INSTITUTO
UNIVERSITÁRIO
DE LISBOA

Exploring the Integration of Artificial Intelligence and DevOps for Agile Product Development

Afonso Costa Brandão de Figueiredo

Master in,
Computer Science and Business Management

Supervisors:
PhD Rúben Filipe de Sousa Pereira,
Assistant Professor, ISCTE-IUL

Miguel Ângelo Rodrigues da Silva, Invited Senior Assistant, Iscte-IUL

September, 2024

TECNOLOGIAS
E ARQUITETURA

Department of Science and Information Technology

Exploring the Integration of Artificial Intelligence and DevOps for Agile Product Development

Afonso Costa Brandão de Figueiredo

Master in,
Computer Science and Business Management

Supervisors:
PhD Rúben Filipe de Sousa Pereira,
Assistant Professor, ISCTE-IUL

Miguel Ângelo Rodrigues da Silva, Invited Senior Assistant, Iscte-IUL

September, 2024

Acknowledgment

A heartfelt thank you goes out to Prof. Ruben Filipe Pereira and Prof. Miguel Rodrigues Silva, whose steadfast guidance, insights, and encouragement have been truly invaluable on this journey. Their willingness to share knowledge and provide support made this research feasible, and I am genuinely thankful for that.

I also want to extend my warmest appreciation to my parents, Adelaide and José, my brothers, Tomás, Jorge, and João, and my partner, Miriam, for their constant support and understanding. Their love and encouragement have been crucial in helping me complete this work.

Finally, I would like to express my sincere thanks to everyone who has played a part in this project, including friends, colleagues, and interview participants, for their ongoing support and for being there through thick and thin. To all of you – thank you!

# Abstract

The increasing demand for efficiency in software development led to growing interest in integrating Artificial Intelligence (AI) with DevOps capabilities. While DevOps accelerates development cycles, improves collaboration, and ensures continuous delivery, the potential of AI to enhance these processes remains underexplored, especially in Agile Product Development. This research investigates the integration of AI within DevOps, addressing gaps in the current understanding, such as the absence of standardized frameworks and tailored AI solutions.

A systematic literature review (SLR) and interviews with DevOps professionals were conducted to assess AI's integration into DevOps practices. The study focused on cultural, measurement, process, and technical areas to identify where AI can be most effective in the DevOps capabilities. Findings reveal that AI integration offers several benefits, including reducing manual tasks, improving performance, providing operational support, and enhancing code quality and development practices.

However, significant challenges were identified, such as managing expectations to avoid over-reliance on AI, resistance to change, skepticism towards AI, and the increased initial effort and time required for training. Looking to the future, interviewees foresee AI becoming an autonomous or permanent tool, complementing human efforts rather than replacing them, gradually becoming a norm in DevOps with phased adoption and integration.

This study contributes to the ongoing discussion about AI's role in DevOps, offering valuable insights into both the practical and technical aspects of this integration.

Keywords:

Artificial Intelligence, AI, DevOps, AIOps, Systematic Literature Review, Field Interviews

# Resumo

A procura crescente por eficiência no desenvolvimento de software impulsionou o interesse na integração da Inteligência Artificial (IA) com as capacidades do DevOps. Embora o DevOps acelere os ciclos de desenvolvimento, melhore a colaboração e garanta a entrega contínua, o potencial da IA para aprimorar estes processos ainda é pouco explorado, particularmente no Desenvolvimento Ágil de Produtos. Esta pesquisa procura preencher essa lacuna, abordando questões como a falta de frameworks padronizados e soluções de IA adequadas para DevOps.

Foi realizada uma revisão sistemática da literatura (SLR) e entrevistas com profissionais que trabalham em DevOps para avaliar a integração da IA nas práticas DevOps. O estudo focou-se nas áreas culturais, de medição, processos e técnicas, visando identificar onde a IA pode ser mais eficaz. Os resultados indicam que a IA oferece benefícios significativos, como a redução de tarefas manuais, melhoria de desempenho, suporte operacional e melhoria da qualidade do código e das práticas de desenvolvimento.

Contudo, foram identificados desafios importantes, como a gestão de expectativas para evitar a dependência excessiva da IA, a resistência à mudança, o ceticismo em relação à IA e o maior esforço inicial e tempo necessário para a sua formação. Os entrevistados acreditam que a IA, no futuro, se tornará uma ferramenta autónoma ou permanente, complementando o trabalho humano e gradualmente tornando-se uma norma no DevOps, através de uma adoção faseada.

Este estudo oferece contributos valiosos para a discussão sobre o papel da IA no DevOps, ao fornecer insights sobre os aspetos técnicos e práticos desta integração.

Keywords:

Inteligência artificial, IA, DevOps, AIOps, Revisão Sistemática de Literatura, Entrevistas de Dominio Específico

Index

Index of Tables

Index of Figures

Introduction

The increasing demand for efficiency and innovation in the industry has driven a deeper integration between Artificial Intelligence (AI) and Development and Operations (DevOps) practices. The pursuit of operational efficiency improvements has been fueled by impressive statistics, such as those presented by Gartner in 2019. Companies adopting DevOps deploy code 46 times more frequently, with a Change Lead Time 440 times faster and an average recovery time 170 times faster (Alessandro et al., 2021).

In response to the challenging industry of software development, DevOps has emerged as an essential approach to shorten development lifecycles, ensure continuous delivery, and promote efficient collaboration. The rapid advancement of cloud and virtualization technologies over the past decades has driven the evolution of DevOps, providing efficient automation in software production delivery. Companies adopting mature automation practices, as evidenced by the Gartner report, experience not only more frequent deployments but also remarkably faster recovery times, demonstrating the effectiveness of DevOps in improving operational efficiency (Giuliano et al., 2021).

The role of AI in the modern industrial revolution extends beyond mere data collection. AI and Machine Learning (ML) algorithms provide advanced solutions to specific challenges of Industry 4.0. A study by Aldo Dagnino, Maciej Kolomyeki and Alok Kucheria from ABB Group reveals that applications in supply chain optimization, processing IoT data, robotics, failure prediction, and anomaly detection are among the domains that can be drastically improved with the use of AI (Aldo et al., 2022). Furthermore, AI-enabled automation significantly reduces production costs, generating cost synergies throughout development and enabling the transfer of these savings to the end customer (Chandrasekar et al., 2020).

The natural convergence between AI and DevOps is evident in the emerging concept of AIOps (Artificial Intelligence-Augmented IT Operations). AIOps represents not only a technological shift but a cultural transformation where humans and AI systems collaborate to achieve efficiencies and scalability. AI in the context of DevOps is not just an aspiration but a necessity. AIOps solutions have the potential to reduce operational costs, improve efficiency on a large scale, and crucially, increase customer satisfaction, as highlighted by Gartner (Alessandro et al., 2021).

The integration of AI and DevOps holds immense potential for revolutionizing Agile Product Development, yet the current landscape is marked by challenges and gaps that necessitate exploration and resolution. While the individual realms of AI and DevOps, have demonstrated substantial benefits, their integration for Agile Product Development remains a challenge. The lack of a comprehensive understanding of how AI can seamlessly intertwine with DevOps in the context of Agile Product Development poses a substantial problem. Issues such as the absence of standardized frameworks, the need for tailored AI solutions, and the intricacies of adapting DevOps practices to accommodate AI technologies collectively contribute to the complexity of this integration. Furthermore, the potential impediments in achieving synergy between AI and DevOps, including challenges related to data management, continuous integration, and real-time decision-making, necessitate a focused inquiry. Addressing these issues is crucial for unlocking the full spectrum of benefits that the integration of AI and DevOps can offer to Agile Product Development.

Therefore, this research aims to explore how AI can provide effective support to DevOps. This exploration seeks to comprehend the way AI can be integrated to offer valuable assistance and enhance DevOps practices.

CHAPTER 1

State of the Art

1.1. Theoretical Background

AI refers to the development and use of computer systems that can perform tasks that typically require human intelligence. It involves the creation of algorithms and models that enable machines to learn from data, make decisions and solve problems. AI has the potential to revolutionize various industries and domains by automating processes, improving efficiency, and enabling advanced capabilities such as natural language processing and computer vision. (Abdallah et al., 2021) The use of AI techniques, particularly machine learning, is becoming increasingly prevalent in software development processes, including requirements gathering, monitoring, modeling, coding, and testing activities (Kenichi, 2021). AI can also be applied to address challenges in data availability, computing power, talent availability, and infrastructure, which are crucial for the effective implementation of AI-enabled interventions (Chandrasekar et al., 2020). The development of AI-augmented software engineering is expected to contribute to the accumulation, sharing, and circulation of higher quality software in society.

DevOps is a software development paradigm that combines the practices of software development (Dev) and IT operations (Ops) to shorten the system development life cycle and ensure continuous delivery of high-quality software. It aims to foster a culture of shared responsibility, seamless communication, and synergy throughout the entire software lifecycle. DevOps enables organizations to navigate the complexities of modern software requirements and facilitates more frequent and reliable software releases through automation and monitoring of all stages of software development. It involves automating and monitoring processes such as integrating, testing, releasing, deploying, and managing the infrastructure. DevOps is characterized by its focus on reducing operational costs, increasing deployment frequency, and releasing reliable products that meet business requirements. (Chandrasekar et al., 2020) (Romina et al., 2021) (Pratik, 2023).

AIOps stands for AI for IT Operations, and it refers to the use of AI and ML techniques to address DevOps challenges. It involves integrating AI/ML into the DevOps pipeline to automate operational tasks, extract insights from data, and enhance continuous deployment and operations management (Kenichi, 2021). AIOps utilizes data science and computational techniques to automate routine tasks and uses inference models to extract actionable insights from data. (Zeqi et al., 2020) (Romina et al., 2021).

1.2. Related Work

By employing a systematic literature review (SLR), which is the preferred approach for aggregating evidence in the field of "Evidence-based Software Engineering" (EBSE), one was able to identify the articles that would address the Research Question. The SLR methodology utilized in this study is established by Kitchenham, 2007, specifically relevant to the Planning, Conducting, and Reporting of the review process. In Figure 1, the methodology for conducting a systematic literature review is presented, providing a succinct summary of the various stages involved.

