guidelines. This helps to understand how this research maps with Thomas' framework.

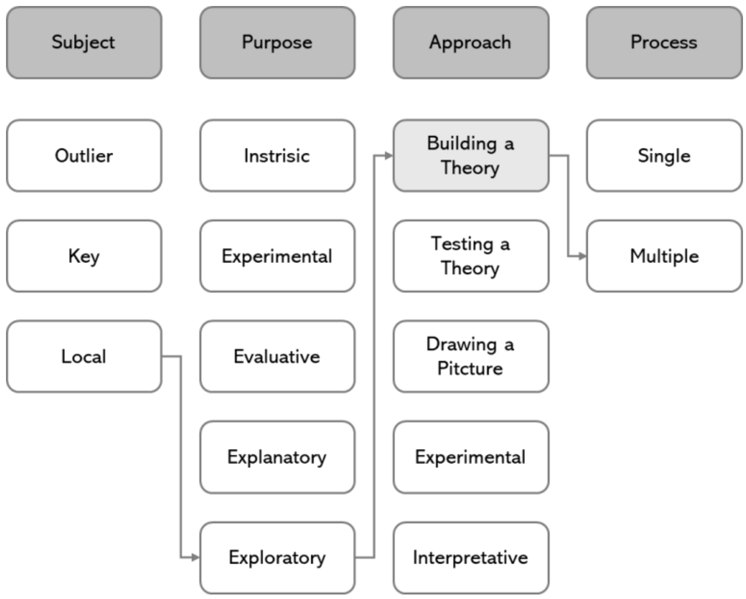


Figure 2 — Path for the CS, adapted from (Thomas, 2016).

Since this is a retrospective CS, the primary data collection procedure is interviews of those who experienced the study phenomenon (Thomas, 2016).

The interview type that was used is Semi-Structure interviews, which are used when one needs to gather more detailed information by giving the interviewees the liberty to express their opinions (Miles & Gilbert, 2005).

In the following sub-sections, the multiple case studies will be presented.

4. Case Study Design Protocol

As previously stated, this research will be done by performing three case studies in three different organisations. The first case study regarding IM was performed over four months, from September until December 2023, in a multi-national IT Consultancy company in Financial Services (FS) based in Lisbon, Portugal. The analysis was based on an Application Management services team for an insurance company, focused on the corrective maintenance of a core insurance application. Even though the team is known to be bi-lingual, the interviews were conducted in Portuguese.

Regarding the CM case study, the research was conducted over two months (August and September 2022) at a multinational fintech company, with one team and a sample of 10 interviewees, using a qualitative approach. The team's average professional experience is 8.3 years, and experience with DevOps is about 0.8 years. It is important to note that for the CM experience, most participants were neither fully aware of the concept nor had knowledge about the process as a whole. So, to address this knowledge gap, the interviewer explained the main ideas behind DevOps in addition to CM and its phases during the interviewing process.

The PM case study was performed in a German multinational conglomerate company. On average, participants have four years of experience in their roles while having close to 10 years of work experience in IT. Most have exercised DevOps and PM practices in at least two previous organisations. Three separate teams are represented in this exercise: a Service Management team responsible for overseeing the implementation of PM and other ITIL processes in the organisation (participants PM-A, PM-G, PM-J), a Service Delivery team managing business interactions with customers and end users (participants PM-C, PM-E, PM-I), and a development team responsible for the maintenance and continuous improvement of Service Management tools utilised in the organisation (participants PM-B, PM-D, PM-F, PM-H).

To ensure that the recorded interviews were understood correctly, each one was reviewed, and the relevant topics were transcribed for written support using the same terminology observed in the recordings, with the authorisation of all participants.

The interviewees' structure is summarised in Table 4, detailing each interviewee's role and professional experience.

Table 4 – Interviewee Details for Case Studies

IM Case Study			
Interviewee	**Position**	**Experience in IT**	**Experience in DevOps**
IM-A	*Experienced Developer*	*3*	*3*
IM-B	*Experienced Developer*	*3*	*3*
IM-C	*Team Leader*	*7*	*6*
IM-D	*Team Leader*	*8*	*6*
IM-E	*Team Leader*	*6*	*6*
IM-F	*Team Leader*	*6*	*6*
IM-G	*Manager*	*10*	*7*
IM-H	*Manager*	*10*	*7*
IM-I	*Manager*	*9*	*6*
IM-J	*Manager*	*15*	*6*
Average	**-**	*7.7*	*5.6*
CM Case Study			
Interviewee	**Position**	**Experience**	**Experience in DevOps**
CM-A	*Quality and Assurance (QA) Engineer*	*15*	*0*
CM-B	*Quality and Assurance (QA) Engineer*	*12*	*2*
CM-C	*Software Engineer*	*1*	*1*
CM-D	*Software Engineer*	*10*	*2*
CM-E	*Software Engineer*	*3*	*1*
CM-F	*Software Engineer*	*7*	*1*
CM-G	*Software Engineer*	*2*	*1*
CM-H	*Technical Writer*	*7*	*0*
CM-I	*User Experience (UX) Designer*	*6*	*0*
CM-J	*Product Manager*	*20*	*2*
Average	*-*	*8.3*	*0.8*
PM Case Study			
Interviewee	**Position**	**Experience**	**Experience in DevOps**
PM-A	*Process Manager*	*5*	*1*
PM-B	*Developer*	*10*	*9*
PM-C	*Service Manager*	*14*	*3*
PM-D	*Developer*	*5*	*4*
PM-E	*Process Manager*	*13*	*2*
PM-F	*Developer Team Lead*	*12*	*12*
PM-G	*Process Manager*	*10*	*1*
PM-H	*Developer*	*12*	*4*
PM-I	*Process Manager*	*10*	*3*
PM-J	*Process Manager*	*6*	*1*
Average	*-*	*9.7*	*4*

5. Case Study Conduct and Evidence Analysis

The following subsections will demonstrate how the interviewees were conducted and how the data collected was prepared, after this each subsection will have its own conclusion about the data. Also, each subsection will correspond to the research questions that are in Table 2.

5.1. Interview Analysis on DevOps Impact on Process Activities

Table 5 shows the participants' matches between the DevOps practices and IM, CM and PM process activities. Please refer to the colour legend to understand the meaning of each colour.

Almost all the DevOps practices identified in Table 1 were matched with at least one activity regarding the IM process. The DevOps practices that were seen to have the most impact were Feedback Loops Between Dev and Ops, Application Monitoring, and Continuous Integration. Analysing from the point of view of the activities, the activities with more matches were Resolution and Recovery, Investigation and Diagnosis and Detection and Recording. This is a good indicator for Resolution and Recovery and Investigation and Diagnosis since the IM process objective is to solve and recover as fast as possible.

Table 6 shows the interviewees' comments about each practice and match. According to the colour legend of Table 5, only the activities with more matches per DevOps practice were considered. Also, only the practices with at least one grey match were considered.

Due to the context of this team, the IM process is not followed 100% according to ITIL, where the objective is to resolve the incident as soon as possible. (AXELOS, 2019). They work using an agile approach based on sprints, where the stakeholders choose the priorities of the incidents or other developments that should be done. So, this team relies on the stakeholders to know what to work on, always based on the business input of what is more important.

Moreover, looking at Tables 5 and 6, they agree that DevOps practices accelerate incident resolution through Continuous Integration, Continuous Delivery and Deployment practices. However, an input shows a different perspective on Continuous Delivery. An interviewee says that increasing deliveries can negatively impact the investigation of incidents since there will be more changes to check what code could cause an incident.

Another aspect that stood out was the detection of incidents. The interviewees claim that incident detection is essential to the application's stability, so more monitoring and new metrics are always welcome so incidents can be detected earlier, reducing the possible impacts on the business units.

Regarding the CM process, nine practices (out of 17 from Table 1) matched at least one activity of the process for the Normal Change process.

The top 3 practices that had more matches were the practices of Continuous Delivery, Automated Deployment and Continuous Deployment. The practices had more impact on the final activities of the process such as Authorise Change Deployment, Coordinate Change Deployment and Review and Close Record.

Moreover, from the point of view of the activities, the activities that matched were Coordinate Change Deployment, Review RFC, and Coordinate Change and Build Test. Regarding Coordinate Change deployment, it is possible to see more matches of Continuous Deployment, Automated Deployment and Continuous Delivery, similar to what has been stated before. Continuous delivery, continuous testing, automated testing, and process standardisation are used for Review RFC activity. For the Coordinate Change and Build Test, more matches can be seen in Continuous Integration, Continuous Delivery, Continuous Testing, and Automated Testing.

Table 7 contains the interviewees' feedback regarding the matches between the DevOps practices and Normal Change process activities.

Looking at Table 5 for the Standard Change activities, the same practices identified as a match in the Normal Change process were also identified in the Standard Change process. It is also possible to conclude that the practices that hit more matches are Continuous Delivery, Stakeholder Participation, and Automated Testing, while the activities with more matches were Assess and Evaluate RFC, Coordinate Change Implementation, and Authorise and Schedule Change.

Table 5 – Matches Between DevOps Practices and Process Activities (Case Studies)

Process / Activity		Practice ID																	
		P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	Total of Matches
IM Process	Detection and Recording	-	3	4	1	-	5	1	-	-	-	-	-	1	1	2	-	2	20
	Classification and Initial Support	3	2	2	-	-	-	2	-	-	1	-	-	-	-	3	-	3	16
	Investigation and Diagnosis	1	7	-	-	-	1	-	2	1	1	1	1	-	-	1	4	1	21
	Resolution and Recovery	2	5	-	-	-	-	-	7	2	4	7	4	1	4	-	2	1	39
	Closure	1	1	-	-	-	-	-	3	-	2	3	4	-	-	2	-	-	16
	Monitor and Tracking	2	1	3	3	-	6	1	-	-	-	-	-	-	-	1	-	-	17
	Total of Matches	9	19	9	4	0	12	4	12	3	8	11	9	2	5	9	6	7	-
Normal Change Process	Create RFC	3	-	-	-	-	-	-	5	-	4	4	4	3	2	4	-	5	34
	Record RFC	1	-	-	-	-	-	-	4	-	3	3	3	2	2	3	-	2	23
	Review RFC	3	-	-	-	-	-	-	3	-	3	3	5	6	5	5	-	3	36
	Assess and Evaluate Change	8	-	-	-	-	-	-	3	-	3	3	4	3	2	3	-	2	31
	Authorise Build and Test	1	-	-	-	-	-	-	5	-	3	3	4	3	4	2	-	-	25
	Coordinate Build and Test	2	-	-	-	-	-	-	5	-	4	4	5	5	5	4	-	1	35
	Authorise Change Deployment	-	-	-	-	-	-	-	4	-	5	5	4	1	1	2	-	1	23
	Coordinate Change Deployment	3	-	-	-	-	-	-	4	-	7	8	6	2	2	3	-	2	37
	Review and Close	1	-	-	-	-	-	-	1	-	3	3	5	2	1	4	-	4	24
	Total of Matches	22	0	0	0	0	0	0	34	0	35	36	40	27	24	30	0	20	-
Standard Change Process	Review RFC	5	-	-	-	-	-	-	2	-	2	1	5	2	3	5	-	5	30
	Assess and Evaluate RFC	5	-	-	-	-	-	-	4	-	4	3	6	5	5	4	-	5	41
	Authorise and Schedule Change	4	-	-	-	-	-	-	6	-	3	2	5	5	5	2	-	2	34
	Coordinate Change Implementation	3	-	-	-	-	-	-	5	-	4	3	5	5	5	4	-	5	39
	Review and Close Record	3	-	-	-	-	-	-	2	-	2	1	4	3	3	4	-	6	28
	Total of Matches	20	0	0	0	0	0	0	19	0	15	10	25	20	21	19	0	23	-
Emergency Change Process	Record RFC	4	-	-	-	-	-	-	3		3	3	4	1	1	-	2	5	26
	Assess Change	6	-	-	-	-	-	-	4		4	4	5	3	4	-	1	3	34
	Change Approval	3	-	-	-	-	-	-	5		3	3	4	4	6	-	2	3	33
	Review Approval	2	-	-	-	-	-	-	3		5	4	3	-	1	-	1	2	21
	Implement Change	3	-	-	-	-	-	-	4		6	7	5	2	3	-	2	3	35
	Review Change	3	-	-	-	-	-	-	4		6	7	5	2	3	-	2	3	35
	Total of Matches	21	0	0	0	0	0	0	23	0	27	28	26	12	18	0	10	19	-
PM Process	Problem Detection	4	8	-	-	-	9	-	3	2	-	2	-	-	3	8	-	2	41
	Problem Logging	4	5	-	-	-	3	-	3	2	-	1	-	-	2	5	-	3	28
	Problem Investigation and Diagnosis	5	7	-	-	-	6	-	3	2	-	1	-	-	3	4	-	3	34
	Known Error Management	5	6	-	-	-	3	-	4	2	-	1	-	-	2	5	-	4	32
	Problem Resolution	7	6	-	-	-	4	-	5	3	-	2	-	-	3	6	-	3	39
	Problem Closure	4	7	-	-	-	3	-	2	2	-	2	-	-	2	5	-	3	48
	Total of Matches	29	39	0	0	0	28	0	20	13	0	9	0	0	15	33	0	18	-

Colour Legend:
White – 1 to 2 matches
Light Grey – 3 to 4 matches
Medium Grey – 5 to 6 matches
Dark Grey – More than seven matches

Table 6 - Insight on DevOps & IM process (Case Studies)

Practice ID	IM Activity	Comment
P1	Classification and Initial Support	"By continuously planning, it will be possible to order the backlog correctly and know what is causing more pain in the users at the moment."
P2	Investigation and Diagnosis	"Close collaboration between dev and ops will allow them to share knowledge to diagnose the incident."
	Resolution and Recovery	"The coordination between devs and ops can result in anticipating if there are issues with a build of a package due to a resolution of an incident; this could help to improve the resolution."
P3	Detection and Recording	"This practice could enable the creating of incidents, and it can attribute the priority and trigger other actions to support the application."
	Monitor and Tracking	"Monitoring the system continuously will identify strange behaviors on the application, therefore identifying incidents. This will also help to identify if a fix for an incident will be working after applying"
P4	Monitor and Tracking	"We can have an overview of the application behaviour of the performance to tell if the incident is solved."
P6	Detection and Recording	"This practice allows for the creation of incidents automatically as soon as a usual behaviour is detected."
	Monitor and Tracking	"Allows to see the behaviour after a resolution being applied to know if the incident is resolved."
P8	Resolution and Recovery	"Continuous Integration allows a faster resolution by ensuring the code is being integrated more often and allowing to be sure that will follow the quality gates." "By integrating every code, we can apply the resolutions faster, hence closing the incident faster."
P10, P11	Resolution and Recovery	"It will allow progressing with the resolution between the environments faster until it reaches production." "Allows faster deployment of new logging to environments to investigate and find the final solution to the issue."
P12	Resolution and Recovery	"Saves time to deliver the new solution into the final environment, ensuring the code is deployable without any issue."
	Closure	"It could harm the investigation; since DevOps allows constant delivery and constant deployments, we can have five deployments (one per day), and an incident is created after seven days. We may need to check all the code delivered in those five deployments."
P14	Resolution and Recovery	"Knowing the correct behaviour, we know what to test to apply TDD." "Saves time to apply the correct tests after a solution is identified."
P15	Classification and Initial Support	"If there is a standard way to detect and record incidents, it will help the end users understand the process. The same applied to the classification so the incidents can be classified correctly."
P16	Investigation and Diagnosis	"Having the infrastructure in code files will help create new environments to help diagnose and resolve the incidents."
P17	Classification and Initial Support	"Depending on the stakeholder, they should be the ones interested in the application stability, so they should be able to detect and escalate incidents so they could be fixed according to the impacts and priority."

The stakeholders continue to be a crucial part of the process since they should know the impact of these changes, even if these types of changes should be classified as low risk. They are seen as the primary approvers of the change to be taken into place since they know the implementation's impacts. Also, they should be accountable for effective communication with all the teams and applications that the change implementation could impact. Table 8 describes the interviewees' feedback about the matches between the Standard Change process activities and the DevOps Practices.

When comparing the Emergency Change process with the other two change processes, it is possible to see that the Process Standarisation practice is not listed, being replaced by Infrastructure as Code (IaC). Many emergency changes are raised due to issues in the systems' infrastructure. This practice can help to solve those issues relatively quickly.

Other practices that stand out are Continuous Deployment and Automated Deployment. These practices impact the last activities of the process to deliver the solution to the emergency change faster and restore the normal operation of IT Services. This can also be seen since the last two activities have more matches, with 35 matches. An activity that is also seen as highly impacted is Assess Change. Due to the Continuous Planning of the teams, it is easier to prioritise what can be seen as an emergency or not, helping to evaluate if the change is an emergency. Table 9 shows the interviewees' insights into Emergency Change and DevOps practices.

The interviewees were asked to assess the relevance of DevOps practices across PM activities by having each select one of the following options for each combination: 1, meaning low relevance; 2, meaning relevant; and 3, meaning high relevance. Answers were compiled into Table 5 by adding the values entered for each cell. Opinions and justifications provided by interviewees as they responded to this assessment are found in Table 14.

Data resulting from semi-structured interviews indicates that 9 out of the 17 contemplated DevOps practices are, in terms of their application, significant in at least one stage of the Problem lifecycle. The DevOps practices of Continuous Planning, Feedback Loops Between Dev and Ops, Automated Monitoring, and Stakeholder Participation stood out and were highly significant to the PM process. Considering captured insight, this results from the improved planning and collaboration these practices tend to enable, allowing development and operations teams to work more closely together. The practices of Application Prototyping, Deployment and Test Automation were considered less significant, the latter neither practiced nor known among selected interviewees.

Table 5 shows that the practices with the higher impact on the PM process are Dev & Ops Feedback Loops, Stakeholder Participation and Automated Monitoring. Discussing Dev & Ops Feedback Loops, the higher impact is in Problem Detection and Problem Investigation and Diagnosis.

Table 14 means that when operators know the developments, it will be easier to alert them about risks and issues, which will be necessary for proactively creating problems. Regarding Problem Investigation and Diagnosis, the interviewees stand out in the collaboration between developers and operators to reach the root cause analysis of the problems.

Stakeholder Participation also significantly impacted Problem Detection and Resolution. The interviewees believe stakeholders are the ones in the field and the best at understanding user pains when using IT services or applications. Thus, they are the key to finding issues and suggesting the correct behaviour to benefit the end users.

Automated monitoring, like Dev & Ops Feedback Loops, significantly impacts problem detection, investigation, and diagnosis. It can be seen as critical in identifying problems since it indicates the IT service or application's behaviour. Depending on the indicator, the monitor can also help with the root cause analysis by identifying the process or part of the IT Service or application that is failing or degrading.

From the PM activities perspective, the activities most impacted by the DevOps practices are Problem Resolution, Problem Investigation and Diagnosis and Problem Detection. Problem Detection was discussed in the last paragraphs regarding Dev & Ops Feedback Loops, Automated Monitoring and Stakeholder Participation. However, continuous planning also impacts it. When planning meetings and discussing the behaviours of the IT Services and applications, problems can be created earlier, and a more resilient application can be created earlier.

Problem Investigation and Diagnosis, likewise Problem Resolution, are impacted by the same DevOps practices. Continuous Planning stands for better planning and prioritisation of the problems to be solved.

Problem Resolution, like the other two activities, the same practices, including continuous planning, were classified as having more impact. Table 10 provides the interviewees insights about the PM process and the DevOps practices

Having analysed the three processes, one can see that DevOps can impact the ITIL processes differently, but always with the focus on improving or accelerating some manual tasks such as assessments and approvals. In more detail, we can see that Continuous Planning is causing more impact on the PM and CM process, where this practice can help to plan changes and their deliverables on the correct timing, likewise, the solutions for the problems existing in the IT Services depending on the impact. There is also the relationship to deliver the solutions for the existing problems; a change is required. Feedback loops between Dev and Ops are seen as more impactful in PM and IM.

This is seen as a practice that helps operators quickly solve issues in IT services due to the collaboration between developers and operators. Likewise, application monitoring is seen as impactful in PM and IM. This practice is essential in identifying issues as early as possible, even in applying a quick fix or a workaround (like in the IM process) or a solution for a root cause of several issues (like in the PM process). Nevertheless, the focus is to stop the damaging business impact.

However, two practices were identified only in the IM practice: Continuous Monitoring, Measuring Performance Metrics, and Automated Dashboards. The interviewees do not apply these practices but recognise their potential to anticipate issues and take quick actions before they affect the business work. Also, having performance metrics at several levels of the SLDC can help the developers anticipate if their new developments will impact the current application behaviour, anticipating a possible issue with the software in production.

Table 7 - Insight on DevOps & Normal Change process (Case Studies)

Practice ID	Normal Change Activity	Comment
P1	Assess and Evaluate Change	"As we continuously plan, we evaluate why the change is necessary and coordinate when it should be deployed." "Evaluate change because it is when we can verify if it is going to bring value to our product."
P8	Create RFC	"A change may need authorisation, and coordination is needed for this change to be deployed."
	Authorise Change and Build Test	"Because it cannot change the functioning without being evaluated, accepted and reviewed. Regarding the pipeline, they must guarantee that the change's entry will not harm the system. The whole process of testing pipelines and how things are working, and there is a whole process where this change will be integrated, is important to be executed."
	Coordinate Change and Build Test	
P10, P11	Authorise Change Deployment	"One of the most important processes where it was more bureaucratic, e.g. for each new deploy was extremely bureaucratic and had a set of requests and documents to be done, tests after deploy, rollback mechanisms and had to be validated."
	Coordinate Change Deployment	
P12	Review RFC	"Because it is about delivering, we need to review it as we release software. That is, review and coordinate such changes."
	Coordinate Change Deployment	
	Review and Close Record	
P13, P14	Review RFC	"Because you must verify what exists, we need to have authorisation; we just need to coordinate what needs to be deployed. You do not need something to be Authorised; it just needs to be coordinated and not Authorised." "To get the requirement tested accordingly, coordinate its change deployment."
	Coordinate Change Build and Test	
P15	Review RFC	"Everything is a process in place and must be taken into consideration. "
P16	Create RFC	"Because it is an important component at the process's beginning and end." "It is important to have them created and communicated. Sometimes it seems too informal and needs to be more effectively communicated."

Table 8 - Insight on DevOps & Standard Change process (Case Studies)

Practice ID	Standard Change Activity	Comment
P1	Review RFC	"As it is already part of the plan, we can review it and then coordinate when it will be deployed and ensure it is aligned with the product planning.
	Assess and Evaluate RFC	"Because we can see a board and do something as a team, we can define our priority as an autonomous team."
P8	Authorise and Schedule Change	"Even being standard changes, they need to go through the entire process as it has the most delicate parts of the SLDC, and we need to ensure quality upon deployment."
	Coordinate Change Implementation	
P10, P11	Assess and Evaluate RFC	
	Coordinate Change Implementation	"Because we do not need authorisations for something that is not urgent and should not be escalated."
P12	Assess and Evaluate RFC	
P13, P14	Assess and Evaluate RFC	"Because we can review the request (for a change) without needing a previous review." "only for scenarios where we need to cover it with tests besides reviewing this change and closing it."
	Authorise and Schedule Change	
	Coordinate Change Implementation	
P15	Review RFC	"It allows standard changes not to impact the planning; if we follow the process, we will not have disturbances on development, and then we will have continuous development and will not affect our planning. Not necessary on authorisation but coordination."

Table 9 - Insight on DevOps & Emergency Change process (Case Studies)

<table>
<tr><th>Practice ID</th><th>Emergency Change Activity</th><th>Comment</th></tr>
<tr><td>P1</td><td>Assess Change</td><td>“Even being an emergency, we need to ensure quality and not have the problem again as we want to ensure confidence on the part of our end users (clients) by escalating the priority. Detailed attention to avoid errors.”</td></tr>
<tr><td>P8</td><td>Change Approval</td><td rowspan="3">“The deliverable also guarantees that will ensure the change was effective and, in an emergency, it must be validated thoroughly, and all steps are required.”
“Because we need authorisation (only if the upper level is too much involved) for something urgent and needs to be escalated.”</td></tr>
<tr><td rowspan="2">P10, P11, P12</td><td>Implement Change</td></tr>
<tr><td>Review Change</td></tr>
<tr><td>P13, P14</td><td>Change Approval</td><td>“Even being an emergency, we need some confidence level in the change being performed and detect regressions (even if it is not an emergency).”</td></tr>
<tr><td>P17</td><td>Record RFC</td><td>“It is important to have the stakeholder’s participation in each part of the process, especially during an emergency.”</td></tr>
</table>

However, the interviewees did not find the Automated Feedback for Performance Models and Performance Predictions practice impactful since they didn’t know about it.

Continuous Integration and Continuous Delivery usually work together due to all the automation to have a deliverable ready to be deployed. Due to this, these practices are seen in the three processes, IM and PM are seen to have more impact on the Resolution and Investigation activities, while the CM process accelerates the implementation and the previous authorisation and approval activities. Since the deliverable is ready right after the build, the approvers have more confidence to move on with the change and implementation. Similarly, Continuous and Automated deployment enables the deployment of fixes or new functionalities to be fully scheduled and automatically accelerates the delivery and the change implementation. This also helps to ensure the successful implementation of changes by reducing human error.

The prototyping application was a practice that was only identified by the PM process. The interviewees see this practice as a way to ensure the problem is solved so the stakeholders can confirm the application's correct behaviour.

Regarding Testing practices, Automated Testing is seen as more present than Continuous Testing since Continuous Testing was only applied to the CM case study. Nevertheless, Automated Testing is seen in all three processes, accelerating the resolution in the case of PM and IM, and it is used in the approval stages of the changes by bringing more confidence in the tests of the new changes.

Process Standardisation is also seen as impactful on all processes. The main finding is that organisations will benefit if processes are adopted in the same way. This way, everyone will know the process and execute it easily.

The Infrastructure as Code practice was only identified in the IM and Emergency Change process. This practice enforces the relationship between the two processes since Emergency Changes are usually used to correct higher incidents. This practice enables the infrastructure to be taken care of, such as code, enabling versioning of the infrastructure and fast changes. This helps to fix the incidents since the infrastructure can be easily rolled back or updated on demand, returning the IT Service to normal operation.

Finally, the last practice from the group of DevOps practices, Stakeholder Participation, is also seen as impactful in the three processes. In the three processes, there is a consensus that the stakeholders are the key to the best management of backlogs, approvals, and authorisations and the main point of contact for knowing how the application should behave. So, they should always be informed about any issues or changes to the applications so they can provide the best inputs.

To summarise, it is possible to conclude that in the opinion of these professionals, DevOps practices can potentially improve the ITIL processes, using automation to accelerate the resolution of issues and to reduce human error, providing more confidence to the process owners, hence accelerating the authorisation and approval activities. Some of the activities can even be automated due to DevOps practices. For example, once the code from a package that was built automatically from CI/CD practices and automatically deployed in an environment can be automatically tested, and based on the input of these automated tests, the authorisation and approval activities from changes can be marked as done, moving with the process forward.

Table 10 - Insight on DevOps & PM process matches (Case Studies)

Practice ID	**PM Activities**	**Comment**
P1	Problem Detection	“Having a regular forum where potential Problem candidates are discussed can be helpful to the process.” “With continuous planning, we can find issues and obstacles that need to be addressed via the PM process.”
	Problem Investigation and Diagnosis	“This would be an important practice to have as it enables better task management (…) knowing where each investigation is on an ongoing manner is helpful for the process.”
	Problem Resolution	“Solution activities for Problems have to be planned. Having [Continuous Planning] is very relevant as it can expedite the implementation.” “Continuous planning sessions, the fact that they allow the opportunity for ongoing discussion, can be a positive element for the process.” “This can also prevent problems even before they get to Production.”
P2	Problem Detection	“Being aware of what development is to be implemented, and pointing out risks or issues, is an important contributor to the creation of Problems.”
	Problem Investigation and Diagnosis	“Performing Problem Root Cause Analysis always requires collaboration. Developers and Operations need to work together to determine the cause of Problems.” “Regardless of who does the Root Cause Analysis, it is important to have consistent feedback between the process teams and the tech teams. It is how we obtain necessary updates and improve communication.”
	Known Error Management	“The validation of Known Errors and Workarounds has to be checked and confirmed by the Development side, who often have the technical awareness to approve or reject this.”
	Problem Resolution	“This combination is needed to clarify the requirements of a problem solution and align how that solution will be introduced properly.” “It is the Developers who take the lead in the working out of a resolution for a Problem; however, the implementation of it already involves Operations. They should be working together.”
P8	Problem Investigation and Diagnosis	“A Problem investigation is an ongoing process. It should be easy to track and simple to manage. The idea of Continuous Integration, with new information added to the Problem piece-by-piece, makes sense.”
	Problem Resolution	“In implementing solutions for Problems, this practice can speed things up to a higher pace.”
P6	Problem Detection	“This is a key practice to identify problems proactively. We need to be aware of what is happening in the environment; having automation helps.” “We could link this with the event management process, working as input for PM.”
	Problem Logging	“There may be opportunities to automate the creation and logging of Problem records based on certain monitoring triggers.”
	Problem Closure	“We could use some sort of automated monitoring to confirm the complete resolution of a Problem investigation.”
P9	Problem Resolution	“The prototyping of a Problem solution could be done.”
P14	Problem Resolution	“We could use this practice to test how effective a Problem solution is before implementing it in Prod.”
P17	Problem Detection	“Additional ‘eyes on the field’ are important to detect things as soon as something goes wrong.”
	Problem Logging	“The prioritisation of a Problem and its classification, based on urgency and impact, depends on the Stakeholders' insight and participation.”
	Problem Resolution	“Stakeholders should be involved in confirming solutions to the Problems.” “They are the best suited to consider, agree, comment on and confirm the solutions to Problem issues.”
P15	Problem Investigation and Diagnosis	“Having standard processes allows us to organise and help carry out investigations. Standardisation also easily points out what may have failed during a Problem.”
	Known Error Management	“Only with a standardised process within our teams can we ensure that a good Known Error Knowledge Base is in place; it prevents wasted time where we have people investigating matters that are already known or under resolution.”
	Problem Closure	“In Closure, everything should be documented, and everyone should be aware of and follow the same process. (…) the outputs of each activity towards Closure should have a predictable outcome.

5.2.Interview Analysis on DevOps Benefits and Challenges

This section analyses the benefits and challenges of adopting DevOps practices in the three referred processes. The interviewers asked what each practice would benefit the interviewees. After performing a qualitative analysis of the answers, Table 14 shows the responses.

From Table 11, it is possible to see many comments about the quality and speed of the delivery, better communication and collaboration, and stability of the application by identifying issues earlier. It is possible to show that these benefits were also found in the existing literature, as per Table 4. The only practice not identified with a benefit was P5. However, none of the interviewees implemented this. Therefore, they did not have a lot to say about this practice.

Regarding challenges, Table 12 shows the interviewees' responses about the challenges found in the processes. There will be only comments for the practices where the weighted average is below three since the interviewees classified the adoption from DevOps practices from 1 to 5, with 3 being the neutral value.

In Table 12, there are a lot of different comments about the adoption of the adoption of DevOps practices. Some comments are about how the organisation faces the practices, such as Process Standarisation and Stakeholders Participation. From Table 11, the interviewees noticed the benefits of having these practices in place. However, the organisation's top management and the proper stakeholders need to be engaged to participate in these practices. This will bring the whole organisation on board with these practices. It is possible to conclude that this could be a cultural challenge.

The challenge regarding monitoring practices is keeping up with application changes and what to monitor. Here, a DevOps culture mindset could be in place so the developers and operators can improve the monitors together due to the new developments and functionalities being delivered.

Regarding other delivery practices, such as Deployment and Delivery practices, the interviewees stand out in the different deployment processes through the different applications that could be difficult to operate and maintain. Also, there could be a lack of technical knowledge in implementing these practices.

Furthermore, even though it does not stand out in Table 12, the researchers would like to comment on Continuous Delivery in the IM Process: "*It could harm the investigation; since DevOps allows constant delivery and constant deployments, we can have five deployments (one per day), and an incident is created after seven days. We may need to check all the code delivered in those five deployments*". Here is a comment where the practice could negatively impact the process. A good organisation of the code and functionalities would be needed to ensure that the code can be easily traced back. This will help to identify which code could cause the issue, facilitating the resolution of the incident.

Table 11 - DevOps Benefits (Case Studies)

ID	IM Process	CM Process	PM Process
P1	Good planning helps us respond to business needs at the right time. This creates a more collaborative environment between business and IT and improves customer satisfaction.	Employing good planning contributes to the product vision, i.e. where the team is heading in terms of what needs to be delivered. The quality results from how the requirements were aligned and incorporated into the process. Also, it empowers the team to react to changes and new requirements that either come from the stakeholders or during the development cycle, creating this effective communication-oriented practice.	Ensure that we are all constantly on the same page. This allows us to be Lean, constantly thinking about the next steps and priorities and allows flexibility.
P2	It contributes to collaboration between the two IT teams and is an opportunity to fix issues before they reach production, improving delivery quality. The teams will also share experiences, maximising their knowledge.	It contributes to continuous improvement among team members. It assesses "what went wrong?" and "what went right?". It is also a means to facilitate communication of the lessons learned for each iteration.	There is a gap between Ops and Dev. Having this in place requires an investment from both ends, where we help each other. In a utopia, we are doubling teams.
P3	It will help to identify issues in several layers to identify issues earlier.	It helps to understand the system performance in production and provides information so that the team can react sooner rather than later. This way, the system achieves stability, and the team can profit from the information to make an action plan whenever an incident arises.	-
P4	It helps to evaluate the application's performance so that action can be taken before it impacts the business.	It identifies the current application state in production and verifies performance bottlenecks. It complements other metrics, such as key performance indicators (KPIs) and operational and delivery flow and is quite helpful in reacting to changes when they arise. This is why, just like Continuous Monitoring, it is advantageous to be reactive to such metrics.	-
P6	Find issues at an early stage to take action on the issues.	Just like Continuous Monitoring, it bears valuable information to the team so that it can be proactive instead of reactive when handling incidents. This way, the team can work on a strategy to tackle an issue before it affects the end users.	"We can see the status of things without human intervention."
P7	Identifies the current behaviour and performance indicators of the application, providing a clear picture of where to act	It summarizes the information graphically instead of tabularly and checks the system performance. By adopting such monitoring tools, the team can react swiftly based on graphical information and combine it with other key performance indicators (KPIs).	-
P8	The main practice for the quality of the team delivery is to integrate the code with other deliveries, which will help us find any issues in the code so we can correct them.	It draws on two beneficial aspects: product delivery and developer experience. The first enables fast, reliable and predictable software that is deliverable in development. The second suggests improving the focus on development tasks, making the concerns about software integration agnostic to the developers.	Developers work to improve our ITSM tool through continuous integration, which involves constantly Implementing small changes and features. This allows us to implement quick corrections in a few hours/minutes.
P9	It shows the new behaviour of a correction or a new functionality for the business to approve before it goes to production, helping to satisfy the customer's needs.	It introduces the product vision and strategy to the team and stakeholders earlier, which helps to evaluate the concepts of the new end product. Further contributing to product improvement, it may impact product planning based on findings from the prototype as potential issues are unveiled at earlier stages in product development.	We benefited from the Dev side. However, it is very important that expectations of what the prototype should be are clearly defined.
P10	It will help the code to reach production or higher environments quickly.	Given its continuous nature, it provides fast deployment to the end users as soon as they meet the quality requirements.	-
P11	This will ensure the code is deployed correctly and automatically, mitigating human error and ensuring delivery quality.	As a process ensured by automation, it promotes predictable and deterministic deployments. Manual interventions may be error-prone, and the risk of failure is reduced when adopting an automated approach.	Our ITSM tool helps with bulk changes, reducing some manual work, but the effort is never entirely automated.

P12	This makes sure the package is ready to be deployed anytime it is needed. Even if a rollback is needed, we can install previous versions of the packages, helping to resolve issues faster.	It reduces time and effort when delivering software, whether improvements/new features or fixing bugs.	-
P13	Continuous testing will give more confidence in the code that is being delivered since it is being passed between different environments and ensuring the same behaviour.	Considering that any new change will be tested, it is a way to minimise the impact on existing production implementation by changing/adding new code. It functions as a quality gate, ensuring the system consistently functions. This is backed by tests, which provide a certain confidence level based on the robustness and certainty given by this quality gate.	-
P14	Automated testing will guarantee the quality of the code by checking whether it changes the behaviour of something that was previously correct.	It serves as a quality guarantee on the deliverable, as for each new change introduced, given the tests' deterministic nature, it increases the odds of shipping nonbreaking features, which creates a robust and safe deliverable.	These "sanity checks" have been applied successfully, and a dedicated team is in place for them specifically.
P15	This practice will help new people to join the teams since the processes will be the same across the organisation.	It facilitates the process for different team members, ensuring deterministic outcomes, because the process will be followed thoroughly and, as a result, it reduces cognitive load as it is something well-known among team members.	There is a standard scrum process in place, but it can change depending on how day to day activities is done. Some flexibility is still needed for motivation, however.
P16	This practice will help to solve issues quickly, if we have a change on the infrastructure, we can just rollback to the previous version and quickly solve the issue	-	-
P17	The stakeholders of the application should always be informed and participate when needed because they will be the main players affected by the issues. This will ensure that everyone is on the same pace about the issues and their impacts and helps to re-prioritize based on business inputs.	By keeping different stakeholders "in the loop", they can contribute to the vision of what is expected and foment knowledge sharing between different parties. Additionally, it supports the decisions made during planning. Therefore, it enables a collaborative and communicative environment between everyone in the team, including product and engineering stakeholders.	Having visibility and participation in the work adds to the reputation of the Team, which can improve performance.

Table 12 - DevOps Adoption Challenges (Case Studies)

Practices	IM Process					Average	CM Process					Average	PM Process					Average	Comments
	1	2	3	4	5		1	2	3	4	5		1	2	3	4	5		
P1	-	2	1	3	-	3,2	-	3	1	4	2	3,5	-	3	1	2	2	3,4	-
P2	-	1	3	1	-	3,0	-	-	3	5	2	3,9	1	2	3	1	2	3,1	-
P3	-	-	1	1	-	3,5	-	2	4	3	1	3,3	-	-	-	-	-	-	-
P4	-	1	2	-	-	2,7	1	4	3	2	-	2,6	-	-	-	-	-	-	CM: "It is hard, given the complexity of its implementation and in-depth systems knowledge. Furthermore, it cannot confer results promptly, as the time needed to collect relevant and useful data can be long." IM: "It is hard to keep the monitors up with the new functionalities and also what to monitor over time."
P5	-	-	-	-	-	-	1	1	8	-	-	2,7	-	-	-	-	-	-	CM: "It is hard, given the complexity of its implementation and in-depth systems knowledge. Furthermore, it cannot confer results promptly, as the time needed to collect relevant and useful data can be long."
P6	1	-	1	2	-	3,0	-	2	4	2	2	3,4	3	1	1	-	1	2,2	PM: "The difficulty is in defining the ideal state perfectly, and then using automated monitoring to spot deviation. It is tough to implement."
P7	-	1	-	1	-	3,0	-	1	5	2	2	3,5	-	-	-	-	-	-	-
P8	-	1	5	2	1	3,3	1	3	3	3	-	2,8	1	2	1	3	-	2,9	PM: "A perfect design of how things are done needs to be in place."
P9	1	-	-	1	-	2,5	-	1	5	3	1	3,4	-	-	2	2	1	3,8	IM: "In our context, it is quite hard to have environments where we can build these prototypes; however, when we can, we can benefit so we can have the correct feedback about the application behaviour."
P10	-	1	-	1	1	3,7	2	5	1	2	-	2,3	-	-	-	-	-	-	CM: "It is hard, as it is complex to attain and inspect, e.g. a "black box", and during the deployment phase, there are nuances that make it hard to implement."
P11	-	2	3	-	2	3,3	2	7	1	-	-	1,9	-	-		1	1	4,5	CM: "It is hard because it relies on nuances in the process, the technical effort, and the potential to introduce new changes to the code base that might not be ready."
P12	-	1	2	1	1	3,4	-	6	2	2	-	2,6	-	-	-	-	-	-	CM: "It is hard, similar to continuous and automated deployment because the team must ensure the system will work when new changes are delivered, for instance, backward compatibility."
P13	-	1	-	-	1	3,5	1	2	2	5	-	3,1	-	-	2	-	1	3,7	-
P14	1	3	-	-	-	1,8	-	1	5	4	-	3,3	-	-	-	-	-	-	IM: "This practice can bring many benefits in regression tests; however, we cannot develop all the missing tests."
P15	-	2	1	1	-	2,8	-	2	5	3	-	3,1	1	3	2	-	1	2,6	PM: "Depends on the willingness of the organisation to adopt standardised processes." IM: "This needs to be raised by the top management. Otherwise, the rest of the organisation will not follow."
P16	-	-	-	-	-	-	1	2	6	1	-	2,7	-	-	-	-	-	-	CM: "It is hard due to the technical knowledge required to implement these scripts. They are very specialised." IM: "Even though the interviewees recognise the benefits, they do not know what needs to be implemented."
P17	-	-	3	1	-	3,3	1	2	1	3	3	3,5	2	1	3	3	-	2,8	PM: "Stakeholders generally want to be involved in what is being done. Participation on its own is not challenging. It is required to determine what forums to use."

5.3.Interview Analysis on Process Performance

For the RQ3 of this research, the researchers aim to conclude whether, in the interviewee's opinion, the practices impacted process performance. Table 13 describes the comments about the DevOps practices regarding ITIL process performance.

Table 13 - Comments About DevOps Practices and ITIL Processes Performance (Case Studies)

ID	IM Process	PM Process
P1	If we consider the performance of the IM process in terms of SLAs, this practice may not help the process performance since continuous planning may change the priority of the team backlog several times. However, it will be able to re-prioritize the backlog based on what is best for the business.	Some investigations are very long (…). They involve complex actions that need to be consistently monitored. If planning for these Problems is not continuous, we will lose track of what has been done, what is being done, and what still needs to be completed.
P2	Feedback Loops can help find resolutions faster and identify new issues that have yet to be reported. So, they contribute to the stability of the system and to process performance.	We could imagine a DevOpsProb team, where the process knowledge is combined with technical expertise to resolve Problems (…) quickly. Specialisation can still exist, but all are working towards the same purpose. Communication between Problem managers and those developing solutions is needed to ensure things are done in an organised way and at the right time.
P8, P10, P11, P12	Due to the automation behind continuous integration, delivery, and deployment, a package can be quickly available and deployed to several environments, ensuring that the resolution will be deployed faster.	-
P14	Automated testing will help determine whether the new solution causes no failures in the existing software, accelerating the tests and, therefore, the resolution.	-
P15	If the processes are standardised across the organisation, everyone will know the process and the activity responsible. There will be no leak times between activities.	-
P16	Even though the interviewees knew little about this practice, they agreed that it would help to identify the issue and also resolve it.	-
P17	Stakeholders are the key between the business and the IT teams. So, their participation is crucial to any IT process since IT is there to support the organisation's business. They can help to identify the issues and the expected behaviour faster so the IT teams can solve the issues faster.	Other processes may need to be more customer-facing, but having more participation from the business in PM makes it possible to know where to focus effort and where to make priorities. If Stakeholders understand the process, they can be essential allies over the time it takes to resolve investigations."

For the CM case study, the interviewees did not have an opinion for each practice, so they considered DevOps globally and how it could impact the process. However, by interpreting the answers from the interviewees, it is possible to verify across each contribution just how substantial DevOps is when a team faces a change that needs to be addressed. This is supported by how automated processes promote faster deliveries, meaning rapid delivery to the end users because of this continuous approach. Automation ensures both quality gate and traceability. Finally, it promotes autonomy and readiness to implement those changes and collaboration among all stakeholders and the team itself.

These conclusions are supported by Table 13 when seen from the perspective of the IM and PM case study, namely, by the quickness to identify the root cause of an issue and deliver the fix or workaround to diminish the business impact. All of this will impact the main activities of the processes, accelerating the process instances and improving the process performance.

6. Conclusion

The aim of this research was to explore the relationship and the possible impact that DevOps practices could have on three ITIL processes, namely, IM, PM and CM.

Due to the lack of insights about this topic on the existing literature, the researchers have opted to use Multiple Case Study as the research methodology, where the case study objects were IT teams from different organizations that uses both DevOps practices and one of the ITIL processes. Case Study as a research methodology was the option since it aimed to explore the experience of these IT teams regarding the experience of working with both DevOps and the ITIL processes.

There is possible to conclude from the previous analysis, based on the interviewees' experience, that each practice could impact at least one of the processes, except Automated Feedback for Performance Models and Performance Predictions. Mostly, because the interviewees didn't know or implemented this practice.

About the DevOps practices impact on the ITIL processes, there is possible to see that for the CM process, the practices related with automation like automated deployment, continuous delivery and continuous deployment, were seen with more impact, due to the reduction of human error, the change managers can approve or automatically approve the changes improving the process performance.

Related with IM and PM, similar practices were seen to cause more impact, such as Automated Monitoring and Continuous Integration. First to identify the issues earlier so the resolution could be applied faster, bringing more stability for the business operation, while for Continuous Integration to accelerate the delivery of a development or a fix. Also, for both processes, Feedback Loops between Dev and Ops was seen quite impactful. This practice requires a cultural change of mindset to enable the collaboration between these two roles to accelerate the resolution of issues and to stabilize the application.

Nevertheless, some challenges were found about adopting the DevOps practices. The ones to stand out would be the mindset of the organization for the collaboration and also the technical knowledge to implement the practices.

Moreover, this research still has some limitations, even though the study converges for a higher level where DevOps can be applied in a ITIL environment, and having an analysis from different perspectives, there could be more case studies about the same process but with different organizations to verify if the same challenges and benefits were found.

Thus, future research could be done to explore how these two topics can be implemented together in different industries. Also, there can also be done some research in other ITIL processes identified in the review publication (Faustino et al., 2023) such as Release Management, Event Management, Service Request and Service Catalog Management. The authors mention that due to the automation offered by DevOps, these processes could be simplified and automated, improving the organisation IT Services performance.

References

Aguiar, J., Pereira, R., & Braga Vasconcelos, V. (2018). An Overlapless Incident Management Maturity Model for Multi-Framework Assessment (ITIL, COBIT, CMMI-SVC). IJIKM, 13, 137–163. https://doi.org/10.28945/4083

Alsolamy, A. A., Khan, U. A., & Khan, P. M. (2014). IT-business alignment strategy for business growth. 2014 International Conference on Computing for Sustainable Global Development, INDIACom 2014, 364–366. https://doi.org/10.1109/IndiaCom.2014.6828160

AXELOS. (2019). ITIL4 Foundations (1st ed.). The Stationary Office.

Ayat, M., Sharifi, M., Sahibudin, S., & Ibrahim, S. (2009). Adoption factors and implementation steps of ITSM in the target organizations. Proceedings - 2009 3rd Asia International Conference on Modelling and Simulation, AMS 2009, 369–374. https://doi.org/10.1109/AMS.2009.114

Bartolini, C., Salle, M., & Trastour, D. (2006). IT service management driven by business objectives An application to incident management. 2006 IEEE/IFIP Network Operations and Management Symposium NOMS 2006, 45–55. https://doi.org/10.1109/NOMS.2006.1687537

Beck, K., Beedle, M., Bennekum, A. van, Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., & Thomas, D. (2001). Agile Manifesto. https://agilemanifesto.org/

Cannon, D., & Wheeldon, D. (2007). ITIL Version 3 Service Operation. In Office of Goverment Commerce (1st ed., Vol. 34, Issue 19). The Stationery Office. https://doi.org/10.1016/j.im.2003.02.002

Cuppett, M. S. (2016). DevOps, DBAs, and DBaaS. In DevOps, DBAs, and DBaaS (pp. 73–85). https://doi.org/10.1007/978-1-4842-2208-9

de Kort, W. (2016). What Is DevOps? In W. de Kort (Ed.), DevOps on the Microsoft Stack (pp. 3–8). Apress. https://doi.org/10.1007/978-1-4842-1446-6_1

E Abreu, F. B., Freitas, J. M., De Bragança V. da Porciúncula, R., & Costa, J. C. (2010). Definition and validation of metrics for ITSM process models. Proceedings - 7th International Conference on the Quality of Information and Communications Technology, QUATIC 2010, 79–88. https://doi.org/10.1109/QUATIC.2010.13

Faustino, J., Adriano, D., Amaro, R., Pereira, R., & Silva, M. M. (2022). DevOps benefits: A systematic literature review. Software: Practice and Experience. https://doi.org/10.1002/spe.3096

Faustino, J., Pereira, R., & Alturas, B. (2018). DevOps Practices in Incident Management Process. ISCTE-IUL.

Faustino, J., Pereira, R., & Silva, M. M. Da. (2023). The Influence of DevOps Practices in ITSM Processes. International Journal of Services and Operations Management, 44(3), 390–407. https://doi.org/10.1504/IJSOM.2021.10038492

Galup, S., Dattero, R., & Quan, J. (2020). What do agile, lean, and ITIL mean to DevOps? Communications of the ACM, 63(10), 48–53. https://doi.org/10.1145/3372114

Guilfoos, B., & Triplett, J. (2022). ITSM in Supercomputing Improving service delivery, reliability, and user satisfaction. https://doi.org/10.1145/3491418

Hasibović, A. Ć., Tanović, A., & Granulo, A. (2023). The importance of ITIL4 adoption for IT service management in insurance companies. 2023 46th MIPRO ICT and Electronics Convention (MIPRO), 1341–1346. https://doi.org/10.23919/MIPRO57284.2023.10159950

Hemon, A., Lyonnet, B., Rowe, F., & Fitzgerald, B. (2020). From Agile to DevOps: Smart Skills and Collaborations. Inf Syst Front, 22(4), 927–945. https://doi.org/10.1007/s10796-019-09905-1

Hussaini, S. W. (2015). A Systemic Approach to Re-inforce Development and Operations Functions in Delivering an Organizational Program. Complex Adaptive Systems, 61, 261–266. https://doi.org/10.1016/j.procs.2015.09.209

Jabbari, R., bin Ali, N., Petersen, K., & Tanveer, B. (2016). What is DevOps? A Systematic Mapping Study on Definitions and Practices. Proceedings of the Scientific Workshop Proceedings of XP2016 on - XP '16 Workshops, 1–11. https://doi.org/10.1145/2962695.2962707

Kaiser, A. K. (2018). Reinventing ITIL® in the Age of DevOps. Apress. https://doi.org/10.1007/978-1-4842-3976-6

Kaplan, A., Busch, K., Koziolek, A., & Heinrich, R. (2018). Categories of change triggers in business processes. Proceedings - 44th Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2018, 252–259. https://doi.org/10.1109/SEAA.2018.00049

Kim, G., Debois, P., Willis, J., & Humble, J. (2016). The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations. IT Revolution Press.

Kush, P. (2013). Difficulties in Implementing Effective Problem Management. International Journal of Scientific & Engineering Research, 4(8), 98.

Lahtela, A., Jäntti, M., & Kaukola, J. (2010). Implementing an ITIL-based IT service management measurement system. 4th International Conference on Digital Society, ICDS 2010, Includes CYBERLAWS 2010: The 1st International Conference on Technical and Legal Aspects of the e-Society, 249–254. https://doi.org/10.1109/ICDS.2010.48

Latrache, A., Habib Nfaoui, E., & Boumhidi, J. (2015). Multi agent based incident management system according to ITIL. Intelligent Systems and Computer Vision (ISCV), 1–7. https://doi.org/10.1109/ISACV.2015.7105552

Lema, L., Calvo-Manzano, J. A., Colomo-Palacios, R., & Arcilla, M. (2015). ITIL in small to medium-sized enterprises software companies: Towards an implementation sequence. Journal of Software: Evolution and Process, 27(8), 528–538. https://doi.org/10.1002/smr.1727

Lou, J.-G., Lin, Q., Ding, R., Fu, Q., Zhang, D., & Xie, T. (2013). Software analytics for incident management of online services: An experience report. 2013 28th IEEE/ACM International Conference on Automated Software Engineering (ASE), 475–485. https://doi.org/10.1109/ASE.2013.6693105

Lwakature, L. E. (2017). Devops Adoption and Implementation in Software Development Practice: Concept, practices, benefits and challenges. 99. https://doi.org/ISBN 978-952-62-1711-6

Marrone, M., & Kolbe, L. M. (2011). Impact of IT Service Management Frameworks on the IT Organization. Business & Information Systems Engineering, 3(1), 5–18. https://doi.org/10.1007/s12599-010-0141-5

Miles, J., & Gilbert, P. (2005). A Handbook of Research Methods for Clinical and Health Psychology (1st ed., p. 315). Oxford University Press.

Pratama, R. Y., & Umaroh, S. (2023). An IT Asset Governance Model Design Using COBIT 2019 And ITIL V4 Framework at BKU Itenas. E3S Web of Conferences, 484. https://doi.org/10.1051/e3sconf/202448402006

Rance, S. (2011). ITIL 2011 Service Transition (A. Hanna, Ed.; 1st Edition). The Stationery Office.

Reiter, M., & Miklosik, A. (2020). DIGITAL TRANSFORMATION OF ORGANISATIONS IN THE CONTEXT OF ITIL® 4. https://www.microsoft.com/sk-sk/microsoft-365/microsoft-teams/free

Richard, Gaol, F. L., Warnars, H. L. H. S., Abdurachman, E., & Soewito, B. (2019). Development of Web Application based on ITIL – Incident Management Framework In Computer Laboratory. 2019 International Conference on Information Management and Technology (ICIMTech), 1, 120–125. https://doi.org/10.1109/ICIMTech.2019.8843799

Riungu-Kalliosaari, L., Mäkinen, S., Lwakatare, L. E., Tiihonen, J., & Männistö, T. (2016). DevOps adoption benefits and challenges in practice: A case study. 17th Internacional Conference, PROFES 2016, Throndheim, Norway, 10027 LNCS, 590–597. https://doi.org/10.1007/978-3-319-49094-6_44

Sarwar, M. I., Abbas, Q., Alyas, T., Alzahrani, A., Alghamdi, T., & Alsaawy, Y. (2023). Digital Transformation of Public Sector Governance With IT Service Management-A Pilot Study. In IEEE Access (Vol. 11, pp. 6490–6512). Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/ACCESS.2023.3237550

Serrano, J., Faustino, J., Adriano, D., Pereira, R., & da Silva, M. M. (2021). An it service management literature review: Challenges, benefits, opportunities and implementation practices. Information (Switzerland), 12(3). https://doi.org/10.3390/info12030111

Sharifi, M., Ayat, M., Rahman, A. A., & Sahibudin, S. (2008). Lessons learned in ITIL implementation failure. Proceedings - International Symposium on Information Technology 2008, ITSim, 1, 8–11. https://doi.org/10.1109/ITSIM.2008.4631627

Sharma, S., & Coyne, B. (2014). DevOps For Dummies (IBM, Ed.; 2nd ed.). John Wiley & Sons, Inc.

Silva, M. A., Faustino, J. P., Pereira, R., & da Silva, M. M. (2018). Productivity gains of DevOps adoption in an IT team: A case study. Proceedings of the 27th International Conference on Information Systems Development: Designing Digitalization, ISD 2018, 351.

Soni, M. (2016). End to End Automation on Cloud with Build Pipeline: The Case for DevOps in Insurance Industry, Continuous Integration, Continuous Testing, and Continuous Delivery. Proceedings - 2015 IEEE International Conference on Cloud Computing in Emerging Markets, CCEM 2015, 85–89. https://doi.org/10.1109/CCEM.2015.29

Steinberg, R. (2011). ITIL Service Operation (A. Hanna, Ed.; Fist Editi). The Stationery Office. http://books.google.com/books?id=w382YAAACAAJ&pgis=1

D. Teixeira, R. Pereira, T. A. Henriques, M. Silva, and J. Faustino, "A Systematic Literature Review on DevOps Capabilities and Areas," International Journal of Human Capital and Information Technology Professionals (IJHCITP), p. 22, 2020, doi: 10.4018/IJHCITP.2020070101.

Tellis, W. (1997). Information technology in a university: a case study. Campus-Wide Information Systems, 14(3), 78–91. https://doi.org/10.1108/10650749710187617

Thomas, G. (2016). How to do Your Case Study (J. Seaman & J. Piper, Eds.; 2nd ed.). Sage Publications Asia-Pacific PTE LTD.

Wahyudin, A., Widodo, Nasuha, A. F., & Junaeti, E. (2020). Strategic Alignment Maturity Level Model Using Drivers of Change in a Business Environment. 2020 6th International Conference on Science in Information Technology: Embracing Industry 4.0: Towards Innovation in Disaster Management, ICSITech 2020, 150–154. https://doi.org/10.1109/ICSITech49800.2020.9392036

Waschke, M. (2015). How Clouds Hold IT Together. In How Clouds Hold IT Together. Apress. https://doi.org/10.1007/978-1-4302-6167-4

Yamami, A. EL, Ahriz, S., Mansouri, K., Illoussamen, E., & Qbadou, M. (2017). Developing an Assessment Tool of ITIL Implementation. International Journal of Advanced Computer Science and Applications, 8(9), 183–190.

Yamami, A. El, Mansouri, K., Qbadou, M., & Illousamen, E. H. (2017). Multi-criteria decision making approach for ITIL processes performance evaluation: Application to a Moroccan SME. 2017 Intelligent Systems and Computer Vision (ISCV), 1–6. https://doi.org/10.1109/ISACV.2017.8054937

Yao, Z., & Wang, X. (2010). An ITIL based ITSM practice: A case study of steel manufacturing enterprise. 2010 7th International Conference on Service Systems and Service Management, Proceedings of ICSSSM' 10, 423–427. https://doi.org/10.1109/ICSSSM.2010.5530204

Yin, R. K. (2009). Case study research : design and methods. In Applied social research methods series ; Vol. 5. (Issue 5). https://doi.org/10.1097/FCH.0b013e31822dda9e

Zaidah, Z. (2007). Case study as a research method. Jurnal Kemanusiaan, 9(Journal Article), 1–6. https://doi.org/10.1177/15222302004003007

9.1. Case Study Data Collection Triangulation

Case studies are about seeing different behaviors from different angles [51]. Many authors advise the triangulation of several data collection methods [52], [53]. This enriches the case study where there could be multiple other findings and provides an internal validation of the case study [54] by proving the findings from multiple data sources.

Therefore, a triangulation of the findings from the interviews in the cases with other data collection methods will be shown in this thesis since it was not possible to include them in Publication nr. #8 due to space limitations.

The data collection methods for triangulation were direct observation and a focus group exercise. However, in the CM case study, this wasn't done. The case study was performed during the COVID-19 phase, when observation wasn't possible, and the organization rejected doing the focus group exercise.

As stated before, ITIL is a framework for managing the IT services lifecycle, providing guidelines for process implementation to guarantee this management [55]. This means that organizations adapt the processes to fit their ways of working.

9.1.1. Direct Observation

Observation can be seen as structured or unstructured [56]. Structured observation occurs when the researcher systematically looks for kinds of behaviors, while unstructured observation happens when the researcher informally observes essential details of what is happening [56]. Unstructured observation may also be called participant observation, where the researcher is also a participant.

Generically, observation is used to analyze the "before and after" of the behavior of a particular phenomenon after some change [54]. However, since the team has already implemented the practices, it is impossible to verify this behavior change in the first place.

Some of the researchers work at the organization where the case studies occurred. Therefore, the type of observation for these case studies should be considered unstructured.

Since organizations usually adapt their processes to fit their ways of working, a mapping between the organization's process activities and the ITIL standard process activities was created for better comprehension.

Table 9.1 shows the mapping between the ITIL IM process and the organization for the first case study IM process activities.

Table 9.1 – Mapping of ITIL & Organizations' IM lifecycle

Standard ITIL IM Activities	**Organization IM Activities**	**Activity Objectives from the organization**
Detection and Recording	Logging	Incident is being recorded, and the details are being filled
Classification and Initial Support	Categorization	The incident was assigned to the responsible team and is waiting to be resolved.
Investigation and Diagnosis	Investigation	The incident started to be analyzed, and work to find the issue started.
Resolution and Recovery	Recovery	A solution has been found and is waiting to be applied
Closure	Resolve	The incident is resolved, marking the service disruption or performance degradation as solved.
Monitor and Tracking	Closure	The incident is monitored for five days. If the issue recurs, it will be moved to Recovery activity again.

Three different participants usually create incidents: technical participants (for example, a member of the Application Management team), Service Desk operators, or incident managers. Usually, the incidents created by the technical users are low priority because it is assumed that if they were found by a technical person, they would not impact the business directly.

When the incident is being recorded in the Logging activity, it will be mandatory to fill in some fields, such as the business impact and urgency, to calculate the priority automatically, incident description, the affected application, steps to replicate, and choose a team that should solve the incident. In the categorization activity, the incident will remain in the solver team queue waiting to be worked. When someone starts to look at the incident, the status should be changed to Work in Progress, which will move the incident to the Investigation activity. There can be several statuses in this activity, like Work in Progress, Pending Customer, Pending others. This means the incident analysis has already started, but no conclusion has yet been reached.

Once the operator finishes the analysis and has a solution, it can be moved directly to resolve if a workaround should be applied or moved to the recovery activity. Usually, this activity is a placeholder for the incidents while the solution still needs to be applied. After the solution is applied, it will be moved to resolved. While the incident is still in the resolved activity stage, it can be monitored for five days, and if the issue happens again, it can be moved to Recovery so that someone can have another look at the solution. Otherwise, the incident will be automatically transferred to closure after five days and cannot be reopened.

Based on the previous explanation of the observed IM organization process and Table 9.1, the organization's activities follow the standard ITIL IM activities with some minor differences.

Table 9.2 shows the observed evidence of the impact of the DevOps practices on the organization's IM process activities. Thus, it is possible to conclude that the practices have a positive impact on the process activities.

In Publication nr. #8 the practices that caused more impact were P2 (19 matches), P6 (12 matches), P8 (12 matches) and P11 (11 matches), and the activities that had more impact were Resolution and Recovery (39 matches), Investigation and Diagnosis (21 matches) and Detection and Recording (20).

Looking at Table 9.2 there is possible to see that the Recovery and Logging activities were the most impacted by the DevOps practices, therefore confirming the findings from the interviews.

Table 9.3 shows the mapping between the ITIL standard PM activities and the organization PM activities.

Teams who have been granted access to the PM module of the ITSM tool being utilized are able to create new Problem candidates based on detected incidents, events and knowledge of existing faults in the business' IT landscape. Said teams tend to be comprised of Service Managers who collaborate closely with their service provider counterparts and other stakeholders from the organization.

Templates containing standard questions and requirements are used to ensure that the necessary information for an investigation's progress is documented. Meetings are also regularly scheduled to confirm that validation is granted for each Problem and reach agreements on ownership for which service provider(s) will be carrying out Root Cause Analysis.

Having validated a Problem investigation, Root Cause Analysis tasks are started. One or multiple providers collaborate to deliver an RCA document, and a proposal of activities required to resolve the Problem. Although no standard RCA methodology is defined, a "5 Whys" approach is most used. Here, technically experienced colleagues investigate the underlying cause of reported issues until their root cause is found; potential solutions are also listed, with respective owners and estimated due dates, aimed at permanently resolving the Problem or mitigating the risk of reoccurrence. The outcome of this effort is presented in an RCA document submitted to the organizations' Service Management community for approval or rejection, should revisions or clarification be needed.

When RCA approval is granted, each identified solution action is logged and tracked to completion in the organizations' ITSM tool. Depending on which party is responsible for implementing each activity, Providers and Service Managers deliver evidence, often from technical counterparts working in the investigation, when closing each task. Solutions can include, for example, the delivery of trainings, the updating of process documents and workflows, implementing system configuration updates through Change Management, replacing hardware or upgrading software versions.

Table 9.2 – DevOps Practices in Organization IM Process Activities

		IM Organization Activities					
		Logging	**Categorization**	**Investigation**	**Recovery**	**Resolved**	**Closure**
DevOps Practices	**P1**	(not observed)	During the planning of the teams' work, the teams usually check their incident backlog, re-prioritize the incidents, and add more detail.	(not observed)	(not observed)	(not observed)	(not observed)
	P2	Dialogs between the developers and operators raised technical incidents	(not observed)	There has been collaboration between developers and operators to find solutions.	(not observed)	(not observed)	(not observed)
	P6	There are monitors performed by the teams where some incidents can be logged from there.	(not observed)	(not observed)	(not observed)	(not observed)	Due to the monitors being applied, some incidents were being reopened since the solution didn't work as supposed
	P8 / P12	(not observed)	(not observed)	(not observed)	CI/CD processes have been seen accelerating the code validation and package build to move the solution between environments.	(not observed)	(not observed)
	P11	(not observed)	(not observed)	(not observed)	By automatically deploying the solutions, a solution could be moved between environments quite quickly to be tested and then moved to production.	(not observed)	(not observed)
	P14				The automated tests helped verify the quality of the developments and whether none of the application's existing functionalities had been negatively impacted.		

Once all solution actions are completed, the responsible provider revises the RCA document to document everything found and done during the investigation. This final document is then submitted for approval to the organization's Service Management community, who may share it with stakeholders and customers impacted by the investigated issue. If approval is granted, the Problem is considered Closed.

Table 9.3 – Mapping of ITIL & Organizations' PM lifecycle

Standard ITIL PM Activities	**Organization PM Activities**	**Activity Objectives from the organization**
Problem Detection Problem Logging, Categorization, and Prioritization	Problem Creation & Validation	Problem candidates are raised following either Major Incidents or based on monitoring patterns, incident analysis, and service management insight. Each candidate is reviewed for validation, ensuring it has the proper categorization, prioritization, and estimated business benefits before an actual investigation is initiated.
Problem Investigation and Diagnosis Known Error Management	RCA Creation and RCA Review	Providers review the issues reported in the Problem record and provide both Root Cause Analysis (RCA) and proposals for solution activities that may be completed towards resolving the Problem. A Known Error may be generated upon delivery of the RCA. Service Management reviews the delivered information and provides an approval or rejection.
Problem Resolution	Problem Resolution	Each solution item is documented and tracked to completion. Teams that own Problem-resolution items provide evidence of the status and outcome of each activity.
Problem Closure and Major Problem review, if applicable.	Resolution Review and Problem Closure	A summary of all Root Cause Analysis findings and all Problem resolution activities completed is delivered. Service Management reviews the delivered summary and provides an approval or rejection. If approval is granted, the Problem record is closed.

A team of dedicated Process Managers ensures that each activity of the four-stage Problem lifecycle is carried out according to expectations. They aid in defining action owners, create and route Problem tasks in the ITSM tool, schedule meetings for Problem validation and handling, and act in the event of escalations or overdue actions. Although they are not involved in delivering RCAs themselves, they ensure the process is properly driven by building connections between the business and its various Service Providers.

Based on the performed analysis, the PM process implemented in the organization is aligned with the understanding of PM presented in ITIL literature. Furthermore, observation of the four-stage Problem lifecycle being utilized indicates that some DevOps practices are already being applied to a certain extent and in select instances. This includes, for example, continuous Stakeholder Participation in the detection, resolution, and closure phases of Problem records; reliance on a Change Management process to carry out Problem Resolution activities; utilization of automated monitoring tools to identify Problem candidates; ongoing alignment of technical resources and developers, coupled with operations teams, to investigate, diagnose and resolve Problem investigations.

Evidence of utilizing DevOps practices in the organizations' current PM process can be found in Table 9.4.

Table 9.4 - DevOps Practices in Org. Problem Management

		Organizational PM Practices			
DevOps Practices		**Problem Creation & Validation**	**RCA Creation & Review**	**Problem Resolution**	**Resolution Review and Problem Closure**
	P1	The organization relies on daily PM meetings with Providers and weekly forums with Service Management to discuss the validation and progress of investigations and respective solution activities, ensuring they are completed according to agreed-upon timelines or adapting timelines if required.			
	P2	The Development and Operations community works closely to identify candidates for Problem investigations (often proactively, based on insight from ongoing deployments).	The Development and Operations community collaborates iteratively on Root Cause analysis. Both parties are required to collect information from Production environments, analyze it to identify the cause, and propose adequate solutions.	The Development and Operations community works together to plan, implement, and monitor each identified solution activity (this would include developing bug fixes, implementation planning, and ongoing monitoring to measure success rates).	(not observed)
	P6	The organization uses monitoring tools to identify common error trends and uses this information to initiate Proactive PM.	(not observed)	Automated monitoring tools are utilized to measure the success of implemented resolution activities and as an indicator of a Problem being resolved.	(not observed)
	P17	Stakeholders impacted by Incidents are encouraged to request the creation of a Problem record aimed at addressing their cause.	(not observed)	(not observed)	Confirmation from impacted Stakeholders that a Problem has been fully resolved is actively sought after in the Resolution Review stage of the PM lifecycle.

9.1.2. Focus Group Exercise

The last data collection method used in the case studies was a focus group. This data collection method is frequently used to provide a deeper qualitative analysis of a problem [57], enabling the researcher to ask questions and to request clarifications of ideas due to the direct contact with the study participants where everyone could provide their opinion [58].

There were two different focus group exercises, one for each case study. Both exercises had a group of 5 participants. The participants were chosen due to their experience in providing more information. The identified participants for the case study in the IM process were IM-C, IM-D, IM-G, IM-H, and IM-

J. The identified participants for the case study in the PM process were PM-B, PM-D, PM-F, PM-I, and PM-J. The details about these participants can be found in Table 4 in Chapter 9 in Publication nr. #8.

In both groups, the participants were shown the results of the Semi-Structured interviews about the impact of DevOps practices, and they could comment on them.

In the IM case study group, there were some comments from the IM-J about the P17 practice that could have more impact. In the opinion of P17, the stakeholders should always be present to know the status of the incident. They are the most interested party to have the incident resolved. The other interviewees agreed but opposed, saying that it depends on the squad. Some stakeholders like to be involved in the process and follow its status, while others only want to know the final output, leaving the team to work on solving the incident because they thrust on the developers.

All the participants also expected more impact from the P14, which brought a moment for recognizing the impact of this practice. They agreed that this practice would have more impact by having a more robust testing framework; however, they agreed that it was a challenge to implement more tests. This matches one of the challenges identified in Table 12 in Publication nr. #8.

They agreed on the impact of the other practices; however, when discussing P6, they also discussed the possibility of creating the incidents automatically instead of having someone go to the application and create them manually.

Moreover, there was also a comment that got the participant's attention there was one interviewee who said the following about P12: "*It could harm the investigation; since DevOps allows constant delivery and constant deployments, we can have five deployments (one per day), and an incident is created after seven days. We may need to check all the code delivered in those five deployments*". In this comment, it is possible to observe a negative impact from this practice in the process. However, IM-G said that even if he could agree with this comment, there would be a specific and rare scenario where all the other participants agreed. Although IM-H thought that would be an opportunity to retrospect if this would be a generic concern from other team members and if there could be some solution for better identification of the code that is inside of those packages, and if this happens, there would be easier to filter which could be the damaging package.

Regarding the PM case study group, interviewee PM-D commented that there was an expectation that the P14 practice "*should have ranked higher in value*" as clear benefits were observed based on experience; interviewee PM-B commented that this was not too surprising "*as the test automation is currently too challenging to implement fully.*"

Only a comment was made on the PM section's outcomes regarding Known Error Management. Interviewee IM-J interpreted this practice's low application by stating that "it rates lower due to frequently lacking formal integration with the Knowledge Management process" in utilized ITSM tools.

Interviewee PM-F questioned the high significance of continuous planning in problem resolution activity. An explanation was provided by interviewee PM-J, stating that "*if we are implementing a* *solution as part of a Problem investigation, it makes sense to check continuously how its deployment is being done*", and that would fall in the purview of the Continuous Planning practice.

An insight presented by Interviewee PM-B received broad consensus in the group when he noted that there seemed to be a pattern in that "*the ones that are rated highest are those focused with either communication or with planning*"; that is, the combinations with highest significance tend to be those in which communication or planning is done.

Having this analyzed, there was questioned about the study's research questions; in this case, there was asked about RQ1 and RQ3. The RQ2 could be considered redundant due to the findings of the practices.

In Tables 9.5 and 9.6, it is possible to see the comments of the interviewees about the RQs and the DevOps practices. In the general opinion of the participants, they would agree that all the practices would have an impact, but they have narrowed it down to the practices that they could see as having a more meaningful impact. There can be seen more practices impacting the IM process than PM process. Even though the ones that are impacting PM are also impacting the IM. These were also the practices seen with more impact in the Publication nr. #8 and on the observation in section 9.1.1. Thus, validating the data collection method triangulation.

It is possible to see that the justifications that the participants gave to justify the impact and process performance improvement are similar between the two processes. Practice P15 comments show that this practice could have an impact, but it feels that processes are not being followed entirely. IM-C said, "*This would be good if the process gets standardized by everyone and there are no exceptions*" and PM-I said "*Organizational culture might not be leaning towards having a lot of standardization in the process, but it is needed in order for it be predictable and consistent*". It seems that in PM-I there could be exceptions to the process when the organization culture is not leaning to have standardization, and IM-C states that. In IM, it seems that P1 is not seen to help improve the process. However, in PM, there is a contradictory opinion where it helps to track the problems so they can be solved. In PM, the delivery practices are not seen to be as impactful as in PM, maybe due to the comment made by PB-B saying the planning practices and communication are the most rated.

In the following sections, a set of extra interviews with other professionals with different experiences will be presented to clarify the impact of DevOps practices on ITIL processes.

Table 9.5 - Focus Group - Impact of DevOps in IM Process

Impact of DevOps on IM Processes			
RQ1 – How do DevOps Practices Impact IM Process		**RQ2 - Has the process performance improved?**	
P1	IM-D: "This allows a good prioritization of our backlog. This gives us an expectation of our work and where to improve." IM-J: "Allows us to know the roadmap of what is supposed to deliver, manage good expectations, and know where we should allocate more effort."	P1	IM-J: "Yes, it has a good impact, but not SLA-wise. If we need to wait for the incidents to get into the backlog and be prioritized, we will not be able to fulfill the SLA. So, the SLA needs to be adjusted or removed, like it was. However, by having business representatives when it comes to planning, we are sure that we will fix the big pains before other items, which will cause an increase in customer satisfaction."
P2	IM-H: "No doubt there is one of the most impactful practices. Having all the experts collaborate to solve an issue saves much time."	P2	IM-C: "Yes, it saves much time investigating the issues." IM-G: "This collaboration will save time and promote quality and confidence between the teams. They will feel secure by having someone on board to solve an issue."
P6	IM-C: "The issues are found earlier. It feels good to find an issue, report it to the business, and have a solution in mind. The business will feel comfortable and happy without work."	P6	IM-H: "This is similar to P1. It can find the issues earlier but will not close the process instance faster. Only if we can see this from a perspective where the monitor's output hints at the issue."
P8 / P11 / P12	IM-G: "All the fixes are implemented smoothly and easily. This brings much security and ensures that everything is going fine and will be implemented correctly."	P8 / P11 / P12	IM-J: "Compared with the old times when we had to do everything manually, the time to implement a solution has decreased. Also, there is a better usage of resources."
P15	IM-C: "This would be good if everyone standardized the process and there were no exceptions."	P17	IM-J: "The stakeholders are a key component to moving the process. Even if they can resolve impediments, they also have the business knowledge to help investigate the issue."
P17	IM-J: "Having the stakeholders involved from end to end in the process is the best way to know who can help if help is needed." IM-H: "They can also unlock several impediments."		

Table 9.6 - Focus Group - Impact of DevOps in PM Process

<table>
<tr><th colspan="4">Impact of DevOps on PM Processes</th></tr>
<tr><th colspan="2">RQ1 – How do DevOps Practices Impact IM Process</th><th colspan="2">RQ2 - Has the process performance improved?</th></tr>
<tr><td>P2</td><td>PM-J: “The Automated Monitoring practice can easily be applied to gather Problem candidates from our systems (…) proactively. This is currently one of our main sources for new investigations.”</td><td>P1</td><td>PM-J: “Some investigations are very long (…) they have complex actions that must be consistently monitored. If planning for these Problems is not done continuously, we will lose track of what has been done, what is being done, and what still needs to be completed. “</td></tr>
<tr><td>P6</td><td>PM-D: “We need to know the results of a new deployment, and having a bridge with Operations helps us respond more quickly.”

PM-J: “This is the most important practice that can be applied (…) the improved communication this and the Continuous Planning practice provide brings a lot of benefit and structure to the work of PM”.</td><td>P2</td><td>PM-D: “We could imagine a DevOpsProb team, where the knowledge of the process is joined with technical expertise to resolve Problems quickly (…) there can still be specialization, but all are working towards the same purpose.”
PM-B: “Communication between Problem managers and those actually developing solutions is needed to ensure things are done in an organized way and at the right time.”
Participant I: “Root Causes need to be found quickly, and data can be lost (…) feedback is important to ensure people are on the right track to conclude.”</td></tr>
<tr><td>P15</td><td>PM-I: “Organizational culture might not be leaning towards having a lot of standardization in the process, but it is needed to be predictable and consistent”.</td><td rowspan="2">P17</td><td rowspan="2">PM-I: “Other processes may need to be more customer facing, but having more participation from the business in PM makes it possible to know where to focus effort and where to make priorities”
Participant J: “If Stakeholders understand the process, they can be important allies over the time it takes to resolve investigations.”</td></tr>
<tr><td>P17</td><td>PM-B: “We need to define stakeholders, including those passively observing and those actively involved (…) for detecting Problems. Everyone is a valid stakeholder.”

PM-J: “All stakeholders can contribute to PM (…) their participation is valuable to us”.</td></tr>
</table>

9.2. Complementing Publication nr. #8 with Extra Interviews

To increase the rigor of this thesis, several case studies per process were initially planned in different organizations. However, due to limitations such as a lack of experience with DevOps and organizational data confidentiality, the researchers were not able to perform more case studies.

Moreover, since the data collection method from the case studies was semi-structured interviews, the researchers conducted interviews with other individuals with professional experience in DevOps and at least one of the ITIL processes to collect more information about this topic.

The main objective of performing these interviews was to verify and cross-check the findings of the case studies and if there could be other findings not existing in the case studies due to different industries and experiences,

Table 9.7 shows the characterization of the interviewees.

Table 9.7 - Interviewees Detail for Additional Interviews

IM Interviewees			
Interviewee	**Industry**	**Position**	**Experience (years)**
IM2-A	*Energy*	*Team Leader*	*15*
IM2-B	*Energy*	*Junior Developer*	*2*
IM2-C	*Energy*	*Developer*	*4*
IM2-D	*Energy*	*Full Stack Developer*	*10*
IM2-E	*Energy*	*Senior Developer*	*10*
IM2-F	*Energy*	*Senior Analyst*	*4*
IM2-G	*Energy*	*DevOps Engineer*	*10*
IM2-H	*IT Consultant*	*DevOps Engineer*	*3.5*
IM2-I	*IT Consultant*	*IT Manager*	*15*
IM2-J	*Banking*	*PMO*	*26*
IM2-K	*IT Consultant*	*Manager*	*18*
CM Interviewees			
Interviewee	**Industry**	**Position**	**Experience (years)**
CM2-A	*Consultant in FS*	*DevOps Engineer*	*3.5*
CM2-B	*Manufacturing*	*Process Manager*	*16*
CM2-C	*Consultant in FS*	*IT Manager*	*15*
CM2-D	*Banking*	*PMO*	*26*
CM2-E	*Insurance*	*Escalation Manager*	*26*
CM2-F	*Insurtech*	*Senior Developer*	*6*
PM Interviewees			
Interviewee	**Industry**	**Position**	**Experience (years)**
PM2-A	*Consultant in FS*	*IT Manager*	*15*
PM2-B	*Banking*	*PMO*	*26*
PM2-C	*Insurance*	*Escalation Manager*	*26*
PM2-D	*Energy*	*Problem Manager*	*14*
PM2-E	*Manufacturing*	*IT Chapter Lead*	*14*
PM2-F	*Manufacturing*	*Problem Manager*	*17*

The following section will detail the interviews about the three ITIL processes.

9.2.1. Extra Interview Analysis on DevOps Impact on Process Activities

This section's objective is to detail the analysis of the data collected from the interviewees about the three processes. After this, the results will be compared to those from Publication nr. #8.

Similar to Publication nr. #8, Table 9.8 will show the details of the interviewees' matches about the DevOps practices and the three processes. The practices identified in the table's header are the same as those in Table 1 in Publication nr. #8.

In Tables 9.9, 9.10, 9.11, 9.12, and 9.13, the insights of the practices related to the process activities can be seen.

Starting with the IM process, Table 9.8 shows that monitoring practices like P3, P4, P6, and P7 have a more significant impact on detecting and classifying incidents and, likewise, on observing and monitoring the application after the incident resolution is applied.

P1 practice provides continuous planning of the work of the IT teams, and it is seen as impactful on classification. Due to the constant planning, the IT teams will be able to understand the impact for better prioritization and to assign to the correct team in case the incident does not fit in the current team scope. Also, this practice is seen as impactful in the investigation and diagnosis activity due to the constant feedback inside the team, which could help fix the issue.

The interviewees were unanimous about the P8 practice. All the interviewees agreed that P8 impacts the incident resolution. It has also been considered that it could impact the incident investigation.

Regarding P10, P11, and P12, which are practices more related to delivering fixes or resolutions, it is possible to see that they have been seen as impacting Resolution, recovery, and Closure activities.

Other practices to stand out would be P15, P16, and P17. P15 has been seen as impactful in detecting and closing incidents. This can be due to the standardization of the incident creation and closure activities across the organization. P16 has been seen as impactful in accelerating the resolution of the incident, helping with the Investigation, resolution, and closure activities. In contrast, P17 impacts the initial activities of the process to detect and classify the incidents.

Also, regarding the IM process, Table 9.8 shows that the practices causing the most impact are P4, P8, P3, and P6, and the activities that were most impacted are incident Resolution and Detection.

Table 9.8 – Matches Between DevOps Practices and Process Activities (Additional Interviews)

Process / Activity		Practice ID																	
		P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	Total of Matches
IM Process	Detection and Recording	1	1	8	10	1	8	7	1	1	1	-	-	1	1	3	-	3	47
	Classification and Initial Support	5	-	7	7	-	8	7	-	-	-	-	-	-	-	2	-	3	39
	Investigation and Diagnosis	5	2	3	3	-	1	1	8	2	-	-	-	-	-	2	3	-	30
	Resolution and Recovery	1	2	3	2	-	1	1	11	-	8	7	7	-	-	2	3	-	48
	Closure	1	-	-	1	1	1	1	6	-	1	7	6	1	1	2	3	-	32
	Monitor and Tracking	1	-	4	5	1	6	7	-	1	-	-	-	1	1	4	-	1	32
	Total of Matches	14	5	25	28	3	25	24	26	4	10	14	13	3	3	15	9	7	-
Normal Change Process	Create RFC	4	1	1	1	1	1	-	-	1	-	-	-	1	1	3	-	5	20
	Record RFC	3	-	-	-	-	-	-	-	-	-	-	-	-	-	3	-	5	11
	Review RFC	4	1	-	-	-	-	-	-	1	-	-	-	-	-	3	2	5	16
	Assess and Evaluate Change	4	2	2	3	4	3	1	1	1	-	-	-	1	2	4	2	5	35
	Authorise Build and Test	1	3	1	1	1	2	-	1	3	1	1	2	1	2	4	2	3	29
	Coordinate Build and Test	2	2	1	1	1	2	-	2	1	1	2	-	2	4	4	2	3	30
	Authorise Change Deployment	2	2	1	1	2	-	1	1	1	4	3	3	-	1	4	2	4	32
	Coordinate Change Deployment	4	2	-	-	-	-	1	2	-	4	5	3	1	2	3	3	3	33
	Review and Close	3	1	3	3	2	3	3	-	-	2	2	2	1	3	2	-	5	35
	Total of Matches	27	14	9	10	11	11	6	7	8	12	13	10	7	15	30	13	38	
Standard Change Process	Review RFC	1	1	1	-	-	1	-	-	-	-	-	-	-	-	3	-	3	10
	Assess and Evaluate RFC	-	1	2	1	1	1	2	-	-	-	-	-	1	1	3	1	3	17
	Authorise and Schedule Change	2	1	2	1	1	2	1	3	-	2	2	2	-	-	3	1	3	26
	Coordinate Change Implementation	3	2	2	1	1	2	1	4	-	4	4	4	1	2	3	1	3	38
	Review and Close Record	2	1	3	2	2	1	1	2	-	2	2	2	1	1	3	1	3	29
	Total of Matches	8	6	10	5	5	7	5	9	-	8	8	8	3	4	15	4	15	-
Emergency Change Process	Record RFC	-	2	1	1	-	2	-	-	-	-	-	-	-	-	3	-	4	13
	Assess Change	1	3	3	1	1	1	1	-	1	-	-	-	1	1	3	-	6	23
	Change Approval	-	3	1	1	1	1	1	-	1	-	-	-	1	1	3	-	6	20
	Review Approval	-	3	-	-	-	-	-	-	-	-	-	-	-	-	5	-	4	12
	Implement Change	-	4	1	-	-	-	-	5	-	5	5	5	2	3	5	3	4	42
	Review Change	-	1	5	5	4	5	4	1	1	2	1	1	-	-	3	-	6	39
	Total of Matches	1	16	11	8	6	9	6	6	3	7	6	6	4	5	22	3	30	-
PM Process	Problem Detection	-	3	2	3	2	3	1	-	1	-	-	-	2	1	3	1	5	27
	Problem Logging	2	2	3	2	2	2	1	-	-	-	-	-	1	1	3	-	4	23
	Problem Investigation and Diagnosis	2	3	-	1	1	1	-	1	1	-	-	-	-	1	3	1	5	20
	Known Error Management	-	1	-	-	-	-	-	1	-	-	-	-	-	-	3	-	4	9
	Problem Resolution	1	3	4	1	1	-	-	6	2	3	3	4	1	2	4	4	4	43
	Problem Closure	2	1	-	-	-	-	1	1	-	1	1	1	-	-	4	1	5	18
	Total of Matches	7	13	9	7	6	6	3	9	4	4	4	5	4	5	20	7	27	-

Color Legend:
White – 1 to 2 matches
Light Grey – 3 to 4 matches
Medium Grey – 5 to 6 matches
Dark Grey – More than seven matches

Table 9.9 - Insights on DevOps & IM process (Additional Interviews)

Practice ID	**IM Activity**	**Comment**
P1	Classification and Initial Support Investigation and Diagnosis	“Regular plannings can help to classify and prioritize incidents when planning future sprints” “Due to the regular planning, prioritization may be increased or decreased since the feedback about the incident will be constant, and we may learn more about the impact. If the impact is reduced, the investigation and diagnosis will be de-prioritized”
P3, P4	Detection and Recording Classification and initial support	“Implementing performance metrics and having tools to monitor the application continuously will allow us to know the expected behavior of the application. This will help to identify incidents, classify and monitor after closure if the impacts have stopped.”
P6, P7	Detection and Recording Classification and initial support	“Helps to detect the Incidents that are impacting the application and to monitor” “By having a meaningful monitor, we can identify the impacts of the issue, helping to classify and prioritize the incident”
P8	Investigation and Diagnosis Resolution and Recovery	“Since the code is being checked in small chunks, it can be easier to track which code change causes the problem “The development cycle will be reduced due to the automatism implemented in CI, resolving faster”
P10	Resolution and Recovery	“Faster deployment of code to solve the issue”
P11	Resolution and Recovery Closure	“Automated deployments allow the resolution for the incident to be deployed faster” “The development cycle will be reduced due to the automatism implemented in CI, resolving faster”
P12	Resolution and Recovery Closure	“Allows a deployable package through several environments which will validate if the incident was solved”
P15	Monitor and Tracking	“It should be standard on all applications to have a kind of monitor to allow to know if the system health is good”
P16	Investigation and Diagnosis Recovery and Resolution	“IaC can help us investigate, troubleshoot, develop a quick fix, and close the incident. Having the infrastructure saved in a save is quite easy to detect any change that can cause the issue and fix it”
P17	Detection and Recording Classification and Initial Support	“Some stakeholders should know the business pains to create the incidents. They are the ones that know how the application should work. Due to this, they should be able to prioritize the incidents correctly”

Table 9.10 – Insights on DevOps & Normal Change process (Additional Interviews)

Practice ID	Normal Change Activity	Comment
P1	Create RFC Record RFC Review RFC Assess and Evaluate Change	"By continuously planning the IT services changes, it will be easier for the change initiators to create, record, and review the RFCs. Consequently, by using agile practices or methodologies, there will be more changes but smaller. Which would help to define the RFC" "Due to smaller RFCs, it will help to assess the RFC since there will be less to assess"
P2	Authorise Change and Build Test	"Several conversations between the developers and operators will help to understand the build and test needed for the change. Accelerating the authorization and coordination"
P4, P5, P6	Assess and Evaluate Change	"The data collected from monitoring and metrics will help to evaluate the change and to understand the complexity required for the build and testing. Helping with the authorization and coordination for the build and test" "These metrics will help to understand if there are other more important changes to move forward which may cause a delay in the current change"
P9	Authorise Change and Build Test	"Having an environment to prototype solutions will help to estimate tests and extra build that could be needed for the change. This helps to authorize the build and test phases"
P10, P11	Authorise Change Deployment Coordinate Change Deployment	"By having the continuous and automated deployment, the changes will proceed between environments automatically without any authorization, simplifying the process activities as Authorize and Coordinate Change deployment" "The continuous and automated deployment can help review the change implementation by sending logs and alerts of the implementation status and how it went. If something is wrong, the deployment will alert to take action earlier. It could also perform some change closure activities to alert certain stakeholders about the change."
P12	Authorise Change Deployment Coordinate Change Deployment	"Continuous Delivery allows having a workable package to be deployed, helping to authorize and coordinate the change deployment"
P14	Coordinate Change Build and Test	"With the automated testing, it is possible to accelerate the activities to access the change by measuring the impacts of the change, and to authorize and coordinate the change build and test since the tests are already known"
P15	All activities	"By standardizing all activities of the changes they could be automatic or standard by all the company"
P16	Coordinate Change Deployment	"Using infrastructure as code will make it possible to quickly create environments for the build and test phases. Also, in case of any issue with the environments, it is easier to correct them, avoiding impact on the build and test period"
P17	All activities	"Stakeholders, by being involved in the process, will accelerate each phase because they know the delivery's priorities and impact. Therefore, their engagement is crucial for the process"

Table 9.11 - Insight on DevOps & Standard Change process (Additional Interviews)

Practice ID	Standard Change Activity	Comment
P1	Coordinate Change Implementation	"Due to the constant planning, it will help to know and to coordinate when the change can be implemented"
P8	Authorise and Schedule Change Coordinate Change Implementation	"In case this change requires the use of the deployment pipeline, the continuous integration will allow to re-check if the change goes fine on all the checkpoints of the continuous integration pipeline, assuring the correct implementation"
P10, P11, P12	Coordinate Change Implementation	"The delivery practices will ensure that the deployment is performed automatically, ensuring the quality of the implementation. This will allow us to coordinate the implementation without any risk."
P15, P17	All activities	"Standardizing all the activities will allow the organization to know the behavior of the process" "Stakeholders are key to the process; even more, if there is an emergency change, they will need to approve the changes to keep their business running. Also, they will be able to balance the pros and cons of delivering the fix"

Table 9.12 - Insight on DevOps & Emergency Change process (Additional Interviews)

Practice ID	Emergency Change Activity	Comment
P2	Assess Change Change Approval Review Approval Implement Change	“Having developers and operators will help find the best way to describe the change that needs fixing. This will help to understand the impact; therefore, it will be easier to assess, approve, and review the change” “The developers and operators can draw the implementation plan together, making the implementation easier”
P3, P4, P5, P6, P7	Review Change	“When the issue is found and the monitor between several environments, it will be easier to analyze the impact. This can help define the RFC and assess if the change is urgent enough to be applied. The logs generated by the monitoring will also help to review the change by understanding if the change was correctly implemented” “Having KPIs and metrics based on the application performance helps to review if the change was successfully implemented and the behavior of the application is expected after implementation”
P8	Implement Change	“The package for the fix is created faster and also is created automatically, reducing the human error for the package creation”
P10, P11, P12	Implement Change	“The package reaches production faster to be deployed”
P14	Change Approval	“Automated and continuous testing allows checking if the proposed solution for the emergency change is working, allowing to implement it faster”
P15	All activities	“Standardise approvals and reviews for a faster moving on the process” “Standardise the implementation between the technologies inside the company so the implementation process can be used several times and be optimized”
P16	Implement Change	“Create test environments quickly to analyze, develop, and test the solution, accelerating the implementation”
P17	All activities	“Having the stakeholders involved in the change assessment and approval may help to accelerate these activities since they know the real impact on the business by the issue” “The stakeholder can help to verify if the change was implemented successfully and if the business is not being affected anymore”

Table 9.13 - Insights on DevOps & PM process (Additional Interviews)

Practice ID	IM Activity	Comment
P2	Problem Detection	“While observing the application's behavior, the operations team could identify some issues. Then, when discussing this with the developers, it may create a new problem”
	Problem Investigation and Diagnosis Problem Resolution	“Developers and operators can discuss the issue and therefore reach a solution faster with higher quality”
P3	Problem Logging Problem Resolution	“The continuous monitor can identify some trend and issues that could help to improve the log of the problem to help to fix it”
P8	Problem Resolution	“The CI process will enable a faster development of the fix, therefore accelerating the resolution”
P10, P11, P12	Problem Resolution	“Having automation in all the deployment processes will accelerate the deployment of the fix. Therefore the delivery will be anticipated”
P15	All activities	“Standardizing all the activities will allow the organization to know the behavior of the process”
P16	Problem Resolution	“This practice is really good for quickly solving issues in the infrastructure. Moreover, it enables the version of all the infrastructure to rollback in case of any issue quickly”
P17	All activities	“Stakeholders will be responsible for moving on with this process because they do not want to have problems with their applications. So, they should be part of the process in each activity so it does not be a bottleneck”

Regarding the CM process and the normal change process, three practices stand out: P1, P15, and P17. The P1 practice is seen as impactful when creating the change and setting all the details. Due to continuous planning, it is possible to know what should be included in the change and its impacts. It also impacts the Coordination of the deployment; since there will be continuous planning of the deliveries, some changes could be rescheduled to a proper timing.

P2 is also seen as impactful but supports the authorization to start the build and test in this case. Operations are often responsible for setting up all the environments, so in this case, due to the feedback between developers and operations, the environments could be set up faster to meet the developers' needs.

Regarding monitoring and metrics practices, P3, P4, P5, P6, and P7 have impacted the changes' assessment and review. Due to constant and continuous monitoring, these practices may alert for other urgent changes, which may postpone the current change or even show that the change is not needed anymore. Also, these practices could set up alerts that check if a change was correctly implemented without causing any harm, helping to review and close the change.

The delivery practices P10, P11, and P12 impact deployment activities, such as Authorize Change Deployment and Coordinate Change Deployment. Automating the deployment package and the deployment itself will ensure that the deployment runs correctly without any issues.

P14 is impacting the Coordination for Build and Test. Having a solid test plan and automated tests when discussing the testing period is essential for the change to move forward.

Two practices that could be impacting all the process activities are P15 and P17. P15, which will standardize all the processes across the organization, will help to know how the changes flow and the main stakeholders for the process. P17 refers to having all the change interested stakeholders on board on the entire process can accelerate to move the change forward.

Regarding Standard Change, the interviewees did not make as many matches as the emergency and normal changes. However, three practices have been identified as more impactful, such as P3, P15, and P17. P3, as a continuous monitor, could check if the changes are correctly implemented or if anything is harmful to the system. P15, due to the standardization of running the process in the entire organization. Plus, P17 is for informing all interested stakeholders about the change so they can participate in any process activity.

For Emergency Changes, Table 9.8 shows more practices impacting this process than the standard change process. P2 is a practice seen as impactful in most process activities due to the collaboration between devs and ops to assess the change for an emergency until its implementation.

The monitoring and metrics practices (P3, P4, P5, P6, and P7) are seen as impactful on the review change activity to measure if the change was correctly implemented and to find any collateral impacts.

P8, P10, P11, and P12 are seen as impactful for implementing the change. Due to continuous integration and the delivery pipeline, the build of a fix will be accelerated and delivered faster, helping to implement the change successfully and ensuring the correct deployment.

The testing practices (P13 and P14) are also seen as impactful in implementing the change since they can detect issues earlier in the development stage. P16 also impacts this activity since it can easily resolve issues in the infrastructure.

In PM, besides P15 and P17 being also practices with more impact for the same reasons in the previous processes, it is possible to stand out P2 and P8. P2 for the constant feedback between operators and developers will help fix the problem most effectively. Meanwhile, P8 could accelerate the development lifecycle of the fix to ensure the problem is solved and will be deployed correctly.

Summarizing, we can see that P15 and P17 are practices that have a major impact on processes due to the standardization of the process in the organization and the shared responsibility of the stakeholders to participate in the process, help make decisions, and take action promptly. Also, communication is a key to solving issues, as seen in IM, PM, and Emergency Changes. The collaboration between developers and operators will ensure that the issue is fixed correctly based on the opinion of each team.

Regarding the monitoring and metrics practices, we can see that these practices are also vital in identifying issues faster and taking the necessary actions to avoid impacts on the business. Nevertheless, these practices can also help review and confirm that the changes were implemented correctly. Looking at other practices related to the delivery, it is possible to see that the delivery lifecycle is accelerated, enabling the delivery fixes faster to reduce the issue's impacts. Moreover, these practices ensure the correct delivery by reducing human error due to all the automation involved. With these statements, it is possible to conclude that DevOps practices can impact the ITIL processes, therefore being able to answer RQ1 from the list of research questions for this thesis.

All the practices impact at least one process. However, P9 is seen as one of the practices with fewer matches. The following section will describe the benefits and challenges of the DevOps implementations.

9.2.2. Interview Analysis for DevOps Benefits and Challenges

Based on the interviews, this section will analyze the benefits and challenges of DevOps adoption, like in Publication nr. #8, the researchers asked the interviewees about the benefits of adopting each practice. After qualitatively analyzing the interviewees' answers, the researchers summarized the responses in Table 9.14. There are only answers for practices that the interviewees employ daily.

Looking at the benefits in Table 9.14, some keywords or expressions related to benefits were highlighted. These included keywords regarding system stability, better collaboration, reduced deployment and build cycles, improved code quality, reduction of errors and business disruptions, and preventive failure detection.

Despite these benefits, adopting these practices also has challenges for organizations, as shown in Table 9.15. The interviewees were asked to rate each practice from 1 to 5 from the practice adoption point of view, where one means very hard to implement and five means very easy to implement. Table 9.15 shows the comments about the practices with an average below three, which is the neutral value. These challenges bring a trade-off to the adoption of DevOps, such as if the tool used to employ a practice is correct for the context, lack of technical knowledge to implement the practices, knowledge about the applications, and business and organization mindset.

People, technology, and processes are the three pillars of DevOps [49], [59]. The same mindset towards the organization's goals must be set in everyone working in the organization, creating a feeling of shared responsibility [59]. The organization mindset seems to be a setback that needs to be surpassed. top management is seen as a major solver for this challenge by leading by example [22]This will encourage people with different technical expertise, knowledge, and backgrounds to work together to better employ DevOps practices. This analysis answers the RQ2 that this study proposes to solve.

The following section will describe how the processes have been improved by using the DevOps practices in the interviewees' opinion.

Table 9.14 - DevOps Benefits (Additional Interviews)

ID	IM Process	CM Process	PM Process
P1	This practice provides a greater vision of the team's priorities. As a result, it is possible to fix incidents at *the proper time without endangering* the other priorities.	This practice provides *better efficiency* to the CM process. This practice allows the teams to anticipate their deliveries due to constant planning, allowing to know which changes should be created and when. This allows the teams to be prepared to discuss their changes knowing their impacts and facilitate the entire process instance.	Continuous planning enables a *better prioritization* of the problems to be solved. This *gives clarity* to the team and other stakeholders of the focus to be followed by the team.
P2	*Enhances the collaboration* between developers and operators to the common objective. This creates a *shared responsibility* between the two teams to have more stability in the system.	Regular and healthy discussions between developers and operators provide a better overview of what needs to be changed and how to achieve it. *Improving the process performance*.	The collaboration between the devs and ops is key to solving the issues and to bring *stability* to the system.
P3	It allows a continuous health status of the system so the teams can react in time to solve the issues, *reducing the business impacts*.	Due to the continuous monitoring the teams will be able monitor the system performance and behavior. *With this, the team can prioritize fixes and create changes with the correct information for an easier approval.*	These practices were identified with the same benefit for the process as finding new problems faster. Due to the monitoring and metrics, it is possible to identify issues and *actions faster than waiting for the business to identify them*. This kind of monitors could also identify trends to help on the pro-active problem detection.
P4	Metrics are used to identify faster what is wrong and provides an idea where the issue relies on.	Metrics are the key to knowing how the system is performing. Due to this is possible to know if a change should be rolled back or not, to *avoid business disruptions.*	
P5	-	Like continuous monitor, this practice enables to know how the system is behaving, providing information of what needs to be changed, *facilitating the assessment* of the change.	
P6	It gives more control of what is happening with the application producing reports where we can compare the status of the application between those reports	The interviewees have the same opinion as in P3.	
P7	These dashboards can be consulted at any time to provide the status of the application allowing to know when to created incidents and the most affected areas. This gives the opportunity to learn more on where to *stabilize* the application.	The interviewees have the same opinion as in P3.	-
P8	It offers some *comfort* to all interested parties that the developer is developing on the most recent code (which caused the incident), and once he check in the code, it will be delivered to production. It also provides a *code quality analysis and vulnerabilities to know what should be improved for a better maintenance* in the future	Continuous Integration is useful to keep the code related with the requirements. So, in case of any failure in some change, it is possible to know what should be rolled back to avoid *business disruption* and keep the *stability* of the application. From the other point of view, it will facilitate what is being added to change since the code and requirements can also be linked to the change.	This practice *accelerates the resolution of the problem* since it will ensure the new code will be according to *the code quality* guidelines.
P9	-	Due to the prototyping, it is possible to *accelerate the testing and build* phase, hence accelerating the change process.	-
P10	Even though this is not applied by the interviewees for the production environments, *the developments are delivered in the test environments faster*	The deployment will be triggered automatically which *accelerates the delivery*. However, the *coordination of the deployment will be easier* because everything is automated *and less error prone*, ensuring the probability of a successful change.	Like the continuous integration**,** *these practices accelerate the delivery of the problem resolution*. Moreover, also ensures that the package is deployable making sure the guidelines are being fulfilled.
P11	We ensure that deployment is *performed correctly automatically* without any issue since the same script is used several times reducing the *human error*.	The deployment script will be used several times in the change lifecycle, adding *more confidence* to the change deployment.	

P12	Having the software versioned in packages is quite useful for a *faster recovery* if something fails. We can rollback to the previous software version.	Due to the small chunks of changes, the delivery will be smaller which *facilitates the assessment* of the change. Which also simplifies the deployment	
P13	-	The testing phase is quite critical for a change. These are quite important to enable regression testing of what was already working before the change. This way ensures the *stability* of the application that no new errors will be added.	-
P14	-		-
P15	Process standardization is seen a mandatory practice inside the organizations so everyone knows how the process works, doesn’t matter the technology or application.		
P16	Guarantees a *good control* of the environment in terms of infrastructure to avoid having environment issues, allowing the development to go as planned.	Provides a quick way to perform changes to the infrastructure. Handling the infrastructure as code is quite beneficial to *perform quick fixes and to automate some of the most standard changes*, accelerating the process.	Proves to be highly effective in swiftly addressing infrastructure-related issues. Furthermore, it enables versioning of the entire infrastructure, facilitating prompt rollback in the event of any complications.
P17	There are several stakeholders involved on the incident, but a business representative is the key for the IM process. He should know how the application should behave and the impacts that incident is causing for the business. Therefore, its participation in the whole process is the key to bring *stability* to the application.	*Application stakeholders are the best to give clarity for the change purpose and coordination*. The application stakeholders are the ones with the best interest that the application keeps *stable* even when introducing new changes. So, they should *share the responsibility* in the change process.	Stakeholders will assume responsibility for advancing this process, as they seek to mitigate risks to their applications. Consequently, their involvement in every phase of the process is essential to prevent it from becoming a bottleneck.

Table 9.15 - DevOps Adoption Challenges (Additional Interviews)

Practices	**IM Process**					**Average**	**CM Process**					**Average**	**PM Process**					**Average**	**Comments**
	1	**2**	**3**	**4**	**5**		**1**	**2**	**3**	**4**	**5**		**1**	**2**	**3**	**4**	**5**		
P1	-	1	1	3	1	3,67	-	-	1	3	-	3,75	-	1	1	2	-	3,25	-
P2	-	1	1	1	1	3,50	-	1	-	1	1	3,67	-	-	-	3	1	4,25	-
P3	-	1	-	6	1	3,88	-	1	1	1	1	3,50	-	2	-	1	-	2,67	PM: This can be seen as hard or easy *depending on the tool*. In this case the interviewees that choose the Hard to Implement option had a tool with a lot of security issues, where they needed to replan and adopt another tool.
P4	-	1	1	5	-	3,57	-	1	2	-	-	2,67	-	1	-	2	-	3,33	CM: This is seen as hard since it is needed to know what is supposed to measure, after having the metrics well define it should be easy.
P5	-	2	-	-	-	2,00	-	1	-	-	1	3,50	-	1	-		-	2,00	IM and PM: Mostly because the interviewees didn't have the *knowledge* of what should be needed to implement
P6	-	1	-	5	1	3,86	-	1	2	-	1	3,25	-	1	-	3	-	3,50	-
P7	-	1	1	3	2	3,86	-	1	1	-	-	2,50	-	1	-	1	1	3,67	CM: The dashboards are usually based on metrics. Like in P4 it is needed to know the metrics and it should be well defined.
P8	1	-	1	5	3	3,90	-	2	1	1	-	2,75	1	1	2	1	-	2,60	CM and PM: The interviewees revealed that didn't have the *knowledge* to know how to implement
P9	-	1	1	1	2	3,80	-	-	1	-	1	4,00	-	-	2	1	-	3,33	-
P10	-	-	-	4	2	4,33	-	1	-	2	-	3,33	-	2	-	1	-	2,67	PM and CM; Since these two practices are often seen as together, for both processes, the interviewees find it hard to implement and most of the times when something goes wrong, is not easy to find where the issue is
P11	-	-	1	3	3	4,29	1	-	2	1	-	2,75	1	1	-	2	-	2,75	
P12	-	-	2	3	3	4,13	-	-	2	1	-	3,33	-	-	1	2	-	3,67	-
P13	-	1	2	1	1	3,40	-	-	1	1	-	3,50	-	1	-	1	-	3,00	-
P14	-	3	2	1	-	2,67	-	1	1	1	-	3,00	-	-	1	2	-	3,67	IM: It is not seen as very hard to implement but requires changing the *team mentality* and development process to make it mandatory for the developers to develop all the automated tests.
P15	2	-	-	1	-	2,00	-	2	-	1	-	2,67	-	1	-	2	1	3,75	IM and CM: *This needs to come from the top management* to be implemented. **A** *cultural or mindset shift is* needed to ensure that everyone will follow the process as supposed.
P16	-	4	2	1	-	2,57	-	1	2	-	-	2,67	-	-	2	-	-	3,00	IM and CM: It is needed to have a specialization to develop these scripts. Even they are useful. The interviewees revealed lack of knowledge to implement this.
P17	-	-	3	1	-	3,25	-	1	-	1	-	3,00	-	-	-	1	1	4,50	-

9.2.3. Interview Analysis for Process Performance

Likewise, in the previous section, the interviewees were asked to provide input on how these practices could improve process performance. After a qualitative analysis, Table 9.16 demonstrates the comments about each practice per process.

Table 9.16 shows that all the practices improve at least one process.

The most impacted process seems to be CM, however, for some practices this process seems to be only improving in case that the change is related with an incident like for example P8 and P16. DevOps has its roots in Agile foundations, which promotes faster and shorter release cycles, which is also beneficial to the CM process. Smaller changes are more accessible to assess and manage, providing more control of what is being changed and its impacts. Moreover, based on the collaboration between dev and ops and with different stakeholders, the CM process could be quickly moved forward because all the interested parties are involved in every process activity. Nevertheless, automation is seen as crucial for this process. Automation ensures that the implementation of the changes occurs faster and without any issues caused by humans.

About the IM process, fewer practices could be found as possible improvement. The collaboration practice for the feedback loops between devs and ops plays an essential role in investigating the issue and solving it by having the experts in the same place looking at the same issue. The monitoring practices do not improve the process because they will identify new incidents, which will not help solve them faster. It is possible to argue that these practices improve the system stability, causing a benefit to the organization and business, but not the process itself. The delivery practices and continuous integration can help move the resolution faster between the different environments, improving the process. Likewise, IaC will improve the resolution of the incidents.

PM has a similar approach to the IM process. The feedback loops between dev and ops will have the same expected result for the process performance, like the delivery practices and continuous integration. However, it has a different point of view regarding monitoring practices. Since a problem is often seen as the root cause for one or more incidents, more incidents and more information recorded could give more hints on where the root cause relies and how to solve it.

Looking at the previous statements and Table 9.16, it is possible to see that 8 of the 17 DevOps practices are improving the IM process, CM is improving all 17 practices, depending on whether the process instance is meant to solve an issue, and PM is improving 12 of the 17 DevOps practices. This answers the RQ3 in Table 1.3 in Chapter 1, showing that the processes can be improved by using the DevOps practices.

The next section synthesizes the results from these interviews and the case studies for a cross verification of the results.

Table 9.16 - DevOps Process Improvement (Additional Interviews)

Practices	IM Process	CM Process	PM Process
P1	-	Improves the process by reducing the changes size, allowing an easier assessment.	-
P2	Improves the process performance by having specialized people to fix the issue	Improves the process performance because more people with different points of view will be involved in describing and assessing the changes	Improves the process performance by having specialized people to fix the issue
P3 P4 P5 P6 P7	These practices could be seen as an improvement and not. These practices could cause more instances of the process; however, this will not make the instances solved quicker or with less resources. But could be seen as an improvement due to the logging they produce to help to resolve the issues.	These practices improve the process on the review and closing activities of the processes. These can indicate if the changes were implemented successfully accelerating those activities.	Looking from the point of view that a problem is the root cause of one or more incidents, while more incidents are created more logging or different scenarios could be found. This provides more information to help to resolve the problem.
P8	This practice improves the resolution of the incident therefore improves the process performance	This practice doesn't necessary improves the process, only if there is an incident to be solved in the change	This practice improves the resolution of the problem therefore improves the process performance
P9	-	Having a prototype of what is going to be on the changes it could accelerate some activities of the process, namely some coordination and authorizations	-
P10 P11 P12	These practices could help to improve the process performance since it will be easier to deploy a fix between different environments, reaching to the solution earlier	By accelerating the build of the developments, the activities to approve the change will also be faster. Also, by having the deployment process all automated the deployment plan will easily be approved, and the change will be implemented faster	These practices could help to improve the process performance since it will be easier to deploy a fix between different environments, reaching to the solution earlier
P13 P14	-	The testing plan is always critical to accept the changes, therefore the automation plays an important step here to move forward with the change.	-
P15	Having the process being standard by the entire organization will avoid having bottlenecks, because everyone should know how it works.		
P16	By having IaC it is possible to perform quick changes to solve incidents, namely related with the performance.	This would only impact if the change is to solve some incident	-
P17	This practice provides a performance improvement at all the incidents because the stakeholders are the key to solve any blockers that the process instance may have		

9.3. Synthetization of Extra Interviews and Publication nr. #8

The objective of this section is to synthesize the results of the case studies in publication #8 and the interviews with other professionals about DevOps practices and the ITIL processes.

To perform this, Table 9.17 will demonstrate the impact that the DevOps practices had on each process answering to the RQ1. Looking at Table 9.17, it is possible to see that the extra interviews complement some of the conclusions from the three case studies from Publication nr. #8 and to validate some of the same conclusions. In the case studies, P1 and P2 were seen to be impactful in CM and PM, and in IM and PM, respectfully, but in the extra interviews, P1 is also impacting IM, and P2 is impacting CM and showing that the practices can impact the three processes.

In the case of P3, P4, P5, P6, and P7, the extra interviews were seen to extend the impact to all three processes. IM and PM use them to detect and log new instances, while CM uses them to review and assess the changes.

For the remaining practices, the extra interviews did not bring any new conclusions. However, these interviews were positive in finding new impacts that the practices could have on the processes.

Regarding benefits, Table 9.18 synthesizes the information about the two information sources. Similar to the DevOps practices impact analysis on the processes, there were found benefits in the case studies from the Publication nr. #8 and the extra interviews performed.

The extra interviews have confirmed the benefits found in Publication nr. #8, however, there is a practice that would stand out in practice P8. In the extra interviews, this practice has been highlighted by ensuring code quality and finding code security vulnerabilities. In today's world, there have been more security breaches in the organization's applications [60]. This practice adds another security level to protect organizations and their application stability, adding one more benefit to the ones found in this research.

Nevertheless, P12 brings a perspective of not just shortening the delivery lifecycle by having a package ready for deployment but also making it easier to have a package in case something goes wrong. It is possible to rollback the application to a previous version faster and easier, giving an opportunity for a faster recovery of a failed change.

The same challenges were identified in the two different data sources. The reported challenges, such as lack of technical or business knowledge to implement some practices, could be solved by the collaboration inside of the team. Having developers and operators working together could help solve the lack of technical knowledge, but it does not solve the lack of business skills. This brings the opportunity not only to have DevOps but to extend to BizDevOps to bring business together with developers and operators [61], having an autonomous and cross-functional team.

Table 9.17 - DevOps Practices and Process Synthetization about Impact

Practices	Publication nr. #8	Section 9.2.1
P1	More impact was seen in CM and PM to plan the changes that are needed to the IT services. Likewise its corrections needed to be performed to correct problems	Continue to show impact in CM. It is also seen in IM to classify the incidents correctly in case that something was not considered correctly
P2	Positive impact in IM and PM to help to investigate the issues and how to tackle them	Found positive feedback in PM to correct the issues while for CM it could impact to prepare the environments so the build and test phases can proceed
P3	Only found impact in IM to find issues earlier avoiding extending the business impacts	These practices could be found in all the three processes. They are causing impacts to detect the issues for PM and IM. However, for PM there is possible to see that these practices also help on the resolution by enriching the information. In case of CM, these practices cause more impact to the change assessment and review
P4		
P5	Not found in any process	
P6	Impact in both IM and PM to find issues earlier.	
P7	Only found impact in IM to find issues earlier avoiding extending the business impacts	
P8	Impact seen in IM and PM to accelerate the resolution of the issue, while for CM it could impact positively the implementation of the change, and the authorizations required	
P9	Found impacts in PM where the developers can check with the stakeholders if the issue is correctly fixed	Not found in any process
P10, P11, P12	Impact seen in IM and PM to accelerate the resolution of the issue, while for CM it could impact positively the implementation of the change, and the authorizations required	
P13	Only found in CM to help to test the changes	
P14	Found in the three processes to test the fixes and changes	Only found in the CM Process
P15	This practice was found in all the processes	
P16	Only found in IM and Emergency change to fix an issue quicker	
P17	This practice was found in all the processes	

Table 9.18 - DevOps Practices and Benefit Synthetization

Practices	**Publication nr. #8**	**Section 9.2.2**
P1	*Better collaboration* between Business and IT leading to more *customer satisfaction*	*Better prioritization* for what needs to be solved
P2	*Better collaboration* between the IT teams leading to a *better delivery and process* management	
P3	*More stability* when issues are raised earlier	*More stability and facilitates the assessment* of changes
P4, P5, P6, P7	*More stability* when issues are raised earlier	
P8	Allows a *faster build and reliable* software	*Ensure the code quality*, leading to find code security vulnerabilities *allowing more stability*
P9	Enables the stakeholders to validate the delivery as it should be leading to *customer satisfaction*	*Accelerate the delivery process* namely for build and testing phases
P10	*Faster delivery cycles*	
P11	*Human error reduction*	
P12	*Faster delivery cycles*	*Faster recovery*
P13	The code being tested continuously *brings more confidence to the delivery*	*More stability* on the application since the code is tested more often
P14	*Ensure the code quality* by not introducing nonbreaking changes	
P15	Standard processes give *more efficiency to the process*	
P16	*Faster fixes to the applications*	
P17	Stakeholders are the key to move with the process and *guarantee the stability* of the application	

The other challenge is the organizational cultural mindset of openness to change and collaboration. The study participants say that the main key to overcoming this comes from the organization's top management, who must lead this change and lead by example.

Table 9.19 synthesizes the process performance improvements. Comparing the two data sources, the extra interviews added more information to complement the findings in Publication nr. #8.

This can be seen in practices P1 to P7, where more detailed information about the CM process was added. The practices P3 to P7 were not considered to cause a process improvement; however, in the extra interviews, the participants found that it could improve the logging of issues, improving the investigation and analysis. The P9 practice was seen to improve the CM process since having a prototype of the application or functionality will help to accept what was developed, accelerating the authorization and coordination activities of the process.

Table 9.19 - DevOps Practices and Process Improvement Synthetization

<table>
<tr><th>Practices</th><th>Publication nr. #8</th><th>Section 9.2.3</th></tr>
<tr><td>P1</td><td>Ensures tracking on problems not letting the PM instances take too long</td><td>By continuous planning it is possible to reduce the changes size improving the way they are assessed</td></tr>
<tr><td>P2</td><td>Collaboration to find the solutions for the issues faster improving the process performance.</td><td>Having the experts about the changes in the room helps to detail and assess the changes correctly</td></tr>
<tr><td>P3, P4, P5, P6, P7</td><td>-</td><td>These practices can see as an improvement to the processes when it comes to log the issues and to review the changes implementations</td></tr>
<tr><td>P8</td><td colspan="2">Faster development of fixes improves the three-process performance delivery and approval activities</td></tr>
<tr><td>P9</td><td>-</td><td>Having a prototype of what is going to be on the changes it could accelerate some activities of the process, namely some coordination and authorizations</td></tr>
<tr><td>P10</td><td colspan="2" rowspan="3">Faster development of fixes improves the three-process performance delivery and approval activities</td></tr>
<tr><td>P11</td></tr>
<tr><td>P12</td></tr>
<tr><td>P14</td><td colspan="2">Accelerates the testing activities impacting all the processes</td></tr>
<tr><td>P15</td><td colspan="2">Improves the process efficiency by employing the process in the same way in the entire organization</td></tr>
<tr><td>P16</td><td colspan="2">Will be able to solve issues quickly improving the process performance</td></tr>
<tr><td>P17</td><td colspan="2">Stakeholders are the face between IT and business so they can improve the process performance by being the facilitator between these two units</td></tr>
</table>

By analyzing Tables 9.17, 9.18, and 9.19, it is possible to observe different conclusions between the two data sources. This highlights the richness of qualitative analysis, as it allows for multiple perspectives and interpretations that go beyond the specific context of each organization. While the case studies were conducted in distinct organizations, each with its own contextual limitations, the additional interviews demonstrate that similar results can be found across different industry contexts. Showing that industry context is not a limitation to apply DevOps in a ITIL environment. Moreover, these interviews provide further insights, contributing to a broader understanding of the impact of DevOps practices on ITIL processes performance.

This ends this Chapter about the last publication and its extra conclusions. The next chapter will show the conclusions of the thesis.

CHAPTER 10

Conclusions

This chapter concludes the investigation regarding this research with a summary of all the findings and the research objectives achieved. This chapter is split into two sections. The first section will present the discussion and contributions that the thesis brings to the academic and scientific community. Next, the researchers will present the limitations that they faced while conducting this study and future work to complement this research.

10.1. Discussion and Contributions

The investigation conducted in this thesis aims to clarify how DevOps and ITIL can coexist by showing the impacts that a DevOps implementation causes on the ITIL processes. This complements the body of knowledge about the impacts of implementing DevOps in this type of organization's IT landscape.

From a chronologic point of view, ITSM and ITIL started to be adopted before DevOps, so it would make sense to analyze the impacts that DevOps could have on the ITIL processes already in place in the organization. The IT department has become one of the most essential departments in the organization due to all the support that is provided to all other departments in the organization. This requires managing how IT provides this support. ITSM and ITIL focus on having the IT department be seen as a service provider, measuring how it answers to the other department's needs. Due to this, ITIL processes have been implemented in organizations to measure how IT is serving the organization.

Nevertheless, these processes can be seen as bureaucratic and focused on process performance, which prevents IT and business from being agile enough to evolve with market needs.

The organization's market has become more competitive and demanding, requiring organizations to adapt their business as the market feedback requires, forcing organizations to move to Agile practices. Thus, the Agile mindset brings business and IT together in short development cycles so feedback can be implemented faster. However, ITIL plays a more significant role in the application's operability; when the development cycle is shortened, it will stress the application's stability by introducing new changes to the application.

This would motivate companies to adopt DevOps practices, in which operations work together with the development teams to create a faster delivery pipeline and ensure the application's stability.

Due to this motivation, this thesis aims to determine the impacts of implementing DevOps practices on the ITIL processes. To this end, several studies were performed to reach conclusions.

Three SLRs were performed to find the motivation for the thesis objective. One SLR aimed to find the benefits and challenges of ITSM implementations and to find gaps and opportunities to solve those implementation challenges. This research found some opportunities, like implementing cloud computing and DevOps, reinforcing the motivation for this thesis. The other SLR had the objective of finding DevOps implementations and what were the issues that were trying to solve, resulting in a list of benefits that DevOps could bring to new implementations. Based on these statements, it was possible to conceive the research concept shown in Figure 1.1 in Chapter 1. Another SLR was made to find DevOps implementations in ITIL environments to know what other researchers have done so far. From this SLR, it was possible to conclude that there were no implementations, but it showed indicators that DevOps could improve ITIL processes like IM, PM, and CM.

Other studies were also performed to show the implementation of DevOps practices in two organizations that follow the IM and PM processes. These studies showed that implementing DevOps practices was a success, benefiting process performance.

Other studies about DevOps were made for research background purposes to explore this topic more deeply. The list of all the publications can be seen in Table 1.4 in Chapter 1.

Based on these statements, this thesis focused on finding the impacts of DevOps implementations in IM, PM, and CM processes through the case study research methodology. This methodology is expected to be done based on a question defined in Chapter 1, which was decomposed into three research questions. Therefore, it was decided to perform three case studies in three organizations, one per process. The primary data collection for the case studies was Semi-structured interviews to guarantee that the interviewees had the liberty to express their thoughts about the question.

Looking at Publication nr. #8 it is possible to see that the DevOps practices caused positive impacts to the activities in the processes, revealing some benefits for the organization. Moreover, the interviewees were asked to highlight some challenges they faced when adopting these practices. Last, the participants were also asked if these practices have improved the process performance. Having these three case studies, one per process was a good indicator that the DevOps practices could positively impact the ITIL processes.

However, only having one data source could be misleading for the study. So, the researchers decided to conduct the same interviews with other professionals who work on the same processes and have knowledge about DevOps practices. The expectation is to perform these interviews to validate the results of the case study and, if possible, extend its findings. Figure 10.1 shows the connections between the DevOps practices and ITIL processes in the two data sources.

Figure 10.1 also shows that this thesis can answer the RQ1 proposed in the study. The RQ2 proposed identifying the benefits and challenges of the adoption of DevOps. Thus, there were benefits when applying the practices to the processes, such as more collaboration, increased customer satisfaction, more stability, short release cycles, system reliability, more code quality, human error reduction, less time to recover, and fewer code vulnerabilities. All these benefits can also be found in Publication nr. #3 in Chapter 4, validating the benefits found in this research. In terms of challenges there was found the lack of technical and business knowledge and the deep-seated organizational mindset in Publication nr. #3 a few challenges in DevOps adoption are listed but not the lack of technical and business knowledge, which contributes to the existing literature.

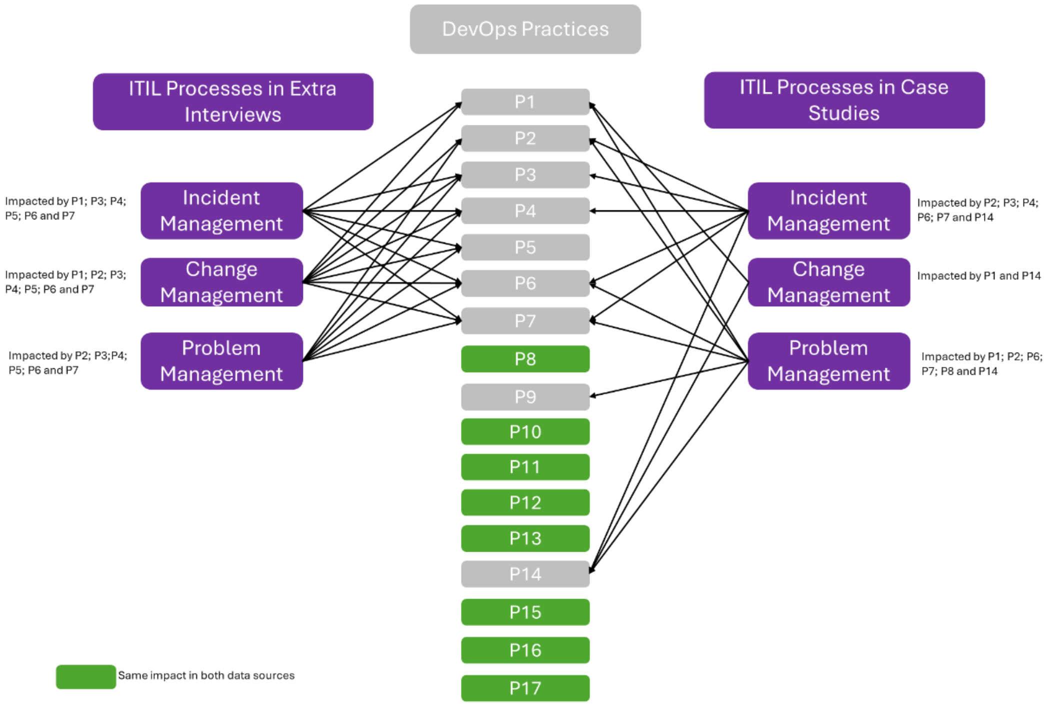


Figure 10.1 - DevOps Practices Impact in ITIL Processes

Regarding RQ3, which questions whether the processes have faced any performance improvement, the same approach was followed by asking the participants how they think the DevOps practices could impact process performance. Similar to RQ1, Figure 10.2 shows the connections between the DevOps practices and the processes regarding performance improvements.

Looking at Figure 10.2, it is possible to conclude that the extra interviews clarified the process's performance impact more than the case studies. It is possible to see more impacts in the three processes, namely on monitoring practices. This matches the qualitative gains of getting a rich analytical picture of having different data sources as Yin [54] and Thomas [56] describe.

Nevertheless, it is evident that the application of DevOps practices produces systemic impacts across the ITIL processes analyzed in this research. As previously discussed, IM, PM and CM are closely connected. The PM process can be seen from two different points of view: one focused on identifying and solving the incidents root cause, and on analyzing trends of the application behavior to prevent the occurrence or recurrence of future incidents. The information collected through incident analysis, workarounds and fixes is crucial for identifying root causes and to have an effective problem analysis.

Additionally, several activities in these processes are similar, such as Investigation and Diagnosis, and Resolution. This overlap suggests that the impact of DevOps practices on one process may implicitly influence others. Furthermore, as the delivery of fixes and workarounds becomes more efficient, the Change Management process is also affected due to the inherent synergy among the three processes.

Implementing a fix or workaround typically requires a formal change. Thanks to the high level of automation in DevOps practices, activities such as change review, assessment, and coordination are performed more rapidly and consistently, supported by the confidence of having standard deployment processes. Moreover, monitoring practices that provide data for diagnostics that support the resolution of incidents and problems also contribute to more accurate change assessments, to understand the impact on affected applications and identifying dependencies with other applications or IT services.

These observations reinforce the notion of process synergy and demonstrate that DevOps practices can have a systemic and holistic impact on ITIL processes, enhancing overall process performance.

Based on the previous statements, it is possible to conclude that all the RQs were answered successfully. This thesis brought some clarity about the impact that the adoption of DevOps practices can have on the ITIL processes, not only showing the benefits but also the possible challenges that this adoption can have. Nevertheless, this study opens the door for a new analysis of the same subject that can be found in the next section about future work.

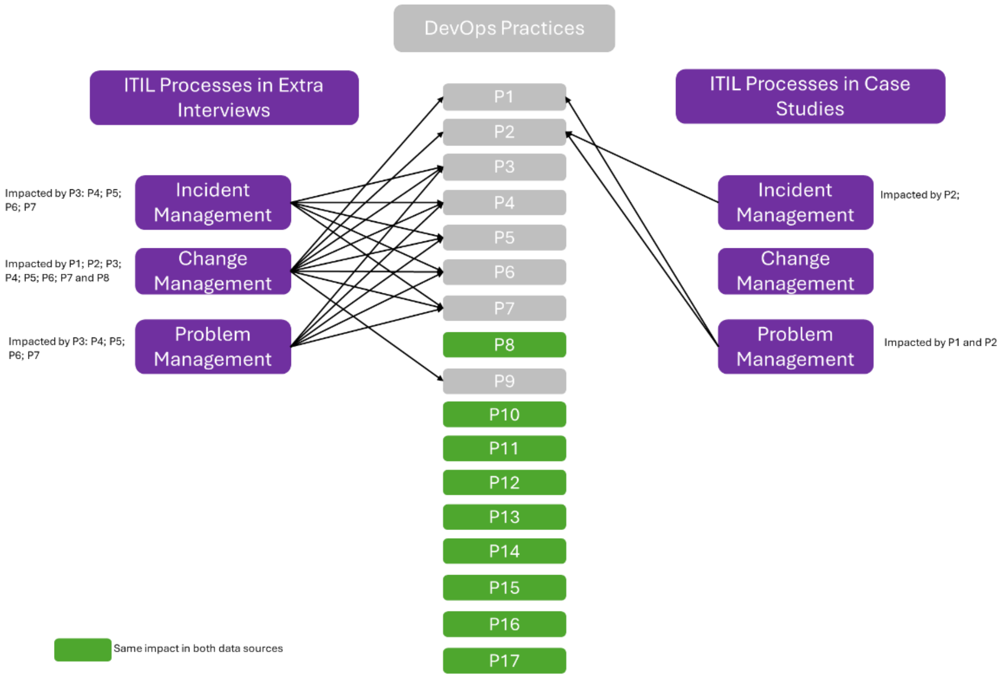


Figure 10.2 - DevOps Practices Impact on Process Performance

10.2. Limitations and Future Work

This section will present the limitations of this study and proposals for future work that other researchers could follow.

The major setback for this thesis was the amount of case studies performed. The initial plan was to perform three case studies per process in different organizations. However, many organizations denied the request to participate due to data sensitivity. More case studies with a different context could bring other perspectives about employing DevOps practices in the processes or validate what was concluded in the other case studies. The mitigation action applied for this was to have the extra interviews with other participants not from the case studies.

Also, one limitation of this study is that it only has data from the interviews, observation and focus group exercise, which is based on the opinions of the interviewees. Having different data collection methods with statistical data would bring another validation of findings, such as document analysis that could support the findings from the interviews. Usually, organizations have reports with metrics that could be helpful to notice how DevOps impacted those metrics.

Thus, the suggestion for future work would be to perform more case studies in different organizations, involving a triangulation of statistical data collection methods to validate the findings. Another suggestion would be to identify the problems before the DevOps adoption so that organizations know which problems DevOps adoption can help with. It would also be interesting to study the DevOps adoption challenges and how the organization has overcome them for successful implementation. Due to the lack of quantitative metrics identified in this study, and to have a better comprehension of the integration between ITSM processes and DevOps, a suggestion for future work would also be to create a framework for "Integration Success Score". This framework could be a combination of DevOps Research and Assessment (DORA) and ITIL metrics to achieve a reliable balance between faster deliveries and stable and operational IT Services.

Artificial Intelligence Disclaimer

As English is not the first language of the PhD candidate, he utilized Grammarly, an AI-based writing assistant, to enhance the clarity and readability of this thesis. While Grammarly has been helpful in improving grammar, spelling, and overall fluency, the ideas, analyses, and interpretations presented are entirely from the PhD candidate.

References

[1] A. Kaplan, K. Busch, A. Koziolek, and R. Heinrich, "Categories of change triggers in business processes," in *Proceedings - 44th Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2018*, Institute of Electrical and Electronics Engineers Inc., Oct. 2018, pp. 252–259. doi: 10.1109/SEAA.2018.00049.

[2] A. Wahyudin, Widodo, A. F. Nasuha, and E. Junaeti, "Strategic Alignment Maturity Level Model Using Drivers of Change in a Business Environment," in *2020 6th International Conference on Science in Information Technology: Embracing Industry 4.0: Towards Innovation in Disaster Management, ICSITech 2020*, Institute of Electrical and Electronics Engineers Inc., Oct. 2020, pp. 150–154. doi: 10.1109/ICSITech49800.2020.9392036.

[3] A. A. Alsolamy, U. A. Khan, and P. M. Khan, "IT-business alignment strategy for business growth," in *2014 International Conference on Computing for Sustainable Global Development, INDIACom 2014*, 2014, pp. 364–366. doi: 10.1109/IndiaCom.2014.6828160.

[4] D. Cannon and D. Wheeldon, *ITIL Version 3 Service Operation*, 1st ed., vol. 34, no. 19. London: The Stationery Office, 2007. doi: 10.1016/j.im.2003.02.002.

[5] R. F. De Sousa Pereira and M. M. Da Silva, “A maturity model for implementing ITIL v3,” *Proceedings - 2010 6th World Congress on Services, Services-1 2010*, pp. 399–406, 2010, doi: 10.1109/SERVICES.2010.80.

[6] F. B. E Abreu, J. M. Freitas, R. De Bragança V. da Porciúncula, and J. C. Costa, “Definition and validation of metrics for ITSM process models,” *Proceedings - 7th International Conference on the Quality of Information and Communications Technology, QUATIC 2010*, pp. 79–88, 2010, doi: 10.1109/QUATIC.2010.13.

[7] J. Aguiar, R. Pereira, and V. Braga Vasconcelos, “An Overlapless Incident Management Maturity Model for Multi-Framework Assessment (ITIL, COBIT, CMMI-SVC),” *IJIKM*, vol. 13, pp. 137–163, 2018, doi: 10.28945/4083.

[8] S. Galup, R. Dattero, and J. Quan, “What do agile, lean, and ITIL mean to DevOps?,” *Commun ACM*, vol. 63, no. 10, pp. 48–53, Sep. 2020, doi: 10.1145/3372114.

[9] AXELOS, *ITIL4 Foundations*, 1st ed. Norwish: The Stationary Office, 2019.

[10] A. Ć. Hasibović, A. Tanović, and A. Granulo, “The importance of ITIL4 adoption for IT service management in insurance companies,” in *2023 46th MIPRO ICT and Electronics Convention (MIPRO)*, IEEE, May 2023, pp. 1341–1346. doi: 10.23919/MIPRO57284.2023.10159950.

[11] M. Reiter and A. Miklosik, “DIGITAL TRANSFORMATION OF ORGANISATIONS IN THE CONTEXT OF ITIL® 4,” 2020. [Online]. Available: https://www.microsoft.com/sk-sk/microsoft-365/microsoft-teams/free

[12] A. El Yamami, K. Mansouri, M. Qbadou, and E. H. Illousamen, “Multi-criteria decision making approach for ITIL processes performance evaluation: Application to a Moroccan SME,” in *2017 Intelligent Systems and Computer Vision (ISCV)*, Apr. 2017, pp. 1–6. doi: 10.1109/ISACV.2017.8054937.

[13] J. Serrano, J. Faustino, D. Adriano, R. Pereira, and M. M. da Silva, “An it service management literature review: Challenges, benefits, opportunities and implementation practices,” *Information (Switzerland)*, vol. 12, no. 3, 2021, doi: 10.3390/info12030111.

[14] A. Lahtela, M. Jäntti, and J. Kaukola, “Implementing an ITIL-based IT service management measurement system,” *4th International Conference on Digital Society, ICDS 2010, Includes CYBERLAWS 2010: The 1st International Conference on Technical and Legal Aspects of the e-Society*, pp. 249–254, 2010, doi: 10.1109/ICDS.2010.48.

[15] S. D. Galup, R. Dattero, J. J. Quan, and S. Conger, “An overview of IT service management,” *Commun ACM*, vol. 52, no. 5, p. 124, 2009, doi: 10.1145/1506409.1506439.

[16] J.-G. Lou, Q. Lin, R. Ding, Q. Fu, D. Zhang, and T. Xie, “Software analytics for incident management of online services: An experience report,” in *2013 28th*

IEEE/ACM International Conference on Automated Software Engineering (ASE), 2013, pp. 475–485. doi: 10.1109/ASE.2013.6693105.

[17] G. Kim, P. Debois, J. Willis, and J. Humble, *The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations*. IT Revolution Press, 2016.

[18] M. Ayat, M. Sharifi, S. Sahibudin, and S. Ibrahim, "Adoption factors and implementation steps of ITSM in the target organizations," *Proceedings - 2009 3rd Asia International Conference on Modelling and Simulation, AMS 2009*, pp. 369–374, 2009, doi: 10.1109/AMS.2009.114.

[19] M. Sharifi, M. Ayat, A. A. Rahman, and S. Sahibudin, "Lessons learned in ITIL implementation failure," in *Proceedings - International Symposium on Information Technology 2008, ITSim*, 2008, pp. 8–11. doi: 10.1109/ITSIM.2008.4631627.

[20] W. de Kort, "What Is DevOps?," in *DevOps on the Microsoft Stack*, W. de Kort, Ed., Berkeley, CA: Apress, 2016, pp. 3–8. doi: 10.1007/978-1-4842-1446-6_1.

[21] R. Jabbari, N. bin Ali, K. Petersen, and B. Tanveer, "What is DevOps? A Systematic Mapping Study on Definitions and Practices," in *Proceedings of the Scientific Workshop Proceedings of XP2016 on - XP '16 Workshops*, 2016, pp. 1–11. doi: 10.1145/2962695.2962707.

[22] S. Sharma and B. Coyne, *DevOps For Dummies*, 2nd ed. Hoboken: John Wiley & Sons, Inc, 2014.

[23] B. Kitchenham, "Procedures for Performing Systematic Reviews," 2004, *Keele*.

[24] C. Okoli, "A guide to conducting a standalone systematic literature review," *Communications of the Association for Information Systems*, vol. 37, no. 1, pp. 879–910, 2015, doi: 10.17705/1cais.03743.

[25] J. Faustino, D. Adriano, R. Amaro, R. Pereira, and M. M. Silva, "DevOps benefits: A systematic literature review," *Softw Pract Exp*, May 2022, doi: 10.1002/spe.3096.

[26] L. E. Lwakatare *et al.*, "DevOps in practice: A multiple case study of five companies," *Inf Softw Technol*, vol. 114, pp. 217–230, Oct. 2019, doi: 10.1016/J.INFSOF.2019.06.010.

[27] J. Faustino, R. Pereira, and M. M. Da Silva, "The Influence of DevOps Practices in ITSM Processes," *International Journal of Services and Operations Management*, vol. 44, no. 3, pp. 390–407, 2023, doi: 10.1504/IJSOM.2021.10038492.

[28] M. I. Sarwar, Q. Abbas, T. Alyas, A. Alzahrani, T. Alghamdi, and Y. Alsaawy, "Digital Transformation of Public Sector Governance With IT Service Management-A Pilot Study," 2023, *Institute of Electrical and Electronics Engineers Inc.* doi: 10.1109/ACCESS.2023.3237550.

[29] L. Lema, J. A. Calvo-Manzano, R. Colomo-Palacios, and M. Arcilla, "ITIL in small to medium-sized enterprises software companies: Towards an implementation sequence,"

Journal of Software: Evolution and Process, vol. 27, no. 8, pp. 528–538, Aug. 2015, doi: 10.1002/smr.1727.

[30] M. Marrone and L. M. Kolbe, “Impact of IT Service Management Frameworks on the IT Organization,” *Business & Information Systems Engineering*, vol. 3, no. 1, pp. 5–18, Feb. 2011, doi: 10.1007/s12599-010-0141-5.

[31] B. Guilfoos and J. Triplett, “ITSM in Supercomputing Improving service delivery, reliability, and user satisfaction,” 2022, doi: 10.1145/3491418.

[32] R. Y. Pratama and S. Umaroh, “An IT Asset Governance Model Design Using COBIT 2019 And ITIL V4 Framework at BKU Itenas,” in *E3S Web of Conferences*, EDP Sciences, Feb. 2023. doi: 10.1051/e3sconf/202448402006.

[33] A. Latrache, E. Habib Nfaoui, and J. Boumhidi, “Multi agent based incident management system according to ITIL,” in *Intelligent Systems and Computer Vision (ISCV)*, IEEE, 2015, pp. 1–7. doi: 10.1109/ISACV.2015.7105552.

[34] C. Bartolini, M. Salle, and D. Trastour, “IT service management driven by business objectives An application to incident management,” in *2006 IEEE/IFIP Network Operations and Management Symposium NOMS 2006*, 2006, pp. 45–55. doi: 10.1109/NOMS.2006.1687537.

[35] Richard, F. L. Gaol, H. L. H. S. Warnars, E. Abdurachman, and B. Soewito, “Development of Web Application based on ITIL – Incident Management Framework In Computer Laboratory,” in *2019 International Conference on Information Management and Technology (ICIMTech)*, 2019, pp. 120–125. doi: 10.1109/ICIMTech.2019.8843799.

[36] R. Steinberg, *ITIL Service Operation*, Fist Editi. Norwich: The Stationery Office, 2011. [Online]. Available: http://books.google.com/books?id=w382YAAACAAJ&pgis=1

[37] P. Kush, “Difficulties in Implementing Effective Problem Management,” *Int J Sci Eng Res*, vol. 4, no. 8, p. 98, Aug. 2013.

[38] S. Rance, *ITIL 2011 Service Transition*, 1st Edition. Norwich: The Stationery Office, 2011.

[39] A. K. Kaiser, *Reinventing ITIL® in the Age of DevOps*. Apress, 2018. doi: 10.1007/978-1-4842-3976-6.

[40] M. Soni, “End to End Automation on Cloud with Build Pipeline: The Case for DevOps in Insurance Industry, Continuous Integration, Continuous Testing, and Continuous Delivery,” in *Proceedings - 2015 IEEE International Conference on Cloud Computing in Emerging Markets, CCEM 2015*, 2016, pp. 85–89. doi: 10.1109/CCEM.2015.29.

[41] M. S. Cuppett, “DevOps, DBAs, and DBaaS,” in *DevOps, DBAs, and DBaaS*, 2016, pp. 73–85. doi: 10.1007/978-1-4842-2208-9.

[42] L. E. Lwakature, “Devops Adoption and Implementation in Software Development Practice: Concept, practices, benefits and challenges,” 2017, p. 99. doi: ISBN 978-952-62-1711-6.

[43] L. Riungu-Kalliosaari, S. Mäkinen, L. E. Lwakatare, J. Tiihonen, and T. Männistö, "DevOps adoption benefits and challenges in practice: A case study," in *17th internacional Conference, PROFES 2016, Throndheim, Norway*, 2016, pp. 590–597. doi: 10.1007/978-3-319-49094-6_44.

[44] M. A. Silva, J. Faustino, R. Pereira, and M. M. Silva, "Productivity Gains of DevOps Adoption in an IT Team: A Case Study," in *Conference: 27TH INTERNATIONAL CONFERENCE ON INFORMATION SYSTEMS DEVELOPMENT*, B. Andersson, B. Johansson, C. S. Carlsson, M. L. Barry, H. Linger, and C. Schneider, Eds., Lund, 2018, p. 13.

[45] K. Beck *et al.*, "Agile Manifesto." Accessed: Jan. 09, 2020. [Online]. Available: https://agilemanifesto.org/

[46] D. Teixeira, R. Pereira, T. Henriques, M. M. da Silva, J. Faustino, and M. Silva, "A Maturity Model for DevOps," *International Journal of Agile Systems and Management*, vol. 13, no. 4, pp. 464–511, 2020, doi: 10.1504/IJASM.2020.112343.

[47] M. Waschke, *How Clouds Hold IT Together*. Apress, 2015. doi: 10.1007/978-1-4302-6167-4.

[48] S. W. Hussaini, "Strengthening harmonization of Development (Dev) and Operations (Ops) silos in IT environment through systems approach," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2014, pp. 178–183. doi: 10.1109/ITSC.2014.6957687.

[49] D. Teixeira, R. Pereira, T. A. Henriques, M. Silva, and J. Faustino, "A Systematic Literature Review on DevOps Capabilities and Areas," *International Journal of Human Capital and Information Technology Professionals (IJHCITP)*, p. 22, 2020, doi: 10.4018/IJHCITP.2020070101.

[50] A. Hemon, B. Lyonnet, F. Roew, and B. Fitzgerald, "From Agile to DevOps: Smart Skills and Collaborations," *Information Systems Frontiers*, 2019, doi: https://doi.org/10.1007/s10796-019-09905-1.

[51] S. Baškarada, "Qualitative Case Study Guidelines.," *Qualitative Report*, vol. 19, no. 40, pp. 1–25, 2014, doi: 10.7748/nr2013.05.20.5.28.e327.

[52] S. Modell, "Triangulation between case study and survey methods in management accounting research: An assessment of validity implications," *Management Accounting Research*, vol. 16, no. 2, pp. 231–254, 2005, doi: 10.1016/j.mar.2005.03.001.

[53] W. Tellis, "Information technology in a university: a case study," *Campus-Wide Information Systems*, vol. 14, no. 3, pp. 78–91, 1997, doi: 10.1108/10650749710187617.

[54] R. K. Yin, *Case Study Research: Design and Methods*. Beverly Hills, CA: Sage Publications, 1994.

[55] T. H. Li *et al.*, "Incident ticket analytics for IT application management services," *Proceedings - 2014 IEEE International Conference on Services Computing, SCC 2014*, pp. 568–574, 2014, doi: 10.1109/SCC.2014.80.

[56] G. Thomas, *How to do Your Case Study*, 2nd ed. Singapore: Sage Publications Asia-Pacific PTE LTD, 2016.

[57] T. O.Nyumba, K. Wilson, C. J. Derrick, and N. Mukherjee, “The use of focus group discussion methodology: Insights from two decades of application in conservation,” *Methods Ecol Evol*, vol. 9, no. 1, pp. 20–32, Jan. 2018, doi: 10.1111/2041-210X.12860.

[58] D. I. M. Hasni, A. Sarlan, and R. Ahmad, “Collaborative Visualization Framework for Cross-field Working Group: A Qualitative Focus Group Study,” in *2020 International Conference on Computational Intelligence (ICCI)*, 2020, pp. 256–260. doi: 10.1109/ICCI51257.2020.9247663.

[59] R. Amaro, R. Pereira, and M. M. Da Silva, “Capabilities and Practices in DevOps: A Multivocal Literature Review,” *IEEE Transactions on Software Engineering*, vol. 49, no. 2, pp. 883–901, Feb. 2023, doi: 10.1109/TSE.2022.3166626.

[60] S. M. Jeba *et al.*, “Recognition of Data Breach Method from Story Using Advanced Deep Learning,” in *2024 3rd International Conference for Innovation in Technology, INOCON 2024*, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/INOCON60754.2024.10512191.

[61] V. Stray, N. B. Moe, and R. Hoda, “Autonomous agile teams: Challenges and future directions for research,” in *ACM International Conference Proceeding Series*, Association for Computing Machinery, 2018. doi: 10.1145/3234152.3234182.

APPENDIX A - A SYSTEMATIC LITERATURE REVIEW ON DEVOPS CAPABILITIES AND AREAS

International Journal of Human Capital and Information Technology Professionals
Volume 11 • Issue 3 • July-September 2020

A Systematic Literature Review on DevOps Capabilities and Areas

Daniel Teixeira, Instituto Universitário de Lisboa, Lisbon, Portugal
Ruben Pereira, Instituto Universitário de Lisboa, Lisbon, Portugal
Telmo Antonio Henriques, Instituto Universitário de Lisboa, Lisbon, Portugal
https://orcid.org/0000-0002-0591-4712
Miguel Silva, Instituto Universitário de Lisboa, Lisbon, Portugal
João Faustino, Instituto Universitário de Lisboa, Lisbon, Portugal

ABSTRACT

Businesses today need to respond to customer needs at an unprecedented speed. Driven by this need for speed, many companies are rushing to the DevOps movement. DevOps, the combination of development and operations, is a new way of thinking in the software engineering domain that recently received much attention. Since DevOps has recently been introduced as a new term and novel concept, no common understanding of what it means has yet been achieved. Therefore, the definitions of DevOps often are only a part relevant to the concept. This research presents a systematic literature review to identify the determining factors contributing to the implementation of DevOps, including the main capabilities and areas with which it evolves.

KEYWORDS

Area, Capabilities, DevOps

1. INTRODUCTION

Making any change to a business is always a complex task and usually requires investment. As such, whenever an organization adopts any new technology, methodology, or approach, that adoption must be driven by a business need. This, in turn, leads to new working methodologies for IT projects.

Traditional and modern (agile) software development usually focuses exclusively on the software development teams. In either case, once the software has been developed, it is typically handed over to the IT operations team, which takes responsibility for its deployment, ongoing maintenance and support (Jones, Noppen, & Lettice, 2016). The Agile movement has brought together programmers, testers, and business representatives. Conversely, operations teams are isolated groups that ensure stability and enhance performance by applying practices such as the Information Technology Infrastructure Library (ITIL), which equates change to risk (Hüttermann, 2012). For Debois (2011), since both development and operations serve the same customer, the needs of both must be discussed simultaneously. Treated separately, they are like separate trains on separate tracks. No matter how fast they go, they can never meet. Due to this fact, the team commonly works in silos, which leads to a lack of information exchange. Lwakatare, Kuvaja, and Oivo (2015a) say that it is impossible to effectively transmit information about all the releases performed between two different teams in continuous release mode while de França, Jeronimo, and Travassos (2016) report that Development and

DOI: 10.4018/IJHCITP.2020070101

Operations are left to themselves and will often struggle to talk to each other, much less collaborate, and will remain mired in manual processes. The result is employees who don't work well together, software that doesn't work reliably, and customers that are thinking about moving to competitors (de França et al., 2016).

When two separate teams do Dev and Ops, problems can emerge and lack of synergies can occur. Separations on a technical and organizational level as well as the use of different tools have experienced an increase among Dev and Ops teams (Silva, Faustino, Pereira, & Silva, 2018). This bottleneck between Dev and Ops can affect and/or compromise the product's quality. According to Hussaini (2014), "leveraging the critical success factors to deliver the change must have shared objectives of Dev and Ops.". So, there is a clear disconnect as the two teams speak two different languages and have traditionally been judged according to different reward structures (McCarthy, Herger, Khan, & Belgodere, 2015).

In the midst of such evidence, DevOps emerged. DevOps applies agile and lean principles throughout the entire software supply chain. This allows a business to maximize the speed of delivery of a product or service, from the initial idea to production release and all the way up to customer feedback to improvements based on that feedback (Sharma & Coyne, 2015).

Seeing as DevOps is a recent and novel set of practices for software development (Rajkumar, Pole, Adige, & Mahanta, 2016), there is no one, agreed-upon definition of it. In a more general approach, DevOps integrates a set of characteristics and principles for software delivery that focuses on: speed of delivery, continuous testing in an environment where production takes place, being ready for shipping at any moment, continuous feedback, the ability to react to change more quickly, and teams working to accomplish a goal instead of a task (no more team boundaries causing a delay) (Sharma & Coyne, 2015). It involves an organizational paradigm shift from distributed siloed groups performing functions separately to cross-functional teams working on continuous operational feature deliveries. Instead of confining themselves to highly artificial process concepts that will never fly, organizations set up continuous delivery with small upgrades (Ebert, Gallardo, Hernantes, & Serrano, 2016).

Mohamed (2015) argues that this level of integration between development and operations will be revolutionary as releases can be driven by the business need, rather than the operational constraints. Virmani (2015) adds that this approach helps to deliver value faster and continuously, reducing problems caused by misunderstandings between team members, and help to accelerate problem resolution. In another perspective (Hüttermann, 2012), DevOps can be understood as rendering operations more agile.

Currently there is a lack of common understanding of what DevOps means for both academia and the practitioners' communities. This knowledge gap demonstrates that there is still a need for research about the DevOps phenomenon in order to examine how it impacts software development and operations areas. Based on what has been described, there is a clear opportunity to develop a Systematic Literature Review (SLR) on the subject, with the goal of deepening our understanding of what DevOps is.

2. DEVOPS AS A DIFFERENT APPROACH

A good cooperation between IT Development and IT Operation teams are viewed to be crucial in order to ensure successful deployment and operations of IT systems (Tessem & Iden, 2008). However, for historical reasons, most IT organizations are characterized by setting clear boundaries between these two teams, which have very different goals, mindsets and cultures (Gazivoda, 2018; Swanson & Beath, 1990).

According to Sharma and Coyne (2015) many organizations are not successful with software projects and their failures are related to the challenges in product development and delivery. Despite this, many companies also find that the development and delivery of software applications are crucial to their business, and that only 25 percent of companies consider their teams to be efficient

(Sharma & Coyne, 2015). This gap in efficiency leads to many losses of business opportunities. This demonstrates that even a disruptive methodology can't be perfect for every project.

Given the distinct nature and typology of the functions of each of these teams, it is easy to understand why there are some conflicts when they interact. Such conflicts are essentially related to the different focuses of both teams. Despite actively seeking collaboration from all its stakeholders, most agile projects do not extend themselves to operations people (Diel, Marczak, & Cruzes, 2016). These two teams (Operations and Development) should maintain a close and agile relationship, as it is this relationship which represents the stream of values between the business (where requirements are defined) and the customer (where value is delivered) (Kim, 2015).

However, the relationship between Dev and Ops is not always linear and transparent enough to be able to create synergies capable of overcoming new problems that appear throughout the application's life cycle. While Dev is focused on faster innovation and doing new things, Ops is mainly focused on stability, control, and predictability (Tingley & Anderson, 1986). This cultural difference between the development and operations departments has been reported to lead to conflicts. For example, developers need to get used to operation personnel not having experience with working on projects (Humble & Molesky, 2011). When development and operations are divided into different departments, some processes cross departmental boundaries. This makes it difficult to automate these processes (DeGrandis, 2011). For Debois (2011), despite the fact that both development and operations serve the same customer, the needs of both should be discussed at the same time.

According to Virmani (2015), as part of the Agile transformation in the past few years, IT organizations have introduced Continuous Integration (CI) principles into their software delivery lifecycle, which has improved the efficiency of development teams. Over time, however, it became clear that the optimization resulting from CI was not helping to make the entire delivery lifecycle efficient nor to increase the efficiency of the organization. Unless all the pieces of a software delivery lifecycle work like a well-oiled machine, the efficiency of the delivery lifecycle cannot be optimized.

In order to address the problems between the development and operations teams a new agile approach appeared, namely DevOps. DevOps has been heralded as a novel paradigm to overcome the traditional boundaries between IT Development (Dev) and IT Operations (Ops) teams (Nielsen, Winkler, & Nørbjerg, 2017). According to Riungu-Kalliosaari et al. (2016), DevOps is a set of practices intended to reduce the time between making a change to a system and this change being placed into normal production, all the while ensuring high quality. The main goal associated with this concept is to avoid common problems when operations and developers are kept as separated teams (Bezemer, Eismann, Ferme, & Grohmann, 2018).

DevOps integrates the two worlds of development and operations, using automated development, deployment, and infrastructure monitoring (Ebert et al., 2016). For Sharma and Coyne (2015), because DevOps improves the way that a business delivers value to its customers, suppliers, and partners, it's an essential business process, not just an IT capability.

3. RESEARCH METHODOLOGY

One of the major tools used to support an evidence-based paradigm in other domains is the generation of SLR, which is used to aggregate the experiences gained from a range of different studies in order to answer a specific research question (Khan, Kunz, Kleijnen, & Antes, 2004).

Our research is based on Kitchenhams Procedures for SLR (Kitchenham, 2004), complemented by the centric approach from Webster and Watson (Webster & Watson, 2002), which contains the following steps:

- **Planning:** It is necessary to confirm the need for such a review. In some circumstances systematic reviews are commissioned and in such cases a commissioning document needs to be written. It is also necessary to define the research question(s) that the systematic review will address and produce a review protocol (i.e. plan) which defines the basic review procedures;

- **Conducting:** Apply the review protocol previously designed in order to obtain studies which will be the object of the review;
- **Reporting:** The final phase of a systematic review, which involves writing up the results of the review and circulating these results to potentially interested parties.

The three SLR phases, described above, are represented in Figure 1, and have been specifically adapted to our research purposes.

We chose SLR as our Research Methodology since we wanted to summarize the existing evidence regarding DevOps' capabilities and areas, with the aim of answering the proposed Research Questions.

4. PLANNING THE REVIEW

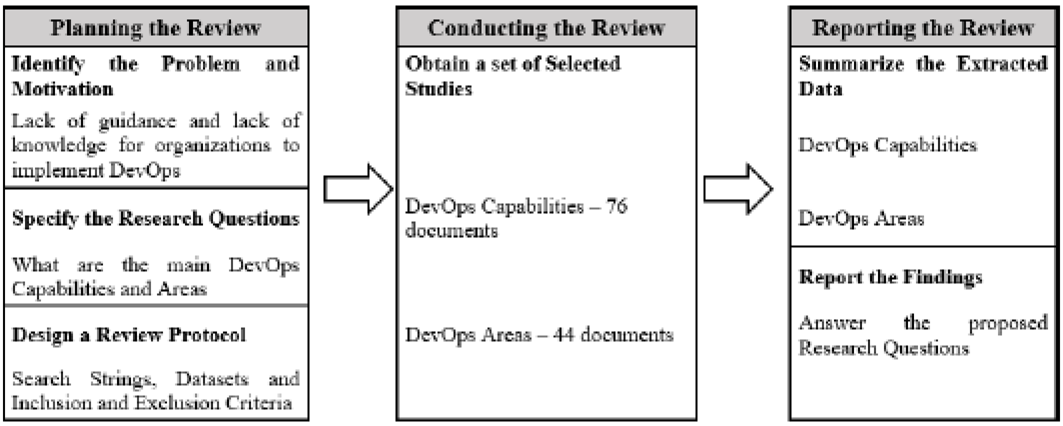
This section corresponds to the first step of the SLR Methodology. We begin by providing the Motivation for our work, followed by the Research Questions we aim to address and answer with our research. Finally, we propose our Review Protocol.

4.1. Motivation

The adoption of DevOps drives a challenging cultural shift towards collaboration and knowledge-sharing between software development, quality control and operations (Colomo-Palacios, Fernandes, Soto-Acosta, & Larrucea, 2018). The tremendous growth in demand for DevOps has, however, led to the appearance of new needs. For St, Ab, and Bosch (2017), despite wanting to implement DevOps, many companies find it difficult to understand what DevOps is and what advantages it will have. Furthermore, they ask themselves how to implement DevOps or how can they improve their DevOps practices. According to Rong, Zhang, and Shao (2016), an increasing number of software companies have adopted the DevOps paradigm in order to adapt to the ever-changing business environment.

According to Bucena and Kirikova (2017), many companies miss the maturity of the concept – with no clear definition of DevOps and its practices, no clear goals available and a lack of understanding about development workflow phases and responsibilities. There is both a lack of understanding around DevOps and a clear definition of what it is (Lwakatare, Kuvaja, & Oivo, 2015b). Therefore, organizations are not sure how to effectively implement DevOps capabilities (Qumer Gill, Loumish, Riyat, & Han, 2018).

Figure 1. SLR methodology for this research work



There are other issues, in addition to the previous ones, that inhibit the adoption of DevOps by companies. As DevOps practices affect the operations and development teams in many ways throughout the software development lifecycle, it requires both cultural and technical transformations (Kamuto & Langerman, 2018). DevOps involves a thorough cultural change. Dev and Ops are traditionally implemented by different organizational structures and are imprinted by different organizational cultures. As such, the DevOps transformation, including not only many technical aspects but also deep cultural issues — especially across two different structures — represents a major challenge (Mikkonen, Lassenius, Männistö, Oivo, & Järvinen, 2018). Lack of trust is another issue. There is both a lack of trust in the idea of DevOps itself as well as in the individuals who promote and work on the DevOps adoption process. This lack of trust results from a lack of understanding or from missing/insufficient communication. It can also be caused by a fear of changes, a fear of potential failure, and a fear of measurements, which could draw one's attention to some unpleasant areas (Bucena & Kirikova, 2017).

The disruptive nature of the changes required to adopt DevOps leads to organizational and business stress. While L. Zhu, Bass, and Champlin-Scharff (2016) consider the organizational strains as being standard for new technologies, for Bucena and Kirikova (2017) the adoption of DevOps is not trivial and can require complex changes in an enterprise's process, organization and workflow. To succeed in adopting DevOps, the enterprises should have an understanding of the different aspects that are related to the DevOps approach and have a well-thought-out strategy. They should start the adoption process with a clear idea of what actions should be performed, how they should be prioritized, what tools could support these actions, and how to measure the success of the adoption process (Bucena & Kirikova, 2017). Adopting DevOps can be affected, both negatively and positively, by an organization's structure. The way an organization is structured may influence DevOps' adoption, for example, when discussing communication, common goals and practices, decision making, and systems thinking within the organization (Lassenius, Dingsøyr, & Paasivaara, 2015).

4.2. Research Questions

In search of a systematic perspective on what is known about DevOps, a focused literature review was undertaken, in order to answer the following research questions:

RQ1: What are the main DevOps areas?
RQ2: What are the DevOps core capabilities?

4.3. Review Protocol

The review protocol starts with a literature search, with the definition of the search string that will be used in the chosen datasets in order to retrieve the maximum number of studies that may address the proposed research questions. The search string which was used and respective datasets are listed below:

Search String.
For DevOps capabilities. *DevOps AND (Capability OR Capabilities OR Practice)*
For DevOps Areas. *DevOps AND (Area, Principles, View, Dimensions and Perspective)*
Datasets. Google Scholar, ScienceDirect, IEEEXplore, ACM.

After that, inclusion and exclusion criteria must be applied to filter the obtained documents. Our criteria are presented in Table 1.

Afterwards, the first set of documents is obtained. Then, in a first phase, the abstracts must be screened to decide their relevance to the research. Finally, these documents are read in order to obtain the final selection of studies to perform the review.

The review protocol is illustrated in Figure 2.

Table 1. Inclusion and exclusion criteria

Inclusion Criteria	Exclusion Criteria
Written in English or Portuguese	Not written in English or Portuguese
Scientific papers in conferences or journals and books	Non-Free documents nor Master Thesis
Title relevance regarding DevOps	No title relevance DevOps

For easier understanding of the peers, as well as to add more scientific rigor to our research, the authors decided to follow the centric approach proposed by Webster and Watson (Webster & Watson, 2002).

5. CONDUCTING THE REVIEW

This section corresponds to the second step of the SLR Methodology. We start by applying the review protocol previously defined and perform an analysis of the extracted data.

5.1. Selection of Studies

After applying the needed search string in the listed datasets, with the inclusion and exclusion criteria presented in Figure 2, 112 papers were obtained, excluding duplicates.

Afterwards, the abstracts were read to further determine the documents' relevance. This resulted in 82 documents, which were, in turn, individually read. As a result of this process, 76 relevant studies were obtained for our research.

Figure 3 shows the number of papers found. As shown, the search conducted aims to find all papers in which DevOps capabilities has been mentioned.

After applying the needed search string in the listed datasets, with the inclusion and exclusion criteria presented in Table 1, 82 papers were obtained, excluding duplicates.

Afterwards, the abstracts were read to further decide the documents' relevance. This resulted in 46 documents. Each one of these documents was rea and 44 relevant studies were obtained for our research.

Figure 4 shows the number of papers found per database. As shown in Figure 4, the search conducted aims to find all papers in which DevOps areas have been mentioned.

Figure 2. Review protocol

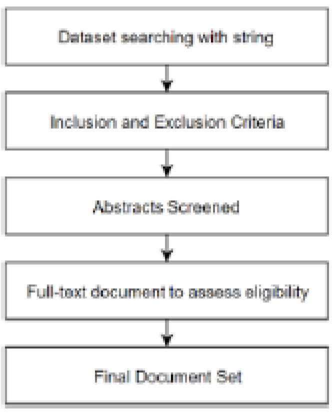
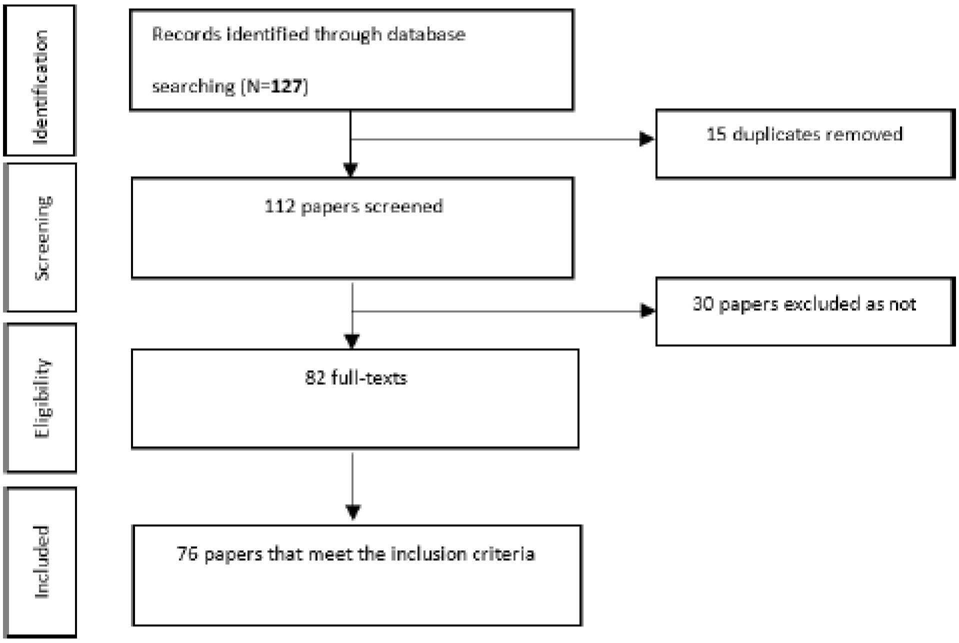


Figure 3. Search strings, databases used and results from search conducted for DevOps capabilities



5.2. Data Extraction Analysis

Looking at Figure 5 it is possible to see the distribution over the years of the articles which deal with DevOps capabilities. In 2011, only two capabilities were related with DevOps. Since then, there has been an increase in the number of documents and capabilities. This can be explained by the fact that DevOps gained popularity and the increase of interest over time would be expected, something that is reflected in the published articles. Since 2015, the quantity of documents rose slightly and in 2016 interest grew exponentially.

It is also possible to see that the interest in Continuous Integration and Continuous Deployment documents has remained above the interest in the rest of the capabilities over the years.

Looking at Figure 6, it is possible to see the distribution of the articles dealing with DevOps areas in the last years. As was pointed out in Figure 5, one can verify by looking at Figure 6 that interest in DevOps grew in 2015 and in 2016 it grew exponentially. Since then, the level of interest seems to have stabilized. The top area changed in 2018 but culture is one of the most consistent areas and has generated more interest in recent years.

Measurement, sharing and automation have maintained the same level of interest in the past three years, while the interest in technology, people and process decreased in 2018 to half of what it was in 2017.

6. REPORTING THE REVIEW

This section corresponds to the third and last step of the SLR Methodology, where we summarize the extracted data from the selected studies. We have identified two main topics, which integrate the following sub-sections: DevOps Capabilities and DevOps Areas.

Figure 4. Search strings, databases used and results from search conducted for DevOps areas

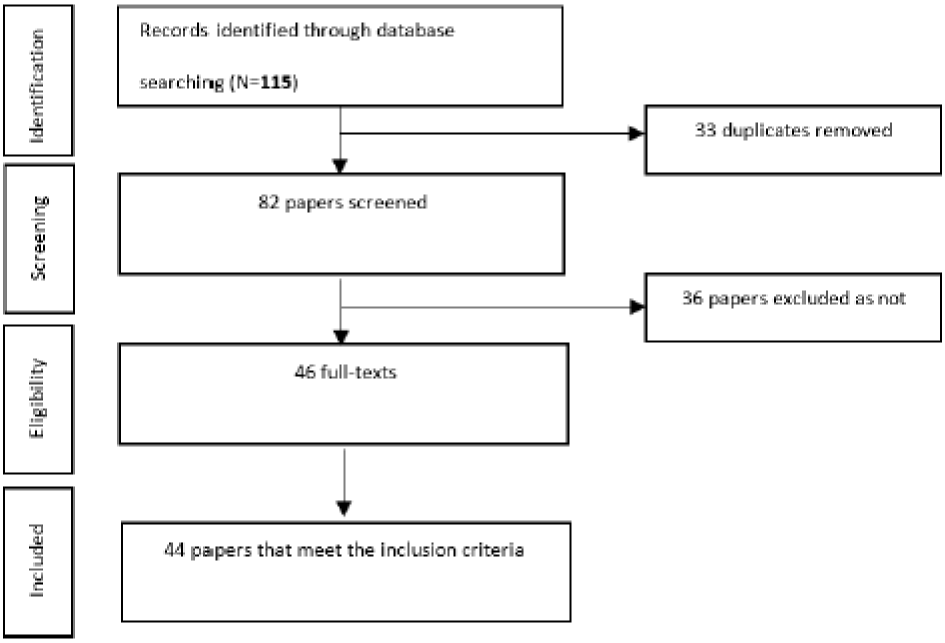
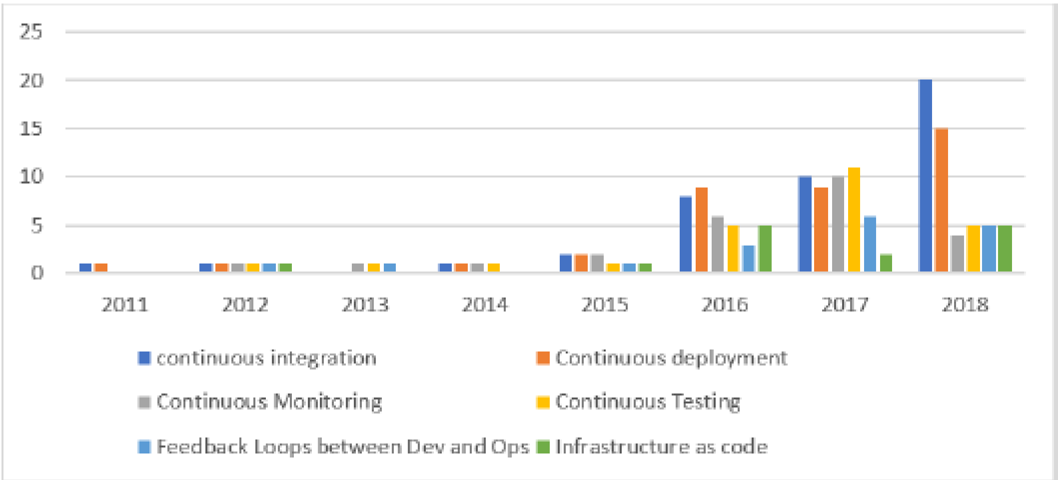


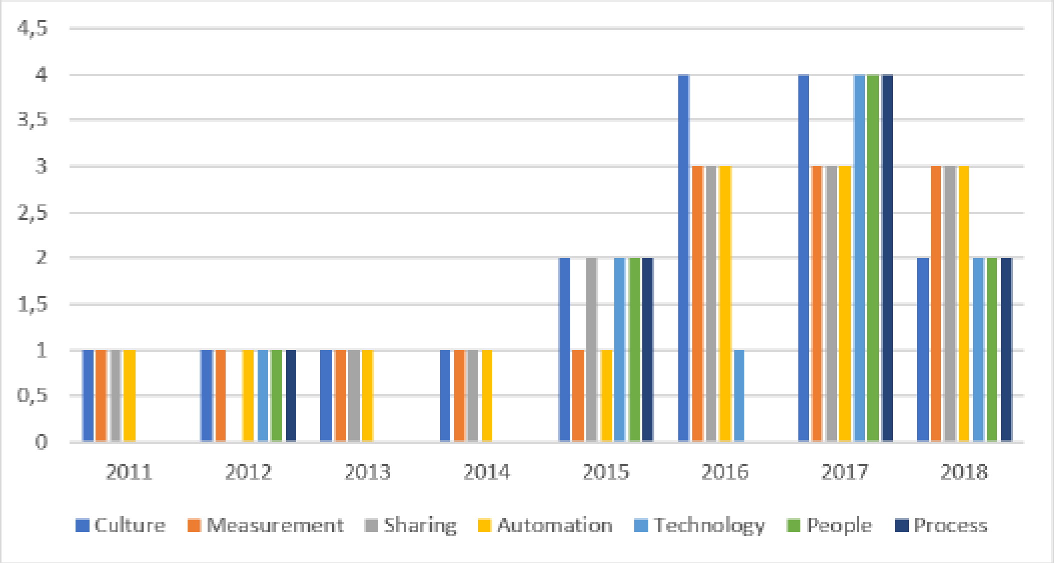
Figure 5. DevOps capabilities articles distribution per year



6.1. DevOps Capabilities

A recent study was published (Jabbari, bin Ali, Petersen, & Tanveer, 2016a) where the authors have synthesized the practices that DevOps practitioners have applied so far (Table 2). Since this study seems to be complete and the authors did not find a single DevOps' practice that was not included in Jabbari's list, the authors decided to use this list assuming that is the most completed collection of DevOps practices among the literature. Other studies related to DevOps capabilities can be found among the literature - in Hüttermann (2012); Sharma (2017a), Punjabi and Bajaj (2017), Soni (2016), and Stoneham et al. (2017). However, they are not as exhaustive as the one presented in Table 2.

Figure 6. DevOps areas articles distribution per year



Within these studies, a capability can be written in a different way, depending on its context, but maintain the same meaning. As such, the authors have grouped these capabilities together by using vectors and basing such groupings on what they have understood of the meaning of the capability. Table 2 shows the grouping that the authors made for these vectors. Although the study was already quite complete, the authors decided to carry out a literature review that could corroborate these abilities presented in the study of Jabbari et al. (2016a).

Having analyzing Table 2 and seeing that there is a considerable gap between C6 and C7, the authors decided to describe all the capabilities from C1 and C6. The description of each capacity is presented considering the various definitions which have been found.

6.1.1. Continuous Integration

The Continuous Integration (CI) concept was first practiced and described as "doing everything in parallel, with frequent synchronizations" in the 1998 book *Microsoft Secrets* (Pang & Hindle, 2017). CI consists of established practices in modern agile software development (Steffens, Lichter, & Döring, 2018a). It accommodates rapid changes (Bai, Li, Pei, Li, & Ye, 2018a) and is widely considered to be the best in software development (Debroy, Miller, & Brimble, 2018). Developers integrate their work frequently (usually each person integrates at least daily), leading to multiple integrations per day (Jabbari et al., 2016a; St et al., 2017). For Sharma and Coyne (2015), continuous integration ensures that each team's work is continuously integrated with that of other development teams and then validated. Continuous integration, thereby, reduces risk and identifies issues earlier in the software development life cycle.

Implementing CI this way ensures that bugs are caught earlier in the development cycle, which makes them less expensive to fix. Automated tests are run for every build, in order to ensure that builds maintain a consistent quality. The main objective of Continuous integration is to foster discussion and fast validation by peers (De Bayser, Azevedo, & Cerqueira, 2015). As Continuous Integration allows developers to immediately see the impact of their code changes and fix problems on the spot in the development environment, it became one of the major points of interest in the DevOps movement as smaller and more frequent changes reduced merge and integration issues (Debois, 2011).

6.1.2. Continuous Deployment

DevOps emphasizes the use of Continuous Deployment, which means deploying a number of smaller changes as soon as they are released, instead of waiting until a ‘full package’ of changes is ready, and follows directly from the practice of frequent releases. (Nielsen et al., 2017). This allows users to benefit from the changes much earlier and developers to see whether their changes work in practice (Feitelson, Frachtenberg, & Beck, 2013). To (Düllmann, Paule, & Van Hoorn, 2018) one important DevOps practice is the usage of continuous deployment as it helps to automate many steps, ranging from a source code commit to the deployment of a software artifact to production. When commonly adopted, continuous integration and continuous deployment can cause the software development lifecycle to shorten (Tuma, Calikli, & Scandariato, 2018). For Debois (2011), this capability is just like exercise: “the more you practice deployment to production, the better you will get at it.

The implementation of continuous deployment should also reduce the effort required in order to carry out a task. Many of the tasks related to the release of DevOps are being automated, and manual tasks such as configurations are being dealt with automatically. As such, the pool of resources can be released immediately after the task is completed (Kuusinen et al., 2018). There is a strong relationship between the quality of the software developed and the agility of the organization to the DevOps practices of software development. Therefore, DevOps practices contribute to the enhancement of these software quality attributes within a continuous deployment process (I. D. Rubasinghe et al., 2017).

6.1.3. Continuous Monitoring

Continuous Monitoring collects data and metrics that come from the different stages of the application lifecycle, allowing all involved parties to react quickly in order to improve or modify the functionalities which are being used (Debois, 2011; Sharma & Coyne, 2015). Effective monitoring is essential to allow DevOps teams to deliver at speed, to get feedback from production, and to increase customers’ satisfaction, acquisition and retention. By aligning development of monitoring with the development of the whole solution (implementing functional and nonfunctional requirements, building up the application, middleware, infrastructure), they will be able to improve monitoring continuously, to catch gaps in monitoring early, and to ensure that monitoring is always aligned with concrete needs (Hüttermann, 2012).

One of the major contributions is that continuous monitoring may enable early detection of quality-of-service problems, such as performance degradation, and also the fulfillment of service level agreements (Fitzgerald & Stol, 2014).

6.1.4. Continuous Testing

Continuous Testing means to test as soon as possible and continuously during the development lifecycle, leading to a development cost reduction as well as to a better software quality. This practice is viable using techniques such as test automation and virtualization, in order to simulate the production environments in which the tests are to be executed and in a scenario that is as realistic as possible (Sharma & Coyne, 2015; Soni, 2016). Also for Sharma and Coyne (2015), continuous testing is known as “shift-left testing”, which stresses integrating development and testing activities to ensure that quality is built in as early as possible in the life cycle and nothing is left behind to later instances.

The importance of this capability is that the benefits of Continuous Testing will eventually increase customer satisfaction, as the customer has a larger and more immediate impact on the product. Because the continuous deployment pipeline relies heavily on testing, the quality of the system will improve over time, as fewer bugs are introduced into the system (Kuusinen et al., 2018). This capability also permits a reduction in overall costs, shortens later testing cycles and ensures continuous feedback on quality (Nielsen et al., 2017).

6.1.5. Continuous Feedback

The goal of this practice is to get as much feedback as possible in order to perform the necessary corrections. Continuous feedback is developer – focused, which means that feedback relates to coding or architectural problems, build failures, test status and uploads of file releases (L. Zhu et al., 2016).

The new technologies provide the ability to monitor customer behavior, which allows the business team or any other interested parties to take the necessary actions to improve the software (Silva et al., 2018). Monitoring information and user feedback can be used for the purpose of improving the application and thereby enhancing the customer experience (Nielsen et al., 2017).

6.1.6. Infrastructure as Code

Infrastructure as code involves fast scaling up and down of infrastructure on demand, treating the configuration code in the same way as the application code (Rana & Staron, 2016). It also emphasizes developing automation logic for deploying, configuring and upgrading software and infrastructure repeatedly and quickly, particularly in a cloud environment (Lwakatare et al., 2015b).

Teams avoid manual environmental configuration and enforce consistency through code to represent the desired state of their environments. Deployment of infrastructure as code is repeatable and prevents runtime problems due to configuration drift or lack of dependency. DevOps teams can work with a unified set of practices and tools to deliver applications and infrastructure support quickly, reliably and on a scale. The use of infrastructure as code was recurrently cited as a means of guaranteeing that everyone knows how the execution environment of an application is provided and managed (Luz, Pinto, & Bonifácio, 2018).

6.2. DevOps Areas

This section presents the findings from a thorough literature analysis aiming to find the DevOps dimensions that characterize this phenomenon. They are either categories that work as DevOps enablers or are expected outcomes of a DevOps adoption process. Table 3 presents the main findings related to DevOps dimensions.

Because there is no standard definition of DevOps and its related processes (Silva et al., 2018) and little has thus far been presented in order to describe and formalize what it constitutes (Lwakatare et al., 2015b) the authors will now go on to detail the areas that best define DevOps practices.