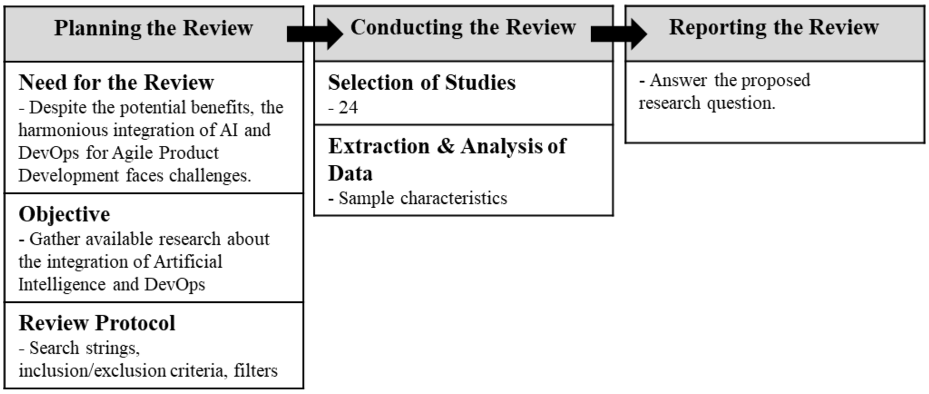

Figure 1. Phases of the SLR.

1.3. Planning the Review

This particular section embodies the initial stage of the SLR methodology and elucidates the necessity of conducting the review, the delineation of the research question, and the establishment of the review protocol.

1.3.1. Need for the Review

In response to the growing demand for efficiency and innovation, the integration of AI and DevOps practices has become crucial (Alessandro et al., 2021). DevOps significantly accelerates code deployment, change leadership, and recovery times (Alessandro et al., 2021). AI plays a pivotal role in Industry 4.0, offering solutions in supply chain optimization, IoT data processing, robotics, failure prediction, and anomaly detection (Aldo et al., 2022).

Despite the potential benefits, the integration of AI and DevOps for Agile Product Development faces challenges (Giuliano et al., 2021). The lack of a comprehensive understanding, absence of standardized frameworks, and the need for tailored AI solutions complicate this integration (Chandrasekar et al., 2020). Impediments related to data management, continuous integration, and real-time decision-making further contribute to the complexity (Alessandro et al., 2021).

1.3.2. Objective – Research Question

The primary objective of this SLR is to systematically investigate and analyze the existing body of knowledge surrounding the collaboration between AI and DevOps practices in the context of Agile Product Development, answering the Research Question: "How can AI and DevOps work together?".

1.3.3. Review Protocol

The initial stages of the review protocol involve the crucial process of determining the keywords that are pertinent to the research question at hand, as well as establishing the main databases to be utilized in the information retrieval process. Additionally, it necessitates the application of filters, which aid in refining the search results and ensuring that only the most relevant and reliable sources are included in the review.

The scientific databases that were taken into consideration for the purpose of this search are IEEE Xplore, ACM Digital Library, Scopus and Web of Science.

The primary search, which is considered the first filter, consists in searching the search string in each database applied to the articles full text.

After that, the second filter considers only the presence of the keywords determined in the search string in the articles abstract.

The third filter considers only the Journal Articles and Conference Papers

The fourth filter represents the manual removal of the duplicate studies.

The fifth and last filter chooses the possible pertinent studies by means of manual selection, which is based on the examination of the abstract.

The Table 1 summarize the Inclusion and exclusion criteria.

Table 1. Inclusion and Exclusion Criteria.

Inclusion	Exclusion
Full Text	Not in Full Text
Abstract	Not in Abstract
Journal Articles and Conference Papers	Not in Journal Articles and Conference Papers
Not Duplicates	Duplicates
Relevance regarding the integration between AI and DevOps, within the Abstract and Full text electronically available.	Non relevance regarding the integration between AI and DevOps, within the Abstract or only Abstract electronically available

The complete review protocol is illustrated in Figure 2.

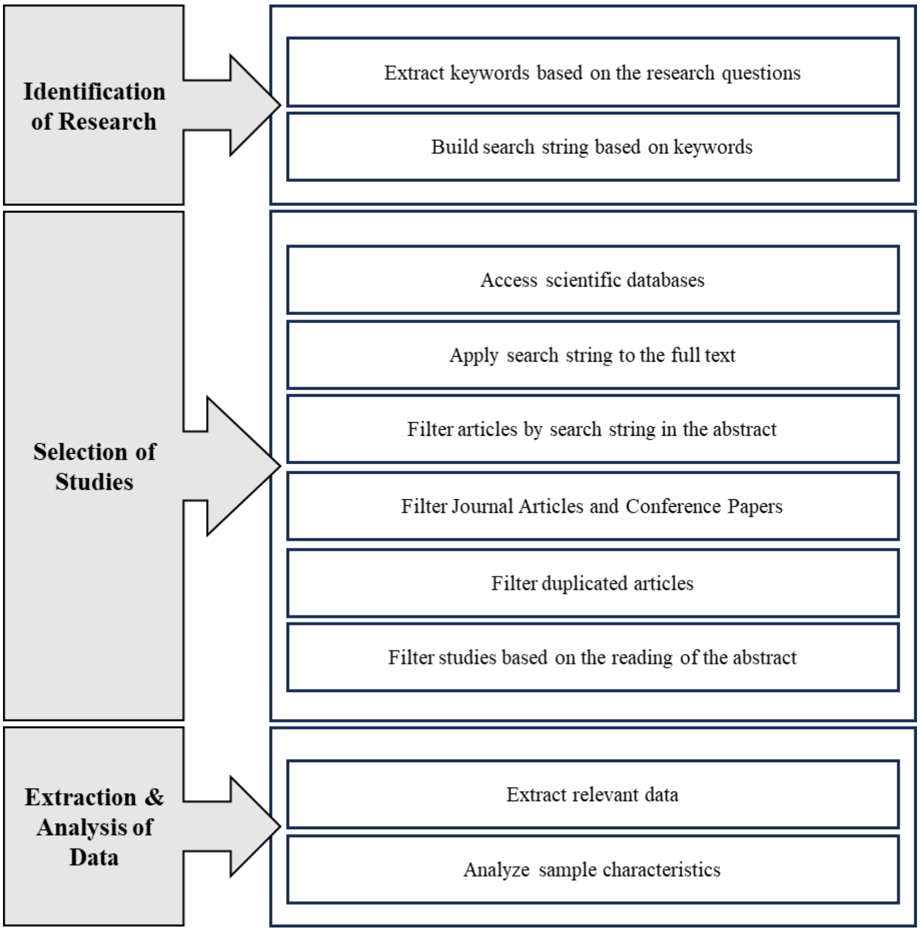

Figure 2. Review protocol.

1.4. Conducting the Review

This second stage of the SLR describes and elucidates the implementation of the review protocol and examination of the gathered data.

1.4.1. Identification of Research

Several keywords were extracted and combined to build the search string in accordance with the research question that had been previously defined in the "Objective – Research Question" section.

The selected ones are "Artificial Intelligence", "AI", and "DevOps". The term "Agile Product Development" does not qualify as a keyword due to the fact that DevOps already encompasses agile principles. By including this as a keyword, the pool of articles would be considerably reduced (Table 2).

Table 2. Representation of keywords and search string.

Keywords	Artificial Intelligence; AI; DevOps
Search String	("Artificial Intelligence" OR AI) AND DevOps

1.4.2. Selection of Studies

The preliminary examination was conducted via a search of databases. Initially, the search query was employed on the full text of the articles, customized to meet the distinct criteria of the various scientific databases' search engines. Consequently, a sum of 5,204 articles was obtained.

The second filter implemented in this research study effectively restricted the search string to exclusively include the article abstract, consequently leading to a notable reduction in the number of studies obtained, with a total of 156 studies being retrieved as a result.

By including only the Journal Articles and Conference Papers, in the third filter, the remaining total of articles reduced to 153.

The fourth filter was implemented through the process of manual identification and elimination of all the 66 duplicated articles, resulting in a remaining 87 studies.

Finally, a total of 24 possible pertinent articles were acquired for the purpose of this investigation through the careful reading of the abstracts from the 87 studies that remained.

Table 3 is the representation of the conducted studies selection process.

Table 3. Filter table.

Database	1st Filter	2nd Filter	3rd Filter	4th Filter	5th Filter
IEEE Xplore Digital Library	1 377	32	32	15	13
ACM Digital Library	415	16	16	7	2
Scopus	3 322	81	79	62	6
Web of Science	90	27	26	6	3
Total	**5 204**	**156**	**90**	**87**	**24**

1.4.3. Extraction & Analysis of Data

After the process of selecting the articles, which led to a total of 24 studies, all pertinent information was extracted and examined, being possible to evaluate the number of conference papers and journal articles (Figure 3) as well as the publication years of each one (Figure 4).

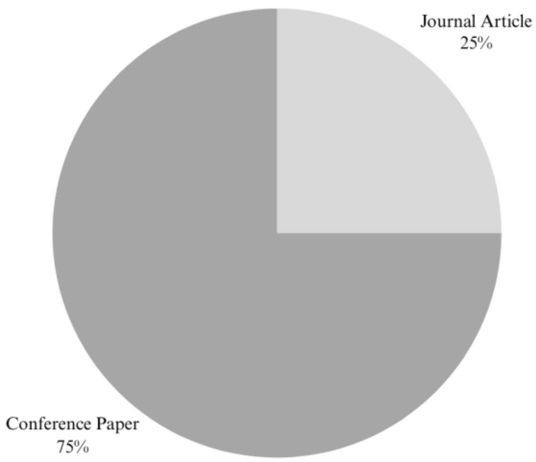

Figure 3. Distribution of selected journal and conferences articles.

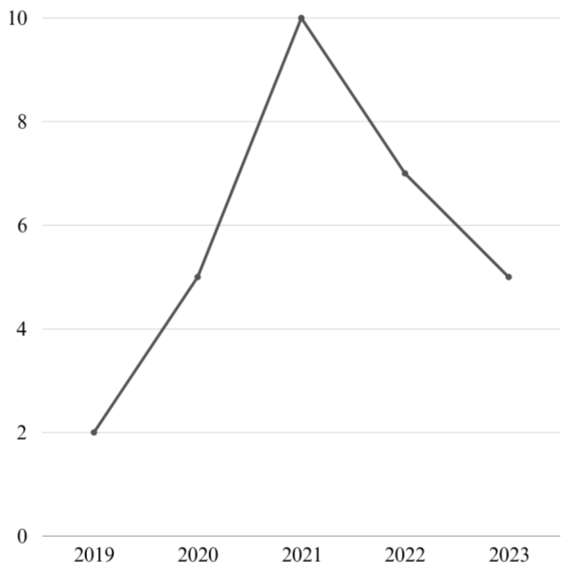

Figure 4. Distribution of selected articles by year.