Having analyzing Table 3 and determining that there is a considerable gap between A7 and A8, the authors have decided to describe all the areas from A1 to A6. The description of each capacity will be presented considering the various definitions which have been found.

6.2.1. Culture

In DevOps, there is a culture of collaboration between the software development organization and the operations organization (Lwakatare et al., 2015b) where there is joint responsibility for the delivery of high quality software (Colomo-Palacios et al., 2018). For de França, Jeronimo, and Travassos (2016) the so-called DevOps culture recognizes trust as a relevant characteristic for influencing organizational change. The culture aims to change the dynamics in which development and operational teams interact, highlighting the tasks between design and operation, such as operational design, test-driven development and continuous integration (Diel et al., 2016).

The DevOps culture encourages small, multidisciplinary teams that work independently and collectively to take responsibility for the experience of actual users of their software (Sharma & Coyne, 2015). There's no place like production for a DevOps team. All they do is improve the live experience of customers. There are no silos and no blame-game, because the team is responsible for each other. DevOps teams stress being able to move fast, understand the impact and react quickly (Hüttermann, 2012).

Table 2. DevOps capabilities literature review

ID	Capabilities	Reference	# of References
C1	Continuous Integration	(Bai, Li, Pei, Li, & Ye, 2018b; Bucena & Kirikova, 2017; Chen, Kazman, Haziyev, Kropov, & Chtchourov, 2015; Cleveland et al., 2018; Colomo-Palacios et al., 2018; Croker & Hering, 2016; De Bayser et al., 2015; de França et al., 2016; Debois, 2011; Debroy et al., 2018; Düllmann et al., 2018; Fitzgerald & Stol, 2014; Hüttermann, 2012; Jabbari, bin Ali, Petersen, & Tanveer, 2016b; Kuusinen, Balakumar, Jepsen, & Larsen, 2018; Laukkarinen, Kuusinen, & Mikkonen, 2017, 2018; Lewerentz et al., 2018; Mackey, 2018; Mansfield-Devine, 2018; Marijan, Liaaen, & Sen, 2018; Mohan & Ben Othmane, 2016; Molto, Caballer, Perez, Alfonso, & Blanquer, 2017; Moore et al., 2016; Palihawadana et al., 2017; Pang & Hindle, 2017; Punjabi & Bajaj, 2017; Rahman, Mahdavi-Hezaveh, & Williams, 2018; Rodríguez et al., 2018; Rubasinghe, Meedeniya, & Perera, 2017; I. Rubasinghe, Meedeniya, Perera, & Practice, 2018; Shahin, Babar, & Zhu, 2016; Sharma, 2017a; Shivakumar, 2017; Snyder & Curtis, 2017; Soni, 2016; Steffens et al., 2018a; Stoneham et al., 2016; Tuma et al., 2018; Vassallo et al., 2017; Wiesche, 2018; Wongkampoo & Kiattisin, 2018; Xia, Zhang, Wang, Coleman, & Liu, 2018; H. Zhu & Bayley, 2018)	*44*
C2	Continuous Deployment	(Ali, Caputo, & Lawless, 2017; Bass, 2017; Bhattacharjee, Barve, Gokhale, & Kuroda, 2018; Bucena & Kirikova, 2017; Chen et al., 2015; Cleveland et al., 2018; Debois, 2011; Debroy et al., 2018; Düllmann et al., 2018; Farshchi, Schneider, Weber, & Grundy, 2015; Fitzgerald & Stol, 2014; Fördős & Cesarini, 2016; Hüttermann, 2012; Jabbari et al., 2016b; Karapantelakis et al., 2016; Kuusinen et al., 2018; Laukkarinen et al., 2018; Mackey, 2018; Mansfield-Devine, 2018; Mohan & Ben Othmane, 2016; Palihawadana et al., 2017; Pang & Hindle, 2017; Perera, Bandara, & Perera, 2017; Punjabi & Bajaj, 2017; Rahman et al., 2018; Rana & Staron, 2016; Rubasinghe et al., 2017; Rubasinghe et al., 2018; Shahin et al., 2016; Sharma, 2017a; Shivakumar, 2017; Soni, 2016; Steffens, Lichter, & Döring, 2018b; Steffens et al., 2018a; Stoneham et al., 2016; Tuma et al., 2018; Ur Rahman & Williams, 2016b; Wiesche, 2018; Xia et al., 2018; Zhu & Bayley, 2018)	*40*
C3	Continuous Monitoring	(Bai et al., 2018b; Bucena & Kirikova, 2017; Chen et al., 2015; de França et al., 2016; Debois, 2011; Düllmann et al., 2018; Fitzgerald & Stol, 2014; Hanappi, Hummer, & Dustdar, 2016; Hüttermann, 2012; John et al., 2015; Karapantelakis et al., 2016; Kuusinen et al., 2018; Li, Zhang, & Liu, 2017; Pang & Hindle, 2017; Perera, Bandara, et al., 2017; Rana & Staron, 2016; Roche, 2013; Rubasinghe et al., 2017; Rufino, Alam, & Ferreira, 2017; Sharma, 2017a; Shivakumar, 2017; Snyder & Curtis, 2017; Soni, 2016; Steffens et al., 2018b; Ur Rahman & Williams, 2016b; Vassallo et al., 2017)	*26*
C4	Continuous Testing	(Bucena & Kirikova, 2017; Chen et al., 2015; Croker & Hering, 2016; de Feijter, Rob, Jagroep, Overbeek, & Brinkkemper, 2017; Debois, 2011; Fitzgerald & Stol, 2014; Hüttermann, 2012; Jabbari et al., 2016b; Kuusinen et al., 2018; Nielsen et al., 2017; Palihawadana et al., 2017; Pang & Hindle, 2017; Punjabi & Bajaj, 2017; Roche, 2013; Rubasinghe et al., 2018; Samarawickrama & Perera, 2018; Shahin et al., 2016; Sharma, 2017a; Shivakumar, 2017; Silva et al., 2018; Snyder & Curtis, 2017; Soni, 2016; St et al., 2017; Stoneham et al., 2016; Vassallo et al., 2017; Wiesche, 2018)	*26*
C5	Feedback Loops between Dev and Ops	(Bucena & Kirikova, 2017; de Feijter et al., 2017; Debois, 2011; Debroy et al., 2018; Hanappi et al., 2016; Hüttermann, 2012; Jabbari et al., 2016b; John et al., 2015; Kuusinen et al., 2018; Mikkonen et al., 2018; Nielsen et al., 2017; Pang & Hindle, 2017; Roche, 2013; Sharma, 2017a; Silva et al., 2018; St et al., 2017; Stoneham et al., 2016; Wongkampoo & Kiattisin, 2018)	*18*
C6	Infrastructure as code	(Bhattacharjee et al., 2018; Bucena & Kirikova, 2017; De Bayser et al., 2015; de França et al., 2016; Debois, 2011; Debroy et al., 2018; Düllmann et al., 2018; Fördős & Cesarini, 2016; Hüttermann, 2012; Jabbari et al., 2016b; Jimenez et al., 2017; Rahman et al., 2018; Rana & Staron, 2016; Shahin et al., 2016; Sharma, 2017a; Steffens et al., 2018b, 2018a)	*17*
C7	Change Management	(Abdelkebir, Maleh, & Belaissaoui, 2017; Debois, 2011; Hüttermann, 2012; Jabbari et al., 2016a; Mohamed, 2015; Rubasinghe et al., 2017; Science & Sciences, 2015; Sharma, 2017b; Zhu & Bayley, 2018)	*9*
C8	Continuous planning	(Fitzgerald & Stol, 2014; Hüttermann, 2012; Jabbari et al., 2016a; Kuusinen et al., 2018; Pang & Hindle, 2017; Sharma, 2017b; Ur Rahman & Williams, 2016a)	*7*
C9	Prototyping application	(Cleveland et al., 2018; De Bayser et al., 2015; Fitzgerald & Stol, 2014; Hüttermann, 2012; Jabbari et al., 2016a; Sharma, 2017b)	*6*
C10	Process Standardization	(Hüttermann, 2012; Jabbari et al., 2016a; Rana & Staron, 2016; Roche, 2013; Sharma, 2017b)	*5*
C11	Stakeholder Participation	(Hüttermann, 2012; Jabbari et al., 2016a; Sharma, 2017b)	*3*
C12	Shift Left	(de Feijter et al., 2017; Hüttermann, 2012; Sharma, 2017b)	*3*

6.2.2. Measurement

The ability to measure the development process by incorporating different metrics will help increase efficiency in product development (Lwakatare et al., 2015b). Based on data rather than instinct, decisions lead to an objective and irreproachable path to improvement. The data should be transparent, accessible to everyone, meaningful and capable of being viewed ad hoc. Furthermore, measurement includes monitoring high-level business metrics such as revenue or end-to-end transactions per unit time (Debois, 2011).

At a lower level, it requires careful choice of key performance indicators, since people change their behavior according to how they are measured (Nielsen et al., 2017). DevOps use various forms of measurements and monitoring which include business metrics (e.g. revenue) to metrics for a technical overview (Rana & Staron, 2016).

6.2.3. Automation

It is believed that manual, and repetitive tasks can be automated to reduce unnecessary effort and improve software delivery. Hence, automation would improve not only the delivery speed, but also the infrastructure consistency, productivity of teams, and repeatability of tasks (de França et al., 2016). Automation is used not just to save time, but it also prevents defects, creates consistency, and enables self-service. Automation is one of the main areas of DevOps: it allows for capabilities such as continuous integration and continuous deployment (Mohamed, 2015). Although transparency and sharing can be used to ensure collaboration even in manual tasks, with automation the points where silos may arise are minimized (Luz et al., 2018).

6.2.4. Technology

Technology enables people to focus on high-value creative work while delegating routine tasks to automation. Technology also allows teams of practitioners to leverage and scale their time and abilities (Sharma & Coyne, 2015).

A technology stack and tools are used to quickly and reliably operate and develop applications. These tools also help engineers carry out tasks independently (e.g. code deployment and infrastructure supply), which would normally require the assistance of other teams, and this further increases the speed of the team (Hüttermann, 2012).

6.2.5. People

The relations between colleagues should be based on trust and confidence. Transparency should be considered the rule of thumb for a DevOps team. The members of the team should also have common goals and incentives. and not only developers for delivering in time (Silva et al., 2018). For Sharma & Coyne (2015), people are the main characters of DevOps culture.

6.2.6. Process

The DevOps process can be considered a business process because it aims to affect the entire lifecycle of an application as being a collection of activities or tasks that produce a specific result for customers (Hüttermann, 2012). When the DevOps approach is in place within an organization, all parties involved from the highest level of the business down to the operations should be able to have transparency and cooperate in the entire lifecycle of a change (Silva et al., 2018).

6.3. Relationship Between DevOps Capabilities and Areas

The authors decided to relate the articles of the DevOps capabilities and areas to see if there is any relationship between the two types of concepts. By analyzing Figure 7, the relationship between DevOps and the two concepts can be seen.

Table 3. DevOps areas literature review

ID	Area	References	# of References
A1	Culture	(Bang, Chung, Choh, & Dupuis, 2013; Bucena & Kirikova, 2017; Colomo-Palacios et al., 2018; de França et al., 2016; Debois, 2011; Diel et al., 2016; Erich, Amrit, & Daneva, 2014; Gupta, Kapur, & Kumar, 2017; Hüttermann, 2012; Jabbari et al., 2016b; Lassenius et al., 2015; Nielsen et al., 2017; Perera, Silva, & Perera, 2017; Rana & Staron, 2016; Sharma & Coyne, 2015; Silva et al., 2018)	16
A2	Measurement	(Bang et al., 2013; Colomo-Palacios et al., 2018; de França et al., 2016; Debois, 2011; Erich et al., 2014; Gupta et al., 2017; Hüttermann, 2012; Jabbari et al., 2016b; Lassenius et al., 2015; Luz et al., 2018; Nielsen et al., 2017; Perera, Silva, et al., 2017; Rana & Staron, 2016; Silva et al., 2018)	14
A3	Sharing	(Bang et al., 2013; Colomo-Palacios et al., 2018; de França et al., 2016; Debois, 2011; Erich et al., 2014; Gupta et al., 2017; Hüttermann, 2012; Jabbari et al., 2016b; Lassenius et al., 2015; Luz et al., 2018; Nielsen et al., 2017; Perera, Silva, et al., 2017; Rana & Staron, 2016; Silva et al., 2018)	14
A4	Automation	(Bang et al., 2013; Colomo-Palacios et al., 2018; de França et al., 2016; Debois, 2011; Erich et al., 2014; Gupta et al., 2017; Hüttermann, 2012; Jabbari et al., 2016b; Lassenius et al., 2015; Luz et al., 2018; Mohamed, 2015; Nielsen et al., 2017; Perera, Silva, et al., 2017; Rana & Staron, 2016; Silva et al., 2018)	14
A5	Technology	(Abdelkebir et al., 2017; Bucena & Kirikova, 2017; Diel et al., 2016; Gazivoda, 2018; Hussain, Clear, & MacDonell, 2017; Hüttermann, 2012; McCarthy et al., 2015; Sharma & Coyne, 2015; Silva et al., 2018; Sturm, Pollard, & Craig, 2017)	10
A6	People	(Abdelkebir et al., 2017; Bucena & Kirikova, 2017; Gazivoda, 2018; Hussain et al., 2017; Hüttermann, 2012; McCarthy et al., 2015; Sharma & Coyne, 2015; Silva et al., 2018; Sturm et al., 2017)	9
A7	Process	(Abdelkebir et al., 2017; Bucena & Kirikova, 2017; Gazivoda, 2018; Hussain et al., 2017; Hüttermann, 2012; McCarthy et al., 2015; Sharma & Coyne, 2015; Silva et al., 2018; Sturm et al., 2017)	9
A8	Quality	(Erich et al., 2014; Luz et al., 2018; Mohamed, 2015)	3
A9	Collaboration	(Luz et al., 2018; Mohamed, 2015)	2
A10	Do it yourself deployments	(Debois, 2011)	1
A11	Agility	(Luz et al., 2018)	1
A12	Resilience	(Luz et al., 2018)	1
A13	Transparency	(Luz et al., 2018)	1
A14	Services	(Erich et al., 2014)	1
A15	Structures	(Erich et al., 2014)	1
A16	Standards	(Erich et al., 2014)	1
A17	Governance	(Mohamed, 2015)	1

Unsurprisingly, some of the areas relate more to certain capabilities. For example, the articles that reference process as a DevOps area, also relate it to Feedback Loops. Culture seems to be one of the major areas, as it is the one that most relate to all capabilities. Measurement, Sharing and Automation are the areas that relate more to ability, after culture. Culture is the area that most relates to all the capabilities. Every area seems to have an impact on Feedback Loops as it influences almost all areas with the same weight. Sharing and Automation are areas that also have an important relationship with Continuous Deployment. Except in the Feedback Loops, Process is the less capability influencer area. Automation and Sharing does have an important share in all the capabilities as these areas keep a high number of documents that relate them to most of the practices. Sharing, Automation and Culture are also areas that seems to have influence in Continuous Integration.

We shall take a closer look at the areas and the capabilities that they most relate to: Culture seems to be the most related area of Continuous Testing, Continuous Integration and Feedback Loops. Measurement is significantly an influencer for Continuous Testing, Continuous Integration and Feedback Loops. Sharing and Automation are areas that seem to both relate the same with Feedback Loops, Continuous Testing, Continuous Deployment and Continuous Integration. Technology, People and Process have a close relationship with Feedback Loops and Continuous Testing.

Furthermore, it is possible to analyze the capabilities and the areas that they most relate to, and some conclusions can be reached by analyzing Figure 7. Continuous Integration and Continuous Deployment, Continuous Monitoring and Continuous Testing relate the most to culture, Measurement, Sharing and Automation. Feedback Loops seems to only have a closer relation with Culture and Infrastructure as code seems to be influenced by Culture, Measurement and Sharing.

By this analysis, it became clear that some areas tend to relate more with some capabilities than with others. One of the main reasons may be due to the fact that some areas function as an enabler for the specific capability practitioners.

7. CONCLUSION

In this research, an SLR was conducted to respond to the call by researchers and practitioners for a deeper theoretical and practical understanding of DevOps capabilities and areas that could work as determinant factors and contribute to the implementation of DevOps. Basing themselves on the previous sections the authors are able to argue that a proper answer has been provided for each proposed Research Question.

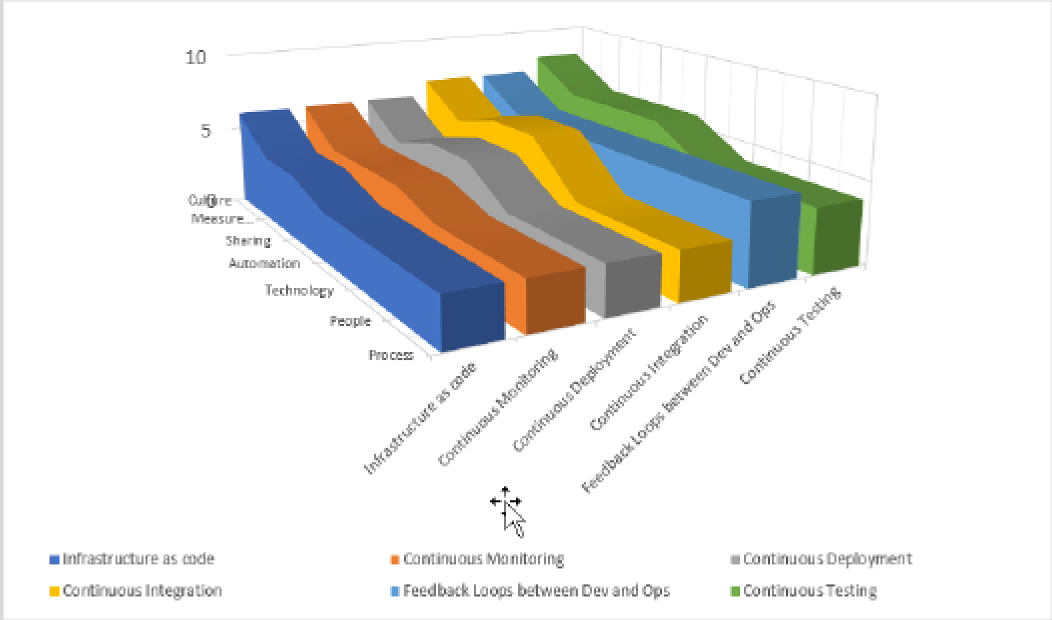
Regarding RQ1, the main DevOps areas were elicited and described, and they specifically include culture, measurement, sharing, automation, technology, people and process. Concerning RQ2, the main DevOps capabilities have been also identified and detailed. The elicited capabilities include continuous integration, continuous deployment, continuous testing, feedback loops between Dev and Ops and infrastructure as code.

As the number of DevOps practitioners continues to increase, the studies that focus on DevOps areas and its capabilities, have also increased since 2015. The authors also identified some relationships between the DevOps areas and capabilities based on the analysis of Figure 7. The documents that focus on the DevOps culture are most likely to relate it to all of the main capabilities found. On the other hand, it is more difficult to find a document that relates Technology, People and Process with the main capabilities.

The capabilities of Continuous Integration and Continuous Deployment are the more relevant in the literature. The areas that most relate with them are Culture, Sharing and Automation. These three areas are the most referred DevOps areas in the literature. Processes seems to be the area that less influences the capabilities, while Infrastructure as Code is the capability which the fewest studies tend to relate with DevOps.

This research aimed to find the main DevOps capabilities and areas. This research has brought contributions to the academic and scientific community by exploring a field that had not yet been explored. It has also improved the knowledge base and endeavored to lay down new bases for further research.

Figure 7. Relationship between DevOps capabilities and areas



This research is a new systematized contribution to knowledge, through the identification of patterns that have been recognized in the literature - and that, as such, corresponds to a new level of knowledge in the approach to the topic. This research also provides some contributions for professionals and practitioners. In the absence of studies exploring the DevOps main capabilities and DevOps areas, and even the relationship between them, this research brings new insights on how and why practitioners should adopt DevOps practices and which areas they have to change or, at least, keep in mind as being relevant for an effective adoption of DevOps.

Regarding limitations, we were not able to gather enough information and present a robust conclusion regarding specific topics, such as Outcomes, since DevOps is a recent subject. The present research cannot fully avoid biases since we excluded literature written in other languages or unavailable in electronic databases.

In the future, research should be carried out into the most referenced capabilities, Continuous Integration and Continuous Deployment and the most referenced areas, Culture, Sharing and Automation, as they seem to be essential in the DevOps movement. Also, it would be interesting to deeply explore the relationship between Continuous Integration and Culture, Sharing and Automation, as these areas seem to relate the most with the main capability found among this literature review.

Based on these findings, and using the summarized information provided in this work as a starting point, the authors will deepen the identified DevOps areas and capabilities to be an a priori and open model, which will be the target of a subsequent research project - which will aim to test and refine this systematized view (in the form of a maturity model), having not only implications for existing scientific knowledge but also being useful for organizational practices of DevOps.

REFERENCES

Abdelkebir, S., Maleh, Y., & Belaissaoui, M. (2017). An Agile Framework for ITS Management In Organizations. *Proceedings of the 2nd International Conference on Computing and Wireless Communication Systems - ICCWCS'17*. Academic Press. doi:10.1145/3167486.3167556

Ali, E., Caputo, A., & Lawless, S. (2017). Entity attribute ranking using learning to rank. *CEUR Workshop Proceedings*, *1883*, 19–24. doi:10.1145/1235

Bai, X., Li, M., Pei, D., Li, S., & Ye, D. (2018a). Continuous delivery of personalized assessment and feedback in agile software engineering projects. *Proceedings of the International Conference on Software Engineering, Part F1373* (pp. 58–67). Academic Press. doi:10.1145/3183377.3183387

Bai, X., Li, M., Pei, D., Li, S., & Ye, D. (2018b). Continuous delivery of personalized assessment and feedback in agile software engineering projects. *Proceedings of the International Conference on Software Engineering, Part F1373* (pp. 58–67). Academic Press. doi:10.1145/3183377.3183387

Bang, S. K., Chung, S., Choh, Y., & Dupuis, M. (2013). A grounded theory analysis of modern web applications. *Proceedings of the 2nd Annual Conference on Research in Information Technology - RIIT '13*. Academic Press. doi:10.1145/2512209.2512229

Bass, L. (2017). The Software Architect and DevOps. *IEEE Software*, *35*(1), 8–10. doi:10.1109/MS.2017.4541051

Bezemer, C., Eismann, S., Ferme, V., & Grohmann, J. (2018). How is Performance Addressed in DevOps? A Survey on Industrial Practices.

Bhattacharjee, A., Barve, Y., Gokhale, A., & Kuroda, T. (2018). (WIP) CloudCAMP: Automating the Deployment and Management of Cloud Services. *Proceedings of the 2018 IEEE International Conference on Services Computing (SCC)* (pp. 237–240). IEEE. doi:10.1109/SCC.2018.00038

Bucena, I., & Kirikova, M. (2017). Simplifying the devops adoption process. In CEUR Workshop Proceedings (Vol. 1898). Academic Press.

Chen, H. M., Kazman, R., Haziyev, S., Kropov, V., & Chtchourov, D. (2015). Architectural Support for DevOps in a Neo-Metropolis BDaaS Platform. *Proceedings of the IEEE Symposium on Reliable Distributed Systems (pp.* 25–30). IEEE Press. doi:10.1109/SRDSW.2015.14

Cleveland, S. B., Dooley, R., Perry, D., Stubbs, J., Fonner, J. M., & Jacobs, G. A. (2018). Building Science Gateway Infrastructure in the Middle of the Pacific and Beyond. *Proceedings of the Practice and Experience on Advanced Research Computing PEARC '18*. Academic Press. doi:10.1145/3219104.3219151

Colomo-Palacios, R., Fernandes, E., Soto-Acosta, P., & Larrucea, X. (2018). A case analysis of enabling continuous software deployment through knowledge management. *International Journal of Information Management*, *40*(October), 186–189. doi:10.1016/j.ijinfomgt.2017.11.005

Croker, M., & Hering, M. (2016). DevOps: Delivering at the speed of today's business. Accenture.

De Bayser, M., Azevedo, L. G., & Cerqueira, R. (2015). ResearchOps: The case for DevOps in scientific applications. *Proceedings of the 2015 IFIP/IEEE International Symposium on Integrated Network Management, IM 2015* (pp. 1398–1404). Academic Press. doi:10.1109/INM.2015.7140503

de Feijter, R., Rob, van V., Jagroep, E., Overbeek, S., & Brinkkemper, S. (2017). Towards the adoption of DevOps in software product organizations: A Maturity Model Approach, (May), 1–173.

de França, B. B. N., Jeronimo, H., & Travassos, G. H. (2016). Characterizing DevOps by Hearing Multiple Voices. *Proceedings of the 30th Brazilian Symposium on Software Engineering - SBES '16* (pp. 53–62). Academic Press. doi:10.1145/2973839.2973845

Debois, P. (2011). DevOps: A software Revolution in the Making. *Cutter IT Journal*, *24*(8), 34. doi:10.1016/B978-0-08-025947-5.50004-2

Debroy, V., Miller, S., & Brimble, L. (2018). Building lean continuous integration and delivery pipelines by applying DevOps principles: a case study at Varidesk. *Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering - ESEC/FSE 2018* (pp. 851–856). ACM. doi:10.1145/3236024.3275528

DeGrandis, D. (2011). Devops: So you say you want a revolution? *Cutter IT Journal*, *24*(8), 34–39.

Diel, E., Marczak, S., & Cruzes, D. S. (2016). Communication Challenges and Strategies in Distributed DevOps. *Proceedings of the 2016 IEEE 11th International Conference on Global Software Engineering (ICGSE)* (pp. 24–28). IEEE. doi:10.1109/ICGSE.2016.28

Düllmann, T. F., Paule, C., & Van Hoorn, A. (2018). Exploiting devops practices for dependable and secure continuous delivery pipelines. *Proceedings of the International Conference on Software Engineering, Part F1378* (pp. 27–30). Academic Press. doi:10.1145/3194760.3194763

Ebert, C., Gallardo, G., Hernantes, J., & Serrano, N. (2016). DevOps. *IEEE Software*, *33*(3), 94–100. doi:10.1109/MS.2016.68

Erich, F., Amrit, C., & Daneva, M. (2014). Cooperation between information system development and operations. *Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement - ESEM '14*. Academic Press. doi:10.1145/2652524.2652598

Farshchi, M., Schneider, J.-G., Weber, I., & Grundy, J. (2015). Experience report: Anomaly detection of cloud application operations using log and cloud metric correlation analysis. *Proceedings of the 2015 IEEE 26th International Symposium on Software Reliability Engineering (ISSRE)* (pp. 24–34). IEEE Press. doi:10.1109/ISSRE.2015.7381796

Feitelson, D. G., Frachtenberg, E., & Beck, K. L. (2013). Development and deployment at facebook. *IEEE Internet Computing*, *17*(4), 8–17. doi:10.1109/MIC.2013.25

Fitzgerald, B., & Stol, K.-J. (2014). Continuous software engineering and beyond: trends and challenges. *Proceedings of the 1st International Workshop on Rapid Continuous Software Engineering - RCoSE 2014* (pp. 1–9). Academic Press. doi:10.1145/2593812.2593813

Fördős, V., & Cesarini, F. (2016). CRDTs for the configuration of distributed Erlang systems. *Proceedings of the 15th International Workshop on Erlang - Erlang 2016* (pp. 42–53). Academic Press. doi:10.1145/2975969.2975974

Gupta, V., Kapur, P. K., & Kumar, D. (2017). Modeling and measuring attributes influencing DevOps implementation in an enterprise using structural equation modeling. *Information and Software Technology*, *92*, 75–91. doi:10.1016/j.infsof.2017.07.010

Hanappi, O., Hummer, W., & Dustdar, S. (2016). Asserting reliable convergence for configuration management scripts. *ACM SIGPLAN Notices*, *51*(10), 328–343. doi:10.1145/3022671.2984000

Humble, J., & Molesky, J. (2011). Why Enterprises Must Adopt Devops to Enable Continuous Delivery. *Cutter IT Journal*, *24*(8), 6–12.

Hussain, W., Clear, T., & MacDonell, S. (2017). Emerging trends for global DevOps: A New Zealand perspective. *Proceedings of the 2017 IEEE 12th International Conference on Global Software Engineering, ICGSE 2017* (pp. 21–30). IEEE. doi:10.1109/ICGSE.2017.16

Hussaini, S. W. (2014). Strengthening harmonization of Development (Dev) and Operations (Ops) silos in IT environment through systems approach. *Proceedings of the 2014 17th IEEE International Conference on Intelligent Transportation Systems, ITSC 2014* (pp. 178–183). Academic Press. doi:10.1109/ITSC.2014.6957687

Hüttermann, M. (2012). Introducing DevOps. In DevOps for Developers (Part I) (pp. 15–31). doi:10.1007/978-1-4302-4570-4_2

Jabbari, R., bin Ali, N., Petersen, K., & Tanveer, B. (2016). What is DevOps? *Proceedings of the Scientific XP '16 Workshops*. Academic Press. doi:10.1145/2962695.2962707

Jimenez, I., Sevilla, M., Watkins, N., Maltzahn, C., Lofstead, J., & Mohror, K. … Arpaci-Dusseau, R. (2017). The popper convention: Making reproducible systems evaluation practical. *Proceedings - 2017 IEEE 31st International Parallel and Distributed Processing Symposium Workshops, IPDPSW 2017* (pp. 1561–1570). IEEE Press. doi:10.1109/IPDPSW.2017.157

John, W., Meirosu, C., Pechenot, B., Sköldström, P., Kreuger, P., & Steinert, R. (2015). Scalable Software Defined Monitoring for Service Provider DevOps. *Proceedings - European Workshop on Software Defined Networks, EWSDN* (pp. 61–66). Academic Press. doi:10.1109/EWSDN.2015.62

Jones, S., Noppen, J., & Lettice, F. (2016). Management challenges for DevOps adoption within UK SMEs. *Proceedings of the 2nd International Workshop on Quality-Aware DevOps - QUDOS 2016* (pp. 7–11). Academic Press. doi:10.1145/2945408.2945410

Kamuto, M. B., & Langerman, J. J. (2018). Factors inhibiting the adoption of DevOps in large organisations: South African context. In *RTEICT 2017 - 2nd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, Proceedings* (pp. 48–51). IEEE Press. doi:10.1109/RTEICT.2017.8256556

Karapantelakis, A., Liang, H., Wang, K., Vandikas, K., Inam, R., & Fersman, E. … Giannokostas, V. (2016). DevOps for IoT applications using cellular networks and cloud. *Proceedings - 2016 IEEE 4th International Conference on Future Internet of Things and Cloud, FiCloud 2016* (pp. 340–347). IEEE Press. doi:10.1109/FiCloud.2016.55

Khan, K., Kunz, R., Kleijnen, J., & Antes, G. (2004). Systematic Reviews to Support Evidence-Based Medicine: How to Review and Apply Findings of Healthcare Research. *Postgraduate Medical Journal*. doi:10.1016/S1036-7314(00)70624-2

Kim, G. (2015). Top 11 Things You Need to Know about DevOps. Leankit. Retrieved from http://info.leankit.com/top-eleven-things-to-know-about-devops

Kitchenham, B. (2004). Procedures for performing systematic reviews. Keele University.

Kuusinen, K., Balakumar, V., Jepsen, S. C., Larsen, S. H., Lemqvist, T. A., Muric, A., & Vestergaard, O. et al. (2018, August). A large agile organization on its journey towards DevOps. *Proceedings of the 2018 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)* (pp. 60-63). IEEE. doi:10.1109/SEAA.2018.00019

Lassenius, C., Dingsøyr, T., & Paasivaara, M. (2015). Agile processes, in software engineering, and extreme programming. doi:10.1007/978-3-319-18612-2

Laukkarinen, T., Kuusinen, K., & Mikkonen, T. (2017). DevOps in regulated software development: Case medical devices. *Proceedings - 2017 IEEE/ACM 39th International Conference on Software Engineering: New Ideas and Emerging Results Track, ICSE-NIER 2017* (pp. 15–18). Academic Press. doi:10.1109/ICSE-NIER.2017.20

Laukkarinen, T., Kuusinen, K., & Mikkonen, T. (2018). Regulated software meets DevOps. *Information and Software Technology*, *97*, 176–178. doi:10.1016/j.infsof.2018.01.011

Lewerentz, M., Bluhm, T., Daher, R., Dumke, S., Grahl, M., Grün, M., & Werner, A. et al. (2018). Implementing DevOps practices at the control and data acquisition system of an experimental fusion device. *Fusion Engineering and Design*, (November). doi:10.1016/j.fusengdes.2018.11.022

Li, Z., Zhang, Y., & Liu, Y. (2017). Towards a full-stack devops environment (platform-as-a-service) for cloud-hosted applications. *Tsinghua Science and Technology*, *22*(1), 1–9. doi:10.1109/TST.2017.7830891

Luz, W. P., Pinto, G., & Bonifácio, R. (2018). Building a collaborative culture. *Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement - ESEM '18* (Vol. 18, pp. 1–10). New York: ACM Press. doi:10.1145/3239235.3240299

Lwakatare, L. E., Kuvaja, P., & Oivo, M. (2015a). Dimensions of devops. Proceedings of the International conference on agile software development (pp. 212-217). Springer. doi:10.1007/978-3-319-18612-2_19

Mackey, T. (2018). Building open source security into agile application builds. *Network Security*, *2018*(4), 5–8. doi:10.1016/S1353-4858(18)30032-1

Mansfield-Devine, S. (2018). DevOps: Finding room for security. *Network Security*, (7), 15–20. doi:10.1016/S1353-4858(18)30070-9

Marijan, D., Liaaen, M., & Sen, S. (2018). DevOps Improvements for Reduced Cycle Times with Integrated Test Optimizations for Continuous Integration. *Proceedings of the 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)* (pp. 22–27). IEEE Press. doi:10.1109/COMPSAC.2018.00012

McCarthy, M. A., Herger, L. M., Khan, S. M., & Belgodere, B. M. (2015). Composable DevOps: Automated Ontology Based DevOps Maturity Analysis. *Proceedings of the 2015 IEEE International Conference on Services Computing, SCC 2015* (pp. 600–607). IEEE Press. doi:10.1109/SCC.2015.87

Mikkonen, T., Lassenius, C., Männistö, T., Oivo, M., & Järvinen, J. (2018). Continuous and collaborative technology transfer: Software engineering research with real-time industry impact. *Information and Software Technology*, *95*(October), 34–45. doi:10.1016/j.infsof.2017.10.013

Mohamed, S. I. (2015). DevOps Shifting Software Engineering Strategy Value Based Perspective. *IOSR Journal of Computer Engineering Ver. IV*, *17*(2), 2278–2661. doi:10.9790/0661-17245157

Mohan, V., & Ben Othmane, L. (2016). SecDevOps: Is it a marketing buzzword? Mapping research on security in DevOps. *Proceedings of the 2016 11th International Conference on Availability, Reliability and Security, ARES 2016* (pp. 542–547). Academic Press. doi:10.1109/ARES.2016.92

Molto, G., Caballer, M., Perez, A., De Alfonso, C., & Blanquer, I. (2017). Coherent Application Delivery on Hybrid Distributed Computing Infrastructures of Virtual Machines and Docker Containers. *Proceedings of the 2017 25th Euromicro International Conference on Parallel, Distributed and Network-Based Processing, PDP 2017* (pp. 486–490). Academic Press. doi:10.1109/PDP.2017.29

Moore, J., Kortuem, G., Smith, A., Chowdhury, N., Cavero, J., & Gooch, D. (2016). DevOps for the Urban IoT. *Proceedings of the Second International Conference on IoT in Urban Space - Urb-IoT '16* (pp. 78–81). Academic Press. doi:10.1145/2962735.2962747

Nielsen, P. A., Winkler, T. J., & Nørbjerg, J. (2017). Closing the IT development-operations gap: The devops knowledge sharing framework. *CEUR Workshop Proceedings*. Academic Press.

Palihawadana, S., Wijeweera, C. H., Sanjitha, M. G. T. N., Liyanage, V. K., Perera, I., & Meedeniya, D. A. (2017). Tool support for traceability management of software artefacts with DevOps practices. *Proceedings of the 3rd International Moratuwa Engineering Research Conference, MERCon 2017* (pp. 129–134). Academic Press. doi:10.1109/MERCon.2017.7980469

Pang, C., & Hindle, A. (2017). Continuous maintenance. *Proceedings of the 2016 IEEE International Conference on Software Maintenance and Evolution, ICSME 2016* (pp. 458–462). IEEE Press. doi:10.1109/ICSME.2016.45

Perera, P., Bandara, M., & Perera, I. (2017). Evaluating the impact of DevOps practice in Sri Lankan software development organizations. *Proceedings of the 16th International Conference on Advances in ICT for Emerging Regions ICTer 2016* (pp. 281–287). Academic Press. doi:10.1109/ICTER.2016.7829932

Perera, P., Silva, R., & Perera, I. (2017). Improve software quality through practicing DevOps. *Proceedings of the 2017 Seventeenth International Conference on Advances in ICT for Emerging Regions (ICTer)*. IEEE. doi:10.1109/ICTER.2017.8257807

Punjabi, R., & Bajaj, R. (2017). User stories to user reality: A devops approach for the cloud. *Proceedings of the 2016 IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology RTEICT 2016* (pp. 658–662). IEEE Press. doi:10.1109/RTEICT.2016.7807905

Qumer Gill, A., Loumish, A., Riyat, I., & Han, S. (2018). DevOps for information management systems. *VINE Journal of Information and Knowledge Management Systems*, *48*(1), 122–139. doi:10.1108/VJIKMS-02-2017-0007

Rahman, A., Mahdavi-Hezaveh, R., & Williams, L. (2018). A systematic mapping study of infrastructure as code research. *Information and Software Technology*, (November). doi:10.1016/j.infsof.2018.12.004

Rajkumar, M., Pole, A. K., Adige, V. S., & Mahanta, P. (2016). DevOps culture and its impact on cloud delivery and software development. *Proceedings - 2016 International Conference on Advances in Computing, Communication and Automation, ICACCA 2016*. Academic Press. doi:10.1109/ICACCA.2016.7578902

Rana, R., & Staron, M. (2016). First International Workshop on Emerging Trends in DevOps and Infrastructure. *Proceedings of the Scientific Workshop Proceedings of XP2016 on - XP '16 Workshops*. Academic Press. doi:10.1145/2962695.2962706

Riungu-Kalliosaari, L., Mäkinen, S., Lwakatare, L. E., Tiihonen, J., & Männistö, T. (2016). DevOps adoption benefits and challenges in practice: A case study. Proceedings of the International Conference on Product-Focused Software Process Improvement (pp. 590-597). Springer. doi:10.1007/978-3-319-49094-6_44

Roche, J. (2013). Adopting DevOps practices in quality assurance. *Communications of the ACM*, *56*(11), 38–43. doi:10.1145/2524713.2524721

Rodríguez, P., Mäntylä, M., Oivo, M., Lwakatare, L. E., Seppänen, P., & Kuvaja, P. (2018). Advances in Using Agile and Lean Processes for Software Development. *Advances in Computers*. doi:10.1016/bs.adcom.2018.03.014

Rong, G., Zhang, H., & Shao, D. (2016). CMMI guided process improvement for DevOps projects. *Proceedings of the International Workshop on Software and Systems Process - ICSSP '16* (pp. 76–85). Academic Press. doi:10.1145/2904354.2904372

Rubasinghe, I., Meedeniya, D., & Perera, I. (2018, June). Automated Inter-artefact Traceability Establishment for DevOps Practice. *Proceedings of the 2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS)* (pp. 211-216). IEEE. doi:10.1109/ICIS.2018.8466414

Rubasinghe, I. D., Meedeniya, D. A., & Perera, I. (2017). Towards Traceability Management in Continuous Integration with SAT-analyzer. *Proceedings of the 3rd International Conference on Communication and Information Processing* (pp. 77–81). Academic Press. doi:10.1145/3162957.3162985

Rufino, J., Alam, M., & Ferreira, J. (2017). Monitoring V2X applications using DevOps and docker. *Proceedings of the 2017 International Smart Cities Conference, ISC2 2017*. Academic Press. doi:10.1109/ISC2.2017.8090868

Samarawickrama, S. S., & Perera, I. (2018). Continuous scrum: A framework to enhance scrum with DevOps. *Proceedings of the 17th International Conference on Advances in ICT for Emerging Regions, ICTer 2017 - Proceedings* (pp. 19–25). Academic Press. doi:10.1109/ICTER.2017.8257808

Šćekić, M., Gazivoda, M., Šćepanović, S., & Nikolić, J. (2018, June). Application of DevOps approach in developing business intelligence system in bank. *Proceedings of the 2018 7th Mediterranean Conference on Embedded Computing (MECO)* (pp. 1-4). IEEE.

Shahin, M., Babar, M. A., & Zhu, L. (2016). The Intersection of Continuous Deployment and Architecting Process. *Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement - ESEM '16*. Academic Press. doi:10.1145/2961111.2962587

Sharma, S. (2017a). DevOps Plays for Driving Innovation. In The DevOps Adoption Playbook (pp. 189–260). Academic Press. doi:10.1002/9781119310778.ch5

Sharma, S. (2017b). *The DevOps Adoption Playbook*. 10.1002/9781119310778

Sharma, S., & Coyne, B. (2015). *DevOps For Dummies* (Vol. 1). John Wiley & Sons, Inc.; doi:10.1017/CBO9781107415324.004

Shivakumar, S. K. (2017). DevOps for Digital Enterprises Brief Introduction to DevOps Scope.

Silva, M. A., Faustino, J., Pereira, R., & Silva, M. M. (2018). Productivity gains of DevOps adoption in an IT team: a case study. *Proceedings of the 27th International Conference on Information Systems Development*. Academic Press. Retrieved from https://repositorio.iscte-iul.pt/handle/10071/16388

Snyder, B., & Curtis, B. (2017). Using Analytics to Guide Improvement during an Agile-DevOps Transformation. *IEEE Software*, *35*(1), 78–83. doi:10.1109/MS.2017.4541032

Soni, M. (2016). End to End Automation on Cloud with Build Pipeline: The Case for DevOps in Insurance Industry, Continuous Integration, Continuous Testing, and Continuous Delivery. *Proceedings of the 2015 IEEE International Conference on Cloud Computing in Emerging Markets, CCEM 2015* (pp. 85–89). IEEE Press. doi:10.1109/CCEM.2015.29

Stahl, D., Martensson, T., & Bosch, J. (2017). Continuous Practices and DevOps : Beyond the Buzz, What Does It All Mean? *Proceedings of the 2017 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA) Continuous* (pp. 440–448). Academic Press. doi:10.1109/SEAA.2017.78

Steffens, A., Lichter, H., & Döring, J. S. (2018a, May). Designing a next-generation continuous software delivery system: concepts and architecture. *Proceedings of the 2018 IEEE/ACM 4th International Workshop on Rapid Continuous Software Engineering (RCoSE)* (pp. 1-7). IEEE.

Steffens, A., Lichter, H., & Döring, J. S. (2018b). Designing a next-generation continuous software delivery system. *Proceedings of the 4th International Workshop on Rapid Continuous Software Engineering - RCoSE '18*. Academic Press. doi:10.1145/3194760.3194768

Stoneham, J., Thrasher, P., Potts, T., Mickman, H., DeArdo, C., & Limoncelli, T. A. (2016). DevOps Case Studies.

Stoneham, J., Thrasher, P., Potts, T., Mickman, H., DeArdo, C., & Limonchelli, T. A. (2017). *DEVOPS CASE STUDIES: The Journey to Positive Business Outcomes* (1st ed.). Portland, Oregon: IT Revolution Press.

Sturm, R., Pollard, C., & Craig, J. (2017). DevOps and Continuous Delivery. In Application Performance Management (APM) in the Digital Enterprise (pp. 121–135). Elsevier. doi:10.1016/B978-0-12-804018-8.00010-3

Swanson, E. B., & Beath, C. M. (1990). Departmentalization in software development and maintenance. *Communications of the ACM*, *33*(6), 658–667. doi:10.1145/78973.78976

Tessem, B., & Iden, J. (2008). Cooperation between developers and operations in software engineering projects. *Proceedings of the 2008 international workshop on Cooperative and human aspects of software engineering - CHASE '08*. Academic Press. doi:10.1145/1370114.1370141

Tingley, G. A., & Anderson, R. M. (1986). Environmental sex determination and density-dependent population regulation in the entomogenous nematode Romanomermis culicivorax. *Parasitology*, *92*(02), 431–449. doi:10.1017/S0031182000064192

Tuma, K., Calikli, G., & Scandariato, R. (2018). Threat analysis of software systems: A systematic literature review. *Journal of Systems and Software*, *144*, 275–294. doi:10.1016/j.jss.2018.06.073

Ur Rahman, A. A., & Williams, L. (2016a). Security practices in DevOps. *Proceedings of the Symposium and Bootcamp on the Science of Security - HotSos '16* (pp. 109–111). Academic Press. doi:10.1145/2898375.2898383

Ur Rahman, A. A., & Williams, L. (2016b). Software security in DevOps. *Proceedings of the International Workshop on Continuous Software Evolution and Delivery - CSED '16* (pp. 70–76). Academic Press. doi:10.1145/2896941.2896946

Vassallo, C., Zampetti, F., Romano, D., Beller, M., Panichella, A., Di Penta, M., & Zaidman, A. (2017). Continuous delivery practices in a large financial organization. *Proceedings of the 2016 IEEE International Conference on Software Maintenance and Evolution, ICSME 2016* (pp. 519–528). IEEE Press. doi:10.1109/ICSME.2016.72

Virmani, M. (2015). Understanding DevOps & bridging the gap from continuous integration to continuous delivery. *Proceedings of the 5th International Conference on Innovative Computing Technology, INTECH 2015* (pp. 78–82). Academic Press. doi:10.1109/INTECH.2015.7173368

Webster, J., & Watson, R. T. (2002). Analyzing the Past to Prepare for the Future: Writing a Literature Review. *Management Information Systems Quarterly*, *26*(2), xiii–xxiii.

Wiesche, M. (2018). Are you ready for Devops? required skill set for Devops teams. *Proceedings of the Twenty-Sixth European Conference on Information Systems (ECIS2018)*. Academic Press.

Wongkampoo, S., & Kiattisin, S. (2018). Atom-Task Precondition Technique to Optimize Large Scale GUI Testing Time based on Parallel Scheduling Algorithm. *Proceedings of the 21st International Computer Science and Engineering Conference 2017* (pp. 229–232). Academic Press. doi:10.1109/ICSEC.2017.8443913

Xia, C., Zhang, Y., Wang, L., Coleman, S., & Liu, Y. (2018). Microservice-based cloud robotics system for intelligent space. *Robotics and Autonomous Systems*, *110*, 139–150. doi:10.1016/j.robot.2018.10.001

Zhu, H., & Bayley, I. (2018). If Docker is the Answer, What is the Question? *Proceedings of the 12th IEEE International Symposium on Service-Oriented System Engineering, SOSE 2018 and 9th International Workshop on Joint Cloud Computing, JCC 2018* (pp. 152–163). Academic Press. doi:10.1109/SOSE.2018.00027

Zhu, L., Bass, L., & Champlin-Scharff, G. (2016). DevOps and Its Practices. *IEEE Software*, *33*(3), 32–34. doi:10.1109/MS.2016.81

464 *Int. J. Agile Systems and Management, Vol. 13, No. 4, 2020*

A maturity model for DevOps

Daniel Teixeira, Rúben Pereira*
and Telmo Henriques

Instituto Universitário de Lisboa (ISCTE-IUL),
Department of Information Science and Technology,
Lisbon, 1649-026, Portugal
Email: dstas@iscte-iul.pt
Email: ruben.filipe.pereira@iscte-iul.pt
Email: telmo_antonio_henriques@iscte-iul.pt
*Corresponding author

Miguel Mira da Silva

Universidade de Lisboa – Instituto Superior Técnico (IST),
Department of Computer Science and Engineering,
Lisbon, 1049-001, Portugal
Email: mms@tecnico.ulisboa.pt

João Faustino and Miguel Silva

Instituto Universitário de Lisboa (ISCTE-IUL),
Department of Information Science and Technology,
Lisbon, 1649-026, Portugal
Email: joao_pedro_faustino@iscte-iul.pt
Email: marsa@iscte-iul.pt

Abstract: Nowadays, businesses aim to respond to customer needs at unprecedented speed. Thus, many companies are rushing to the DevOps movement. DevOps is the combination of Development and Operations and a new way of thinking in the software engineering domain. However, no common understanding of what it means has yet been achieved. Also, no adoption models or fine-grained maturity models to assist DevOps maturation and implementation were identified. Therefore, this research attempt to fill these gaps. A systematic literature review is performed to identify the determining factors contributing to the implementation of DevOps, including the main capabilities and areas with which it evolves. Then, two sets of interviews with DevOps experts were performed and their experience used to build the DevOps Maturity Model. The DevOps maturity model was then developed grounded on scientific and professional viewpoints. Once developed the Maturity Model was demonstrated in a real organisation.

Keywords: DevOps; maturity model; CMMI; practices; capabilities; areas; systematic literature review; interviews; definition; design science research.

Reference to this paper should be made as follows: Teixeira, D., Pereira, R., Henriques, T., Mira da Silva, M., Faustino, J. and Silva, M. (2020) 'A maturity model for DevOps', *Int. J. Agile Systems and Management*, Vol. 13, No. 4, pp.464–511.

Biographical notes: Daniel Teixeira is an information systems consultant. He graduated as MSc in Computer Engineering at ISCTE. He has been working in several projects in industry. Besides his professional work, he is very enthusiastic about trending information systems topics. So far, his research interests rely on DevOps and maturity models.

Rúben Pereira is an Assistant Professor at ISCTE. He has a PhD in information systems at Instituto Superior Técnico where he also graduated as a Master's in Computer Engineering and Computer Science. He has been a consultant in several industries, such as services, banking, telecommunications, and e-commerce, among others. He is the author of several scientific papers in the area of IT Services Management and IT Governance, covering the main IT Frameworks like ITIL and COBIT. His areas of scientific interest extend to information technology risk management, BPM, continuous improvement and innovation, process optimisation, digital transformation, DEVOPS, RPA, among others.

Telmo Henriques is currently an Invited Assistant Professor at ISCTE-IUL and a Research Associate at ISTAR-IUL. He is an Associate Editor of the *Journal of Global Information Management* and of the *Information Resource Management Journal*, having also collaborated, as a member of the Editorial Advisory Board, on the *IGI Encyclopedia of Organizational Knowledge, Administration, and Technology* and the *IGI Encyclopedia of Information Science and Technology*. His past long professional management experience, within the areas of management consulting and banking, has involved the assumption of main responsibilities on ICT consultancy, information systems methodologies, quality management and organisational change & development.

Miguel Mira da Silva is currently Associate Professor of Information Systems in the University of Lisbon, leader of the research group 'Digital Transformation' at INOV, and coordinator of the MISE online MSc. He has a PhD in Computing Science from the University of Glasgow and an MSc in Management from the London Business School. Miguel created five companies, published four books and 200 research papers, managed dozens of research and consulting projects, graduated nine PhDs and 150 MSc students, and created a MOOC about digital transformation. His current interests include digital transformation, IT governance, and online learning.

João Faustino is an IT Consultant. He is also a PhD student in Information Science and Technology at ISCTE-IUL where he also graduated as MSc in Computer Engineering. Most of his career, he has worked as a consultant on the Financial Services industry, on application management teams for both corrective and enhancement maintenance of Core Systems for Insurance companies. Besides his professional work, which is mainly related with Oracle Technologies, he is very enthusiastic about open source technologies, data science, mobile development and process optimisation. Most of his research is about IT service management and DevOps.

Miguel Silva is the head of a Software Engineering team in one of the biggest European engineering company. He has MSc in Computer Engineering at ISCTE and he achieved his BSc graduation at FCT-UNL also in Computer Engineering. He has been working since 2009 in the ITSM area in the telecommunications industry, and he is responsible to manage a DevOps team that manages and implements one of the biggest Service Now instances in the world. His main interest is related with agile methodologies (SCRUM, DevOps, etc).

1 Introduction

In recent years, the advancements on DevOps area have facilitated a lot of new growth opportunity for software companies (Nidagundi and Novickis, 2017) as it improves the way how a business delivers value to its customers, suppliers, and partners, it is an essential business process, not just an IT capability (Katal et al., 2019). This is one of the main reasons why the DevOps' adoption is growing and is a new tendency in business and IT alignment (Bucena and Kirikova, 2017). DevOps allows a business to maximise the speed of delivery of a product or service, from the initial idea to production release and all the way up to customer feedback to improvements based on that feedback (Koilada, 2019).

Businesses today need to respond to customer needs at unprecedented speed. Driven by this need for speed, many companies are then rushing to the DevOps movement and implementing Continuous Delivery (Chen, 2018).

The growth opportunities for DevOps continue to increase. Ovum, a market-leading data, research and consulting company, sees plenty of evolution potential in DevOps as there is potential for improved integration with Application Lifecycle Management on the dev side and improved integration with operations and IT business services (Azoff, 2016). According with the 2018 State of DevOps Report has been registered a steady increase in survey responses from people on DevOps teams, from just 16% in 2014 to 29% in 2018 (Velasquez et al., 2018).

The adoption of DevOps drives a challenging cultural shift towards collaboration and knowledge-sharing between SD, quality control and operations (Colomo-Palacios et al., 2018). The tremendous growth in demand for DevOps has, however, led to the appearance of new needs. For instance, many companies find it difficult to understand what DevOps is and what advantages it will have (St et al., 2017).

Many companies miss the maturity of the concept – with no clear definition of DevOps and its practices, no clear goals available and a lack of understanding about development workflow phases and responsibilities (Bucena and Kirikova, 2017). There is both a lack of understanding around DevOps and a clear definition of what it is (Lwakatare et al., 2015). Therefore, organisations are not sure how to effectively implement DevOps capabilities (Chen, 2019). Plus, complexity is evolving since DevOps security concerns start to be raised (Prates et al., 2019).

The disruptive nature of the changes required to adopt DevOps leads to organisational and business stress. While Zhu et al. (2016) consider the organisational strains as being standard for new technologies, for Bucena and Kirikova (2017) the adoption of DevOps is not trivial and can require complex changes in an enterprise's process, organisation and workflow. To succeed in adopting DevOps, the enterprises should understand the different aspects that are related to the DevOps approach and have a well-thought-out strategy. They should start the adoption process with a clear idea of what actions should be performed, how they should be prioritised, what tools could support these actions, and how to measure the success of the adoption process (Bucena and Kirikova, 2017). Moreover, the way an organisation is structured may influence DevOps' adoption, for example, when discussing communication, common goals and practices, decision making, and systems thinking within the organisation (Smeds et al., 2015b).

Whereas DevOps benefits are widely discussed regarding DevOps culture and available tools, it makes sense to exist a Maturity Model (MM) for DevOps approaches. A MM is a widely used technique that has proven valuable for assessing business processes or certain aspects of organisations, as it represents a path towards an increasingly organised and systematic way of doing business (Proenca, 2016). They also allow for a better positioning of the organisation and help find better solutions for change (Becker et al., 2009). Moreover, MM are an important tool for business-IT alignment (Pereira and Da Silva, 2011; Aguiar et al., 2018).

According to the literature, both areas and capabilities play an important role in DevOps adoption and maturation. Therefore, this study aims to: Develop a MM for DevOps (RO1). To achieve this objective, it may be necessary to identify both DevOps capabilities (RQ1.1) and DevOps areas (RO1.2). Plus, the two sets of interviews with DevOps experts were performed to tune the final artefact.

2 Theoretical background

2.1 DevOps

A good cooperation between IT Development and IT Operation teams is viewed to be crucial in order to ensure successful deployment and operations of IT systems (Tessem and Iden, 2008). However, for historical reasons, most IT organisations are characterised by clear boundaries between these two teams, which have very different goals, mindsets and cultures (Swanson and Beath, 1990; Gazivoda, 2018).

According to Sharma (2014), many organisations are not successful with software projects and their failures are related to the challenges in product development and delivery. Despite this, many companies find that the development and delivery of software applications are crucial to their business, and that only 25% of companies consider their teams to be efficient (Sharma, 2014). This gap in efficiency leads to many losses of business opportunities. This demonstrates that even a disruptive methodology cannot be perfect for every project.

In order to address the problems between the development and operations teams a new agile approach appeared, namely DevOps. DevOps has been heralded as a novel paradigm to overcome the traditional boundaries between IT Development (Dev) and IT Operations (Ops) teams (Nielsen et al., 2017) aiming to improve collaboration between these teams. It represents a change in IT culture, focusing on rapid IT service delivery through the adoption of agile, lean practices in the context of a system-oriented approach (Jabbari et al., 2016). However, DevOps is not only influenced by cultural aspects as also supported by technological enablers (Smeds et al., 2015a).

According to Riungu-Kalliosaari et al. (2016), DevOps is a set of practices intended to reduce the time between making a change to a system and this change being placed into normal production, while ensuring high quality. The main goal associated with this concept is to avoid common problems when operations and developers are kept as separated teams (Bezemer et al., 2018).

To sum up, DevOps replaces siloed units with cross-functional teams so organisations may leverage automated development, deployment, and infrastructure and enables teams to continuous work and deliver operational features (Ebert et al., 2016).

2.2 Maturity model

MM's are commonly used as an instrument to conceptualise and measure maturity of an organisation or a process regarding some specific target state (Schumacher et al., 2016). Further, MM intended for a prescriptive purpose of use include good or best practices which is helpful to provide practical guidance (Röglinger et al., 2018). They refer that maturity not only implies a potential for growth in capability, but also focuses on richness and consistency regarding execution. In this regard, Andersen and Jessen (2003) define maturity as the quality or state of being mature. The maturity concept must be related to a state in which organisations are in perfect conditions to achieve their goals (Berssaneti et al., 2012).

Two approaches for implementing MMs exist. With a top-down approach, such as proposed by Becker et al. (2009) a fixed number of maturity stages or levels is specified first and further corroborated with characteristics (typically in form of specific assessment items) that support the initial assumptions about how maturity evolves. On the other hand, when using a bottom-up approach, such as suggested by Lahrmann et al. (2011), distinct characteristics are determined first and clustered in a second step into maturity levels to induce a more general view of the different steps of maturity evolution. This research follows the top-down MM approach proposed by Becker et al. (2009).

2.3 CMMI

Capability maturity model integration (CMMI) (and its predecessor CMM) is a framework intended to cover many software engineering best practices and can be used for SPI. CMMI is most well known in its 'staged' representation, which has five maturity levels. To reach a maturity level, a company must satisfy the goals of the process areas for that and all lower levels. The expected capacity of an organisation that operates in a more mature way depends directly on your ability to perform, control, and improve performance in one or more areas of implementation of the model practices (Barbosa et al., 2007).

CMMI evokes barriers in some because of the processes involved in certification. However, CMMI at its core is not a methodology but rather a set of principles. In the case of CMMI, the set of principles focuses on maturation of a SD process. CMMI is concerned with defining metrics and practices to ensure continuous improvement (Chrissis et al., 2010). The goal of CMMI is not just to support a minimum set of standards to achieve to a particular level, but to enable increasing improvement in organisational processes. CMMI's approach is based on MM. It supports both a staged approach and a continuous model for improvement. It provides several key process areas at different levels. Maturity levels are those that are related to the path which helps organisations to apply improvements to a set of related processes by incrementally addressing successive sets of process areas and goes through 1 to 5.

3 Related work

Since this research aims to study DevOps' maturity, it is mandatory to search literature where it is possible to study other proposals for DevOps' MMs. However, given that DevOps is a new term and concept recently introduced, the author decided to extend the scope of the study to SDMMs. To do that, the author performed a literature review.

A literature review may be helpful distinguishing what has been done from what needs to be done, discovering important variables relevant to the topic, synthesising and gaining a new perspective or identifying relationships between ideas and practice (Hart, 1998). An effective review creates a solid foundation for advancing knowledge. It facilitates theory development, closes areas where a plethora of research exists, and uncovers areas where research is needed (Webster and Watson, 2002). For easier understanding of the peers, as well as to add more scientific rigor to our research, the author decided to follow the concept centric approach proposed by Webster and Watson (2002).

To perform the literature, review the authors have searched and consulted the following digital repositories: IEEExplore, ACM, Research Gate and it was also used the search engine of Google Scholar.

This research was carried out between September of 2018 and January of 2019. The keywords used to perform this research were: 'DevOps maturity model', 'DevOps maturity', 'Software Development Projects maturity model', 'Software development projects maturity', 'Scrum maturity model' and 'Scrum maturity'.

In this section, the main findings regarding Software Development Processes (SDP), Scrum and DevOps MMs are presented (Table 1). Plus, Table 2 details these studies characteristics, while Table 3 contains all the studies mapped with the corresponding maturity vectors found by the authors in the proposed MMs. Together, these three tables explain how this research increases the body of knowledge.

Table 1 SDP, and DevOps MMs

ID	*Author*	*MMs*			*Model*	*Maturity levels*	*Dimension*
		DevOps	*Scrum*	*SDP*			
S.1	Mohamed (2015)	X			CMMI	5	4
S.2	Bucena and Kirikova (2017)	X			Not defined	5	4
S.3	Yin (2011)		X		CMMI	5	Not defined
S.4	Srivastava et al. (2017)		X		Not defined		Not defined
S.5	Kawamoto and De Almeida (2017)		X		CMMI	Not defined	Not defined
S.6	Baskarada et al. (2005)			X	CMMI	5	Not defined
S.7	Patel and Ramachandran (2009)			X	CMMI	5	Not defined
S.8	Buglione (2011)			X	CMMI	5	4
S.9	Santana et al. (2013)			X	CMMI	5	Not defined
S.10	Fontana et al. (2015)			X	CMMI	Not defined	6
S.11	Stojanov et al. (2015)			X	Not defined	5	5

Since DevOps is a recent theme and there are not a lot of dedicated maturity studies in literature (Rong et al., 2016a). So, the authors have decided to include agile and scrum MMs.

Both Scrum and DevOps have in common to broaden the usage of Agile practices to operations to streamline the entire software delivery process in a holistic way (Hüttermann, 2012; Bang et al., 2013). Table 1 presents all the MMs for SDP, Scrum and DevOps found among the literature.

From the analysis of Table 1 some conclusions can be withdrawal. The low number of DevOps MMs that has been found indicate that few studies exist deep studying DevOps. The number of studies on SDP is greater than for scrum and DevOps. One of the main reasons for this is that most of the SDP uses Agile methodology, which in turn is the basis for both DevOps and Scrum so it is expected that there exist more studies about this theme than for the others.

CMMI seems to be the basis of these models since it was used in 73% of these studies. It was not explicit any of the vectors that constitutes the Scrum' MMs and, apart from one study, the same happened to the number of levels used. This is justified by the fact that CMMI is a well-known methodology used to develop and refine an organisation's SD process (Farkas and Walsh, 2002). CMMI is an approach to improve processes that provides elements that are essential for an effective process. It brings together best practices that address development and maintenance activities, thus covering the entire lifecycle of a product from conception to delivery and maintenance (Chrissis et al., 2010). It has been also included a vector named 'Dimension' that represents the number of vectors that were represented in model. From Table 1, it is possible to see that the study with less dimensions had four and on the opposite side, the study with more dimensions has six. This helps the authors to put into perspective the number of dimensions used in other MM, to understand the number of dimensions that should be used in this study.

Studies' characteristics are better detailed in Table 2 where vectors are used for proper analysis, such as the year in which the model was developed, which MM was based on, if it follows Becker's top-down approach, if the author justified the vectors used, whether they comply with the Design Science Research (DSR) steps and if any demonstration of the model was performed in practice.

Overall, two MMs for DevOps were identified in literature. However, as one can see in Table 2, both MMs lack the use of structured methods in the design process which may raise doubts on their scientific rigor. For instance, only one is based on CMMI and none adopts Becker theory or DSR to build the MM.

Moreover, Table 3 lists and synthesises the related work and identifies what vectors were used to design each analysed MM. By doing it, the authors aimed to identify the main vectors that were applied on those case studies and understand the reasons behind those.

For a better understanding, the studies have been grouped by approach. A vector can be written on a different way depending on its context, so the authors have grouped these vectors by the meaning of the vector. Table 3 shows the vectors grouped by study.

Through the analysis of Table 3, it can be devised that several MM exist in the literature (11 studies). In six of these studies, the authors did not specify the vectors that would be used. Although DevOps studies are less than agile studies, some agile MMs use the same vectors defined by the DevOps MMs. This may be due to the fact that, first, DevOps and agile keep a close relationship and, secondly, DevOps is a recent topic and there is not much information available about it (Hussain et al., 2017). On agile studies, with some exceptions, it appears that each author defined most of their vectors.

Focusing on DevOps studies, there are no common characteristics present among the two models found. This also proves that the field needs further developments to reach more consensus and completeness. Each author decided to establish their own vectors based on what they thought best defines the characteristics and that could help define the maturity of DevOps in the context of their studies.

Table 2 MMs characteristics

ID	*Year*	*Proposed MM*	*Based MM*	*Becker's top-down approach*	*Vectors validation*	*DSR*	*Demonstration*
S.1	2015	DevOps	CMMI	Not used	Not validated	Not used	Not applied
S.2	2017	DevOps	Not defined	Not used	Validated	Not used	Applied
S.3	2011	Scrum	CMMI	Not used	Not defined	Not used	Not applied
S.4	2017	Scrum	Not defined	Not used	Not defined	Not used	Not applied
S.5	2017	Scrum	CMMI	Not used	Not defined	Not used	Not applied
S.6	2005	SDP	CMMI	Not used	Not defined	Not used	Not applied
S.7	2009	SDP	CMMI	Not used	Not defined	Not used	Not applied
S.8	2011	SDP	CMMI	Not used	Not Validated	Not used	Applied
S.9	2013	SDP	CMMI	Not used	Not defined	Not used	Not applied
S.10	2015	SDP	Not defined	Not used	Validated	Not used	Applied
S.11	2015	SDP	CMMI	Not used	Validated	Not used	Not applied

Table 3 Vectors used in the MMs from related work

	DevOps		*Scrum*			*Agile*					
Vector	*S.1*	*S.2*	*S.3*	*S.4*	*S.5*	*S.6*	*S.7*	*S.8*	*S.9*	*S.10*	*S.11*
Culture		X									
Collaboration	X									X	
Process		X									
Quality	X										
Automation	X										
Governance	X										
Technology		X									
People		X								X	X
General								X			
Sustained Success								X			
Organisation's Environment								X			
Interested parties, needs and expectations								X			
Embrace Change to Deliver Customer Value										X	

Table 3 Vectors used in the MMs from related work (continued)

Vector	*DevOps*		*Scrum*					*Agile*			
	S.1	*S.2*	*S.3*	*S.4*	*S.5*	*S.6*	*S.7*	*S.8*	*S.9*	*S.10*	*S.11*
Plan and Deliver Software Frequently										X	
Technical Excellence										X	
Practices											X
Deliveries											X
Requirements											X
Product											X
Customer											X

To Mohamed (2015), the keys to successful adoption of DevOps are quality, automation, collaboration, and governance/process, while claiming that, together, these fundamental elements can unify the traditional IT silos to enable agility across the end-to-end application life cycle. On the other hand, Bucena and Kirikova (2017) DevOps MM was developed on the basis of analysis of related work and includes five levels of maturity with respect to the four enterprise areas, namely, technology, process, people, and culture. No surprises with the absence of DevOps as possible vectors to assess DevOps maturity.

With the lack of consensus among the studies as well as the absence of both the use of rigorous methods/methodologies in the design process and DevOps capabilities as vectors of maturity assessment, the design of a new MM for DevOps can be faced as an opportunity and a step forward on the perspective of associated mature practices.

4 Research methodology

4.1 Design science research

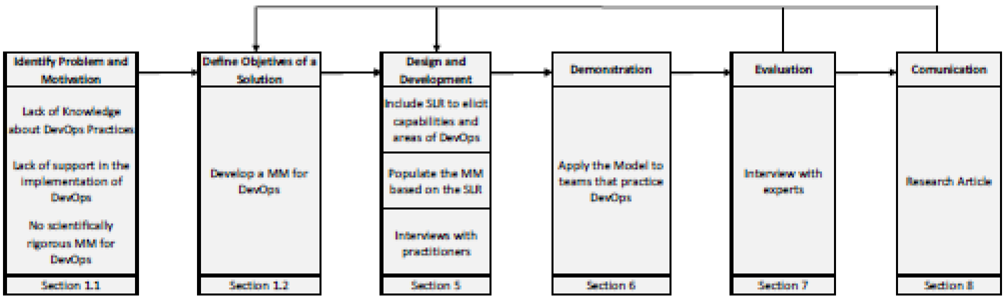
For the development of the proposed DevOps MM, it was applied the design science research methodology (DSRM) presented by Peffers et al. (2006) and the seven guidelines for DSR proposed by Hevner et al. (2004). DSR approach was selected since this research aims at solving practical problems by creating and evaluating IT artefacts intended to solve identified organisational problems (Hevner et al., 2004). DSR was recently indicated as the main methodology to develop MM (Pereira and Serrano, 2020), thus reinforcing its choice in this study.

IT artefacts are broadly defined as constructs (i.e., vocabulary and symbols), models (i.e., abstractions and representations), methods (i.e., algorithms and practices), and instantiations (i.e., implemented and prototype systems) (Hevner et al., 2004). According to Becker et al. (2009) and Mettler (2009), it can be assumed that the development of MMs falls within the application area for the guidelines by Hevner et al. (2004).

According to Peffers et al. (2006), the DSRM consists of six activities (i.e., steps). Figure 1 presents our applied techniques and performed activities in each DSRM step. In order to achieve rigorous as well as relevant research results, we draw upon the following DSRM steps, whereby the paper is structured accordingly:

- *Problem identification and motivation*: In the first section, it was specified the problem, provided practical relevance and justified the value of a solution. Additionally, based on problem scope, research questions were derived guiding this research.
- *Define the objectives for a solution*: The second section provides objectives of the intended collaboration MM. Based on a literature review, design recommendations in MM design and assessment will be identified and suggestions for circumvention will be proposed.
- *Design and development*: This activity is present in Section 5 and describes the MM development. Based on a literature review the MM will be designed and iteratively developed according to the requirements of MM construction (Becker et al., 2009).
- *Demonstration*: By means of an application test with three participant organisations the applicability and usability of the artefact was demonstrated. The utility of the MM will be further validated DevOps experts.
- *Evaluation*: According to Hevner et al. (2004), the artefact will be evaluated in terms of quality, utility and efficacy which cannot be demonstrated fully in this research.
- *Communication*: Communicate the problem, the importance, the utility, the rigor and the effectiveness of its design.

Figure 1 Applied DSR guidelines



4.2 Systematic literature review

One of the major tools used in other domains to support an evidence-based paradigm is the generation of Systematic Literature Reviews (SLR), which is used to aggregate the experiences gained from a range of different studies in order to answer a specific research question (Khan et al., 2004).

A SLR is a literature review method that aims to address a problem by identifying, evaluating, integrating all relevant findings, and interpreting research on research topics to answer research questions based on the stages used in SLR (Siddaway, 2014). The process of addressing the problem of lack of knowledge aims to identify the relationships and gaps in the existing literature. The identification process is used to describe directions for future research, because it consists of the process of formulating a general statement or an overarching conceptualisation, commenting on, evaluating, extending, or developing theory from existing literature (Siddaway, 2014).

This research follows Kitchenham procedures for SLR (Kitchenham, 2004), complemented by the concept centric approach from Webster and Watson (2002).

4.3 Semi-structured individual interviews and email interviews

The interview study reported here was carried out with DevOps practitioners Professionals from all over the world. The study took place as a qualitative interview study in the tradition of the qualitative research interview.

Semi-structured interviews are characterised by the use of a script consisting of closed or open predefined questions (Rijo, 2008). They are suitable when the research wants to validate several hypotheses but also to know the fieldwork and to explore new ones (Pozzebon, 2006). Particularly, they enable the interviewee to discuss the subject matter without being too attached to the formulated inquiry (Manzini, 2004). They also facilitate the interviewer to have clear support following the questions (Manzini, 2004). Moreover, they ensure to authors that their hypotheses or assumptions will be broadly covered by the conversation (Minayo, 2004).

Qualitative research has become essential to the humanities over the past twenty years (Ratislavová and Ratislav, 2014). Synchronous and asynchronous interviews and virtual focus groups are the most common methods (Ratislavová and Ratislav, 2014). The use of Email Interview can be employed quickly, conveniently, and inexpensively and can generate high-quality data when handled carefully. While a mixed mode interviewing strategy should always be considered when possible, semi-structured email interviewing can be a viable alternative to the face-to-face and telephone interviews, especially when time, financial constraints, or geographical boundaries are barriers to an investigation (Meho, 2006).

5 Design and development

To design the artefact, the author followed the steps listed below:

Step 1: Identify which are the main DevOps capabilities

Method(ology): SLR

Step 2: Identify which are the areas that most relate with DevOps.

Method(ology):SLR

Step 3: Identify the main practices of each DevOps capability

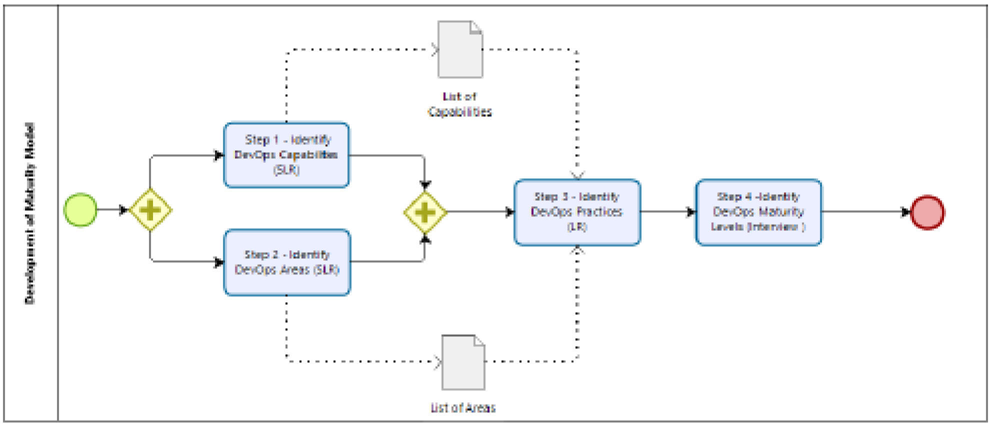
Method(ology): Literature Review

Step 4: Identify the maturity level of each DevOps practice

Method(ology): Interview

For a better understanding of the Design and Development's phase, the authors built the workflow (Figure 2) of the four previously described steps.

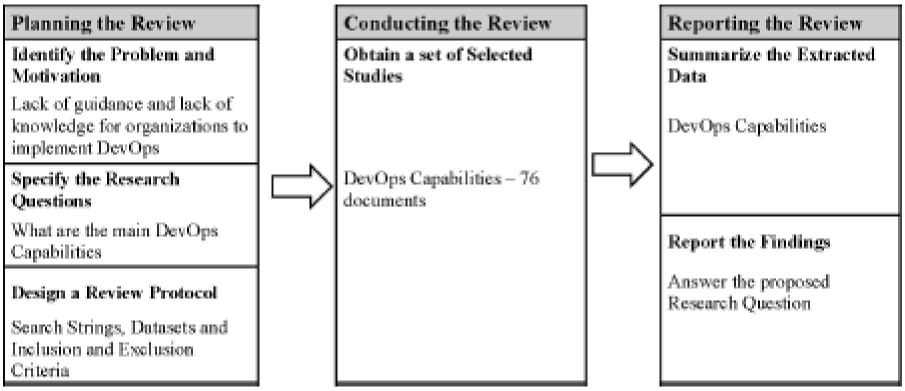
Figure 2 Workflow of the design and development's phase (see online version for colours)



5.1 Step 1 (Capabilities)

Figure 3 details the SLR phases adopted in Step 1. The SLR was chosen as a starting point to develop our Research Methodology since we wanted to summarise the existing evidence regarding DevOps' capabilities, with the aim of answering the proposed Research Objectives.

Figure 3 SLR methodology for DevOps' capabilities



The search string which was used and respective datasets are listed below.

- *Search String*: DevOps AND (Capability OR Capabilities OR Practice)
- *Datasets*: Google Scholar, ScienceDirect, IEEEXplore, ACM.

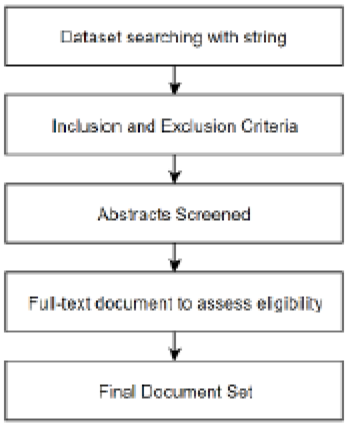
After that, inclusion and exclusion criteria must be applied to filter the obtained documents. Our criteria are presented in Table 4.

Afterwards, the first set of documents is obtained. Then, in a first phase, the abstracts were screened to decide their relevance to the research. Finally, these documents were red in order to obtain the final selection of studies to perform the review. The review protocol is illustrated in Figure 4.

Table 4 Inclusion and exclusion criteria for DevOps' capabilities

Inclusion criteria	*Exclusion criteria*
Written in English or Portuguese	Not written in English or Portuguese
Scientific papers in conferences or journals and books	Non-Free documents nor Master Thesis
Title relevance regarding DevOps	No title relevance DevOps

Figure 4 Review protocol for DevOps' capabilities



For a better understanding, as well as to add more scientific rigor to the research, the authors decided to follow the centric approach proposed by Webster and Watson (2002).

After applying the review protocol, 76 relevant studies were obtained for our research. Table 5 lists all the DevOps capabilities that were found, with its respective scientific references that support each capability.

Table 5 DevOps capabilities SLR

ID	*Capabilities*	*Reference*	*No. of references*
C1	Continuous Integration	Yin et al. (2004), Debois (2011), Hüttermann (2012), Fitzgerald and Stol (2014), Chen et al. (2015), De Bayser et al. (2015), Jabbari et al. (2016), Mohan and Ben Othmane (2016), Moore et al. (2016), Shahin et al. (2016), Soni (2016), Stoneham et al. (2016), Croker and Hering (2016), de França et al. (2016), Bucena and Kirikova (2017), Laukkarinen et al. (2017, 2018), Molto et al. (2017), Palihawadana et al. (2017), Pang and Hindle (2017), Punjabi and Bajaj (2017), Rubasinghe et al. (2017, 2018), Sharma (2017a), Shivakumar (2017), Snyder and Curtis (2017), Vassallo et al. (2017), Bai et al. (2018), Düllmann et al. (2018), Kuusinen et al. (2018), Lewerentz et al. (2018), Mackey (2018), Mansfield-Devine (2018), Marijan et al. (2018), Rahman et al. (2018), Rodriguez et al. (2018), Cleveland et al. (2018), Steffens et al. (2018), Tuma et al. (2018), Colomo-Palacios et al. (2018), Wiesche (2018), Wongkampoo and Kiattisin (2018), Xia et al. (2018), Zhu and Bayley (2018) and Debroy et al. (2018)	44

Table 5 DevOps capabilities SLR (continued)

ID	*Capabilities*	*Reference*	*No. of references*
C2	Continuous Deployment	Yin et al. (2004), Debois (2011), Hüttermann (2012), Fitzgerald and Stol (2014), Chen et al. (2015), Farshchi et al. (2015), Fördős and Cesarini (2016), Jabbari et al. (2016), Karapantelakis et al. (2016), Mohan and Ben Othmane (2016), Rana and Staron (2016), Shahin et al. (2016), Soni (2016), Stoneham et al. (2016), Ur Rahman and Williams (2016b), Ali et al. (2017), Bass (2017), Palihawadana et al. (2017), Pang and Hindle (2017), Perera et al. (2017a), Punjabi and Bajaj (2017), Rubasinghe et al. (2017), Sharma (2017a), Bucena and Kirikova (2017), Shivakumar (2017), Kuusinen et al. (2018), Laukkarinen et al. (2018), Mackey (2018), Mansfield-Devine (2018), Bhattacharjee et al. (2018), Rahman et al. (2018), Rubasinghe et al. (2018), Steffens et al. (2018), Tuma et al. (2018), Wiesche (2018), Xia et al. (2018), Zhu and Bayley (2018), Cleveland et al. (2018), Debroy et al. (2018) and Düllmann et al. (2018)	38
C3	Continuous Monitoring	Yin et al. (2004), Hüttermann (2012), Roche (2013), Fitzgerald and Stol (2014), Chen et al. (2015), John et al. (2015), Rana and Staron (2016), Soni (2016), Ur Rahman and Williams (2016b), de França et al. (2016), Hanappi et al. (2016), Karapantelakis et al. (2016), Bucena and Kirikova (2017), Li et al. (2017), Pang and Hindle (2017), Perera et al. (2017a), Rubasinghe et al. (2017), Rufino et al. (2017), Sharma (2017a), Shivakumar (2017), Snyder and Curtis (2017), Vassallo et al. (2017), Bai et al. (2018), Kuusinen et al. (2018), Steffens et al. (2018) and Düllmann et al. (2018)	25
C4	Continuous Testing	Yin et al. (2004), Hüttermann (2012), Roche (2013), Fitzgerald and Stol (2014), Chen et al. (2015), Shahin et al. (2016), Croker and Hering (2016), Soni (2016), Stoneham et al. (2016), Jabbari et al. (2016), Bucena and Kirikova (2017), Palihawadana et al. (2017), Pang and Hindle (2017), Punjabi and Bajaj (2017), Sharma (2017a), Shivakumar (2017), Snyder and Curtis (2017), St et al. (2017), Vassallo et al. (2017), Feijter et al. (2017), Murugesan (2017), Nielsen et al. (2017), Rubasinghe et al. (2018), Samarawickrama and Perera (2018), Silva et al. (2018), Wiesche (2018) and Kuusinen et al. (2018)	26
C5	Feedback Loops between Dev and Ops	Yin et al. (2004), Hüttermann (2012), Roche (2013), John et al. (2015), Stoneham et al. (2016), Hanappi et al. (2016), Jabbari et al. (2016), Bucena and Kirikova (2017), Feijter et al. (2017), Nielsen et al. (2017), Pang and Hindle (2017), Sharma (2017a), St et al. (2017), Murugesan (2017), Silva et al. (2018), Wongkampoo and Kiattisin (2018), Debroy et al. (2018), Kuusinen et al. (2018) and Mikkonen et al. (2018)	18
C6	Infrastructure as code	Yin et al. (2004), Hüttermann (2012), De Bayser et al. (2015), Rana and Staron (2016), Shahin et al. (2016), de França et al. (2016), Fördős and Cesarini (2016), Jabbari et al. (2016), Bucena and Kirikova (2017), Sharma (2017a), Jimenez et al. (2017), Bhattacharjee et al. (2018), Rahman et al. (2018), Steffens et al. (2018), Debroy et al. (2018) and Düllmann et al. (2018)	14

Table 5 DevOps capabilities SLR (continued)

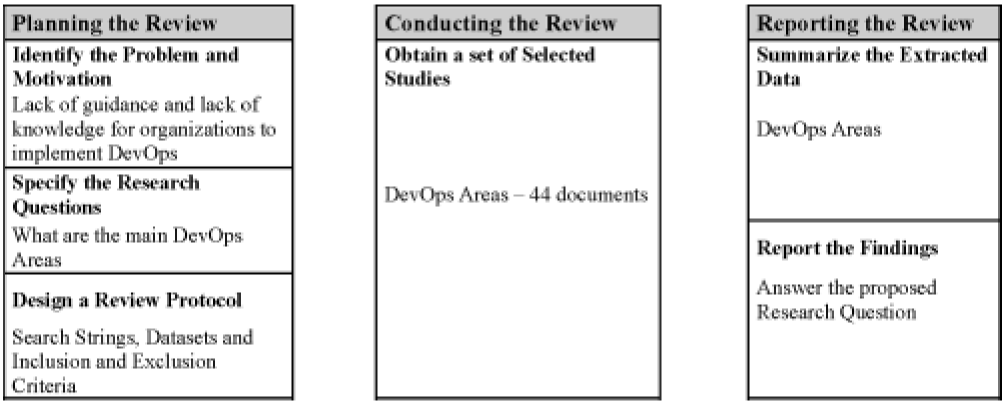
ID	*Capabilities*	*Reference*	*No. of references*
C7	Change Management	Debois (2011), Hüttermann (2012), Mohamed (2015), Science and Sciences (2015), Jabbari et al. (2016), Abdelkebir et al. (2017), Rubasinghe et al. (2017), Sharma (2017b) and Zhu and Bayley (2018)	9
C8	Continuous planning	Hüttermann (2012), Fitzgerald and Stol (2014), Jabbari et al. (2016), Ur Rahman and Williams (2016a), Pang and Hindle (2017), Sharma (2017b) and Kuusinen et al. (2018)	7
C9	Prototyping application	Hüttermann (2012), Fitzgerald and Stol (2014), De Bayser et al. (2015), Jabbari et al. (2016), Sharma (2017b) and Cleveland et al. (2018)	6
C10	Process Standardisation	Hüttermann (2012), Roche (2013), Jabbari et al. (2016), Rana and Staron (2016) and Sharma (2017b)	5
C11	Stakeholder Participation	Hüttermann (2012), Jabbari et al. (2016) and Sharma (2017b)	3
C12	Shift Left	Hüttermann (2012), Feijter et al. (2017) and Sharma (2017b)	3

5.2 *Step 2 (Areas)*

The three SLR phases, described in Section 4.1 are represented in Figure 5, and were specifically adapted to this section purpose.

We have chosen SLR as Research Methodology since it was intended to summarise the existing evidence regarding DevOps' areas, with the aim of answering the proposed Research Question.

Figure 5 SLR methodology for DevOps areas



The search string which was used and respective datasets are listed below.

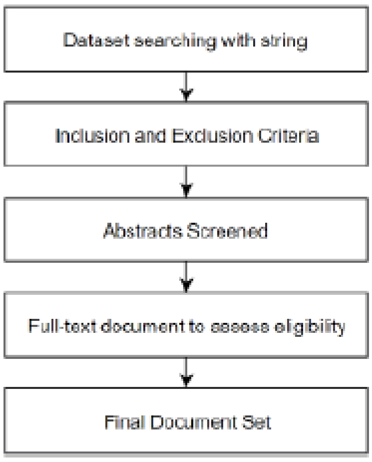
- *Search String*: DevOps AND (Area, Principles, View, Dimensions and Perspective)
- *Datasets*: Google Scholar, ScienceDirect, IEEEXplore, ACM.

After that, inclusion and exclusion criteria must be applied to filter the obtained documents. Our criteria are presented in Table 6.

Table 6 Inclusion and exclusion criteria for DevOps areas

Inclusion criteria	*Exclusion criteria*
Written in English or Portuguese	Not written in English or Portuguese
Scientific papers in conferences or journals and books	Non-Free documents nor Master Thesis
Title relevance regarding DevOps	No title relevance DevOps

Figure 6 Review protocol for DevOps areas



Afterwards, the first set of documents is obtained. Then, in a first phase, the abstracts must be screened to decide their relevance to the research. Finally, these documents are read in order to obtain the final selection of studies to perform the review. The review protocol is illustrated in Figure 6.

For a better understanding, as well as to add more scientific rigor to our research, the authors decided to follow the centric approach proposed by Webster and Watson (2002).

After applying the review protocol, 44 relevant studies were obtained for our research. Table 7 lists all the DevOps capabilities that were found, with its respective scientific references that support each capability.

5.3 Step 3 (DevOps practices)

Having analysed Table 5, and considering that there is a considerable gap between C6 and C7, the authors have decided to identify all the practices for each capability from C1 and C6. Since that the information regarding these capabilities are spread in a lot of studies, each capability's practices will be synthesised by grouping it by Area.

After analysing the descriptions of the areas from C1 to C12, the authors have concluded that some areas identify themselves with other areas. Considering that it would be complex to detail all the practices of all these areas, and since there are areas that cover other areas, the authors have decided to group some Areas. Thus, Technology will include Automation, Culture includes Sharing and Process includes Measurement.

Table 7 DevOps areas SLR

ID	*Area*	*References*	*No. of references*
A1	Culture	Debois (2011), Hüttermann (2012), Bang et al. (2013), Erich et al. (2014a), Sharma and Coyne (2015), Smeds et al. (2015b), Rana and Staron (2016), de França et al. (2016), Diel et al. (2016), Jabbari et al. (2016), Bucena and Kirikova (2017), Nielsen et al. (2017), Perera et al. (2017b), Gupta et al. (2017), Silva et al. (2018) and Colomo-Palacios et al. (2018)	16
A2	Measurement	Debois (2011), Hüttermann (2012), Bang et al. (2013), Erich et al. (2014a), Smeds et al. (2015b), Rana and Staron (2016), de França et al. (2016), Jabbari et al. (2016), Perera et al. (2017b), Gupta et al. (2017), Nielsen et al. (2017), Colomo-Palacios et al. (2018), Silva et al. (2018) and Luz et al. (2018)	14
A3	Sharing	Debois (2011), Hüttermann (2012), Bang et al. (2013), Erich et al. (2014a), Smeds et al. (2015b), Rana and Staron (2016), de França et al. (2016), Jabbari et al. (2016), Perera et al. (2017b), Gupta et al. (2017), Nielsen et al. (2017), Colomo-Palacios et al. (2018), Silva et al. (2018) and Luz et al. (2018)	14
A4	Automation	Debois (2011), Hüttermann (2012), Bang et al. (2013), Erich et al. (2014a), Smeds et al. (2015b), Mohamed (2015), Rana and Staron (2016), de França et al. (2016), Jabbari et al. (2016), Nielsen et al. (2017), Perera et al. (2017b), Gupta et al. (2017), Colomo-Palacios et al. (2018), Silva et al. (2018) and Luz et al. (2018)	14
A5	Technology	Hüttermann (2012), McCarthy et al. (2015), Sharma and Coyne (2015), Diel et al. (2016), Abdelkebir et al. (2017), Bucena and Kirikova (2017), Hussain et al. (2017), Sturm et al. (2017), Gazivoda (2018) and Silva et al. (2018)	10
A6	People	Hüttermann (2012), McCarthy et al. (2015), Sharma and Coyne (2015), Abdelkebir et al. (2017), Bucena and Kirikova (2017), Hussain et al. (2017), Sturm et al. (2017), Gazivoda (2018) and Silva et al. (2018)	9
A7	Process	Hüttermann (2012), McCarthy et al. (2015), Sharma and Coyne (2015), Abdelkebir et al. (2017), Bucena and Kirikova (2017), Hussain et al. (2017), Sturm et al. (2017), Gazivoda (2018) and Silva et al. (2018)	9
A8	Quality	Erich et al. (2014a), Mohamed (2015) and Luz et al. (2018)	3
A9	Collaboration	Mohamed (2015) and Luz et al. (2018)	2
A10	Diy Deployments	Debois (2011)	1
A11	Agility	Luz et al. (2018)	1
A12	Resilience	Luz et al. (2018)	1
A13	Transparency	Luz et al. (2018)	1
A14	Services	Erich et al. (2014a)	1
A15	Structures	Erich et al. (2014a)	1
A16	Standards	Erich et al. (2014a)	1
A17	Governance	Mohamed (2015)	1

This leave us with the four main areas: Culture, Technology, People and Process. In order to study the practices from the Capabilities in a determined areas, all the documents that were used in the SLR of the Capabilities and the Areas were analysed.

Table 8 presents all the Continuous Deployment practices found for DevOps capability, ordered by area. The rest of the capabilities can be seen in Appendix (Tables 14–18).

5.4 *Step 4 (maturity levels)*

The results of each conducted interview iteration are presented, followed by the associated emerging final MM for DevOps.

5.4.1 *First iteration*

To perform the first round of interviews, 15 DevOps professionals were interviewed.

The LinkedIn database was used to find the interviewees. Overall, 87 invites were made to DevOps experts and 33 were accepted. In this list of 33 contacts, only 15 responded to the interview.

In this research, it was considered the position of the possible participant, always willing to interview professionals with higher positions than DevOps developers. Interviewees information can be seen in Table 9.

Although some of the DevOps capabilities already exists, the term DevOps was born in 2011. The average age of the 15 interviewed is 39.4 years, while the average experience in DevOps is 5.6 years. Since DevOps was born nine years ago, 5.6 years in average of experience means that the interviewed have been working in this area during more than half of its existence as a practice. Plus, 13 out of the 15 interviewees work in the IT sector.

The same interviewer conducted all the 15 interviews ensuring that the same interview guides and protocol were used throughout the interviews. The first, second, third, fourth and last interviews were conducted in the participants' workplace, while the rest were carried out by Skype. The interview was semi-structured and aimed at exploring practitioners' experiences with DevOps practices. All the 15 interviews were conducted between March and June 2019.

The authors have interviewed DevOps practitioners according to a preset script which included semi-structured open-ended questions. The interview guideline addressed topics such as the expert's background, expert's team and company information, DevOps practices and observations about it.

Grounded on maturity levels classification, and since all organisation are at level 1 (ad-hoc) by default, the authors have only asked the interviewees to associate the practices with levels 2, 3, 4 and 5. The distribution of the practices by levels is presented in Table 10.

5.4.2 *Second iteration*

All the 15 interviewees from the first iteration were asked to participate in a second round. From those, 13 accepted to participate. The objective of this phase was to breakdown the practices that had the same number of votes to more than one level of maturity and try to reach consensus on all practices. Therefore, the participant had a chance to choose between the most voted levels of the first phase in each of the enlisted practices.

Table 8 CD practices

	Continuous Deployment	
	Practice	*Author*
People	–	–
Process	Orchestrated deployments	Sharma and Coyne (2015)
	Track which version is deployed	
	Manage the configurations of the environments of all the stages	
	Manage the software components that get deployed	
	Manage the middleware components and middleware configurations that need to be updated	
	Manage the database components that need to be changed	
	Manage the configuration changes to the environments to which these components are to be deployed	
	Release working software any time, any place	Duvall et al. (2007)
	Label a repository's assets	
	Produce a clean environment	
	Label each build	
	Create build feedback reports	
	Possess capability to roll back release	
	Multiple deployments to production	Mohamed (2016)
	Deploy a new release whenever one is needed	
Technology	Development and production share a homogenous infrastructure	Ebert et al. (2016)
	Configuration management tools	
	Automated deployment of software to different environments	Nielsen et al. (2017)
	Deployments should include the automated provisioning of all environments	Debois (2011)
	Automated deployment	Nielsen et al. (2017)
	Continuous deployment	
Culture	Early and frequent involvement of operations staff in the planning stages of major new releases	Debois (2011)

All the interviews were conducted by email. The interviews were semi-structured and aimed at exploring practitioners' experiences with DevOps practices. All the 13 interviews were conducted between June and August 2019.

DevOps practitioners were interviewed according to a preset script which included semi-structured open-ended questions. The interview guideline addressed topics such as DevOps practices and observations about it. Since no relevant conclusions could be drawn from the first iteration, in this second phase the authors changed the possible answers for the DevOps practices maturity levels to the most voted levels from the first phase. This was held since there were many maturity levels for each practice.

Table 9 Interviewees details

ID	*Role*	*Age*	*DevOps experience (Years)*	*Industry*	*First iteration*	*Second iteration*
I1	Head of DevOps Transformation	41	6	Software development	X	X
I2	Solution Architect	46	8	Software development	X	X
I3	Senior Manager	41	8	Software development	X	X
I4	Senior DevOps Engineer / Team Lead	26	3	Software development	X	X
I5	Head of Agile and DevOps Transformation	38	3	Software development	X	X
I6	DevOps Manager/Evangelist	42	3	Finance	X	X
I7	Lead DevOps specialist	39	3	Healthcare	X	X
I8	DevOps Architect	38	8	Software development	X	X
I9	DevOps Operations Lead	40	3	Software development	X	X
I10	DevOps Engineer	33	4	Software development	X	X
I11	Managing Director	48	8	Software development	X	X
I12	Senior Developer	38	6	Software development	X	X
I13	Lead DevOps specialist	45	8	Software development	X	
I14	Senior Manager	39	7	Software development	X	
I15	IT Development T. Leader – Applications	37	6	Software development	X	X
Average		*39.4*	*5.6*			

Table 10 Distribution of the number of practices per level from first iteration

Level	*Frequency*
Level 2	31
Level 3	50
Level 4	19
Level 5	9

Grounded on maturity levels classification, and since all organisation are at level 1 (ad-hoc) by default, the authors only asked the interviewees to associate the practices with the most voted levels for each practice from the first phase. The distribution of the practices by levels and the difference from the first iteration are presented in Table 11.

Table 11 Distribution of the number of practices per level from second iteration

Level	*Frequency*	*Difference*
Level 2	10	–21
Level 3	54	+4
Level 4	27	+8
Level 5	18	+9

Analysing Table 11, one of the most relevant difference between the two phases is the migration of some level two responses to the other levels. There is a clear increase of level 5 votes. On the other hand, level 3 continues to be the most voted level.

Only about one third of the previous level two votes remained. Although none of the participants said anything about this, it seems that, since each participant had the chance to choose from the most voted level from the first iteration, they considered a higher level since that it was a possibility. Also, since that two from the first iteration interview did not answer this issue, it may have had an influence on this result.

The most voted levels are concentrated in two levels: three and four. The participants only considered 18 practices to belong to a much higher maturity level (level 5). Since level three is one of the most basic level, it had a much higher number of practices.

5.4.3 Maturity model

Although it is a single model, for its better comprehension, it was divided into six parts, one for each capability. Even though the interviewees had the chance to add or remove practices from the initial list, none of them did. This means that the initial list of DevOps practices remained unchanged through all these interview phases. Although every participant had the chance to remove a practice and/or add an observation, there were only few cases where it happened. However, since it was not coherent nor consistent among the participants, those removed practices and observations were not taken in consideration.

Each MM table is divided by areas (People, Process, Technology and Culture) in which are presented the respective practices. Table 12 present the MM for Continuous Deployment. The rest of the MM can be seen in Appendix (Tables 19–24). According these tables that, together, integrate the MM for DevOps, an analysis has been made.

Observing Table 19, it is possible to devise that there is only one practice from level 2. Level 3 is the level with more practices and level 4 and level 5 almost have the same number of practices. If we look to the practices per area, since the author was not able to find any practice associated with this area and the interviewees did not add any, People does not have any practice. On the other hand, Process seems to be the area with more practices, since it has at least one at each level.

In both Tables 19 and 20, people area does not have any associated practice. On the other hand, level 2 is more populated than it was in Table 19. Level 3 is the level with more practices, while Process continues to be the area of DevOps with more practices. Technology has at least a practice per level.

In the Continuous Monitoring (Table 21) it is possible to see the first practice for the People's area and is the only practice for the level 2 on this table. Process and Technology have practices from the level 3 to level 5.

Table 22 is the people's area contains more practices than the tables before. There are three People practices and they are all in level 3. Culture is the most completed area in this table, since it has practices in every level. Level 5 only has one practice.

Table 23 is the one with less practices. The author could not identify more practices from the literature and the interviewees did not add any. Level 3 is the most populated level and there is only on practice that does not belong to this level. Technology is the Area with most practices. On the other hand, there is no practice in People's area.

Last but not least, Table 24 presents all the practices from Feedback Loops capability. There was not found any practice in level 2. Level 3 only have practices for the Process area, while level 4 contains practices for People, Process and Culture. Culture seems to be an area where all its practices are from a greater maturity, since three out of four practices presented in this area belong to level 5. The level with more practices is level 4.

After analysing all the tables that contained the MM for DevOps, a last analysis must be conducted. The preliminary list for the MM was conducted by the author, through a literature review. Although the fact that all the interviewees had the chance to add or remove any practices they want, none of them did. This result in some capabilities with less practices than others, and some areas with just few practices. If any of them had less than four practices, it means that there will be levels with no practices.

People is the area with less practices from the four. On the other hand, Process, followed by Technology are the areas with more practices. Level 3 is the level with most practices while level 2 is the one with less practices. This may be due to the lack of literature about this theme.

6 Demonstration

In order to demonstrate the artefact, two teams fully compliant with DevOps were assessed. Then, an interview was held with DevOps teams where the proposed MM was tested. The objective is to demonstrate that the MM fulfils the purpose it was designed to applying it in a professional environment. Since not all capabilities or areas have practices, only the capabilities/areas with at least one practice have been considered to assess team's maturity. According with CMMI, which has been previously presented, a level can only be reached if all the practices from that level are executed.

6.1 First demonstration

The first team assessed operates in the services sector, in the field of Cloud and DevOps consulting. The person responsible to conduct this demonstration is the DevOps Operations Lead with three years of experience in DevOps. Figure 7 shows the maturity of the DevOps in this team.

Figure 7 shows the maturity of the first team. As it evidences, the most matured capability is the Feedback Loops, followed by CI.

At level 4, Feedback Loops has a maturity level almost all areas at level 5, if it was not by the People's area. This means that the team has all the practices implemented for Culture and Processes, and a big part of the People's practices. Looking to the CI, Technology is at its maximum, level 5. Culture is the next area with more maturity and Process is at the end.

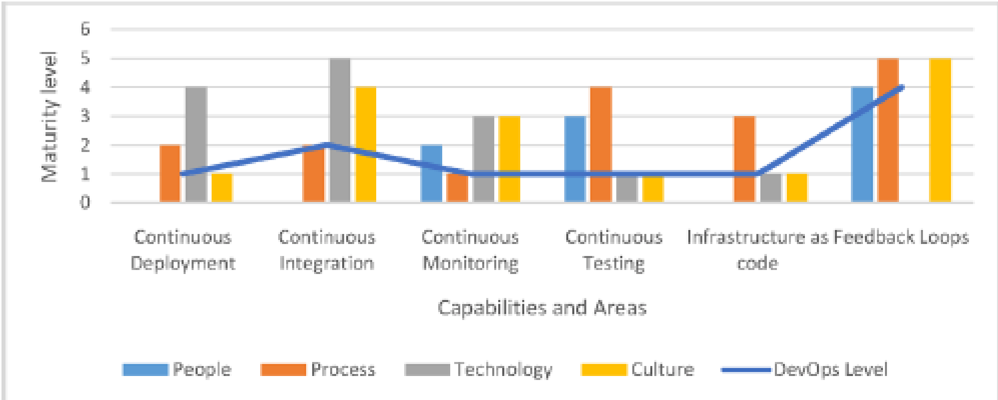
Table 12 CDMM

		Level 2	*Level 3*	*Level 4*	*Level 5*
Continuous deployment	People				
	Process	CD9 Label a repository's assets	CD2 Track which version is deployed CD3 Manage the configurations of the environments of all the stages CD4 Manage the software components that get deployed CD5 Manage the middleware components and middleware configurations that need to be updated CD6 Manage the database components that need to be changed CD10 Produce a clean environment CD11 Label each build CD12 Create build feedback Reports CD14 Deploy a new release whenever one is needed CD17 Automated deployment CD18 Continuous deployment	CD1 Orchestrated deployments CD16 Deployments should include the automated provisioning of all environments	CD1 Orchestrated deployments CD7 Manage the configuration changes to the environments to which these components are to be deployed CD8 Release working software any time, any place CD15 Multiple deployments to production
	Technology	–	CD19 Development and production share a homogenous infrastructure CD20 Configuration management tools	CD21 Automated deployment of software to different environments	–

Table 12 CDMM (continued)

		Level 2	*Level 3*	*Level 4*	*Level 5*
Continuous deployment	Culture	–	CD22 Team must provide overall visibility into your application release activities and timing to all major stakeholders CD25 Unite the two teams that worked independently to work at tighter integration CD26 Both development and operations personnel should share the same knowledge management resources CD27 Testers and operations personnel would be able to self- service deployments of the required version of the system to their environments on demand CD28 Early and frequent involvement of operations staff in the planning stages of major new releases	CD24 Team must be able to speed lead times and make more frequent application deployments at the pace demanded by the business	CD23 Teams must be able to provide self-service, on-demand provisioning and management of cloud environments and infrastructure resources

Figure 7 First demonstration maturity (see online version for colours)



Looking to the other capabilities, they all are at level 2. Continuous Monitoring has 3 areas at level 3 and seems to be the next most maturated capability.

In a more general view, the most maturated capability is Feedback Loops. The most maturated area is Process.

6.2 Second demonstration

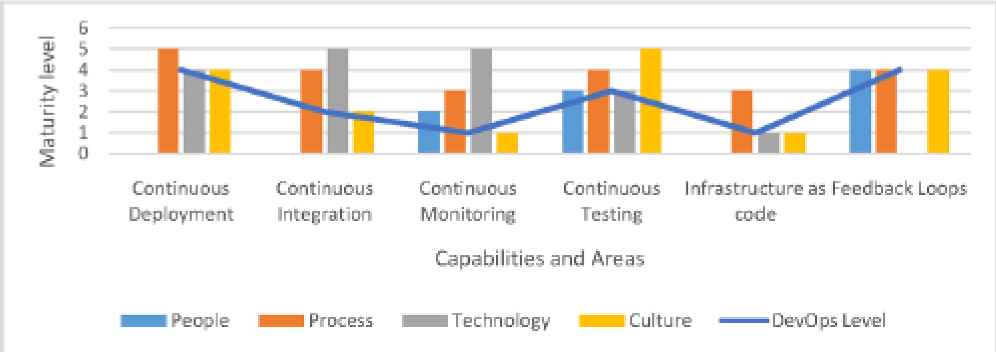
The second team is from the SD industry. The person responsible to conduct this demonstration is the Senior Manager with eight years of experience in DevOps. Figure 8 shows the maturity of the DevOps in this team. The Y axis represent the maturity level and the X axis represent the DevOps capabilities and areas.

Looking at this figure, it is perceptible that this team has, in general, a much higher maturity than the previous one. Two capabilities at level 4 and one in level 3. CD, Feedback Loops are the most matured capabilities while Infrastructure as a Code is the less matured one.

Looking to the CD graphic, one of the areas reached level 5, while the others are at level 4. Feedback loops has all its areas with similar maturity levels. Continuous Testing has one area in level 5, one in level 4 and the others in level 3.

CI, although it has 1 area in level 5 and another one in level 4, it is only in the maturity level 2, due to its lack of culture maturity. Continuous Monitoring has the same problem: although it has 1 area in level 5, one in level 4 and another in level 3, its maturity is only 2. The most immature capability is Infrastructure as a Code. On the three areas evaluated, only one is above level 2.

Figure 8 Second demonstration maturity (see online version for colours)



7 Evaluation and communication

Following the Pries-Heje et al. (2008) approach, in which the authors present the importance of an ex ante perspective, with the evaluation occurring both prior to the construction of an artefact IS, and an ex post evaluation, that is, evaluations that take place after the artefact has been built. Plus, Venable identifies two main forms for the DSRM evaluation (Venable, 2006):

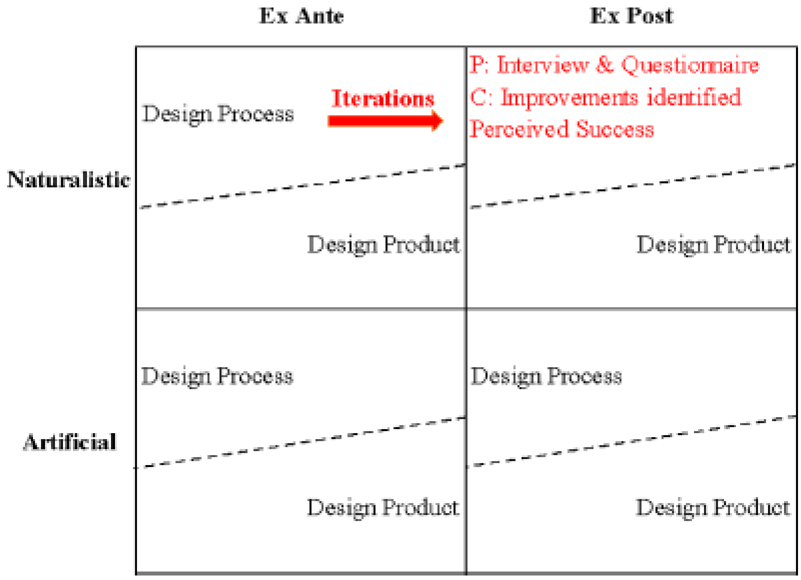
- *artificial evaluation* is evaluating a solution technology in a contrived, non-real way
- *naturalistic evaluation* enables the authors to explore how well or poorly a solution technology works in its real environment – the organisation.

Furthermore, an additional dichotomy is incorporated into the Pries-Hege's framework, which is comprised of the design product and design process. Using the definition of Dubin for each aspect of design theory (Dubin, 1976):

- *design product* is "a plan of something to be done or produced"
- *design process* is "to so plan and proportion the parts of a machine or structure that all requirements will be satisfied"

By distinguishing all these concepts, it is possible to map the objectives of evaluation and what is more accurately adapted to the artefact constructed in order to prove the utility, effectiveness and other criteria, as shown in Figure 9. This framework for the DSRM evaluation is supposed to facilitate the answer to the following questions – 'What' is evaluated, 'When' to evaluate, and 'How' to evaluate. Figure 9, helps us to answer these questions by providing a high-level perspective, also considering that "P summarises the essential characteristics of the evaluation Process, while C indicates the evaluation criteria" (Pries-Heje et al., 2008).

Figure 9 Strategic DSRM evaluation framework (see online version for colours)



Source: Adapted from Pries-Heje et al. (2008)

However, further details are needed to answer these questions and several decisions need to be made. This non-compliance is fulfilled with the proposed framework by Venable et al. (2012) that is intended to be a complement to the strategic DSRM evaluation framework mentioned above, providing for example a guide on how to select evaluation methods.

The DSRM evaluation method selection framework suggests possible evaluation methods. For the current study, Survey was selected, in a form of interviews and questionnaires.

Concerning research communication, a part of this research is presented by one paper and the whole research is represented by this document. The authors will now show the evaluation that was given by the demonstration inquires, where the constructed MM was applied by DevOps practitioners in its teams. The authors asked the participant to evaluate the proposed MM: the inquired person had the chance to say anything he wanted about this MM, if it was useful, complete or applicable in real life cases.

This first evaluation corresponds to the First demonstration case, where the participant of 40 years old and three years of experience on the DevOps field applied the MM in his team. The second evaluation is from the SD industry, where the participant is responsible to conduct this demonstration is the Senior Manager with eight years of experience in DevOps. The following was stated (Table 13):

The participants evaluated the MM positively as it can be seen in evidenced by their feedback. On the first case, the participant said that it is a valuable work and it can be a good help for the DevOps implementation. The participant also said that as a service provider, some practices can be hard to get through because they are a true challenge to implement.

The second participant in the evaluation stated that this MM is a useful tool to know the maturity of DevOps in a team. The fact that the MM was build based on the literature and improved with DevOps practitioners, gives this research more credibility. Although the participant considers this MM complete, for him, it could get better if all the Areas had at least one practice, so it can measure the maturity of all the DevOps.

Taking these two evaluations in consideration, the feedback received is positive. Both participants thought this is a useful tool to measure the DevOps adoption. By the feedback, it is possible to perceive that this MM is applicable in real cases. The suggestion of improving the model to have at least one practice in each area is shared by the authors. However, it was not possible to find in the literature studies that deeply explore DevOps and the people interviewed for the construction of this MM did not add any practice.

Table 13 Evaluations of the MM applicability

ID	*Evaluation*
E1	"You produced such valuable work. This list can act as a service menu for a DevOps process and culture implementation and at the same time this will help the person in charge of the DevOps transformation keep the focus on what should be delivered to the stakeholders. As a service provider, I cannot deny the difficulty to address some targets of your work with my clients. For example, when you are working to transform an ITIL organisation to an Agile/DevOps organisation, people tend to refrain the changes and points as the "Share the feedback freely without blame" are a true challenge to be implemented. For me, decide which parts of your practices should or not be implemented is a matter to balance the client needs, the size of the client organisation and keep the process as simple as possible."
E2	"It is hard to find DevOps practices in the existent literature. It is even harder to understand what is important and what is the correct order to implement, so the team has solids basis. This work provides an interesting set of DevOps practices, divided by the most important capabilities. It is even better because I can have a vision by area. Applying this MM to our team gave me insight into what should be implemented and in what order. Knowing that this was made with interviews to DevOps practitioners give me more confidence in using this model as basis to future team improvements decisions, as I can rely on this research. This is a useful tool if you want to know the maturity of your team in DevOps. Although I believe that it is a complete tool, I would consider it more complete if it has more practices. At least, if every capability and every area had at least one practice."

8 Conclusions

In this research two SLR were conducted to respond to the call by authors and practitioners for a deeper theoretical and practical understanding of DevOps capabilities and areas that could ~~work as determinant factors~~ and contribute to the implementation of

DevOps. Then, a total of 28 interviews were performed with DevOps practitioners. With their experience, the interviewees helped to assign a specific maturity level for each DevOps practice. At the end of the previous steps, the proposed MM for DevOps was then completed. Grounded on the previous sections one may argue that all the proposed Research Objectives were achieved:

- Regarding RO1.1, the main DevOps areas were elicited and described, and they specifically include culture, measurement, sharing, automation, technology, people and process.
- Concerning RO1.2, the main DevOps capabilities have been also identified and detailed. The elicited capabilities include CI, CD, continuous testing, feedback loops between Dev and Ops and infrastructure as code.
- After these sub-objectives are met, a MM for DevOps was built. It was sustained on the previous main areas and main capabilities. It was developed a new DevOps MM based on CMMI MM to enable assessing any organisation working model/state against DevOps model.

Regarding this, the main objectives that this research proposed were hit. Despite this, it was possible to conclude the following set of insights:

- Both DevOps practitioners and scientific studies continue to increase since 2015. This study also identified some relationships between the DevOps areas and capabilities based on the analysis of Figure 7. The documents that focus on the DevOps culture are most likely to relate it to all of the main capabilities found. On the other hand, it is more difficult to find a document that relates Technology, People and Process with the main capabilities.
- The capabilities of CI and CD are the more investigated in the literature. The areas that most relate with them are Culture, Sharing and Automation. These three areas are the most referred DevOps areas in the literature. Processes seems to be the area that less influences the capabilities, while Infrastructure as Code is the capability which the fewest studies tend to relate with DevOps.
- This research has brought contributions to the academic and scientific community by exploring a field that had not yet been explored and proposing a novel artefact. It has also improved the knowledge base and endeavoured to lay down new bases for further research.
- This research is a new systematised contribution to knowledge, through the identification of patterns that have been recognised in the literature –and that, as such, corresponds to a new level of knowledge in the approach to the topic. This research also provides some contributions for professionals and practitioners. In the absence of studies exploring the DevOps main capabilities and DevOps areas, and even the relationship between them, this research brings new insights on how and why practitioners should adopt DevOps practices and which areas they have to change or, at least, keep in mind as being relevant for an effective adoption of DevOps.

- Based on these findings, and using the summarised information provided in this work as a starting point, the authors deepened the identified DevOps areas and capabilities to be an a priori and open model, which was the target of this research project – which aimed to test and refine this systematised view (in the form of a MM), having not only implications for existing scientific knowledge but also being useful for organisational practices of DevOps.

8.1 Limitations

Regarding limitations, it was not possible to gather enough information and present a robust conclusion regarding specific topics, such as Outcomes, since DevOps is a recent subject. The current research cannot fully avoid biases since it has excluded literature sources written in other languages or unavailable in electronic databases. Since DevOps is recent, there are not a lot of experts in this area. This limited the interviews on each phase (Prates et al., 2019).

8.2 Future work

In the future, research should be carried out into the most referenced capabilities, CI and CD and the most referenced areas, Culture, Sharing and Automation, as they seem to be essential in the DevOps movement. Also, it would be interesting to deeply explore the relationship between CI and Culture, Sharing and Automation, as these areas seem to relate the most with the main capability found among this literature review. It is also expected that the proposed MM needs to be evolved in the future. As any other research this process is normal. But in this case is even more evident since DevOps is recent and more organisations are implementing it and academics investigating it.

References

Abdelkebir, S., Maleh, Y. and Belaissaoui, M. (2017) 'An agile framework for ITS management in organizations', *Proceedings of the 2nd International Conference on Computing and Wireless Communication Systems – ICCWCS'17*, pp.1–8, doi: 10.1145/3167486.3167556.

Aguiar, J., Pereira, R., Vasconcelos, J. and Bianchi, I. (2018) 'An overlapless incident management maturity model for multi-framework assessment (ITIL, COBIT, CMMI-SVC)', *Interdisciplinary Journal of Information, Knowledge, and Management*, doi: 10.28945/4083.

Ali, E., Caputo, A. and Lawless, S. (2017) 'Entity attribute ranking using learning to rank', *CEUR Workshop Proceedings*, Vol. 1883, pp.19–24, doi: 10.1145/1235.

Andersen, E.S. and Jessen, S.A. (2003) 'Project maturity in organisations', *International Journal of Project Management*, Vol. 21, No. 6, pp.457–461, doi: 10.1016/S0263-7863(02)00088-1.

Azoff, M. (2016) 'Ovum decision matrix: selecting a DevOps release management how enterprises can improve their software application delivery', *Ovum*, pp.1–8.

Bai, X., Li, M., Pei, D., Li, S. and Ye, D. (2018) 'Continuous delivery of personalized assessment and feedback in agile software engineering projects', *Proceedings – International Conference on Software Engineering*, ACM, Part F1373, pp.58–67, doi: 10.1145/3183377.3183387.

Bang, S., Chung, S. Choh, Y. and Dupuis, M. (2013) 'A grounded theory analysis of modern web applications', *Proceedings of the 2nd Annual Conference on Research in Information Technology – RIIT '13*, p.61, doi: 10.1145/2512209.2512229.

Barbosa, D.F., Furtado, E.S. and Gomes, A.S. (2007) 'Uma proposta de institucionalização da usabilidade alinhada com práticas do modelo CMMI e foco nas necessidades da organização', *Proceedings of VII Brazilian Symposium on Human Factors in Computing Systems*, pp.45–48, doi: 10.1145/1298023.1298060.

Baskarada, S., Gao, J. and Koronios, A. (2005) 'Agile maturity model approach to assessing and enhancing the quality of asset information in engineering asset management information systems', *Proceedings of the 9th International Conference on Business Information Systems*, pp.66–68.

Bass, L. (2017) 'The software architect and DevOps', *IEEE Software*, Vol. 35, No. 1, pp.8–10, doi: 10.1109/MS.2017.4541051.

Becker, J., Knackstedt, R. and Pöppelbuß, J. (2009) 'Developing maturity models for IT management', *Business & Information Systems Engineering*, Vol. 1, No. 3, pp.213–222, doi: 10.1007/s12599-009-0044-5.

Berssaneti, F.T., De Carvalho, M.M. and Muscat, A.R.N. (2012) 'Impacto dos modelos de referência e maturidade no gerenciamento de projetos: estudo exploratório em projetos de tecnologia da informação', *Production*, Vol. 22, No. 3, pp.404–435, doi: 10.1590/S0103-65132012005000027.

Bezemer, C., Eismann, S. Ferme, V., Grohmann, J., Heinrich, R., Jamshidi, P., Shang, W., Hoorn, A., Villaviencio, M., Walter, J. and Willnecker, F. (2018) *How is Performance Addressed in DevOps? A Survey on Industrial Practices*, arXiv:1808.06915v1 [cs.SE], August, doi: 10.1145/3297663.3309672.

Bhattacharjee, A., Barve, Y., Gokhale, A. and Kuroda, T. (2018) '(WIP) CloudCAMP: automating the deployment and management of cloud services', *2018 IEEE International Conference on Services Computing (SCC)*, IEEE, pp.237–240, doi: 10.1109/SCC.2018.00038.

Bucena, I. and Kirikova, M. (2017) 'Simplifying the devops adoption process', *CEUR Workshop Proceedings*, pp.32–35.

Buglione, L. (2011) 'Light maturity models (LMM)', *Proceedings of the 12th International Conference on Product Focused Software Development and Process Improvement – Profes '11*, Vol. 5, No. Lmm, p.57, doi: 10.1145/2181101.2181115.

Chen, B. (2019) 'Improving the software logging practices in DevOps', *2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)*, IEEE, pp.194–197, doi: 10.1109/icse-companion.2019.00080.

Chen, H.M., Kazman, R., Haziyev, S., Kropov, V. and Chtchourov, D. (2015) 'Architectural support for DevOps in a neo-metropolis BDaaS platform', *Proceedings of the IEEE Symposium on Reliable Distributed Systems*, January, pp.25–30, doi: 10.1109/SRDSW.2015.14.

Chen, L. (2018) 'Microservices: architecting for continuous delivery and DevOps', *Proceedings – 2018 IEEE 15th International Conference on Software Architecture, ICSA 2018*, pp.39–46, doi: 10.1109/ICSA.2018.00013.

Chrissis, M.B., Konrad, M.D. and Shrum, S. (2010) *CMMI for Development*, Version 1.3, Carnegie Mellon University, doi: CMU/SEI-2010-TR-033 ESC-TR-2010-033.

Clevand, S.B., Dooley, R., Perry, D., Stubbs, J., Fonner, J.M. and Jacobs, G.A. (2018) 'Building science gateway infrastructure in the middle of the pacific and beyond', *Proceedings of the Practice and Experience on Advanced Research Computing – PEARC '18*, Vol. Ci, pp.1–8, doi: 10.1145/3219104.3219151.

Colomo-Palacios, R., Fernandes, E., Soto-Acosta, P. and Larrucea, X. (2018) 'A case analysis of enabling continuous software deployment through knowledge management', *International Journal of Information Management*, Elsevier, Vol. 40, October, pp.186–189, doi: 10.1016/j.ijinfomgt.2017.11.005.

Croker, M. and Hering, M. (2016) *DevOps: Delivering at the Speed of Today's Business DevOps: A Matter of Survival in the Digital Age.*

De Bayser, M., Azevedo, L.G. and Cerqueira, R. (2015) 'ResearchOps: the case for DevOps in scientific applications', *Proceedings of the 2015 IFIP/IEEE International Symposium on Integrated Network Management, IM 2015*, Vol. 59, No. 3, pp.1398–1404, doi: 10.1109/INM.2015.7140503.

de França, B.B.N., Jeronimo, H. and Travassos, G.H. (2016) 'Characterizing DevOps by hearing multiple voices', *Proceedings of the 30th Brazilian Symposium on Software Engineering – SBES '16*, pp.53–62, doi: 10.1145/2973839.2973845.

Debois, P. (2011) 'DevOps: a software revolution in the making', *Cutter IT Journel*, Vol. 24, No. 8, p.34, doi: 10.1016/B978-0-08-025947-5.50004-2.

Debroy, V., Miller, S. and Brimble, L. (2018) 'Building lean continuous integration and delivery pipelines by applying DevOps principles: a case study at Varidesk', *Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering – ESEC/FSE 2018*, pp.851–856, doi: 10.1145/3236024.3275528.

Diel, E., Marczak, S. and Cruzes, D.S. (2016) 'Communication challenges and strategies in distributed DevOps', *2016 IEEE 11th International Conference on Global Software Engineering (ICGSE)*, IEEE, pp.24–28, doi: 10.1109/ICGSE.2016.28.

Dubin, R. (1976) 'Theory building in applied areas', *Handbook of Industrial and Organizational Psychology*, pp.13–16.

Düllmann, T.F., Paule, C. and Van Hoorn, A. (2018) 'Exploiting devops practices for dependable and secure continuous delivery pipelines', *Proceedings – International Conference on Software Engineering*, Part F1378, pp.27–30, doi: 10.1145/3194760.3194763.

Duvall, P.M., Matyas, S. and Glover, A. (2007) *Continuous Integration: Improving Software Quality and Reducing Risk*, Addison-Wesley signature series, Boston, USA.

Ebert, C., Gallardo, G., Hernantes, J. and Serrano, N. (2016) 'DevOps', *IEEE Software*, Vol. 33, No. 3, pp.94–100, doi: 10.1109/MS.2016.68.

Erich, F., Amrit, C. and Daneva, M. (2014a) 'Cooperation between information system development and operations', *Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement – ESEM '14*, pp.1–1, doi: 10.1145/2652524.2652598.

Erich, F., Amrit, C. and Daneva, M. (2014b) *DevOps Literature Review*, University of Twente, October, p.27, doi: 10.1007/978-3-319-13835-0.

Farkas, A. and Walsh, C. (2002) 'A perspective of the common criteria in modern IT business', *3rd International Common Criteria Conference*, May, pp.1–8, Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.134.1073&rep=rep1&type=pdf

Farshchi, M., Schneider, J., Weber, I. and Grundy, J. (2015) 'Experience report: anomaly detection of cloud application operations using log and cloud metric correlation analysis', *2015 IEEE 26th International Symposium on Software Reliability Engineering (ISSRE)*, pp.24–34, doi: 10.1109/ISSRE.2015.7381796.

Feijter, R., Vliet, R., Jagroep, E., Overbeek, S. and Brinkkemper, S. (2017) *Towards the Adoption of DevOps in Software Product Organizations: A Maturity Model Approach*, Technical Report UU-CS-2017-009, Department of Information and Computing Sciences, Utrecht University, Utrecht, The Netherlands, May, pp.1–173.

Feitelson, D.G., Frachtenberg, E. and Beck, K.L. (2013) 'Development and deployment at facebook', *IEEE Internet Computing*, Vol. 17, No. 4, pp.8–17, doi: 10.1109/MIC.2013.25.

Fitzgerald, B. and Stol, K-J. (2014) 'Continuous software engineering and beyond: trends and challenges', *Proceedings of the 1st International Workshop on Rapid Continuous Software Engineering – RCoSE 2014*, pp.1–9, doi: 10.1145/2593812.2593813.

Fontana, R.M., Junior, V.M., Reinehr, S. and Malucelli, A. (2015) 'Progressive outcomes: a framework for maturing in agile software development', *Journal of Systems and Software*. Elsevier Ltd., Vol. 102, pp.88–108, doi: 10.1016/j.jss.2014.12.032.

Fördős, V. and Cesarini, F. (2016) 'CRDTs for the configuration of distributed Erlang systems', *Proceedings of the 15th International Workshop on Erlang – Erlang 2016*, pp.42–53, doi: 10.1145/2975969.2975974.

Gazivoda, M. (2018) *Application of DevOps Approach in Developing Business Intelligence System in Bank*, June, pp.11–14.

Gupta, V., Kapur, P.K. and Kumar, D. (2017) 'Modeling and measuring attributes influencing DevOps implementation in an enterprise using structural equation modeling', *Information and Software Technology*, Elsevier B.V., Vol. 92, pp.75–91, doi: 10.1016/j.infsof.2017.07.010.

Hanappi, O., Hummer, W. and Dustdar, S. (2016) 'Asserting reliable convergence for configuration management scripts', *ACM SIGPLAN Notices*, Vol. 51, No. 10, pp.328–343. doi: 10.1145/3022671.2984000.

Hart, C. (1998) 'Doing a literature review: releasing the social science research imagination', *SAGE: London*, Vol. 1, No. 1, pp.1–25, doi: 10.1080/01422419908228843.

Hevner, A.R., March, S.T., Park, J. and Ram, S. (2004) 'Design science in information systems research', *MIS Quarterly*, doi: 10.2307/25148625.

Humble, J. and Farley, D. (2011) *Continuous Delivery*, Addison-Wesley Signature Series, doi: 10.1017/CBO9781107415324.004.

Hussain, W., Clear, T. and MacDonell, S. (2017) 'Emerging trends for global DevOps: a New Zealand perspective', *Proceedings – 2017 IEEE 12th International Conference on Global Software Engineering, ICGSE 2017*, IEEE, pp.21–30, doi: 10.1109/ICGSE.2017.16.

Hüttermann, M. (2012) 'Introducing DevOps', *DevOps for Developers (Part I)*, pp.15–31, doi: 10.1007/978-1-4302-4570-4_2.

Jabbari, R., Ali, N., Petersen, K. and Tanveer, B. (2016) 'What is DevOps?', *Proceedings of the Scientific Workshop Proceedings of XP2016 on – XP '16 Workshops*, pp.1–11, doi: 10.1145/2962695.2962707.

Jimenez, I., Sevilla, M., Watkins, N., Maltzahn, C., Lofstead, J., Mohror, K., Arpaci-Dusseau, A.C. and Arpaci-Dusseau, R. (2017) 'The popper convention: Making reproducible systems evaluation practical', *Proceedings – 2017 IEEE 31st International Parallel and Distributed Processing Symposium Workshops, IPDPSW 2017*, pp.1561–1570, doi: 10.1109/IPDPSW.2017.157.

John, W., Meirosu, C., Pechenot, B., Sköldström, P., Kreuger, P. and Steinert, R. (2015) 'Scalable software defined monitoring for service provider DevOps', *Proceedings – European Workshop on Software Defined Networks, EWSDN*, pp.61–66, doi: 10.1109/EWSDN.2015.62.

Karapantelakis, A., Liang, H., Wang, K., Vandikas, K., Inam, R., Fersman, E., Mulas-Viela, I., Seyvet, N. and Giannokostas, V. (2016) 'DevOps for IoT applications using cellular networks and cloud', *Proceedings – 2016 IEEE 4th International Conference on Future Internet of Things and Cloud, FiCloud 2016*, pp.340–347, doi: 10.1109/FiCloud.2016.55.

Katal, A., Bajoria, V. and Dahiya, S. (2019) 'DevOps: bridging the gap between development and operations', *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, IEEE, pp.1–7, doi: 10.1109/iccmc.2019.8819631.

Kawamoto, S. and De Almeida, J.R. (2017) 'Scrum-DR: an extension of the scrum framework adherent to the capability maturity model using design rationale techniques', *2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies, CHILECON 2017 – Proceedings*, January, pp.1–7, doi: 10.1109/CHILECON.2017.8229530.

Khan, K.S., Kunz, R., Kleijnen, J. and Antes, G. (2004) 'Systematic reviews to support evidence-based medicine: how to review and apply findings of healthcare research', *Postgraduate Medical Journal*, pp.53–62, doi: 10.1016/S1036-7314(00)70624-2.

Kitchenham, B. (2004) *Procedures for Performing Systematic Reviews*, Joint Technical Report, Computer Science Department, Keele University, doi: 10.1.1.122.3308.

Koilada, D.K. (2019) 'Business model innovation using modern DevOps', *2019 IEEE Technology & Engineering Management Conference (TEMSCON)*, IEEE, (Cd), pp.1–6, doi: 10.1109/temscon.2019.8813557.

Kuusinem, K., Balakumar, V., Jepsen, S.C., Larsen, S.H., Lemqvist, T.A., Muric, A., Nielsen, A. and Vestergaard, O. (2018) 'A large agile organization on its journey towards DevOps', *Proceedings of the 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, Prague, pp.60–63. doi: 10.1109/SEAA.2018.00019.

Lahrmann, G., Marx, F., Mettler, T., Winter, R. and Wortmann, F. (2011) 'Inductive design of maturity models: applying the Rasch algorithm for design science research', *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, doi: 10.1007/978-3-642-20633-7_13.

Laukkarinen, T., Kuusinen, K. and Mikkonen, T. (2017) 'DevOps in regulated software development: case medical devices', *Proceedings – 2017 IEEE/ACM 39th International Conference on Software Engineering: New Ideas and Emerging Results Track, ICSE-NIER 2017*, pp.15–18, doi: 10.1109/ICSE-NIER.2017.20.

Laukkarinen, T., Kuusinen, K. and Mikkonen, T. (2018) 'Regulated software meets DevOps', *Information and Software Technology*, Elsevier B.V., Vol. 97, pp.176–178, doi: 10.1016/j.infsof.2018.01.011.

Lewerentz, M., Bluhm, T., Daher, R., Dumke, S., Grahl, M., Grün, M., Holtz, A., Krom, J., Kühner, G., Laqua, H., Riemann, H., Spring, A., Werner, A. and W7-X Team (2018) 'Implementing DevOps practices at the control and data acquisition system of an experimental fusion device', *Fusion Engineering and Design*, Elsevier, November, pp.0–1, doi: 10.1016/j.fusengdes.2018.11.022.

Li, Z., Zhang, Y. and Liu, Y. (2017) 'Towards a full-stack devops environment (platform-as-a-service) for cloud-hosted applications', *Tsinghua Science and Technology*, Vol. 22, No. 1, pp.1–9, doi: 10.1109/TST.2017.7830891.

Luz, W.P., Pinto, G. and Bonifácio, R. (2018) 'Building a collaborative culture', *Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement – ESEM '18*, ACM Press, New York, USA, pp.1–10, doi: 10.1145/3239235.3240299.

Lwakatare, L.E., Kuvaja, P. and Oivo, M. (2015) 'Dimensions of devOps', *Lecture Notes in Business Information Processing*, pp.212–217, doi: 10.1007/978-3-319-18612-2_19.

Mackey, T. (2018) 'Building open source security into agile application builds', *Network Security*, Elsevier Ltd, Vol. 2018, No. 4, pp.5–8, doi: 10.1016/S1353-4858(18)30032-1.

Mansfield-Devine, S. (2018) 'DevOps: finding room for security', *Network Security*, Elsevier Ltd, Vol. 2018, No. 7, pp.15–20, doi: 10.1016/S1353-4858(18)30070-9.

Manzini, E. (2004) 'Entrevista semi-estruturada: análise de objetivos e de roteiros', *Seminário internacional sobre pesquisa e estudos qualitativos*, doi: 10.1590/S0036-36342005000100012.

Marijan, D., Liaaen, M. and Sen, S. (2018) 'DevOps improvements for reduced cycle times with integrated test optimizations for continuous integration', *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, IEEE, pp.22–27, doi: 10.1109/COMPSAC.2018.00012.

McCarthy, M.A. *et al.* (2015) 'Composable DevOps: automated ontology based DevOps maturity analysis', *Proceedings – 2015 IEEE International Conference on Services Computing, SCC 2015*, pp.600–607, doi: 10.1109/SCC.2015.87.

Meho, L.I. (2006) 'E-mail interviewing in qualitative research: a methodological discussion', *Journal of the American Society for Information Science and Technology*, Vol. 57, No. 2004, pp.1284–1295, doi: 10.1002/asi.

Mettler, T. (2009) *A Design Science Research Perspective on Maturity Models in Information Systems*, Technical Report: BE IWI/HNE/03, Universiteit St. Gallen, doi: 10.2174/97816080506351100101.

Mikkonen, T., Lassenius, C., Männistö, T., Oivo, M. and Järvinen, J. (2018) 'Continuous and collaborative technology transfer: software engineering research with real-time industry impact', *Information and Software Technology*, Elsevier, Vol. 95, October, pp.34–45, doi: 10.1016/j.infsof.2017.10.013.

Minayo, M.C.S. (2004) 'O desafio do conhecimento: pesquisa qualitativa em saúde', *Saúde em debate*.

Mohamed, S. I. (2016) 'DevOps maturity calculator DOMC-value oriented approach', *International Journal of Engineering Research & Science*, Vol. 2, No. 2, pp.2395–6992.

Mohamed, S.I. (2015) 'DevOps shifting software engineering strategy value based perspective', *IOSR Journal of Computer Engineering Ver. IV*, Vol. 17, No. 2, pp.2278–661, doi: 10.9790/0661-17245157.

Mohan, V. and Ben Othmane, L. (2016) 'SecDevOps: Is it a marketing buzzword? Mapping research on security in DevOps', *Proceedings – 2016 11th International Conference on Availability, Reliability and Security, ARES 2016*, pp.542–547, doi: 10.1109/ARES.2016.92.

Moltó, G., Caballer, M., Pérez, A., Alfonso, C. and Blanquer, I. (2017) 'Coherent application delivery on hybrid distributed computing infrastructures of virtual machines and docker containers', *Proceedings – 2017 25th Euromicro International Conference on Parallel, Distributed and Network-Based Processing, PDP 2017*, pp.486–490, doi: 10.1109/PDP.2017.29.

Moore, J., Kortuem, G., Smith, A., Chowdhury, N., Cavero, D. and Gooch, D. (2016) 'DevOps for the Urban IoT', *Proceedings of the Second International Conference on IoT in Urban Space – Urb-IoT '16*, pp.78–81, doi: 10.1145/2962735.2962747.

Murugesan, S. (2017) 'Opening statement', *Cutter IT Journal*, Vol. 30, No. 3, pp.3–5, doi: 10.1111/j.1467-8268.1991.tb00046.x.

Nidagundi, P. and Novickis, L. (2017) 'Towards utilization of lean canvas in the Devops software', *Environment. Technology. Resources. Proceedings of the International Scientific and Practical Conference*, Vol. 2, p.107, doi: 10.17770/etr2017vol2.2522.

Nielsen, P.A., Winkler, T.J. and Nørbjerg, J. (2017) 'Closing the IT development-operations gap: the devops knowledge sharing framework', *CEUR Workshop Proceedings*, Vol. 1898, pp.3–9.

Palihawadana, S., Wijeweera, C.H., Sanjitha, M.G., Liyanage, V.K., Perera, I. and Meedeniya, D.A. (2017) 'Tool support for traceability management of software artefacts with DevOps practices', *3rd International Moratuwa Engineering Research Conference, MERCon 2017*, pp.129–134, doi: 10.1109/MERCon.2017.7980469.

Pang, C. and Hindle, A. (2017) 'Continuous maintenance', *Proceedings – 2016 IEEE International Conference on Software Maintenance and Evolution, ICSME 2016*, pp.458–462, doi: 10.1109/ICSME.2016.45.

Patel, C. and Ramachandran, M. (2009) 'Agile Maturity Model (AMM): a software process improvement framework for agile software development practices', *Int. J. of Software Engineering, IJSE*, Vol. 2, No. I, pp.3–28, doi: 10.4304/jsw.4.5.422-435.

Peffers, K., Tuunanen, T., Gengler, C., Rossi, M., Hui, W., Virtanen, V. and Bragge, J. (2006) 'The design science research process: a model for producing and presenting information systems research', *The Proceedings of Design Research in Information Systems and Technology DESRIST 2006*, pp.12–14, doi: 10.2753/MIS0742-1222240302.

Pereira, R. and Da Silva, M.M. (2011) 'A maturity model for implementing ITIL V3 in practice', *Proceedings – IEEE International Enterprise Distributed Object Computing Workshop, EDOC*, doi: 10.1109/EDOCW.2011.30.

Pereira, R. and Serrano, J. (2020) 'A review of methods used on IT maturity models development: a systematic literature review and a critical analysis', *Journal of Information Technology*, p.026839621988687, doi: 10.1177/0268396219886874.

Perera, P., Bandara, M. and Perera, I. (2017a) 'Evaluating the impact of DevOps practice in Sri Lankan software development organizations', *16th International Conference on Advances in ICT for Emerging Regions, ICTer 2016 – Conference Proceedings*, December, pp.281–287, doi: 10.1109/ICTER.2016.7829932.

Perera, P., Silva, R. and Perera, I. (2017b) 'Improve software quality through practicing DevOps', *2017 Seventeenth International Conference on Advances in ICT for Emerging Regions (ICTer)*, IEEE, pp.1–6, doi: 10.1109/ICTER.2017.8257807.

Pozzebon, M. (2006) 'Conducting and evaluating critical interpretive research: examining criteria as a key component in building a research tradition', *Information Systems Research*, doi: 10.1007/1-4020-8095-6_16.

Prates, L., Faustino, J., Silva, M. and Pereira, R. (2019) *DevSecOps Metrics*, doi: 10.1007/978-3-030-29608-7_7.

Pries-Heje, J., Baskerville, R. and Venable, J.R. (2008) 'Association for information systems ais electronic library (AISeL) strategies for design science research evaluation', *European Conference on Information Systems*, p.87, Available at: http://aisel.aisnet.org/ecis2008

Proenca, D. (2016) 'Methods and techniques for maturity assessment', *2016 11th Iberian Conference on Information Systems and Technologies (CISTI)*, pp.1–4, doi: 10.1109/CISTI.2016.7521483.

Punjabi, R. and Bajaj, R. (2017) 'User stories to user reality: a devops approach for the cloud', *2016 IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, RTEICT 2016 – Proceedings*, pp.658–662, doi: 10.1109/RTEICT.2016.7807905.

Rahman, A., Mahdavi-Hezaveh, R. and Williams, L. (2018) 'A systematic mapping study of infrastructure as code research', *Information and Software Technology*, Elsevier B.V., November, pp.2–7, doi: 10.1016/j.infsof.2018.12.004.

Rana, R. and Staron, M. (2016) 'First international workshop on emerging trends in devops and infrastructure', *Proceedings of the Scientific Workshop Proceedings of XP2016 on – XP '16 Workshops*, pp.1–3, doi: 10.1145/2962695.2962706.

Ratislavová, K. and Ratislav, J. (2014) 'Asynchronous email interview as a qualitative research method in the humanities', *Human Affairs*, Vol. 4, No. 24, pp.452–460, doi: 10.2478/s13374-014-0240-y.

Rijo, R.P.C.L. (2008) *Framework para a Gestão de Projectos de Sistemas de Informação de Contact Centers*, February, p.326.

Riungu-Kalliosaari, L., Mäkinen, S., Lwakatare, L.E., Tiihonen, J. and Männistö, T. (2016) 'DevOps adoption benefits and challenges in practice: a case study', *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, pp.590–597, doi: 10.1007/978-3-319-49094-6_44.

Roche, J. (2013) 'Adopting DevOps practices in quality assurance', *Communications of the ACM*, Vol. 56, No. 11, pp.38–43, doi: 10.1145/2524713.2524721.

Rodríguez, P., Mäntylä, M., Oivo, M., Lwakatare, L.E., Seppänen, P. and Kuvaja, P. (2018) 'Advances in using agile and lean processes for software development', *Advances in Computers*, doi: 10.1016/bs.adcom.2018.03.014.

Röglinger, M., Schwindenhammer, L. and Stelzl, K. (2018) 'How to put organizational ambidexterity into practice – towards a maturity model', *BPM: International Conference on Business Process Management – Lectures Notes in Business Information Processing*.

Rong, G., Zhang, H. and Shao, D. (2016a) 'CMMI guided process improvement for DevOps projects', *Proceedings of the International Workshop on Software and Systems Process – ICSSP '16*, ACM Press, New York, USA, pp.76–85, doi: 10.1145/2904354.2904372.

Rong, G., Zhang, H. and Shao, D. (2016b) 'CMMI guided process improvement for DevOps projects', *Proceedings of the International Workshop on Software and Systems Process – ICSSP '16*, ACM Press, New York, USA, pp.76–85, doi: 10.1145/2904354.2904372.

Rubasinghe, I., Meedeniya, D. and Perera, I. (2018) 'Automated inter-artefact traceability establishment for devops practice', *Proceedings of the IEEE/ACIS 17th International Conference on Computer and Information Science*, pp.211–216, doi: 10.1109/ICIS.2018.8466414.

Rubasinghe, I.D., Meedeniya, D.A. and Perera, I. (2017) 'Towards traceability management in continuous integration with SAT-analyzer', *Proceedings of the 3rd International Conference on Communication and Information Processing*, pp.77–81, doi: 10.1145/3162957.3162985.

Rufino, J., Alam, M. and Ferreira, J. (2017) 'Monitoring V2X applications using DevOps and docker', *2017 International Smart Cities Conference, ISC2 2017*, pp.2–9, doi: 10.1109/ISC2.2017.8090868.

Samarawickrama, S.S. and Perera, I. (2018) 'Continuous scrum: a framework to enhance scrum with DevOps', *17th International Conference on Advances in ICT for Emerging Regions, ICTer 2017 – Proceedings*, pp.19–25, doi: 10.1109/ICTER.2017.8257808.

Santana, F., Soares, F. and Meira, L. (2013) 'An agile maturity model for software development organizations', *Proceedings of the Eighth International Conference on Software Engineering Advances (ICSEA'13)*, pp.324–328, Available at: http://s3.amazonaws.com/academia.edu.documents/31692670/ICSEA-AgileMM_v2.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1492019091&Signature=%2B72RILftao361ieM7EnKqlsWA38%3D&response-content-disposition=inline%3Bfilename%3DAn_Agile_Maturity_Model_for_

Schumacher, A., Erol, S. and Sihn, W. (2016) 'A maturity model for assessing industry 4.0 readiness and maturity of manufacturing enterprises', *Procedia CIRP*, The Author(s), Vol. 52, pp.161–166, doi: 10.1016/j.procir.2016.07.040.

Science, C. and Sciences, N. (2015) 'Full-scale software engineering FsSE 2015', *Full-scale Software Engineering FsSE*, pp.31–36.

Science, M. (2016) *Research Article Goal Oriented DevOps Transformation Framework – Metric Phased Approach*, Samer I. Mohamed Modern Science and Arts University, Faculty of Engineering, Electrical and Communication Department.

Senapathi, M., Buchan, J. and Osman, H. (2018) 'DevOps Capabilities, practices, and challenges', *Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018 – EASE'18*, pp.57–67, doi: 10.1145/3210459.3210465.

Shahin, M., Babar, M. A. and Zhu, L. (2016) 'The intersection of continuous deployment and architecting process', *Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement – ESEM '16*, pp.1–10, doi: 10.1145/2961111.2962587.

Sharma, S. (2017a) 'DevOps plays for driving innovation', *The DevOps Adoption Playbook*, pp.189–260, doi: 10.1002/9781119310778.ch5.

Sharma, S. (2017b) *The DevOps Adoption Playbook*, John Wiley & Sons, Inc., Indianapolis, Indiana, doi: 10.1002/9781119310778.

Sharma, S. and Coyne, B. (2015) *DevOps for Dummies*, John Wiley & Sons, Inc., doi: 10.1017/CBO9781107415324.004.

Shivakumar, S.K. (2017) *DevOps for Digital Enterprises*, White Paper, Infosys, Available at: https://www.infosys.com/digital/insights/documents/devops-digital-enterprises.pdf

Siddaway, A. (2014) *What is a Systematic Literature Review and How Do I Do One?*, University of Sterling, Sterling, UK.

Silva., M., Fautino, J., Pereira, R. and Mira da Silva, M. (2018) 'Productivity gains of DevOps adoption in an IT team: a case study', *27th International Conference on Information Systems Development*, pp.1–10, Available at: https://repositorio.iscte-iul.pt/handle/10071/16388

Smeds, J., Nybom, K. and Porres, I. (2015a) 'DevOps: a definition and perceived adoption impediments', *Lecture Notes in Business Information Processing*, doi: 10.1007/978-3-319-18612-2_14.

Smeds, J., Nybom, K. and Porres, I. (2015b) 'DevOps: a definition and perceived adoption impediments', in Lassenius, C., Dingsøyr, T. and Paasivaara, M. (Eds.): *Lecture Notes in Business Information Processing*, Springer International Publishing (Lecture Notes in Business Information Processing), Cham, pp.166–177, doi: 10.1007/978-3-319-18612-2_14.

Snyder, B. and Curtis, B. (2017) 'Using analytics to guide improvement during an agile-DevOps transformation', *IEEE Software*, Vol. 35, No. 1, pp.78–83, doi: 10.1109/MS.2017.4541032.

Soni, M. (2016) 'End to end automation on cloud with build pipeline: the case for DevOps in insurance industry, continuous integration, continuous testing, and continuous delivery', *Proceedings – 2015 IEEE International Conference on Cloud Computing in Emerging Markets, CCEM 2015*, pp.85–89, doi: 10.1109/CCEM.2015.29.

Srivastava, A., Bhardwaj, S. and Saraswat, S. (2017) 'SCRUM model for agile methodology', *Proceeding – IEEE International Conference on Computing, Communication and Automation, ICCCA 2017*, January, pp.864–869, doi: 10.1109/CCAA.2017.8229928.

St, D., Ab, E. and Bosch, J. (2017) 'Continuous practices and DevOps: beyond the buzz, what does it all mean?', *2017 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA) Continuous*, pp.440–448, doi: 10.1109/SEAA.2017.78.

Steffens, A., Lichter, H. and Döring, J.S. (2018) 'Designing a next-generation continuous software delivery system', *Proceedings of the 4th International Workshop on Rapid Continuous Software Engineering – RCoSE '18*, pp.1–7, doi: 10.1145/3194760.3194768.

Stojanov, I., Turetken, O. and Trienekens, J.J.M. (2015) 'A maturity model for scaling agile development', *Proceedings – 41st Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2015*, pp.446–453, doi: 10.1109/SEAA.2015.29.

Stoneham, J., Thrasher, P., Potts, T., Mickman, H. and DeArdo, C. (2016) 'DevOps case studies', *DevOps Enterprise Forum 2016*, p.46.

Sturm, R., Pollard, C. and Craig, J. (2017) 'DevOps and continuous delivery', *Application Performance Management (APM) in the Digital Enterprise*. Elsevier, pp.121–135, doi: 10.1016/B978-0-12-804018-8.00010-3.

Swanson, E.B. and Beath, C.M. (1990) 'Departmentalization in software development and maintenance', *Communications of the ACM*, pp.4–5, doi: 10.1145/78973.78976.

Tessem, B. and Iden, J. (2008) 'Cooperation between developers and operations in software engineering projects', *Proceedings of the 2008 International Workshop on Cooperative and Human Aspects of Software Engineering – CHASE '08*, pp.54–66, doi: 10.1145/1370114.1370141.

Tuma, K., Calikli, G. and Scandariato, R. (2018) 'Threat analysis of software systems: a systematic literature review', *Journal of Systems and Software*, Elsevier Inc., Vol. 144, pp.275–294, doi: 10.1016/j.jss.2018.06.073.

Ur Rahman, A.A. and Williams, L. (2016a) 'Security practices in DevOps', *Proceedings of the Symposium and Bootcamp on the Science of Security – HotSos '16*, pp.109–111, doi: 10.1145/2898375.2898383.

Ur Rahman, A.A. and Williams, L. (2016b) 'Software security in DevOps', *Proceedings of the International Workshop on Continuous Software Evolution and Delivery – CSED '16*, pp.70–76, doi: 10.1145/2896941.2896946.

Vassallo, C., Zampetti, F., Romano, D., Beller, M., Panichella, A., Penta, M. and Zaidman, A. (2017) 'Continuous delivery practices in a large financial organization', *Proceedings – 2016 IEEE International Conference on Software Maintenance and Evolution, ICSME 2016*, pp.519–528, doi: 10.1109/ICSME.2016.72.

Velasquez, N.F., Mann, A., Brown, A., Stahnke, M. and Kersten, N. (2018) *State of DevOps Report 2018*, Puppetlabs, doi: 10.1016/S0022-3913(12)00047-9.

Venable, J., Pries-Heje, J. and Baskerville, R. (2012) 'A comprehensive framework for evaluation in design science research', *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, pp.2–20, doi: 10.1007/978-3-642-29863-9_31.

Venable, J.R. (2006) 'A framework for design science research activities', *Emerging Trends and Challenges in Information Technology Management*, Vol. 2, pp.184–187, doi: 10.4018/978-1-59904-019-6.ch044.

Webster, J. and Watson, R.T. (2002) 'Analyzing the past to prepare for the future: writing a literature review', *MIS Quarterly*, Vol. 26, No. 2, pp.xiii–xxiii, doi: 10.1.1.104.6570.

Wiesche, M. (2018) 'Are you ready for devops? required skill set for devops teams DevOps teams', *Twenty-Sixth European Conference on Information Systems (ECIS2018)*, pp.12–14.

Wongkampoo, S. and Kiattisin, S. (2018) 'Atom-task precondition technique to optimize large scale GUI testing time based on parallel scheduling algorithm', *ICSEC 2017 – 21st International Computer Science and Engineering Conference 2017, Proceeding*, Vol. 6, pp.229–232, doi: 10.1109/ICSEC.2017.8443913.

Xia, C., Zhang, Y., Wang, L., Coleman, S. and Liu, Y. (2018) 'Microservice-based cloud robotics system for intelligent space', *Robotics and Autonomous Systems*, Elsevier B.V., Vol. 110, pp.139–150, doi: 10.1016/j.robot.2018.10.001.

Yin, A.P.G. (2011) 'Scrum Maturity Model. Dissertacao para obtencao do Grau de Mestre em Engenharia Informática e de Computadores', *Universidade Técnica de Lisboa*, p.165.

Yin, X., Zhang, J. and Wang, X. (2004) 'Summary for policymakers', in Intergovernmental Panel on Climate Change (Ed.): *Climate Change 2013 – The Physical Science Basis*, Cambridge University Press, Cambridge, pp.1–30, doi: 10.1017/CBO9781107415324.004.

Zhu, H. and Bayley, I. (2018) 'If docker is the answer, what is the question?', *Proceedings – 12th IEEE International Symposium on Service-Oriented System Engineering, SOSE 2018 and 9th International Workshop on Joint Cloud Computing, JCC 2018*, pp.152–163, doi: 10.1109/SOSE.2018.00027.

Zhu, L., Bass, L. and Champlin-Scharff, G. (2016) 'DevOps and its practices', *IEEE Software*, Vol. 33, No. 3, pp.32–34, doi: 10.1109/MS.2016.81.

Appendix

Table 14 CI practices

	Continuous integration	
	Practice	*Author*
People	–	–
Process	Automation of tasks	Nielsen et al. (2017)
	Provision of virtualised hardware resources via scripts (instead of doing manual configuration work)	
	Developers should make use of continuous integration, that is branch-out and merge- back their work with the software mainline (the trunk) several times a day, in order to discover integration risks as early as possible	
	Continuous integration cycles to include also software release	de França et al. (2016)
	Continuous feedback loop	
	Enable rapid automated regression testing of code changes	Marijan et al. (2018)
	Test in a clone of the production environment	Sharma (2017a)
	Make it easy for anyone to get the latest executable	
Technology	Use of cloud services	Nielsen et al. (2017)
	Tools interoperability for unifying force across diverse teams, skills, technology languages, and methodologies	
	Version Control	Humble and Farley (2011)
	An Automated Build	
	Use build servers	Sharma (2017a)
	Maintain a single-source repository	
	Automate the build	
Culture	Collaboration between teams	Luz et al. (2018)
	Development and QA teams perform unit and integration testing	Sturm et al. (2017)
	Operations participates in integration and load testing to assess operational readiness	
	Agreement of the team	Humble and Farley (2011)
	Make sure everyone can see what is happening	Sharma (2017a)

Table 15 Continuous monitoring practices

	Continuous monitoring	
	Practice	*Author*
People	Analysis skills	Wiesche (2018)
Process	Define some useful measurement metrics	Nielsen et al. (2017)
	Ensure continuous feedback provided through the monitoring process and the users	

Table 15 Continuous monitoring practices (continued)

	Continuous monitoring	
	Practice	*Author*
Process	Application monitoring	Sharma (2017b)
	System monitoring	
	Application user behaviour	
	User sentiment	
	Delivery pipeline metrics	
	Systems are monitored after deployment	Zhu et al. (2016)
	• Instrumenting your applications and your infrastructure so you can collect the data you need	Humble and Farley (2011)
	• Storing the data so it can easily be retrieved for analysis	
	• Creating dashboards which aggregate the data and present it in a format suitable for operations and for the business	
	• Setting up notifications so that people can find out about the events they care about	
Technology	Analytics can be used to integrate the system and infrastructure performance data with customer usage behaviour	Lwakatare et al. (2015)
	Not just gather this data but also run analytics on it	Sharma (2017b)
	Basic services such as dashboards	Senapathi et al. (2018)
	Use a Realtime User Monitoring tool	Erich et al. (2014b)
	APIs or services	Humble and Farley (2011)
	The application should use to notify the operations team of its state	
Culture	Collaboration between developers and operations so that the systems are designed to expose relevant information	Lwakatare et al. (2015)

Table 16 Continuous testing practices

	Continuous testing	
	Practice	*Author*
People	Understand test automation functions	Wiesche (2018)
	Automate tests	
	Understand functionalities for test management	
Process	Script-based testing early and throughout the software delivery process	Nielsen et al. (2017)
	Shorten later testing cycles	
	Ensure continuous feedback on quality	
	Testing earlier and continuously across the life cycle	Sharma and Coyne (2015)
	High test coverage of high-risk areas	Marijan et al. (2018)
	Integrate testing activities as closely as possible with coding	Fitzgerald and Stol (2014)

Table 16 Continuous testing practices (continued)

	Continuous testing	
	Practice	*Author*
Technology	Virtualisation to simulate the production environments	Silva et al. (2018)
	Test case generation	Vassallo et al. (2017)
Culture	Both IT Development and IT Operations should carry out quality assurance and be responsible for test automation	Nielsen et al. (2017)
	Each developer should take personal responsibility for their code and write the test cases	De Bayser et al. (2015)
	Testing on real users at scale	Feitelson et al. (2013)
	Driving development with tests	Vassallo et al. (2017)
	TDD is a development practice that starts with writing tests before you write any code	Perera et al. (2017b)
	BDD encourages working with the business stakeholder to describe the desired business functionality of the application	
	ATDD builds on TDD and BDD, and it is involved in finding scenarios from the end user perspective	
	Testing/quality team is connected with Development team early in the development cycle to create the required test cases	Mohamed (2015)

Table 17 Infrastructure as a code practices

	Infrastructure as code	
	Practice	*Author*
People	–	–
Process	Versioning environments	Mohamed (2016)
Technology	Entire infrastructure in a common language	Luz et al. (2018)
	Automate server	Sharma and Coyne (2015)
	Generic tools	
	Application or middleware-centric tools	
	Environment and deployment tools	
Culture	Everyone knows how the execution environment of an application is provided and managed	Luz et al. (2018)

Table 18 Feedback loops practices

	Feedback loops between Dev and Ops	
	Practice	*Author*
People	Feedback ability, in both directions – so, to give feedback but also to accept it	Wiesche (2018)
Process	Shorten later testing cycles to ensure continuous feedback	Nielsen et al. (2017)
	Ensure continuous feedback provided through the monitoring process and the users	

Table 18 Feedback loops practices (continued)

	Feedback loops between Dev and Ops	
	Practice	*Author*
Process	The frequency of integration is also important in that it should be regular enough to ensure quick feedback to developers	Fitzgerald and Stol (2014)
	Mechanisms to involve users in the development process and collect user feedback from deliveries as early as possible	Rodríguez et al. (2018)
	Techniques need to be nonintrusive so that users are not stressed with continuous feedback requests	
	Short feedback loops	
	Feedback loops strategy	Science (2016)
	The measurement results should be provided to not only the operation people, but also the development people	Rong et al. (2016b)
	Any change, of whatever kind, needs to trigger the feedback process	Humble and Farley (2011)
	The feedback must be delivered as soon as possible	
	The delivery team must receive feedback and then act on it	
Technology	–	–
Culture	Share feedback freely without blame	Perera et al. (2017a)
	High focus on requirements	Nielsen et al. (2017)
	Management through close relationship with the users to determine their needs and quickly react on their feedback	
	Keeping a constant feedback about the current state of the system	Rodríguez et al. (2018)

Table 19 CDMM

		Level 2	*Level 3*	*Level 4*	*Level 5*
Continuous deployment	People	–	–	–	–
	Process	CD9 Label a repository's assets	CD2 Track which version is deployed CD3 Manage the configurations of the environments of all the stages CD4 Manage the software components that get deployed CD5 Manage the middleware components and middleware configurations that need to be updated CD6 Manage the database components that need to be changed	CD1 Orchestrated deployments CD16 Deployments should include the automated provisioning of all environments	CD1 Orchestrated deployments CD7 Manage the configuration changes to the environments to which these components are to be deployed CD8 Release working software any time, any place CD15 Multiple deployments to production

Table 19 CDMM (continued)

		Level 2	Level 3	Level 4	Level 5
Continuous deployment	Process		CD10 Produce a clean environment CD11 Label each build CD12 Create build feedback Reports CD14 Deploy a new release whenever one is needed CD17 Automated deployment CD18 Continuous deployment		
	Technology	–	CD19 Development and production share a homogenous infrastructure CD20 Configuration management tools	CD21 Automated deployment of software to different environments	–
	Culture	–	CD22 Team must provide overall visibility into your application release activities and timing to all major stakeholders CD25 Unite the two teams that worked independently to work at tighter integration CD26 Both development and operations personnel should share the same knowledge management resources CD27 Testers and operations personnel would be able to self-service deployments of the required version of the system to their environments on demand CD28 Early and frequent involvement of operations staff in the planning stages of major new releases	CD24 Team must be able to speed lead times and make more frequent application deployments at the pace demanded by the business	CD23 Teams must be able to provide self-service, on-demand provisioning and management of cloud environments and infrastructure resources

Table 20 CI MM

		Level 2	*Level 3*	*Level 4*	*Level 5*
Continuous integration	People	–	–	–	–
	Process	CI8 Make it easy for anyone to get the latest executable	CI1 Automation of tasks CI2 Provision of virtualised hardware resources via scripts (instead of doing manual configuration work) CI3 Developers should make use of continuous integration, that is branch-out and merge-back their work with the software mainline (the trunk) several times a day, in order to discover integration risks as early as possible CI5 Continuous integration cycles to include also software release CI6 Enable rapid automated regression testing of code changes	CI4 Continuous feedback loop CI7 Test in a clone of the production environment	–
	Technology	CI11 Version control CI15 Automate the build	CI12 An automated build CI13 Use build servers	CI9 Use of cloud services	CI10 Tools interoperability for unifying force across diverse teams, skills, technology languages, and methodologies CI14 Maintain a single-source repository
	Culture	CI16 Collaboration between teams CI19 Agreement of the Team	CI17 Development and QA teams perform unit and integration testing	CI18 Operations participates in integration and load testing to assess operational readiness CI20 Make sure everyone can see what is happening	–

Table 21 Continuous monitoring MM

		Level 2	Level 3	Level 4	Level 5
Continuous monitoring	People	CM1 Analysis skills	–	–	–
	Process		CM4 Application monitoring CM5 System monitoring CM8 Delivery pipeline metrics CM11 Storing the data so it can easily be retrieved for analysis CM13 Setting up notifications so that people can find out about the events they care about	CM2 Define some useful measurement metrics CM6 Application user behaviour CM7 User sentiment CM9 Systems are monitored after deployment CM10 Instrumenting your applications and your infrastructure so you can collect the data you need CM12 Creating dashboards which aggregate the data and present it in a format suitable for operations and for the business	CM3 Ensure continuous feedback provided through the monitoring process and the users
	Technology		CM16 Basic services such as dashboards CM17 Use a realtime user monitoring tool CM18 APIs or services	CM19 The application should use to notify the operations team of its state	CM14 Analytics can be used to integrate the system and infrastructure performance data with customer usage behaviour CM15 Not just gather this data but also run analytics on it
	Culture		CM20 Collaboration between developers and operations so that the systems are designed to expose relevant information		

Table 22 Continuous testing MM

		Level 2	*Level 3*	*Level 4*	*Level 5*
Continuous monitoring	People	–	CT1 Understand test automation functions CT2 Automate tests CT3 Understand functionalities for test management	–	–
	Process	–	CT4 Script-based testing early and throughout the software delivery process CT6 Ensure continuous feedback on quality	CT5 Shorten later testing cycles CT7 Testing earlier and continuously across the life cycle CT8 High test coverage of high-risk areas CT9 integrate testing activities as closely as possible with coding	–
	Technology	–	CT10 Virtualisation to simulate the production environments CT11 test case generation	–	–
	Culture	CT15 driving development with tests CT16 TDD is a development practice that starts with writing tests before you write any code CT19 Testing/quality team is connected with Development team early in the development cycle to create the required test cases	CT13 Each developer should take personal responsibility for their code and write the test cases CT17 BDD encourages working with the business stakeholder to describe the desired business functionality of the application CT18 ATDD builds on TDD and BDD, and it is involved in finding scenarios from the end user perspective	CT12 Both IT development and IT Operations should carry out quality assurance and be responsible for test automation	CT14 Testing on real users at scale

Table 23 Infrastructure as code MM

		Level 2	*Level 3*	*Level 4*	*Level 5*
Continuous monitoring	People	–	–	–	–
	Process	–	IAC1 Versioning environments	–	–
	Technology	–	IAC2 Entire infrastructure in a common language IAC3 Automate server IAC4 Generic tools IAC5 Application or middleware-centric tools IAC6 Environment and deployment tools	–	–
	Culture	–	–	–	IAC7 Everyone knows how the execution environment of an application is provided and managed

Table 24 Feedback loops MM

		Level 2	*Level 3*	*Level 4*	*Level 5*
Continuous monitoring	People	–	–	FL1 Feedback ability, in both directions – so, to give feedback but also to accept it	–
	Process	–	FL2 Shorten later testing cycles to ensure continuous feedback FL4 The frequency of integration is also important in that it should be regular enough to ensure quick feedback to developers FL7 Short feedback loops FL11 The delivery team must receive feedback and then act on it	FL3 Ensure continuous feedback provided through the monitoring process and the users FL5 Mechanisms to involve users in the development process and collect user feedback from deliveries as early as possible FL8 Feedback loops strategy the measurement results should be provided to not only the operation people, but also the development people FL10 The feedback must be delivered as soon as possible	FL6 Techniques need to be nonintrusive so that users are not stressed with continuous feedback requests FL9 Any change, of whatever kind, needs to trigger the feedback process

Table 24 Feedback loops MM (continued)

		Level 2	*Level 3*	*Level 4*	*Level 5*
Continuous monitoring	Technology	–	–	–	–
	Culture	–	–	FL13 High focus on requirements	FL12 Share feedback freely without blame
					FL14 Management through close relationship with the users to determine their needs and quickly react on their feedback
					FL15 Keeping a constant feedback about the current state of the system

APPENDIX C – AGILE INFORMATION TECHNOLOGY SERVICE MANAGEMENT WITH DEVOPS: AN INCIDENT MANAGEMENT CASE STUDY

Int. J. Agile Systems and Management, Vol. 13, No. 4, 2020 339

Agile information technology service management with DevOps: an incident management case study

João Faustino*, Rúben Pereira and Bráulio Alturas

Department of Information Science and Technology,
Instituto Universitário de Lisboa (ISCTE-IUL),
Avenida das Forças Armadas,
Lisboa, 1649-026, Portugal
Email: jpcfo11@iscte-iul.pt
Email: ruben.filipe.pereira@iscte-iul.pt
Email: braulio.alturas@iscte-iul.pt
*Corresponding author

Miguel Mira da Silva

Instituto Superior Técnico,
University of Lisbon,
Av. Rovisco Pais 1,
Lisboa, 1049-001, Portugal
Email: mms@tecnico.ulisboa.pt

Abstract: This research aims to investigate how DevOps culture can be applied in the incident management process. The authors believe, based on experience as practitioners, that agile software development methodologies are fair enough to be used on Incident Management process, to quickly restore the business interruption. An application management team which solves incidents and applies DevOps practices was studied. Three data collection methods were used: interviews, document analysis and observation. This research provides novel findings supported by metrics and real experience implementing DevOps practices in incident management process. The novelty of the findings brings advantages for academics, and due to the exploratory nature of this research, it extends the body of knowledge. It also provides contributions for practitioners, by showing how these practices can be applied and the result of the implementation of these practices. Directions of future work are also presented.

Keywords: information technology; case study; interviews; document analysis; observation; DevOps; incident management; application management; agile methodologies; service management.

Reference to this paper should be made as follows: Faustino, J., Pereira, R., Alturas, B. and Mira da Silva, M. (2020) 'Agile information technology service management with DevOps: an incident management case study', *Int. J. Agile Systems and Management*, Vol. 13, No. 4, pp.339–389.

Biographical notes: João Faustino is an IT Consultant. He is also a PhD student in Information Science and Technology at ISCTE-IUL where he also graduated as MSc in Computer Engineering. Most of his career, he has worked

as a consultant on the Financial Services industry, on application management teams for both corrective and enhancement maintenance of Core Systems for Insurance companies. Besides his professional work, which is mainly related with Oracle Technologies, he is very enthusiastic about open source technologies, data science, mobile development and process optimisation. Most of his research is about IT service management and DevOps.

Rúben Pereira is an Assistant Professor at ISCTE. He has a PhD in information systems at Instituto Superior Técnico where he also graduated as a Master's in Computer Engineering and Computer Science. He has been a consultant in several industries, such as services, banking, telecommunications, and e-commerce, among others. He is the author of several scientific papers in the area of Information Technology Services Management and Information Technology Governance, covering the most known IT Frameworks like ITIL and COBIT. His areas of scientific interest extend to information technology risk management, business process management, continuous improvement and innovation, process optimisation, digital transformation, DEVOPS, robotic process automation, among others.

Bráulio Alturas is an Associate Professor from Instituto Universitário de Lisboa (ISCTE-IUL) and is currently researcher of the Information Systems Group of ISTAR-IUL (University Institute of Lisbon), Lisbon, Portugal. He holds a PhD in Management with specialisation in Marketing, MSc in Management Information Systems, and BSc in Business Organisation and Management, all from ISCTE-IUL. His research interests and publications are in acceptance and use of technology, digital marketing, social media, e-commerce, information management, information systems, direct selling and consumer behaviour.

Miguel Mira da Silva is currently Associate Professor of Information Systems in the University of Lisbon, leader of the research group 'Digital Transformation' at INOV, and coordinator of the MISE online MSc. He has a PhD in Computing Science from the University of Glasgow and an MSc in Management from the London Business School. Miguel created five companies, published four books and 200 research papers, managed dozens of research and consulting projects, graduated nine PhDs and 150 MSc students, and created a MOOC about digital transformation. His current interests include digital transformation, IT governance, and online learning.

1 Introduction

Organisations, since a few decades, have been changing their business management due to the constant competitive behaviour and new technologies, thus, organisations have begun to consider their core business proposition to provide services, changing the world economy to a service-based economy (Badinelli et al., 2012). Services are considered interactive processes between customers and service providers where the customer benefits from the expertise of the service provider (Stokburger-Sauer et al., 2016). Thus, to measure the efficiency of services, the discipline of service management was created so organisations could understand how value can be created (Verma, 2000; Stokburger-Sauer et al., 2016).

To stay competitive, organisations, need to respond to the dynamic changes that markets require, to offer a better experience to their customers and to innovate with new services and products (Soni, 2016). Part of these dynamic changes are grounded on technologic advances. Therefore, organisations have been realising that the information technology (IT) is fundamental to their success (Park et al., 2006). IT changes how organisations work, changing business processes, internal and external communication and most importantly, affects how organisations deliver services to customers (Alsolamy et al., 2014).

Since organisations have started to see the importance of IT, they have begun to implement complex and dynamic IT systems to support their business processes (Jamous et al., 2017). Given the increasing dependence on IT and to support these business processes, organisations began using the term service (Cannon and Wheeldon, 2007). Thus, the concept of IT service started to grow.

Due to both the expansion of IT services and changes in the world economy to a service-based economy, organisations have started to adopt IT Service Management (ITSM) (Pereira and da Silva, 2012). The ITSM is becoming an integral part of organisations (Mora et al., 2015), since it provides a set of activities to align, design, deliver, manage and improve how IT is used within an organisation (Wang et al., 2010).

Despite the existence of some IT frameworks to assist organisations in ITSM implementation, some organisations still struggle to understand the concept behind ITSM, how its processes are implemented (Remfert, 2017) and how to identify which process should be implemented first (Jamous et al., 2017). However, one of the most implemented ITSM processes is the Incident Management (IM) process (Gacenga et al., 2011; Jäntti, 2011; Aguiar et al., 2018).

The Information Technology Infrastructure Library (ITIL) for example, which is one of the most used frameworks to implement ITSM (Pereira and da Silva, 2011; Aguiar et al., 2018) includes the IM process as a core service operations process (Cao and Zhang, 2016). IT operations execute daily tasks to ensure the normal business operation and manage the IT infrastructure (Cannon and Wheeldon, 2007; Cao and Zhang, 2016).

The IM process focuses on restoring a service downtime as quickly as possible (Tan et al., 2010) to avoid any impact to the business users in their daily activities (Ghrab et al., 2016). Due to the competitiveness of the market, organisations want to provide a service of excellence to their customers, and one way to do it is to minimise the negative impact of service interruption on businesses by implementing IM correctly to restore services promptly (Yun et al., 2017). The agile software development methodologies emphasise the rapid software development using frequent and small iterations of development, which is the ideal to restore downtime services to avoid a larger impact for the business (Beddle et al., 2001; Gotel and Leip, 2007; Uikey and Suman, 2016).

To deal with the constant change of requirements, agile methodologies, like SCRUM, Extreme Programming and DevOps were created (Highsmith and Cockburn, 2001; Gotel and Leip, 2007; Laukkarinen et al., 2017). Software developers started to realise that on the traditional software development methodologies it was very difficult to incorporate the clients feedback as the development lifecycle is progressed (Barksdale and McCrickard, 2012).

These agile methodologies, are grounded by the effective communication, collaboration and coordination (Hannola et al., 2013; Suomalainen et al., 2015; Suomalainen and Xu, 2016), not only inside the software development teams but also between the software development teams with the business (Liu et al., 2015; Soni, 2016).

The IM process resolution also has a lot of manual tasks and is time consuming (Gupta et al., 2008) while one of the premises of DevOps is to automate manual processes, like testing, to deliver new functionalities and bug fixes (Sharma and Coyne, 2014).

DevOps is a software development culture that tries to eliminate the lack of collaboration between development and operations teams (Mahanta et al., 2016; Chen, 2019) by teaming them up to promote cooperation, collaboration and communication (Guerriero et al., 2016; Silva et al., 2018).

Building on the previous statements and context there is a gap on the IM process that may be solved by using the DevOps culture. So, this research has the following objective: Explore the relationship between DevOps and the IM process and understand the impacts that DevOps adoption may cause on the IM process.

This research aims to contribute to better a understanding of the impacts that the agile philosophy DevOps may cause on both practitioners and business users, which currently is seen as an unclear area (Kamuto and Langerman, 2017; Prates et al., 2019). More specifically, this research contributes by exploring the impact of DevOps adoption in the IM process.

The remaining article is organised as follows: Section 2 describes the main concepts that frame this research and also examines DevOps case studies (CSs) to confirm that this is an exploratory research; then, Section 3 describes the advisable CS research methodology; subsequently, in Section 4, the authors list all the data that will be needed to conduct the CS; next, Section 5 explains how the authors transform the collected data for analysis; lastly, Section 6 details the main conclusions of the findings discovered during the analysis phase as well as explain the main contributions for academics and professionals.

2 Literature review

To add more scientific rigor to our research, the authors decided to follow the concept-centric approach proposed by Webster and Watson (2002). The literature was collected when analysing several databases such as IEEExplore, ACM, Research Gate and the search engine Google Scholar. Also, this research was made between September 2017 and January 2018, but the author has maintained currency to date.

2.1 Incident management process

Mentioned as one of the key pieces to support any IT system (Pereira and da Silva, 2012) and one of the most implemented ITIL processes (Limanto et al., 2017), the IM process aims to solve incidents and restore services (Wang et al., 2017) while mitigate the impact on business activities and avoiding economic losses (Lou et al., 2013).

Since the organisations' main goal is to generate profit, economic losses must be avoided or mitigated as soon as possible. Moreover, the IM process is not only about solving the issues quickly, but also anticipating and preventing future or repeated incidents from happening (Bezerra et al., 2014; Kikuchi, 2015; Saarelainen and Jäntti, 2016).

An incident can be defined by the interruption of the organisation activity causing negative impacts, like the customers' confidence and financial and productivity loss (Latrache et al., 2015).

In ITIL framework, the Application (AM) is one of the specialist groups, which sometimes also plays the role of application development: "In many cases the same team will be responsible for Application Developments as well as support" (Cannon and Wheeldon, 2007). Such statement indicates that the team that should operate applications and solve incidents can also develop new features for the application, bridging the gap between the IT operations and development that DevOps culture tries to solve.

2.2 DevOps

When developing products and services, there exists a lack of communication between the development and operation teams that are responsible for delivering these products (Rong et al., 2017). The main gap between development and operations teams is the attitude toward changes: the development side embrace the changes as something they need to achieve, but on the other side, operations try to avoid the changes to not compromise the system stability (Hussain, 2015). Besides the fear of change, there exists other problems: risky deployments; the blame-game, where the operations find the production issues and blame the developers for bad developments; and isolation, where developments from programmers, testers and quality assurance occur in silos, while the operations silo includes database administrators, systems administrators and operators (Wahaballa et al., 2015; Katal et al., 2019). To face these problems between the development and operations teams, a new agile culture appeared, DevOps.

The Dev is from Developers and Ops from operations, promoting the collaboration between this two teams sharing tasks and responsibilities while being empowered with full accountability of their service and its underlying technology stack, from development, to deployment and to support (Perera et al., 2017a). This research follows the definition of Dyck et al. (2015) which define DevOps as: "*DevOps is a mindset, encouraging cross-functional collaboration between teams – especially development and IT operations – within a software development organisation, in order to operate resilient systems and accelerate the delivery of changes*". Besides collaboration. DevOps has another main concept, which is the automation to configure and manage deployment environments (Riungu-Kalliosaari et al., 2016; Gupta et al., 2019).

Soni (2016) also says that the philosophy behind the DevOps concept is "the faster you fail, the faster you recover" (Soni, 2016). This means that the faster the deployment of a solution including customers' feedback, the faster developers will be able to make the necessary improvements to enable a better customer experience.

Based on the fast feedback from the application users it would be possible to identify possible incidents sooner. Automation is encouraged by DevOps. Joining the automation with the faster feedback from the application users, the resolution of the incidents can be deployed in production faster, avoiding economic losses for the organisations and contributing to the stability of the application.

2.3 DevOps practices, benefits and challenges

This section lists the main DevOps practices, benefits and challenges found in the literature. More information about the authors that identified challenges and benefits can be seen in Table 1. The authors have given an ID for each benefit (Bx) and challenge (Cx). For a better understanding is not our goal to ring consensus about DevOps practices. A recent study (Jabbari et al., 2016) synthesised DevOps' practices that practitioners have been applying so far. Other studies referring to DevOps practices can be found amongst the literature (Sharma and Coyne, 2014; Soni, 2016; Punjabi and Bajaj, 2017; Stoneham et al., 2017) but not so complete. Therefore, Jabbari' list is used to guide this research about the practices that DevOps includes. These practices can be found in Annex 1.

Table 1 DevOps benefits and challenges

	ID	*Concepts*	*References*	*No. of references*
Benefits	B1	Improved code quality, quality assurance and reliability	Erich et al. (2014), Shahin (2015), Mahanta et al. (2016), Riungu-Kalliosaari et al. (2016), Soni (2016), Laukkarinen et al. (2017), Palihawadana et al., 2017) and Perera et al. (2017b)	8
	B2	Better communication	Erich et al. (2014), Karapantelakis et al. (2016), Riungu-Kalliosaari et al. (2016), Soni (2016), Laukkarinen et al. (2017) and Perera et al. (2017b)	6
	B3	Application stability	Roche (2013), Gottesheim (2015), Guerriero et al. (2016), Mahanta et al. (2016) and Soni (2016)	5
	B4	Visibility to the customer of the implemented features	Roche (2013), Gottesheim (2015), Riungu-Kalliosaari et al. (2016) and Soni (2016)	4
	B5	Continuous experimentation	Erich et al. (2014),Mahanta et al. (2016), Riungu-Kalliosaari et al. (2016) and Soni (2016)	4
	B6	Maximising competences	Shahin (2015) and Riungu-Kalliosaari et al. (2016)	2
	B7	Testing with real customers	Riungu-Kalliosaari et al. (2016)	1
Challenges	C1	Insufficient communication	Riungu-Kalliosaari et al. (2016), Hussain et al. (2017) and Perera et al. (2017b)	3
	C2	Industry constraints	Sharma and Coyne (2014) Riungu-Kalliosaari et al. (2016) and Laukkarinen et al. (2017)	3
	C3	Deep-seated company culture	Shahin (2015) and Riungu-Kalliosaari et al. (2016)	2
	C4	DevOps is unclear but also Evolving	Riungu-Kalliosaari et al. (2016)	1
	C5	Deployment automation for several technologies	Mahanta et al. (2016)	1

2.4 DevOps outcomes vs. benefits

This section aims to elicit which benefits one could expect from DevOps' practices implementation. To do that, the authors searched in the literature for DevOps CSs (Table 2) and synthesised the outcomes and benefits reported by each one. At the end, one can confirm that none of the CSs found was applied to the IM context which proves the novelty of this research.

Table 2 Extant DevOps case studies in the literature in application development

ID	*DevOps practices implemented*	*Reference (author, year)*	*Industry*
CS.1	Automated tests; automated monitor; feedback loops; process standardisation	Roche (2013)	N/A
CS.2	Continuous integration; automated monitor; deployment and test automation	Soni (2016)	Financial industry – insurance
CS.3	Continuous integration	Laukkarinen et al. (2017)	Health
CS.4	Continuous improvement; test automation; shift-left; infrastructure as code	Sharma and Coyne (2014)	Software development
CS.5	Test automation; deployment automation; continuous integration;	Punjabi and Bajaj (2017)	Software development
CS.6	Continuous monitor; deployment automation	Karapantelakis et al. (2016)	Software development
CS.7	Continuous integration; test automation; deployment automation	Stoneham et al. (2017)	Retail
CS.8	Continuous integration; feedback loops	Stoneham et al. (2017)	Government agencies
CS.9	Continuous integration; infrastructure as code	Stoneham et al. (2017)	N/A (Regulated)
CS.10	Stakeholder participation; continuous integration; automated monitors; continuous planning	Stoneham et al. (2017)	Large consume
CS.11	Continuous integration; test automation; feedback loops; stakeholder participation	Stoneham et al. (2017)	N/A
CS.12	Continuous integration; automated tests	Croker and Hering (2016)	N/A
CS.13	Continuous integration; deployment automation; test automation; automated monitor	Shivakumar (2017)	N/A

For better comprehension, the authors grouped the outcomes with a generic description of the outcome. An outcome can be written in a different way depending on its context but mean the same, so the authors have grouped these outcomes by what they have understood from the meaning of the outcome.