As shown in Figure 3, out of the 24 studies examined, the prevailing type of studies is journal articles, comprising 75% of the articles included in the analysis.

Although the filtering process did not take into account any specific date criteria, all studies were published in 2019 or after, which demonstrates the current nature of the topic covered in this SLR. Furthermore, the progression of the quantity of relevant studies published per year shows an increase of publications in recent years, demonstrating a non-significant decrease since 2021 (it should be noted that the data for 2023 reflects studies published until October, the time at which this investigation was carried out), which denotes an increase in the attention towards the integration between AI and DevOps.

1.5. Reporting the Review

Research Question. How can AI and DevOps work together?

Expanding into the realm of real-time systems, the integration of ML components in systems has become a desirable evolution following the success of DevOps. MLOps, positioned as the combination of ML and operations, advocates for automation and monitoring at every stage of the ML system development and deployment lifecycle. This integration necessitates robust support for system integration, testing, releasing, deployment, monitoring, and infrastructure management (Tuomas et al., 2021).

In the context of AI, MLOps emerges as a framework within which the standardization and streamlining of ML lifecycle management occurs. The Requirements Engineering (RE) community's expertise in dealing with traceability and reproducibility aspects becomes crucial in this context. The integration of the RE perspective throughout the entire lifecycle is essential. MLOps pipelines can be configured to accommodate advanced engineering tools, ensuring that quality assurance (QA) measures align with the unique characteristics of each AI system. The intertwining of data and source code in ML engineering necessitates engineering tools within the pipelines that support QA measures, covering a spectrum of techniques. It is crucial to customize the MLOps pipeline for each AI system, combining different QA techniques depending on the application. The research literature (Markus, 2022) proposes diverse solutions for AI QA, including data completeness checks, formal verification of ML models, neural network adequacy testing, and simulation-based system testing. Adhering to the principles of QA, organizations are advised not to rely solely on one method but to employ a set of complementary methods to construct a convincing assurance case. A conceptual model of an MLOps pipeline, designed for an ML-based surveillance application, provides a visual representation of the pipeline's phases, encompassing data testing, model training, model testing, packaging, deployment in test environment, deployment in production environment, and continuous monitoring during operation. Feedback from development and operations teams is consolidated in dashboards, contributing to ongoing regulatory compliance assessments (Markus, 2022).

Digging into existing studies exploring the optimization of DevOps processes through ML within the DevOps pipeline reveals a nuanced approach. Customized AI platforms like Sankie, “which is an AI Platform for Azure DevOps which is a scalable and general service that is developed to assist and impact all stages of the modern software development life cycle”, have emerged, demonstrating a capacity to offer intelligent recommendations to developers and system administrators. These recommendations span a spectrum, covering training, recommendation, explanation, and evaluation (Giuliano et al., 2021).

In the context of AIDOaRt, an AI-augmented automation toolkit, it contributes significantly to various engineering activities within the DevOps cycle. These activities include requirements, monitoring, modeling, coding, and testing. The outcomes of these operations manifest as valuable feedback, insights, or actions that can be harnessed in specific phases of the DevOps cycle (Romina et al., 2021).

Artificial Intelligence for IT Operations (AIOps) harnesses the power of machine learning algorithms to execute crucial tasks, including identifying anomalies and conducting root cause analysis. This approach provides indispensable support to operations engineers, ensuring the seamless functioning of applications and actively preventing any instances of system downtime, as illustrated in the insights derived from the Edge Anomaly Detection Framework for AIOps in Cloud and IOT (Pieter et al., 2023).

The transformative impact of AI on DevOps is further elucidated through the concept of AIOps. This integration involves the application of AI to network operation and management, promising a robust framework for the deployment of AI software. Automated quality assessment becomes a cornerstone, enabling organizations to make data-driven decisions and prioritize tasks effectively. The advanced data science techniques leveraged by AI contribute not only to optimizing various steps in the DevOps cycle but also enhancing the efficiency of steps not directly managed by data scientists. The integration of AI-driven tools introduces a crucial element for tracking model quality, facilitating real-time monitoring, and providing valuable insights for ongoing adjustments and improvements. The comprehensive approach ensures the optimization of every aspect of the DevOps process, promising benefits such as improved efficiency, enhanced decision-making capabilities, and increased adaptability. As the field of AI continues to progress, the potential for further enhancements in the DevOps process appears vast, clearing the path for a future where AI and DevOps seamlessly collaborate to drive innovation and success (Lixuan et al., 2021).

The emergent concept of AIOps, is explicitly designed to address challenges highlighted by Gartner. This concept empowers software and system engineers through the efficient operation of services via ML and AI solutions. The benefits it promises, including high-quality service, increased service intelligence, elevated productivity, and reduced operating costs, position AIOps as a key paradigm in the future of DevOps. Predictions indicate a substantial adoption rate, with around 60% of companies expected to incorporate AI and ML analytics for DevOps by 2024 (Giuliano et al., 2021).

The increasing adoption of AI and ML algorithms within the industry reflects a strategic effort to enhance and oversee performance efficiently. This involves swiftly gaining insights into fundamental performance patterns across a diverse range of metrics, a capability crucial for diagnosing system complications (Giuliano et al., 2021).

Additionally, incorporating a framework that considers human factors and evaluates human capital in software DevOps, alongside AI chatbots, introduces an innovative approach to enhancing human resource allocation for improved project outcomes. This conceptual framework envisions placing the right person and AI in the right place, aligning human capabilities with AI support (Kenichi, 2021).

The concept of Intelligent DevOps represents a paradigm shift, offering DevOps management access to multiple algorithms and machine learning capabilities. The continuous learning facilitated by machine learning allows engineers to navigate the complexities of DevOps with reduced reliance on intricate alarms, essentially transforming DevOps into an intelligent, adaptive platform. The centralized management of data in complex multi-cluster environments, coupled with features like dynamic load balancing and elastic scaling of containers, further exemplifies the potential of AIOps platform (Zeqi et al., 2020).

The nuances of monitoring in contemporary DevOps environments present a considerable challenge, which AI/ML techniques are well-positioned to alleviate. The application of ML in fault diagnosis within DevOps operations, such as continuous updates, has exhibited a commendable average recall and accuracy rate of 90% (Giuliano et al., 2021), underlining the substantial potential of these technologies to significantly improve fault detection processes.

Nevertheless, the successful implementation of AI/ML in DevOps is not without its challenges. The necessity for substantial real-world datasets, the dynamic nature of system characteristics, and the intricate nature of large-scale integration all pose potential hurdles. Maintaining modularity in large-scale DevOps projects, especially with the segregation of development environments from AI models, emerges as a significant challenge. Addressing these challenges necessitates advanced research efforts and the development of effective solutions (Giuliano et al., 2021).

Within the DevOps landscape, the integration of AI/ML stands out as a significant force for streamlining processes, particularly in dynamic, data-intensive, and distributed environments. The trajectory of DevOps appears to be steering towards a future heavily influenced by the convergence of intensive AI, ML, and data, offering not only potential functionality enhancements but also promising a revolutionary shift in how developers and system administrators conceptualize, assess, deploy, and uphold systems (Giuliano et al., 2021).

CHAPTER 2

Methodology

The methodology employed in this research is structured in two crucial phases, each serving a distinct yet interconnected purpose. The initial phase involves the execution of a SLR. This systematic review is crucial as it serves as the foundation, providing a comprehensive overview of existing knowledge and insights related to the integration of AI and DevOps for Agile Product Development. The SLR aims to identify key themes, gaps, and challenges in the current landscape.

Building on the insights from the SLR, the second phase of the methodology involves field interviews. This qualitative approach adds depth to the research by engaging with people who work actively with DevOps on a daily basis. These interviews are vital for understanding real-world insights, practical challenges, and innovative solutions that might not be fully covered in existing literature.

The field interviews aim is to provide targeted outcomes aligning with the investigation's purpose. This combined methodology ensures a comprehensive and well-rounded exploration of the research topic, enhancing the overall quality and relevance of the study.

2.1. Preparing the Interviews

Following the SLR, presented on section 1.2, the next phase of the research methodology involved preparing and conducting interviews to gather qualitative data from professionals actively engaged in DevOps practices. The preparation for these interviews aimed at ensuring the selection of participants who could provide deep and practical insights into the integration of AI and DevOps in the context of Agile Product Development.

To begin with, a comprehensive interview guideline was developed, outlining the primary objectives of the interviews, which were to identify potential areas for AI integration within the DevOps capabilities identified by Ricardo Amaro in his research "Capabilities and Practices in DevOps: A Multivocal Literature Review", and to understand the practical benefits, challenges, and future prospects of such integration.

The selection of the interview participants was a crucial step in this phase. The goal was to include a diverse group of professionals who could provide a range of perspectives based on their different roles and experiences within the DevOps field. The selection process involved identifying and reaching out to individuals who were actively working with DevOps methodologies in their organization. To ensure a comprehensive understanding, it was important to include individuals from various roles within the DevOps ecosystem.

Ultimately, ten professionals were selected for the interviews. This group included three Scrum Masters, six Product Owners, and one Principal Software Engineer. The diversity in roles among the participants was intentional, as it provided a broader perspective on the integration of AI and DevOps. Scrum Masters could offer insights into team dynamics and process management, Product Owners could provide views on product development and delivery, and the Principal Software Engineer could share technical and implementation-related perspectives. As shown in Table 4 and Figure 5Figure 5, 3 out of 10 participants are SCRUM Masters, 6 out of 10 are Product Owners, and 1 out of 10 holds the position of Principal Software Engineer. All participants have at least 5 years of experience working in DevOps, with 50% possessing 10 or more years of experience. On a scale from 0 to 5, all participants rated their knowledge in DevOps at 3 or higher, with 60% rating it at 4 and 20% at 5.

Table 4. Practitioners' details.