Information is synthesised in Tables 3 and 4 to identify which outcomes lead to which benefit and how many times each benefit and outcome is identified by the CSs. It is interesting to note that all CSs reported benefits but only two reported challenges and that the benefits "Breaking Down the Silos" and "Short Release Cycles" are reported by most of the CSs.

Analysing Table 3, the following conclusions can be drawn. The improvements in code quality and reliability are present in DevOps in nine CSs (69%), respectively. The objective of DevOps is to deliver fast of high quality, and our literature review confirms it in practice. Better communication was only matched with "Breaking down the Silos"; however, this was found in seven of 13 CSs, which also represents more than 50% of the sample of CSs. Application stability was also found nine times in the CSs. This shows that DevOps culture works. The final objective of each software project is to deliver with quality, but it is also important to deliver stable software. Having developers and operators working together monitoring the application stability brings great results, building confidence between the IT and business teams.

Table 3 DevOps outcomes and benefits

CS ID	*Outcome*	*Benefit/Challenge ID (Section 2.3)*
CS.1	Breaking down the silos	B2; B6
	Short release cycles	B3; B4
CS.2	Short release cycles	B1; B3; B4
	Application availability	B3
CS.3	CI brings quality	B1
	Industry constraints	C2
CS.4	Short release cycles	B1; B3; B5
	Industry constraints	C2
	Breaking down the silos	B2
CS.5	Short release cycles	B4; B5
CS.6	Short release cycles	B3
	Application availability	B3
CS.7	Breaking down the silos	B2
	Short release cycles	B1; B3
CS.8	Short release cycles	B1
	Breaking down the silos	B2; B6
	High scalability	B3
CS.9	Application availability	B1; B3
CS.10	Breaking down the silos	B2
	Short release cycles	B1
CS.11	Breaking down the silos	B2
	Short release cycles	B5
CS.12	Short release cycles	B5
	Improved security with resiliency	B1
CS.13	Short release cycles	B1; B5

Table 4 Outcome vs. benefit

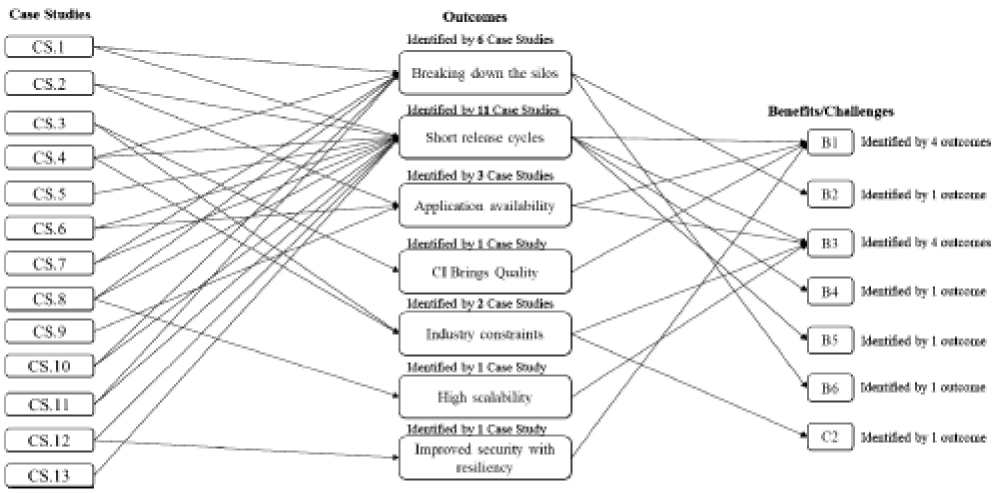
	B1	*B2*	*B3*	*B4*	*B5*	*B6*	*C2*	*Total*
Breaking down the silos		6				2		9
Short release cycles	6		5	2	5			**18**
Application availability	1		2					3
CI brings quality	1							1
Industry constraints			1				2	3
Improved security with resiliency	1							1
High scalability			1					1
Total	**9**	6	**9**	2	5	2	2	

Analysing Table 3, "Short Release Cycles" appears to be the most beneficial practice. Having short release cycles bring a lot of benefits since developers and operators deal with smaller chunks of code, it is easier to maintain, test and deploy. This also allows the business to see their application grow step by step and be able to provide feedback to include in a possible next release, creating a user engagement between the application and the business users.

As far as the authors could find, the application of DevOps culture on the IM process remains an unstudied field. Therefore, grounded on the motivation presented in the Introduction, this research intends to provide more insights about the possible application of DevOps culture on the IM process. At the end of this research, the authors propose to answer the research questions (RQ) presented in Table 5. Since no researches exist about the application of DevOps on the IM process, it is possible to conclude that this research is exploratory.

As shown in Table 2, some scientific studies exist about DevOps application but none of these researches aimed to study or elicit any conclusions/implications regarding the IM process. In Figure 1 there is possible to analyse the relationship between Tables 1–4.

Figure 1 Relationship between case studies, outcomes and benefits and challenges



3 Research methodology

Since the research in the domain of DevOps application in the IM process is in its very early stages, as stated in the previous section, the nature of this research is exploratory rather than hypothesis testing. Exploratory research is meant to start a study on a determined phenomenon observed, where there are no prior (or few) works (Zaidah, 2007). Zaidah (2007, p.1), argues that "a case study enables the researcher to examine the data within a specific context".

Yin (2009) argues that questions like 'what' are exploratory since the purpose is to develop propositions for further inquiry, which fits the questions that were previously stated. A CS also has 'how' and 'why' questions, where the researcher does not have control over the variables, which suits this research (Perry et al., 2004).

Moreover, a CS is built around a question (Thomas, 2016), which in this case is, "How do DevOps affect professionals working on the Incident Management Process"? For a better synthesis of this research, the authors detailed the main question in several research questions that can be found in Table 5. Following Thomas (2016) theory, this CS is classified as a local knowledge case since the study focus on a team that applies DevOps practices and use the IM process. More information regarding this team can be found in Section 3.2.

Perry et al. (2004) also argues that CS is a powerful method for exploratory researches because they try to understand and explain the phenomenon or construct theory.

Since this research focuses on the analysis of a single team, which will be a single unit of analysis as described by Yin (2009), the authors argue that this research follows a single CS approach.

Table 5 Research questions

Research question ID	*Description*	*Article section*
RQ1	What DevOps practices can be used in each phase of IM?	5.1.1; 5.2; 5.3;
RQ1.1	How can these practices be applied?	5.1.1; 5.2; 5.3;
RQ1.2	Why should these practices be applied?	5.1.1; 5.2; 5.3;
RQ2	What are the benefits of using DevOps practices in IM?	5.1.2
RQ3	What are the challenges of using DevOps practices in IM?	5.1.3
RQ4	How do DevOps improve the resolution of incidents?	5.1.4

For a better understanding on how this research maps with Thomas' framework, to build a CS, Figure 2 describes the different classifications of our research according to Thomas' framework and guidelines.

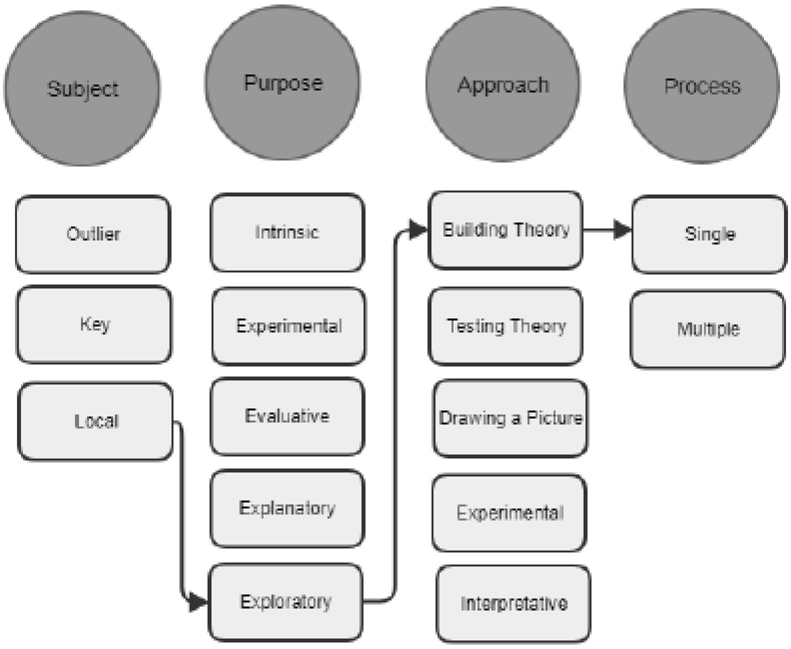
Thomas (2016) also says that time is important and defines three timeframes: Retrospective (where the studied phenomenon happened in the past); Snapshot (where the study happens on a timeframe); and Diachronic (where the study shows a change over time). Since this CS is based on the experience (past) of a single team, this CS is considered as retrospective.

According to Thomas (2016), CSs are about seeing different behaviours from different angles, so many authors advise the triangulation of several data collection methods (Tellis, 1997; Modell, 2005). Therefore, the authors will use triangulation between the following research methods: semi-structured interviews, document analysis and observation recommended by Thomas (2016).

To sum up, performed CS will follow a single, retrospective approach, and the triangulation of methods (semi-structured interviews, data analysis and observation) will be used to enrich the research findings.

In the following sub-sections, the authors explain how this research maps the CS stages proposed by Telis (1997) and Yin (2009): Design the CS Protocol; Conduct the CS; Analyse CS Evidences; and Develop Conclusions. Subsequently, to demonstrate the validity of our CS, the authors use the CS validity test proposed by Yin (2009) to demonstrate our research rigor and relevance.

Figure 2 Path for the CS, adapted from (Thomas, 2016)



3.1 Designing the case study protocol

On this stage, it is required to determine the necessary skills to conduct the CS and develop a protocol where a reading about the topic should be done, to create some draft questions. Tellis (1997) uses Yin as an example arguing that researchers should be good listeners and have a good interpretation of the responses.

In this research, the most required skill is to have a good knowledge of software engineering (SE) and IM process; thus, the authors can interpret the results and know what to ask the target audience.

For the CS protocol, the authors performed a literature review about IM process and DevOps to reach a deep understanding about these domains and how they have been applied so far. To support the interviews, a questionnaire was built to guide the authors.

3.2 Conducting the case study

In this stage, the authors performed interviews to collect practitioners' opinions and experience about the implementation of DevOps practices and their impact for the project and for themselves as IT professionals.

Since our RQs aim to explore what or how DevOps practices influence the work of professionals in the IM process, the authors used semi-structured interviews. This type of interview is used when one needs to gather more detailed information by giving the interviewees the liberty to express their opinions (Miles and Gilbert, 2005). To accomplish the triangulation goal, other techniques for data collection were also used, such as data extraction from performance reports and direct observation.

The interviews were performed with members and ex-members of a maintenance team of a big corporation, which currently employs around of 3000 employees and has offices in six different locations, five locations in Portugal and one in Angola. This team works on an AM workstream, analysing and developing the solutions for production incidents. All the team members work for the same corporation, and they work on the same project as consultants. The client is a Danish organisation that operates in the financial sector. This team also uses several software's in their daily tasks: HP Service Management (HPSM) to manage incidents and changes; Microsoft Team Foundation Server (TFS) as a code repository and to perform CI; Jenkins for building changes and packages of the code checked in TFS; SonarQube to validate the code quality; Artifactory to save the packages that are built on Jenkins; and CA LISA to perform the installation of packages. Additionally, this team works at three different sites at the same time (two offices in Portugal and one office in Denmark), so communication is very important for their success. This team supports the business users in their daily work by helping them when they face some errors with the application, sometimes proving workarounds and making some extractions for business reports. The team also supports the development teams. The dev teams usually present the solutions of the new features to members of the AM team, so weaknesses can be identified before going to production, to improve the quality of the delivery. Also, the AM team is required to help to define the requirements and performance metrics. Generically, observation is used to analyse the 'before and after' of the behaviour of a certain phenomenon after some change (Yin, 2009). However, since the practices were already implemented by the team, it is not possible to verify this change of behaviour in the first place. Thus, the authors will use the observation to validate the findings that were gathered during the interviews.

Observation can be seen as structured or unstructured (Thomas, 2016). Structured observation occurs when the researcher systematically looks for particular kinds of behaviours, while unstructured observation happens when the researcher informally observes important details of what is happening (Thomas, 2016). Unstructured observation may also be called participant observation, where the researcher is also a participant. The kind of observation that should be used in this research is unstructured observation, since the observation will only be used to validate some of the results of the interviews, such as taking notes. We also analysed some performance reports on team performance discrepancies that this team produced weekly to present to business users.

3.3 Analysing the case study evidence

The authors analysed the data that was collected from the semi-structured interviews and from the reports. Furthermore, all the data from performance documentation provided from the team under study was also analysed as well as from direct observation.

3.4 Developing conclusions

This stage must describe all the main findings regarding the data previously collected and analysed. The authors intend to condense all the data collected from practitioners, documentation and observation.

3.5 Case study validity

Yin (2009) proposes four tests to validate CS reliability. These tests are: Construct Validity; Internal Validity; External Validity; and Reliability. All these tests were applied to this case study except for Internal Validity, since Yin (2009) says this test should not be applied to exploratory research. For Construct Validity test, multiple data sources were used on this case study, such as, semi structured interviews and document analysis. Regarding External Validity test the existing literature was reviewed in section two, where was not found any reference about the DevOps application on the IM process, showing the novelty of this research. For the last test, Reliability, there was created a path on how the researchers have built this case study to show to future research how they can proceed with the investigation.

4 Case study protocol and conduct

Since our RQs aim to explore what or how DevOps practices influence the work of professionals in the IM process, the authors used semi-structured interviews. This type of interview is used when one needs to gather more detailed information by giving the interviewees the liberty to express their opinions (Miles and Gilbert, 2005). To accomplish the triangulation goal, other techniques for data collection were also used, such as data extraction from performance reports and direct observation.

Table 6 Interviewees details

Interviewee	*Position*	*Experience*			*Projects in IM*
		Years	*IT*	*IM*	
A	Developer	3.5	3.5	3.5	2
B	Developer	3.5	3.5	3.5	2
C	Senior Developer	4	4	4	3
D	Developer	2.5	2.5	2.5	1
E	Team Leader	7	7	7	3
F	Team Leader	10	10	10	3
G	Developer	3	3	1	1
H	Manager	13	13	10	3
I	Developer	1.5	1.5	1.5	1.5
J	Team Leader	6	6	5	4
Average		*5.4*	*5.4*	*5.4*	*2.35*

At the end of the CS, the authors were able to interview 10 members of the studied team. The details about each interviewee are listed in Table 6. These team members were chosen to be interviewed since they are the ones that put the DevOps practices in place inside the team. There is possible to find the questions used in the questionnaire in Appendix E. All the interviews were performed by one of the authors. The interviews were recorded, where the authors have collected an agreement to authorise the usage of the data collected during the interview. The average time of the interviews was about of 63 minutes and they were performed between March 2018 and August 2018. The interviews were all performed by the same interviewer. There was no software involved on the interview data analysis. All the data analysis was performed by the authors.

The average experience of the team members is about five years. Moreover, most of the interviewees have been involved in more than one IM project, allowing us to retrieve a range of ideas on best practices.

5 Case study analysis

5.1 Semi-structured interviews data analysis

In the questionnaire, the authors asked some basic questions about DevOps, like what practices the respondents are familiar with and what they apply or had applied on previous/current projects. When enquiring about the practices already applied, the authors made a scale from 1 to 3, where 1 meant did not apply, 2 meant partially applied and 3 meant fully applied. One should assume partial implementation as a practice that is incomplete or could not be implemented in the entire context it was expected to work. For example, for deployment automation, a developer cannot use the deployment automation tool for production deployments while a team leader has permission to do it.

Table 7 shows the results for these two questions. From Table 7 one can see that the interviewees have considerable knowledge about the existence of DevOps practices. From the 12 practices addressed in Section 2.3 (Jabbari et al., 2016), Shift-left and Infrastructure as Code were the only practices that the interviewees had no prior knowledge of. Furthermore, from Table 7 one can conclude that the most known practices are being fully or partially applied. The authors also noted that there appears to exist a relation between the experience of the interviewee and the practices implemented. For example, the deployment automation practice is fully applied by interviewees E, B and F, while the others only applied it partially. The CI is being fully applied by the entire team, likely because it is an intuitive and easy practice to employ due to the existence of tools that allow this practice, like Jenkins.

5.1.1 Incident management phases vs. DevOps practices (RQ1)

Given the practical experience and knowledge of the interviewees, the authors introduced a matrix (like Table 8) to gain better understating of the questionnaire where each DevOps practice can be applied in each IM process phase as shown in Table 8. The authors highlighted and coded interviewee's answers (grey cells in Table 8), why (Wx in Table 8) and how (Hx on Table 8) the practices can help on each IM phase. The grey cells are coded by three tones which go from a lighter to a darker grey, where one or two matches are identified with the lighter grey tone, three matches are the medium grey tone, and greater than three matches are the darker grey tone.

From Table 8 one can see that the only practice where the interviewees did not point any possible correlation is the Shift-left. The interviewees' lack of knowledge on the corresponding practice is a possible reason for such finding. Regarding all the other practices, the interviewees engaged them in one or more IM phases. The IM phases considered in this research are the IM phases described by Cannon and Wheeldon (2007).

Table 7 Practices known vs. fully and partially applied

	Continuous planning	*Feedback loops between Dev & Ops*	*Continuous integration*	*Deployment automation*	*Test automation*	*Change management*	*Automated monitoring*	*Prototyping application*	*Stakeholder participation*	*Process standardisation*	*Shift left*	*Infrastructure as code*	*Total*	*Percentage*
							Practices known							
A			●	●	●	●	●						*5/12*	*42%*
B		●	●	●	●	●	●	●	●	●			*9/12*	*75%*
C	●	●	●	●	●	●							*6/12*	*50%*
D	●		●	●	●	●		●	●	●			*8/12*	*67%*
E	●	●	●	●	●	●	●	●	●	●			*10/12*	*83%*
F	●	●	●	●	●	●	●	●	●	●			*10/12*	*83%*
G	●		●				●						*3/12*	*25%*
H	●		●	●	●	●	●		●	●			*8/12*	*67%*
I	●	●	●		●	●	●	●		●			8/12	67%
J		●	●	●	●	●	●	●	●				8/12	67%
Total	*7*	*6*	*10*	*8*	*9*	*9*	*8*	*6*	*6*	*6*	*0*	*0*		
							Practices fully vs. partially applied							
A			●	◖	◖	●	◖						*5/12*	*42%*
B		●	●	●	◖	●	◖	◖		●			*8/12*	*67%*
C	◖	◖	●	◖	◖								*5/12*	*42%*
D	●		●	◖		◖			●				*5/12*	*42%*
E	●	●	●	●	◖	●	◖		●				*8/12*	*67%*
F	◖	◖	●	●	◖	●	◖		◖	◖			*6/12*	*50%*
G	◖		◖				◖						*1.5/12*	*12%*
H	●		◖	◖		●			◖	●			*4.5/12*	*37.5%*
I	●		●			●				●			*4/12*	*33.3%*
J		◖		●		●	◖	●	◖				*4.5/12*	*37.5*
Total	*5.5*	*3.5*	*8*	*6*	*2.5*	*7.5*	*3*	*1.5*	*3.4*	*3.5*	*0*	*0*		

To extend Table 8, Table 9 lists interviewees' opinions on why organisations would benefit by applying DevOps culture on IM process and how one could achieve such benefits. The collected information answers the RQ1 by describing the relation between

DevOps practices and IM process phases in more detail, grounded on the experience of the AM team under study. Such mapping is a step forward in this subject. The qualitative data present in Table 9 give us interesting and novel qualitative information to answer RQ 1.1 and RQ 1.2. This table shows arguments from the interviewees to justify why and how DevOps practices may be applied in each IM process phase.

Table 8 IM phases where DevOps practices can be applied

	Detection and recording	*Classification and initial support*	*Investigation and diagnosis*	*Resolution and recovery*	*Closure*	*Monitor and tracking*
Shift-left						
Continuous planning	2 W1, H1	3 W2, H1, H2	1 W3, H1			
Feedback loops between Dev and Ops			3 W4, H3			
Continuous integration			3 W5, H4	4 W6, H4		1 W7, H5
Automated monitoring	1 W8, H6					5 W9, H6
Prototyping application				3 W10, H7		
Deployment automation				2 W11, H8	3 W12, H8	
Test automation	1 W13, H9		1 W14, H10	2 W15, H10		1 W16, H10
Infrastructure as code			3 W17, H11			
Stakeholder participation	4 W18, H12	3 W19, H13	4 W20, H12			
Process standardisation	2 W21, H13, H14	2 W22, H15	2 W23, H16	1 W24, H16		
Change management				1 W25, H17	5 W26, H18, H19	1 W27, H20

Each practice matches an average of approximately three IM phases, which indicates that the practices are, in fact, compatible with the IM process. Moreover, there are 2.3 matches for each grey cell 2.3 which shows that on average, two different interviewees have identified a match between the practice and the IM phase.

The practices with more matches in different IM phases were 'Process Standardisation' and 'Test Automation' matching four different IM phases. Since IM is a process it makes sense that teams who apply this process will try to make a standard for each phase, so it can be easier for everyone on the team to follow it. The Test Automation framework is used to ensure that the testing of new functionalities and incident fixes have the desired quality, ensuring that everything works appropriately.

Table 9 Interviewees' feedback

Why	
W1	"Continuous planning helps the business to know what needs to be fixed and the negative impact that it is causing"
	"It's important to register and centralise incidents to identify the ones that affect multiple users' ability to execute daily work"
W3	"Feedback provided while planning and selecting the next priorities will help in the investigation"
W5	"All the code will be easily merged, facilitating its diagnosis"
	"With CI it is possible to keep track of the changed code, which will be easier to find the person who changed it (given that person is still working for the company) to know why the code was changed that way, since that person could have different thinking on how the functionality should work"
	"Having all the code integrated on the last version and ready to be deployed in any environment may help with replicating incidents, avoiding misalignments between lower environments and production environment"
W8	"Automated monitors are useful to check the health of the system detecting incidents"
W9	"Constant monitoring of the system to find if the incident was solved"
	"Helps to find issues and to guarantee that the fixes are working"
	"The automated monitor will check if the system is ok; this way will also monitor if the fix for the incident was successful"
How	
H2	"By perform quick analysis of the issue reported and affected portfolio"
H3	"Promote Knowledge Transfer sessions"
	"Consider inviting operations for discussions when analysing incidents"
	"Having more sessions between Dev and Operations"
H19	"By requesting action requests/changes to responsible teams"
H20	"By planning in advance future releases in the system"

But these tests can be applied in different contexts according to the interviewees (Table 9). This explains why these tests are not only related to the 'Resolution and Recovery' phase where the solutions are being taken and tested.

The practices that matched fewer IM phases were as follows: 'Automated Monitor', matching two different IM phases; 'Prototyping Application', matching one phase; and 'Deployment Automation', matching two different phases. Regarding the 'Automated Monitor' practice, half of the interviewees placed a match on the Monitor phase, as they have found that this practice is helpful to determining how the incident was really solved. However, there was one interviewee who matched the Detection phase, justifying that the automated monitor might find issues that were never identified. Prototyping Application was only matched with 'Resolution and Recovery' but it was matched by three different interviewees, which is almost half of the interviewees. For Deployment Automation, it was found in 'Resolution and Recovery' and 'Closure'. The purpose of this practice is to speed up the delivery of the code in several environments. Therefore, it will speed up the closure of the incident and its resolution.

To analyse the qualitative data provided in Table 9, the authors opted to concentrate on the most quoted matches (equal or greater than three) from Table 8. Table 9 is just a sample of the interviewees' answers, and the full table can be seen in Appendix C. Also, the authors analysed some matches that seems contradictory between the interviewees, which makes a relevant discussion (presented in Table 10) on how the DevOps practices might help the IM process phases.

From Table 10, it is possible to conclude that all the IM phases were approached regarding the matching with DevOps practices. This reinforces the idea that DevOps practices can help in several phases of the IM process.

Table 10 Discussion on interviewees' feedback

Wx, Hx	*Practice*	*IM phase*	*Comments*
W2, H1, H2	Continuous planning	Classification and initial support	The quotes for this practice show that the interviewees are greatly concerned with the prioritisation of their tasks. Their objective is to help the firm's client's, but they need to know what the most critical tasks are, so there can be better alignment between the AM team and the business. Two of the quotes shown talk about this prioritisation, while one of the quotes cites their concern on collecting feedback from the customer to provide the initial support This may be achieved by having meetings with the business and regularly reviewing the incidents backlog
W4, H3	Feedback loops between Dev and Ops	Investigation and diagnosis	Here the interviewees have focussed on feedback and knowledge sharing between Dev and Ops teams, bringing better cohesion between both teams and quality to the final solution to deliver to the business
W5, W6, H5	CI	Investigation and diagnosis; resolution and recovery	For both matches, the interviewees have mentioned the importance of the integration and alignment of the code. Having the last code version installed on lower environments will help with diagnosing the root cause of incidents, thus, accelerating resolutions. It was interesting to see that the same 'why' can benefit two different IM phases using the same 'how' (H4)
W8, W9, H6	Automated monitor	Detection and recording; monitor and tracking	Even though W7 only had one match, the authors found quite curious how all the interviewees had focussed their answers on the Monitor and Tracking phase, creating the match W8. On W8 the interviewees showed that their concern was to use the Automated Monitor to check the system health and if their fixes had indeed solved the incident. While in W7 the only interviewee had justified his choice by using the Automated Monitor to find new incidents. To achieve this, the interviewees suggest implementing some automatic dashboards or scripts to produce reports related with system health

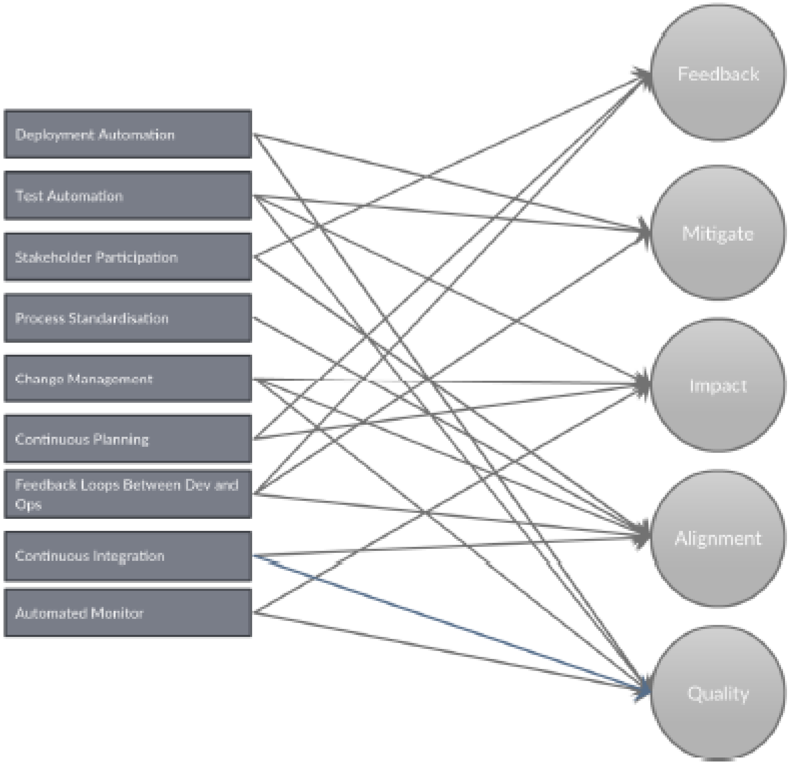
Table 10 Discussion on interviewees' feedback (continued)

Wx, Hx	*Practice*	*IM phase*	*Comments*
W10, H7	Prototyping application	Resolution and recovery	The interviewees have commented this match by showing the result to the business. It looks like this team thinks this practice will lead to a better alignment between them and the business
W11, W12, H8	Deployment automation	Resolution and recovery; closure	Automation deployment is another example where it is possible to see that a practice can be applied for different contexts. The interviewees who found that Deployment Automation is useful for Resolution and Recovery have said that it can be used to deploy fixes for different environments quickly, so the users can approve of the fixes. While the interviewees who have matched the Deployment automation with the Closure phase have justified this by saying that the move accelerates incident closure due to the time saving that this practice brings to the team
W17, H11	Infrastructure as code	Investigation and diagnosis	In their answers, the interviewees have shown some concern regarding the readiness of their environments. None of them have ever applied this practice, but they seem to be intrigued while discussing this practice with the authors, showing interest in applying it later. The interviewees suggested using cloud environments, which provide better tools for implementation
W18, W19, W20, H12, H13	Stakeholder participation	Detection and recording; classification and initial support; investigation and diagnosis	Analysing the quotes from the interviewees, it is possible to conclude that they depend a lot on the business to succeed in their work. The major need that they require from the business is feedback. Without feedback to obtain answers on how to replicate the issues or to get the correct prioritisation for the incidents, they will not be able to solve the incidents as quickly. To achieve this, the interviewees recommend planning meetings with the business users to discuss the incidents impacts
W26, H18	Change management	Closure	The interviewees believe that this practice will facilitate the closure of the incident by guaranteeing the quality requirements that are needed to do the deployments in production. By guaranteeing the quality of the delivery, they can confirm that the incident was correctly solved, contributing to the closure of the incident. There is a need to implement this process very carefully, so it can guarantee the quality required for the deliveries

5.1.2 DevOps benefits (RQ2)

To find the benefits that the DevOps practices brought to this team, the authors have asked the interviewees, "Why have you started to apply this practice?" to determine its benefits as viewed by the participants. The answers are visible in the table provided in Appendix D and serve as the answer to RQ2. In this table we show the number of matches and some quotes from the interviewees citing their justifications.

Figure 3 Practices vs. keywords



Analysing Appendix D, the authors tried to identify keywords that could translate into generic benefits of each practice. The keywords identified by the authors were the following: feedback, mitigate, impact, alignment and quality. By looking at Appendix D, these words are largely used by the interviewees in several practices. For better understanding, the authors highlighted these keywords on the quotes column of Appendix D.

In analysing the table provided in Appendix D, it is possible to find that there is a relationship between these keywords and the practices, which enabled the authors to investigate the benefit behind that practice. One can find these relationships in Figure 3.

Based on Figure 3 and Appendix D, the authors were able to elicit and synthesise the benefits described by the interviewees for each practice, as shown in Table 11.

After analysing Table 11, the authors summed up the benefits of DevOps adoption in the IM process, raising five major concepts, which where possible to map with the benefits identified in the literature review (Section 2.3). This can be seen in Table 12.

5.1.3 DevOps challenges (RQ3)

To determine the DevOps challenges, the authors asked, "What was the adoption of these practices like?" This question was rated from 1 to 5, meaning 'Very Hard', 'Hard', 'Neutral', 'Easy' and 'Very Easy'. The interviewees were given the opportunity to justify their answers. By doing so, it was possible to collect their opinion about the challenges of adopting each DevOps practice. In Table 13, one can find the interviewees' answers to this question in a condensed format; the full set of answers can be seen in Appendix A.

The columns in the table list the different ratings that the interviewees could choose (with respective comments) and the practices in the rows, creating a matrix. One of the columns presents the sum of the interviewees' ratings so the reader can have an idea of which practices are easier or harder to adopt. Not all the interviewees had experience in the practice in question. So, the sums of the ratings can be different from practice to practice.

Regarding the challenges, the authors have reviewed the answers from the interviewees and take the main idea from their comments to identify the challenge. The authors only considered the practices where the average was less than three, since three in the questionnaire means neutral.

Table 11 Keyword conclusions about the benefits

Practice	*Keywords*	*Conclusion*
Continuous planning	Feedback Impact	With this practice development teams, together with business, can plan the next steps based on the feedback and the impact of the incidents on the business, contributing to business satisfaction and business engagement with the development teams
Feedback loops between Dev and Ops	Feedback Impact Alignment	All the teams can bring their feedback to the table regarding the new developments, reducing the impacts to the business. This practice also guarantees an alignment between the developers and operators, where they can learn from each other, contributing to the quality of software delivery and engagement between developers and operators and maximising competences
Continuous integration	Alignment Quality	This practice will bring alignment between the developments, contributing to the quality of software deliveries and engagement between teams
Automated monitoring	Impact Quality	The Automated Monitor is essential to guaranteeing a fast response to the recent issues, minimising their impact and guaranteeing the quality of the fixes from the AM team, which contributes to software quality
Deployment automation	Mitigate Quality	Deployment Automation is a key practice to mitigate human error, ensuring better quality software delivery
Test automation	Mitigate Impact Quality	Regression tests can be performed automatically, mitigating human error, which will result in less impact on existing functionalities. This will bring more quality to the software solution that was developed
Stakeholder participation	Feedback Alignment	Feedback from all the stakeholders is the key to the success of any application, leading to engagement between stakeholders
Process standardisation	Alignment	The standardisation of processes will lead to alignment between all individuals and later between teams, which will guarantee that everyone will work the same way, leading to quality developments
Change management	Impact Alignment Quality	The change management process measures the impacts of the software change, where all the involved teams will need to be aligned to ensure the quality required for the software

Table 12 Conclusions about the benefits

Benefits	*LR (Section 2.3)*	*CS*
Quality	B1 B3	The participants identified quality of the software delivery in several topics. The quality of software delivery is key for every development team. The quality of the software delivery should not be measured when the software is delivered but during all stages until delivery: meaning, requirement gathering, designing, building, testing. If the quality is improved in all phases, the software delivery quality will be higher
Engagement	B2 B4	The engagement of all the stakeholders on the application is a key success factor for the application. Everyone, this means business users, developers, operators, managers, etc., need to be on the same page; otherwise, the success of the application will not be maximised
Value	Not found in the LR	The objective of every project is to bring value to the business. From the quotes of the interviewees, they are very focussed on getting the feedback from the business and to provide their feedback to businesses to improve them. They know that the business is depending on the applications and since they are responsible for maintaining these applications, they not only try to fix them but also to improve them and avoid possible issues. They implement practices that help them to find the issues quickly to minimise impacts and even find them before the issues happen
Integrity	Not found in the LR	The interviewees are currently maintaining an application that currently is not finished. The development teams are currently working and adding more functionalities to the application. This requires substantial integration between these two workstreams
Personnel development	B6	The concept behind DevOps is to join Operators and Developers. Joining these two workstreams will make them share knowledge between them, which will create more capable professionals who are able to work for these two workstreams

From Table 14, it is possible to identify that the interviewees had challenges in implementing five of the 10 practices that they currently apply. The main challenges found were as follows: time spent on documentation, culture, communication and the technical challenge of implementing the respective practice. Like the previous section, the authors mapped the findings of Section 2.3 and the challenges the interviewees identified.

In Table 15, the authors identify four concepts and some challenges. Comparing the results from the LR in Section 2.3 with the results of the interviewees, it is possible to see that for Technical Challenges the interviewees have stated the difficulty of maintaining the monitors updated to observe the current solution. The Automated Monitors are both important for the developers of the AM team and for the Ops team. This results in a new challenge for the DevOps: since this culture promotes a quick delivery life cycle, it will be possible to deliver with more frequency, not giving enough time to maintain the monitors. Unless, maintaining the monitors should be a task for the developers when they make new developments that could affect these monitors. Regarding the time consumption, all the other authors never found that time was an issue. However, based on the interviewees they lost ample time producing the necessary documentation to support the agile meetings, resulting also in a new challenge.

Table 13 DevOps challenges in each practice

Practices	*Rates* 1	2	3	4	5	*Average*	*Comments*
Continuous planning		3	1	2		2.8	"It was hard to define which ceremonies should be part of this practice"
Feedback loops between Dev and Ops	1		2	1		2.75	"Challenges due to busy agendas and different time zones between Dev and Ops"
Continuous integration			2	4	2	4	"Easy to implement with the correct tools. Also, easy to understand its benefit"
							"Easy process to use"
Automated monitoring	1		2			2.3	"Requires some time to build these monitors and know what should be monitored"
Deployment automation		2	3	1	1	3.1	"The automated deployment was complex to use due to the team's lack of knowledge of the tools"
Test automation	1	1	2			2.25	"Hard to configure and to maintain due to the continuous delivery"
Stakeholder participation	2		3			1.6	"Lack of engagement from the stakeholders to participate in some decision processes"
Process standardisation		1	1	1		3	"The project management encourages this since it will reduce the mistakes during the process execution"
Change management		1	2	4		3.4	"On the change management process, there are a lot of people involved. The communication between all these people is not easy"

Table 14 Conclusions about the challenges

Practice	*Average*	*Challenge*
Continuous planning	2.8	The challenge for this practice was the time spent to build the support documentation for the required meetings
Feedback loops between Dev and Ops	2.75	The interviewees felt some lack of will from the Ops side to break down the silos; also, the combination between agendas of different time zones was a challenge
Automated monitoring	2.3	It requires time to build these monitors, and the management does not see much value on implementing them since it will take some time to maintain these monitors due new releases
Test automation	2.25	The interviewees have stated this practice is hard to incorporate in their system and to maintain due to new releases
Stakeholder participation	1.16	The interviewees stated there are challenges due to the technical/functional language that can lead to a lack of interest from stakeholders. Also, some stakeholders do not want to be included in the decision-making process since they do not want to have to be accountable if something goes wrong

Table 15 Challenges crosscheck between LR and CS

Concepts	*LR (Section 2.3)*	*CS*
Technical challenges	C5	The interviewees have stated they have found some challenges while implementing the Automated Monitor and Test Automation. The automated monitor needs to be aligned with the current solution that is in production; since the DevOps culture promotes a fast delivery life cycle, it is hard for the Automated Monitor to keep up with these changes. Regarding Test Automation, these tools are hard to configure, which makes this AM team spend a lot of time just to configure/reconfigure the tool
Culture	C2 C3	The interviewees stated the fear of the stakeholders to take accountability for their decisions, which may be related with deep-seated company culture. It is common to have a deep-seated company culture in the financial sector, like the company where these interviewees provide their services
Communication	C1	The interviewees also stated the reluctance of the Ops to provide feedback on the sections of "Feedback between Dev and Ops". This may be related with the insufficient communication challenge since the Ops side does not want to communicate with the developer
Time spending	Not found on the LR	The interviewees have stated they need to spend a lot of time on preparing meetings and all the documentation that is required for these meetings

5.1.4 Team performance (RQ4)

To evaluate if the team performance was improved due to the implementation of the DevOps practices, the authors added a question on the questionnaire where the interviewees could rate from 1 to 3, 1 meaning did not improve while 3 means improved; a box was also provided for comments to justify why a practice had or had not improved. Again, an abridged version appears in Table 16, while the full table can be found in Appendix B.

Analysing Table 16, it is possible to conclude that most of the practices that the interviewees have in place, have improved their work. All the practices have an average greater than two, except for Test Automation, where the final average was 2. From the interviewees' point of view, the test automation practice can be a good practice to improve their work, since they can apply this practice to execute regression tests, which usually takes a lot of time. However, when a test is marked as failed on this tool, it takes a lot of time to check why the test was marked as failed; moreover, some of the failed tests are false positives, which may lead to wasted time.

The authors want to highlight the practices 'CI', 'Automated Monitoring' and 'Continuous Planning'. These three practices got an average of three, which was the maximum rate for this question. For the CI the interviewees focussed their answers on saving time since they have reduced their merge activities and the human error of these activities. When working with other teams, the code merging activities can be very time consuming, as no one wants to make an error on others' code.

Table 16 Team performance

Practices	*Rate* *1*	*2*	*3*	*Average*	*Comments*
Shift-left					
Continuous planning			6	3	"It helps the team to define objectives and keep their focus on the tasks"
Feedback loops between Dev and Ops		2	2	2.5	"It helps to prevent issues, but it takes a lot of time in meetings"
					"Due to a better relationship between Dev and Ops, errors may be found earlier and even avoided"
Continuous integration			8	3	"It will ensure the code merge, avoiding placing effort on merging activities"
					"It prevents errors in merging and facilitates the alignment between teams"
Automated monitoring			3	3	"Reduces time on monitoring activities"
Deployment automation		2	5	2.71	"Reduces time on deployment activities so there are no worries on creating manual packages"
Test automation		4		2	"Even when automated tests are made, we always need to perform unit and integrated tests
Stakeholder participation		1	3	2.75	"It can improve the performance if all the stakeholders that are involved on the discussion are interested on the topic"
Process standardisation	1		2	2.33	"It requires a lot of time and efforts to define the processes"
Change management	1	2	3	2.33	"It wastes some time to ensure that the right participants are doing their tasks"

Regarding continuous planning, the interviewees have focussed on having a defined scope for their tasks. As previously said in other sections, this team seems to have problems on working within priorities. This practice will allow a continuous scope of activities, so the interviewees do not need to be changing all their tasks from day by day. The interviewees for the Automated Monitor, focussed on how it saves time. The Automated Monitors will create reports or a dashboard, so this team can check the health status of the system and react on time if something goes wrong.

For the remaining practices, the interviewees talked about other topics, like maximising competences (feedback between dev and ops), saving time (deployment automation), the importance of the business of taking decisions (stakeholder participation), making an easier process for everyone to follow (process standardisation) and achieving quality (Change Management).

One can conclude that these interviewees are concerned about the improvement of their work. It is possible to identify that they consider that these practices improve their work, since most of the practices that they implement got an average of greater than two (neutral). They have focussed their improvement of performance on the time saving and on the quality of the delivery. They seem to be satisfied with the time that has been reduced in support and routine tasks, so they can focus on the problems that their business users face every day. Also, they appear to be concerned with the quality of their deliveries and this was also a focus on their answers when asked about performance increase.

Based on the previous statements, it is possible to conclude that DevOps practices may improve the IM process by reducing the time to reach the resolution.

5.2 Document analysis

The documents analysed by the authors were provided by one of the interviewed Team Leaders. Due to the risk of broken confidentiality between this consulting team and their client, all the documents were anonymised, hiding the identity of their client. Plus, the authors could only see the documents using the laptop of the interviewed Team Leader. The authors had the opportunity to analyse two kinds of reports from this team: the one produced at the end of each sprint to evaluate what needs to be changed. The other were weekly performance reports to highlight key issues that were the focus of the AM team for that week. By analysing these reports, the authors intended to cross-check information from the interviewees' feedback. The authors analysed all the reports produced between March 2016 and June 2018, Overall, 18 sprint reports (major releases) and 115 weekly reports were analysed.

Table 17 Report analysis

Type of report	*Report findings*	*Interviewees cross-check*	*Metrics*			
			Detail	*Baseline*	*Evolution*	*Δ% to baseline*
Sprints	In a former report, AM team evidences the lack of engagement of the business on providing feedback and help on the analysis of the incidents. While on the last one, business engagement is already pointed as something positive	W15 H9	N/A	N/A	N/A	N/A
	On the most recent reports, the AM team evidences how important could be implement automated deployments during the test phase of the sprint, optimising the tests of the users	W8	Time to deploy a bugfix	48 h	2 h	–95%
	Changes to scope were also mentioned on these reports. Due to the continuous planning, business users could change the incidents' scope to be delivered on that sprint. The users can prioritise these incidents by their impact, having them solved more quickly.	W1 W2	N/A	N/A	N/A	N/A

Table 17 Report analysis (continued)

Type of report	*Report findings*	*Interviewees cross-check*	*Detail*	*Metrics* *Baseline*	*Evolution*	*Δ% to baseline*
Sprints	There is no record of the AM team not delivering any incident where was compromised to deliver	N/A	N/A	N/A	N/A	N/A
	Before of the implementation of the Continuous Planning and other agile practices, the AM team did not have any evidence of over deliver incidents that were not planned on the plan of the release. After the implementation of those practices the team was able to show 'Out of Sprint' scope that was being delivered on the release (an average of 10,1 incidents per release)	N/A	Over delivered incidents	0 incidents	10.1 incidents	+100%
	The AM describe that the Business was not interest on showing up on the sprint planning ceremonies, which result on a lot of changes on the sprint scope. After they started to show, the sprint scope started to have less changes, and the changes that were made was a well-defined agreement between the AM team and the business	Appendix B, Continuous Planning quotes	Number of attendees from the business side	1 attendee	5 attendees	+400%
	Before the implementation of the CI and automated deployments, all the environments were misaligned, which impacts a lot the acceptance tests environment. After the implementation of these practices the environments come more stable and aligned	W5	N/A	N/A	N/A	N/A
	On the first, the AM team realises on how the agile ceremonies for the continuous planning consumes time. However, they do not see this as an issue anymore since they have made standard documents	N/A	Time to prepare documentation for ceremonies	4 days	1.5 days	–62.5%
	Stakeholder Participation – The AM team misses the expertise of some areas on the continuous planning meetings. After some time, they do not see this as a problem anymore	Appendix B, Stakeholder Participation description	From the five attendees before mentioned, there are at least one representative of each application module	0 attendees	1 attendee	+100%

Table 17 Report analysis (continued)

Type of report	*Report findings*	*Interviewees cross-check*	*Metrics* *Detail*	*Baseline*	*Evolution*	*Δ% to baseline*
Performance reports	The AM team could implement biweekly releases with only hotfixes for the most critical incidents (identified after the sprint planning and due to the complexness were not able to include on the sprint). Such biweekly releases exist due to the deployment automation and CI performed by the team. Having a good integration of the software allow to have several tracks of development without having merge errors. Also, the deployment automation saves developers' time, so they can focus on solving the incidents	W4 W5 W9 H6 H9	Number of releases per month	1 release per month	3 releases per month	+200%
	These reports also evidence the existence of some problems on the production infrastructure. The AM team have implemented these monitors to have a reactive posture in case that something was not right. On these reports is stated that the issues were found in time, minimising the impact for the business users	W6	Average of infrastructure problems per month	3 problems per month	0.5 problems per month	–83.33%

The analysis of the documentation was useful to bring consensus on the information collected from the interviewees. These findings can be seen in Table 17. The main findings were the time to deliver a fix, in which the time was reduced in 48 h, over deliver of fixed and more releases per month. These findings demonstrate that some of the premises of DevOps were fulfilled, such as lead time between releases and less time to deploy a delivery.

5.3 Direct observation analysis

As previously stated before, the authors have chosen to perform a unstructured observation, which may also be named as participant observation (Thomas, 2016). The team manager allowed to the authors to perform the observation, however only during periods which causes less impact for the business. The observer was a different author than the author that performed the interviews, this way the team members would not recognise the observer and they would be more transparent while performing their job. Unfortunately, it was not possible to observe how the implementation of the practices affected the interviewees behaviour, therefore, the observation will just be used to validate the responses and findings from the other data collection methods. According to Thomas (2016), this, reinforces the definition of this CS (retrospective) stated in Section 3. In Table 18 one can analyse the findings within its sources and if it can be confirmed by observations. Overall, only 10 of the 19 findings were not able to be confirmed by the observation, due to not having a baseline to compare the before and after. It is also possible to note that some of the findings are supported by both semi-structured interviews and report analysis.

Validation of the triangulation between data collection methods will be discussed in the next section.

Table 18 Research findings

ID	*Main findings*	*Source*	*Confirmed by observation?*	*Comments*
F.1	Shift-left was not considered by the Interviewees	Semi structured interviews	Yes	This team usually does not evolve the Ops since the beginning of the software development cycle. Only when it is just really needed
F.2	Each DevOps practice matches at least in an average of 3 IM phases (50% of the IM phases)	Semi structured interviews	No	By observing the interviewees behaviour, it is not possible to see in which phases they apply each practice. Only by looking at documentation
F.3	All the IM phases have at least a match	Semi structured interviews	No	By observing the interviewees behaviour, it is not possible to see in which phase of the IM process the practice is being applied
F.4	Automated Monitor, Prototyping application and Deployment automation matched fewer phases since they can only be applied to reduced contexts	Semi structured interviews	Yes	These practices are applied in reduced scenarios by the AM team
F.5	CI helps on the incident analysis and resolution	Semi structured interviews/ Performance reports	Yes	Sometimes, there were some situations where the team could not replicate the issue reported by the business user. After performing the integration of the most recent code to a lower environment, it was possible to replicate the issue like in production. The environment alignment is also discussed in the report analysis
F.6	Prototyping Application helps to understand the business needs	Semi structured interviews	Yes	It was possible to see that the business was satisfied with seeing some demos before deployment, so they can check the behaviour after the new code

Table 18 Research findings (continued)

ID	Main findings	Source	Confirmed by observation?	Comments
F.7	Infrastructure as Code may help ensure the readiness of the environments	Semi structured interviews	No	Since this team does not apply infrastructure as code, it was not possible to observe
F.8	Stakeholder Participation may help on Detection and Recording, Classification and Initial Support, Investigation and Diagnosis	Semi structured interviews	Yes	It was possible to see that the AM team often looks for the business when they find issues, expresses the need to prioritise them and needs help with the investigation. However, it is not possible to confirm in which phase the AM team is
F.9	Change Management is related with Closure IM phase	Semi structured interviews	Yes	It was possible to confirm that the Change Management is related with the closure of the incident. An incident that requires code fix needs to be related with a Change, and the incident will only be closed once the change is approved by the several quality controls
F.10	Several benefits were uncovered after applying the DevOps Practices: Quality; Engagement; Value; Integrity; Personal Development	Semi structure interviews	No	Since the authors could not follow the implementation of the practices, it is not possible to analyse if there are benefits after implementation. However, the increase of engagement is also referred to in the analysis
F.11	It was possible find the following challenges when implementing the DevOps practices: Technical Challenges; Culture; Communication; Time Spending	Semi structured interviews	No	Since the authors could not follow the implementation of the practices, it is not possible to analyse if there are benefits after implementation. Time resources are also discussed in the report, saying that over time, this problem subsides
F.12	All the practices improved the performance except test automation	Semi structured interviews	No	Since the authors could not follow the implementation of the practices, it is not possible to analyse if there are benefits after implementation
F.13	CI and Deployment Automation saves time	Semi structure Interviews/ Performance reports	Yes	There is possible to see that this team saves a lot of time when performing continuous merges using CI where the human error is reduced. This also applies for Deployment Automation

Table 18 Research findings (continued)

ID	*Main findings*	*Source*	*Confirmed by observation?*	*Comments*
F.14	With continuous planning it is possible to have a well-defined scope, and in case of making changes to the scope, it could be aligned between business and team management	Semi structure interviews/ Sprint reports	Yes	In some meetings it was possible to see that the business users wanted to change the scope of the releases a few times, since the plan is made on a continuous way, it was possible to align and change the scope. This is also mentioned that the scope was changed several times during the sprints, on the reports
F.15	While discussing the performance improvement of the team it was referred to as Time saving; Maximising competences	Semi structured Interviews	No	Even that the authors do not have a baseline to compare times for time saving, the authors recognise that applying some of the practice reduces time due to all the manual work that was performed before. Regarding the maximising of competences, it was possible to see that some of the developers do not have any knowledge of the database/environment maintenance, as they are now able to make analysis on the database/environment issues by themselves
F.16	Regarding the performance it was also mentioned an increase of quality	Semi structured interviews	No	There is no baseline to consider before and after the quality that is delivered by this team
F.17	More releases	Performance reports	Yes	It was possible to see this team have several releases per month
F.18	More performance (overdeliver)	Sprint reports	Yes	By looking at the scope delivered incidents by this team, was possible to check that they can deliver a lot of out of sprint incidents
F.19	Infrastructure problems	Performance reports	No	Even not having a baseline to compare the before and after, it is possible to analyse that this team does not handle so much infrastructure issues

5.4 *Synthesis of results*

In this section authors compare the findings that were described on Sections 5.1–5.4, to validate the data collection methods triangulation, as can be seen in Table 19. This table relates all the findings that will be able to answer to each RQ. It also shows in which data collection method it was collected. It is possible to see that most of the findings were found on the Interviews and more than 50% of the findings can be found at least in 2 of the data collection methods, showing that the triangulation of data collection methods was useful in this case study.

Table 19 Data analysis synthesis

Research question	*Finding*	*I*[1]	*R*[2]	*O*[3]	*S*[4]
RQ.1	F.1	X		X	5.1.1
RQ.1	F.2	X			5.1.1
RQ.1	F.3	X			5.1.1
RQ.1	F.4	X		X	5.1.1
RQ.1	F.5	X	X	X	5.1.1/5.2
RQ.1	F.6	X		X	5.1.1
RQ.1	F.7	X			5.1.1
RQ.1	F.8	X		X	5.1.1
RQ.1	F.9	X		X	5.1.1
RQ.2	F.10	X			5.1.2/5.2
RQ.3	F.11	X			5.1.3
RQ.2; RQ.4	F.12	X			5.1.2/5.1.4
RQ.2; RQ.4	F.13	X	X	X	5.1.2/5.1.4
RQ.2	F.14	X	X	X	5.1.2
RQ.2	F.15	X			5.1.2/5.1.4
RQ.4	F.16	X			5.1.4
RQ.2	F.17		X	X	5.2
RQ.2	F.18		X	X	5.2
RQ.4	F.19		X		5.2

[1]Semi structured interview.

[2]Report analysis.

[3]Observation.

[4]Section where this finding is described.

6 Conclusion

Thanks to the interviews made to IT professionals that apply DevOps practices while working with the IM process, and due to their documentation regarding their performance, it was possible to collect a dataset, presented in Section 5, with a lot of findings to answer to the RQ proposed for this study

With these interviews and documentation, it is possible to conclude that these practices can help to increase AM team performance as well as the engagement with business users by making them involved with the solutions that are provided by the AM team, when diagnosing and solving the incidents.

Due to the automation practices like testing and deployment, the interviewees also pointed that they could perform more emergency changes, contributing to the health of the application and to solve the incidents that cause more impact faster. They have also shared, that they would like to fully apply some of the practices like test automation, automated monitoring and infrastructure as code because they understand that by applying this, they have more benefits. Most of the practices were implemented by request of the AM team's client, however some of them, like feedback loops between Dev and Ops and Process Standardisation, were practices that are encouraged to be practiced by the team management, due to the performance improvement that these practices can bring. Also, using Feedback Loops, the AM team could expose some issues regarding new developments, increasing the quality and preventing future problems on the application.

In general, the interviewees are happy to apply these practices due to the agility of DevOps and the involvement of all the stakeholders, they feel their work has impact and it is recognised by the entire organisation.

It is possible to see that the agile principles are well grounded on the DevOps culture. This culture encourages the communication and collaboration not only between the IT teams, but also between the IT teams and the business units, to gather and include as much feedback as possible. Also, more but smaller releases help both developers and IT operations to stabilise the system.

To conclude, all the RQs that this research proposed to answer, were addressed. Regarding RQ1, according to the interviewees all the DevOps practices may be used in the several phases of the IM process, except for Shift-left, which is considered a limitation in this research. Regarding RQ2, several benefits were found for each practice and they were common for all practices, such as, quality of the deliverables, engagement from all the stakeholders and personnel development. Also, interviewees reported some benefits that were not found on the LR, like Value and Integrity. Regarding RQ3, some challenges were also identified as similar as the in the LR, but in this research it was also possible to identify some more challenges that were not previously identified like Time Spending. Regarding RQ4, it was possible to see that the practices reduce time in several manual tasks, IM performance.

Theoretical contribution of the study could be identified in terms of establishing new baselines for further research. In addition, this research provides new insights for the practitioners. In the absence of studies exploring the relation between DevOps and ITSM (IM and AM team), this research brings new insights on why and how an AM team should adopt DevOps practices. Benefits of the practices are also mentioned on this research, as well as the adoption challenges such as time spending on documentation and communication, so the practitioners be aware of their possible outcomes.

6.1 Research limitations

This research also has some limitations. First, DevOps is a very recent culture and few strong studies exist in respectful journals and conference proceedings that can be related to this topic. Second, this research is based on data obtained from a single team. An example of this limitation is that no member of this team knows about the Shift-Left practice. So, this research lacks any conclusions regarding its impact on the IM process.

6.2 Future work

Future researches may investigate how DevOps practices may be applied in other ITSM processes. This is a goal that the authors intend to pursue. Also, the authors suggest the exploration of additional challenges regarding the DevOps implementation, since most researchers appear to be focused on exploring the benefits.

References

Aguiar, J., Pereira, R., Vasconcelos, J.B. and Bianchi, I. (2018) 'An overlapness incident management maturity model for multi-framework assessment (ITIL,COBIT,CMMI-SVC)', *Interdisciplinary Journal of Information, Knowledge and Management*, Vol. 13, pp.137–163, doi: https://doi.org/10.28945/4083.

Alsolamy, A.A., Khan, U.A. and Khan, P.M. (2014) 'IT-business alignment strategy for business growth', *2014 International Conference on Computing for Sustainable Global Development, INDIACom 2014*, pp.364–366, doi: 10.1109/IndiaCom.2014.6828160.

Badinelli, R., Polese, F., Saviano, M. and Di Nauta, P. (2012) 'Viable service systems and decision making in service management', *Journal of Service Management*, 23(4), pp. 498–526. doi: 10.1108/09564231211260396.

Barksdale, J.T. and McCrickard, D.S. (2012) 'Software product innovation in agile usability teams: an analytical framework of social capital, network governance, and usability knowledge management', *International Journal of Agile and Extreme Software Development*, Vol. 1, No. 1, p.52, doi: 10.1504/IJAESD.2012.048302.

Beddle, M., Bennekum, A.v., Cockburn, A., Cunningham, W., Fowler, M., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C. Schwaber, K., Sutherland, J. and Thomas, D. (2001) *Principles behind the Agile Manifesto*, Available at: http://agilemanifesto.org/history.html (Accessed 6 January, 2017).

Bezerra, G., Pinheiro, V. and Bessa, A. (2014) 'Incident management optimization through the reuse of experiences and natural language processing', *2014 9th International Conference on the Quality of Information and Communications Technology*, pp.58–65, doi: 10.1109/QUATIC.2014.14.

Cannon, D. and Wheeldon, D. (2007) *ITIL Version 3 Service Operation*. 1st ed., Office of Government Commerce, The Stationery Office, London, doi: 10.1016/j.im.2003.02.002.

Cao, J. and Zhang, S. (2016) 'IT operation and maintenance process improvement and design under virtualization environment', *Proceedings of 2016 IEEE International Conference on Cloud Computing and Big Data Analysis, ICCCBDA 2016*, pp.263–267, doi: 10.1109/ICCCBDA.2016.7529568.

Chen, B. (2019) 'Improving the software logging practices in devops', *Proceedings – 2019 IEEE/ACM 41st International Conference on Software Engineering: Companion, ICSE-Companion 2019*, IEEE, pp.194–197, doi: 10.1109/ICSE-Companion.2019.00080.

Croker, M. and Hering, M. (2016) *DevOps: Delivering at the Speed of Today's Business DevOps: A Matter of Survival in the Digital Age*, Retrieved from https://www.accenture.com/_acnmedia/PDF-27/Accenture-DevOps-Brochure-Final.pdf

Dyck, A., Penners, R. and Lichter, H. (2015) 'Towards definitions for release engineering and DevOps', *Proc. 3rd International Workshop on Release Engineering*, IEEE, Florence, Italy, p.3, doi: 10.1109/RELENG.2015.10.

Erich, F., Amrit, C. and Daneva, M. (2014) 'Report: DevOps literature review', *University of Twente*, October, p.27, doi: 10.1007/978-3-319-13835-0.

Gacenga, F., Cater-Steel, A., Toleman, M. and Tan, W. (2011) 'Measuring the performance of service orientated IT management', *Sprouts: Working Papers on Information Environments, Systems and Organizations*, Vol. 11, No. 162, p.14, Available at: http://eprints.usq.edu.au/20396/1/Gacenga_Cater-Steel_Toleman_Tan_2011_SIGSVC_Workshop_PV.pdf

Ghrab, I., Ketata, M., Loukil, Z. and Gargouri, F. (2016) 'Using constraint programming techniques to improve incident management process in ITIL', *2016 Third International Conference on Artificial Intelligence and Pattern Recognition (AIPR)*, pp.1–6, doi: 10.1109/ICAIPR.2016.7585231.

Gotel, O. and Leip, D. (2007) 'Agile Software development meets corporate deployment procedures: stretching the agile envelope', in Concas, G., Damiani, E., Scotto, M., and Succi, G. (Eds.): *International Conference on Extreme Programming and Agile Processes in Software Engineering*, Springer, Berlin, Heidelberg, pp.24–27, doi: https://doi.org/10.1007/978-3-540-73101-6_4

Gottesheim, W. (2015) 'Challenges, benefits and best practices of performance focused DevOps', *Proceedings of the 4th International Workshop on Large-Scale Testing – LT '15*, p.3, doi: 10.1145/2693182.2693187.

Guerriero, M., Ciavotta, M., Gibilisco, G.P. and Ardagna, D. (2016) 'A model-driven DevOps framework for QoS-Aware cloud applications', *Proceedings – 17th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, SYNASC 2015*, pp.345–351, doi: 10.1109/SYNASC.2015.60.

Gupta, R., Prasad, K.H. and Mohania, M. (2008) 'Information integration techniques to automate incident management', *NOMS 2008 – IEEE/IFIP Network Operations and Management Symposium: Pervasive Management for Ubiquitous Networks and Services*, pp.979–982, doi: 10.1109/NOMS.2008.4575262.

Gupta, R.K., Venkatachalapathy, M. and Jeberla, F.K. (2019) 'Challenges in adopting continuous delivery and DevOps in a globally distributed product team: a case study of a healthcare organization', *Proceedings – 2019 ACM/IEEE 14th International Conference on Global Software Engineering, ICGSE 2019*, IEEE, pp.30–34, doi: 10.1109/ICGSE.2019.00020.

Hannola, L., Friman, J. and Niemimuukko, J. (2013) 'Application of agile methods in the innovation process', *International Journal of Business Innovation and Research*, Vol. 7, No. 1, p.84, doi: 10.1504/IJBIR.2013.050557.

Highsmith, J. and Cockburn, A. (2001) 'Agile software development: the people factor', *Computer*, Vol. 34, pp.131–134.

Hussain, W., Clear, T. and MacDonell, S. (2017) 'Emerging trends for global DevOps: a new zealand perspective', *2017 IEEE 12th International Conference on Global Software Engineering (ICGSE)*, pp.21–30, doi: 10.1109/ICGSE.2017.16.

Hussaini, S.W. (2015) 'A systemic approach to re-inforce development and operations functions in delivering an organizational program', *Complex Adaptive Systems*, Elsevier Masson SAS, pp.261–266, doi: 10.1016/j.procs.2015.09.209.

Jabbari, R., bin Ali, N., Petersen, K. and Tanveer, B. (2016) 'What is DevOps? A systematic mapping study on definitions and practices', *Proceedings of the Scientific Workshop Proceedings of XP2016 on – XP '16 Workshops*, pp.1–11, doi: 10.1145/2962695.2962707.

Jamous, N., Bosse, S., Gorling, C., Hintsch, J., Khan, A., Kramer, F., Muller, H. and Turowski, K. (2017) 'Towards an IT service lifecycle management (ITSLM) concept', *Proceedings – 4th International Conference on Enterprise Systems: Advances in Enterprise Systems, ES 2016*, pp.29–38, doi: 10.1109/ES.2016.10.

Jäntti, M. (2011) 'Improving incident management processes in two it service provider companies', *22nd International Workshop on Database and Expert Systems Applications, DEXA*, pp.26–30, doi: 10.1109/DEXA.2011.42.

Kamuto, M.B. and Langerman, J.J. (2017) 'Factors inhibiting the adoption of DevOps in large organisations: South African context', *RTEICT 2017 – 2nd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, Proceedings*, pp.48–51, doi: 10.1109/RTEICT.2017.8256556.

Karapantelakis, A., Karapantelakis, A., Liang, H., Wang, K., Vandikas, K., Inam, R., Fersman, E., Mulas-Viela, I., Seyvet, N. and Giannokostas, V. (2016) 'DevOps for IoT applications using cellular networks and cloud', *Proceedings – 2016 IEEE 4th International Conference on Future Internet of Things and Cloud, FiCloud 2016*, pp.340–347, doi: 10.1109/FiCloud.2016.55.

Katal, A., Bajoria, V. and Dahiya, S. (2019) 'DevOps: Bridging the gap between development and operations', *Proceedings of the 3rd International Conference on Computing Methodologies and Communication, ICCMC 2019*, IEEE, pp.1–7, doi: 10.1109/ICCMC.2019.8819631.

Kikuchi, S. (2015) 'Prediction of workloads in incident management based on incident ticket updating history', *Proceedings – 2015 IEEE/ACM 8th International Conference on Utility and Cloud Computing, UCC 2015*, pp.333–340, doi: 10.1109/UCC.2015.53.

Latrache, A., Nfaoui, E.H. and Boumhidi, J. (2015) 'Multi agent based incident management system according to ITIL', *2015 Intelligent Systems and Computer Vision, ISCV 2015*, doi: 10.1109/ISACV.2015.7105552.

Laukkarinen, T., Kuusinen, K. and Mikkonen, T. (2017) 'DevOps in regulated software development: case medical devices', *2017 IEEE/ACM 39th International Conference on Software Engineering: New Ideas and Emerging Technologies Results Track (ICSE-NIER)*, pp.15–18, doi: 10.1109/ICSE-NIER.2017.20.

Limanto, A., Khwarizma, A.F.I., Rumagit, R.Y., Pietono, V.P., Halim, Y. and Liawatimena, S. (2017) 'A study of information technology infrastructure library (ITIL) framework implementation at the various business field in Indonesia', *2017 5th International Conference on Cyber and IT Service Management, CITSM 2017*, pp.1–4, doi: 10.1109/CITSM.2017.8089244.

Liu, L., Kong, X. and Chen, J. (2015) 'How project duration, upfront costs and uncertainty interact and impact on software development productivity? A simulation approach', *International Journal of Agile Systems and Management*, Vol. 8, No. 1, pp.39–52, doi: 10.1504/IJASM.2015.068605.

Lou, J-G., Lin, Q., Ding, R., Fu, Q., Zhang, D. and Xie, T. (2013) 'Software analytics for incident management of online services: an experience report', *2013 28th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pp.475–485, doi: 10.1109/ASE.2013.6693105.

Mahanta, P., Pole, A.K., Adige, V.S. and Rajkumar, M. (2016) 'DevOps culture and its impact on cloud delivery and software development', *Proceedings – 2016 International Conference on Advances in Computing, Communication and Automation, ICACCA 2016*, pp.1–6, doi: 10.1109/ICACCA.2016.7578902.

Miles, J. and Gilbert, P. (2005) *A Handbook of Research Methods for Clinical and Health Psychology*, Oxford University Press, Oxford, p.315.

Modell, S. (2005) 'Triangulation between case study and survey methods in management accounting research: an assessment of validity implications', *Management Accounting Research*, Vol. 16, No. 2, pp.231–254, doi: 10.1016/j.mar.2005.03.001.

Mora, M., Gomez, J. M., O'Connor, R. V., Raisinghani, M. and Gelman, O. (2015) 'An extensive review of IT service design in seven international ITSM processes frameworks', *International Journal of Information Technologies and Systems Approach*, Vol. 8, No. 1, pp.69–90, doi: 10.4018/ijitsa.2015010104.

Palihawadana, S., Palihawadana, S., Wijeweera, C.H., Sanjitha, M.G.T.N., Liyanage, V.K., Perera, I. and Meedeniya, D.A. (2017) 'Tool support for traceability management of software artefacts with DevOps practices', *3rd International Moratuwa Engineering Research Conference, MERCon 2017*, pp.129–134, doi: 10.1109/MERCon.2017.7980469.

Park, H.Y., Jung, S.H., Lee, Y.J. and Jang, K.C. (2006) 'The effect of improving IT standard in IT governance', *International Conference on Computational Intelligence for Modelling Control and Automation, and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'OG)*, pp.1–6, doi: 10.1109/CIMCA.2006.210.

Pereira, R. and da Silva, M.M. (2011) 'A maturity model for implementing ITIL V3 in practice', in *Proceedings – IEEE International Enterprise Distributed Object Computing Workshop, EDOC*, pp.259–268, doi: 10.1109/EDOCW.2011.30.

Pereira, R. and da Silva, M.M. (2012) 'Designing a new integrated IT governance and IT management framework based on both scientific and practitioner viewpoint', *International Journal of Enterprise Information Systems (IJEIS)*, Vol. 8, No. 4, p.43, doi: doi:10.4018/jeis.2012100101.

Perera, P., Bandara, M. and Perera, I. (2017b) 'Evaluating the impact of DevOps practice in Sri Lankan software development organizations', *16th International Conference on Advances in ICT for Emerging Regions, ICTer 2016 – Conference Proceedings*, pp.281–287, doi: 10.1109/ICTER.2016.7829932.

Perera, P., Silva, R. and Perera, I. (2017a) 'Improve software quality through practicing DevOps', *2017 Seventeenth International Conference on Advances in ICT for Emerging Regions (ICTer)*, pp.1–6, doi: 10.1109/ICTER.2017.8257807.

Perry, D.E., Sim, S.E. and Easterbrook, S.M. (2004) 'Case studies for software engineers', *Proceedings. 26th International Conference on Software Engineering*, pp.736–738, doi: 10.1109/ICSE.2004.1317512.

Prates, L., Faustino, J., Silva, M. and Pereira, R. (2019) 'DevSecOps metrics', in Wrycza, S. and Maślankowski, J. (Eds.): *Information Systems: Research, Development, Applications, Education*, Vol. 359, pp.77–90.

Punjabi, R. and Bajaj, R. (2017) 'User stories to user reality: a devops approach for the cloud', *2016 IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, RTEICT 2016 – Proceedings*, pp.658–662, doi: 10.1109/ RTEICT.2016.7807905.

Remfert, C. (2017) 'Developing a common understanding of it services – the case of a german university', *Lecture Notes in Computer Science (including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer, Cham, pp.407–421, doi: 10.1007/978-3-319-58521-5_32.

Riungu-Kalliosaari, L., Mäkinen, S., Lwakatare, L. E., Tiihonen, J. and Männistö, T. (2016) 'DevOps adoption benefits and challenges in practice: a case study', *17th International Conference, PROFES 2016*, Throndheim, Norway, pp.590–597, doi: 10.1007/978-3-319-49094-6_44.

Roche, J. (2013) 'Adopting DevOps practices in quality assurance', *Communications of the ACM*, pp.38–43, doi: 10.1145/2524713.2524721.

Rong, G., Gu, S., Zhang, H. and Shao, D. (2017) 'DevOpsEnvy: an education support system for DevOps', *2017 IEEE 30th Conference on Software Engineering Education and Training (CSEE&T)*, pp.37–46, doi: 10.1109/CSEET.2017.17.

Saarelainen, K. and Jäntti, M. (2016) 'Quality and human errors in IT service infrastructures – human error based root causes of incidents and their categorization', *Proceedings – 2015 11th International Conference on Innovations in Information Technology, IIT 2015*, pp.207–212, doi: 10.1109/INNOVATIONS.2015.7381541.

Shahin, M. (2015) 'Architecting for DevOps and continuous deployment', *Proceedings of the ASWEC 2015 24th Australasian Software Engineering Conference on – ASWEC '15*, Vol. II, pp.147–148, doi: 10.1145/2811681.2824996.

Sharma, S. and Coyne, B. (2014) *DevOps for Dummies*, 2nd ed., IBM (Ed.), John Wiley & Sons, Inc., Hoboken.

Shivakumar, S.K. (2017) *DevOps for Digital Enterprises Brief Introduction to DevOps Scope*, Bengaluru, India, Retrieved from https://www.infosys.com/digital/insights/documents/ devops-digital-enterprises.pdf

Silva, M.A., Faustino, J., Pereira, R. and Silva, M.M. (2018) 'Productivity gains of DevOps adoption in an IT team□: a case study', *27th International Conference on Information Systems Development*, Available at: https://repositorio.iscte-iul.pt/handle/10071/16388

Soni, M. (2016) 'End to end automation on cloud with build pipeline: the case for DevOps in insurance industry, continuous integration, continuous testing, and continuous delivery', *Proceedings – 2015 IEEE International Conference on Cloud Computing in Emerging Markets, CCEM 2015*, pp.85–89, doi: 10.1109/CCEM.2015.29.

Stokburger-Sauer, N.E., Scholl-Grissemann, U., Teichmann, K. and Wetzels, M. (2016) 'Value cocreation at its peak: the asymmetric relationship between coproduction and loyalty', *Journal of Service Management*, Vol. 27, No. 4, pp.563–590, doi: 10.1108/JOSM-10-2015-0305.

Stoneham, J., Thrasher, P., Potts, T., Mickman, H., DeArdo, C. and Limonchelli, T.A. (2017) *DEVOPS CASE STUDIES: The Journey to Positive Business Outcomes*, 1st ed., IT Revolution Press, Oregon, Portland.

Suomalainen, T. and Xu, Y. (2016) 'Continuous planning through the three horizons of growth', *International Journal of Agile Systems and Management*, Vol. 9, No. 4, p.269, doi: 10.1504/IJASM.2016.081556.

Suomalainen, T., Kuusela, R. and Tihinen, M. (2015) 'Continuous planning: an important aspect of agile and lean development', *International Journal of Agile Systems and Management*, Vol. 8, No. 2, p.132, doi: 10.1504/IJASM.2015.070607.

Tan, W-G., Cater-Steel, A. and Toleman, M. (2010) 'Implementing IT service management: a case study focussing on critical success factors', *Journal of Computer Information Systems*, Vol. 50, No. 2, pp.1–12.

Tellis, W. (1997) 'Information technology in a university: a case study', *Campus-Wide Information Systems*, Vol. 14, No. 3, pp.78–91, doi: 10.1108/10650749710187617.

Thomas, G. (2016) *How to do Your Case Study*, 2nd ed., Seaman, J. and Piper, J. (Eds.), Sage Publications Asia-Pacific PTE LTD, Singapore.

Uikey, N. and Suman, U. (2016) 'Tailoring for agile methodologies: a framework for sustaining quality and productivity', *International Journal of Business Information Systems*, Vol. 23, No. 4, p.432, doi: 10.1504/IJBIS.2016.080216.

Verma, R. (2000) 'An empirical analysis of management challenges in ervice factories, service shops, mass services and professional services', *International Journal of Service Industry Management*, Vol. 11, No. 1, pp.8–25, doi: 10.1108/09564239810199923.

Wahaballa, A., Wahballa, O., Abdellatief, M., Xiong, H. and Qin, Z. (2015) 'Toward unified DevOps model', *6th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, pp.1–4, doi: 10.1109/ICSESS.2015.7339039.

Wang, Q., Song, J., Liu, L., Luo, X. and XinHua, E. (2010) 'Building IT-based incident management platform', *ICPCA10 – 5th International Conference on Pervasive Computing and Applications*, pp.359–364, doi: 10.1109/ICPCA.2010.5704127.

Wang, Q., Zhou, W., Zeng, C., Li, T., Shwartz, L. and Grabarnik, G.Y. (2017) 'Constructing the knowledge base for cognitive IT service management', *2017 IEEE International Conference on Services Computing (SCC)*, pp.410–417, doi: 10.1109/SCC.2017.59.

Webster, J. and Watson, R.T. (2002) 'Analyzing the past to prepare for the future: writing a literature review', *MIS Quarterly*, Vol. 26, No. 2, pp.xiii–xxiii, doi: 10.1.1.104.6570.

Yin, R.K. (2009) *Case Study Research□: Design and Methods, Applied Social Research Methods Series*, doi: 10.1097/FCH.0b013e31822dda9e.

Yun, M., Lan, Y. and Han, T. (2017) 'Automate incident management by decision-making model', *2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA)*, pp.217–222, doi: 10.1109/ICBDA.2017.8078811.

Zaidah, Z. (2007) 'Case study as a research method', *Jurnal Kemanusiaan*, pp.1–6, doi: 10.1177/15222302004003007.

Annex 1

Practice name	*Description*
Shift left	This practice refers to include operations as early as possible on the SDCL
Continuous planning	Business owners will see the growth of the application, so they can give feedback on whether the application is corresponding to their needs
Continuous integration (CI)	The developers will check in their code on the source control repository and integrate it with the code from other teams, allowing CI
Feedback loops between Dev and Ops	The goal of this practice is to get as much feedback as possible to perform the necessary corrections
Automated monitoring	Allows a better perception of the health of the system. This will allow continuous monitoring of the application
Prototyping application	This will give a better idea of what requirements are needed for the application, reducing time on redesigned requirements
Deployment automation	These tools facilitate by managing the software components that need to be deployed and what middleware components and configurations need to be updated. This will allow continuous deployment
Test automation	Test automation will save some time by performing regression tests to be sure that older functionalities will not be impacted by new developments. This will also allow a continuous testing approach
Infrastructure as code	Allows the organisations to manage which environments need to be provisioned and configured to enable continuous delivery
Stakeholder participation	The participation of stakeholders will provide more feedback to the DevOps teams
Process standardisation	By standardising the processes, they will be perfected over time by identifying errors and correcting them
Change management	Process for the efficient handling of IT changes

Appendix A

Practices	*Rates* 1	2	3	4	5	*Average*	*Comments*
Continuous planning		3	1	2		2.8	"It was hard to define which ceremonies should be part of this practice"
							"The major challenge of this practice was the definition of dates to book the meetings"
							"The attendees from the business could not realise of the benefits of these meetings, so, they came reluctant to the implementation of this practice"
							"There was a lot of effort to create the template documentation for support for these ceremonies"

Appendix A (continued)

	Rates						
Practices	*1*	*2*	*3*	*4*	*5*	*Average*	*Comments*
Feedback loops between Dev and Ops	1		2	1		2.75	"We felt a lot of lack of engagement from the Ops teams to participate in on our meetings. Maybe due to the different consultant teams between the Development teams and Ops team. After showing this to the client responsible, the Ops, started to accept to attend the meetings and we were able to seem more engagement from their side"
							"Challenges due to busy agendas and different time zones between Dev and Ops"
							"The knowledge transfer sessions that exists between Dev and Ops take a lot of time from the attendees that could be used to perform their tasks"
Continuous integration			2	4	2	4	"Easy to implement with the correct tools. Also, easy to understand its benefit"
							"Easy process to be used"
Automated monitoring	1		2			2.3	"Hard to maintain due the continuous delivery"
							"Lack of interest from the project management. The project management is focussed in fixing incidents and doesn't want to waste the time of their resources in maintain these automated monitors"
							"Requires some time to build these monitors and know what should be monitored"
Deployment automation		2	3	1	1	3.1	"The automated deployment was complex to use due to the lack of knowledge of the team about the tools"
							"It was hard to use due to the lack of the debugging tools to find why the deployment fails"
							"The use is hard due to the lack of knowledge on how to configure this kind of tools, but easy to use"
Test automation	1	1	2			2.25	"It is hard to configure the tools to the system"
							"Hard to configure and to maintain due to the continuous delivery"
Stakeholder participation	2		3			1.6	"Lack of engagement from the stakeholders to participate in some decision processes"
							"There are some challenges to break the barrier between the technical and functional language. In these meetings it is needed some stakeholders that can make this bridge"

Appendix A (continued)

Practices	*Rates* 1	2	3	4	5	*Average*	*Comments*
Process standardisation		1	1	1		3	"The project management encourages to do this, since it will reduce the mistakes on the process execution"
							"It takes a lot of time to achieve the perfection on the process. Every time that someone makes a mistake on the process, it needs to be redesigned"
Change management		1	2	4		3.4	"On the change management process there are a lot of people involved. The communication between all these people is not easy"

Appendix B

Practices	*Rate* 1	2	3	*Average*	*Comments*
Shift-left					
Continuous planning			6	3	"Creates more interaction between the development teams and the business, creating better relationships"
					"With continuous planning we are able to take care of the incidents already discussed with the business, saving time on the analysis"
					"With continuous planning it will be possible to always have scope to deliver on next releases, which will leave to the decrease of incidents"
					"It will guarantee that the delivery will be what the business has requested, avoiding wastes of time to code the functionality over again"
					"It helps the team to define objectives and keep their focus on the tasks"
					"With this practice, we can have a well-defined scope from the business, so we were able to keep our focus on the scope that was defined"
Feedback loops between Dev and Ops		2	2	2.5	"It helps to prevent issues, but it takes a lot of time on meetings"
					"Due to a better relationship between Dev and Ops, errors may be found earlier and even avoided"
					"The team will have a broader knowledge, where the team members will be more autonomous"

Appendix B (continued)

Practices	Rate 1	2	3	Average	Comments
Continuous integration			8	3	"It will ensure the code merge, avoiding having effort on merging activities"
					"Prevent errors on merging and facilitates the alignment between teams"
					"Guarantee the code integration saving time from merging activities"
					"Saves time on merging activities with other teams, avoiding errors of overwritten code"
					"Avoids errors and wasting time on merging activities"
					"It facilitates the code versioning, avoiding mistakes on merging"
					"It allows to have updated environments and ready to work"
					"Our productivity was increased due to the reducing of the merging tasks that we did before"
Automated monitoring			3	3	"Reduces time on monitoring activities"
					"Saves time"
					"Saves time on performing manual monitors that were being done, and allows to find errors and react on time"
Deployment automation		2	5	2.71	"Reduces time on deployment activities and there are no worries on creating manual packages"
					"Reduces the human error and ensure the correct deployment"
					"When it goes fine it can save a lot of time, but when it goes wrong it may result on a big amount of time to understand why"
					"It improves the deployment process, mitigating the human error"
					"Saves time but it can be painful to analyse in case of errors"
					"Allows better release management in terms of time consumed by the deployments"
					"We spent a lot of time on supporting the deployments, but with this practice, we almost don't spend any of our time on supporting these tasks"
Test automation		4		2	"Even when automated tests are made, we are always needed to perform unit and integrated tests"
					"It may have false positives and when this happens, the developers waste a lot of time to find the false error"
					"Tests will be executed faster, but in case of false positives it may result on a big waste of time"
					"Even this practice was in place, our process also includes unit testing, so the developer also needs to take time to perform these tests"

Appendix B (continued)

Practices	*Rate* 1	2	3	*Average*	*Comments*
Stakeholder participation		1	3	2.75	"Helps to understand the business needs"
					"It can improve the performance if all the stakeholders that are involved on the discussion are interested on the topic"
					"Having business present while fixing a functionality it will give us the right path to follow, instead of trying to fix something which can lead to another error"
					"The participation of all the stakeholders is important because the accountability of the decisions can be distributed. So, every time that we need to have a decision from the business, we don't need to be reduced a small amount of points of contact"
Process standardisation	1		2	2.33	"It makes the development process easier since the process will be the same for everyone"
					"It requires a lot of time and efforts to define the processes "
					"The standard processes make the team to work on the same way for all the processes, there the space for errors will be reduced over the time"
Change management	1	2	3	2.33	"It's essential, but we lose a lot of time on requesting to the other participants on the change process to do their tasks"
					"It wastes some time to ensure that the right participants are doing their tasks"
					"All the participants need to be careful on their tasks and do them with the maximum of attention"
					"Guarantees that all the changes follow the a restrict quality control, contributing to the quality of our delivery"
					"It guarantees that all the quality procedures have been done, to avoid errors"
					"The change management process will guarantee that our deliver will follow all the quality standards that are imposed by our client. However, the change management process that we follow today, takes some time since that are required a lot of participants to act, which also impact our timings"

Appendix C

Why	
W1	"Continuous planning helps the business to know what needs to be fixed and the negative impact that is causing"
	"It's important to register and centralise incidents to identify the ones that affect multiple users to execute daily work"

Appendix C (continued)

Why	
W2	"During the planning, business will show their needs and will expose the next priorities"
	"Feedback from the client will help on establishing a correct classification providing an initial support"
	"The correct incident classification in terms of severity and priority allow to better select mandatory incidents"
W3	"Feedback provided while planning and selecting the next priorities will elk on the investigation"
W4	"By providing feedback from both teams with different perspectives, the issues can be found and analysed easily"
	"Both teams will have different approaches on how to solve the issues generating brainstorms and a better cohesion"
	"Sharing knowledge will maximise the capabilities for both teams, contributing for a faster analysis"
W5	"All the code will be easily merged, facilitating its diagnosis"
	"With CI it is possible to keep a track of the changed code, which will be easier to find the person who change it (in case that person is still working at the company) to know why the code was changed on that way, since that person could have different thinking on how the functionality should work"
	"Having all the code integrated on the last version and ready to be deployed in any environment may help on replicating incidents, avoiding misalignments between lower environments and production environment"
W6	"By always working on the last version of the code will help to find the resolutions"
	"Resolution of the code will be easier, and the code may not be smashed"
	"By using the latest code version, we will be working on the last software version, finding the resolution faster"
	"Since all the integrated code is easier to evaluate the impacts of the resolution, thereby accelerating the resolution"
W7	"It allows to detect any problems originated by teams" parallel developments"
W8	"Automated monitors are useful to check the health of the system detecting incidents"
W9	"Constant monitoring of the system to find if the incident was solved"
	"Helps to find issues and to guarantee that the fixes are working"
	"Automated monitoring will save time for everyone and find issues in production"
	"Can help on checking the system health and if the incidents were indeed fixed"
	"The automated monitor will check if the system is ok; this way it will also monitor if the fix for the incident was successful"
W10	"It will show a proposal of the solution and final behaviour, so the business can accept it faster"
	"It will guarantee that the solution is what the business is expecting"
	"By having a prototype of the fix of the functionality, we can show the result to the users, so they can approve the resolution or tell us what is supposed to be the final result"

Appendix C (continued)

Why	
W11	"Automated deployments will allow more deployments for several test environments, accelerating the user tests to approve the resolutions"
	"The deployment automation may help on the resolution, accelerating the fixes deployment for other environments, finding issues that those fixes may cause in production"
W12	"Incidents may be closed faster since more deployment windows are available"
	"Due to the time saving there will be more deployments, which will give the possibility to install hotfixes more often"
	"To allow deploying release and related incidents/change requests without manual process it will help on the closure of the incident"
W13	"Reduce the manual tests performed to the build solution and application stability"
W14	"If we receive an incident that we guess could fail on an automatic test, we can run that test and check where it fails, giving an idea of where the issue can be found"
W15	"By knowing the final result, it's possible to design the automatic test, saving time, instead of doing the manual test; therefore, the resolution will be found earlier"
	"Automated tests to perform integrated tests to check if the resolution doesn't impact other functionalities"
W16	"By executing automated tests, we can find if the incidents are fixed or not"
W17	"Environments can be easily provisioned to have all the needed components"
	"It will help to have environments ready for the analysis of incidents"
	"This practice will help with having the environments work for the necessities of the developers, helping to analyse the incidents"
W18	"Communicating with the stakeholders will aid in understanding the real impacts and issues that one incident is causing"
	"Will help the business to understand how the functionalities are working and create incidents if needed"
	"By discussing with the business, we can understand if the functionalities are correctly implemented and if there is a misalignment, an incident should be created"
	"Helps understand what the real requirement was and what was implemented"
W19	"By providing feedback to the business, they can categorise the incidents correctly"
	"We can help the business to evaluate the impact of an issue, so it can have a better prioritisation"
	"Due to the stakeholder participation, it is possible to have an initial support in order to help the business in order to understand if there is an issue or not"
W20	"By evolving all the stakeholders, including technical stakeholders, not only the business, it may help on the investigation phase by contributing with other knowledge areas"
	"Business users may help replicate the issues facilitating the analysis"
	"The functional knowledge of the business may be a great plus on investigating the root cause of the incidents"
	"Having businesses participate in the investigation and diagnosis will help to find the root cause for the incidents and finding if the software is working as it was designed. From this we might get two different conclusions: there is no issue and there was an error from the user when interpreting the result of the functionality, or a Change Request may be raised to change the functionality design"

Appendix C (continued)

Why	
W21	"Having standardised processes on how to report incidents will help the users to report incidents properly"
	"In order to report incidents with necessary detail to allow identify the root cause"
W22	"Implement processes to evaluate impacts in order to have a better prioritisation"
	"Having processes to define priorities"
W23	"Having procedures to report incidents properly will help on the diagnosing the incidents"
	"Implementing processes on how to replicate certain behaviours may help on diagnosing the incidents"
W24	"Standard processes may help on the incident resolution facilitating what should be done to progress with the solution that was made while diagnosing the incident"
W25	"Important to detect any undesired effect in the system due to implemented changes"
W26	"Helps with guaranteeing process to deliver a change into production"
	"Manages all the process of the change reducing the impacts that may cause"
	"By being a rigid process, it certifies that the change is in condition to go to production"
	"This process will evaluate the required change to fix the incident, minimising the impact that may cause on the application health"
	"Production/lower environments application changes and incident closure should follow defined process/rules"
W27	"It allows to collect better environment interventions and allocate resources for implementing them"
How	
H1	"Promote planning meetings with the business"
	"Use the Agile ceremonies: Spring Planning, Sprint Retrospective and Sprint Review. Even if the goal of retrospective and review is not planning, it will help to understand the status of the application and the remaining incidents that need to be fixed; therefore, it needs to be prioritised"
	"Regular meetings with the business"
	"Promoting business meetings and discussing the priority incidents to be addressed in following releases"
H2	"By perform quick analysis of the issue reported and affected portfolio"
H3	"Promote knowledge transfer sessions"
	"Consider inviting operations for discussions when analysing incidents"
	"Having more sessions between Dev and Operations"
H4	"Having tools to enable this"
	"TFS and Jenkins are good tools to do this"
	"Having tools that facilitate this integration"
	"The Version Control Software should be able to integrate with a build software"
	"Tools should be used to enable this, like TFS and Jenkins"
H5	"By updating main source code repository and refreshing lower environments"

Appendix C (continued)

How	
H6	"Scripts that can be executed and produce reports"
	"Having tools that trigger alerts when something is wrong with the system"
	"Having dashboards that are automatically refreshed in time to time to detect something wrong with the system"
	"Having reports that are generated automatically are very useful to evaluate the system health"
	"Scripts that are executed everyday generating reports checking the system health"
H7	"Having environments that are not used to analyse incidents but just to install the solutions, so the users can see the final results"
	"Having environments with similar data as production so the users can test the solutions"
	"Lower environments with production data"
H8	"Having tools that can deploy the changes without user action"
	"Tools to enable the automatic deployment for several environments"
	"Tools that deploy changes that are needed"
	"By implementing automation process and reducing human error"
H9	"Executing regression tests and programming specific tests scenarios"
H10	"Having a tool that allow us to provide the final result so that the tool can follow several flows in order to reach that result"
	"Having testing tools that can test several modules of the application at the same time"
	"Having tools where we can insert break points in order to check the flow of the test"
	"Test tools that can make the tests based on final outputs provided by the business to check if the functionality is working as it is supposed to, confirming that the incident was solved"
H11	"Using cloud environments"
	"Cloud environments are an enabler for this"
	"Having scripts and tools that can configure the environments quickly"
H12	"Have regular meetings with technical and functional stakeholder to discuss the health of the system so it can help on diagnosing issues and finding new issues"
	"Involving business on the incident analysis and asking them questions when we find something that looks wrong"
	"By booking meetings to discuss the incident status and ask for help to replicate"
	"Book regular meetings to provide statuses of the most urgent incidents. This way business will participate in case of any doubt that we may have"
H13	"By trying to get involved with the business to help"
	"Having prioritisation meetings with the business"
	"Due to the stakeholder participation, it is possible to have an initial support in order to help the business to understand if there is an issue or not"
	"Prioritisation meetings are needed where the root cause of the incidents is explained and how is impacting the application, in order to have better prioritisation"
H14	"Having a report document template that the business should use when reporting incidents"
	"By defining templates to report mandatory information and this way facilitate root cause identification"

Appendix C (continued)

How	
H15	"Create an incident prioritisation matrix comparing impacts vs. affected people"
	"Having templates with the parameters that should be considered when prioritising incidents"
H16	"Include steps to reproduce when reporting incidents"
	"Setting the steps to reproduce the incidents"
	"Document all the process since the investigation until having the change in production, so everybody can follow the same process"
H17	"Validating the outputs and implementing rollback tasks if needed"
H18	"Have a checklist to check if the change is following the right path"
	"Follow the process step-by-step in order to reduce the impacts"
	"Define the correct path that this process should follow or consider having a software that already has this kind of process"
	"Have the process well defined. However, due to the changes of other processes or teams, this process may need to be redefined. It is needed to adapt this process to all other changes around on the company"
H19	"By requesting action requests/changes to responsible teams"
H20	"By planning in advance future releases in the system"

Appendix D

Practice	*No. of matches*	*Quotes*
Continuous planning	6	"To receive *feedback* from the client as soon as possible in order to enhance incident management/resolution if required"
		"Showing the progress of developments to the business to check if a re-plan is needed"
		"Plan in medium-long time to guarantee a continuous delivery"
		"There were implemented some meetings to re-prioritise the incidents in case of need"
		"Due to the changes of requirements due to the developments"
		"Meetings are made to consider the most critical incidents on the pipeline to be solved"
Feedback loops between Dev and Ops	4	"To *mitigate* errors on deployment activities and enhance recovery activities"
		"To guarantee a better *alignment* between teams"
		"Getting *feedback* from other teams"
		"There are knowledge transfer sessions between the Dev's and the Ops where the dev's share their new developments; so, the ops could share their concerns on how these developments may *impact* the software"

Appendix D (continued)

Practice	*No. of matches*	*Quotes*
Continuous integration	8	"To facilitate the process of having teams working simultaneous on the same application"
		"It helps the development since the developers will always work on the latest software version"
		"Developing the most recent code version allows us to find the errors easily"
		"To keep the integrity to decrease the amount of errors to ensure the *quality* of the software"
		"Due to the increase of deliveries by all the teams it's needed to have all the code integrated to avoid that the code gets overwritten and guarantees the *alignment* between teams"
		"To guarantee all the integration of the software between teams, to avoid merge issues"
		"Allows the team to work on the latest code version, avoiding merge issues"
		"Allows the integration of the most recent code in lower environments, guaranteeing that the team is working on an environment with the most recent code"
Automated monitoring	5	"To monitor system health"
		"It verifies the system health before, during and after the deployments"
		"Saves time and finds new issues"
		"Saves time and find issues introduced by new software deliveries or middleware issues, ensuring *quality*"
		"Finds issues in preliminary stages causing less *impact* to businesses"
Deployment automation	6	"*Mitigates* human error and the process becomes standard"
		"Saves time for the developers by deploying their changes to test environments"
		"Saves time and makes a standard process that everyone will follow"
		"Helps on the deployment reducing human error"
		"*Mitigates* the human error"
		"Saves time and *mitigates* human error"
Test automation	5	"*Mitigates* the risk of breaking existing functionalities"
		"So, the regression tests can be done in a more severe way"
		"More *quality* on testing"
		"Guarantees a rigid regression test plan verifying that the new developments will not result in new errors on the software"
		"Regression tests are made to guarantee the *quality* of the solution"

Appendix D (continued)

Practice	*No. of matches*	*Quotes*
Stakeholder participation	3	"Provides continuous *feedback* of the existing processes"
		"Helps on understanding the needs of the business"
		"Helps to guarantee that everything is delivered as intended"
		"Guarantees that the stakeholders are aware of the status of the application, to know what the most critical issues that need to be solved"
Process standardisation	3	"Defining rules to be followed by everyone"
		"Guarantees that everyone will follow the same process"
		"Implementing standard processes will make sure that everyone will follow the same process, reducing errors"
Change management	6	"To guarantee *quality* on the Software Delivery"
		"To be sure that *quality* control process is made to register the software changes"
		"To guarantee that the code change follows all the defined steps of the *quality* control process"
		"This process helps minimise the *impact* of the change"
		"Process that follows all the code change to ensure that will not cause other issues and guarantees that the problem will be solved"
		"All the deployments are address by following the same rules"

Appendix E

1 Do you know the DevOps methodology? If yes, please give a brief description?

2 Which DevOps practices do you know?

3 Which DevOps practices do you apply/applied?

4 Why have you applied these practices?

5 When have you started to apply these practices?

6 On the beginning of the implementation of these practices, they were applied by all members or just a few? Why?

7 How was the adoption of these practices?

8 Have these practices improved your team performance?

9 How did you do before applying the practices?

10 In which phases of IM do you apply the following practices? How do you apply? Is there any practice that doesn't apply to IM?