Role/Position in the company	**Years of experience in DevOps**	**Years of experience in AI**	**How do you assess your knowledge in DevOps (0-5)**	**How do you assess your knowledge in AI (0-5)**
Product Owner	6	N/A	4	2
SCRUM Master	6	N/A	5	3
SCRUM Master	10	2	4	2
Product Owner	5	N/A	5	1
SCRUM Master	12	2	4	3
Product Owner	12	2	4	2
Principal Software Engineer	10	3	4	3
Product Owner	5	1	3	1
Product Owner	7	N/A	4	3
Product Owner	10	N/A	3	2

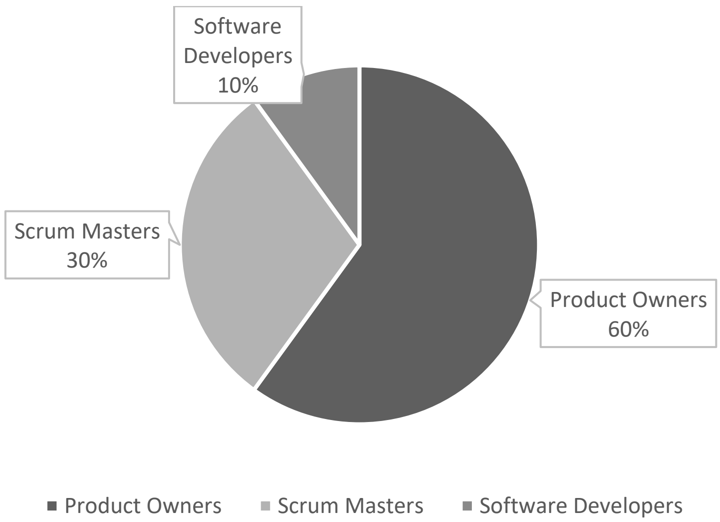

Figure 5. Role distribution among interviewees.

2.2. Conducting the Interviews

The interviews were conducted in a technology company which operates with a flat, agile organizational structure, fostering teamwork and collaboration across its diverse projects, specialized in software development for connected cars, autonomous driving, electrification, data intelligence, cloud security, sales ecosystems, financial services, logistics, and production, the company employs around 3,000 people in Portugal.

The semi-structured interviews were guided by four main questions aimed at understanding the integration of AI into the DevOps framework. These questions were: “In your opinion, considering the capabilities/practices listed, where do you see AI being useful in the context of DevOps?”, “What benefits do you identify in integrating AI with DevOps?”, “What challenges do you foresee in this integration?”, and “What are your future prospects for this integration?”. These questions were asked to practitioners to gather insights and complement the theoretical findings from the SLR phase. Each interview provided a nuanced understanding of the potential areas for AI integration, the practical benefits and challenges, and the future outlook of incorporating AI into DevOps practices.

The first interviewee focused primarily on the outcomes outlined in the provided framework (Amaro et al, 2023), which addressed the second question regarding the benefits of AI in DevOps. He emphasized AI's potential to reduce burnout by alerting the team about intense workloads based on past patterns and providing early warnings about upcoming high-demand periods. For predictable and faster releases, he noted that AI can analyze upcoming production changes more abstractly and quantitatively, reducing human error and offering "quick wins" by suggesting immediate-impact alterations, such as reducing task repetition or managing resources efficiently. In terms of fast recovery, the interviewee highlighted AI's ability to enhance predictability by having a comprehensive view of the sprint activities, enabling timely rollback suggestions if necessary.

For customer satisfaction, AI can analyze the frequency and nature of incidents, helping the team better address client needs and monitor satisfaction trends. AI can also foster a performance-oriented culture by acting as a performance guide, providing concrete data for team evaluation and suggesting performance improvements. The interviewee mentioned that AI enhances job satisfaction among engineers by offering proactive insights, reducing repetitive tasks, and providing objective, data-driven feedback, thus boosting motivation and development focus. Moreover, AI can improve security and compliance by identifying security breaches and ensuring adherence to standards.

When asked about the benefits of integrating AI with DevOps, the interviewee identified several advantages: increased team focus on critical tasks, enhanced performance, contributions to the product backlog by suggesting incident-reducing measures, and support for developers in pre-deploy validations, integrations, and updates, thereby reducing human error in complex systems.

Regarding challenges, the interviewee foresaw an initial increased effort, particularly in terms of time required to train the AI and integrate it into the team's routine. Managing team expectations was also highlighted, emphasizing that AI should be seen as a useful tool requiring training and attention, rather than a solution to all problems.

Looking ahead, the interviewee envisioned AI becoming more autonomous over time, reducing integration difficulties and effort. He anticipated AI evolving alongside DevOps into "BizDevOps" and "DevSecOps," serving as a bridge between business and DevOps teams, facilitating alignment. He referred that AI should be viewed as a complementary tool, akin to a new team member needing training.

The second interviewee expressed reservations about using AI for aspects involving human interaction, particularly excluding Cultural Capabilities, as AI lacks the "gut-feeling" necessary to understand people. However, he identified several areas where AI could be beneficial within the DevOps routine:

Emergency response/proactive failure notification(C17), AI can provide alert notifications and manage shared on-call rotations. Continuous integration(C02) AI is useful for testing software but not for building it. Continuous delivery and deployment automation(C03) AI can handle automated deployments, although human oversight is needed during the hyper-care period post-deployment. Test automation and environments(C05) AI can manage testing environments. Containerization(C18) AI can assist in creating containers. Test data management(C34) AI can handle test data effectively.

When discussing the benefits of integrating AI with DevOps, the interviewee highlighted the reduction of manual tasks, freeing up time for more complex activities such as deployments and content checks. AI's capability to provide alerts, notifications, and detect failures before human identification through pattern analysis was noted. Additionally, AI can identify common cases and suggest incident resolutions based on similar past incidents.

The interviewee foresaw challenges in adapting processes to AI, requiring translation of processes to feed into the AI system. Resistance to change, choosing the right tool, and managing expectations regarding the technology were also mentioned as potential obstacles.

Looking ahead, the interviewee asserted that AI in DevOps is not a passing trend but a permanent tool. While they didn't elaborate much on the future integration, they were clear that AI would not replace developers but rather serve as a valuable tool.

The third interviewee expressed that AI would not be suitable for Cultural Capabilities but could be particularly beneficial in several Technical Capabilities. For Continuous Integration (C02) and Continuous Delivery and Deployment Automation (C03), AI could streamline these processes effectively. They emphasized the importance of Test Automation and Environments (C05), noting that teams often neglect or delay testing, and AI could automate this process to ensure it is completed consistently. AI could also enhance Version Control System (C07) and Trust/Empower Teams to Make Decisions and Changes (C09) by providing valuable input and assessing the impact of project modifications. Additionally, AI could play a significant role in Configuration Management (C11) and Cloud Infrastructure and Cloud Native (C12). For Analyzing Stories within a Sprint (C28), AI could deduce the necessary competencies required.

The interviewee identified several benefits of integrating AI with DevOps, including support for code development by avoiding repetitive code and implementing best practices. AI could suggest changes based on target customers, provide alarm systems, and facilitate the automatic resolution of incidents.

Regarding challenges, the interviewee stressed the importance of maintaining knowledge of what is being developed, even if AI handles the development, since the DevOps team will need to operationalize the developed content. They also highlighted the necessity of managing expectations regarding AI technology to avoid assuming it is a solution for all problems and to prevent over-reliance on it.

Looking to the future, the interviewee does not foresee a complete replacement of DevOps teams by AI but suggests that up to 50% of their work might be supplemented by AI, viewing it as a complement rather than a substitute. They anticipate potential legal issues related to data protection and intellectual property, as well as financial questions about the value of human developers if AI is producing code.

The fourth interviewee sees AI being useful in various DevOps capabilities and practices. In Cross Team Collaboration and Communication (C01), AI can automate tools and enhance interactions among team members and stakeholders. Continuous Integration (C02) and Continuous Delivery and Deployment Automation (C03) can benefit significantly from AI implementation. Proactive Monitoring (C04) is another area where AI can identify system failures and reduce manual workload. AI can also be applied to Test Automation and Environments (C05), Continuous Improvement of Processes and Workflows (C06), Version Control System (C07), and Code Maintainability (C36).

In terms of benefits, the interviewee highlighted that AI could ensure smooth communication processes, maintain software and deployment quality through automated testing, and increase customer satisfaction.

Regarding challenges, they pointed out the resistance to change as a significant hurdle. It is crucial to ensure that the team's routine adapts to AI, facilitating its learning process. There are also concerns about data protection and intellectual property, as AI requires data for training. Determining how much and which data can be provided while ensuring security and intellectual property rights is a critical issue.

For future prospects, the interviewee believes that AI integration is not just a trend but is here to stay and will increasingly become a part of DevOps routines. However, they caution that AI could be dangerous if not properly managed, but they are confident that the future of DevOps will undoubtedly involve AI integration in the coming years.

The fifth interviewee sees AI being useful in multiple aspects of DevOps, encompassing cultural, measurement, process, and technical capabilities. In terms of cultural capabilities, AI can enhance Cross Team Collaboration and Communication (C01) and Open-Source Software Adoption (C19). Measurement capabilities such as Proactive Monitoring (C04), Emergency Response and Proactive Failure Notification (C17), Monitoring Systems to Inform Business Decisions (C23), and Visual Management Capabilities (C37) can significantly benefit from AI integration. For process capabilities, Continuous Improvement of Processes and Workflows (C06), Working in Small Batches (C25), Lightweight and Streamlining Change Approval (C27), Visibility of Work in the Value Stream (C28), and Data-Driven Approach for Improvements (C32) are identified as areas where AI can play a crucial role. Technical capabilities, including Continuous Integration (C02), Continuous Delivery and Deployment Automation (C03), Test Automation and Environments (C05), Version Control System (C07), Cloud Infrastructure and Cloud Native (C12), Artifacts Versioning and Registry (C13), Database Change Management and Release Alignment (C15), Infrastructure as Code (C16), Containerization (C18), Shift Left on Security (C20), Trunk-Based Development (C22), Centralized Log Management (C26), Test Data Management (C34), Chaos Engineering (C35), and Code Maintainability (C36), are also seen as benefiting from AI.

The interviewee identifies several benefits of integrating AI with DevOps. These include increased productivity, freeing team members to focus on more challenging tasks, and improved response to critical situations. AI can enable more effective system monitoring, facilitate the release and deployment process in terms of both quantity and quality, enhance the testing framework, provide more accurate and high-quality predictions, and ultimately increase customer satisfaction.