Appendix E (continued)

11 Would you like to apply any of the following practices that you don't use today? If so in which IM phases?

12 Would you like to suggest any practice that wasn't identified here?

13 In which other ITIL processes, DevOps would make sense to be applied? In your opinion, how?

information

Review

An IT Service Management Literature Review: Challenges, Benefits, Opportunities and Implementation Practices

João Serrano [1], João Faustino [1,*], Daniel Adriano [1], Rúben Pereira [1] and Miguel Mira da Silva [2]

1 Department of Information Science and Technology, Instituto Universitário de Lisboa (ISCTE-IUL), 1649-026 Lisboa, Portugal; jpsso11@iscte-iul.pt (J.S.); daniel_adriano@iscte-iul.pt (D.A.); Ruben.Filipe.Pereira@iscte-iul.pt (R.P.)
2 Instituto Superior Técnico, University of Lisbon, 1049-001 Lisboa, Portugal; mms@tecnico.ulisboa.pt
* Correspondence: jpcfo11@iscte-iul.pt

Abstract: Information technology (IT) service management is considered a collection of frameworks that support organizations managing services. The implementation of these kinds of frameworks is constantly increasing in the IT service provider domain. The main objective is to define and manage IT services through its life cycle. However, from observing the literature, scarcely any research exists describing the main concepts of ITSM. Many organizations still struggle in several contexts in this domain, mainly during implementation. This research aims to develop a reference study detailing the main concepts related with ITSM. Thus, a systematic literature review is performed. In total, 47 articles were selected from top journals and conferences. The benefits, challenges, opportunities, and practices for ITSM implementation were extracted, critically analysed, and then discussed.

Keywords: IT service management; systematic literature review; benefits; challenges; opportunities; implementation practices

check for updates

Citation: Serrano, J.; Faustino, J.; Adriano, D.; Pereira, R.; da Silva, M.M. An IT Service Management Literature Review: Challenges, Benefits, Opportunities and Implementation Practices. *Information* **2021**, *12*, 111. https://doi.org/10.3390/info12030111

Academic Editor: Willy Susilo

Received: 24 January 2021
Accepted: 1 March 2021
Published: 5 March 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

Nowadays, for most companies, information technology (IT) is one of the most important assets in the organizations' infrastructure. In fact, from early 1980s until now, the utilization and continuous improvement of IT has become an essential support to the business, regardless of the sector that organization belongs to [1]. This dependency on IT has been growing, as has its complexity, forcing organizations to increasingly have an effective management [2]. The management of IT operations has received more attention given its relevance on IT costs [3].

IT services have a big impact on the competitive advantage [4], being important to have an effective and efficient management. To do it, many IT organizations invest in IT service management (ITSM) frameworks such as the IT Infrastructure Library (ITIL) or the Control Objectives for Information and Related Technologies (COBIT) [5,6]. ITSM concentrates on IT operations, in particular in service delivery and support [7]. This "portfolio" of frameworks can produce several benefits to IT organizations, by helping them creating strategies and impelling fundamental changes, in order to become more adaptive, compliant and profitable [8].

If well implemented, ITSM frameworks can produce many benefits to organizations, such as decreases in operational costs and increases in operational efficiencies [5]. Some studies point that as process maturity levels grow, more benefits and lower issues organizations will face. These include a positive impact in business performance [9], an increase of organization profitability [10] and competitive leverage [11,12].

However, despite the benefits, many challenges have also been pointed out. Some claim that these frameworks are not easy to implement [13–15], criticizing the substantial complexity of the models and its abstraction in terms of knowledge and content [13,16,17], not generating the benefits that are expected. Additionally, due to the fact that organizations need to implement different frameworks and standards to mitigate the difficulties and needs that only one standard or framework cannot address [18], the practices overlap each other [19,20]. Such overlap turns into an important issue since it increases organizations' costs, time, and resources [21,22]. These challenges generate opportunities for new research.

Based on previous paragraphs one may conclude that despite the proven benefits, challenges also exist regarding ITSM implementation, but new opportunities may arise with most recent versions of ITSM frameworks. Also, due to the evolution of new methodologies or cultures adopted by IT teams, ITSM frameworks need to be adapted to be able to work with these methodologies and cultures [23–25]. Thus, this suggests opportunities for improvement and adaptations to ITSM frameworks. So, based in previous statements the authors have reached to four concepts that are intrinsically connected in ITSM domain: challenges, benefits, opportunities, and implementation practices. It is possible to state from the existing literature that organizations may benefit from ITSM implementation; challenges can also be faced, which in turn can generate new opportunities.

Despite several investigations into the ITSM domain, and more organizations adopting it, the literature lacks a centralized consensus on benefits, challenges, and opportunities of ITSM implementation. This gap suggests an opportunity for a research to provide to practitioners the opportunity to know what to expect from an ITSM implementation, in terms of benefits, challenges and future opportunities. The present research adopted a systematic literature review (SLR) to review not only the ITSM concept but also to gather information about these four concepts in the ITSM area and their relations. This research intends to build solid foundations on these domains to assist further investigation in the ITSM domain. By doing so, this research can support organizations that intend to implement ITSM. This article centralizes all the findings of each concept, found in the SLR.

The present research is structured as follows. In Section 2, the contextualization of the IT Service Management concept is realized. Afterwards, in Section 3 the methodology adopted to perform the SLR is presented. Section 4, outlines all the filtration processes adopted by this research. In Section 5, some characteristics of the final article set are demonstrated. Then, in Section 6, all findings relating to each concept are presented. Subsequently, in Section 7, a discussion of each concept is realized. Finally, conclusions and future work are outlined in Section 8.

2. IT Service Management

The term service emerged in the 1930s when the U.S. Department of Commerce defined this concept to characterize economic sectors [7]. However, the notion of service and service management in the IT area just started to appear in the 1980s when the IT systems and all the IT environment increased in complexity [19]. IT services can be defined as a group of "tasks" [26] provided by an IT system or an IT department [27], that is, IT service can be characterized as the application of specialized capacities on IT assets [28]. IT services have a big impact on IT organization budget, being estimated that the maintenance and operability of these services are between 80–90 percent of the total cost attributed to IT resources [4]. With the purpose of managing IT services, over years, IT organizations became interested in ITSM frameworks [29], since these frameworks are "specific services-oriented best practices" [6].

The central focus of ITSM is the management of IT services [30]. In the earlier 1990s, with the increase in IT systems complexity, IT management became one of the main and more widely regarded functions in organizations. It was subsequently perceived that comprehensive management of IT was needed [31]. ITSM emerged from two main events: when the British Government developed the Information Technology Infrastructure Library (ITIL) framework, with the purpose of having better management and lower IT service costs [31], and with the creation of the service level management (SLM) framework [3]. Afterwards, with the demand and evolution of the IT domain, many other standards and frameworks were developed [32].

As mentioned before, the ITSM domain "has been developed to define, manage and deliver IT services" [33]. It is supposed that ITSM can be seen as a subset of Service Science [34] that concentrates on IT Operations [30]. ITSM can be defined as "(. . .) an IT management framework that promotes service-oriented best practices to deliver value to organizations" [35], providing a methodical approach to the management of an IT service life cycle, from design, implementation, operation to continual improvement [30]. Not restricted to the operational costs, ITSM has also a great focus on the costs of the whole service life cycle [6].

ITSM models and standards have been created for the implementation and evaluation of processes [32], with ITIL being the most adopted framework by IT organizations [17,27,34]. Nevertheless, other frameworks and standards have been developed and applied such as the Microsoft Operation Framework (MOF), the Capability Maturity Model Integration for Services (CMMI-SVC) or even the ITSM main standard ISO/IEC 20,000 [36].

Several studies exist in literature approaching solutions for some of the ITSM challenges. As an example, Pereira et al. [37] have elaborated a maturity model for the Incident Management process by merging multiple frameworks and eliminating overlapped practices to mitigate said issues. Another example is the method created for implementing the ITIL framework, based on process management and simulation [38]. This method partially or completely supports organizations implementing the ITIL framework, serving as a solution for the implementation problem that organizations have.

3. Research Method

To enlighten the ITSM domain, this chapter details how the SLR methodology was performed. With all the selected articles, an overview of the ITSM area is presented. On the final section, an SLR summary is performed.

A literature review (LR) can be seen as an approach to evaluate and review the state of a certain domain [39]. The SLR is a methodology to conduct a rigorous and accurate LR, providing a transparent and reproducible protocol so other authors can extend the review or to reach to the same or similar results [40,41]. This methodology is considered as a "systematic, explicit and reproducible method" with the purpose of "identifying, evaluating and synthesizing the existing body of completed and recorded work produced by researchers, scholars, and practitioners" [42]. By adhering to this kind of methodology, the LR quality will be improved in several ways [43].

With the intention of reviewing the ITSM main concepts and seeking its benefits, challenges and opportunities, an SLR was conducted. This review will help new researchers to have an easily accessible unification of ITSM information and might be a starting point for investigators to develop new solutions for opportunities or challenges found. This research is in accordance with the guidelines for conducting an SLR created by Kitchenham [44], as can be seen in Figure 1.

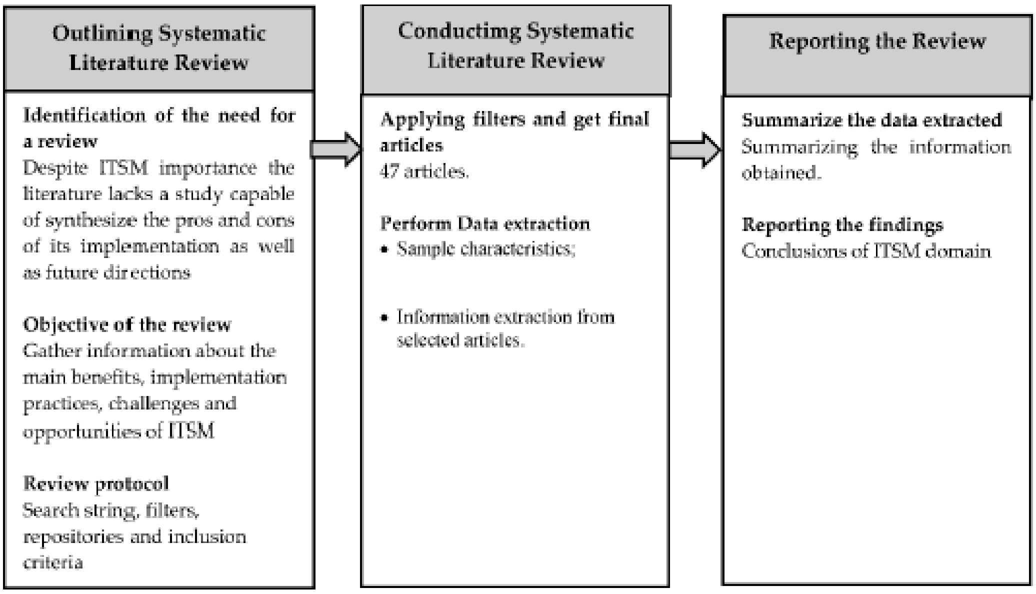


Figure 1. Systematic Literature Review Steps.

4. Outlining Systematic Literature Review

This SLR has not only the objective to perform an ITSM overview but in addition to detail all the challenges, benefits, opportunities, and implementation practices that companies are currently facing. More precisely, this research intends to answer the following research questions:

- RQ.1: What are the main ITSM benefits?
- RQ.2: What challenges organizations have been struggling with during the implementation of ITSM?
- RQ.3: What research opportunities exist in the ITSM field?
- RQ.4: What are the best implementation practices to implement the ITSM framework?

To proceed with the SLR and collect proper information to pursue the established objective, answering the formulated research questions, five electronic repositories were selected:

- IEEE Online Library (https://ieeexplore.ieee.org/Xplore/home.jsp (accessed on 4 March 2021));
- Google Scholar (https://scholar.google.com/ (accessed on 4 March 2021));
- SpringerLink (https://link.springer.com (accessed on 4 March 2021));
- Elsevier (https://www.sciencedirect.com (accessed on 4 March 2021));
- ACM (https://dl.acm.org (accessed on 4 March 2021)).

Next, specific keywords in conjunction with the AND and OR operators were adopted. The search string is presented below.

"IT Service Management" AND

("Frameworks" OR "Best Practices" OR "Standards" OR "Main Concepts" OR "Difficulties" OR "Benefits" OR "Advantages" OR "Positive effects" OR "Barriers" OR "Disadvantages")

The review included only English and Portuguese articles and accepted articles exclusively published on Journals, Scientific Magazines, and Conferences Proceedings. Lastly, no date filter was used.

In the first phase, with the selected keywords in each repository, without any filter, a search was carried out. After that, four filters were created. Since each of the electronic libraries uses different "search approaches", a keyword adaptation for each repository was used. It should be mentioned that in Google Scholar, considering that it does not have a filtering option, and that it is not possible to download all the results at once, each article was added to the Atlas support tool manually.

The description of the filters applied in this SLR is as follows: the 1st filter applies the keywords on the article title, on the abstract or on the author keywords; The 2nd one has the objective of removing duplicate articles; with the 3rd filter, articles that are published in lower ranked publications/journals are removed. For that reason, two websites, Scimago (Scimago website: https://www.scimagojr.com/ (accesse on 4 March 2021)) and Conference Ranks (Conference Ranks website: http://www.conferenceranks.com/ (accesse on 4 March 2021)), were used. Each respectively provide journal and conference ranks. For conferences, only A, B, A1, A2, B1, and B2 ranks of ERA and Qualis rankings were accepted. When an article was assessed by both rankings, the Qualis ranking prevailed. For journals only Q1 and Q2 ranks were accepted. Finally, the last stage of filtration was done by assessing article introduction and conclusion. Since this last filter had a "subjective judgement", the following criteria inclusions were followed:

- Only articles about the ITSM theme were accepted.
- Only articles about benefits, good practices, disadvantages/problems and opportunities of the ITSM area were accepted.

5. Conducting a Systematic Literature Review

As aforementioned, the selected articles needed to progress through four filters in each repository. The filtration process, by each online repository and each filter, is visible in Table 1.

Table 1. Results of the Filtration Process.

	No Filter	1st Filter	2nd Filter	3rd Filter	4th Filter
IEEE	1104	647	614	132	6
ACM	110	35	20	5	3
SpringerLink	1337	45	45	28	5
ScienceDirect	6903	189	189	143	11
Google Scholar	200	200	129	64	22
Total	9654	1116	997	372	47

Furthermore, the articles from the Google Scholar repository were added manually. Taking into account that this repository cannot extract the final set of articles at once, the 1st filter was not applied in this repository.

The 1st filter had the intention of splitting the articles that are exclusively related with the ITSM domain from those who just referred to these concepts in the article body. This was completed by just selecting the ones that had the keywords in the title, abstract and author keywords. These three article sections were chosen for being the main parts that summarize the article's subject matter. With this filter, it was possible to reject a considerable number of articles. The 2nd filter had the goal of eliminating duplicate articles. On the 3rd filter, 957 articles were ranked by their publication rank, which totaled to nearly 389 publications given that various articles had the same publication (conference/journal). Consequently, in the 4th filter, 372 articles were read and assessed with the inclusion criteria described above, which resulted in 47 final articles. The flow of the filtration process can be seen in Figure 2.

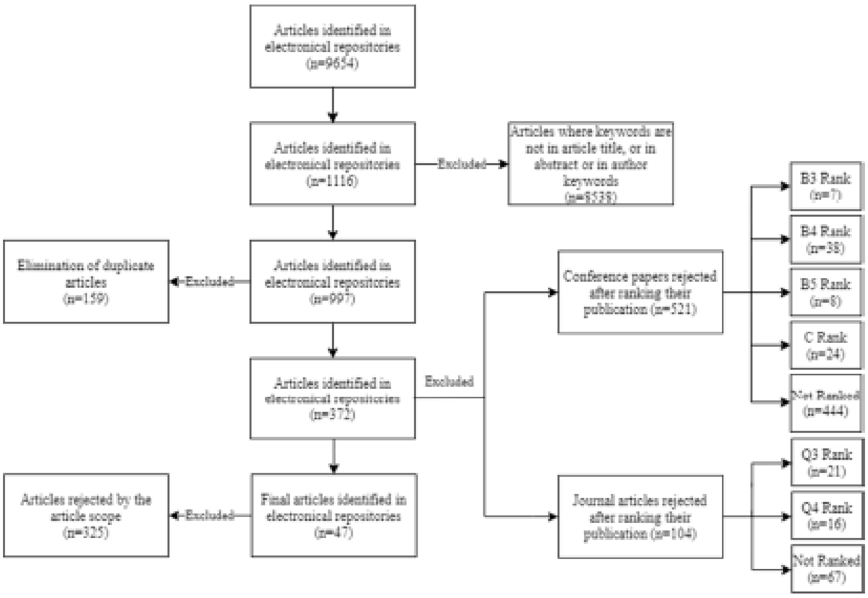


Figure 2. Flow of the Filtration Process.

5.1. *Sample Characteristics*

The final set of articles was composed of 47 publications. As mentioned before, only articles from conference proceedings and journals were accepted. As can be observed in Figure 3, which represents the distribution of the articles by their publication rank, only the B rank of ERA ranking is absent of publications amongst the final set.

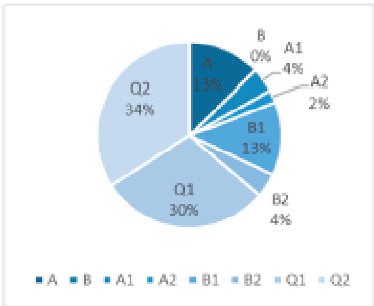


Figure 3. Articles Percentage by Publication Rank.

The distribution of the total number of each publication type is visible in Figure 4. The journal articles were the main contributor, being nearly 62% of the final set of articles. The journal or conference that stands out is the Journal of Computer Standards & Interfaces, with five publications present on the final article set.

Figure 5 lists the field's evolution in the scientific community since 2010 (https://scholar.google.com/ (accessed on 4 March 2021)), demonstrating the relevance of the domain while Figure 6 shows the distribution of the articles by year. It can be observed that the year interval goes from 2002 to 2019. Considering the evolution of the articles throughout the years and the article's sample in study, it can be stated that over the last three years, articles related to the four chosen ITSM concepts have decreased, emphasizing the need for a centralized article with all the ITSM concepts reunited.

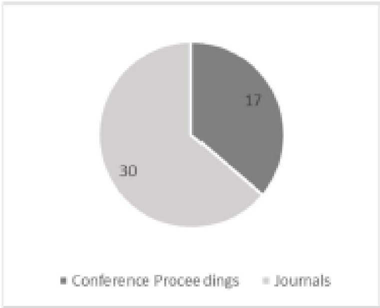


Figure 4. Articles Percentage by Publication Type.

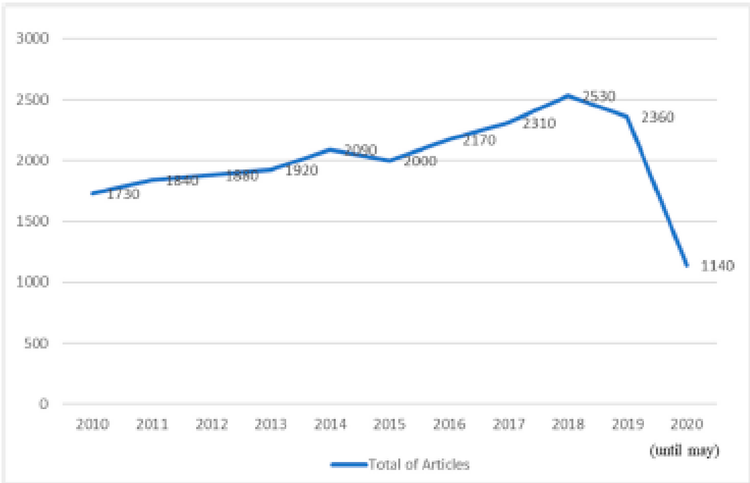


Figure 5. Total of ITSM Articles Published Since 2010.

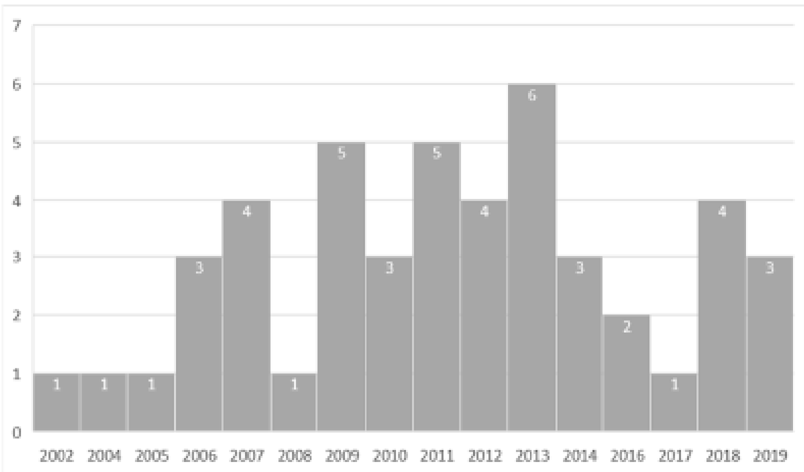


Figure 6. Articles by Year (final set).

5.2. Information Extraction Process

After applying the filters, the final set of articles were read. To support the information extraction from these articles, software for qualitative analysis was used. As mentioned before, all the details surrounding the ITSM domain were extracted.

6. Reporting the Review

6.1. ITSM Benefits

When implemented in a planned and "conscious" way, ITSM frameworks can produce many organizational benefits. As mentioned before, an SLR methodology was followed. After the analysis of the selected researches, the authors have summarized and characterized the benefits (Table 2).

ITSM frameworks support organizations in having a proper management strategy [45]. In fact, throughout the articles, it is commonly described that these frameworks help Management in developing better processes documentation [5] and monitoring [11], allowing for more detailed audits and IT process reports [33]. Moreover, the enhancement of documentation and monitoring brings higher transparency and comparability for the organization [11], producing an increase of processes maturity [38] and allowing management to have greater control of the processes, supporting organizations situated in environments of uncertainty [6].

Table 2. ITSM Benefits.

Benefits	Articles
Better processes control/documentation	[2,5,6,11,33,34,38,46–48]
Tangible improvements in process metrics (i.e., incident resolution times, change implementation time, predictable failures)	[1,2,6,11,33,45,46,48,49]
IT service quality improvement	[5–7,11,29,33,38,45,49]
Increase of customer satisfaction	[5,6,11,33,34,38,49,50]
Decrease in IT expenses	[2,5,7,19,29,33,38,49]
Higher efficiency/performance	[1,2,5,11,28,38,49,51]
Better IS-business alignment	[6,7,11,28,38,46,48]
Efficiency in the internal communication process/information sharing efficiency	[5,28,29,46–48]
Increase of organizational competitiveness	[6,7,19,20,33,52]
Mature processes	[11,38,46,51]
Increase of organization revenue	[5–7,33]
Better employee satisfaction	[5,33]
Reduction in staff	[5]

Most IT organizations implement ITSM frameworks due to the most perceivable improvements that these kinds of "tools" produce in the organization [2]. Many of these observable improvements occur at an operational level [6], where processes metric improvements captured by the organization are easily perceptible [11]. Also, it is defended that this type of benefit improved infrastructure predictability, reducing server or application faults [46]. Additionally, some research refers that incidents and error volume is drastically decreased [33] given that, with processes implemented, problems that may arise during normal operations can be dealt with proactively [45].

Furthermore, some authors also argue that the quality of IT services provided is increased [38] by a continuous-improvement method [7], increasing IT service flexibility and adaptability [29].

The increase of service quality mentioned by the scientific community is a generic benefit since the quality is increased due to the better control and performance of the processes. However, these quality increments generate one of the most important benefits to an organization: customer satisfaction [11]. It was described that customer satisfaction is enhanced as the quality of framework's activities is improved [38], satisfying the present and future demands of customers [33]. Additionally, these frameworks support customers with higher availability, responsiveness, and expertise [5].

In addition, the proactivity of dealing with adverse and unfavorable events that may appear might prevent wastes of money [2], being this benefit one of the most important for higher management. The articles affirm that applying ITSM principles can lead to a substantial decrease in operation and maintenance costs [7,19], such as in service provision costs [38], being intrinsically connected with the increment of the infrastructure and operational efficiency [5].

In fact, ITSM practices come under investigation for organizations that desire to increase the effectiveness of their functions [28]. It is argued that, with ITSM, organizations' processes become more mature, thus turning the performance of operational activities more robust [1]. Also, adjusting organizational practices to the implemented processes can help organizations increase productivity [51].

Moreover, another major benefit found was the alignment that the ITSM applies between the IS and the vision and strategy of the organization [28]. ITSM frameworks not only provide benefits at an operational level but also help organizations situate themselves in a more strategic position [6]. Such alignment further contributes to improving the effectiveness of the company [38].

By adopting ITSM frameworks and having the processes well-established and defined, communication is enhanced through all organizational layers, providing a "common language" [28]. This type of approach improves the consultation between groups within an organization [5] and increases the effectiveness and service quality of IT providers [29].

It is important for companies not to be mere support for the business but to also demonstrate attentiveness to new opportunities and needs of the "global" market [6]. With the implementation of ITSM frameworks and standards, organizations recognize opportunities to improve organizational competitiveness [7] in terms of client confidence and/or reputation [33] or even in relation to international competition advantage [20].

Another benefit pointed in literature is that ITSM frameworks turn the organizations' processes more mature [46] and consequently improve productivity [51]. By having internal guidelines provided by these frameworks, it is possible to develop and automate standard processes [11], helping to establish and enable higher maturity towards a better vision of the processes and their global understanding [38].

Most of the companies implement ITSM frameworks not just help them in having better service management but to increase revenue as well [5]. This relates to previous benefits since the reduction of operational costs and the increase of service quality translates into new customers, which subsequently converts into the increment of organizational revenue. Posteriorly, it is defended that the companies can easily return the investment made [6] and have greater financial control [33].

Some authors also point out that ITSM frameworks help in the transparency and robustness of companies' internal procedures [5]. By having improved and well-established "protocols", employee satisfaction will increase [33], which will support achieving expectations of IT staff [5]. Having employees pleased and confident not only helps organizations reorient or change their culture, but also improves company productivity.

Last but not least, perhaps the most controversial benefit present in literature is the reduction of staff. Although being found only in one article, it is stated that, given the frameworks' ability to increase process maturity, via partial or total automation, staff may be relocated or even reduced [5]. In fact, several attempts have been carried out to automate ITSM processes [53], having contemplated the concern of relocating staff [54].

6.2. ITSM Challenges

Despite the benefits that ITSM can produce, many organizations still face a variety of challenges daily. The challenges found in the literature (Table 3) were gathered and analysed.

Table 3. ITSM Challenges.

Challenges	Articles
Organization resistance	[2,29,30,47,48,50,55,56]
Frameworks complexity	[1,15,19,20,36,47]
Lack of knowledge/skills	[1,15,19,29,30,55]
Processes assessment (costly and time-consuming)	[4,5,20,35,57,58]
Lack of management support	[29,30,48,50,59]
Lack of resources	[29,30,56,59]
Difficult on quantifying the benefits	[5,29,48,56]
Big investment needed (Implementation and maintaining momentum)	[3,5,11,30]
Hard of planning to implement multiple frameworks	[20,21,33,52]
Steady lower costs	[49]

As mentioned by Keel et al. [2], the major challenges in ITSM can be divided into four areas: technology, data, process and people. In fact, resistance to change from co-workers is the most cited challenge. It is defended that this resistance is due to organizational culture [55], associated with the difficulties in changing and implementing a new strategy at the organizational level [48]. This general and natural tendency of people to resist change triggers a substantial "mentality" challenge for management [56]. It can also cause delays in implementation [48]. To sum up, such problems are due to organizational culture that lacks subjacent values like innovation or even support for change. To adopt a substantial change, management should prepare the technical staff first.

Additionally, ITSM frameworks are criticized by their complexity. Some of the critics point to the broad and generic definitions that the frameworks have [14,19]. Plus, the frameworks and standards do not provide guidelines for implementing them [36]. Furthermore, although the frameworks characterize the flows and interactions between processes, they lack a solution for process measurement and improvement [20].

Analysing Table 3, one can see that most of the challenges refer to the framework's implementation phase. Despite the growth interest in ITSM frameworks, they still lack clear implementation guidelines, creating many challenges for IT organizations.

Moreover, the lack of knowledge/skills of ITSM frameworks is another big challenge that organizations are facing [15]. In the literature, it is pointed out that "a gap between the knowledge of ITSM frameworks and their implementation" exists [1], which demonstrates that this challenge is intrinsically connected with framework complexity. In addition to that, the inexperience of project managers or core members of the organization [29] turn the implementation and maintenance of the practices difficult to execute [55].

Many IT organizations that have an ITSM framework implemented consider process assessments a difficult challenge, with this "act" of assessment important for continuous ITSM improvement [57]. However, process assessment methods are expensive and time-consuming [35]. Additionally, the frameworks do not have a measurement system for process improvement that could help organizations [20]. In the authors view, an organization without an assessment tool can lead to failures and will not be beneficial in mid-term, since an improvement strategy will not be defined because the as-is state of the processes is unknown.

Another challenge cited in the literature was the lack of management support. It is referenced in a case study that, although the "management layer" provided a big endorsement to the implementation, the commitment was not adequate [59]. In this case, the implementation failed. Also, it is defended that the lack of management support can create delays in the implementation process [48], which can increase the associated costs. Furthermore, without the support of senior management, the implementation could not be in compliance with the organizations' vision and strategy [50].

In light of this research, lack of management support is inherent to the lack of resources challenge. The big difficulty in framework implementation is the definition of the right "amount" of resources to provide [29]. Without management support, gathering sufficient resources, not just for the implementation, but also for process maintenance, can become a significant challenge [56].

Justifying the cost of the implementation and the investment of a framework by trying to show the benefits can be considered as a challenge [56]. Despite potential benefits, it is defended by literature that the implementation of frameworks can be slowed, not only by the amount of investment, but also due to the difficulty of realising the benefits and linkage between operational "performance measurement" and the organizations' revenue [5]. Quantifying the benefits can turn into a difficult task, and can be a reason for lack of continual investment [48].

It is estimated that IT service costs are between 70% and 80% of IT organization expenditure [30], therefore a big investment is needed, this being an enormous organizational challenge. The implementation of a framework will require needed investment given that the entire organizational "way of working" will change [3]. In addition to that, an extra cost and investment should be considered given that by running necessary infrastructure, the continual improvement and continuous performance monitoring also has associated costs [11].

Several organizations, with the purpose of being compliant with regulations and for process improvement, have the requirement of implementing several frameworks/standards [20]. In fact, this is becoming a challenge for organizations, given that planning these implementations requires taking into account the integration and interoperability of both frameworks for cost savings, complexity reduction and efficiency [52]. Also, the process' practices of various frameworks overlap each other [19,33,37], increasing the need for a good strategy when implementing multiple frameworks.

Finally, improving and maintaining the quality of provided services is an important task, not just for revenue but for continual ITSM improvement. However, it has been stated that it is challenging to maintain constant or lower costs while improving delivery quality [49].

6.3. ITSM Opportunities

As mentioned before, organizations face challenges when attempt to implement ITSM which increases their difficulty along the process. As Albert Einstein once sad "in the middle of every difficulty lies an opportunity". In this sense, this section does not have the objective to introduce solutions for problems that organizations are confronting, but to list some opportunities worthy of further research (Table 4).

Table 4. ITSM Opportunities.

Opportunities	Articles
Lack of guidelines to processes improvement	[15,60–62]
Processes assessment	[4,58,60]
Identifying processes interdependencies and their overlap	[20]
Maturity Models for the needs of IT management providers	[17]
Cloud computing and DevOps	[49]
Lack of guidelines to processes improvement	[15,60–62]

Processes improvement is important for organizations [60]. However, organizations still struggle with improvement and in defining what to improve [14]. It is defended that process capability grows and costs are reduced with increasing maturity levels [61], that is, improved processes can reduce costs to organizations. A research with the purpose of creating guidelines to improve processes could save resources to IT organizations.

As stated above, organizations find it difficult to assess their processes; ITSM frameworks do not provide guidelines to help on this matter. Most assessments are conducted by third-party organizations and have a big associated cost [4]. Also, it is criticized that the guidelines that do exist lack transparency [58]. It is visible that a gap exists between the implementation of processes and their assessment.

The "phenomenon" of adopting multiple frameworks can create opportunities. Some experienced organizations are doing business with this "phenomenon", by providing services that help organizations with multiple frameworks, reducing their complexity. A potential opportunity lies in understanding the process interdependencies and linkage, to help determine the overlap between each process [20]. This research can help organizations that need to implement various frameworks and processes, to have a better plan to understand which processes overlap and their relative dependencies.

Many maturity models have been developed. These kinds of tools can be useful for organizations that want to assess a certain domain in their organization. Machado et al. [17] argued that maturity models still lack in meeting the needs of IT management providers and appoints that a maturity model that would have a focus on service improvement could help organizations enhance their processes and services. Grounded on the previous statements, further investigation relating ITSM and the "maturity model" concept should be incentivized.

Although cloud computing is not a new concept, it is starting to be used and related to several domains. In fact, cloud computing allows for higher performance. IT service implementation could be faster via the utilization of cloud computing and DevOps [49]. Solutions with the intercalation of both areas could not just decrease implementation times, but also provide computing infrastructure to small organizations that would help in maintaining service operability.

6.4. ITSM Implementation

The implementation process of an ITSM framework is one of the most significant challenges faced by organizations, as observed in Section 6.2. This research has collected some advisable practices found in literature (Table 5) to help ITSM implementation.

Table 5. ITSM Implementation Practices.

Implementation Practices	Articles
Commitment from all levels of management	[2,11,48,59]
IT and business processes must be aligned	[1,28,47]
Follow-up and assessment	[4,11,47]
Quick wins	[11,48]
Training	[1,11]
Commitment from all levels of management	[2,11,48,59]

When an organization moves to adopt a new ITSM framework, it is important to have the commitment from all levels of management [2]. Senior management support is considered a critical success factor in any major IT project [59]. In fact, research has mentioned that in five case studies, all guaranteed that for successful implementation senior management should support the project, which would help to fund and provide the right resources for implementation [48].

Also, planning the implementation is an essential phase. One of the most important practices found when implementing a framework is the consideration of the IT strategy and vision of the organization when planning the adoption of a framework [28]. Additionally, the implementation must be aligned with business objectives, in order not only to solve a problem or need, but to also create a positive impact in the strategy of the organization [1].

Creating and having an improvement and follow-up strategy established is also mentioned as a good practice [47]. Having follow-up and implementation plans will help in the successful and effective implementation at the operational level [4]. Defining process improvement strategies can help organizations to have mature processes and achieve cost savings.

When planning an implementation, seeking quick-wins can be a good option [11]. It is defended that there is no correct order to implement processes, although looking for quick wins can be a good strategy [48], which will help the organization achieve visible benefits at a faster rate.

Before performing a big change in the way that an organization works, staff must be advised and trained. Training is considered one of the most important tasks before the implementation of a framework [1]. The affected staff must be prepared, and a personnel development plan must be enforced in order to qualify co-workers in terms of mentality and knowledge [11].

7. Key Findings

7.1. RQ.1: ITSM Benefits

The present research has analysed studies developed by the scientific community and collected 13 benefits. By observing

Table 2, it is visible that all these benefits are connected, where several benefits translate into others. Many of the identified benefits are tangible in nature, being the easiest to perceive. Most of the organizations implement ITSM frameworks to help them better manage their infrastructure and services. It is important to note that these tools create an "environment" of certainty where all processes are well-defined and co-workers know what it is necessary to perform to accomplish specific goals, that is, inside the organization what must be done is known. This can turn the IT infrastructure of an organization more robust and increase employee satisfaction, therefore improving company performance.

With this "atmosphere" of efficiency, where the bulk of conditions to have a strong infrastructure with controlled processes are reunited, in a natural way, the quality of provided services by these companies will grow. This can be considered the central benefit of ITSM framework application. Some companies also aim to increase their revenue and new customers, however, to achieve this it is necessary to have more flexible and robust services.

Moreover, service quality is dependent on its "surroundings", that is, to reach a high level of service quality it is necessary to create an effective infrastructure, including well-documented and settled processes and also an IT staff that is motivated and trained to perform their tasks.

As mentioned before, the raising of service quality will create satisfied customers as well as bring new ones, consequently increasing revenue while decreasing operational costs. When correctly implemented, there is prevention of unnecessary waste of money by employees, and by possible errors and problems that may appear. Conceptual models can be useful to demonstrate the flows of information and to give a high level "picture" of themes [63] in a way that can easily demonstrate the correlation and complexity between concepts [64]. From this perspective and by observing the connection and correlation between the benefits, the authors decided to interpret these connections and a conceptual model for the ITSM benefits was developed (Figure 7).

Since the benefits produce other benefits and are connected, the expression leads to was used. Following the line of reasoning described before, by having an environment where the processes are controlled and well-defined, not just communication efficiency increases but also employee satisfaction is improved upon, since knowledge of what to do is more widely known.

Also, possessing controlled processes coupled with their well-definition, will originate mature processes [11]. Tangible improvements in process metrics can be achieved by mature processes [17,45], creating here an important "conceptual triangle" with these three benefits. Considering that the processes are at a certain level of maturity, metrics can be used to adopt continual improvement, enabled greater process control. This "conceptual triangle" can be seen as an improvement cycle.

Several investigations detail that as process maturity increases, business performance will also be improved [6,9,10]. Following the conclusions of these articles, the authors defined that mature processes originate higher efficiency and performance in organizations [5,17]. Additionally, with mature processes, the waste of resources will decrease, being correlated with the benefit of "decrease in IT expenses" [5,7,61].

Figure 7. ITSM Benefits Conceptual Model.

With all the metrics of processes controlled and with higher performance, the quality of services provided by the organization will grow as a result [11,38]. As mentioned before, this benefit is considered by the authors of this research the central one, since this is one of the main objectives of ITSM frameworks, and the principal benefit to increase the customer satisfaction and bring more clients to the organization [33,38].

By having customers pleased and via the achievement of gaining new clients, revenue generated in the organizations will rise. Besides that, the fact of having IT expenses decreased coupled with potential staff reduction, which occurs as a result of process maturity and automation, this will also contribute to the increment in revenue of organizations. Frequently, this benefit is the principal objective for higher management to implement an ITSM framework.

Additionally, having controlled processes supports the IS-business alignment [6], helping organizations to have strategic objectives aligned with IT infrastructure, which will create and grow competitiveness [6] in the organization since a robust and efficient infrastructure, prepared for any change in the vision, strategy or prepared for any demand, is in place [19]. This condition can help organizations achieve higher competitiveness in an international environment.

It is perceivable that having controlled processes can originate an interconnected "chain of benefits". It is important to note that, by observing the developed conceptual model, the main identified benefits are correlated with others, however, those benefits are originated by others, that is, a benefit will not produce other benefits without first being improved itself.

With this in mind, and with the fact that several benefits are dependent on the "maturity" of other benefits to achieve a good mature level, the authors developed a table (Table 6) that presents which benefits promote (out) and are promoted by (in) other benefits.

Table 6. Relation between Benefits and Inputs/Outputs.

Benefits	→In	→Out
Mature processes	1	5
Better processes control/documentation	1	4
Tangible improvements in process metrics (i.e., incident resolution times, change implementation time, predictable failures)	1	2
IT service quality improvement	2	2
Higher efficiency/performance	3	2
Increase of organizational competitiveness	2	2
Increase of customer satisfaction	2	1
Decrease in IT expenses	2	1
Better IS-business alignment	2	1
Efficiency in the internal communication process/information sharing efficiency	1	1
Better employee satisfaction	1	1
Reduction in staff	1	1
Increase of organization revenue	4	0

Given the identified relationship amongst benefits, it is reasonable to argue that organizations should first focus on the benefits that promote (out) more other benefits rather than being promoted (in) by other benefits. Therefore, by observing Table 6, the benefit "Mature processes" can be considered one of the most important to achieve, since on its own it can trigger five further benefits.

On the other hand, the last benefit organizations should be focused on and aiming to improve is the "Increase of organizational revenue", given that it does not promote other benefits, and is promoted by or generate from other four benefits.

Although being the last benefit to aim, as we mentioned earlier, it is also the goal that most organizations desire. This may somehow justify why a considerable part of ITSM implementation fail [25,50,59]. Organizations may have not been focused on the right priorities to achieve the final aim. This highlights the fact that to achieve substantial benefits, all the environment around ITSM must be well understood and implemented holistically, being imperative not to focus merely on revenue growth (main goal).

Table 6 can be compared to a "Quick-Wins" strategy, where the objective is to demonstrate which benefit should be addressed first in order to achieve further benefits.

7.2. RQ.2: ITSM Challenges

Despite benefits, a not planned and haphazard adoption of ITSM frameworks can create many challenges and problems. Nevertheless, by observing the challenges in Table 3, it is possible to affirm that most challenges faced by organizations are before and during implementation. It is noted that people face a big challenge in the implementation and operability of an ITSM framework. The application of frameworks changes the way how collaborators work, creating a change-resistant environment. In fact, this challenge transcends to any type of organization in which employees are comfortable with the manner that they work, feeling uncomfortable to change. Most of this resistance to change can be considered a result of organizational culture that underlines the organization itself, lacking the "right principles" to this kind of action.

The lack of management support is pointed out as a nuclear challenge for ITSM framework implementation. It stands out that most of the challenges are due to the fact of lacking management support, which without said support a lack of resources or even a lack of an "example" to operational co-workers is generated, producing in that way a possible resistance to change. In addition, the lack of support from management can be directly connected with the big investment needed to implement a framework, and with not having enough budget to invest.

Also, one of the main challenges is the actual complexity of frameworks. Nowadays, frameworks do not describe an implementation process, making it challenging for organizations that do not have adequate knowledge or skills to adopt them. Companies, in an attempt to apply a framework with lacking knowledge, can have disastrous results in revenue. Furthermore, these frameworks do not characterize how processes should be assessed or even improved upon, not increasing the processes maturity and not producing the benefits that are expected, probably creating the challenge of being difficult to quantifying the benefits.

When taking everything into account there still exists lack of information in frameworks targeting the implementation, assessment and improvement of processes, generating a loss of money for organizations, consequently creating doubt by the management level of a company.

In this section, a conceptual map was also created (Figure 8) to illustrate the relationships and associations among the challenges found in literature. As aforementioned, challenges, if not mitigated, can lead to other challenges, creating a difficult environment for a successful ITSM implementation and operation.

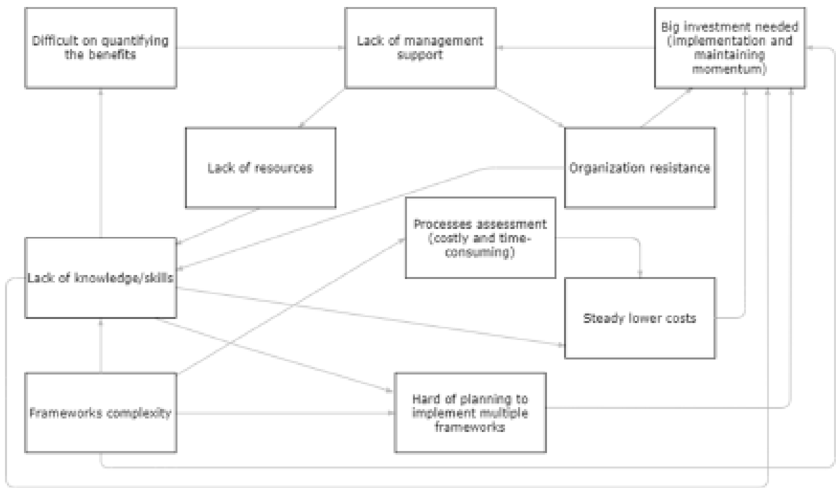


Figure 8. ITSM Challenges Conceptual Model.

Lack of management support in organizations leads to a lack of resources [29,50], since it is the management that decides where to apply all the means that companies have. Besides that, as mentioned, this lack of support can promote an environment of resistance from operational lines [48], since the organizations' managers do not take the lead in giving an "example".

Considering that organizations have no sufficient resources applied in ITSM, it will create a lack of knowledge and skills, given the fact that there is no funds to invest in training that would help to understand and acquire the "know-how" of ITSM. However, not only the lack of resources leads to this challenge, but also organizational resistance, since the co-workers would not intend to apply themselves in this kind of training.

Having the knowledge and the skills of ITSM can help collaborators justify to management the use and implementation of ITSM, by demonstrating the benefits that it can produce. However, without awareness and understanding, it can become difficult to quantifying the benefits, leading to the challenge of lack of management support [50]. Additionally, with lack of ITSM know-how, it can be hard to steadily lower costs of operational lines, creating in this situation a waste of resources. Also, for organizations that need to implement multiple frameworks, this can turn into a big issue when they do not have the appropriate knowledge and skills for that type of implementation.

Besides the fact that most of the challenges found are based on the bad practices that an organization "adopts", the ITSM framework complexity is considered an external challenge, given that it is not produced by organizations. Organizations and ITSM experts have been criticizing the ITSM frameworks for being too generic and complex. This challenge leads to the hard planning of an implementation and also results in lack of knowledge. Furthermore, these frameworks are criticized for not supporting and demonstrating how organizations should assess their processes [47,57], being difficult to improve and produce maturity [20], which by itself will not create the benefits that should be created.

Also, the frameworks' complexity can lead to a need of a big investment, not just for the ITSM implementation but also for its maintenance, considering that for organizations that do not have the "know-how" mistakes can easily be made. The implementation of ITSM needs a big investment of organizational resources since companies will change the way they operate. However, with lack of good practices, a waste of resources will happen, and further big investments will be needed to overcome committed errors. This challenge, in the authors view, is one of the reasons for lack of management support in the provision of proper resources to implement and maintain ITSM.

As done with the benefits, a table with the relation between the challenges was developed (Table 7). Contrary to the benefits, observing Table 7 we will note that the challenges that have more arrows coming out are the ones that should be avoided since these challenges will produce many more.

Table 7. Relation between Challenges and Inputs/Outputs.

Challenges	→In	→Out
Frameworks complexity	0	4
Lack of knowledge/skills	3	4
Organization resistance	1	2
Lack of management support	2	2
Processes assessment (costly and time-consuming)	1	1
Lack of resources	1	1
Difficult on quantifying the benefits	1	1
Hard of planning to implement multiple frameworks	2	1
Steady lower costs	2	1
Big investment needed (Implementation and maintaining momentum)	5	1

7.3. RQ.3: ITSM Opportunities

Additionally, this article has gathered five opportunities from the research community. Despite being few in numbers, the opportunities focus mostly on the problems that organizations face with ITSM. Observing Table 4, the opportunities for research are related mainly with the assessment of the processes. Since organizations should have a continual improvement plan to produce benefits that should be created with the implementation of ITSM, for the authors this opportunity is considered as of great importance.

Both opportunities of processes assessment and improvement are intrinsically connected with the opportunity of developing maturity models for the ITSM domain. Maturity models support organizations to assess their processes and practices, by assigning a level of maturity to a process that is being evaluated in accordance with the practices established by the model. In fact, several maturity models have been developed for ITSM [37,64,65]. However, other possibilities could be studied, since many organizations still have difficulties in the assessments, and maturity models have been criticized.

Nevertheless, cloud computing was mentioned as an opportunity for the future of ITSM, by helping small or medium organizations with the resources that they could originally not afford. With the ITSM opportunities, considering that those found were small, the authors could not produce a conceptual model with the connections and correlations between them.

7.4. RQ.4: ITSM Implementation

Finally, since organizations face challenges during ITSM implementation, five practices were found. Looking at Table 5 it is possible to conclude that management support in ITSM implementation is one of the most crucial practices, considering that it is one of the main factors that can dictate if the implementation will have the proper resources.

Additionally, using quick wins with a strategy to choose which processes should be implemented, supports organizations on having the infrastructure aligned with the business and prepared for the demand.

Also, the training and preparing of co-workers for the change is important as well. From this research's' view, training practice to staff could mitigate the challenge of resistance to change in organizations.

As occurred with the opportunities, it was not possible to correlate the best implementation practices; consequently, a conceptual model for the implementation practices was not developed.

As mentioned before, the complexity of the frameworks can be considered as an external challenge, being difficult to prevent. Another challenge that should be avoided is the lack of knowledge/skills, given that this challenge "defines" how well ITSM will be implemented and operated.

Also, it is important to mention that the challenge of "Big investment needed" is the challenge that is more dependent on others. A big investment to change organizations will always be needed; however, in the view of the authors, to decrease the value of this investment, all other challenges surrounding it must be mitigated or controlled.

This table can help an understanding of which challenges should be mitigated first to produce the fewest challenges possible.

7.5. ITSM Environment and Contextualization

The best implementation practices can mitigate several challenges and produce benefits. Also, as mentioned before, challenges can create opportunities for research. A connection exists between all these concepts. Having this in mind, the authors created a conceptual model with the relationships among all concepts (Figure 9).

The best practice of training is connected to both challenges and benefits. Training not only helps organizations achieve the knowledge of how to implement ITSM but also how to maintain it. By training collaborators, they will gain the proper "know-how" of how to operate the processes [1,29], increasing their maturity and improving their metrics, making ITSM processes more efficient. Additionally, training co-workers can help partially mitigate the external challenge of framework complexity. Also, with collaborators knowing what-to-do, the waste of resources will decrease; consequently, the investment in implementation and operability will also decrease.

As mentioned earlier, the "Quick-Wins" strategy is considered a good practice of implementation, for those who do not know which processes to implement. From the research's' point of view, this strategy is able to exhibit benefits, mitigating the difficulty of showing ITSM benefits [11], since the seeking for earlier benefits underlines this strategy. Furthermore, this can decrease the investment needed to implement ITSM. A Quick-Wins strategy also produces several benefits, mainly the enhancement to IS-business alignment given that a choice is made to implement the processes that "fits" better in the organization's IT infrastructures. Also, this strategy may increase the organization's revenue by creating several benefits in an earlier stage of ITSM implementation and operation.

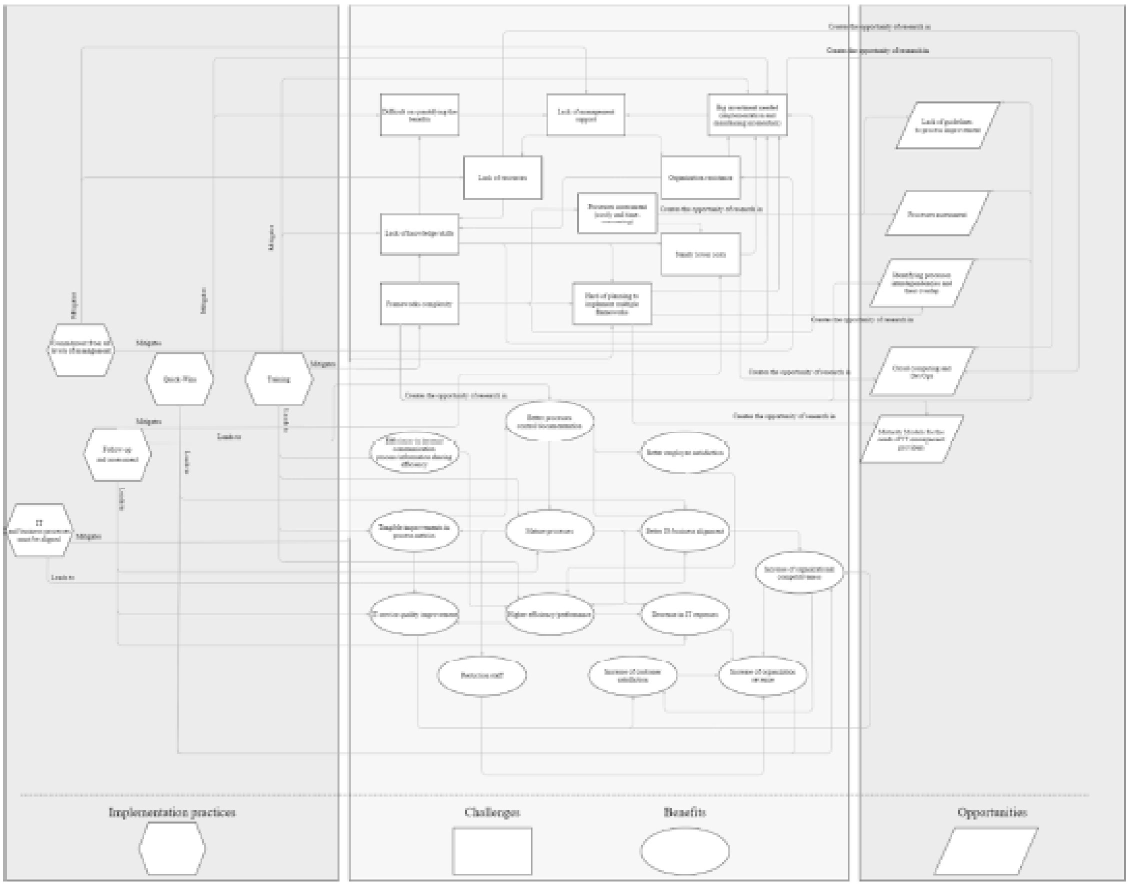


Figure 9. Conceptual Model of ITSM Main Concepts.

The commitment of all management hierarchy is not necessarily connected with the benefits found in the literature. However, it is intrinsically correlated with the mitigation of several challenges, which by itself produces benefits. As demonstrated before, with all levels of management committed with this big project, the resistance of the organization to change could be avoided [2,29,48], and the lack of resources that should be provided to the ITSM implementation and operation could be mitigated [59].

The follow-up and assessment of the processes is viewed as a practice with a perspective towards continual improvement [60]. This practice not only helps organizations have processes well-documented but well controlled as well. The assessment will allow for the development of process maturity [29], since organizations will have a basis that will dictate where processes must be improved [4]. Additionally, the follow up should support organizations on the process's operability, helping to decrease the waste of resources, mitigating the challenge of maintaining steady lower costs and producing the benefit of decreasing IT expenses.

Many organizations have the necessity of implementing more than one framework face to the regulations of the market [20]. However, this kind of implementation brings many difficulties since it is necessary to have a good plan defined. Having the processes aligned with the business and with the organizations' strategy, can help organizations partially mitigate hard planning and implementation. Also, this good practice produces the benefit of having IS aligned with the organization's business.

As aforementioned, challenges originate opportunities for research. Although several practices could mitigate various challenges, there still exists the necessity of new research for the creation of innovative and new solutions. The integration of DevOps and Cloud computing is a good example for research. Several organizations still have a lack of resources for the implementation of ITSM [49], and cloud providers, with the provision of proper resources, could help small and medium organizations have an infrastructure that would make possible to have ITSM allocated. Also, the difficulty of making costs stable and low should be an opportunity for investigation in conjunction with cloud computing.

Companies still have the difficulty of assessing their own processes due to framework complexity. Also, with the requirement of implementing several frameworks, some organizations face a variety of challenges [20]. The authors found that the junction between maturity models and ITSM could help organizations mitigate these kinds of challenges [17]. Furthermore, these opportunities are also related to the lack of guidelines for assessment and improvement that frameworks do not provide.

This conceptual model demonstrates how the four concepts relate to each other. Moreover, this map could help organizations on planning their implementation and have a plan of mitigation for the future challenges that they could face.

8. Conclusions

Traditional IT organizations have evolved. This evolution made organizations change from technology providers to service providers, looking at IT management with a different perspective. ITSM is a conjunction of frameworks that help IT service providers enhance the management practices of their services. Additionally, ITSM supports organizations in adding value to their services by improving their quality.

Although ITSM produces many benefits, organizations still face various challenges due to lack of guidelines in several contexts. This research aimed to investigate the benefits, challenges, opportunities, and implementation best practices of the ITSM domain. To conduct this research, a SLR was adopted and a discussion of results for each concept was completed. A total of 47 articles were analysed. The main contributions of this research are:

- A holistic view of ITSM implementation including how the benefits and challenges relate to each other. With this, organizations have the awareness of which benefit should be addressed first, with the purpose of achieving their main goals and avoiding undesirable challenges.
- The identification of the benefits that the organizations should focus first, to produce more benefits.
- Determination of which challenges should be avoided and considered when planning and operating the ITSM, with the view of generating as few challenges as possible to the organizations.
- The identification of main guidelines that may promote the desirable benefits in the initial phases of ITSM implementation.
- The identification of opportunities to be explored by new research, based on the difficulties that many organizations face daily.

Despite this, for future work, the challenges, benefits, opportunities, and implementation practices appointed in this research could be used to lead new investigations that would contribute not just for the ITSM domain but also for IT organizations that struggle with these issues every day. It seems that the lack of guidelines for processes improvement and assessment could be the most important "future work" in this domain, given that it is necessary for creating an improvement plan, in a view of developing mature processes that would produce the expected benefits. Considering that, research with innovative solutions for the assessment and improvement of processes should be considered. Also, a detailed examination of the relationship between benefits and challenges must be considered for new investigation. Plus, less supported concepts according to Tables 2–5 should also be better explored in the future. For instance, which ITSM practices may exist to manage services provided through cloud technology; or how can DevOps assist ITSM discipline in practice (some researchers are already investigating it [24]); or provide some empirical evidences on how ITSM can help organizations reducing employees and cut costs; or even further explore and propose new maturity models to assist ITSM implementation or even reduce ITSM complexity (please see [37]).

This research has the limitation of the selected articles being from limited ranks. It is reasonable to admit that other important and relevant information could be provided by other research published in lower ranks. Additionally, this research has the limitation of the conceptual models being developed based on critical analysis, which can be criticised for several choices made.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Melendez, K.; Dàvila, A.; Pessoa, M. Information technology service management models applied to medium and small organizations: A systematic literature review. *Comput. Stand. Interfaces* **2016**, *47*, 120–127. [CrossRef]
2. Keel, A.J.; Orr, M.A.; Hernandez, R.R.; Patrocinio, E.A.; Bouchard, J. From a technology-oriented to a service-oriented approach to IT management. *IBM Syst. J.* **2007**, *46*, 549–564. [CrossRef]
3. Winniford, M.A.; Conger, S.; Erickson-Harris, L. Confusion in the ranks: IT service management practice and terminology. *Inf. Syst. Manag.* **2009**, *26*, 153–163. [CrossRef]
4. Shrestha, A.; Cater-Steel, A.; Toleman, M.; Behari, S.; Rajaeian, M.M. Development and evaluation of a software-mediated process assessment method for IT service management. *Inf. Manag.* **2019**, *57*, 103213. [CrossRef]
5. Gacenga, F.; Cater-Steel, A.; Toleman, M. An International Analysis of IT Service Management Benefits and Performance Measurement. *J. Glob. Inf. Technol. Manag.* **2010**, *13*, 28–63. [CrossRef]
6. Marrone, M.; Kolbe, L.M. Uncovering ITIL claims: IT executives' perception on benefits and Business-IT alignment. *Inf. Syst. E-Bus. Manag.* **2011**, *9*, 363–380. [CrossRef]
7. Galup, S.D.; Dattero, R.; Quan, J.J.; Conger, S. An overview of IT service management. *Commun. ACM* **2009**, *52*, 124. [CrossRef]
8. Pollard, C.; Cater-Steel, A. Justifications, Strategies, and Critical Success Factors in Successful ITIL Implementations in U.S. and Australian Companies: An Exploratory Study. *Inf. Syst. Manag.* **2009**, *26*, 164–175. [CrossRef]
9. Marrone, M.; Kolbe, L. ITIL and the Creation of Benefits: An Empirical Study on Benefits, Challenges and Processes. In Proceedings of the 18th European Conference on Information Systems, Pretoria, South Africa, 7–9 June 2010.
10. Salman, R.; Daim, T.; Raffo, D.; Dabic, M. Exploring capability maturity models and relevant practices as solutions addressing information technology service offshoring project issues. *Int. J. Manag. Sci. Eng. Manag.* **2018**, *13*, 147–157. [CrossRef]
11. Hochstein, A.; Tamm, G.; Brenner, W. Service Oriented IT Management: Benefit, Cost and Success Factors. In Proceedings of the European Conference on Information Systems, Regensburg, Germany, 26–28 May 2005; Volume 98, p. 12.
12. Koch, H.; Gierschner, C. Advantages of an ITIL-based Process Framework in a Complex SAP® System Landscape. In Proceedings of the 2007 4th IEEE Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, Dortmund, Germany, 6–8 September 2007; pp. 431–433.
13. Göbel, H.; Cronholm, S.; Seigerroth, U. Towards an agile method for ITSM self-assessment: A Design Science Research Approach. In Proceedings of the International Conference on Management, Leadership and Governance (ICMLG2013), Bangkok, Thailand, 7–8 February 2013.
14. Küller, P.; Vogt, M.; Hertweck, D.; Grabowski, M. A domain specific IT service management approach for small & medium enterprises: A Domain Specific Approach. In Proceedings of the 16th International Business Information Management Association Conference, IBIMA 2011, Kuala Lumpur, Malaysia, 29–30 June 2011; Volume 3, pp. 1356–1369.
15. Jantti, M. Improving Incident Management Processes in Two IT Service Provider Companies. In Proceedings of the 2011 22nd International Workshop on Database and Expert Systems Applications, Toulouse, France, 29 August–2 September 2011; pp. 26–30.
16. Hoerbst, A.; Hackl, W.O.; Blomer, R.; Ammenwerth, E. The status of IT service management in health care—ITIL in selected European countries. *BMC Med. Inf. Decis. Mak.* **2011**, *11*, 76. [CrossRef]
17. Machado, R.F.; Reinehr, S.; Malucelli, A. Towards a Maturity Model for IT Service Management Applied to Small and Medium Enterprises. *Eur. Conf. Softw. Process Improv.* **2012**, *301*, 157–168.
18. Gehrmann, M. Combining Itil, Cobit and Iso/Iec 27002 for Structuring Comprehensive Information Technology for Management in Organizations. *Navus Rev. Gestão Tecnol.* **2012**, *2*, 53–65. [CrossRef]

19. Conger, S.; Winniford, M.; Erickson-Harris, L. Service Management in Operations. In Proceedings of the Americas Conference on Information Systems, Toronto, ON, Canada, 14–17 August 2008; p. 11.
20. Cater-Steel, A.; Tan, W.-G.; Toleman, M. Challenge of adopting multiple process improvement frameworks. In Proceedings of the 14th European Conference on Information Systems (ECIS 2006), Göteborg, Sweden, 12–14 June 2006; pp. 1375–1386.
21. Pereira, R.; da Silva, M. A Literature Review: Guidelines and Contingency factors. In Proceedings of the European, Mediterranean & Middle Eastern Conference on Information Systems, Windsor, UK, 4–5 October 2012; pp. 342–360.
22. Vicente, M.; Gama, N.; da Silva, M.M. Using archimate to represent ITIL metamodel. In Proceedings of the IEEE International Conference Business Informatics, IEEE CBI, Vienna, Austria, 15–18 July 2013; pp. 270–275.
23. Kaiser, A.K. *Reinventing ITIL® in the Age of DevOps*; Apress: New York, NY, USA, 2018.
24. Faustino, J.; Pereira, R.; Alturas, B.; da Silva, M.M. Agile information technology service management with DevOps: An incident management case study. *Int. J. Agil. Syst. Manag.* **2020**, *13*, 339–389. [CrossRef]
25. Sharifi, M.; Ayat, M.; Rahman, A.A.; Sahibudin, S. Lessons learned in ITIL implementation failure. In Proceedings of the International Symposium on Information Technology 2008, ITSim, Kuala Lumpur, Malaysia, 1–7 March 2008; Volume 1, pp. 8–11.
26. Beachboard, J.; Conger, S.; Galup, S.D.; Hernandez, A.; Probst, J.; Venkataraman, R. AMCIS 2007 Panel on IT Service Management: IT Service Management in the IS Curriculum. *CAIS* **2007**, *20*, 35. [CrossRef]
27. Braun, C.; Winter, R. Integration of IT Service Management into Enterprise Architecture. In Proceedings of the 2007 ACM Symposium on Applied Computing, Seoul, Korea, March 2007; pp. 1215–1219.
28. Winkler, T.J.; Wulf, J. Effectiveness of IT Service Management Capability: Value Co-Creation and Value Facilitation Mechanisms. *J. Manag. Inf. Syst.* **2019**, *36*, 639–675. [CrossRef]
29. Wan, J.; Jones, J.D. Managing IT service management implementation complexity: From the perspective of the Warfield Version of systems science. *Enterp. Inf. Syst.* **2013**, *7*, 490–522. [CrossRef]
30. Marrone, M.; Kolbe, L.M. Impact of IT Service Management Frameworks on the IT Organization: An Empirical Study on Benefits, Challenges, and Processes. *Bus. Inf. Syst. Eng.* **2011**, *3*, 5–18. [CrossRef]
31. Salle, M. *IT Service Management and IT Governance: Review, Comparative Analysis and their Impact on Utility Computing*; Hewlett-Packard Co.: Palo Alto, CA, USA, 2004; p. 26.
32. McNaughton, B.; Ray, P.; Lewis, L. Designing an evaluation framework for IT service management. *Inf. Manag.* **2010**, *47*, 219–225. [CrossRef]
33. Cots, S.; Casadesús, M.; Marimon, F. Benefits of ISO 20000 IT service management certification. *Inf. Syst. E-Bus. Manag.* **2016**, *14*, 1–18. [CrossRef]
34. Proehl, T.; Erek, K.; Limbach, F.; Zarnekow, R. Topics and Applied Theories in IT Service Management. In Proceedings of the 2013 46th Hawaii International Conference on System Sciences, Wailea, HI, USA, 7–10 January 2013; pp. 1367–1375.
35. Shrestha, A.; Cater-Steel, A.; Toleman, M.; Rout, T. Benefits and relevance of International Standards in a design science research project for process assessments. *Comput. Stand. Interfaces* **2018**, *60*, 48–56. [CrossRef]
36. Arcilla, M.; Calvo-Manzano, J.A.; Feliu, T.S. Building an IT service catalog in a small company as the main input for the IT financial management. *Comput. Stand. Interfaces* **2013**, *36*, 42–53. [CrossRef]
37. Aguiar, J.; Pereira, R.; Vasconcelos, V.B. An Overlapless Incident Management Maturity Model for Multi-Framework Assessment (ITIL, COBIT, CMMI-SVC). *IJIKM* **2018**, *13*, 137–163. [CrossRef]
38. Orta, E.; Ruiz, M. Met4ITIL: A Process Management and Simulation-based Method for Implementing ITIL. *Comput. Stand. Interfaces* **2018**, *61*, 1–19. [CrossRef]
39. Brereton, P.; Kitchenham, B.A.; Budgen, D.; Turner, M.; Khalil, M. Lessons from applying the systematic literature review process within the software engineering domain. *J. Syst. Softw.* **2007**, *80*, 571–583. [CrossRef]
40. Snyder, H. Literature review as a research methodology: An overview and guidelines. *J. Bus. Res.* **2019**, *104*, 333–339. [CrossRef]
41. Linnenluecke, M.K.; Marrone, M.; Singh, A.K. Conducting systematic literature reviews and bibliometric analyses. *Aust. J. Manag.* **2020**, *45*, 175–194. [CrossRef]
42. Okoli, C.; Schabram, K. Working Papers on Information Systems A Guide to Conducting a Systematic Literature Review of Information Systems Research. *Work. Pap. Inf. Syst.* **2010**, *10*, 10–26.
43. Mallett, R.; Hagen-Zanker, J.; Slater, R.; Duvendack, M. The benefits and challenges of using systematic reviews in international development research. *J. Dev. Eff.* **2012**, *4*, 445–455. [CrossRef]
44. Kitchenham, B. *Procedures for Performing Systematic Reviews*; Joint Technical Report; Computer Science Department, Keele University: Keele, UK; National ICT Australia Ltd.: Sydney, Australia, 2004.
45. Wickboldt, J.A.; Bianchin, L.A.; Lunardi, R.C.; Granville, L.Z.; Gaspary, L.P.; Bartolini, C. A framework for risk assessment based on analysis of historical information of workflow execution in IT systems. *Comput. Netw.* **2011**, *55*, 2954–2975. [CrossRef]
46. Marrone, M.; Gacenga, F.; Cater-Steel, A.; Kolbe, L. IT Service Management: A Cross-national Study of ITIL Adoption. *CAIS* **2014**, *34*, 49. [CrossRef]
47. Lucio-Nieto, T.; Colomo-Palacios, R.; Soto-Acosta, P.; Popa, S.; Amescua-Seco, A. Implementing an IT service information management framework: The case of COTEMAR. *Int. J. Inf. Manag.* **2012**, *32*, 589–594. [CrossRef]
48. Cater-Steel, D.A.; Toleman, P.M.; Tan, D.W.-G. Transforming IT Service Management—the ITIL Impact. In Proceedings of the Australasian Conference on Information Systems, Brisbane, Australia, 10–13 December 2006; p. 11.

49. Kubiak, P.; Rass, S. An Overview of Data-Driven Techniques for IT-Service-Management. *IEEE Access* **2018**, *6*, 63664–63688. [CrossRef]
50. Tan, W.-G.; Cater-Steel, A.; Toleman, M.; Seaniger, R. Implementing Centralised IT Service Management: Drawing Lessons from the Public Sector. In Proceedings of the 18th Australasian Conference on Information Systems, Toowoomba, Australia, 5–7 December 2007.
51. Barafort, B.; di Renzo, B.; Merlan, O. Benefits Resulting from the Combined Use of ISO/IEC 15504 with the Information Technology Infrastructure Library (ITIL). In Proceedings of the International Conference on Product Focused Software Process Improvement, Oulu, Finland, 20–22 June 2002; Volume 2559, pp. 314–325.
52. Barafort, B.; Mesquida, A.-L.; Mas, A. Integrating risk management in IT settings from ISO standards and management systems perspectives. *Comput. Stand. Interfaces* **2017**, *54*, 176–185. [CrossRef]
53. Garg, S.; Misra, A. Automation of Incident Management Processes and Benefits of Hosting Servers on Cloud. In Proceedings of the International Conference on Current Trends in Computer, Electrical, Electronics and Communication, Dalian, China, 19–21 December 2017; pp. 700–702.
54. Silva, R.; Pereira, R.; Riberiro, R. Machine learning in incident categorization automation. In Proceedings of the Iberian Conference on Information Systems and Technologies, Caceres, Spain, 13–16 June 2018; pp. 1–6.
55. Heikkinen, S.; Jäntti, M. Identifying IT Service Management Challenges: A Case Study in Two IT Service Provider Companies. In Proceedings of the 2012 23rd International Workshop on Database and Expert Systems Applications, Vienna, Austria, 3–6 September 2012; pp. 55–59.
56. McBride, N. Exploring service issues within the IT organisation: Four mini-case studies. *Int. J. Inf. Manag.* **2009**, *29*, 237–243. [CrossRef]
57. Asrowardi, I.; Putra, S.D.; Subyantoro, E.; Daud, N.H.M. IT Service Management System Measurement using ISO20000-1 and ISO15504-8: Developing a Solution-Mediated Process Assessment Tool to {Enable Transparent and SMS Process Assessment. *IJECE* **2018**, *8*, 4023. [CrossRef]
58. Shrestha, A.; Cater-Steel, A.; Toleman, M.; Tan, W.-G. Building a Software Tool for Transparent and Efficient Process Assessments in IT Service Management. *Int. Conf. Des. Sci. Res. Inf. Syst.* **2014**, *8463*, 241–256.
59. Tan, W.-G.; Cater-Steel, A.; Toleman, M. Implementing IT service management: A case study focussing on critical success factors. *J. Comput. Inf. Syst.* **2010**, *50*, 1–12.
60. Shrestha, A.; Cater-Steel, A.; Toleman, M.; Tan, W.-G. A Decision Support Tool to Define Scope in IT Service Management Process Assessment and Improvement. *Int. Conf. Des. Sci. Res. Inf. Syst.* **2013**, *7939*, 308–323.
61. Deutscher, J.; Felden, C. Model Concept to Determine the Optimal Maturity of IT Service Management Processes. In Proceedings of the 2009 Eighth IEEE/ACIS International Conference on Computer and Information Science, Shanghai, China, 1–3 June 2009; pp. 543–548.
62. Jia, R.; Reich, B.H. IT service climate, antecedents and IT service quality outcomes: Some initial evidence. *J. Strateg. Inf. Syst.* **2013**, *22*, 51–69. [CrossRef]
63. Pereira, R.; da Silva, M.M. A maturity model for implementing ITIL V3 in practice. In Proceedings of the IEEE International Enterprise Distributed Object Computing Workshop, EDOC, Helsinki, Finland, 29 August–2 September 2011; pp. 259–268.
64. De Sousa Pereira, R.F.; da Silva, M.M. A maturity model for implementing ITIL v3. In Proceedings of the 2010 6th World Congress on Services, Miami, FL, USA, 5–10 July 2010; pp. 399–406.
65. Richter, C.; Schaaf, T. A maturity model for tool landscapes of IT service providers. In Proceedings of the 12th IFIP/IEEE International Symposium on Integrated Network Management, Dublin, Ireland, 23–27 May 2011; pp. 1050–1057.

Received: 13 October 2021 | Revised: 9 February 2022 | Accepted: 8 April 2022

DOI: 10.1002/spe.3096

RESEARCH ARTICLE

WILEY

DevOps benefits: A systematic literature review

João Faustino[1] | Daniel Adriano[1] | Ricardo Amaro[1] | Rubén Pereira[1] | Miguel Mira da Silva[2]

[1]Instituto Universitário de Lisboa (ISCTE-IUL), Lisboa, Portugal

[2]Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal

Correspondence
João Faustino, Instituto Universitário de Lisboa (ISCTE-IUL), Lisboa, Portugal.
Email: jpcfo11@iscte-iul.pt

Abstract

Among current IT work cultures, DevOps stands out as one of the most adopted worldwide. The focus of this culture is on bridging the gap between development and operations teams, enabling collaborative effort toward quickly producing software, without sacrificing its quality and support. DevOps is used to tackle a variety of issues; as such, there are differing benefits reported by authors when performing their analysis. For this research, we aim to reach consensus on the DevOps benefits reported in existing literature. To accomplish this objective, two systematic literature reviews. The first intends to find all benefits reported in the literature, while the second review will be used to map the benefits found in the first one with DevOps implementation case studies, providing empirical evidences of each benefit. To strengthen the results, the concept-centric approach is used. During this research it was possible to observe that the most reported benefits are aligned with the DevOps premises of better collaboration between developers and operators, delivering software and products quicker. Based on DevOps implementation case studies, most reported benefits include a faster time to market as well as improvements in synergy and automation. Less reported benefits include a reduction in failed changes and security issues.

KEYWORDS

agile, benefits, case study, DevOps, systematic literature review

1 | INTRODUCTION

With recent technological advances, information technology (IT) departments have taken an increasingly strategic role in organizations[1] given the importance of IT in helping the accomplishment of business objectives.[2] Several disciplines like IT governance and IT service management (ITSM) have built mechanisms and processes so that both IT and business can be aligned in terms of aims and expectations, helping organizations satisfy their objectives.[3–5]

Just as organizations' strategic view has evolved, software development lifecycles (SDLC) have also matured to satisfy current demands. To face the great changes observed in the modern-day markets, businesses need to have greater speed and flexibility. This translates to challenges for IT departments worldwide.[6] As stated, the SDLC has been evolving, no longer strictly focusing on the performance of its own processes, as seen on traditional software development methodologies like waterfall,[7,8] but on the iterations and relationships between the intervenient on the SDLC process and the value that the software can bring to the business.[9] These kinds of software development methodologies are considered agile methodologies and follow the "Agile Manifesto."[10] Even though Business and IT development are brought closer, a gap is still observed within the IT department's development and operations teams.[11] The major issue between these two teams are the different objectives for each team, IT development team is focused on delivering new features or products, while the IT operators are focused on the stability.[12] It's believed by introducing changes on systems would lead to instability.[13] however due to the constant market changes, IT of the organizations need to evolve into new functionalities.[14] A DevOps culture has emerged to address this gap. The DevOps word itself comes from the junction of two other words: development and operations.[15]

DevOps has been adopted across the globe and new research articles flourish. Several studies have reported practices, benefits, and challenges however not always in a structured, clear, and concise way.[16–18] In literature, one can find studies that synthesize DevOps practices, as for example, deployment automation[19–21]; however, it presently lacks research that specifically synthesize its benefits and challenges, guiding professionals in what they may expect during and after DevOps practice implementation.[22] Lack of willingness to share can be a challenge to DevOps implementation.[23]

Being a contemporary topic, with both theoretical and empirical studies found in literature, this research aims to synthesize the benefits organizations may expect with DevOps implementation and how to achieve them. Being said, by synthetizing the DevOps benefits, this research also provides which problems organizations faced before the DevOps adoption and what was the benefit achieved after that. This will help organizations to know what problems could be fixed by the DevOps adoption. The adopted research methodology will include two systematic literature reviews (SLR).

2 | RESEARCH BACKGROUND

The term DevOps started to be researched after Patrick Debois introduced it at a conference entitled "Agile Infrastructure and Operations" in 2008.[24] A DevOps culture aims to bridge the gap between IT development and IT operations, who support applications after they are delivered to production.[25] The focus of DevOps is on improving communication, collaboration, and synergy of IT teams,[26,27] enabling the continuous development and enhancement of applications to meet both market changes and the dynamic needs of the business.[28,29]

In order to achieve said objectives, DevOps builds a foundation in the following areas: culture, automation, lean, measurement and sharing.[6] By looking at Wiedemann et al.'s[6] work, one may note that of the perspectives presented above, *people* play an important role for culture and sharing. Willingness to share is needed, allowing for colleagues and team members to learn and improve their knowledge and experiences. On automation and measurement, one can state that technological tools are the main factor; tools that are used to improve performance, automating what is being done manually, removing the element of human error, and be used to measure tasks and find improvements.[30] Process optimization is a focal point for lean methodologies. They are used in DevOps to identify opportunities for process enhancement, leveraging feedback loops between a its main actors. In later studies,[31] people, technology and processes are considered the cornerstones of DevOps.

Since 2001, organizations have adopted agile methodologies for its SDLC[32] where the most implemented methodologies are XP and SCRUM.[33] These methodologies are the foundation of DevOps and DevOps can be seen as its extension, since they are based on the same principles of introducing short release cycles and to develop forward the customer or user feedback.[34] However, DevOps includes the operations team on early stages of the software development, being able to develop the software already with the operations team input, thus developing software more stable including the business feedback.[35] Also, DevOps stands out due to the collection of techniques and tool to enable software continuous delivery, clearing the path of the software to production.[36,37]

In conclusion, a DevOps culture seems to be very attractive to organizations worldwide, being based in a "*The faster you fail, the faster you recover*" philosophy[14(p1)], enabling a culture of experimentation to release new products, services and software, allowing the organization to grow and to satisfy their customers.[38]

3 | RESEARCH METHODOLOGY

To achieve this research goal, the authors have chosen the systematic literature review (SLR) methodology. It is seen as one of the most widely used research methods to collect and synthesize evidence.[39] SLR's are meant to have a well-defined process to identify, evaluate, and interpret all the evidence collected during research.[40,41] Thus, for this investigation, the authors have followed the framework proposed by Kitchenham[40] where the SLR is split into three stages (Figure 1). Moreover, to add rigour to this research, the authors have chosen to perform two SLR's: the first to find all the benefits

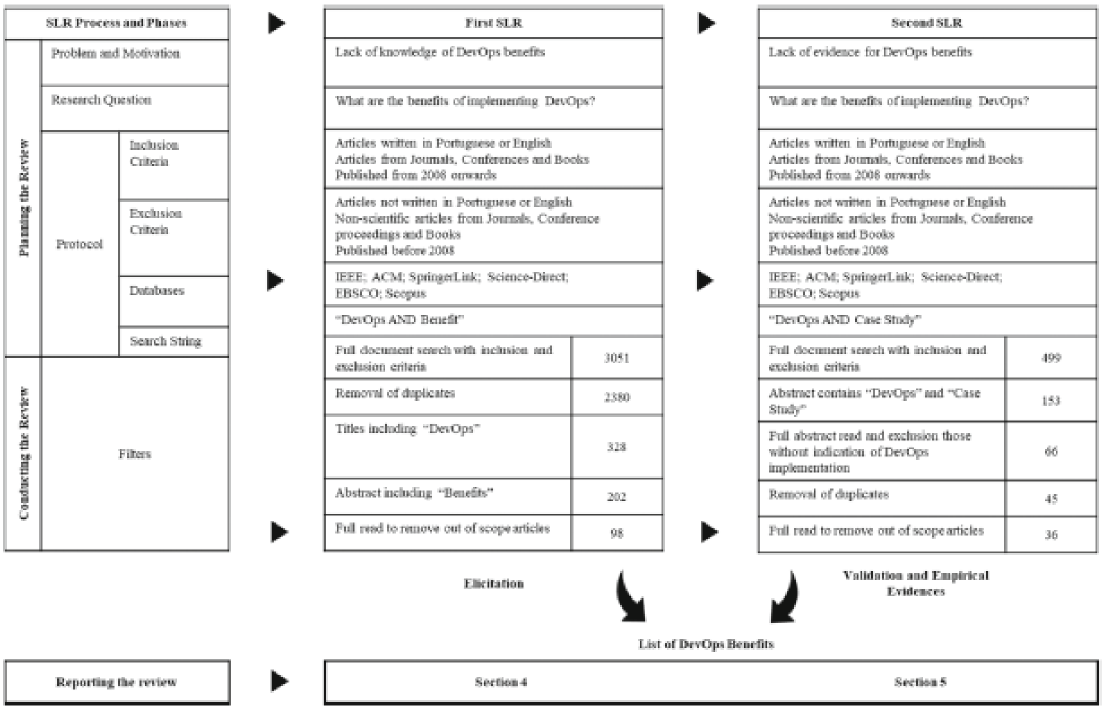


FIGURE 1 Steps performed in each of the performed SLRs

described on existing literature, while the second one being used to find instances of those benefits being reported on case studies from DevOps implementation. This second SLR will confirm and evaluate the findings from the first SLR, where all the DevOps benefits were gathered from literature. The authors believe that searching for case studies is a reliable method of evaluation given that these are a research methodology known by providing evidence of a certain phenomenon.[42] The first SLR was conducted between May and September 2020, while the second SLR was carried out between August and October 2020.

The process designed by Kitchenham[40] was followed by both SLR's. The authors have started by defining the Problem and Motivation for the review. For the first SLR where the expected result was to identify the reported benefits on the literature, the motivation was to acknowledge the DevOps benefits on the literature, while for the second SLR the motivation was to find evidence of the DevOps benefits. The next step of the process was to define the research question (RQ) for the review. In this case, the same RQ was identified for both SLRs "What are the benefits of implementing DevOps?"

After the RQ definition, the next step is to define a protocol where inclusion and exclusion criteria was defined, along with the search databases and the search string of each SLR. The inclusion and exclusion criteria were based on the language of the publications, scientific publications, and publication date. Regarding inclusion and exclusion criteria (IEC) a minimum date was set, considering that the DevOps concept was born after the aforementioned "Agile Infrastructure and Operations" conference, in 2008.[24] For the databases, the authors have used some of the most known and used databases on the scientific community. All these criteria were the same for both SLRs except the search string. For the first SLR the search string was focused on DevOps benefits while for the second SLR the search string was focused on DevOps case studies.

After applying these criteria, some filters were added to the review to exclude publications that wouldn't provide the necessary information for this research. One example of the used filters was the removal of duplicated. All this process definition can be seen with more detail in Figure 1 for both SLRs.

To evaluate the research subject's trend, the researchers have analyzed the date of publication of each relevant piece of literature from a chronological point of view. This is extremely helpful to prove that the research topic has a corresponding trend and is largely demanded by the market. The researchers used the concept-centric approach[43] to better synthesize and analyze the concepts elicited from the literature. This helps to understand the focus of the review, for a better understanding of the readers. Also, it helps the researchers to structure the review. The usage of the concept-centric approach can be seen in Section 4 where the concepts identified are the benefits found per reference, while in Section 5 one can see the case studies identified per reference.

4 | FIRST SLR: LIST OF BENEFITS

After performing the first SLR and analyzing the articles, the list of DevOps benefits was elicited and can be seen in Table 1. The concept-centric approach taken by the researchers can be found in Appendix A, where it is possible to see the match between the concepts and the authors that have identified those concepts in literature.