However, the interviewee also foresees challenges in this integration. One major challenge will be optimizing AI, particularly in terms of training. Managing expectations is another critical issue, as there is a risk of over-reliance on AI and creating dependency.

Looking to the future, the interviewee believes AI will become an integral part of DevOps work. They anticipate that it will be rare to find services or systems that do not utilize AI in some capacity.

The sixth interviewee focuses on the more technical and less human aspects of DevOps where AI can be beneficial. They highlight Proactive Monitoring (C04), Test Automation and Environments (C05), Version Control System (C07), Monitoring Systems to Inform Business Decisions (C23), and Test Data Management (C34) as the key areas where AI can be particularly useful.

The interviewee identifies several benefits of integrating AI with DevOps, including the reduction of manual and repetitive work and the enhancement of predictive analysis. These benefits can streamline operations and improve the efficiency and accuracy of various processes within DevOps.

However, the interviewee also foresees significant challenges in this integration. Security and data privacy are major concerns, particularly regarding the protection of core business information and deciding which project information to share. There is also a general distrust towards AI, even when it is well-trained, and resistance to change remains a hurdle.

Looking to the future, the interviewee believes that with the integration of AI, developers will have more time to focus on more complex developments. AI is seen as a complement to human effort, though it also presents challenges related to the potential replacement of jobs.

The seventh interviewee recognizes the potential of AI across a wide range of DevOps capabilities, particularly in facilitating communication and alleviating operational workloads within Cross Team Collaboration and Communication (C01) and Job Satisfaction (C33). AI can also enhance capabilities like Supporting a Learning Culture and Experimentation (C08). In terms of measurement, AI can improve Proactive Monitoring (C04), Emergency Response and Proactive Failure Notification (C17), Monitoring Systems to Inform Business Decisions (C23), and Visual Management Capabilities (C37). For process capabilities, AI can aid in Continuous Improvement of Processes and Workflows (C06) and a Data-driven Approach for Improvements (C32). Technically, AI is seen as beneficial across all technical capabilities.

The benefits of integrating AI with DevOps were numbered. AI can enhance team performance, redirect problems to the appropriate personnel, improve code quality and delivery, and provide better reporting and anomaly detection through advanced monitoring and predictive capabilities.

However, challenges were also identified, including optimizing AI models and overcoming resistance to change within teams. Training the AI model effectively can be particularly difficult, posing a significant barrier to seamless integration.

Looking to the future, the interviewee notes the current high demand for AI, but often without clear objectives. They foresee an initial stagnation in AI adoption as companies struggle to define precise goals for AI integration, followed by renewed growth once these objectives are better understood and articulated.

The eighth interviewee highlights several areas where AI can be useful in the context of DevOps. Proactive Monitoring (C04) can be significantly enhanced by AI's ability to analyze patterns and identify anomalies. Supporting a Learning Culture and Experimentation (C08) can benefit from AI by alleviating human workload, allowing the team more time for self-improvement. Emergency Response and Proactive Failure Notification (C17) can be improved through AI's analytical capabilities, which can assess the severity of issues and direct notifications to the appropriate team members. Lightweight and Streamlining Change Approval (C27) can also be facilitated by AI, which can analyze the impact of changes and make automated decisions based on this analysis.

The interviewee identifies several benefits of integrating AI with DevOps, such as saving time for more strategic thinking and evolution, as well as freeing team members from manual tasks. However, there are challenges in this integration, including difficulties in integrating AI with existing processes, the need to dedicate time to training the AI, and resistance to change within the team.

Looking towards the future, the interviewee does not anticipate a rapid growth in AI adoption. Instead, they expect AI to be introduced in very specific areas, leading to gradual rather than radical changes. They see AI as a complementary tool rather than a replacement for human roles.

The ninth interviewee provided insightful perspectives on the integration of AI in DevOps. They identified several Cultural Capabilities, including Transformational Leadership (C21), Supporting a Learning Culture and Experimentation (C08), and Job Satisfaction (C33), where AI can play a significant role. For Measurement Capabilities, they believe AI can be useful across all areas, enhancing the ability to measure and analyze various metrics. In terms of Process Capabilities, Continuous Improvement of Processes and Workflows (C06) was highlighted as a key area for AI application. For Technical Capabilities, they saw potential in Continuous Delivery and Deployment Automation (C03), Test Automation and Environments (C05), Database Change Management and Release Alignment (C15), Test Data Management (C34), Trunk-Based Development (C22), Centralized Log Management (C26), Chaos Engineering (C35), and Code Maintainability (C36).

The interviewee noted several benefits of integrating AI with DevOps, such as the automation of incident resolution, which can streamline operations and save the team time, as well as improving the quality of software delivery. However, they also foresaw challenges, including skepticism and distrust towards incident resolutions or developments handled by AI, resistance to change, and the need to manage expectations, as AI will require time to be properly trained.

Looking to the future, the interviewee predicted that in specific areas, particularly operations, there might be a shift towards AI taking on more responsibilities, potentially replacing some roles. They also anticipated that training AI algorithms would become quicker and potentially more cost-effective than training people.

The tenth interviewee provided a detailed view on the potential applications of AI in DevOps, focusing on the Measurement, Process, and Technical Capabilities, while excluding Cultural Capabilities. For Measurement Capabilities, they saw AI being useful in Proactive Monitoring (C04), Emergency Response and Proactive Failure Notification (C17), Monitoring Systems to Inform Business Decisions (C23), Work in Progress (WIP) Limits or Work in Process Limits (C29), and Visual Management Capabilities (C37). Within Process Capabilities, they highlighted Continuous Improvement of Processes and Workflows (C06), Focus on People (C10), Working in Small Batches (C25), Lightweight and Streamlining Change Approval (C27), Visibility of Work in the Value Stream (C28), Customer Focus and Feedback (C30), and a Data-Driven Approach for Improvements (C32). For Technical Capabilities, they identified Continuous Integration (CI) (C02), Continuous Delivery and Deployment Automation (C03), Test Automation and Environments (C05), Version Control System (C07), Trust and Empower Teams to Make Decisions and Changes (C09), Configuration Management (C11), Cloud Infrastructure and Cloud Native (C12), Artifacts Versioning and Registry (C13), Loosely Coupled Architecture and Microservices (C14), Database Change Management and Release Alignment (C15), Infrastructure as Code (C16), Containerization (C18), Shift Left on Security (C20), Trunk-Based Development (C22), Centralized Log Management (C26), Test Data Management (C34), Chaos Engineering (C35), and Code Maintainability (C36).

The interviewee identified several benefits of integrating AI with DevOps, such as the automation of repetitive tasks, improving code quality, increasing delivery speed, automating testing, error detection, reducing operational workload, and freeing up more time for development tasks.

However, they also foresaw challenges in this integration, including managing expectations, the learning curve for AI, building trust in the technology, ensuring the quality of information, and the usability and difficulty of integrating AI.

Looking to the future, the interviewee saw AI becoming another member of the team, with its integration being phased but ultimately leading to a structural change akin to an industrial revolution. They predicted a reduction in developer personnel, with new roles emerging for AI maintenance.

In Table 5 is presented the content extracted from each interview for each question, offering a concise summary of the overall interview process.

Table 5. Interview's conduction summary

Interviewee	AI Usefulness in DevOps Capabilities	Benefits of Integrating AI with DevOps	Challenges in AI Integration	Future Prospects for AI Integration
1	N/A	1. alerting about workloads 2.predicting releases 3. fast recovery 4. customer satisfaction 5. performance culture 6. job satisfaction 7. security and compliance 8. Increased focus on critical tasks 9. proactive insights	1. Initial increased effort 2. time for AI training 3. managing expectations.	1. AI as an autonomous tool 2. evolving into "BizDevOps" and "DevSecOps" 3. complementary to the team.
2	C02;C03;C05; C17;C18;C34	1. Reduction of manual tasks 2. alert notifications 3. pattern analysis 4. incident resolution.	1. Adapting processes to AI 2. resistance to change 3. choosing the right tool 4. managing expectations.	1. AI as a permanent tool 2. not replacing developers but complementing them.
3	C02;C03;C05; C07;C09;C11; C12;C28	1. Support for code development 2. incident resolution 3. alarm systems 4. best practices implementation.	1. Maintaining knowledge of developments 2. managing expectations 3. avoiding over-reliance on AI.	1. AI supplementing up to 50% of DevOps work 2. potential legal and financial issues.
4	C01;C02:C03: C04:C05:C06: C07:C36	1. Enhanced communication 2. software quality 3. deployment 4. customer satisfaction.	1. Resistance to change 2. adapting routines 3. data protection, intellectual property concerns.	1. AI as a stable 2. integral part of DevOps 3. requiring careful management.

Interviewee	AI Usefulness in DevOps Capabilities	Benefits of Integrating AI with DevOps	Challenges in AI Integration	Future Prospects for AI Integration
5	C01;C02;C03; C04;C05;C06; C07;C12;C13; C15;C16;C17; C18;C19;C20; C22;C23;C25; C26;C27;C28; C32;C34;C35; C36;C37	1. Increased productivity 2. effective monitoring 3. improved release and deployment 4. better predictions 5. customer satisfaction.	1. Optimizing AI 2. training 3. managing expectations 4. avoiding over-reliance.	1. AI becoming a norm in DevOps 2. rarely any services without AI use.
6	C04;C05;C07; C23;C34	1. Reduction of manual work 2. enhanced predictive analysis 3. streamlined operations.	1. Security and data privacy concerns 2. distrust towards AI 3. resistance to change.	1. Developers focusing on complex tasks 2. AI complementing human efforts.
7	C01;C02;C03; C04;C05;C06; C07;C08;C09; C11;C12;C13; C14;C15;C16; C17;C18;C20; C22;C23;C26; C32;C33;C34; C35;C36;C37	1. Enhanced performance 2. better problem resolution 3. improved code quality, 4. anomaly detection.	1. Optimizing AI models 2. overcoming resistance to change 3. training AI effectively.	1. Initial stagnation in adoption, followed by growth with clear AI objectives.
8	C04;C08;C17; C27	1. Time-saving 2. strategic thinking 3. reduction of manual tasks.	1. Integration difficulties 2. time for training AI 3. managing expectations.	1. Gradual introduction in specific areas 2. AI as a complementary tool.
9	C03;C04;C05; C06;C08;C15; C17;C21;C22; C23;C26;C29; C33;C34;C35; C36;C37	1. Automation of incident resolution 2. improved software delivery quality.	1. Skepticism towards AI 2. resistance to change 3. managing expectations.	1. AI taking on more responsibilities in operations 2. faster and cost-effective training.