In the next section, one can see a discussion and some conclusions about the benefits found on the literature, regarding Table 1. After the full read of the publications, the authors have identified the benefits described on the publications and grouped those benefits by the concepts, also seen on Table 1.

Several authors among literature claim to see an improvement on the communication and collaboration (as seen in Appendix A) between developers and operators,[22,28,44,45] creating a synergetic environment where both teams desire to work together toward accomplishing overall objectives.[11,46]

Before DevOps, operators and developers may have had different mindsets when facing change. With the disappearing of the waterfall software development methodology and the emergence of the "Agile Manifesto",[10,47] the developer's mindset shifted to deliver more features as fast as possible to production, while the operator's continued to have the mindset of guaranteeing the stability of the systems it was solely responsible for.[12] These divergent views on change typically lead to finger-pointing, with operators blaming developers for the production impact of deployments when they might have been involved earlier in the development process to try to anticipate possible problems before they reached production.[48]

TABLE 1 List of benefits identified in literature

Concept ID	Benefits	# References
B01	Cross team collaboration and communication	49
B02	Faster time to market	41
B03	Faster and better feedback loops	38
B04	Increase of code quality	32
B05	Increase of value	26
B06	Improvement of system reliability	22
B07	Less mean time to recover	17
B08	Increase of team performance	17
B09	Costs reduction	13
B10	Processes and tools standardization	13
B11	Maximization of competences	13
B12	Decrease of manual work	11
B13	Increase of customer satisfaction	11
B14	Less failed changes	11
B15	Increase of employees motivation	9
B16	More innovation	8
B17	Better deployment management	5
B18	Less security issues	5
B19	Organizational cultural changes	41

Because of the resultant DevOps synergy, both operators and developers are more driven to collaborate across teams. They will feel that they are working toward a common and greater goal for everyone.[49] However, this can also be extended to the business. Just as Agile practices and principles brought business and developers together,[10] DevOps introduces operators into the mix, emphasizing the significance of operations management in the organization.[50]

As seen in Appendix A, faster time to market, related to continuous integration and continuous delivery capabilities, is one of the most reported benefits from DevOps. Organizations can design new, better features for their products as a result of faster feedback.[51] Through DevOps enabled automation they are then able to put said features into the market at a quicker rate than competition.[52]

It is noteworthy to mention that various authors were able to identify the different sources that contribute for a better and faster feedback under DevOps: customers and end users (business users)[14] as well as between the DevOps team itself.[49] Customers and end users are those who use the application; they are best to identify issues and potential improvements.[53,54] DevOps has a practice to shorten the feedback loops between operators and developers, which also leads to faster feedback when something is going wrong and requires further work.[55]

Improved feedback does not only contribute for better development and application stability, but also leads to opportunities for DevOps teams to learn about its components (for example, operators can learn about the development process, and developers learn about processes which guide operators work) as well as the business itself.[56]

Code quality can be increased as a result of implementing improved delivery pipelines under which code is built into packages and introduced to the respective repository after each check.[57] During the packaging of a new build, code can be submitted through quality gates, ensuring that best practices defined for that application are being adhered to.[20] Due to the continuous integration capability encouraged by DevOps culture, developers from several teams will be working in collaboration with other developers. There will be opportunities to find issues or needed improvements to other developers' code, improving the overall code quality of applications.[58]

There is great consensus in literature about the increase of value when using DevOps. DevOps is a culture that uses lean and agile practices. DevOps phenomenon arose as an extension of agile software development inspired by lean concepts.[59] The first Agile Manifesto principle is about value: "*Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.*"[10] Due to the continuous delivery capability that DevOps employs, shorter development and release cycles[60] can be achieved, where business and customers will notice the ongoing improvement of software, realizing the continuous increase in value of their applications.[29]

Automation brings an additional benefit in the ability to perform defined actions after an event is triggered by automated monitoring.[61] By automating infrastructure using infrastructure as code, the availability and reliability of applications will be improved. Such infrastructure can scale up and scale down according to its reported usage and demand.[14]

Related to the faster time to market benefit, feedback and automation is not only used to deliver new or improved products.[62] Due to the premises of DevOps in having a solid IT team, both developers and operations work together to guarantee that fixes are deployed in production instantly.[63] This contributes to the stability and availability of applications, so that defects do not cause greater impact.[23]

The objective of implementing any framework, practice or methodology is to improve performance. However, operators and developers commonly have different objectives and use different metrics to measure their performance, as explained in Section 1. Thus, to improve a team's performance, an alignment is needed for the definition of clear and visible goals.[64] In case of operations, these would be aimed at the stability and reliability of an application, while for developers the focus should be on the features delivered for it.[44,65] Since DevOps is also targeted toward using lean and agile practices,[59] it concentrates its aim on improving people, processes and technologies capabilities, specifically in the way the work in process is limited and done in small batches, therefore contributing to the well-being of their teams.[29]

Cost reduction is among the top goals of every organization in the world. As discussed before, DevOps can help reduce costs by reducing bottlenecks in the SDLC, optimizing time to deploy changes in production and enabling better resource management.[22] This can help balance software quality with costs, helping organizations to have an increased return on investment.[66,67]

To optimize the SDLC it is essential that operations can react quickly, helping developers have their environments stable, up and running. DevOps encourages operators to use the infrastructure as a code capability in order to help manage and configure environments more quickly and in a standardized way.[68,69] This allows developers to have development and preproduction environments, which aids in the discovery of issues early in the SDLC.[70] Likewise, the environment process configuration and tools used by each team should be standardized, avoiding common situations like "it was working on my machine."[71]

With the mixing of IT teams by making developers and operators work together, the competences of these resources will be increased.[25] Developers will be able to get abilities that most often regard to operations, while operators will get abilities on areas of development.[65] This also contributes for improved knowledge management,[22] allowing a DevOps team to be more complete in terms of their joint competences.

On a decrease of manual work a consensus can be concluded in literature. This is accomplished by using automation. There are three major areas where it can be applied: testing,[72] the delivery pipeline,[73] and configuration or provisioning.[74] Test scripts can be automated by using tools that will perform the actions of the testers, verifying if the final output is the desired one. Thus, this capability reduces the manual work of testers as well as the risk for human error. Moreover, automated tests enable continuous testing capability which helps find integration issues earlier in the development cycle, making defect resolution faster and with less impact on production environments.[72] This also frees up the tester to create other, exploratory testing activities.

Operations are usually not only responsible to guarantee the stability of production environments but also of the lower environments. If a development team requires several development environments, each requiring operators to configure manually, a blocking of development resources may occur. DevOps encourages the usage of infrastructure-as-code to allow the operators to manage their infrastructure and environment configurations by using code, replicating said configurations for several alternative environments, speeding up configuration.[70] Furthermore, it is possible to automatically provision environments with resources based on predetermined thresholds, guaranteeing their stability and availability.[67]

DevOps encourages developers to continuously integrate their code so that issues can be found earlier.[75] However, this requires a lot of work if every time a developer checks-in his code, a manual package needs creation for other developers to review. Under DevOps, every time that a developer commits code to a code repository, a script is triggered that will automatically test and create a package or artifact, checking and giving immediate feedback if there is any error and, if successful, storing it properly.[56] From this point onwards, the developed package can be used for installation across all environments. With the package stored, one could also trigger a script that will deploy the package with new code in a test environment, making it available for testers; alternatively, once the deployment is completed, more complex automated testing can be triggered, like integration or end-to-end tests and developers informed if the new code failed in any test, speeding up the bug fixing and increasing the software's stability.[22,76]

Customer satisfaction can be seen as a consequence, resulting from a variety of previously described benefits. Since DevOps will continuously improve the stability of applications while reaching for customer feedback, customer satisfaction will increase.[22] By reducing bottlenecks on the SDLC process, the customers' feedback is deployed on the application faster, further increasing satisfaction.[77] Also, looking from a perspective in which a customer is internal, DevOps can also contribute to cost reduction.

Less failed changes can be seen as a consequence from both the standardization of processes and tools, as well as from other DevOps capabilities in general. With a standardization of processes, like those used in a deployment, for example, issues on a deployment script can be found and fixed in other environment, before reaching a production deployment.[22] With all the automation (in testing areas, for example) and continuous integration that DevOps encourages, help is obtained toward identifying issues with development work earlier on the SDLC, helping to avoid failed changes when moving to production.[78]

Employees of an organization will feel more motivated by working on a more communicative environment, in which they feel that their team will back them up.[12] This will contribute to reduced blame-games between developers and operators.[54] Due to the sharing culture that DevOps promotes, developers will learn about operators' tasks just as much as operators will learn about developers' tasks. Thus, employees will be more capable to backup each other up on different types of work.[15,79]

Due to the increased speed of development, and by enabling a faster time to market, DevOps allows organizations to experiment new solutions, features and products[29] without incurring in significant economic impacts. Start-up companies are known for creating new market segments due to the innovative solutions they create. DevOps brings a great opportunity for these organizations, which does not have much revenue, allowing a spirit of the "*Faster you fail, faster you recover*"[14(p1)].

The setup of IT Teams before DevOps were structured in a way that deployments were manually performed by single or multiple operation teams that had the responsibility for configuring and setting up production environments, database configuration, backups of software build and reversing bad builds on the new software.[56] This raises the possibility and concern of human errors, which can impact the entire service stack of an organization.[69] Automation is one of the most used capabilities in DevOps which can help on this matter. By building automatic deployment mechanisms it is possible to decrease the volume of potential outages from applications.[65] Moreover, DevOps gives the ability for developers to perform their own deployments under the motto "You build it, you run it",[73] which empowers developers to find bad builds before operators, resulting in improved deployment management.

DevOps promotes monitoring during the entire deployment pipeline, using tools to notify developers and operators in case of something going wrong, or the need for manual actions, like rolling back the software to a previous version,[63] contributing also to a better deployment management.

DevOps is usually allied with cloud implementations which help deploy security integration and carry out penetration tests between applications.[80,81] Nowadays, cloud providers offer services that promote the usage of DevOps, in which a security model for their customers is ensured.[82]

As discussed earlier, DevOps is not only focused on automating processes and improved performance, but also on cross team collaboration and interaction between people. For DevOps, or other agile software development methodologies, organizations need to have a culture that allows for these interactions. Lean, Agile, and DevOps appeared in various times to meet various requirements,[83] but they all concentrate on organizational culture by forming interdisciplinary teams, cutting waste, concentrating on the customer, embracing change, and providing value on a continual basis. Under DevOps, sharing is the key for operators and developers to work together. As such, organizational culture needs to be adapted to promote this kind of involvement.[76]

5 | SECOND SLR: EMPIRICAL EVIDENCES OF DEVOPS BENEFITS

A second SLR was carried out to confirm and evaluate the findings from the first SLR, in which all DevOps benefits were gathered from literature. To do so, the authors captured and analyzed a total of 36 DevOps implementation case studies. Each of the studies was read for data on the outcomes of introducing DevOps capabilities in a business environment. A list of these articles, their references, and basic vectors of analysis, are found on Table 2.

Due to the data provided in Table 2, it was possible to produce Figure 2 with a segregation of the case studies by continent, country, and business sector. It is possible to see that DevOps is more present in Europe or on multinational organizations that work in several countries from multiple continents. Regarding the business sector, the IT business sector clearly stands out from the other sectors. Since DevOps is a culture that is focused on IT developers and operators, it makes sense that IT organizations implement this culture before other sectors. However, from professional experience from the authors, the DevOps culture have been expanding on the financial sector (banking and insurance).

Having identified and analyzed the final list of DevOps implementation articles, we proceeded to map business benefit concept IDs to case studies in which they are mentioned. Some of the documents included findings from more than one case study; for these, we relied on decimals to differentiate implementation results from each organization as much as we possibly could. However, it is important to note that some authors merged in a single body the observations and results of multiple, different DevOps case studies, making full differentiation impossible. In total, 69 case studies were identified and reviewed as part of our research. The results of this effort are presented in Table 3 (also refer to Appendix B). Moreover, one of the case studies didn't presented any benefit, where the authors have identified to study the benefits as their own future work. The benefit ID and benefit description columns are referring to the concepts previously presented in Table 1. Lastly, it is relevant to add that most of the case studies did not provide quantitative evidence of these benefits, but often referred to them in a qualitative manner.

An improvement in the rate by which new development is produced, deployed, and reaches the market was by a considerable margin the most widely and explicitly observed benefit of a DevOps adoption. The implementation of DevOps practices, particularly when it comes to establishing a bridge between development and operations teams,[91] was commonly pointed out as an enabling factor toward faster delivery.[108] The added flexibility associated with DevOps practices allows for new software evolutions to be implemented faster, while sustaining a quality standard.[75] Shorter response times[53] and increased deployment speed[104] are likely to be observed in a successful DevOps integration. In Luz et al.'s[28] study it is stated that "after the DevOps adoption, it was possible to make 29 deployments on a single day" whereas before, due to rigid and conflicting policies at the operational level, deployment were only scheduled to occur once, weekly.

As established, the development of synergies between teams is a foundational principle for applying DevOps practices. From our research, improved collaboration and communication between developers and operational staff was a frequently reported benefit resulting from DevOps implementation. Increased awareness of the overall software development processes, standard deployment practices and service management took place[69] as teams abandoned traditional

TABLE 2 List of DevOps implementation case studies analyzed

ID	Reference	Country	Continent	Business sector
CS.01	84	Sweden	Europe	Information technology
CS.02	85	Spain	Europe	Human resources
CS.03	86	Italy	Europe	Lighting business
CS.04	56	Denmark	Europe	Information technology
CS.05	28	Brazil	South America	Government organization
CS.06	64	Morocco	Africa	Information technology
CS.07	87	Montenegro	Europe	Banking industry
CS.08	88	Germany	Europe	Information technology
CS.09	55	USA	North America	Information technology
CS.10	89	Multinational	Multinational	Healthcare
CS.11	90	USA	North America	University project
CS.12	91	Finland	Europe	Information technology
CS.13	92	New Zealand	Oceania	Finance & insurance industry
CS.14	93	USA	North America	Government organization
CS.15	68	Spain	Europe	Information technology
CS.16	53	Multinational	Multinational	Mixed
CS.17	15	Finland	Europe	Information technology
CS.18	94	Australia	Oceania	Information technology
CS.19	95	Finland	Europe	Information technology
CS.20	96	USA	North America	Information technology
CS.21	18	Multinational	Multinational	Information technology
CS.22	97	N/A	Europe	Information technology
CS.23	98	Multinational	Multinational	Mixed
CS.24	75	(Not provided)	(Not provided)	Finance & insurance industry
CS.25	99	(Not provided)	(Not provided)	Information technology
CS.26	23	Multinational	Multinational	Mixed
CS.27	100	Multinational	Multinational	Information technology
CS.28	101	UK	Europe	Information technology
CS.29	69	Multinational	Multinational	Information technology
CS.30	102	Spain	Europe	University project
CS.31	103	USA	North America	Government organization
CS.32	104	Multinational	Multinational	Information technology
CS.33	105	Sweden	Europe	Information technology
CS.34	106	Finland	Europe	Information technology
CS.35	107	Germany	Europe	Information technology
CS.36	67	(Not provided)	(Not provided)	Information technology

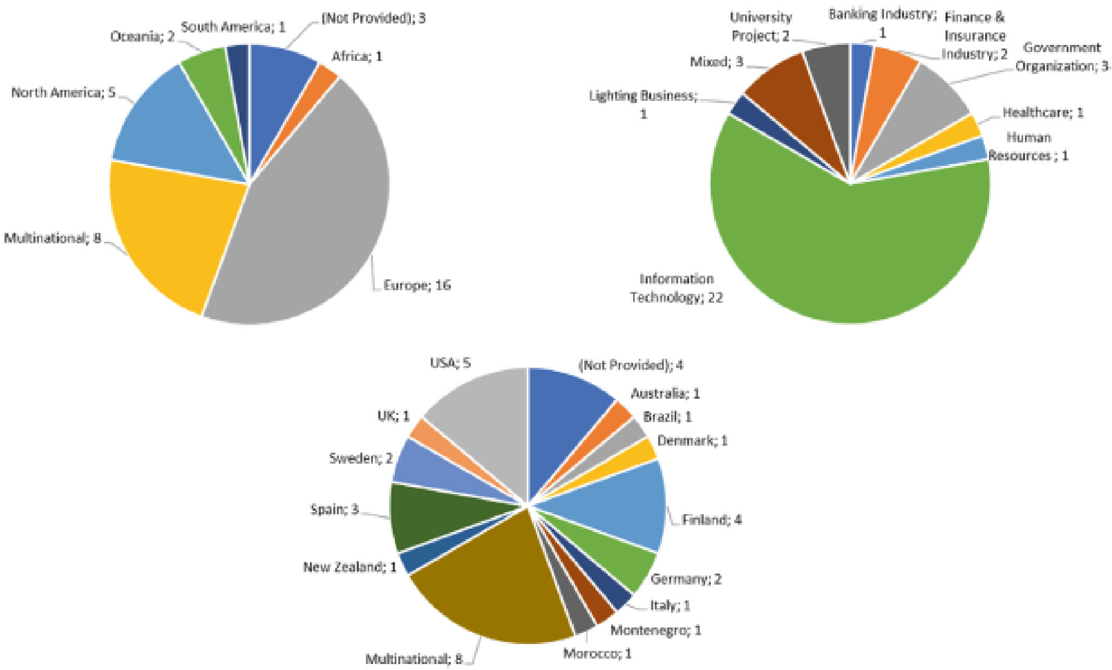


FIGURE 2 Case studies segregation

TABLE 3 DevOps benefits analysis

Benefit ID (from Table 1)	Benefit description (from Table 1)	Occurrences in case studies	Percentage of case studies
B02	Faster time to market	49	71%
B01	Cross team collaboration and communication	39	57%
B12	Decrease of manual work	38	55%
B08	Increase of team performance	30	43%
B04	Increase of code quality	27	39%
B17	Better deployment management	25	36%
B06	Improvement of system reliability	23	33%
B03	Faster and better feedback	22	32%
B10	Processes and tools standardization	19	28%
B11	Maximization of competences	20	29%
B13	Increase of customer satisfaction	19	28%
B15	Increase of employees motivation	18	26%
B09	Costs reduction	12	17%
B07	Less mean time to recover	10	14%
B19	Organizational cultural changes	9	13%
B16	More innovation	6	9%
B05	Increase of value	6	9%
B14	Less failed changes	5	7%
B18	Less security issues	2	3%

"work silos" in favor of DevOps.[91] Furthermore, this "empowerment of teamwork"[96] between development and operations seems to heavily tie in with other business benefits ranging from improved reliability, quality, and security[15] to competence maximization, innovation and employee motivation. Referring to Shahin's[104] work, in interviews that were held with participants of a DevOps implementation study, the opportunity of learning about overall operational and architecture aspects was often commented as a deeply useful and "growing exercise."

DevOps practices emphasize automation over manual work in the development, testing and deployment of software.[23] Over 50% of the reviewed case studies clearly mention a reduction in the volume of manual tasks. For example, in Laukkanen et al's[105] study, "manual test[ing] had been the bottleneck" for reducing feature freeze periods; with DevOps implemented, release tests for specific systems were automated, causing a reduction in the time necessary for completion. Luz et al.[28] also describe how before having DevOps implemented a vast majority of automatable tasks were done manually, often causing errors and need for rework. Similar to what we observed in our analysis of B01 (cross team collaboration and communication), the benefit of reducing manual work appears to tie in with faster delivery,[14] less failed changes, improved code quality and even employee motivation, as was observed in Gupta et al.'s[89] case study. Here, teams focused on incremental automation, focusing on a single, critical workflow at a time; upon reviewing progress, it is stated that "such small successes motivated the team", encouraging them to pursue further automation.

Although increase of value (B05), less failed changes (B14) and less security issues (B18) were not commonly and explicitly discussed in the analyzed case studies, there is room for further investigation toward better understanding how business benefits can relate to each other. Despite said links not being subject to investigation under the present research, it may not be unreasonable to consider that organizations who increase release rates and quicken their time to market (B02) are in a better position to deliver greater value to stakeholders (B05); or that those who significantly improve communication and collaboration between developers and operations (B01) may also observe a reduction in failed changes or release faults (B14).

6 | RESULTS AND DISCUSSION OF DEVOPS EMPIRICAL EVIDENCES

Table 4 presents examples for each of the 19 business benefits identified as part of our research. Where applicable, cells referring to the "Problem Solved" column are also filled in, indicating the motivation or reasoning that led to the implementation of DevOps, which led to the observed benefits. This section shows that DevOps can solve different problems on the organizations, indicating an empirical evidence of the benefits got after the DevOps implementation.

7 | CHALLENGES IN DEVOPS ADOPTION: THE OTHER SIDE OF THE COIN

Even thought, this research is about the benefits of the DevOps culture adoption, the main objective is to show what to expect when adopting DevOps. Thus, for this article some DevOps challenges will also be presented, since some of the researchers that identified DevOps benefits, were also able to identify challenges to the DevOps implementation. In Table 5, one can see which challenges were identified by the researchers that also identified benefits.

As it can be seen, some of the challenges shown in Table 5, are more related with the culture, environment, and business industry where the DevOps culture is being implemented, rather than the technologic point of view of DevOps, such as "Insufficient communication," "Deep-seated company culture" and "Industry constraints." This shows that when an organization is thinking to adopt DevOps, should self-assess if it is culturally ready for this change. Moreover, to help to mitigate this challenge, the top management of the organization should be propelling for this change so it could be example for the rest of the organization.[25] But there is a technologic challenge regarding the automation of the deployment scripts for several technologies. Organizations have multiple applications, where each of them can have different coding languages which needs its own deployment script. This requires a lot of different skills for DevOps to be able to automate these different deployment scripts.

DevOps has been evolving constantly, which could help regarding the challenge "DevOps is unclear but also evolving." The amount of publications shows that the DevOps adoption has been growing over the time, showing that organizations have been able to understand how to implement DevOps.

TABLE 4 Case study analysis: DevOps benefit and problem solved

Benefit	Example of benefit	Problem solved	Case study
(B01)–Cross team collaboration and communication	*"The inclusion of operation team members and operation topics help the operation team to know the development topics and plan their readiness accordingly. Additionally, they take building knowledge and feedback for risk assessment without additional effort."*	*"We soon realized that with the current approach we would not be able to release the first couple of version increments. Team members in India and USA have experience in traditional software development and product management group in Germany has no experience in software development."*	CS.10
(B02)–Faster time to market	*The organization achieved a one deployment per week frequency, with one hour / one day lead time for changes.*	*"The organization size, the diversity of its departments (development, operations, security, service, QA, architecture, etc.) as well as the interaction between them, and the complexity of its processes, hampered reducing time to market, and made this company less competitive"*	CS.15
(B03)–Faster and better feedback	*"The flexibility afforded by the DevOps approach allowed the development teams to recognize, characterize and accommodate- date changes to DART's control algorithms for NEXT-C in real time. The team was able to update the test specifications and procedures in real time, and ultimately achieve the goal of demonstrating NEXT-C at Technology Readiness Level."*	*"While NEXT-C was well characterized from its own development and test perspective, there were unknowns in the specifics of DART's tailored use-case for the thruster."*	CS.14
(B04)–Increase of code quality	*"Higher levels of automation were found to drive improved quality assurance. (…) The automated DevOps production pipeline helps to ensure that every change is verified before it is pushed forward for delivery."*	*Description of a Problem / Motivation was not provided.*	CS.12
(B05)–Increase of value	*"Increase in deployment frequency from about 30 releases a month to an average of 120 releases per month."*	*"Need for a change by the business in order to remain agile and competitive. (…) Prior to DevOps, the company had been maintaining and developing its aging monolith application that was hosted in a traditional data center."*	CS.13
(B06)–Improvement of system reliability	*"The time spent in the queue for the Basic approach is about 330 times that of the Containerized approach, and similarly the queue time using the Hosted agent is 1,110 times that of the Containerized approach, which translates to significant time saved. Since all of the infrastructure is managed without any new cost incurred, yet the throughput is high, our CI/CD pipeline is very lean."*	*"We recently decided to move toward a micro-services-based architecture (…) Consequently, the number of build and release definitions would increase significantly, and the infrastructure that was utilized may no longer be sufficient."*	CS.8
(B07)–Less mean time to recover	*"This case study illustrates how rapid and simple its deployment was, in accordance with the DevOps principles, and therefore focusing on how self-service monitoring infrastructure for threats detection provided engineers—both developers and IT operators—fast and continuous feedback of the Library Energy-Efficiency System deployed into production. (…) it provides evidence of how this cybersecurity monitoring infrastructure enabled to detect threats, such as denial attacks, and helped to better anticipate spoofing problems."*	*"The development and deployment of such systems [IoT] into production as well as their operation and monitoring are highly complex due to the heterogeneity of delivery endpoints. (…) The Cluster of European Projects on Software Engineering for Services and Applications highlights the importance of ensuring Quality of Service (QoS) and correctness of IoT systems together with the complexity of such purpose as devices and software could change for various reasons such as bugs and failures, changing interfaces and implementations, and changing requirements."*	CS.30

(Continues)

TABLE 4 (Continued)

Benefit	Example of benefit	Problem solved	Case study
(B08)–Increase of team performance	*"Considering the e-TCE system, after the DevOps adoption, it was possible to make 29 deployments on a single day. Before the DevOps adoption, and due to the rigid policies of the operations team, the deployments were schedule to occur once a week."*	*"Before DevOps, deployment activities were historically a controversial point at the TCU. Several conflicts occurred over time. Rigid procedures were created to try to avoid problems."*	CS.5
(B09)–Costs reduction	*"Most companies confirm that DevOps brings shorter response time and more frequent deployments, higher productivity, better feedback from the client and lower IT cost."*	*Description of a Problem / Motivation was not provided.*	CS.16
(B10)–Processes and tools standardization	*"Although having simpler deployment pipeline for each component or service can bring a lot of benefits but the requirement of a dedicated pipeline needs extra effort to set up the dedicated pipelines for the first time. Some of the participants reported that they were employing automation technologies such as Docker to simplify the deployment process."*	*"Our analysis of the data revealed that it was challenging for a couple of practitioners to design applications for different operations environments, in which they may have had difficulty to make consistency in a set of heterogeneous operations environments"*	CS.32
(B11)–Maximization of competences	*"The advantage is that the DevOps team teaches the student the necessary activities and attempt to integrate him\her into the team. There are no educational programs, for example, from the university that teach all necessary competencies that are required to work in a DevOps team. Hence, companies train their students or team members to get ready for the role."*	*"In the traditional silo organized IT department, there is a high level of specialist knowledge. However, in the DevOps setups, these departments are linked, and the human capitals move from highly specialized to more generalized knowledge."*	CS.26
(B12)–Decrease of manual work	*"Overall, developers are able to perform the defect validations much more quickly without having to wait to manually configure the hardware with latest software bundles having their fix in it. With this automation, developers have full control – to validate any defect they have to just pick and choose the config and within few clicks they will have a setup up and running on which, they can validate the defect in production like environment."*	*"No organizations can afford to live with manual, error prone and repeated activities in the software delivery lifecycle (…) the project teams identify this precise business need and adopt DevOps to optimize their processes, it is going to reap more fruits."*	CS.36
(B13)–Increase of customer satisfaction	*"The more and faster development team adds new features, more citizens visit the website or in the web application. (…) The deliverables may be released daily or at the end of the release cycle time. Subsequently, the development team gains faster feedback from end-users that would help in mitigating several risks"*	*Description of a Problem / Motivation was not provided.*	CS.23
(B14)–Less failed changes	*"Because every change in the code is checked at every stage of the development, and errors are discovered and resolved on the fly, the end products have fewer bugs, and the software can be readily released."*	*Description of a Problem / Motivation was not provided.*	CS.12
(B15)–Increase of employees motivation	*"The instantiation of the role rotation in the cross-functional DRR practice in our case enabled large-scale learning and KS since all team members were able to perform several roles and become more knowledgeable. (…) When team members rotate, they can take on responsibilities, develop skills, and acquire knowledge. This fosters the team's autonomy."*	*Cross-functional collaboration and team self-organization were described as major challenges.*	CS.25

(Continues)

TABLE 4 (Continued)

Benefit	Example of benefit	Problem solved	Case study
(B16)–More innovation	*"The single-case study presented in this research was helpful to answer the two research questions. First, DevOps may be considered an approach that contributes to implementing innovation for software-defined business environments, (…) Second, the case shows that (IT) consulting companies need to transform themselves for DevOps."*	*"To develop its own consulting approach, T-Systems MMS initiated a DevOps program, which explicitly aims to improve the company's offering in the area of innovative digital services."*	CS.35
(B17)–Better deployment management	*"This has reduced errors caused by builds with wrong dependencies, incorrect deployment documents, and human errors in general, since only automated processes would deploy in the environments. (…) Initially there will be the impression that some legacy systems and technologies will not be able to be automated or benefitted by the Continuous Delivery process, but in the case of the institution of the case study, even COBOL and Power builder systems have benefited from process automation."*	*"It was identified that the deployment process executed until the beginning of this work required a lot of effort and there was a lot of bureaucracy."*	CS.24
(B18)–Less security issues	*"The success so far shows that organizations with large bureaucratic obstacles and stringent software security and accreditation requirements are able to use (Sec)DevOps processes and toolsets to produce software that meets security and accreditation requirements and ultimately satisfies their customers."*	*"Ensure that security became a continuous practice rather than being tacked on at the end."*	CS.31
(B19)–Organizational cultural changes	*"DevOps culture and mind-set, which were enriched with colocation, were observed in the wider dissemination of DevOps approach across the organization."*	*"Prior to this improvement, the team spent huge efforts in merging code and resolving merge conflicts, which were causing broken builds often."*	CS.17

TABLE 5 DevOps adoption challenges

ID	Challenge	# of references	References
C.01	Industry constraints	2	29 109
C.02	Deep-seated company culture	2	39 29
C.03	Insufficient communication	1	29
C.04	DevOps is unclear	1	29

Every new adoption for an organization takes time to learn, and DevOps is not an exception for it. To adopt DevOps, it is important to give training to the organizations employees so they can understand how to implement DevOps.

8 | VALIDITY OF THE SLRS

The authors have submitted this research to validity tests where the validity is made in four different categories, construct validity, external validity, internal validity and conclusion validity.[110] Zhou et al.[110] have performed a research to synthetize

TABLE 6 Validity tests

Pitfall description	Review test
Nonspecification of SLR's setting and sufficient details Incorrect or incomplete search terms in automatic search Incomprehensive venues or databases Inappropriate inclusion & exclusion criteria	These pitfalls are regarding the planning phase of the review. However, this research has a process and protocol correctly defined describing the decisions for the criteria, databases and search terms used. This shows a path that other researchers can follow to reproduce and replicate this research, adding more validity to this research.
Inadequate size and number of samples	For both SLRs on this study, it was possible to gather a significative amount of publications. From these samples, the authors were able to identify several benefits on the first SLR likewise, on the second review where was possible to identify several DevOps case studies.
Restricted time span	The only time restriction defined was the minimum date of research since DevOps was first presented in 2008.
Bias in study selection	To avoid the bias study selection, the authors have defined filters and criteria to select the studies on the same way for all of them.
Paper/database inaccessible	The databases used are some of the known databases by the academic/scientific and software engineering communities, showing the reliability of these databases.
Primary study duplication	To avoid duplication, the authors have applied a filter to remove duplicated articles.
Bias in data extraction	The several authors of this research have reviewed the data extracted from each author to avoid that some researchers have not identified important data.

the most common pitfalls when performing literature reviews by the different review phases. In Table 6 one can see some of these common pitfalls and how the authors have passed the test for this research.

9 | CONCLUSION

DevOps is a novel culture being adopted worldwide. The authors noticed a lack of synthetization for DevOps implementations benefits in present literature. Thus, the objective for this research was to consolidate the benefits of DevOps implementation so new practitioners know what to expect when adopting the methodology.

To accomplish this objective, the authors have chosen to perform an SLR on the benefits reported in literature. The SLR methodology is known for adding rigor to research due to the well-defined protocol that one must comply to when defining it. Additionally, a second SLR was carried out to find case studies of DevOps implementation. This second SLR was important for research, allowing for the mapping between issues that organizations faced, what were the achieved benefits, and what empirical evidence are there, respectively. Given the accomplishment of the study objective, it is possible to note that this study brings contributions to the theoretical body of knowledge by synthetizing the DevOps implementations benefits.

Regarding the findings originated from this research it is possible to state that even though there was a small number of studies in common between both SLR's, all benefits listed from the first SLR were also found on the second SLR. This demonstrates that empirical evidence exists for said benefits. It was also interesting to note that the top five benefits with more references from the first SLR are not the same as the top five benefits with more occurrences in the second SLR. Of the top five from the first SLR one can find benefits B03 and B05, while on the second SLR one finds benefits B08 and B11. Comparing B05 with B08, the authors can understand that it is easier to measure an improvement in team performance rather than a measure of value increase. As such, it makes sense to find B08 with more occurrences with empirical evidence. Furthermore, when comparing B03 with B11, one can also suppose that all the automation that DevOps encourages makes it easier to record a decrease of manual work, as the effect should be immediate, while faster and better feedback often results from willingness by individuals themselves.

It is possible to see that the most reported benefits are common between the two SLR's. Those benefits are B01 and B02. This is aligned with the premises of DevOps, bridging the gap between developers and operators, working together in delivering software or products faster to their customers.

Regarding the least reported benefit it is possible to see B18 on the bottom of each SLR. It seems that this benefit is related with DevOps, but it is more specifically studied as an own discipline for security, called DevSecOps.

The fact that case study authors did not frequently provide quantitative evidence regarding the observed business benefits did increase the difficulty of establishing fully consolidated findings. This brings the opportunity of future researchers to expose metrics on how to measure the DevOps benefits, to compare how the organizations business units behave with these DevOps benefits. Another limitation to this study is due to the novelty of DevOps, the authors couldn't apply a quality filter on the SLR's for top conferences and top journals, otherwise, the total amount of articles for analysis would be low. As future work, the authors suggest performing a similar study for DevOps, but instead of benefits it could be directed at finding adoption challenges and how to overcome them. The authors believe that combining this research with a study where adoption challenges are tackled would help new DevOps practitioners clarify what is expected to be achieved with DevOps and how to go about its implementation. Moreover, this research would help organization on the decision to implement DevOps, since this research shows the trade-off between challenges and benefits. Furthermore, there may be value in studying to what extent do identify DevOps business benefits can relate to each other, building a potential series of linked, expected improvements for business.

DATA AVAILABILITY STATEMENT

Since this publication is a literature review where the literature is based on several digital libraries, all the data used on this publication can be fetched by checking the references section for each publication.

ORCID

João Faustino https://orcid.org/0000-0002-1743-5348
Daniel Adriano https://orcid.org/0000-0003-2907-855X
Rubén Pereira https://orcid.org/0000-0002-3001-5911
Miguel Mira da Silva https://orcid.org/0000-0002-0489-4465

REFERENCES

1. Ajayi BA, Hussin H. Influence of ITG on organisation performance: the mediating effect of absorptive capacity. Proceedings of the 6th International Conference on Information and Communication Technology for the Muslim World, ICT4M; 2016:1-6. doi:10.1109/ICT4M.2016.13
2. Setyadi R, Fattah A, Waseso B. Trust Effect on Business-IT Governance Alignment in Society Culture (A Case Study in Indonesia). Proceedings of the 2019 7th International Conference on Cyber and IT Service Management, CITSM 2019:1-5. doi:10.1109/CITSM47753.2019.8965411
3. Bartolini C, Salle M, Trastour D. IT service management driven by business objectives An application to incident management. Proceedings of the 2006 IEEE/IFIP Network Operations and Management Symposium NOMS 20066:45-55. doi:10.1109/NOMS.2006.1687537
4. Cavalcante V, Bianchi S, Braz A, Amorin F, Nauata N. Investigating business needs fluctuations on IT delivery operations. Proceedings of the Annual SRII Global Conference, SRII; 2014:19-26. doi:10.1109/SRII.2014.13
5. Setyadi R. Assessing trust variable impact on the information technology governance using business-IT alignment models: a model development study. Proceedings of the ICSECC 2019 - International Conference on Sustainable Engineering and Creative Computing: New Idea, New Innovation, Proceedings; 2019:218-222. doi:10.1109/ICSECC.2019.8907224
6. Wiedemann A, Forsgren N, Wiesche M. Research the practice: the DevOps phenomenon. *Commun ACM*. 2019;62(8):68.
7. Trivedi P, Sharma A. A comparative study between iterative waterfall and incremental software development life cycle model for optimizing the resources using computer simulation. Proceedings of the 2013 2nd International Conference on Information Management in the Knowledge Economy, IMKE 2013:188-194.
8. Michener JR, Clager AT. Mitigating an oxymoron: compliance in a DevOps environments. Proceedings of the International Computer Software and Applications Conference; Vol. 1, 2016:396-398. doi:10.1109/COMPSAC.2016.155
9. Chen L. Continuous delivery: overcoming adoption challenges. *J Syst Softw*. 2017;128:72-86. doi:10.1016/j.jss.2017.02.013
10. Beck K, Beedle M, van Bennekum A, et al. Agile manifesto; 2001. Accessed January 9, 2020. https://agilemanifesto.org/

11. Katal A, Bajoria V, Dahiya S. DevOps: bridging the gap between development and operations. Proceedings of the 3rd International Conference on Computing Methodologies and Communication, ICCMC 2019; 2019:1-7; IEEE. doi:10.1109/ICCMC.2019.8819631
12. Hussaini SW. A systemic approach to re-inforce development and operations functions in delivering an organizational program. *Complex Adaptive Systems*. Vol 61. Elsevier Masson SAS; 2015:261-266. doi:10.1016/j.procs.2015.09.209
13. Rance S. In: Hanna A, ed. *ITIL 2011 Service Transition*. 1st ed. The Stationery Office; 2011.
14. Soni M. End to end automation on cloud with build pipeline: the case for DevOps in insurance industry, continuous integration, continuous testing, and continuous delivery. Proceedings of the 2015 IEEE International Conference on Cloud Computing in Emerging Markets, CCEM 2015; 2016:85-89. doi:10.1109/CCEM.2015.29
15. Lwakatare LE, Kilamo T, Karvonen T, et al. DevOps in practice: a multiple case study of five companies. *Inf Softw Technol*. 2017;2019(114):217-230. doi:10.1016/j.infsof.2019.06.010
16. Sharma S, Coyne B. *DevOps For Dummies*. 2nd ed. John Wiley & Sons, Inc; 2014.
17. Ebert C, Gallardo G, Hernantes J, Serrano N. DevOps. *IEEE Softw*. 2016;33(3):94-100. doi:10.1109/MS.2016.68
18. Caprarelli A, Nitto E, Tamburri D. Fallacies and pitfalls on the road to DevOps: a longitudinal industrial study; 2020:200-210. doi:10.1007/978-3-030-39306-9_15
19. Jabbari R, bin Ali N, Petersen K, Tanveer B. What is DevOps? a systematic mapping study on definitions and practices. Proceedings of the Scientific Workshop Proceedings of XP2016 on – XP '16 Workshops; 2016:1-11. doi:10.1145/2962695.2962707
20. Ivanov V. *Implementation of DevOps pipeline for Serverless Applications*. Springer; 2018. doi:10.1007/978-3-030-03673-7_4
21. Teixeira D, Pereira R, Henriques T, da Silva MM, Faustino J, Silva M. A maturity model for DevOps. *Int J Agile Syst Manag*. 2020;13(4):464-511. doi:10.1504/IJASM.2020.112343
22. Jabbari R, bin Ali N, Petersen K, Tanveer B. Towards a benefits dependency network for DevOps based on a systematic literature review. *J Softw Evol Process*. 2018;30(11):1-26. doi:10.1002/smr.1957
23. Wiedemann A, Wiesche M, Krcmar H. Integrating development and operations in cross-functional teams — Toward a DevOps competency model. SIGMIS-CPR 2019 - Proceedings of the 2019 Computers and People Research Conference; 2019:14-19. doi:10.1145/3322385.3322400
24. Lwakature LE. Devops adoption and implementation in software development practice: concept, practices, benefits and challenges; 2017:99.
25. Leite L, Rocha C, Kon F, Milojicic D, Meirelles P. A survey of DevOps concepts and challenges. *ACM Comput Surv*. 2019;52(6):1-35. doi:10.1145/3359981
26. Punjabi R, Bajaj R. User stories to user reality: a devops approach for the cloud. In: 2016 IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, RTEICT 2016 - Proceedings.; 2017:658-662. doi:10.1109/RTEICT.2016.7807905
27. Silva MA, Faustino J, Pereira R, Silva MM. Productivity gains of DevOps adoption in an IT team: a case study. In: Andersson B, Johansson B, S. Carlsson C, Barry ML, Linger H, Schneider C, eds. *Proceedings of the 27th International Conference on Information Systems Development*. ISD2018; 2018:13.
28. Luz WP, Pinto G, Bonifácio R. Adopting DevOps in the real world: a theory, a model, and a case study. *J Syst Softw*. 2019;157:1-16. doi:10.1016/j.jss.2019.07.083
29. Riungu-Kalliosaari L, Mäkinen S, Lwakatare LE, Tiihonen J, Männistö T. DevOps adoption benefits and challenges in practice: a case study. *Proceedings of the 17th Internacional Conference, PROFES 2016, Throndheim, Norway*. Vol 10027 LNCS. Springer International Publishing; 2016:590-597. doi:10.1007/978-3-319-49094-6_44
30. Faustino J, Pereira R, Alturas B, da Silva MM. Agile information technology service management with DevOps: an incident management case study. *Int J Agile Syst Manag*. 2020;13(4):339-389. doi:10.1504/IJASM.2020.112331
31. Teixeira D, Pereira R, Henriques TA, Silva M, Faustino J. A systematic literature review on DevOps capabilities and areas. *Int J Human Capital Inf Technol Profess (IJHCITP)*. 2020;22:1-22. doi:10.4018/IJHCITP.2020070101
32. Silva CC, Goldman A. Agile methods adoption on software development: a pilot review. Proceedings of the 2014 Agile Conference, AGILE 2014; 2014:64-65; Institute of Electrical and Electronics Engineers Inc. doi:10.1109/AGILE.2014.14
33. Rauf A, Alghafees M. Gap analysis between state of practice and state of art practices in agile software development. Proceedings of the 2015 Agile Conference, Agile 2015; 2015:102-106; Institute of Electrical and Electronics Engineers Inc. doi:10.1109/Agile.2015.21
34. Govil N, Sayrakhia M, Agnihotri P, Shukla S, Agarwal S. Analyzing the behaviour of applying agile methodologies & DevOps culture in e-commerce web application. Proceedings of the 4th International Conference on Trends in Electronics and Informatics; 2020; IEEE. doi:10.1109/ICOEI48184.2020.9142895
35. Snyder B, Curtis B. Using analytics to guide improvement during an agile-DevOps transformation. *IEEE Softw*. 2017;35(1):78-83. doi:10.1109/MS.2017.4541032
36. Dornenburg E. The path to DevOps. *IEEE Softw*. 2018;35(5):71-75. doi:10.1109/MS.2018.290110337
37. Silva MA, Faustino J, Pereira R, Mira M. Productivity gains of DevOps adoption in an IT team: a case study. Proceedings of the 27th International Conference on Information Systems Development; 2018. https://repositorio.iscte-iul.pt/handle/10071/16388
38. Furfaro A, Gallo T, Garro A, Saccà D, Tundis A. ResDevOps: a software engineering framework for achieving long-lasting complex systems. Proceedings of the 2016 IEEE 24th International Requirements Engineering Conference, RE 2016. Published online 2016:246-255. doi:10.1109/RE.2016.15
39. Shahin M. Architecting for DevOps and continuous deployment. Proceedings of the ASWEC 2015 24th Australasian Software Engineering Conference on - ASWEC '15; Vol II, 2015:147-148. doi:10.1145/2811681.2824996

40. Kitchenham B. Procedures for performing systematic reviews; 2004:33.
41. Okoli C. A guide to conducting a standalone systematic literature review. *Commun Assoc Inf Syst*. 2015;37(1):879-910. doi:10.17705/1cais.03743
42. Thomas G. In: Seaman J, Piper J, eds. *How to Do Your Case Study*. 2nd ed. Sage Publications Asia-Pacific Pte Ltd; 2016.
43. Webster J, Watson RT. Analyzing the past to prepare for the future: writing a literature review. *MIS Q*. 2002;26(2):xiii-xxiii. doi:10.2307/4132319
44. Veres O, Kunanets N, Pasichnyk V, Veretennikova N, Korz R, Leheza A. Development and operations - the modern paradigm of the work of IT project teams. Proceedings of the IEEE 2019 14th International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT 2019 – Proceedings; Vol. 3, 2019:103-106. doi:10.1109/STC-CSIT.2019.8929861
45. Sousa L, Trigo A, Varajão J. DevOps – Foundations and perspectives [DevOps – Fundamentos e perspetivas]. Atas da Conferencia da Associacao Portuguesa de Sistemas de Informacao. 2019.
46. Céspedes D, Angeleri P, Melendez K, Dávila A. Software product quality in DevOps contexts: a systematic literature review. In: Mejia J, Muñoz M, Rocha ÁA, Calvo-Manzano J, eds. *Advances in Intelligent Systems and Computing*. Springer International Publishing; 2020:51-64. doi:10.1007/978-3-030-33547-2_5
47. Ottosson S. Agile principles and innovation development stig. developing and managing innovation in a fast changing and complex world: benefiting from dynamic principles; 2018:1-281. doi:10.1007/978-3-319-94045-8
48. Wahaballa A, Wahballa O, Abdellatief M, Xiong H, Qin Z. Toward unified DevOps model. Proceedings of the 6th IEEE International Conference on Software Engineering and Service Science (ICSESS); 2015:1-4. doi:10.1109/ICSESS.2015.7339039
49. Beigi-Mohammadi N, Litoiu M, Emami-Taba M, et al. A DevOps framework for quality-driven self-protection in web software systems a DevOps framework for quality-driven self-protection in web software. *System*. 2018;(October):270-274. doi:10.1145/nnnnnnn.nnnnnnn
50. Roche J. Adopting DevOps practices in quality assurance. *Commun ACM*. 2013;11(9):38-43. doi:10.1145/2524713.2524721
51. Wettinger J, Andrikopoulos V, Leymann F. Enabling DevOps collaboration and continuous delivery using diverse application. Proceedings of the On the Move to Meaningful Internet Systems: OTM 2015 Conferences; Vol 9415; 2015:348-358. doi:10.1007/978-3-319-26148-5
52. Schaefer A, Reichenbach M, Fey D. Continuous integration and automation for DevOps. *IAENG Trans Eng Technol*. 2012;170:345-358. doi:10.1007/978-94-007-4786-9
53. Díaz J, Almaraz R, Pérez J, Garbajosa J. DevOps in practice - An exploratory case study. *ACM Int Conf Proc Ser*. 2018;18-20. doi:10.1145/3234152.3234199
54. Sánchez-Gordón M, Colomo-Palacios R. Characterizing DevOps culture: a systematic literature review. In: Stamelos I, O'Connor RV, Rout T, Dorling A, eds. *Communications in Computer and Information Science*. Springer International Publishing; 2018:3-15. doi:10.1007/978-3-030-00623-5_1
55. Debroy V, Miller S, Brimble L. Building lean continuous integration and delivery pipelines by applying devops principles: a case study at varidesk. ESEC/FSE 2018 - Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering; 2018:851-856. doi:10.1145/3236024.3275528
56. Kuusinen K, Balakumar V, Jepsen SC, et al. A large agile organization on its journey towards DevOps. Proceedings of the 2018 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA); 2018:60-63. doi:10.1109/SEAA.2018.00019
57. Machiraju S, Gaurav S. In: Srivastava S, Moodie M, Modi D, eds. *DevOps for Azure Applications*. 1st ed. Apress; 2018. doi:10.1007/978-1-4842-3643-7
58. Atwal H. *DevOps for DataOps. Practical DataOps*. Apress; 2019:161-189. doi:10.1007/978-1-4842-5104-1_7
59. Forsgren N, Humble J, Kim G. *Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations*. IT Revolution Press; 2018.
60. Bruneo D, Fritz T, Keidar-Barner S, et al. CloudWave: where adaptive cloud management meets DevOps. Proceedings of the 2014 IEEE Symposium on Computers and Communications (ISCC) Workshops; 2014:1-6. doi:10.1109/ISCC.2014.6912638
61. Perez-Palacin D, Ridene Y, Merseguer J. Quality assessment in DevOps; 2017:133-138. doi:10.1145/3053600.3053632
62. Riti P. Cloud and DevOps. *Practical Scala DSLs*. 1st ed. Apress; 2018:209-220. doi:10.1007/978-1-4842-3036-7_11
63. Elberzhager F, Arif T, Naab M, Süß I, Koban S. From agile development to DevOps: going towards faster releases at high quality - Experiences from an industrial context. *Lect Notes Bus Inf Process*. 2017;269:33-44. doi:10.1007/978-3-319-49421-0_3
64. Sahid A, Maleh Y, Belaissaoui M. An agile framework for ITS management in organizations. A case study based on DevOps. ICCWCS'17: Proceeding of the 2nd Edition of the International Conference on Computing and Wireless Communication Systems; 2017:8. doi:10.1145/3167486.3167556
65. de França BBN, Jeronimo H, Travassos GH. Characterizing DevOps by hearing multiple voices. *ACM Int Conf Proc Ser*. 2016;53-62. doi:10.1145/2973839.2973845
66. Ravichandran A, Taylor K, Waterhouse P. DevOps and real world ROI. In: Ravichandran A, Taylor K, Waterhouse P, eds. *DevOps for Digital Leaders: Reignite Business with a Modern DevOps-Enabled Software Factory*. Apress; 2016:139-150.
67. Virmani M. Understanding DevOps & bridging the gap from continuous integration to continuous delivery. Proceedings of the 5th International Conference on Innovative Computing Technology, INTECH 2015; 2015;(Intech):78-82. doi:10.1109/INTECH.2015.7173368
68. Díaz J, Perez JE, Yague A, Villegas A, de Antona A. Devops in practice - A preliminary analysis of two multinational companies. arXiv; 2019:1-8. doi:10.1007/978-3-030-35333-9_23

69. Nybom K, Smeds J, Porres I. On the impact of mixing responsibilities between Devs and Ops. In: Sharp H, Hall T, eds. *Lecture Notes in Business Information Processing*. Springer International Publishing; 2016:131-143. doi:10.1007/978-3-319-33515-5_11
70. Siebra C, Lacerda R, Cerqueira I, et al. Empowering continuous delivery in software development: the DevOps strategy. In: van Sinderen M, Maciaszek LA, eds. *Software Technologies*. Communications in Computer and Information Science. Springer International Publishing; 2019:247-265. doi:10.1007/978-3-030-29157-0_11
71. Benguria G, Alonso J, Etxaniz I, Orue-Echevarria L, Escalante M. Agile development and operation of complex systems in multi-technology and multi-company environments: following a DevOps approach. *Commun Comput Inf Sci*. 2018;896:15-27. doi:10.1007/978-3-319-97925-0_2
72. Angara J, Gutta S, Prasad S. DevOps with continuous testing architecture and its metrics model. In: Sa PK, Bakshi S, Hatzilygeroudis IK, Sahoo MN, eds. *Advances in Intelligent Systems and Computing*. Springer; 2018:271-281. doi:10.1007/978-981-10-8633-5_28
73. Kaiser AK. *Reinventing ITIL® in the Age of DevOps*. Apress; 2018. doi:10.1007/978-1-4842-3976-6
74. Cuppett MS. DevOps, DBAs, and DBaaS; 2016:73-85. doi:10.1007/978-1-4842-2208-9
75. Albuquerque AB, Cruz VL. Implementing DevOps in legacy systems. In: Silhavy R, Silhavy P, Prokopova Z, eds. *Advances in Intelligent Systems and Computing*. Springer International Publishing; 2019:143-161. doi:10.1007/978-3-030-00184-1_14
76. Baudry B, Harrand N, Schulte E, et al. A spoonful of DevOps helps the GI go down. *Proc Int Conf Softw Eng*. 2018;35-36. doi:10.1145/3194810.3194818
77. Lai ST, Leu FY. *A Micro Services Quality Measurement Model for Improving the Efficiency and Quality of DevOps*. Vol 773. Springer International Publishing; 2019. doi:10.1007/978-3-319-93554-6_55
78. Mala DJ, Reynold PA. Towards green software testing in agile and devops using cloud virtualization for environmental protection. *Software Engineering in the Era of Cloud Computing*. Springer; 2020:277-297. doi:10.1007/978-3-030-33624-0_11
79. Masombuka T, Mnkandla E. A DevOps collaboration culture acceptance model. *ACM Int Conf Proc Ser*. 2018;279-285. doi:10.1145/3278681.3278714
80. Jaatun MG, Cruzes DS, Luna J. DevOps for better software security in the cloud. *ACM Int Conf Proc Ser*. 2017;F1305. doi:10.1145/3098954.3103172
81. Jaatun MG. Software security activities that support incident management in secure DevOps. *ACM Int Conf Proc Ser*. 2018. doi:10.1145/3230833.3233275
82. Rahman AAU, Williams L. Software security in DevOps: synthesizing practitioners' perceptions and practices. Proceedings - International Workshop on Continuous Software Evolution and Delivery, CSED; 2016. doi:10.1145/2896941.2896946
83. Kim G, Debois P, Willis J, Humble J. *The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations*. IT Revolution Press; 2016.
84. Šmite D, Gonzalez-Huerta J, Moe NB. "When in Rome, Do as the Romans Do": cultural barriers to being agile in distributed teams. In: Stray V, Hoda R, Paasivaara M, Kruchten P, eds. *Agile Processes in Software Engineering and Extreme Programming*. Lecture Notes in Business Information Processing. Springer International Publishing; 2020:145-161. doi:10.1007/978-3-030-49392-9_10
85. Colomo-Palacios R, Fernandes E, Soto-Acosta P, Larrucea X. A case analysis of enabling continuous software deployment through knowledge management. *Int J Inf Manag*. 2018;40:186-189. doi:10.1016/j.ijinfomgt.2017.11.005
86. De Sanctis M, Bucchiarone A, Trubiani C. A DevOps perspective for QoS-aware adaptive applications. In: Bruel JM, Mazzara M, Meyer B, eds. *Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment*. Lecture Notes in Computer Science. Springer International Publishing; 2020:95-111. doi:10.1007/978-3-030-39306-9_7
87. Šćekić M, Gazivoda M, Šćepanović S, Nikolić J. Application of DevOps approach in developing business intelligence system in bank. Proceedings of the 2018 7th Mediterranean Conference on Embedded Computing (MECO); 2018:1-4. doi:10.1109/MECO.2018.8406047
88. Mohan V, Ben Othmane L, Kres A. BP: security concerns and best practices for automation of software deployment processes: an industrial case study; 2018:21-28. doi:10.1109/SecDev.2018.00011
89. Gupta RK, Venkatachalapathy M, Jeberla FK. Challenges in adopting continuous delivery and DevOps in a globally distributed product team: a case study of a healthcare organization. In: *Proceedings - 2019 ACM/IEEE 14th International Conference on Global Software Engineering, ICGSE 2019*. IEEE; 2019:30-34. doi:10.1109/ICGSE.2019.00020
90. Sampedro Z, Holt A, Hauser T. Continuous integration and delivery for HPC: using singularity and Jenkins. *Proceedings of the Practice and Experience on Advanced Research Computing PEARC '18*. Association for Computing Machinery; 2018:1-6. doi:10.1145/3219104.3219147
91. Riungu-Kalliosaari L, Mäkinen S, Lwakatare LE, Tiihonen J, Männistö T. DevOps adoption benefits and challenges in practice: A case study. Proceedings of the 17th Internacional Conference, PROFES 2016, Throndheim, Norway; Vol 10027, 2016:590-597; LNCS. doi:10.1007/978-3-319-49094-6_44
92. Senapathi M, Buchan J, Osman H. DevOps capabilities, practices, and challenges: insights from a case study. *ACM Int Conf Proc Ser*. 2018;F1377. doi:10.1145/3210459.3210465
93. Heistand C, Thomas J, Tzeng N, et al. DevOps for spacecraft flight software. Proceedings of the IEEE Aerospace Conference Proceedings; March 2019:1-16. doi:10.1109/AERO.2019.8742143
94. Ghantous GB, Gill AQ. DevOps reference architecture for multi-cloud IOT applications. Proceedings of the 2018 IEEE 20th Conference on Business Informatics (CBI); Vol. 01; 2018:158-167. doi:10.1109/CBI.2018.00026
95. Smeds J, Nybom K, Porres I. DevOps: a definition and perceived adoption impediments. In: Lassenius C, Dingsøyr T, Paasivaara M, eds. *Agile Processes in Software Engineering and Extreme Programming*. Lecture Notes in Business Information Processing. Springer International Publishing; 2015:166-177. doi:10.1007/978-3-319-18612-2_14

96. Jiménez M, Rivera LF, Villegas NM, Tamura G, Müller HA, Gallego P. DevOps' shift-left in practice: an industrial case of application. In: Bruel JM, Mazzara M, Meyer B, eds. *Lecture Notes in Computer Science*. Springer International Publishing; 2019:205-220. doi:10.1007/978-3-030-06019-0_16
97. Hemon A, Lyonnet B, Rowe F, Fitzgerald B. From agile to DevOps: smart skills and collaborations. *Inf Syst Front*. 2020;22(4):927-945. doi:10.1007/s10796-019-09905-1
98. AL-Zahran S, Fakieh B. How DevOps practices support digital transformation. *Int J Adv Trends Comput Sci Eng*. 2020;9:2780-2788. doi:10.30534/ijatcse/2020/46932020
99. Hemon-Hildgen A, Fitzgerald B, Lyonnet B, Rowe F. Innovative practices for knowledge sharing in large-scale DevOps. *IEEE Softw*. 2019;37(3):30-37. doi:10.1109/MS.2019.2958900
100. Gall M, Pigni F. Leveraging DevOps for mission critical software. AMCIS; 2018. https://aisel.aisnet.org/amcis2018/ITProjMgmt/Presentations/2
101. Jones S, Noppen J, Lettice F. Management challenges for devops adoption within UK SMEs. QUDOS 2016 - Proceedings of the 2nd International Workshop on Quality-Aware DevOps, co-located with ISSTA 2016; 2016:7-11. doi:10.1145/2945408.2945410
102. Diaz J, Pérez JE, Lopez-Peña MA, Mena GA, Yagüe A. Self-service cybersecurity monitoring as enabler for DevSecOps. *IEEE Access*. 2019;7:100283-100295. doi:10.1109/ACCESS.2019.2930000
103. Bruza M. An Analysis of Multi-domain Command and Control and the Development of Software Solutions through DevOps Toolsets and Practices; 2018.
104. Shahin M, Babar MA, Zhu L. The intersection of continuous deployment and architecting process: practitioners' perspectives. Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement. ESEM '16. Association for Computing Machinery; 2016:1-10. doi:10.1145/2961111.2962587
105. Laukkanen E, Paasivaara M, Itkonen J, Lassenius C, Arvonen T. Towards continuous delivery by reducing the feature freeze period: a case study. Proceedings of the 2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering in Practice Track (ICSE-SEIP); 2017:23-32. doi:10.1109/ICSE-SEIP.2017.21
106. Lwakatare LE, Kuvaja P, Oivo M. An exploratory study of DevOps: extending the dimensions of DevOps with practices. Proceedings of the 11th International Conference on Software Engineering Advances (ICSEA); 2016.
107. Alt R, Auth G, Kögler C. Transformation of consulting for software-defined businesses: lessons from a DevOps case study in a German IT company. *Advances in Consulting Research*. Springer; 2018:385-403. doi:10.1007/978-3-319-95999-3_19
108. Düllmann TF, Paule C, van Hoorn A. Exploiting DevOps practices for dependable and secure continuous delivery pipelines. Proceedings of the 2018 IEEE/ACM 4th International Workshop on Rapid Continuous Software Engineering (RCoSE); 2018:27-30.
109. Laukkarinen T, Kuusinen K, Mikkonen T. DevOps in regulated software development: case medical devices. Proceedings - 2017 IEEE/ACM 39th International Conference on Software Engineering: New Ideas and Emerging Results Track, ICSE-NIER 2017. doi:10.1109/ICSE-NIER.2017.20
110. Zhou X, Jin Y, Zhang H, Li S, Huang X. A map of threats to validity of systematic literature reviews in software engineering. Proceedings - Asia-Pacific Software Engineering Conference, APSEC; 2016:153-160. doi:10.1109/APSEC.2016.031
111. Erich FMA, Amrit C, Daneva M. A qualitative study of DevOps usage in practice. *J Softw Evol Process*. 2017;29(6):1-20. doi:10.1002/smr.1885
112. Metzger S, Durden D, Sturtevant C, et al. eddy4R 0.2.0: a DevOps model for community-extensible processing and analysis of eddy-covariance data based on R, Git, Docker, and HDF5. *Geosci Model Dev*. 2017;10(9):3189-3206. doi:10.5194/gmd-10-3189-2017
113. Gupta V, Kapur PK, Kumar D. Modeling and measuring attributes influencing DevOps implementation in an enterprise using structural equation modeling. *Inf Softw Technol*. 2017;92:75-91. doi:10.1016/j.infsof.2017.07.010
114. Chen B. Improving the software logging practices in devops. Proceedings - 2019 IEEE/ACM 41st International Conference on Software Engineering: Companion, ICSE-Companion 2019; 2019:194-197; IEEE. doi:10.1109/ICSE-Companion.2019.00080
115. Barna C, Khazaei H, Fokaefs M, Litoiu M. Delivering elastic containerized cloud applications to enable DevOps. Proceedings - 2017 IEEE/ACM 12th International Symposium on Software Engineering for Adaptive and Self-Managing Systems, SEAMS 2017; 2017:65-75. doi:10.1109/SEAMS.2017.12
116. Olszewska M, Waldén M. DevOps meets formal modelling in high-criticality complex systems. Proceedings of the 1st International Workshop on Quality-Aware DevOps, QUDOS 2015 – Proceedings; 2015:7-12. doi:10.1145/2804371.2804373
117. Muñoz M, Negrete M. Reinforcing DevOps generic process with a guidance based on the basic profile of ISO/IEC 29110. In: Mejia J, Muñoz M, Rocha Á, A. Calvo-Manzano J, eds. *Advances in Intelligent Systems and Computing*. Springer International Publishing; 2020:65-79. doi:10.1007/978-3-030-33547-2_6
118. Cruzes DS, Melsnes K, Marczak S. Testing in a DevOps era: perceptions of testers in Norwegian organisations. In: Misra S, Gervasi O, Murgante B, et al., eds. *Lecture Notes in Computer Science*. Springer International Publishing; 2019:442-455. doi:10.1007/978-3-030-24305-0_33
119. Daneva M, Bolscher R. What we know about software architecture styles in continuous delivery and DevOps? In: van Sinderen M, Maciaszek LA, eds. *Communications in Computer and Information Science*. Springer International Publishing; 2020:26-39. doi:10.1007/978-3-030-52991-8_2
120. Scheaffer J, Ravichandran A, Martins A. In: McDermott S, Berendson L, Fernando R, eds. *The Kitty Hawk Venture*. 1st ed. Apress; 2018. doi:10.1007/978-1-4842-3661-1
121. Hüttermann M. *DevOps for Developers*. 1st ed. Apress; 2012. doi:10.1007/978-1-4302-4570-4
122. Davis A. DevOps. *Mastering Salesforce DevOps*. Apress; 2019. doi:10.1007/978-1-4842-5473-8_3

123. Kuruba M. DevOps for IT service reliability and availability. In: Karanki D, Vinod G, Ajit S, eds. *Advances in RAMS Engineering*. 1st ed. Apress; 2019:149-183. doi:10.1007/978-3-030-36518-9_6
124. Familiar B, Barnes J. DevOps using PowerShell, ARM, and VSTS. *Business in Real-Time Using Azure IoT and Cortana Intelligence Suite*. 1st ed. Apress; 2017:21-93. doi:10.1007/978-1-4842-2650-6_2
125. Muñoz M, Díaz O. DevOps: foundations and its utilization in data center. In: J. MG, M. M, M. R, W. N, R O. *Engineering and Management of Data Centers*. Springer; 2017:205-225. doi:10.1007/978-3-319-65082-1_10
126. Nagarajan AD, Overbeek SJ. A DevOps implementation framework for large agile-based financial organizations. In: H. P, C. D, H. P, C. A, D. R, R. M. *On the Move to Meaningful Internet Systems. OTM 2018 Conferences*. Springer; 2018:1-17. doi:10.1007/978-3-030-02610-3_10
127. Wang W, Casale G, Iuhasz G. Closing the loop between ops and dev. In: Di Nitto E, Matthews P, Petcu D, Solberg A eds. *Model-Driven Development and Operationof Multi-Cloud Applications*. Springer; 2017:95-105. doi:10.1007/978-3-319-46031-4_10
128. de Feijter R, Overbeek S, van Vliet R, Jagroep E, Brinkkemper S. DevOps competences and maturity for software producing organizations. In: Gulden J, Reinhartz-Berger I, Schmidt R, Guerreiro S, Guédria W, Bera P, eds. *Lecture Notes in Business Information Processing*. Springer International Publishing; 2018:244-259. doi:10.1007/978-3-319-91704-7_16
129. Cito J, Wettinger J, Lwakatare LE, Borg M, Li F. Feedback from operations to software development—A DevOps perspective on runtime metrics and logs. In: Bruel JM, Mazzara M, Meyer B, eds. *Lecture Notes in Computer Science*. Springer International Publishing; 2019:184-195. doi:10.1007/978-3-030-06019-0_14
130. Alban D, Eynaud P, Malaurent J, Richet JL, Vitari C. *Information Systems Management: Governance, Urbanization and Alignment*. ISTE Ltd and John Wiley & Sons, Inc; 2019. doi:10.1016/B978-012226570-9/50107-1
131. Beulen E. Implementing and contracting agile and DevOps: a survey in the Netherlands. In: Kotlarsky J, Oshri I, Willcocks L, eds. *Lecture Notes in Business Information Processing*. Springer International Publishing; 2019:124-146. doi:10.1007/978-3-030-15850-7_7
132. Colavita F. DevOps movement of enterprise agile breakdown silos, create collaboration, increase quality, and application speed. In: Ciancarini P, Sillitti A, Succi G, Messina A, eds. *Advances in Intelligent Systems and Computing*. Springer International Publishing; 2016:203-213. doi:10.1007/978-3-319-27896-4_17
133. Fishman N, Stryker C. In: Minatel J, ed. *Smarter Data Science*. 1st ed. John Wiley & Sons, Inc; 2020. doi:10.1002/9781119697985
134. Johng H, Kalia AK, Xiao J, Vuković M, Chung L. Harmonia: a continuous service monitoring framework using DevOps and service mesh in a complementary manner. In: Yangui S, Bouassida Rodriguez I, Drira K, Tari Z, eds. *Service-Oriented Computing. Lecture Notes in Computer Science*. Springer International Publishing; 2019:151-168. doi:10.1007/978-3-030-33702-5_12
135. Kang H, Le M, Tao S. Container and microservice driven design for cloud infrastructure DevOps. Proceedings of the 2016 IEEE International Conference on Cloud Engineering (IC2E); 2016:202-211. doi:10.1109/IC2E.2016.26
136. Lewerentz M, Bluhm T, Daher R, et al. Implementing DevOps practices at the control and data acquisition system of an experimental fusion device. *Fusion Eng Design*. 2018;146(October):40-45. doi:10.1016/j.fusengdes.2018.11.022
137. Lewis B, Smith I, Fowler M, Licato J. Enabling DevOps for containerized data-intensive applications: an exploratory study. Proceedings of the Modern Artificial Intelligence and Cognitive Science Conference, MAICS 2017; 2017:189-190. doi:10.1145/1235
138. Palermo J. In: Murray J, Berendson L, Balzano J, eds. *NET DevOps for Azure*. 1st ed. Apress; 2019. doi:10.1007/978-1-4842-5343-4
139. Ravichandran A, Taylor K, Waterhouse P. DevOps for digital leaders; 2016. doi:10.1007/978-1-4842-1842-6
140. Rong G, Zhang H, Shao D. CMMI guided process improvement for DevOps projects. Proceedings of the International Workshop on Software and Systems Process - ICSSP '16; 2016:76-85. doi:10.1145/2904354.2904372
141. Senapathi M, Buchan J, Osman H. DevOps capabilities, practices, and challenges: insights from a case study. EASE; 2018. doi:10.1145/3210459.3210465
142. Tomlinson T. Drupal 8 DevOps. *Enterprise Drupal 8 Development*. Apress; 2017:271-279. doi:10.1007/978-1-4842-0253-1_11
143. Wettinger J, Breitenbücher U, Falkenthal M, Leymann F. Collaborative gathering and continuous delivery of DevOps solutions through repositories. *Comput Sci Res Dev*. 2017;32(3-4):281-290. doi:10.1007/s00450-016-0338-z
144. Ali N, Daneth H, Hong JE. A hybrid DevOps process supporting software reuse: a pilot project. *J Softw Evol Process*. 2020;32(7):1-23. doi:10.1002/smr.2248
145. Gotimer G, Stiehm T. DevOps advantages for testing: Increasing quality through continuous delivery. *CrossTalk*. 2016;29(3):13-18.
146. Morales JA, Yasar H, Volkman A. Implementing DevOps practices in highly regulated environments. *ACM Int Conf Proc Ser*. 2018;F1477. doi:10.1145/3234152.3234188
147. Pérez JF, Wang W, Casale G. Towards a DevOps approach for software quality engineering. WOSP-C 2015 - Proceedings of the 2015 ACM/SPEC Workshop on Challenges in Performance Methods for Software Development, in Conjunction with ICPE; 2015. doi:10.1145/2693561.2693564
148. de Kort W. In: DeWolf J, Pundick D, eds. *DevOps on the Microsoft Stack*. 1st ed. Apress; 2016. doi:10.1007/978-1-4842-1446-6
149. Rittgen P, Cronholm S, Göbel H. Towards a model for assessing collaboration capability between development and operations. In: Walker A, O'Connor R V, Messnarz R, eds. *Communications in Computer and Information Science*. Springer International Publishing; 2019:111-122. doi:10.1007/978-3-030-28005-5_9
150. Murphy GC, Kersten M. Towards bridging the value gap in DevOps. In: Bruel JM, Mazzara M, Meyer B, eds. *Lecture Notes in Computer Science*. Springer International Publishing; 2020:181-190. doi:10.1007/978-3-030-39306-9_13
151. Arulkumar V, Lathamanju R. Start to finish automation achieve on cloud with build channel: by DevOps method. *Proc Comput Sci*. 2019;2019(165):399-405. doi:10.1016/j.procs.2020.01.032
152. Medina O, Schumann E. In: Murray J, Berendson L, Balzano J, eds. *DevOps for SharePoint*. 1st ed. Apress; 2018. doi:10.1007/978-1-4842-3688-8

153. Koilada DK. Business model innovation using modern DevOps. Proceedings of the 2019 IEEE Technology and Engineering Management Conference, TEMSCON 2019; 2019:1-6. doi:10.1109/TEMSCON.2019.8813557
154. Rong G, Jin Z, Zhang H, Zhang Y, Ye W, Shao D. DevDocOps: enabling continuous documentation in alignment with DevOps. *Softw Pract Exp*. 2019;50(3):210-226. doi:10.1002/spe.2770
155. Forsgren N, Kersten MIK. DevOps metrics. *ACM Queue*. 2017;61(december):1-16.
156. Cukier D. DevOps patterns to scale web applications using cloud services. SPLASH 2013 - Proceedings of the 2013 Companion Publication for Conference on Systems, Programming, and Applications: Software for Humanity; 2013;(Figure 2):143-152. doi:10.1145/2508075.2508432
157. Combemale B, Wimmer M. Towards a model-based DevOps for cyber-physical systems. In: Bruel JM, Mazzara M, Meyer B, eds. *Lecture Notes in Computer Science*. Springer International Publisheing; 2020:84-94. doi:10.1007/978-3-030-39306-9_6

How to cite this article: Faustino J, Adriano D, Amaro R, Pereira R, da Silva MM. DevOps benefits: A systematic literature review. *Softw: Pract Exper*. 2022;52(9):1905-1926. doi: 10.1002/spe.3096

APPENDIX A. CONCEPT-CENTRIC APPROACH FOR BENEFITS AND LITERATURE

This appendix provides a mapping between all the authors (references) that mention a certain DevOps benefit.

Concept ID	Reference
B01	6,12,15,17,22,25,28,29,44–46,49–51,56,57,62–66,68,69,72,75,79,82,89,92,107,111–129
B02	6,12,14,17,22,23,25,39,51–54,56,58,63,65,74,78,91,101,113,121–123,125,126,129–143
B03	6,14,15,22,23,39,44,45,49,51,53–56,58,60,63,66,72,74,75,93,109,111,117,118,121,122,125,133,138,144–150
B04	14,15,20,22,28,29,45,51,57,65,70,75–77,92,93,107,109,113,118–120,122,125,128,132,134,145,148,150–152
B05	6,17,22,28,29,45,50,56,58,60,65,68,89,92,101,118,120–122,125,133,134,138,153–155
B06	6,14,22,46,57,61,64,65,67,70,93,107,116,118,119,121–123,130,147,152,156
B07	6,14,23,45,46,56,62–64,66,68,81,122,123,135,142,145
B08	12,22,28,29,44,45,53,57,63–65,69,113,134,144,151,153
B09	22,28,51,65–67,74,107,125,129,130,143,145
B10	22,65,67–71,109,124,125,140,144,146
B11	15,22,23,25,29,54,65,66,69,92,117,118,123
B12	22,56,65,67,70,72–74,76,117,148
B13	22,29,51,55,56,58,64,65,77,113,133
B14	22,46,58,65,66,74,75,78,118,142,157
B15	12,15,54,63,69,79,92,113,121
B16	14,29,117,133,136,150,153,157
B17	56,63,65,69,73
B18	80–82,122,134
B19	15,76

APPENDIX B. IDENTIFIED BUSINESS BENEFITS PER DEVOPS IMPLEMENTATION CASE STUDY

In this appendix there is the mapping between the case studies and the benefits identified on each of these case studies.

ID	Case study number	DevOps benefit concept ID
CS.1	1	B01
CS.2	2	B02; B04; B17
CS 3	3	B01; B06
CS.4	4	B11
CS.5	5	B01; B02; B08; B10; B17
CS.6	6	B01; B03; B13; B04; B06; B10; B12;
CS.7	7	B01; B02; B06; B10; B17
CS.8	8	B01; B02; B06; B10; B18; B17;
CS.9	9	B01; B04; B06; B16
CS.10	10	B01; B02; B03; B10; B17;
CS.11	11	B04; B06
CS.12	12.1	B01; B02; B03; B04; B05; B06; B07; B08; B10; B12; B14; B15; B17
	12.2	B01; B02; B03; B04; B05; B06; B07; B08; B10; B12; B14; B15; B17
	12.3	B01; B02; B03; B04; B05; B06; B07; B08; B10; B12; B14; B15; B17
CS.13	13	B01; B04; B05; B10; B13; B15; B16
CS.14	14	B06; B17
CS.15	15.1	B02; B06; B07; B17
	15.2	B02; B06; B07; B17
CS.16	16.1	B02; B03; B08; B09; B12; B13
	16.2	B02; B03; B08; B09; B12; B13
	16.3	B02; B03; B08; B09; B12; B13
	16.4	B02; B03; B08; B09; B12; B13
	16.5	B02; B03; B08; B09; B12; B13
	16.6	B02; B03; B08; B09; B12; B13
	16.7	B02; B03; B08; B09; B12; B13
	16.8	B02; B03; B08; B09; B12; B13
	16.9	B02; B03; B08; B09; B12; B13
	16.10	B02; B03; B08; B09; B12; B13
	16.11	B02; B03; B08; B09; B12; B13
CS.17	17.1	B4; B6; B7; B9; B10; B11; B17; B19
	17.2	B02; B04; B08; B15; B19
	17.3	B01; B02; B04; B06; B07; B08; B15; B19
	17.4	B02; B04; B06; B07; B08; B15; B19
	17.5	B02; B04; B08; B15; B19
CS.18	18	B01; B02; B04; B08; B12
CS.19	19	B01
CS.20	20	B04; B06
CS.21	21	No benefits identified
CS.22	22	B01; B03; B08; B11
CS.23	23.1	B02; B06; B11; B13; B15; B16; B17
	23.2	B02; B03; B06; B11; B13; B16; B15
CS.24	24	B01; B02; B03; B04; B06; B07; B12 B13; B17
CS.25	25	B01; B08; B11; B12; B15; B19
CS.26	26.1	B01; B08; B11; B12; B15
	26.2	B01; B08; B11; B12; B15
	26.3	B01; B08; B11; B12; B15
	26.4	B01; B08; B11; B12; B15
CS.27	27	B01; B02
CS.28	28	B01
CS.29	29	B01; B06; B11; B12; B14; B15; B19
CS.30	30	B01; B06; B07; B08; B12; B19
CS.31	31	B01; B02; B03; B04; B05; B06; B08; B13; B15; B16; B17; B18
CS.32	32.1	B01; B02; B04; B10; B11; B12; B17
	32.2	B01; B02; B04; B10; B11; B12; B17
	32.3	B01; B02; B04; B10; B11; B12; B17
	32.4	B01; B02; B04; B10; B11; B12; B17
	32.5	B01; B02; B04; B10; B11; B12; B17
	32.6	B01; B02; B04; B10; B11; B12; B17
	32.7	B01; B02; B04; B10; B11; B12; B17
	32.8	B01; B02; B04; B10; B11; B12; B17
	32.9	B01; B02; B04; B10; B11; B12; B17
	32.10	B01; B02; B04; B10; B11; B12; B17
CS.33	33	B02; B06; B14
CS.34	34.1	B02; B12
	34.2	B02; B12
	34.3	B02; B12
	34.4	B02; B12
CS.35	35	B01; B02; B03; B05; B13; B16
CS.36	36	B02; B03; B08; B09; B12; B13; B19

DevOps and Problem Management: A Case Study

Daniel Adriano[a], João Faustino[a], Rúben Pereira[a], Rafael Almeida[b], Miguel Mira da Silva[c]

[a]Instituto Universitario de Lisboa (ISCTE-IUL), Av. das Forças Armadas, 1649-026 Lisboa, Portugal; [b]Neobrain, 6 Rue de Rome, 75008 Paris, France; [c]Universidade de Lisboa – Instituto Superior Técnico (IST), Department of Computer Science and Engineering, 1049-001 Lisboa, Portugal.

Abstract

The use of DevOps is a predominant attribute of businesses engaged in the development and maintenance of Information Technology systems. Although literature exploring DevOps practices has expanded, there is still much unexplored territory on its operational ramifications. This is particularly observed when considering their potential impact on Information Technology Service Management frameworks. This research aims to establish what DevOps practices can be applied to Problem Management, a core Service Management process. An exploratory Case Study was carried out with the participation of Problem Managers operating in a DevOps environment. Three data collection methods were applied: Semi-structured interviews; documental analysis and observation; and a focus group exercise. This research indicates that DevOps practices have varying degrees of significance for a Problem Management process. Practices associated with continuous planning and collaboration are prone to having greater significance in a Problem lifecycle, with the potential of enabling benefits such as quicker Problem identification, higher quality Root Cause Analysis, and improved resolution times. The novelty of insight gathered in this study benefits both academics, through its contribution to an expanding body of knowledge, and professionals, considering the practical and applicable nature of findings. Future work is also presented.

Keywords: DevOps; ITSM; ITIL; Problem Management.

1. Introduction

The increased introduction of Information Technology (IT) resources into modern-day products and services has profoundly transformed the nature of business worldwide (Limanto et al., 2017). One could say that this development triggered the start of a new "information age" (Varga et al., 2019) in which the proper management of knowledge, processes and service delivery are present at the very center of organizational priorities, being key for success (Alsolamy et al., 2014; Park et al., 2006). Technological developments have driven up the investment on IT, as organizations strive to capture the benefits resulting from innovation as well as protect themselves against new and emerging threats (Benitez et al., 2018; Luo et al., 2016). In an effort to cope with the constant competition present in increasingly dynamic markets (Badinelli et al., 2012) organizations have developed and put in place equally "complex and dynamic IT systems to support their business processes" (Jamous et al., 2016; Soni, 2015).

Due to changing market demands, the very methodologies by which software is delivered have evolved (Virmani, 2015). As stated by Šćekić et al., (2018), ever-changing business needs, coupled with the present expediency requirements for introducing software into the market, have "created a paradigm shift towards a 3rd generation Software Development philosophy called DevOps." (See also Laukkanen et al., 2017). Many have defined the new sets of practices contemplated in DevOps as the merging of people, process, and product, aimed at delivering greater value to customers (de Kort, 2016) through quicker, yet equally competent and reliable, development and release cycles (Farroha & Farroha, 2014; Mohan et al., 2018). As such, it should not come as a surprise why these have become so increasingly popular in present-day software development (Luz et al., 2019; Ståhl et al., 2017).

The emerging DevOps philosophy not only relates to software development standards but plays a significant role in the level and frequency of interaction between development and operational staff (Aiello & Sachs, 2016; Guerriero et al., 2015). Literature examining the processes by which DevOps is adopted frequently mention collaborative links being formed between said teams; these reportedly result into greater knowledge sharing (Senapathi et al., 2018), improved risk assessment and error detection (Gupta et al., 2019) as well as the reduction of deployment cycle duration (Kuusinen et al., 2018). This has led to new research being performed on the potential challenges, risks, and benefits of the interaction between DevOps practices and Information Technology Service Management (ITSM) frameworks, which direct operations (Pilorget & Schell, 2018). It is through the "specialized organizational capabilities" ITSM offers, in the form of processes, functions and roles, that value can be consistently generated and delivered (Cartlidge & Lillycrop, 2009).

The outcomes of DevOps practices in organizations operating via ITSM have been largely unexplored (Abdelkebir et al., 2017; Kamuto & Langerman, 2017). Amongst established frameworks such as COBIT and ISO/IEC, the Information Technology Infrastructure Library (ITIL) stands out as the most widely adopted ITSM system (Sharifi et al., 2008). Prevalently used ITIL processes such as Incident and Problem Management (PM) are of great significance for businesses in virtually any sector (Krishna Kaiser, 2018). As such, one can perceive an opportunity of delving deeper into the potential links between these key processes and the DevOps methodology. More specifically, the study of what DevOps practices can be applied in each stage of a Problem Management process, and the extent to which these can improve the resolution of Problems, may lead to practical business benefits being captured by organizations working with said processes and methodologies. Research has already been performed to understand how DevOps practices may have a role in executing the Incident Management process (Faustino, 2018). We will give continuity to this effort, focusing our analysis on exploring DevOps interactions with PM through a case study in the IT department of a German multinational conglomerate company.

2. Theoretical Background

This chapter introduces the central topics which are covered throughout this research. In addition to providing an overview of the current understanding of DevOps and PM, it also contains a summary on to what extent these concepts have been articulated in present literature.

2.1 Defining DevOps

The DevOps (Developer + Operation) approach has its origins dating back to 2009, when the DevOpsDays conference was first held in Ghent, Belgium (Kim et al., 2016; Perera et al., 2017). It embodies a "change in IT culture" (Šćekić et al., 2018) in which high-speed software delivery (Virmani, 2015), improved collaboration (Perera et al., 2017) and the implementation of continuous testing, integration and feedback activities are emphasized (Soni, 2015). Its end goal is as far reaching as it is ambitious: To minimize the time between the initial concept and the "production ready application" (Soni, 2015); to maximize speed of delivery, having new code and software in "shippable state" at any point in time (Virmani, 2015); to capture, to the fullest extent, the benefits resulting from ongoing innovation (Sharma et al., 2015).

DevOps is often defined as a software development methodology in which integration between operations teams and software developers takes place (Wahaballa et al., 2015), and where. a "tight coupling" of development and operational roles is observed (Kuusinen et al., 2018). DevOps has also been defined as a broader organizational approach, or a movement, aimed at integration and collaboration in the delivery process of software products or services (Díaz et al., 2019), emphasizing fluidity (Ebert et al., 2016) between traditionally isolated silos (Colomo-Palacios et al., 2018). Independent from how one would conceptualize DevOps, there are common, distinguishable practices associated with its implementation (Díaz et al., 2018). The establishing of Agile planning principles in which small, cross-functional teams, having defined, high-level objectives, collaborate frequently (using Scrum or Kanban tools) towards iterative development (Šmite et al., 2020); the introduction of a Continuous Integration (CI) concept in the coding and building of software (Dyck et al., 2015); a reliance on automated deployment pipelines (Kuusinen et al., 2018; Jiménez et al., 2019) through the practices of Continuous Delivery (CDE) and Continuous Deployment (CD). Here, a bridge between development and operations is important: On one hand, developers ensure that reliable, high quality software is being built and released (Chen, 2017); on the other, operations staff review its behavior and monitor its performance in a production environment (Shahin et al., 2016). A visual representation of a DevOps framework, demonstrating the consolidation of planning, CI, and CD/CDE practices, as well as the link of Development and Operations activity, can be found in Figure 1.

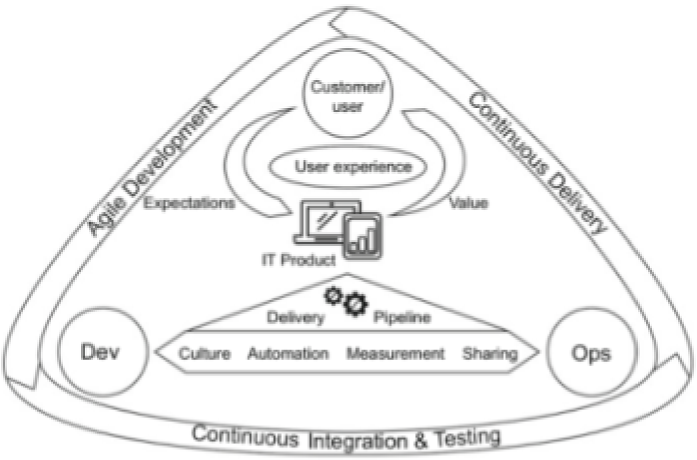


Figure 1 - A DevOps Framework (Alt et al., 2018)

2.2 Defining Problem Management

IT Service Management frameworks have become a vital part of organizations (Mora et al., 2015), providing a set of processes to "align, design, deliver, manage and improve" both the internal use of IT resources (Wang et al., 2010), as well as the delivery of IT products to customers (Limanto et al., 2017; Faustino, Pereira & Mira da Silva, 2019). ITIL currently stands out as the most widely used ITSM framework (Sharifi et al., 2008; Cater-Steel et al., 2006) with its concepts being successfully adapted and adopted by a vast number of companies (Marrone & Kolbe, 2011). The overarching goal of the framework has prevailed throughout its versions: That of providing value, in the form of high-quality services, to organizations (ITIL® Foundation Handbook 3rd ed. 2012).

The overall result of a successful implementation of ITIL models has shown organizations achieving better performance in their key operational activities, with improved service quality and customer satisfaction as main outputs (Melendez et al., 2016). The introduction of a PM process, aimed at resolving the underlying cause of Incidents, has been observed as "one of the most important processes to ensure service stability" (Krishna Kaiser, 2018). A Problem is defined as the underlying cause of one or more Incidents. It is the purpose of PM, by way of monitoring, reporting and investigation, to discover, document and fix root causes of Incidents, preventing their reoccurrence (Sukmandhani et al., 2017). According to the ITIL framework (ITIL® Foundation Handbook 3rd ed. 2012), this is done through six key activities, as described in Table 1.

Outputs of an effective PM process includes both the sharing of knowledge, reducing Incident duration times by establishing Workarounds (Krishna Kaiser, 2018), and the implementation of Problem solutions, permanently removing underlying IT service errors, effectively mitigating the risk of future interruptions (Abdelkebir et al., 2017). In summary, PM is considered a crucial process towards providing reliable, high-quality IT services and maximizing customer satisfaction (Radhakrishnan et al., 2008; Sukmandhani et al., 2017).

Table 1 - PM Process (ITIL® Foundation Handbook 3rd ed. (2012)

Problem Activity	Description
Problem Detection	Candidates for investigation are pinpointed as an output of Incident analysis.
Problem Logging, Categorization, and Prioritization	The Problem is recorded, described, and linked with associated Incident records.
Problem Investigation and Diagnosis	RCA activities take place, identifying the underlying cause of the Incident(s).
Known Error Management	A Workaround, or temporary solution, is delivered to reduce the impact of Problems for which a complete solution is not yet implemented.
Problem Resolution	The implementation of solutions is carried out.
Problem Closure and Review	With the permanent solution in place, the investigation is formally finalized, with all findings and actions being documented. This can also be the stage used to define, document and implement any lessons learned.

As mentioned in the introduction, very limited research has been done towards proposing and defining specific articulation between DevOps and ITSM frameworks such as ITIL (Kamuto & Langerman, 2017). Nevertheless, researchers such as Forsgren & Humble (2015) have recognized the potential relationship between both methodologies. Here, DevOps serves as the "the link between Software Development and Operations", with Operations being systematized via ITSM frameworks (Hüttermann, 2012). Other authors have also presented their case for the integration of development and operational practices (Riungu-Kalliosaari et al., 2016; Nybom et al., 2016; Lucy Ellen Lwakatare et al., 2019; Luz et al., 2019); however, no description of which specific practices can or should be integrated are provided. Lastly, the release of ITIL4, the most recent framework iteration, also includes mention of DevOps, but not to the level of detail aimed at through the present Case Study on PM.

3. Research Methodology

Exploratory research is aimed at examining observed phenomenon in which there is either an absence or scarcity of related works (zaid Zainal, 2007). This is often done by way of Case Study (CS), a valuable source from where researchers can both develop an "in-depth" understanding of complex issues in a "real life" setting (Dobson, 1999) or construct new theory (Perry et al., 2004). Given that the researcher works on a team that applies PM as well as an organization which has implemented DevOps, we classified this CS as a local knowledge case. Considering the early stage of research done on the subject of DevOps applicability toward the PM process, this investigation is exploratory in nature and purpose. Our approach is focused on building theory on the subject matter and, as what is studied only targets a single unit of analysis, this is considered a single-case approach (Aberdeen, 2013). Figure 2 presents our CS path, based on Thomas' (2016) model.

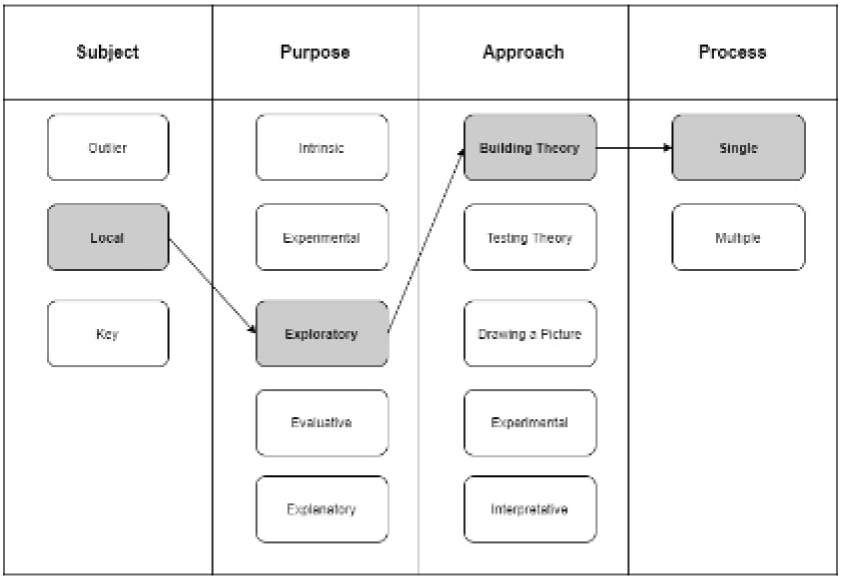


Figure 2 - Case Study Path, adapted from Thomas (2016)

A CS is built around a central question (Thomas, 2016). For the present research, that question is the following: "How can DevOps practices interface with an ITIL PM process?". The two central Research Questions (RQ's) used to shape and direct the design and course of our investigation (Tashakkori & Creswell, 2007) were identified and are presented in Table 2.

As this research relies on the experience and insight of a team operating under already implemented ITIL PM practices, this must be considered as a retrospective CS.

Table 2 - Research Questions

Research Question ID	Description
RQ1	Can DevOps practices be applied in each stage of the PM process lifecycle? If so, which?
RQ2	Does DevOps improve the resolution of Problems? If so, how?

A triangulation approach was followed in the carrying out of this CS. The author began by conducting 10 semi-structured interviews to capture practitioner experience and insight regarding the implementation of DevOps practices in a PM process. Selected participants are part of the IT department of a German multinational conglomerate company. An analysis of process data and workflows implemented by interviewed practitioners followed, in which the author examined conformity with establish ITIL frameworks. Lastly, a focus group exercise was performed with 5 of the 10 interviewees. These activities enabled the analysis of our issue through multiple angles, which is advisable in CS exercises (Modell, 2005). Further details on the participants of the present CS are presented in Table 3.

Table 3 - Participant Details

Interviewee	Position	Experience in Role	Experience in IT	Experience in DevOps	Experience in PM	Number of Projects**
A	Process Manager	3	5	1	4	2
B	Developer	5	10	9	3	2
C	Service Manager	3	14	3	14	2
D	Developer	4	5	4	3	1
E	Process Manager	6	13	2	6	3
F	Developer Team Lead	4	12	12	12	4
G	Process Manager	6	10	1	6	1
H	Developer	4	12	4	4	3
I	Process Manager	3	10	3	3	4
J	Process Manager	1	6	1	4	2
Average		3,9	9,7	4	5,9	2,4

**experience is given in years.*
***by "number of projects" we mean, in how many organizations has the person applied PM / DevOps.*

On average, participants have 4 years of experience in their roles, while having close to 10 years of work experience in IT. The majority has exercised DevOps and PM practices in at least 2 previous organizations. Three separate teams are represented in this exercise; a Service Management team responsible for overseeing the implementation of PM and other ITIL processes in the organization (participants A, G, J), a Service Delivery team managing business interactions with customers and end users (participants C, E, I), and a development team responsible for the maintenance and continuous improvement of Service Management tools utilized in the organization (participants B, D, F, H).

4. Case Study Results

This chapter presents the outcomes of the CS carried out, laying out the results obtained from each of its three stages. The format, structure and product of semi-structured interviews, process analysis and observation, as well as the focus group exercise carried out are documented herein.

4.1 Semi-Structured Interviews

Interviews were started by asking foundational questions regarding DevOps and PM practices, gauging which were known and the level to which they were applied. Although DevOps knowledge and experience varied based on role and background, favoring participants with Development-oriented responsibilities, all interviewees shared a common understanding of the PM process as defined in ITIL. Participants were asked to assess the relevance of DevOps practices across PM activities by having each select one of the following options for each combination: 1, meaning low relevance; 2, meaning relevant; and 3, meaning high relevance. Answers were compiled into Table 4 by adding the values entered for each cell and grading the result using three colors: Light grey indicating slight significance, where the added total of answers was between 10 and 14; dark grey indicating significance, where the added total of answers was between 15 and 19; and black, indicating high significance, with the added total of answers equaling to or being over 20. Opinions and justifications provided by interviewees as they responded to this assessment are found in Table 5.

Table 4 - DevOps Practice significance in the PM lifecycle

	Problem Management Practices						
		Problem Detection	Problem Logging	Problem Investigation and Diagnosis	Known Error Management	Problem Resolution	Problem Closure
DevOps Practices	Shift-Left	8	6	6	10 (F1)	6	6
	Continuous Planning	14 (F2)	12	16 (F3)	17	20 (F4)	12
	Dev & Ops Feedback Loops	22 (F5)	16	21 (F6)	17 (F7)	19 (F8)	17
	Continuous Integration	9	10	12 (F9)	12	16 (F10)	7
	Automated Monitoring	24 (F11)	12 (F12)	17	12	13	12 (F13)
	Application Prototyping	6	7	8	7	9 (F14)	6
	Deployment Automation	5	4	4	4	5	5
	Test Automation	8	7	8	6	9 (F15)	6
	Infrastructure as Code	0	0	0	0	0	0
	Stakeholder Participation	21 (F16)	16 (F17)	14	16	19 (F18)	16
	Process Standardization	9	11	13 (F19)	14 (F20)	12	12 (F21)
	Change Management	13	10 (F22)	12	11	17 (F23)	10 (F24)

Legend: Slight significance; Significant; High significance.

Data resulting from semi-structured interviews indicates that 10 out of the 12 contemplated DevOps practices are, in terms of their application, significant in at least one stage of the Problem lifecycle. The DevOps practices of Continuous Planning, Dev & Ops Feedback Loops, Automated Monitoring, and Stakeholder Participation stand out, and were found to be highly significant to the PM process. Considering captured insight, this is a result of the improved planning and collaboration these practices tend to enable, by allowing development and operations teams to work more closely together. The practices of Shift-Left, Application Prototyping, Deployment and Test Automation, as well as Infrastructure as Code were considered as less significant, the latter neither practiced nor known among selected interviewees.

Table 5 - Insight on DevOps & PM Practice matches

Feedback	DevOps Practice	PM Practice	Comment
F1	Shift Left	Known Error Management	"When a proper Known Error Database is implemented, there are plenty of opportunities for shift-left (…) Known Error workarounds can often be applied at the 1st level of support." "This is a key element of the link between the PM and Knowledge Management process (…) Having this set of practices in place is an enabler to proper Knowledge Management."
F2	Continuous Planning	Problem Detection	"Having a regular forum where potential Problem candidates are discussed can be helpful to the process." "With continuous planning we can continually find issues and obstacles to be addressed via the PM process."
F3		Problem Investigation and Diagnosis	"This would be an important practice to have as it enables better task management (…) knowing where each investigation is on an ongoing manner is helpful for the process."
F4		Problem Resolution	"Solution activities for Problems have to be planned. Having [Continuous Planning] is very relevant as it can expedite the implementation." "Continuous planning sessions, the fact that they allow the opportunity for ongoing discussion, can be a positive element for the process." "This can be used to also prevent Problems even before they get to Production."
F5	Dev & Ops Feedback Loops	Problem Detection	"Being aware of what development is to be implemented, and pointing out risks or issues, is an important contributor to the creation of Problems."
F6		Problem Investigation and Diagnosis	"Performing Problem Root Cause Analysis always requires collaboration. Developers and Operations need to work together when it comes to figuring out the cause of Problems." "Regardless of who actually does the Root Cause Analysis, it is important to have consistent feedback between us process teams and the tech teams. It is how we obtain necessary updates and improve communication."
F7		Known Error Management	"The validation of Known Errors and Workarounds has to be checked and confirmed by the Development side, who often have the technical awareness to approve or reject this."
F8		Problem Resolution	"This combination is needed to make the requirements of a Problem solution clear and align on how that solution will be introduced properly." "It is the Developers who take the lead in the working out of a resolution for a Problem; however, the implementation of it already involves Operations. They should be working together."
F9	Continuous Integration	Problem Investigation and Diagnosis	"A Problem investigation is an ongoing process. It should be easy to track and simple to manage. The idea of Continuous Integration, with new information being added to the Problem piece-by-piece, makes sense."
F10		Problem Resolution	"In the implementation of solutions for Problems, this practice can speed thing up to a higher pace." "This [Continuous Integration] is how we should aim to implement Problem actions."
F11	Automated Monitoring	Problem Detection	"This is a key practice to proactively identify Problems. We need to be aware of what is happening in the environment; having automation helps." "We could link this with the Event management process, working as an input to PM."
F12		Problem Logging	"There may be opportunities to automate the creation and logging of Problem records based on certain monitoring triggers."
F13		Problem Closure	"We could use some sort of automated monitoring to actually confirm the complete resolution of a Problem investigation."
F14	Application Prototyping	Problem Resolution	"The prototyping of a Problem solution could be done."
F15	Test Automation	Problem Resolution	"We could use this practice to test how effective a Problem solution is before we implement it in Prod."
F16	Stakeholder Participation	Problem Detection	"Additional 'eyes on the field' are important to detect things as soon as something goes wrong."
F17		Problem Logging	"The prioritization of a Problem and its classification, based on urgency and impact, is dependent on the Stakeholders insight and participation."
F18		Problem Resolution	"Stakeholder should be involved in the confirming of solutions to the Problems" "They are the best suited to consider, agree, comment on and confirm the solutions to Problem issues."
F19	Process Standardization	Problem Investigation and Diagnosis	"Having standard processes allows us to organize and help the carrying out of investigations. Standardization also more easily points out what may have failed during a Problem."
F20		Known Error Management	"Only with a standardized process within our teams can we ensure that a good Known Error Knowledge Base is in place; it prevents wasted time where we have people investigating matters that are already known or under resolution."
F21		Problem Closure	"In Closure, everything should be clearly documented, and everyone should be aware of and follow the same process for it. (…) the outputs of each activity towards Closure should have a predictable outcome.
F22	Change Management	Problem Logging	"We could trigger the automatic creation of Problems from Changes that are considered 'failed'."
F23		Problem Resolution	"Solutions implemented via PM should also go through a Change Management process."
F24		Problem Closure	"The outcome of the Change can be used as a confirmation that the issue was actually resolved. It is evidence that the Problem is ready for Closure."

4.2 Process analysis and observation

A review of the PM process being applied in the organization by its teams was conducted. Our objective consisted in analyzing its main activities to understand how aligned these are with the broader ITIL PM literature, observe if and what DevOps practices are used over the course of Problem investigations, and how they may influence the results of the process.

According to the PM process applied in the organization, and in line with ITIL literature, a Problem is defined as the "underlying cause of one or more Incidents or potential incidents". Each Problem investigation regardless of source, priority, categorization, or type, progresses through a standard four-stage lifecycle. A team of dedicated Process Managers ensures that each activity of the Problem lifecycle is carried out according to expectations. They aid in defining action owners, create and route Problem tasks in the ITSM tool used by the organization, schedule meetings for Problem validation and handling, and act in the event of escalations or overdue actions. Although they are not involved in the delivery of Root Cause Analysis themselves, they ensure the process is properly driven by building connections between the business and its various Service Providers. A flowchart depicting the 4 key activities of the lifecycle is presented in Figure 3.

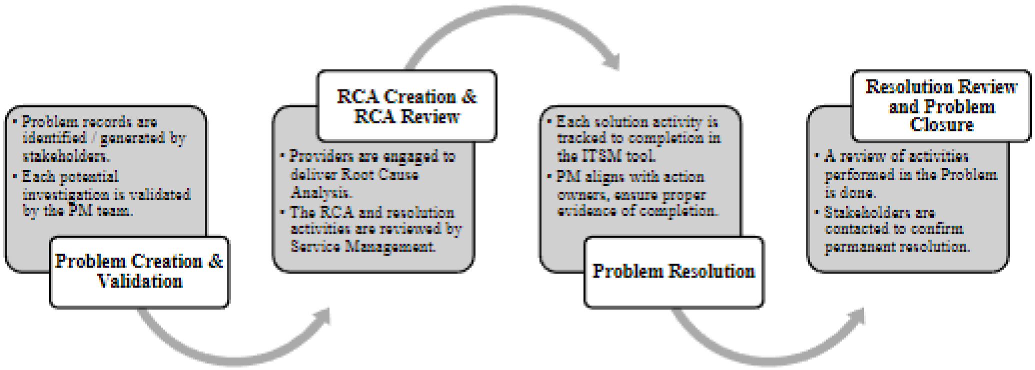


Figure 3 - Four-stage Problem lifecycle

Based on the author's analysis, the PM process implemented in the organization is aligned with the understanding of PM presented in ITIL literature. Observation and study of the four-stage Problem lifecycle being utilized indicates, to a certain extent and in select instances, that some DevOps practices are already being applied within the organization for PM. This includes, for example, continuous Stakeholder Participation in the detection, resolution, and closure phases of Problem records; reliance on a Change Management process to carry out Problem Resolution activities; utilization of automated monitoring tools to identify Problem candidates; ongoing alignment of technical resources and developers, coupled with operations teams, to investigate, diagnose and resolve Problem investigations.

Observed evidence of utilizing DevOps practices in the organizations' current PM process are found in Table 6. It is noteworthy that the 4 most significant DevOps practices for PM identified according to the interview phase (Continuous Planning, Dev & Ops Feedback Loops, Automated Monitoring, and Stakeholder Participation) were all observed at this stage of the CS.

Table 6 - DevOps Practices in Organizations' PM

		Organizational Problem Management Practices			
		Problem Creation & Validation	RCA Creation & Review	Problem Resolution	Resolution Review and Problem Closure
Dev Ops Practices	Continuous Planning	The organization relies on daily PM meetings with Providers, as well as weekly forums with Service Management, to discuss the validation and progress of investigations and respective solution activities, ensuring that they are completed according to agreed timelines or adapt timelines if required.			
	Dev & Ops Feedback Loops	The Development and Operations community works closely to identify candidates for Problem investigations (often proactively, based on insight from ongoing deployments).	The Development and Operations community collaborates iteratively in the development of Root Cause analysis. Both parties are required in order to collect information from Production environments and analyze it to identify cause and propose adequate solutions.	The Development and Operations community works together to plan, implement and monitor each identified solution activity (this would include the development of bug fixes, implementation planning, and ongoing monitoring to measure success rates).	(Not observed in analysis)
	Automated Monitoring	The organization relies of monitoring tools to identify common error trends and relies on this information to initiate Proactive PM.	(Not observed in analysis)	Automated monitoring tools are utilized to measure the success of implemented resolution activities, and as an indicator of a Problem being resolved.	(Not observed in analysis)
	Stakeholder Participation	Stakeholders impacted by Incidents are encouraged to request the creation of Problem record aimed at addressing their cause.	(Not observed in analysis)	(Not observed in analysis)	Confirmation from impacted Stakeholders that a Problem has been fully resolved is actively sought after in the Resolution Review stage of the PM lifecycle.
	Change Management	(Not observed in analysis)	(Not observed in analysis)	Problem resolution activities frequently require the engagement of the Change Management process to ensures all business requirements are adhered to for implementation.	(Not observed in analysis)

4.3 Focus group exercise

Focus group discussions are frequently used as a qualitative approach to deepen the understanding of issues by "draw[ing] from the complex personal experiences, beliefs, perceptions and attitudes of the participants through a moderated interaction" (O.Nyumba et al., 2018). A focus group discussion was held with 5 select individuals from the organization, all of which were part of the group of 10 professionals previously interviewed. Referring to Table 3, these were Interviewees D, F, B, I and J. This selection allowed for representation of a diverse background and experience, with individuals from each of the three involved teams. With this discussion, the authors aim to establish a triangulation of data by linking insight from the focus group, the outcomes of process observation, and the results captured from semi-structured interviews.

The discussion was initiated by presenting an overview of the CS and laying out the purpose and objectives of the exercise. A review of the findings from semi-structured interviews and process analysis, as set forth in previous chapters, was done. Afterwards, we opened the conversation for participants to discuss each Research Question.

On the first RQ of the study: *"Can DevOps practices be applied in each stage of the PM process? If so, which?"*, participants highlighted the DevOps practices of Automated Monitoring, Dev & Ops Feedback Loops, Stakeholder Participation, Change Management, and Process Standardization. These were considered applicable as enablers for the collaboration required during PM activities, in addressing the need for proactivity, identifying emerging issues and risks to be tackled via PM, and the delivering consistency in carrying out and implementing resolution activities. On the second RQ of the study: *"Does DevOps improve the resolution of Problems? If so, how?"*, participants discussed how professionals working in DevOps teams and those working in PM tend to have differing, yet complementary, skillsets. Four DevOps practices were highlighted as having a role in directly improving Problem resolution: Dev & Ops Feedback Loops, Stakeholder Participation, Change Management, and Continuous Planning. Summaries of the made for each RQ are found in Table 7 and 8.

In closing, participants agreed that although without formalization, some DevOps practices are already being applied in various Problem investigations, with positive results, within their organization. For example, the carrying out of detailed Root Cause Analysis was mentioned as a result of continuous feedback loops between Development and Operations; the possibility of quickening the implementation of Problem resolution activities, done through Continuous planning; or the increased identification of issues requiring PM intervention, as a result of greater Stakeholder participation. It was clear to participants that not all DevOps practices are equally relevant to a PM process, but that those which have a role in planning, communicational and collaborative efforts of teams are of significant value.

Table 7 - Focus Group comments on RQ1

DevOps Practice	Comment
Automated Monitoring	Participant J: "The Automated Monitoring practice can quite easily be applied to proactively gather Problem candidates from our systems (…) this is currently one of our main sources for new investigations."
Dev & Ops Feedback Loops	Participant D: "It is important for us to know the results of a new deployment and having a bridge with Operations helps us be quicker in how we respond." Participant J: "This is the most important practice that can be applied (…) the improved communication this and the Continuous Planning practice provide brings a lot of benefit and structure to the work of PM".
Stakeholder Participation	Participant B: "We need to define stakeholder, including those passively observing as well as those actively involved (…) for the detection of Problems, everyone is a valid stakeholder." Participant J: "All stakeholders can contribute to PM (…) their participation is very valuable to us".
Change Management	Participant B: "Many activities done over the course of a Problem require the involvement of Change Management to ensure they are done correctly (…) many Problems are [also] created from Changes that do not have a positive outcome."
Process Standardization	Participant I: "Organizational culture might not be leaning towards having a lot of standardization in the process, but it is needed in order for it be predictable and consistent".

Table 8 - Focus Group comments on RQ2

DevOps Practice	Comment
Dev & Ops Feedback Loops	Participant D: "We could imagine a DevOpsProb team, where the knowledge of process is joined with technical expertise to quickly resolve Problems (…) there can still be specialization, but all working towards the same purpose." Participant B: "Communication between Problem managers and those actually developing solutions is needed to make sure things are done in an organized way and at the right time." Participant I: "Root Causes need to be found quickly and data can be lost (…) feedback is important to make sure people are on the right track to reach a conclusion."
Stakeholder Participation	Participant I: "Other processes may need to be more customer facing, but having more participation from the business in PM makes it possible to know where to focus effort, where to make priorities" Participant J: "If Stakeholders understand the process, they can be important allies over the time it takes to resolve investigations."
Change Management	Participant F: "PM benefits indirectly: If all teams are using this [Change Management], it will naturally feed into PM."
Continuous Planning	Participant J: "Some investigations are very long (…) they have complex actions that need to be consistently monitored. If planning for these Problems is not done continuously, we would lose track of what has been done, what is being done and what still needs to be completed."

5. Conclusion

The potential role of DevOps practices in ITSM environments is, by and large, unexplored territory. A CS was carried out in an organization where both DevOps and PM are applied according to modern standards. Professionals working with various backgrounds, experiences and roles in the organizations' IT department were interviewed on their understanding of DevOps, PM practices, and on how these could potentially be used collaboratively. Process observation and analysis, as well as a focus group exercise were carried out. Various trends were observed over the course of the CS, with some DevOps practices clearly standing out from others both in terms of their applicability and potential benefit to a PM process.

In regard to the first Research Question, data resulting from semi-structured interviews indicates that 10 out of the 12 contemplated DevOps practices hold some degree of significance at at least one stage in the Problem lifecycle. Of these practices, Continuous Planning, Dev & Ops Feedback Loops, Automated Monitoring, and Stakeholder Participation stand out as highly significant. These same practices, plus Change Management and Process Standardization, were also highlighted during the process observation stage and focus group exercise of the CS. The practices of Shift-Left, Continuous Integration, Application Prototyping and Test Automation we found as moderate to slightly significant during the interview phase but were not observed in latter research stages. Deployment Automation and Infrastructure as Code were not found to be significant to PM, according to this research.

On the second Research Question, focusing on *if* and *how* DevOps practices can improve a PM process, more qualitative results can be considered. Insight captured from the CS indicate that DevOps is relevant and beneficial to a PM process, with some practices having the potential to significantly improve how it is applied in an organization. The underlying reasons for this appear to be strongly aligned with what was reviewed in the Theoretical Background; particularly when referring to the speed, improved collaboration, and silo removal that DevOps offers to organizations. The aforementioned practices of Dev & Ops Feedback loops and Stakeholder Participation, when implemented, were found to improve the collaboration between teams and departments in the organization, enabling improved Problem handling. The Continuous Planning and Change Management practices were observed in both the communication taking place between the PM Team, Service Management, Providers, and customers themselves, as well as the structure given to Problem resolution activities. The Automated Monitoring practice was seen in action through the monitoring tools implemented by the organization, proactively identifying new Problem candidates and preventing recurring incidents. Practices such as Application Prototyping, Continuous Integration, Test and Deployment Automation, and Shift-Left were discussed as being more pertinent to teams working specifically in Development areas, and not considered as relevant for a PM process.

In the absence of studies focused on the relationship between DevOps and ITSM, particularly PM, this research provides a valuable contribution to professionals and practitioners alike. For organizations adopting DevOps in an IT service management environment, the insight captured on the significance and benefit each DevOps practice may have for a PM process can motivate its increased application in business, generating benefits as implementation takes place. For organizations and individuals who have already adopted DevOps, and are operating in or with a PM process, the knowledge captured within this research can be used to focus efforts on maturing the DevOps practices that can lead to greater improvements in PM. This research also embodies an academic contribution to a novel yet expanding body of knowledge for the interactions between DevOps practices and IT processes.

This research does have some limitations. Although the authors sought to encompass a comprehensive variety of DevOps practices, some were not known nor applied by the participants that contributed to the Case Study; this is the case for the DevOps practice of Infrastructure as Code. As such, it was not possible to conclude the potential role of said practice in the PM process. Other DevOps practices may also be proposed to have been included in the study. Furthermore, the Case Study relied on experience and insight from professionals working in various teams, DevOps, Service and Process Management, within a single organization. The PM process that was observed within the organization was found to be aligned with the ITIL framework, yet other perspectives may exist.

Future work possibilities were identified over the course of this research. In it, we analyzed the scope of DevOps practices in a PM process, akin to what was done by Faustino (2018) for Incident Management. Other ITIL processes, such as Knowledge Management, Release Management and Deployment Management would also potentially benefit from being subject to this investigation. Additionally, despite providing examples of instances where DevOps practices were practically observed in a live PM process, the authors did not explore the challenges and effort required for their implementation. It may be of value to consider a new Case Study aimed at analyzing and documenting how DevOps practices can be adopted by IT processes. Lastly, the new ITIL iteration (ITIL4) acknowledges DevOps practices; research could be performed on how the transition from previous ITIL iterations to ITIL4 can influence the integration of DevOps practices such as those contained in this research.

6. Bibliographical references

Abdelkebir, S., Maleh, Y., & Belaissaoui, M. (2017). An Agile Framework for ITS Management In Organizations: A Case Study Based on DevOps. *Proceedings of the 2nd International Conference on Computing and Wireless Communication Systems*, 1–8. https://doi.org/10.1145/3167486.3167556

Aberdeen, T. (2013). Yin, R. K. (2009). Case study research: Design and methods (4th Ed.). Thousand Oaks, CA: Sage. *The Canadian Journal of Action Research*, *14*(1), 69–71. https://doi.org/10.33524/cjar.v14i1.73

Aiello, B., & Sachs, L. (2016). *Agile Application Lifecycle Management: Using DevOps to Drive Process Improvement* (1st ed.). Addison-Wesley Professional.

Alsolamy, A. A., Khan, U. A., & Khan, P. M. (2014). IT-business alignment strategy for business growth. *2014 International Conference on Computing for Sustainable Global Development (INDIACom)*, 364–366. https://doi.org/10.1109/IndiaCom.2014.6828160

Alt, R., Auth, G., & Kögler, C. (2018). Transformation of Consulting for Software-Defined Businesses – Lessons from a DevOps Case Study in a German IT Company. Em *Contributions to Management Science*. https://doi.org/10.1007/978-3-319-95999-3_19

Badinelli, R., Barile, S., Ng, I., Polese, F., Saviano, M., & Di Nauta, P. (2012). Viable service systems and decision making in service management. *Journal of Service Management*, *23*, 498–526. https://doi.org/10.1108/09564231211260396

Benitez, J., Chen, Y., Teo, T. S. H., & Ajamieh, A. (2018). Evolution of the impact of e-business technology on operational competence and firm profitability: A panel data investigation. *Information & Management*, *55*(1), 120–130. https://doi.org/10.1016/j.im.2017.08.002

Cartlidge, A., & Lillycrop, M. (2009). *An Introductory Overview of ITIL V3*. ITSMF.

Cater-Steel, A., Toleman, M., & Tan, W.-G. (2006). Transforming IT service management- The ITIL impact. *ACIS 2006 Proceedings - 17th Australasian Conference on Information Systems*.

Chen, L. (2017). Continuous Delivery: Overcoming adoption challenges. *Journal of Systems and Software*, *128*, 72–86. https://doi.org/10.1016/j.jss.2017.02.013

Colomo-Palacios, R., Fernandes, E., Soto-Acosta, P., & Larrucea, X. (2018). A case analysis of enabling continuous software deployment through knowledge management. *International Journal of Information Management*, *40*, 186–189. https://doi.org/10.1016/j.ijinfomgt.2017.11.005

de Kort, W. (2016). What Is DevOps? Em W. de Kort (Ed.), *DevOps on the Microsoft Stack* (pp. 3–8). Apress. https://doi.org/10.1007/978-1-4842-1446-6_1

Díaz, J., Almaraz, R., Pérez, J., & Garbajosa, J. (2018). DevOps in practice: An exploratory case study. *Proceedings of the 19th International Conference on Agile Software Development: Companion*, 1–3. https://doi.org/10.1145/3234152.3234199

Díaz, J., Perez, J. E., Yague, A., Villegas, A., & de Antona, A. (2019). DevOps in Practice – A Preliminary Analysis of Two Multinational Companies. Em X. Franch, T. Männistö, & S. Martínez-Fernández (Eds.), *Product-Focused Software Process Improvement* (pp. 323–330). Springer International Publishing. https://doi.org/10.1007/978-3-030-35333-9_23

Dobson, P. (1999). *Approaches to Theory Use in Interpretive Case Studies—A Critical Realist Perspective*.

Dyck, A., Penners, R., & Lichter, H. (2015). Towards definitions for release engineering and DevOps. *Proceedings of the Third International Workshop on Release Engineering*, 3.

Ebert, C., Gallardo, G., Hernantes, J., & Serrano, N. (2016). DevOps. *IEEE Software*, *33*(3), 94–100. https://doi.org/10.1109/MS.2016.68

Farroha, S., & Farroha, D. (2014). *A Framework for Managing Mission Needs, Compliance, and Trust in the DevOps Environment*. 288–293. https://doi.org/10.1109/MILCOM.2014.54

Faustino, J. P. C. (2018). *DevOps practices in incident management process*. https://repositorio.iscte-iul.pt/handle/10071/18294

Forsgren, N., & Humble, J. (2015). DevOps: Profiles in ITSM Performance and Contributing Factors. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.2681906

Guerriero, M., Ciavotta, M., & Gibilisco, G. P. (2015). *SPACE4Cloud: A DevOps environment for multi-cloud applications*. 29–30. https://doi.org/10.1145/2804371.2804378

Gupta, R. K., Venkatachalapathy, M., & Jeberla, F. K. (2019). Challenges in Adopting Continuous Delivery and DevOps in a Globally Distributed Product Team: A Case Study of a Healthcare

Organization. *2019 ACM/IEEE 14th International Conference on Global Software Engineering (ICGSE)*, 30–34. https://doi.org/10.1109/ICGSE.2019.00020

How to Do Your Case Study. (2021, Setembro 20). SAGE Publications Ltd. https://uk.sagepub.com/en-gb/eur/how-to-do-your-case-study/book242255

Hüttermann, M. (2012). *DevOps for Developers*. https://doi.org/10.1007/978-1-4302-4570-4

ITIL® Foundation Handbook 3rd ed. (2012). (sem data). AXELOS. Obtido 17 de Dezembro de 2020, de https://www.axelos.com/store/Book/itil-foundation-handbook-3rd-ed-2012

Jamous, N., Bosse, S., Goerling, C., Hintsch, J., Khan, A., Kramer, F., Muller, H., & Turowski, K. (2016). *Towards an IT Service Lifecycle Management (ITSLM) Concept*. 29–38. https://doi.org/10.1109/ES.2016.10

Jiménez, M., Rivera, L. F., Villegas, N. M., Tamura, G., Müller, H. A., & Gallego, P. (2019). DevOps' Shift-Left in Practice: An Industrial Case of Application. Em J.-M. Bruel, M. Mazzara, & B. Meyer (Eds.), *Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment* (pp. 205–220). Springer International Publishing. https://doi.org/10.1007/978-3-030-06019-0_16

Kamuto, M. B., & Langerman, J. J. (2017). Factors inhibiting the adoption of DevOps in large organisations: South African context. *2017 2nd IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT)*, 48–51. https://doi.org/10.1109/RTEICT.2017.8256556

Kim, G., Debois, P., Willis, J., Humble, J., & Allspaw, J. (2016). *The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations* (Illustrated edition). IT Revolution Press.

Krishna Kaiser, A. (2018). Problem Management Adaption. Em A. Krishna Kaiser (Ed.), *Reinventing ITIL® in the Age of DevOps: Innovative Techniques to Make Processes Agile and Relevant* (pp. 201–230). Apress. https://doi.org/10.1007/978-1-4842-3976-6_8

Kuusinen, K., Balakumar, V., Jepsen, S. C., Larsen, S. H., Lemqvist, T. A., Muric, A., Nielsen, A. Ø., & Vestergaard, O. (2018). A Large Agile Organization on Its Journey Towards DevOps. *2018 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, 60–63. https://doi.org/10.1109/SEAA.2018.00019

Laukkanen, E., Paasivaara, M., Itkonen, J., Lassenius, C., & Arvonen, T. (2017). Towards Continuous Delivery by Reducing the Feature Freeze Period: A Case Study. *2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering in Practice Track (ICSE-SEIP)*, 23–32. https://doi.org/10.1109/ICSE-SEIP.2017.21

Limanto, A., Khwarizma, A., Imelda, Rumagit, R., Pietono, V., Halim, Y., & Liawatimena, S. (2017). *A study of Information Technology Infrastructure Library (ITIL) framework implementation at the various business field in Indonesia*. 1–4. https://doi.org/10.1109/CITSM.2017.8089244

Luo, A., Fu, J., & Liu, J. (2016). *An impact analysis method of business processes evolution in enterprise architecture*. 733–737. https://doi.org/10.1109/PIC.2016.7949595

Luz, W. P., Pinto, G., & Bonifácio, R. (2019). Adopting DevOps in the real world: A theory, a model, and a case study. *Journal of Systems and Software*, *157*, 110384. https://doi.org/10.1016/j.jss.2019.07.083

Lwakatare, L. E., Kilamo, T., Karvonen, T., Sauvola, T., Heikkilä, V., Itkonen, J., Kuvaja, P., Mikkonen, T., Oivo, M., & Lassenius, C. (2019). DevOps in practice: A multiple case study of five companies. *Information and Software Technology*, *114*, 217–230. https://doi.org/10.1016/j.infsof.2019.06.010

Marrone, M., & Kolbe, L. (2011). Impact of IT Service Management Frameworks on the IT Organization. *Business & Information Systems Engineering*, *3*, 5–18. https://doi.org/10.1007/s12599-010-0141-5

Melendez, K., Dávila, A., & Pessoa, M. (2016). Information technology service management models applied to medium and small organizations: A systematic literature review. *Computer Standards & Interfaces*, *47*, 120–127. https://doi.org/10.1016/j.csi.2015.10.001

Modell, S. (2005). Triangulation between case study and survey methods in management accounting research: An assessment of validity implications. *Management Accounting Research*, *16*(2), 231–254. https://doi.org/10.1016/j.mar.2005.03.001

Mohan, V., Ben Othmane, L., & Kres, A. (2018). *BP: Security Concerns and Best Practices for Automation of Software Deployment Processes: An Industrial Case Study*. 21–28. https://doi.org/10.1109/SecDev.2018.00011

Mora, M., Marx-Gomez, J., O'Connor, R., Raisinghani, M., & Gelman, O. (2015). An Extensive Review of IT Service Design in Seven International ITSM Processes Frameworks. *International Journal of Information Technology and Systems Approach*, *8*, 69–90. https://doi.org/10.4018/ijitsa.2015010104

Nybom, K., Smeds, J., & Porres, I. (2016). On the Impact of Mixing Responsibilities Between Devs and Ops. Em H. Sharp & T. Hall (Eds.), *Agile Processes, in Software Engineering, and Extreme Programming* (pp. 131–143). Springer International Publishing. https://doi.org/10.1007/978-3-319-33515-5_11

O.Nyumba, T., Wilson, K., Derrick, C. J., & Mukherjee, N. (2018). The use of focus group discussion methodology: Insights from two decades of application in conservation. *Methods in Ecology and Evolution*, *9*(1), 20–32. https://doi.org/10.1111/2041-210X.12860

Park, H. Y., Jung, S. H., Lee, Y., & Jang, K. C. (2006). The Effect of Improving IT Standard in IT Governance. *2006 International Conference on Computational Inteligence for Modelling Control and Automation and International Conference on Intelligent Agents Web Technologies and International Commerce (CIMCA'06)*, 22–22. https://doi.org/10.1109/CIMCA.2006.210

Perera, P., Silva, R., & Perera, I. (2017). *Improve software quality through practicing DevOps*. 1–6. https://doi.org/10.1109/ICTER.2017.8257807

Perry, D., Sim, S., & Easterbrook, S. (2004). *Case studies for software engineers*. *26*, 736–738. https://doi.org/10.1109/ICSE.2004.1317512

Pilorget, L., & Schell, T. (2018). *IT Management: The art of managing IT based on a solid framework leveraging the company's political ecosystem*. Springer Vieweg. https://doi.org/10.1007/978-3-658-19309-6

Radhakrishnan, R., Mark, K., & Powell, B. (2008). IT service management for high availability. *IBM Systems Journal*, *47*, 549–561. https://doi.org/10.1147/SJ.2008.5386521

Riungu-Kalliosaari, L., Mäkinen, S., Lwakatare, L. E., Tiihonen, J., & Männistö, T. (2016). DevOps Adoption Benefits and Challenges in Practice: A Case Study. Em P. Abrahamsson, A. Jedlitschka, A. Nguyen Duc, M. Felderer, S. Amasaki, & T. Mikkonen (Eds.), *Product-Focused Software Process Improvement* (pp. 590–597). Springer International Publishing. https://doi.org/10.1007/978-3-319-49094-6_44

Šćekić, M., Gazivoda, M., Šćepanović, S., & Nikolić, J. (2018). Application of DevOps approach in developing business intelligence system in bank. *2018 7th Mediterranean Conference on Embedded Computing (MECO)*, 1–4. https://doi.org/10.1109/MECO.2018.8406047

Senapathi, M., Buchan, J., & Osman, H. (2018). DevOps Capabilities, Practices, and Challenges: Insights from a Case Study. *EASE*. https://doi.org/10.1145/3210459.3210465

Shahin, M., Babar, M. A., & Zhu, L. (2016). The Intersection of Continuous Deployment and Architecting Process: Practitioners' Perspectives. *Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, 1–10. https://doi.org/10.1145/2961111.2962587

Sharifi, M., Ayat, M., Rahman, A. A., & Sahibudin, S. (2008). Lessons learned in ITIL implementation failure. *2008 International Symposium on Information Technology*, *1*, 1–4. https://doi.org/10.1109/ITSIM.2008.4631627

Sharma, S., Coyne, B., Wiley, J., & Inc, S. (2015). *DevOps For Dummies, 2nd IBM Limited Edition*. 76.

Šmite, D., Gonzalez-Huerta, J., & Moe, N. B. (2020). "When in Rome, Do as the Romans Do": Cultural Barriers to Being Agile in Distributed Teams. Em V. Stray, R. Hoda, M. Paasivaara, & P. Kruchten (Eds.), *Agile Processes in Software Engineering and Extreme Programming* (pp. 145–161). Springer International Publishing. https://doi.org/10.1007/978-3-030-49392-9_10

Soni, M. (2015). *End to End Automation on Cloud with Build Pipeline: The Case for DevOps in Insurance Industry, Continuous Integration, Continuous Testing, and Continuous Delivery* (pp. 85–89). IEEE Computer Society. https://doi.org/10.1109/CCEM.2015.29

Ståhl, D., Mårtensson, T., & Bosch, J. (2017). The Continuity of Continuous Integration: Correlations and Consequences. *Journal of Systems and Software, 127*. https://doi.org/10.1016/j.jss.2017.02.003

Sukmandhani, A. A., Wijanarko, B. D., Gunawan, E., Pratama, D., Gaol, F. L., & Sutedja, I. (2017). Measurement effectiveness and efficiency to improve the IT services using ITSM. *2017 International Conference on Information Management and Technology (ICIMTech)*, 334–339. https://doi.org/10.1109/ICIMTech.2017.8273561

Tashakkori, A., & Creswell, J. (2007). Editorial: The New Era of Mixed Methods. *Journal of Mixed Methods Research*, *1*, 3–7. https://doi.org/10.1177/2345678906293042

Varga, S., Barreto, G., & Battaglin, P. D. (2019). Increasing Information Systems Availabiliy Through Accuracy, Awareness, Completeness and Manageability of ITSM. *2019 14th Iberian Conference on Information Systems and Technologies (CISTI)*, 1–4. https://doi.org/10.23919/CISTI.2019.8760686

Virmani, M. (2015). Understanding DevOps bridging the gap from continuous integration to continuous delivery. *Fifth International Conference on the Innovative Computing Technology (INTECH 2015)*, 78–82. https://doi.org/10.1109/INTECH.2015.7173368

Wahaballa, A., Wahballa, O., Abdellatief, M., Xiong, H., & Qin, Z. (2015, Setembro 23). *Toward unified DevOps model*. https://doi.org/10.1109/ICSESS.2015.7339039

Wang, Q., Song, J., Liu, L., Luo, X., & Xinhua, E. (2010). Building IT-based incident management platform. *5th International Conference on Pervasive Computing and Applications*. https://doi.org/10.1109/ICPCA.2010.5704127

Zainal, Z. (2007). Case study as a research method. *Jurnal Kemanusiaan*, *9*.

390 *Int. J. Services and Operations Management, Vol. 44, No. 3, 2023*

The influence of DevOps practices in ITSM processes

João Faustino* and Rúben Pereira

Department of Information Science and Technology,
Instituto Universitário de Lisboa (ISCTE-IUL),
Avenida das Forças Armadas, Lisboa, 1649-026, Portugal
Email: jpcfo11@iscte-iul.pt
Email: ruben.filipe.pereira@iscte-iul.pt
*Corresponding author

Miguel Mira da Silva

Instituto Superior Técnico,
University of Lisbon,
Av. Rovisco Pais 1, Lisboa, 1049-001, Portugal
Email: mms@tecnico.ulisboa.pt

Abstract: In an information technology (IT) perspective, the world is evolving at a great speed. The IT services have expanded, most organisations started to implement processes that enable the lifecycle of these IT services. The discipline that covers all these processes is called IT service management (ITSM). Nevertheless, ITSM nowadays is seen as a bureaucratic discipline, where the process of creation and support of the IT services are two separate teams. There is an emerging agile culture called DevOps that aims to bridge these two teams: development and operations. The focus of this bridge-into-one team is to deliver quality software and secure its operation. This research systematic analyses the literature and shed light on how DevOps practices can assist ITSM processes. This research concludes that *automated deployment* as the most useful practice and *incident and problem management* is the most beneficial processes. Future directions are pointed.

Keywords: DevOps practices; agile; ITSM processes; influence.

Reference to this paper should be made as follows: Faustino, J., Pereira, R. and da Silva, M.M. (2023) 'The influence of DevOps practices in ITSM processes', *Int. J. Services and Operations Management*, Vol. 44, No. 3, pp.390–407.

Biographical notes: João Faustino is an IT consultant. He is also a PhD student in Information Science and Technology at the ISCTE-IUL where he also graduated as an MSc in Computer Engineering. Most of his career, he has worked as a consultant on the financial services industry, on application management teams for both corrective and enhancement maintenance of core systems for insurance companies. Besides his professional work, which is mainly related with Oracle technologies, he is very enthusiastic about open-source technologies, data science, mobile development, and process optimisation. Most of his research is about IT service management and DevOps.

Rúben Pereira is an Assistant Professor at the ISCTE. He has a PhD in Information Systems at the Instituto Superior Técnico where he also graduated as a Master's in Computer Engineering and Computer Science. He has been a consultant in several industries, such as services, banking, telecommunications, and e-commerce, among others. He is the author of several scientific papers in the area of information technology services management and information technology governance, covering the most known IT frameworks like ITIL and COBIT. His areas of scientific interest extend to information technology risk management, business process management, continuous improvement and innovation, process optimisation, digital transformation, DEVOPS, robotic process automation, among others.

Miguel Mira da Silva is currently an Associate Professor of Information Systems in the University of Lisbon, leader of the research group 'Digital Transformation' at INOV, and coordinator of the MISE online MSc degree. He has a PhD in Computing Science from the University of Glasgow and an MSc in Management from the London Business School. He created five companies, published four books and more than 200 research papers, managed dozens of research and consulting projects, supervised 11 PhDs and 160 MSc students, and created a MOOC about digital transformation. His current interests include digital transformation, IT governance, and online learning.

1 Introduction

Technologic evolution is impacting parties worldwide. With this evolution, organisations started to invest on IT to keep up with their competition, despite of all the changes that IT also brings, such as security threads (Luo et al., 2017; Benitez et al., 2018). The IT becomes a fundamental part of the organisations to support the business processes (Alsolamy et al., 2014), so the term IT Service starts to grow (Cannon and Wheeldon, 2007). One of the objectives of most of the organisations is to become sustainable (Pereira and Serrano, 2020) and since organisations are investing in IT, it is imperative for organisations to measure the behaviour and performance of IT services.

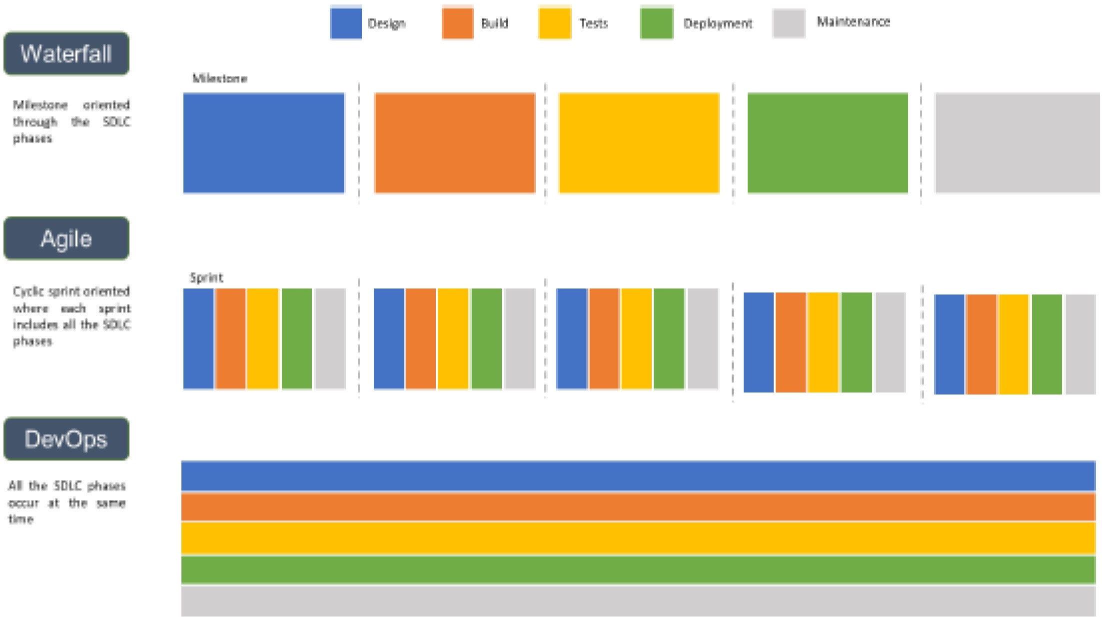
Therefore, IT service management (ITSM) becomes fundamental (Lucio-Nieto and Colomo-Palacios, 2012; El Yamami et al., 2019). ITSM is a set of activities that goes from the aligning and the understanding of the need of an IT service, through its development, until its operation (e Abreu et al., 2010).

Several ITSM frameworks exist to assist organisations. The most globally used framework is ITIL (Aguiar et al., 2018). ITIL is known by improving the IT services quality and have a great efficiency due to the process standardisation (Hochstein et al., 2005; Barcelo-Valenzuela and Leal-Pompa, 2020).

To surpass some challenges felt on the software development (SD), the agile SD methodologies started to be present on the day-by-day of the organisations (Kettunen, 2009). These agile SD methodologies follow the agile manifesto which privilege people and their relationships over processes and tools (Beck et al., 2001).

Even though recent agile movements have brought the business and IT to work together, there was a gap between IT development and IT operations, where a culture called DevOps bridge that gap (Guerriero et al., 2016). It is possible to see the evolution on how software development lifecycle (SDLC) has been evolving over the time as it can be seen in Figure 1. The SDLC phases of DevOps never end contrary to waterfall and agile. This is due to the mindset of continuous improvement promoted by DevOps culture.

Figure 1 Evolution of SDLC adapted from Rolf Consulting (2020) (see online version for colours)



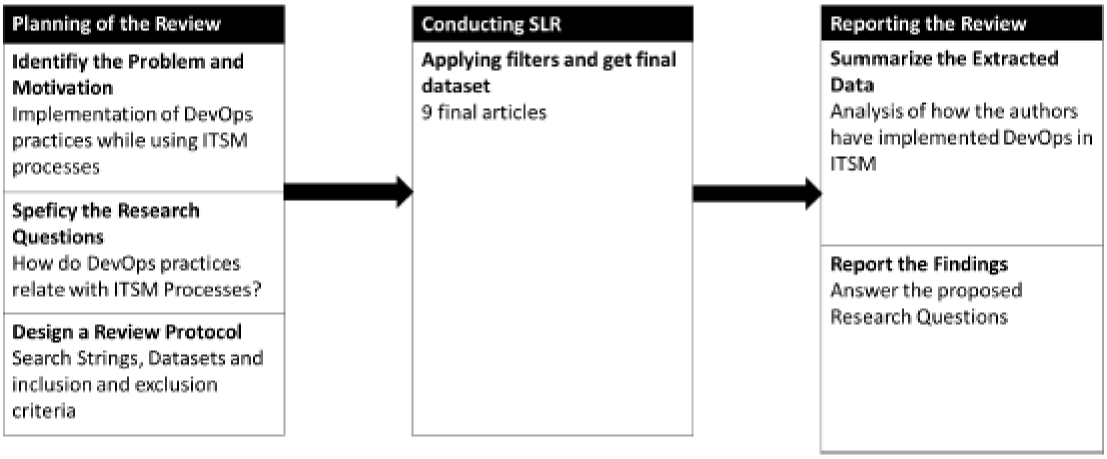
Based on the previous statements, one can conclude that ITSM essentially uses processes to manage IT Services, but, when looking to the evolution of SD, it shows that the relationship between people is privileged over the processes. This shows a contradictory tendency between the implementation of ITSM processes and the usage of the recent SD methodologies. However, DevOps is focused on both people and processes, thus being a possible relationship between ITSM and DevOps.

Despite some evidences that DevOps and ITSM can coexist (Kaiser, 2018) and that ITSM processes can benefit from DevOps adoption (Silva et al., 2018), literature lacks a broader study investigating how DevOps practices and ITSM processes relate. Therefore, this research aims to improve the body of knowledge with further insights on how DevOps practices can be used in ITSM processes.

2 Research methodology

To pursue our research objective, the systematic literature review (SLR) methodology was selected. A SLR is a proper methodology to synthesise existing work related with the research subject (Kitchenham, 2004). This research adopts the steps proposed by Kitchenham (Figure 2). To add more rigor to a SLR must follow three different characteristics such as; being explicit where all the procedures and decisions are described, comprehensive in its scope by including all the relevant materials and reproducible, so other authors who also want to perform reviews about the same topic (Okoli, 2015).

Figure 2 Design of the SLR protocol



3 Planning and conducting of the review

Some authors have already published about the implementation of the DevOps practices, however, Jabbari et al. (2016), performed a review of the existing literature where they have summarised the existing practices of DevOps, for this research it will also be used those practices.

Therefore, this research intends to shed light on how DevOps practices relate with ITSM processes. To pursue this research goal a SLR is performed to identify studies reporting DevOps practices being used in ITSM processes.

To perform the SLR, the following digital databases where analysed:

- IEEE Xplore Digital Library (https://ieeexplore.ieee.org/Xplore/home.jsp)
- ResearchGate (https://www.researchgate.net)
- SpringerLink (https://link.springer.com)
- Elsevier (https://www.sciencedirect.com/)
- ACM (https://dl.acm.org).

The chosen search string is DevOps AND practices AND (ITSM OR ITIL OR ITSM OR ISO 20000). The SLR took place between February and April 2020.

After the initial search using the above string, some inclusion and exclusion criteria as well as filters were applied. These inclusion and exclusion criteria can be seen in Table 1 and the filters can be seen in Table 2.

Table 1 Inclusion and exclusion criteria

Inclusion	*Exclusion*
Articles written in Portuguese or English	Articles not written in Portuguese or English
Articles from journals, conference proceedings and books	Non-scientific articles from journals, conference proceedings and books
Articles from 2007 onwards	Articles before 2007

Table 2 SLR filters

Filter ID	*Filter description*
F.1	Removal of duplicate articles
F.2	The title should contain the words or be related with IT service management or DevOps
F.3	The abstract should show a relationship between DevOps, ITSM, ITIL or ISO 20000
F.4	Full read of the article to understand if it fits in the scope

The DevOps concept was started to be spoken after conference entitled 'agile infrastructure and operations' at 2008 (Lwakature, 2017). Thus, the authors have only search for articles published after 2007.

Table 3 Filter application

Databases	*Only with keywords and inclusion criteria*	*Applying F.1*	*Applying F.2*	*Applying F.3*	*Applying F.4*
IEEExplore	12	12	4	0	0
ResearchGate	37	33	15	3	1
SpringerLink	142	86	34	9	7
Elsevier	106	53	4	0	0
ACM	20	12	6	2	1
Total	317	196	63	14	9

Table 3 details the number of articles that were found on each database and how many were remaining after applying each filter.

Due to the novelty of DevOps culture and due to the evolution of the SDLC the low number of found literature was expected.

4 Reporting of the review

This section aims to detail the outputs of the thorough analysis of the articles selected in Section 3. Overall, the relationship between ITSM and DevOps has not been well explored so far in literature since only nine articles passed our filters. DevOps culture is quite recent which may lead to this low number of articles.

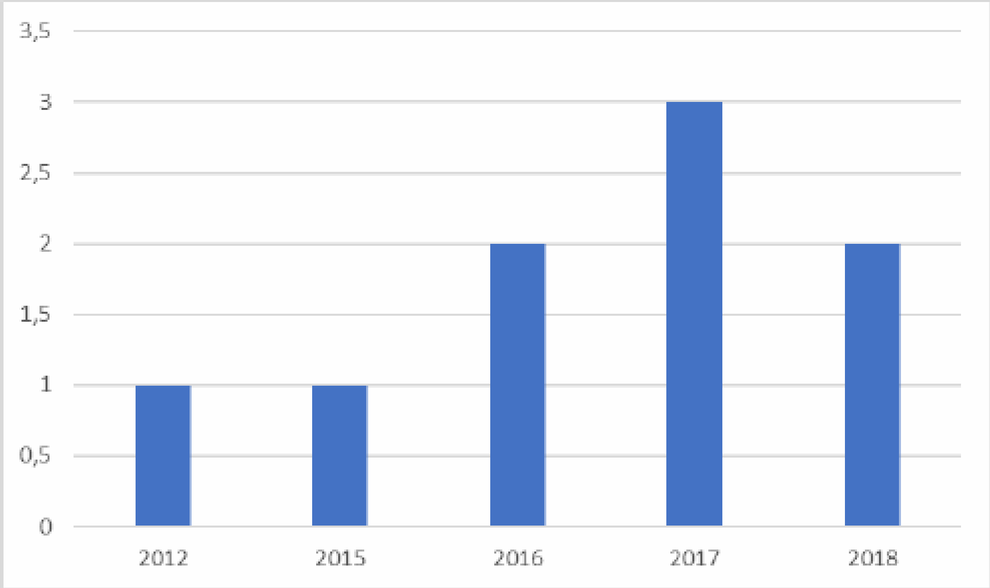
4.1 Data extraction analysis

As one can see in Table 3 from Section 3, the authors have collected articles from five digital libraries and when applying the inclusion criteria plus the search strings defined in Section 3 there is a total of 317 articles. It is also possible to conclude that F.1 and F.2 were the filters with most impact on the number of articles reduced. After applying all the filters, the final set of articles to be analysed is nine.

As referred in Section 3, the DevOps concept born in 2008, so the authors tried to build a timeline of the evolution of the relationship between DevOps and ITSM

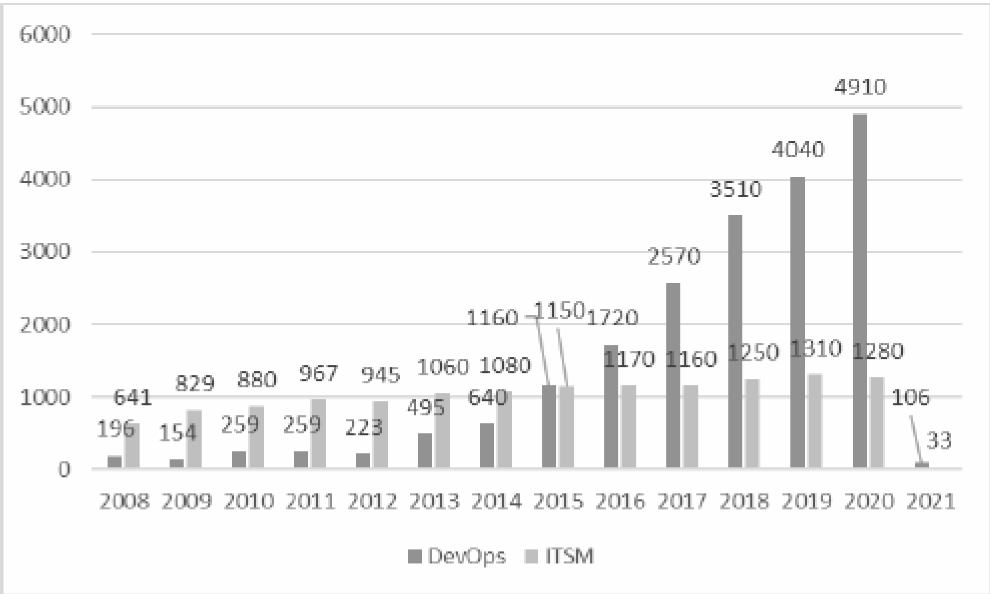
(Figure 3) and the evolution of literature for each subject (Figure 4) based on Google Scholar (https://scholar.google.com/).

Figure 3 Chronologic DevOps and ITSM relationship publications (see online version for colours)



It is possible to see in Figure 3 that despite its existence since 2008, that only in the most recent years this relation began to be explored in literature. The first article date from 2012 but most of the articles found for this subject are from 2017. It is also possible to see an increment of articles across the years between 2012 and 2017.

Figure 4 Chronologic publications of DevOps and ITSM



In Figure 4 one can see that both subjects are being a target of a lot of scientific studies, moreover, it is possible to see a growth of studies about DevOps, while ITSM had a smaller growth of studies, compared with DevOps, but it still a large amount of studies.

Aiming to explore the relationship between the ITSM processes and the DevOps practices, several concepts were elicited along with the proper references as can be seen in Table 6. The authors have split these concepts in two approaches:

- *generic* – analyse the relationship between ITSM/ITIL and DevOps practices where the implementation of DevOps was transversal way and not to certain ITSM/ITIL processes
- *objective* – analyse the relationship between ITSM/ITIL processes sand DevOps where the implementation of DevOps was applied to certain ITSM/ITIL processes

On the following sections the authors describe the findings for each of the main concepts listed in Table 5.

Table 5 Concepts to be analysed

Approach	*Concept*	*Author*	*# of references*
Generic	DevOps as model for ITSM	Hüttermann (2012), Waschke (2015), Forsgren and Humble (2016), Keller (2017), Sahid et al. (2017), Kaiser (2018) and Pilorget and Schell (2018)	7
Objective	Change management	Cuppett (2016), Ravichandran et al. (2016), Keller (2017) and Kaiser (2018)	4
	Release management	Keller (2017) and Kaiser (2018)	2
	Incident management	Keller (2017) and Kaiser (2018)	2
	Problem management	Kaiser (2018)	1
	Event management	Keller (2017)	1
	Service request and service catalogue management	Keller (2017)	1

4.2 *Report findings*

In this section, it is intended to report the findings of the analysis of the literature found in Table 5 from Section 4.1. This section is divided in subsections for each concept from Table 5. The approach taken by the authors to analyse this concept was focused on the concept itself and describe the conclusions taken from the analysis from the literature.

4.2.1 *DevOps as model for ITSM*

Forsgren and Humble (2016) claim that DevOps is a new ITSM paradigm to deliver and operate services for a better response to the market. The work of these authors is based on web-based surveys that were available for technical professionals of all specialties involved in DevOps. From an ITSM perspective, shorter lead times and more frequent releases lead to poor stability and quality in the operational environment (Office Government Commerce, 2007). Some of the DevOps practices applied by the respondents were also listed, such as automated tests and deployment automation, continuous integration, and infrastructure-as-code. Moreover, Forsgren and Humble (2016) have defined three IT Performance profiles (high, medium and low), based in two variables: throughput (based in deployment frequency and lead time to delivery) and stability (mean time to restore). The investigation of these authors points that the higher IT performance profile that has a greater throughput led to more stability. This shows a possible benefit of implementation of DevOps practices in an ITSM environment. The authors claim that strong organisational cultural shift is needed, to give privilege to better relationships between departments, instead of a siloed culture as Sharifi et al. (2008) describes about ITIL framework.

Another, Hüttermann (2012), recognises that DevOps is the link between SD and operations, where operations is influenced by the ITSM processes. DevOps came to bring the operations to inside of the agile processes. With DevOps, the IT teams consist in developers, testers, quality assurance and operators. However, individually, developers are intended to develop and deliver more and new systems' functionalities, while operators are intended to maintain stability. Thus, DevOps team members should keep in mind both priorities: deliver new functionalities and promote stability.

Keller (2017) also supports Forsgren and Humble (2016) ideas, arguing that short release periods bring more quality and stability to applications. This investigation is centred on the automation in a ITSM environment, where DevOps is referred by being known as continuous delivery. However, there are some challenges that the author highlight on the adoption of this automation, likewise, the rigidity of the ITIL framework, implementation of the ITSM processes and mentality that everything should be marked as a ticket or a request, while sometimes a simple conversation between the service desk and the end users should be enough. This author has also identified several ITSM processes where DevOps practices can be applied. This is described on the following chapters.

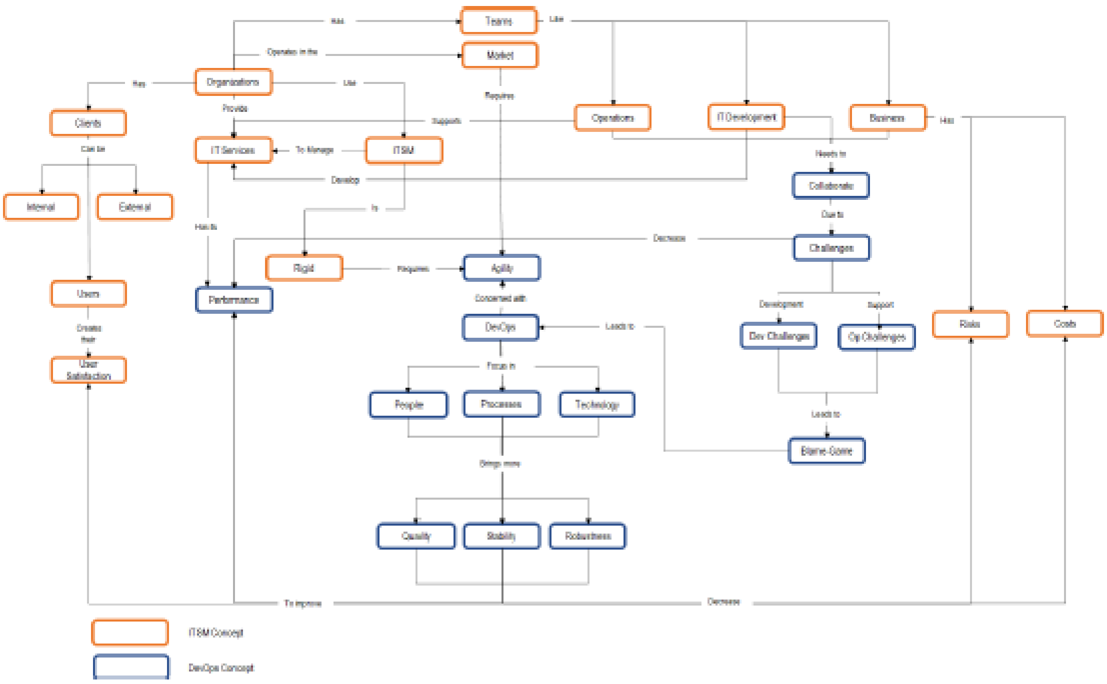
Pilorget and Schell (2018) recognise that DevOps can be implemented on a ITSM environment. These authors also mention that this brings a discussion between the agility and robustness of IT services. Moreover, these authors claim that the new agile approaches that promote the collaboration between the development and the operations are becoming even more important for a service oriented IT organisation.

On the other hand, Sahid et al. (2017) research aimed to create a proposal of a strategic framework to improve the ITSM processes using the agility management based on DevOps and an agility process maturity framework. This framework was measured by adopting a continuous improvement process based on DevOps and the plan-do-check-act (PDCA) Deming cycle (Legowo et al., 2019). These authors have based on three of the DevOps areas (people, processes and technology) (Teixeira et al., 2020). The authors claim that people should be the main driver on the coexistence between DevOps and ITSM processes, since they should immersive on the ITSM processes and they should know what and how the technology should be employed. It is important to keep the continuous improvement posture. IT teams should keep focused on the success of the organisation and not only on their team success. Moreover, this posture should not only be applied on the IT teams, since not only IT people participate in the ITSM processes. After the implementation of the new framework, the authors observed some benefits such as: improved service and performance support, reduced costs and business risks, improved user satisfaction and improved visibility of operations through reports and dashboards. In Hüttermann (2012), predicted this visibility of operations when extending the agile from development and business to development, business and operations.

Waschke (2015) also uses the same approach the continual improvement, like the previous authors. This author describes that practitioners usually act on a waterfall perspective. For example, the developers should only act when a service is ready for the transaction phase and once they finish the tests and the service is delivered in production, they move on to the next service and it will be taken over by other team. However, continual improvement is used to identify improvement chances and change what is needed to be changed. The author believes that all the ITIL phases should work on a cyclic way and responsible of each ITIL phase, should be involved on the other phases as well. This means that operations will participate in design, code and testing, while the developers will also experience the operations challenges, decreasing the blame game that Hussaini (2015) describes. This author also mentions the importance of tooling, because without the right tools it will not be possible to have automated build and testing.

Last but not least, Kaiser (2018) recognises that ITIL is a rigid, bulky and strictly sequential framework that should be seen as a set of guidelines to be followed in order to develop and operate IT services. The main challenge of the implementation of DevOps and agile in an ITSM environment is due to this rigidity and sequential mindset. As previously stated, DevOps and agile require iterations instead of sequential handovers. However, the fluidity offered by DevOps and agile it is not a conflict for ITSM, since the way to build a service is not changed, but essentially it is needed to identify the pieces of the framework and processes that should be included in the DevOps processes, without changing the existing service management. To sum up, it is needed to recognise what DevOps can offer and how to fit it on the ITSM framework, in order to optimise this framework to its best. To support this, Kim's et al. (2016) said "even releasing 10,000+ deployments/day requires processes, but what goes against the DevOps objectives are the approvals" which suggest some adaptation to the change management process like one can see in Section 4.2.2.

Figure 5 Conceptual map ITSM and DevOps (see online version for colours)



To synthesise all the previous findings and statements from several authors, we have created a conceptual map with the different presented concepts and ideas (Figure 5). The concepts highlighted in orange are concepts mainly related with ITSM, while the concepts highlighted as blue are concepts related with DevOps.

Table 6 Concepts relation with authors

Concept	*Reference*
Agility	Sahid et al. (2017) and Pilorget and Schell (2018)
Blame-game	Hussaini (2015) and Waschke (2015)
Business	Hüttermann (2012)
Challenges	Keller (2017)
Collaborate	Pilorget and Schell (2018)
Costs	Sahid et al. (2017)
Dev challenges	Waschke (2015)
DevOps	Hüttermann (2012), Forsgren and Humble (2016), Keller (2017), Sahid et al. (2017), Kaiser (2018) and Pilorget and Schell (2018)
IT development	Hüttermann (2012)
IT services	Kaiser (2018) and Pilorget and Schell (2018)
ITSM	Hüttermann (2012), Forsgren and Humble (2016), Keller (2017), Sahid et al. (2017), Kaiser (2018) and Pilorget and Schell (2018)
Market	Forsgren and Humble (2016)
Operations	Hüttermann (2012)
Ops challenges	Waschke (2015)
Organisations	Forsgren and Humble (2016), Sahid et al. (2017) and Pilorget and Schell (2018)
People	Sahid et al. (2017)
Performance	Forsgren and Humble (2016) and Sahid et al. (2017)
Processes	Sahid et al. (2017)
Quality	Forsgren and Humble (2016) and Keller (2017)
Rigid	Keller (2017) and Kaiser (2018)
Risks	Sahid et al. (2017)
Robustness	Pilorget and Schell (2018)
Stability	Hüttermann (2012), Forsgren and Humble (2016), Keller (2017) and Sahid et al. (2017)
Teams	Hüttermann (2012), Waschke (2015) and Sahid et al. (2017)
Technology	Sahid et al. (2017)
User satisfaction	Sahid et al. (2017)
Users	Sahid et al. (2017)

Looking at Figure 5, one can see the main ideas of the authors from the previous section can be interrelated. This also demonstrates that ITSM and DevOps can co-exist on the same environment, showing that these two concepts are not a replacement of each other, but they complement each other. Plus, it is also possible to see a flow and how each concept relates with each other, telling a story. Organisations have different types of clients. Their users build their idea of the company based on their satisfaction. Organisations use ITSM to manage IT services and their performance. However, ITSM is seen a rigid discipline and, since organisations operate on the competitive market, they need agility to answer to their customer needs. To provide such IT services, organisations have a set of teams. Business, IT operations to support IT services and IT development to develop IT services are among them. These teams need to collaborate. This collaboration brings challenges, and many times lead to the blame game. That is when DevOps comes to tackle. DevOps is concerned with the agility of the IT and it is focused on people, processes and technology. It aims to get the best of each of the above concepts, to bring more quality, stability, and robustness to the IT services. The results for improve quality, stability, and robustness improves the user satisfaction and performance, and to decrease costs and risks to the business. Therefore, it is possible to argue that DevOps can bring improvements to the challenges that IT and business teams face on their work during IT services provision. In Table 6 list the main concepts (Figure 5) and respective references that support them.

4.2.2 Change management

Change management (ChM) process should control the entire lifecycle of all changes, enabling beneficial changes to be made and to minimise the disruption of IT services (Rance, 2011).

Since the objective of DevOps is to reduce the time to go to production with changes, Keller (2017) suggests to have more pre-approved changes to facilitate the change creation and management.

Cuppett (2016) provides an example on how DevOps and ChM was used to tackle a performance issue introduced by a change in a production environment. He argues that DevOps only speeds a change delivery after an extensive testing phase, to do not allow defects to move on production. It is important to shift-left the ChM process to a prior milestone of the service development. By doing it, it will be possible to access the impact of the change earlier, thus reducing he change impact when deployed into production.

For Ravichandran et al. (2016), the ChM process proposed in ITIL is seen as very inflexible from a DevOps perspective. Therefore, is suggested that the change advisory board (CAB) should review which DevOps changes could avoid the normal change process and may be considered as standard change, which is related with Keller's (2017) suggestion.

Another author, Kaiser's (2018), argues that the ChM process is rigid and inflexible, and with DevOps should be considered as steering process instead of a governance process. However, there is something interesting to analyse. If one of the premises of DevOps is to include smaller batches of code in changes, to have more changes in production for a continuous growing of applications and services, it is assumed that the impact of these changes is minimal, due to the size of the change. Since the objective of the ChM is to assess the change impact, is this necessary? According to Kaiser (2018) ChM has been proven to be effective against malicious changes, so it should be necessary. Thus, due to this necessity, the author, adapted this process to fit on the continuous delivery and continuous deployment DevOps processes.

To adapt ChM to continuous delivery, Kaiser (2018) suggests that the ChM process initiates when the request for change is also initiated. Therefore, the ChM team will analyse the change way before developers start their build and testing phases. The author is shifting the ChM to an early stage of the SDLC, like Cuppett's (2016) suggests. During this analysis of the ChM team a CAB meeting is convoked where the change requester should attend to explain the reason for that change and provide further clarifications. Once this is clarified, the ChM gives the authorisation to build and test and the developers start their work. Once the developers finish their build and tests, the change requester inform the ChM team that is ready for production, after this, the ChM authorise the implementation of the change. During this phase, the ChM needs to perform some checks agreed scope from the request for change is fulfilled and if the timelines for the deployment is agreed by the stakeholders. Once the implementation is done the deployment and verification of the change is done. Regarding the adaptation of ChM to continuous deployment, is quite similar continuous delivery. As per Kaiser (2018) the continuous delivery require a manual trigger in order to implement a change into production, however, in continuous deployment, the code package is deployed once it satisfies the testing requirements, so the difference between the adaptation between continuous delivery and deployment adaptation is that the CAB will provide the authorisation to implement the change before the developers start their code, since that once they check in their code and test, it will be deployed to the next environment.

4.2.3 Release management (RM)

This process is responsible for planning, scheduling, controlling the build, tests and deployment of releases (Rance, 2011).

Keller (2017) mentions that the initial DevOps enthusiasts say that ChM and RM should disappear with DevOps. However, Kaiser (2018), presents an adaptation of RM, to the DevOps era. RM is known as a sequential process, however, to fit on an agile perspective needs to follow the same iteration approach. This process is divided in four tasks (release and deployment planning, build and test, deployment and review and close) where they run on a sequential manner. Since the SDLC has been changing to an iterative way, this needs to also be changed otherwise it would not fit. DevOps brought two different concepts (continuous deployment and continuous delivery). By the continuous deployment concept, if a package is tested successfully, it is ready to be deployed in an upper environment. The RM role in here is to guarantee that the path of this package deployment is stable. Based on this, the RM will spend more time on the last two tasks rather than the two first, since a lot of deployments should be occurring and for a lot of environments at the same time, so a lot of coordination is needed from the RM.

This enables the continuous delivery. When all the release packages are ready to be deployed in production, this deployment happens at will be followed by the review of the release.

4.2.4 Incident management (IM)

IM is the process responsible for restoring unexpectedly degraded or disrupted service to users as quickly as possible (Steinberg, 2011).

One of the key issues from IM is the user interface between users and service desk, where a form is usually presented to the user, while a simple chat between the SD operator and the user is the necessary to get the information for the incident creation. Keller (2017) suggests the implementation of cognitive systems and chatbots to perform these functions. This is another suggestion based on the automation practice of DevOps.

Kaiser (2018) defends that if the IM process is implemented on the right way with DevOps, the remaining components will also fit on this culture. The IM is the basis of the implementation of ITSM. However, it is needed to implement a concept of single that is responsible for both development and maintenance activities, to work with DevOps. It is interesting to see the perspective of this author. Due to the implementation of the single team, all the agile ceremonies will include incidents and problems (for operators) and user stories (for developers). This brings an adaption of the IM process where the incidents need to be reported or analysed according to the definition of ready (DOR) so the product backlog of the DevOps team can be filled in. The role of the incident manager is crucial in this part, since that he is the one that will analyse the incident and accept it on the DevOps queue. Once this is performed, the incident manager informs the scrum master and product owner of that team to prioritise it. The person with the scrum master role is responsible to be sure that the people from the DevOps team can complete their tasks within the sprint, while the product owner, who has the overall view of the product or application where the DevOps develop and operate, is responsible to prioritise tasks upon the pains from the business. Together, these two people, should define a priority and discuss the possible impact that may cause on the ongoing sprint. After this prioritisation and addition to a sprint, the developer picks the incident from a sprint and check-in the related code and using the continuous integration, continuous testing and automatic deployment, the fix for the incident gets into production and the incident gets fixed and closed.

4.2.5 Problem management (PM)

PM is the process responsible to fix the root cause of one or more incidents. So, this process can be seen as one of the most important processes to ensure the service stability (Kaiser, 2018). Thus, it is possible to conclude that both PM and IM are related. The objective of IM is to restore the normal service operation, and this is usually done by workarounds, so the user can proceed with their daily business activities. If workarounds are being applied and no one corrects the root cause, incidents will continue to be registered. Thus, the reason of the existence of PM is to fix the root causes of the incidents so the registration of incidents can be decreased. Likewise IM, Kaiser (2018) proposes a new PM agile oriented approach. A problem is created, detected by the problem manager, or created automatically by some background process, and the problem manager needs to inform the scrum master of the DevOps team. The scrum master creates necessary work items so the product owner can prioritise. Once the problem gets prioritised it will be added to a sprint backlog by the scrum master, after this, the developer will analyse and identify the root cause, develop the permanent fix and check-in his code. Then the DevOps flow of continuous integration, continuous test and automatic deployment will be in action.

4.2.6 Event management (EM)

This process is responsible to catch any change of state that as a significant impact on the management of any configuration item or IT service (Steinberg, 2011).

Keller defends that EM is the most advanced process in terms of automation due to the previous successes in pattern detection and event correlation. Due to advances in machine learning it is possible to automate the troubleshooting of the most frequent events which may cause service disruptions. To keep traceability of the actions performed by the automation, Keller also suggests to open incidents or changes, to keep track of everything.

In this process, the authors did not mention other DevOps practices besides a generic automation.

4.2.7 Service request and service catalogue management

By extending the suggestion of Keller (2017) about the cognitive systems in Section 4.2.4, the users can request services just by using text messages, where these requests can be added automatically to the system and to the service catalogue.

5 Research findings

From Section 4.2.2 until Section 4.2.7 the authors have analysed what DevOps practices were used in each ITSM process, according to the literature. To synthesise this information the authors have built Table 7.

Table 7 DevOps practices and ITSM processes

Process/practice	*Automated deployment*	*Continuous integration*	*Automated tests*	*Shift left*	*Total of matches*
RM					1
IM					*3*
PM					*3*
ChM					1
Total	*3*	2	2	1	

In Table 7, did not consider EM and SRSCM since the authors did not describe a practice. In terms of DevOps practices only four out of 12 practices identified in Table 2 were referred in literature. While in terms of ITSM processes only four out of 26 ITIL processes (Rance, 2011) where referenced. From a statistical point of view only 33% of the DevOps practices are related with 15% of the ITIL processes, which sounds short. However, looking from a practitioner point of view, ITIL is divided into five publications and it is possible to see in that processes described in Table 7, only two of these publications are represented, service operations (IM and PM) and service transition (ChM, RM) which contains a total of 12 processes. In this case, the four practices have a match with four of the 12 processes, thus 33% of the ITIL processes.

Yet from Table 7, it is possible to note that the automated deployment practice tend to be more used across processes, by being applied in three of the four processes (RM, IM, and PM), while IM and PM are the processes where more practices are applied (automated deployment, continuous integration and automated tests).

Regarding ChM process it is only matched with shift-left. For this relationship, the authors have suggested an adaption of the process to shift-left this process to an early stage of the SDLC. It was interesting to see that the same author has suggested adaptions of the ChM, IM, and PM, while the remaining authors have only pointed how DevOps could impact that processes by itself.

6 Conclusions

The authors have chosen to perform a SLR due to the rigor that this research methodology brings to any literature review. Plus, since few studies exist about the union of these two domains, a SLR is important to consolidate what is known and define future directions. During the protocol definition, the authors tried to keep a broad search string to compensate the short research universe in such a recent topic. After applying the inclusion and exclusion criteria, only 317 articles were considered, and after applying the filters defined on the protocol, including a manual analysis, only nine articles were selected. This leads us to conclude that the relationship between DevOps practices and ITSM processes is on a very early stages, which shows that some future work can be done regarding this subject.

Most of the articles regarding this subject were written in 2017, while the first article is from 2012. This shows that the interest in this subject has been growing over the time. However, in most of the articles, it was possible to find a relationship between a practice and an ITSM process, but some articles just describe, on a generic point of view, how the implementation of DevOps can improve ITSM. A conceptual map was designed based on the main concepts and keywords. This conceptual map is a data representation method that helps the reader to easily understand what the authors want to demonstrate, by showing how the concepts relate.

Nevertheless, there was possible to identify some ITSM processes where the implementation of DevOps practices could be applied, such as RM, IM, PM, and ChM. The practices pointed as useful were automated deployment, continuous integration, automated tests and shift-left. The ITSM processes that have matched with more DevOps practices are IM and PM which matches with automated deployment, continuous integration and automated tests.

Grounded on this research findings, it is feasible to argue that the match between DevOps practices and ITSM processes exist and is promising. Since DevOps embrace the relation between SD and operations, it is expected to observe an increase match in service transaction (RM and ChM) and service operations (IM and PM) processes. However, the relationship between DevOps and ITSM it is on a very early stage of investigation. More relations may be discovered in the future.

To conclude, the relationship between DevOps and ITSM exists and its evidence point to be beneficial. However, it still in very early stage and may be thoroughly explored to raise new knowledge for practitioners and academics. The existent DevOps and ITSM relations are listed and detailed in this research.

If it is true that the synergy of these domains in its very early stage, it is also true that further research is necessary. Due to the small number of matches between the DevOps practices and ITSM processes, other ITSM processes should be investigated in the future. Due to the strong component of planning and collaboration from the DevOps practices, it may be interesting to explore other ITIL publications that service transaction and service operations. Plus, future researches must focus on the analysis of the adaptation of the existing ITIL processes and not only how the DevOps practices can fit in these processes.

Moreover, it is known that IT implementations may vary with several factors. Therefore, the authors also suggest that future researchers consider possible contingency factors (Pereira and da Silva, 2012) and elicit conclusions regarding the influence of type of industry, size, strategy, among others in the adoption of DevOps in ITSM teams.

It would be very interesting to analyse what benefits would be found on each environment and on each industry and compare where more benefits could be found.

This research was subjected to some limitations that may threat the validity of results. To test the validity of the research, a discussion is made regarding four different categories, construct validity, internal validity, external validity and conclusion validity (Zhou et al., 2016). Most of the pitfalls for construct validity category are related with the initial protocol of the SLR. Some of the most common threats found by Zhou et al. (2016) are 'inappropriate or incomplete search terms in automatic search', 'incomprehensible venues or database' and 'incorrect search method'. For this research, the authors specified all steps performed, detailing the entire protocol and all the decisions taken during the research. Also, the databases used for this research were some of the most used databases used by the scientific community for software engineering. All the authors have validated and reviewed the protocol. Internal validity tries to remove the bias on selection and data extraction. To avoid this, the authors have implemented filters and inclusion and exclusion criteria that every researcher can also apply, this way, all the researchers that apply these filters and inclusion and exclusion criteria can reach to the same result of articles. However, due to the small sample of articles the authors were not able to implement quality filters for this research, for example, to only select articles from journals with Q1 rank from Qualis. But this small sample considered in this research shows that the subject is in early stages of investigation, which shows the importance of this study. Regarding external validity there is a limitation where the research occurred within a timespan and future work made other researchers can impact the results of this research. For conclusion validity, Zhou et al. (2016) alert for the 'subjective interpretation about the extracted data', the authors of this research have knowledge about the main topics of this research, both ITSM and DevOps, so to avoid any subjective interpretation of the studies found in this SLR, the authors performed an extensive analysis to gather as much data so it was possible to reach to specific conclusions over subjective conclusions.

References

Aguiar, J. et al. (2018) 'An overlapness incident management maturity model for multi-framework assessment (ITIL, COBIT, CMMI-SVC)', *Interdisciplinary Journal of Information, Knowledge and Management*, Vol. 13, pp.137–163, DOI: https://doi.org/10.28945/4083.

Alsolamy, A.A., Khan, U.A. and Khan, P.M. (2014) 'IT-business alignment strategy for business growth', *in 2014 International Conference on Computing for Sustainable Global Development, INDIACom 2014*, pp.364–366, DOI: 10.1109/IndiaCom.2014.6828160.

Barcelo-Valenzuela, M. and Leal-Pompa, C.M. (2020) 'An ITSM framework adaptation: case study in an electoral institution', pp.468–473, DOI: 10.1109/cds49703.2020.00098.

Beck, K. et al. (2001) *Agile Manifesto* [online] https://agilemanifesto.org/ (accessed 9 January 2020).

Benitez, J. et al. (2018) 'Evolution of the impact of e-business technology on operational competence and firm profitability: a panel data investigation', *Information and Management*, Vol. 55, No. 1, pp.120–130, Elsevier, DOI: 10.1016/j.im.2017.08.002.

Cannon, D. and Wheeldon, D. (2007) *ITIL Version 3 Service Operation*, 1st ed., Office of Government Commerce, The Stationery Office, London, DOI: 10.1016/j.im.2003.02.002.

Cuppett, M.S. (2016) *DevOps, DBAs, and DBaaS*, pp.73–85, DOI: 10.1007/978-1-4842-2208-9.

e Abreu, F.B. et al. (2010) 'Definition and validation of metrics for ITSM process models', *Proceedings – 7th International Conference on the Quality of Information and Communications Technology, QUATIC 2010*, pp.79–88, DOI: 10.1109/QUATIC.2010.13.

El Yamami, A. et al. (2019) 'Introducing ITIL framework in small enterprises: tailoring ITSM practices to the size of company', *International Journal of Information Technologies and Systems Approach*, Vol. 12, No. 1, pp.1–19, DOI: 10.4018/IJITSA.2019010101.

Forsgren, N. and Humble, J. (2016) 'DevOps: profiles in ITSM performance and contributing factors', in *Proceedings of the Western Decision Sciences Institute*, Las Vegas, p.25, DOI: 10.1017/CBO9781107415324.004.

Guerriero, M. et al. (2016) 'A model-driven DevOps framework for QoS-aware cloud applications', in *Proceedings – 17th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, SYNASC 2015*, pp.345–351, DOI: 10.1109/SYNASC.2015.60.

Hochstein, A., Tamm, G. and Brenner, W. (2005) 'Service oriented IT management : benefit, cost and success factors', *ECIS*, Paper 98, p.12.

Hussaini, S.W. (2015) 'A systemic approach to re-inforce development and operations functions in delivering an organizational program', in *Complex Adaptive Systems*, pp.261–266, Elsevier Masson SAS, DOI: 10.1016/j.procs.2015.09.209.

Hüttermann, M. (2012) *DevOps for Developers*, 1st ed., Apress, DOI: 10.1007/978-1-4302-4570-4.

Jabbari, R. et al. (2016) 'What is DevOps? A systematic mapping study on definitions and practices', in *Proceedings of the Scientific Workshop Proceedings of XP2016 on – XP '16 Workshops*, pp.1–11, DOI: 10.1145/2962695.2962707.

Kaiser, A.K. (2018) *Reinventing ITIL® in the Age of DevOps*, Apress, DOI: 10.1007/978-1-4842-3976-6.

Keller, A. (2017) 'Challenges and directions in service management automation', *Journal of Network and Systems Management*, Vol. 25, No. 4, pp.884–901, Springer US, DOI: 10.1007/s10922-017-9437-9.

Kettunen, P. (2009) 'Adopting key lessons from agile manufacturing to agile software product development – a comparative study', *Technovation*, Vol. 29, Nos. 6–7, pp.408–422, DOI: 10.1016/j.technovation.2008.10.003.

Kim, G. et al. (2016) *The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations*, IT Revolution Press.

Kitchenham, B. (2004) *Procedures for Performing Systematic Reviews*, Keele, p.33.

Legowo, M.B., Indiarto, B. and Prayitno, D. (2019) 'Agile software methodology with scrum for developing quality assurance system', *Proceedings – 2019 2nd International Conference of Computer and Informatics Engineering: Artificial Intelligence Roles in Industrial Revolution 4.0, IC2IE 2019*, pp.104–109, DOI: 10.1109/IC2IE47452.2019.8940831.

Lucio-Nieto, T. and Colomo-Palacios, R. (2012) 'ITIL and the creation of a service management office (SMO): a new challenge for IT professionals: an exploratory study of Latin American companies', *7th Iberian Conference on Information Systems and Technologies (CISTI 2012)*, pp.1–6.

Luo, A., Fu, J. and Liu, J. (2017) 'An impact analysis method of business processes evolution in enterprise architecture', in *PIC 2016 – Proceedings of the 2016 IEEE International Conference on Progress in Informatics and Computing*, pp.733–737, DOI: 10.1109/PIC.2016.7949595.

Lwakature, L.E. (2017) 'DevOps adoption and implementation in software development practice: concept, practices, benefits and challenges', in *OULU 2017 a 702 Universitatis Ouluensisuniversitatis Ouluensis a Scientiae Rerum Scientiae Rerum Naturalium Naturalium Acta Lucy Ellen Lwakatare ACTA DevOps Adoption and Implementation in Software Development Practice Concept, Practices, Benefits and Chal*, p.99, DOI: ISBN 978-952-62-1711-6.

Office Government Commerce (2007) *The Official Introduction to the ITIL Service Lifecycle, Online*, The Stationery Office (TSO), London.

Okoli, C. (2015) 'A guide to conducting a standalone systematic literature review', *Communications of the Association for Information Systems*, Vol. 37, No. 1, pp.879–910, DOI: 10.17705/1cais.03743.

Pereira, R. and da Silva, M.M. (2012) 'Designing a new integrated IT governance and IT management framework based on both scientific and practitioner viewpoint', *International Journal of Enterprise Information Systems (IJEIS)*, Vol. 8, No. 4, p.43, DOI: doi:10.4018/jeis.2012100101.

Pereira, R. and Serrano, J. (2020) 'A review of methods used on IT maturity models development: a systematic literature review and a critical analysis', *Journal of Information Technology*, Vol. 35, No. 2, DOI: 10.1177/0268396219886874.

Pilorget, L. and Schell, T. (2018) *IT Management: the Art of Managing IT Based on a Solid Framework Leveraging the Company's Political Ecosystem*, pp.1–194, Springer Vieweg, Wiesbaden, DOI: 10.1007/978-3-658-19309-6.

Rance, S. (2011) *ITIL 2011 Service Transition*, in Hanna, A. (Ed.): 1st ed., The Stationery Office, Norwich.

Ravichandran, A., Taylor, K. and Waterhouse, P. (2016) *DevOps for Digital Leaders*, DOI: 10.1007/978-1-4842-1842-6.

Rolf Consulting (2020) [online] https://www.rolf-consulting.com/blog/waterfall-vs-agile-vs-devops/ (accessed 9 May 2020).

Sahid, A., Maleh, Y. and Belaissaoui, M. (2017) 'An agile framework for ITS management in organizations. A case study based on DevOps', in *ICCWCS'17 : Proceeding of the 2nd Edition of the International Conference on Computing and Wireless Communication Systems*, Larache, Morocco, p.8, DOI: 10.1145/3167486.3167556.

Sharifi, M. et al. (2008) 'Lessons learned in ITIL implementation failure', in *Proceedings – International Symposium on Information Technology 2008, ITSim*, pp.8–11, DOI: 10.1109/ITSIM.2008.4631627.

Silva, M., Faustino, J., Pereira, R. and Mira Da Silva, M. (2018) 'Productivity gains of DevOps adoption in an IT team: a case study', in Andersson, B., Johansson, B., Carlsson, S., Barry, C., Lang, M., Linger, H. and Schneider, C. (Eds.): *Designing Digitalization (ISD2018 Proceedings)*, Lund University, Lund, Sweden, ISBN: 978-91-7753-876-9.

Steinberg, R. (2011) *ITIL 2011 Service Operation*, in Hanna, A. (Ed.), 1st ed., The Stationery Office, Norwich.

Teixeira, D. et al. (2020) 'A systematic literature review on DevOps capabilities and areas', *International Journal of Human Capital and Information Technology Professionals (IJHCITP)*, p.22, DOI: 10.4018/IJHCITP.2020070101.

Waschke, M. (2015) *How Clouds Hold IT Together*, Apress, DOI: 10.1007/978-1-4302-6167-4.

Zhou, X. et al. (2016) 'A map of threats to validity of systematic literature reviews in software engineering', in *Proceedings – Asia-Pacific Software Engineering Conference, APSEC*, pp.153–160, DOI: 10.1109/APSEC.2016.031.