Interviewee	AI Usefulness in DevOps Capabilities	Benefits of Integrating AI with DevOps	Challenges in AI Integration	Future Prospects for AI Integration
10	C02;C03;C04; C05;C06;C07; C09;C11;C12; C13;C14;C15; C16;C17;C18; C20;C22;C23; C25;C26;C27; C28;C29;C32; C34;C35;C36; C37	1. Automation of repetitive tasks 2. improved code quality 3. increased delivery speed 4. error detection 5. reduced operational workload.	1. Managing expectations 2. learning curve for AI 3. trust in technology 4. quality of information 5. integration difficulty.	1. AI as a team member 2. phased integration leading to structural changes 3. new roles for AI maintenance.

2.3. Data Saturation

In this study, data saturation is monitored to assess the state of the artifact and prepare it for the conclusion phase. Within qualitative research, saturation is widely recognized as a key methodological concept, used to determine when additional data collection or analysis becomes unnecessary based on the information already obtained and analyzed (Saunders et al., 2018).

Saturation occurs when new data adds only marginally to what has already been gathered. For instance, during interviews, data saturation is evident when the researcher consistently encounters the same responses. At this stage, it is appropriate to halt further data collection and begin evaluating the existing data (Power et al., 2015). At this stage, it is appropriate to halt further data collection and begin evaluating the existing data. (Grady, 1998).

The saturation point in this study was determined by observing a consistent pattern of repeated insights from participants and a noticeable decline in the percentage of new ideas emerging from the data, as illustrated in Table 6. Although the percentage of new contributions only dropped to zero in the eighth and final interviews, the consistent decline leading up to that point indicated a clear saturation of insights. The gradual decrease in new information, particularly the sharp drop to minimal new insights from Interview 6 onward, suggested that further data collection would not yield significant new findings. Consequently, the decision was made to conclude the data collection phase after ten interviews and transition to the next phase of the research, focusing on a detailed evaluation of the data collected.

Table 6. Total new insights per interview per question

Interview	Total Unique Insights	New Insights	Percentage	Usefulness	Benefits	Challenges	Prospects
I01	14	14	100.00%	0	9	2	3
I02	25	11	44.00%	6	2	3	0
I03	34	9	26.47%	5	1	1	2
I04	42	8	19.05%	4	2	1	1
I05	55	13	23.64%	13	0	0	0
I06	56	1	1.79%	0	0	0	1
I07	60	4	6.67%	3	0	0	1
I08	60	0	0.00%	0	0	0	0
I09	63	3	4.76%	2	0	0	2
I10	63	0	0.00%	0	0	0	0

2.4. RQ1 - Considering the capabilities listed, where do you see AI being useful in the context of DevOps?

In this section, the focus is to explore the intersection of AI and DevOps by analyzing where specialists believe AI could be most beneficial across various DevOps capabilities. To achieve this, a framework of 37 DevOps capabilities derived from the study (Amaro et al, 2023) was utilized. This framework provided a comprehensive overview of essential DevOps practices, serving as the basis for the first question in our interviews. By organizing these capabilities into cultural, measurement, process, and technical categories, it was possible to systematically identify areas where AI could enhance DevOps practices, offering valuable insights into the future of AI-driven DevOps.

2.4.1. DevOps Capabilities Enunciation

The capabilities identified in this framework span a range of critical areas within DevOps, each contributing to the overall effectiveness and efficiency of DevOps practices.

Cultural Capabilities:

C01 - Cross-team collaboration and communication: This capability emphasizes the importance of collaboration between application, operations, and security teams. By involving all relevant stakeholders and fostering a culture of trust and communication, teams can work together more effectively, enhancing overall productivity and innovation.

C08 - Support learning culture and experimentation: Encouraging a learning culture within teams allows for continuous improvement. This involves creating an environment where engineers feel safe to experiment with new ideas, update requirements, and test different approaches without fear of failure.

C19 - Open-source software adoption: The adoption of open-source software is crucial in DevOps practices. Leveraging these solutions can lead to more efficient testing and deployment processes, ultimately enhancing the agility and responsiveness of the development cycle.

C21 - Transformational leadership: Effective leadership plays a pivotal role in driving team performance and growth. Transformational leaders inspire their teams to strive for better performance, encouraging continuous improvement and adaptation to change.

C24 - Performance/Westrum organizational culture: A Westrum organizational culture focuses on fostering cooperation, trust, and information sharing among teams. This culture promotes a performance-oriented environment where teams are more likely to collaborate effectively and innovate.

C31 - Blameless postmortems/reduced fear of failure: Conducting blameless postmortems is a practice that allows teams to learn from mistakes without assigning blame. This approach reduces the fear of failure, encouraging teams to take calculated risks and innovate.

C33 - Job satisfaction: Ensuring job satisfaction among team members is essential for improving organizational performance and retention. A satisfied team is more likely to be productive, engaged, and committed to the organization's goals.

Measurement Capabilities:

C04 - Proactive monitoring: Proactive monitoring involves continuously monitoring infrastructure to identify potential issues before they impact users. Combined with autoscaling, this capability allows for automatic management of capacity, ensuring a stable and responsive system.

C17 - Emergency response/proactive failure notification: This capability focuses on improving the response to failures by providing early notifications based on monitored values. By being alerted to potential issues early, teams can respond more effectively to prevent or mitigate the impact of failures.

C23 - Monitor systems to inform business decisions: Using visual dashboards to track changes and metrics is crucial for making informed business decisions. This capability ensures that decision-makers have access to accurate and up-to-date information about system performance and trends.

C29 - Work in progress (WIP) limits: Implementing WIP limits helps teams prioritize tasks and manage workflow more efficiently. By limiting the amount of work in progress, teams can focus on completing tasks more quickly, leading to improved productivity and throughput.

C37 - Visual management capabilities: Visual management tools are used to communicate progress, manage changes, and improve overall workflow efficiency. These tools help teams stay aligned on project goals and milestones, ensuring that work is progressing as planned.

Process Capabilities:

C06 - Continuous improvement of processes and workflows: Continuous improvement involves regularly assessing and refining processes and workflows to enhance efficiency and effectiveness. This iterative approach ensures that teams are always striving for better performance and outcomes.

C10 - Focus on people: Emphasizing the importance of people in DevOps involves ensuring that processes, technology, and team members work together harmoniously toward common business goals. This human-centered approach is critical for achieving long-term success.

C25 - Working in small batches: Working in small batches allows teams to move tasks quickly through the development pipeline. This approach reduces the time between the start and completion of work, leading to faster delivery of features and improvements.

C27 - Lightweight/streamlining change approval: Streamlining the change approval process is essential for improving release frequency without compromising stability. By reducing the complexity and time required for approvals, teams can deploy changes more rapidly and with greater confidence.

C28 - Visibility of work in the value stream: Documenting and visualizing the flow of work from idea to customer outcome is crucial for driving higher performance. This visibility allows teams to identify bottlenecks, optimize processes, and ensure that work is aligned with business objectives.

C30 - Customer focus/feedback: Incorporating customer feedback into product design and roadmaps is vital for improving customer satisfaction. This capability ensures that the development process remains aligned with customer needs and expectations.

C32 - Data-driven approach for improvements: A data-driven approach involves analyzing factual data to drive performance improvements across scalability, testing, and deployment. By relying on objective metrics, teams can make informed decisions that lead to better outcomes.

Technical Capabilities:

C02 - Continuous integration (CI): Continuous integration is the practice of automating testing and building processes to produce high-quality software. This capability reduces development costs and increases team productivity by ensuring that code changes are integrated and tested frequently.

C03 - Continuous delivery and deployment automation: This capability involves automating the software release process to deploy changes quickly and frequently. By reducing deployment risk and providing faster feedback, teams can ensure that new features and updates are delivered to users more efficiently.

C05 - Test automation and environments: Using automated test environments allows teams to receive quick feedback on changes across the Software Development Life Cycle (SDLC). This capability reduces lead times and helps ensure that code is reliable and meets quality standards.

C07 - Version control system: Version control systems are used to manage and version all production artifacts, including application code, configurations, and scripts. This capability ensures that changes are tracked, and that previous versions can be restored if needed.

C09 - Trust/empower teams to make decisions and changes: Empowering teams to make decisions about tools and technologies fosters a culture of trust and autonomy. This capability improves software delivery performance by enabling teams to take ownership of their work and make informed decisions.

C11 - Configuration management: Configuration management involves automating the management and maintenance of infrastructure and systems. By using a Software Configuration Management (SCM) system, teams can ensure consistency across environments and reduce the risk of errors.

C12 - Cloud infrastructure and cloud-native: Leveraging cloud computing characteristics allows for flexible and scalable application deployment. This capability enables teams to take advantage of cloud-native features such as auto-scaling and on-demand resources.

C13 - Artifacts versioning and registry: Centrally storing and versioning artifacts and dependencies is essential for reliable and repeatable software delivery. This capability ensures that teams have access to the correct versions of components needed for deployment.

C14 - Loosely coupled architecture/microservices: Designing applications with independent, communicating services improves agility and scalability. This capability, often implemented through microservices, allows teams to deploy and scale individual components without affecting the entire system.

C15 - Database change management/release alignment: Managing database updates and system configurations is crucial for improving release reliability. This capability ensures that database changes are aligned with application releases, reducing the risk of deployment issues.

C16 - Infrastructure as code: Expressing infrastructure procedures in code allows for repeatable and efficient management. This capability enables teams to automate the provisioning and management of infrastructure, ensuring consistency and reducing manual effort.

C18 - Containerization: Containerization involves using containers for efficient application development, deployment, and management. This capability allows teams to package applications and their dependencies in a consistent environment, ensuring that they run reliably across different platforms.

C20 - Shift left on security: Integrating security practices early in the development process, known as "shift left on security," helps improve security and compliance. This capability ensures that security is considered from the beginning of the development cycle, reducing the risk of vulnerabilities.

C22 - Trunk-based development: Trunk-based development involves working in small batches and frequently merging changes into the main branch. This capability supports continuous integration and reduces the complexity of merging code, leading to more stable releases.

C26 - Centralized log management: Collecting and centralizing logs simplifies debugging and management. This capability allows teams to access logs from multiple sources in one place, making it easier to identify and resolve issues.

C34 - Test data management: Effective test data management is essential for supporting testing processes and ensuring fast, secure access to test data. This capability helps teams maintain the quality and security of their test environments.

C35 - Chaos engineering: Chaos engineering is the practice of building systems capable of dealing with unexpected problems and shortcomings in production environments. This capability helps teams identify weaknesses and improve system resilience.

C36 - Code maintainability: Ensuring code is easy to identify, reuse, and alter is critical for producing high-quality outcomes. This capability reduces technical debt and makes it easier for teams to maintain and enhance the codebase over time.

2.4.2. RQ1 – Report the Findings

The Table 7 highlights the key DevOps capabilities identified by respondents, grouped into Cultural, Measurement, Process, and Technical categories. It serves as a foundation for addressing RQ1. By analyzing these capabilities, we can identify areas where AI could enhance efficiency, automation, and decision-making, offering valuable insights into the integration of AI within DevOps practices.

Table 7. DevOps Capabilities identified AI usefulness by interviewee

	Capabilities	INT1 (PO)	INT2 (SM)	INT3 (SM)	INT4 (PO)	INT5 (SM)	INT6 (PO)	INT7 (PE)	INT8 (PO)	INT9 (PO)	INT10 (PO)	TOTAL (nº/%)
Cultural 7	**C01**				✓	✓		✓				3(33%)
	C08							✓	✓	✓		3(33%)
	C19					✓						1(11%)
	C21									✓		1(11%)
	C24											0(0%)
	C31											0(0%)
	C33							✓				1(11%)
	C04				✓	✓	✓	✓	✓	✓	✓	7(78%)
	C17		✓			✓		✓	✓	✓	✓	6(67%)

	Capabilities	INT1 (PO)	INT2 (SM)	INT3 (SM)	INT4 (PO)	INT5 (SM)	INT6 (PO)	INT7 (PE)	INT8 (PO)	INT9 (PO)	INT10 (PO)	TOTAL (nº/%)
Measurement 5	**C23**					✓	✓	✓		✓	✓	5(56%)
	C29									✓	✓	2(22%)
	C37					✓		✓		✓	✓	4(44%)
Process 7	**C06**				✓	✓		✓		✓	✓	5(56%)
	C10											0(0%)
	C25					✓					✓	2(22%)
	C27					✓			✓		✓	3(33%)
	C28			✓		✓					✓	3(33%)
	C30											0(0%)
	C32					✓		✓		✓	✓	4(44%)
Technical 18	**C02**		✓	✓	✓	✓		✓			✓	6(67%)
	C03		✓	✓	✓	✓		✓		✓	✓	7(78%)
	C05		✓	✓	✓	✓	✓	✓		✓	✓	8(89%)
	C07			✓	✓	✓	✓	✓			✓	6(67%)
	C09			✓				✓			✓	3(33%)
	C11			✓				✓			✓	3(33%)
	C12			✓		✓		✓			✓	4(44%)
	C13					✓		✓			✓	3(33%)
	C14							✓			✓	2(22%)
	C15					✓		✓		✓	✓	4(44%)
	C16					✓		✓			✓	3(33%)
	C18		✓			✓		✓			✓	4(44%)
	C20					✓		✓			✓	3(33%)
	C22					✓		✓		✓	✓	4(44%)
	C26					✓		✓		✓	✓	4(44%)
	C34		✓			✓	✓	✓		✓	✓	6(67%)
	C35					✓		✓		✓	✓	4(44%)
	C36				✓	✓		✓		✓	✓	5(56%)
TOTAL		0	6	8	8	26	5	27	4	17	28	

Cultural Capabilities

Of those who identified capabilities, 56% pointed to at least one Cultural Capability. Interestingly, 80% of these respondents singled out two or fewer from the seven Cultural Capabilities available. When we break it down, 60% of Product Owners (POs) who recognized any capabilities identified at least one Cultural Capability, while only 33% of Scrum Masters (SMs) did the same. The Cultural Capabilities C19, C21, and C33 were each identified by just one person, and no one identified C24 or C31.

Measurement Capabilities

Around 89% of respondents who recognized capabilities identified at least one Measurement Capability. Within this group, half of the respondents identified four or more of the five Measurement Capabilities, while the other half identified two or fewer. Notably, every Product Owner (100%) who pointed out capabilities also recognized at least one Measurement Capability, compared to 67% of Scrum Masters. Certain Measurement Capabilities, like C04, C17, and C23, were identified by over 55% of respondents. While none were unanimously recognized, C04 was particularly notable, being identified by 78% of respondents.

Process Capabilities

In total, 78% of those who identified capabilities recognized at least one Process Capability. Among these respondents, 71% highlighted two or fewer out of the seven Process Capabilities available. Specifically, 80% of Product Owners and 67% of Scrum Masters recognized at least one Process Capability. It's interesting to note that no one identified Process Capabilities C10 and C30. The Process Capability 06 was identified by over 55% of respondents.

Technical Capabilities

Of those who identified capabilities, 89% pointed to at least one Technical Capability. In this group, 33% identified 15 or more of the 18 Technical Capabilities, and 78% identified five or more. Among Product Owners, 80% recognized at least one Technical Capability, while every Scrum Master (100%) did so. Certain Technical Capabilities, like C02, C03, C05, C07, C34, and C36, were identified by over 55% of respondents. While none were unanimously recognized, C03 and C05 were particularly notable, being identified by 78% and 89% of respondents, respectively.

2.5. RQ2 - What benefits do you identify in integrating AI with DevOps?

In addressing Research Question 2 (RQ2) — What benefits do you identify in integrating AI with DevOps? — this section explores the insights gathered from the interviewees. As illustrated earlier in Table 5, the interviewees identified a total of 45 benefits of integrating AI within DevOps practices. These benefits were subsequently analysed and grouped based on thematic similarity, with the aim of consolidating those that expressed similar concepts in different words. This process of grouping allowed for a clearer understanding of the core benefits recognized by the participants.

The most frequently highlighted benefit, *Reduction of Manual Tasks*, was consistently recognized across interviews. Respondents noted its role in reducing human error, automating repetitive tasks, and significantly lowering the overall manual workload, which helps in streamlining processes and improving efficiency.

Performance Improvement was another crucial benefit, with interviewees frequently mentioning enhanced operational performance, increased productivity, more streamlined workflows, and faster delivery speeds due to AI integration. This was seen as an essential advantage for maintaining competitive operations in DevOps.

Ops Support also emerged as a significant category, where AI-driven automation of incident resolution, improved problem-solving capabilities, and reduced operational burdens were noted by respondents. These improvements help DevOps teams resolve issues faster and more effectively, minimizing downtime.

A key benefit frequently discussed was *Better Code Quality and Development Support*. Interviewees emphasized how AI enhances code quality, supports the development process, and aids in implementing industry best practices, leading to more robust and reliable software.

Monitoring and Anomaly Detection was also highly valued, with interviewees pointing out AI's capability to identify patterns, monitor systems effectively, and detect errors early, which contributes to better system reliability and faster issue resolution.

The category *Alerting and Notifications Improvement* encompasses benefits related to better workload monitoring, real-time alert systems, and more effective alarm notifications. These improvements help teams stay informed about potential issues before they escalate into larger problems.

Customer Satisfaction was consistently emphasized as a key outcome of AI integration. Respondents linked enhanced operations and code quality improvements to higher levels of customer satisfaction, a critical success metric for DevOps teams.

The *Predictivity Enhancement* benefit captures various predictive capabilities that AI offers, such as predicting software releases, improving predictive analysis, making better forecasts, and enabling faster recovery from issues. These predictive tools help teams plan more effectively and act proactively.

Additional benefits were also identified, though mentioned less frequently. These include *Focus and Efficiency*, where AI helps DevOps teams maintain focus on more complex tasks and improve overall team efficiency. *Deployment Improvement* was noted for faster and more reliable deployment processes due to AI-driven automation. *Improved Security and Compliance* was seen as a valuable tool for enhancing security measures and ensuring compliance with industry regulations. *Job Satisfaction and Culture* contributed to an improved work environment, enhancing job satisfaction and fostering a positive team culture. *Proactive Insights* were highlighted for enabling teams to anticipate issues before they occur, and *Enhanced Communication* was recognized for its role in improving team coordination and streamlining workflows.

The Table 8 maps these grouped benefits to the specific interviewees who identified them, offering a clear visualization of which participants mentioned each benefit. This matrix highlights the frequency and distribution of the identified benefits across the different interviews, providing insight into the consensus and diversity of perspectives regarding the advantages of integrating AI with DevOps.

Table 8. DevOps and AI integration benefits Identified by Interviewee

Benefit	**INT 1**	**INT 2**	**INT 3**	**INT 4**	**INT 5**	**INT 6**	**INT 7**	**INT 8**	**INT 9**	**INT 10**
Reduction of Manual Tasks	✓	✓				✓		✓		✓
Performance Improvement	✓				✓	✓	✓			✓
Ops Support		✓	✓				✓		✓	✓
Better Code Quality and Development Support			✓	✓			✓		✓	✓
Monitoring and Anomaly Detection		✓			✓		✓			✓
Alerting and Notifications Improvement	✓	✓	✓							
Customer Satisfaction	✓			✓	✓					
Predictivity Enhancement	✓				✓	✓				
Focus and Efficiency	✓							✓		
Deployment Improvement				✓	✓					
Improve Security and Compliance	✓									
Job Satisfaction and Culture	✓									
Proactive Insights	✓									
Enhanced Communication				✓						

2.6. RQ3 - What challenges do you foresee in this integration?

In addressing Research Question 3 (RQ3) — What challenges do you foresee in this integration? — this section delves into the obstacles identified by the interviewees in integrating AI with DevOps. As previously presented in Table 5, the interviewees highlighted a total of 34 challenges. These challenges were analyzed and grouped by similar themes, aiming to consolidate those that expressed the same concern in different terms, thereby providing a clearer and more organized view of the key issues.

Managing Expectations and Over-Reliance on AI represents challenges related to the human-AI relationship. This challenge addresses the need to manage expectations surrounding AI's capabilities and to avoid becoming overly reliant on AI systems, ensuring that they are used as tools rather than replacements for human judgment.

Resistance to Change and Scepticism Towards AI is another significant challenge. This grouping captures both the general resistance to change and the specific scepticism about AI's reliability. Interviewees expressed concerns about distrust towards AI, scepticism regarding the quality of information it provides, and broader apprehensions about relying on technology.

Another identified challenge is the *Increased Initial Effort and Time to Train*, which revolves around the additional effort and time required to effectively train AI systems, particularly during the initial stages of implementation. For instance, interviewees mentioned the increased effort needed at the outset, the time commitment required for AI training, and the learning curve associated with AI adoption.

The challenge category, *Adapting Processes and Routines*, reflects the necessity of modifying existing processes and routines to accommodate AI, along with the difficulties encountered in integrating AI into established workflows. This challenge highlights the organizational adjustments required to seamlessly incorporate AI into current operations.

Another practical challenge is *Choosing and Optimizing AI Tools* which focuses on the difficulties in selecting the appropriate AI tools and optimizing them to meet organizational needs. Interviewees pointed out the importance of making informed decisions when choosing AI models and the continuous effort required to fine-tune these tools for optimal performance.

Security and Data Privacy challenges highlight concerns about protecting data and intellectual property in the context of AI integration. This challenge underscores the importance of maintaining robust security measures and ensuring data privacy as AI systems are increasingly deployed.

Lastly, *Knowledge Maintenance* was identified as a standalone challenge. This challenge emphasizes the ongoing need to stay updated with the rapid developments in AI, which requires continuous learning and adaptation to new advancements in the field.

The Table 9 maps these grouped challenges to the specific interviewees who identified them, providing a clear visualization of which participants mentioned each challenge. This matrix not only highlights the frequency with which certain challenges were cited but also offers insight into the distribution of these challenges across different interviewees.

Table 9. DevOps and AI integration challenges Identified by Interviewee

Challenge	**INT 1**	**INT 2**	**INT 3**	**INT 4**	**INT 5**	**INT 6**	**INT 7**	**INT 8**	**INT 9**	**INT 10**
Managing Expectations and Over-Reliance on AI	✓	✓	✓		✓			✓	✓	✓
Resistance to Change and skepticism towards AI		✓		✓		✓	✓		✓	✓
Increased Initial Effort and Time to train	✓				✓		✓	✓		✓
Adapting Processes and Routines		✓		✓				✓		✓
Choosing and Optimizing AI Tools		✓			✓		✓			
Security and Data Privacy				✓		✓				
Knowledge Maintenance			✓							

2.7. RQ4 - What are your future prospects for this integration?

In addressing the research question, "What are your future prospects for this integration?" this section explores the visions and expectations that interviewees have regarding the future role of AI within DevOps environments. As previously illustrated in Table 5 , a total of 21 prospects were identified by the interviewees. To streamline the analysis, these prospects were grouped by similar themes, allowing for a more cohesive understanding of the overarching trends and expectations.

The first grouping, *AI as an Autonomous or Permanent Tool,* covers the perspectives of interviewees who see AI becoming an integral, enduring element of DevOps, functioning either autonomously or as a constant presence within teams. This view suggests that AI will not only persist but also evolve into a crucial team member, fundamentally altering how DevOps teams operate.

AI Complementing Human Efforts (Not Replacing) highlights the belief that AI will play a supportive role, enhancing human productivity without displacing developers. Interviewees emphasized that AI's function will be to augment rather than replace human efforts, thereby facilitating a more productive and harmonious collaboration between AI and human workers.

The prospect of *AI Becoming a Norm in DevOps* reflects the expectation that AI will soon become a standard feature within DevOps, expanding into broader areas such as business and security operations. This view is shared by those who foresee AI's integration into DevOps as inevitable and transformative, leading to the development of new operational paradigms like "BizDevOps" and "DevSecOps."

Gradual Adoption and Integration of AI reflects the idea that the integration of AI into DevOps will follow a phased approach. Interviewees describe an initial period of slow adoption, followed by a more rapid expansion as AI's benefits become clearer, ultimately leading to structural changes within DevOps as AI’s presence increases over time.

In the prospect of *AI Taking on More Responsibilities,* interviewees foresee AI assuming greater operational duties, which will necessitate the creation of new roles dedicated to AI maintenance. This indicates a future where AI is not only a tool but also a key player in the operational landscape, requiring ongoing management and support.

The grouping *AI Supplementing a Significant Portion of DevOps Work* envisions AI taking on up to 50% of DevOps tasks. This would represent a major shift in the division of labor between AI systems and human developers, potentially leading to a substantial reconfiguration of DevOps workflows as AI handles routine responsibilities.

Potential Legal and Financial Issues acknowledges the challenges that may accompany AI's deeper integration into DevOps. This grouping highlights the need to anticipate and address the legal and financial implications that could arise as AI becomes more embedded in these environments.

The *Focus Shift for Developers to Complex Tasks* suggests that as AI increasingly manages routine operations, developers will be freed to concentrate on more complex and creative challenges. This shift could lead to a revaluation of roles within DevOps teams, with AI handling repetitive tasks while developers focus on innovation and problem-solving.

The prospect of *Faster and Cost-Effective Training* reflects the efficiency gains anticipated from AI, envisioning AI contributing to more streamlined and economical training processes. Interviewees suggested that AI could reduce the time and cost associated with training, thus enhancing the overall efficiency of DevOps operations.

Lastly, the need for *Careful Management of AI* underscores the importance of deliberate and strategic management practices to ensure that AI is effectively integrated and utilized within DevOps. This prospect points to the necessity of vigilance and oversight in AI deployment to mitigate potential risks and maximize its benefits.

The Table 10 maps the identified prospects to the respective interviewees, providing a clear overview of which interviewees expressed each prospect and how frequently each was mentioned. This mapping not only highlights the collective expectations for the future integration of AI within DevOps but also reveals the alignment or divergence in perspectives among the interviewees.

Table 10. DevOps and AI integration prospects Identified by Interviewee

Prospect	INT 1	INT 2	INT 3	INT 4	INT 5	INT 6	INT 7	INT 8	INT 9	INT 10
AI as an Autonomous or Permanent Tool	✓	✓			✓					✓
AI Complementing Human Efforts (Not Replacing)	✓	✓				✓		✓		
AI becoming a norm in DevOps	✓			✓	✓					
Gradual Adoption and Integration of AI							✓	✓		✓
AI Taking on More Responsibilities									✓	✓
AI Supplementing a Significant Portion of DevOps Work			✓							
Potential Legal and Financial Issues			✓							
Careful Management of AI				✓						
Focus Shift for Developers to Complex Tasks						✓				
Faster and Cost-Effective Training									✓	

CHAPTER 3

Comparative Analysis of SLR and Interviews Findings on AI Integration in DevOps

The findings from the SLR and the interviews conducted share both common themes and notable differences, reflecting the evolving understanding of AI integration within DevOps environments.

One of the key similarities between the SLR and the interview findings is the recognition of AI's potential to significantly enhance various aspects of the DevOps lifecycle. Both sources highlight AI's role in automating repetitive tasks, improving efficiency, and enabling more sophisticated monitoring and prediction capabilities. For instance, the SLR discusses AI's contribution to continuous monitoring, fault detection, and predictive maintenance within the DevOps pipeline, which aligns with interviewees' observations that AI can provide proactive monitoring, alert notifications, and emergency response capabilities. These shared views underscore the broad consensus on AI's value in reducing manual workloads and enhancing system reliability.

Another area of convergence is the emphasis on AI's ability to support decision-making processes. The SLR points out that AI can aid in prioritizing tasks and making data-driven decisions, particularly through advanced quality assessment tools and intelligent recommendations. Similarly, the interviewees noted AI's capacity to provide data-driven feedback, suggest performance improvements, and assist in incident resolution by analyzing historical patterns. This suggests that both the academic literature and practitioners see AI as a powerful tool for enhancing the analytical capabilities of DevOps teams, leading to more informed and timely decisions.

However, there are also differences in emphasis between the SLR findings and the interviews. The SLR tends to focus more on the technical challenges and solutions associated with integrating AI into DevOps, such as the need for modularity in large-scale projects, the complexity of real-time data processing, and the development of customized QA techniques for AI systems. The literature highlights the necessity for advanced engineering tools and the configuration of MLOps pipelines to accommodate the specific needs of AI applications. In contrast, the interviews place a stronger emphasis on the practical and cultural challenges of AI integration. Interviewees frequently mentioned issues like resistance to change, the need for team training, and managing expectations regarding AI's capabilities. These concerns reflect a more human-centric perspective, focusing on how AI impacts team dynamics and the day-to-day operations of DevOps teams.

Moreover, the interviews bring to light concerns about the potential over-reliance on AI and the need for careful management of AI tools, something that is less pronounced in the SLR findings. Interviewees cautioned against viewing AI as the solution for every problem and stressed the importance of maintaining human oversight, particularly in areas requiring nuanced judgment, such as cultural capabilities and decision-making during critical deployments. This contrasts with the more optimistic tone in the SLR, which largely focuses on the technical benefits and advancements brought by AI, suggesting a more seamless integration into the DevOps pipeline.

In terms of future outlook, both the SLR and interviews acknowledge the growing importance of AI in DevOps, but they differ in their predictions about its trajectory. The SLR often points to a future where AI is deeply embedded in all aspects of DevOps, driving continuous innovation and efficiency. In contrast, interviewees were more cautious, with some predicting a gradual adoption of AI, particularly in specific areas where it can complement rather than replace human efforts. This difference highlights the gap between the theoretical potential of AI, as discussed in the literature, and the practical realities and reservations expressed by practitioners.

Overall, while there is considerable alignment between the SLR findings and the interviews regarding the benefits of AI in DevOps, the interviews provide a more nuanced understanding of the challenges and limitations that may arise during implementation. The SLR offers a more technical and optimistic perspective, whereas the interviews provide a grounded view, emphasizing the importance of human factors and the need for careful management of AI integration. This comparison underscores the need for a balanced approach that considers both the technical possibilities and the practical implications of AI in the DevOps context.

CHAPTER 4

Conclusion

4.1. Research Conclusions

The integration of AI within DevOps processes offers a promising way for enhancing agile product development. This research sought to explore the potential of AI in DevOps by addressing four research questions (RQs) concerning AI's usefulness, benefits, challenges, and future prospects in this context. The findings from interviews with practitioners provide valuable insights into how AI can be effectively integrated into DevOps, while also highlighting the complexities and challenges associated with this integration.

The study identified several DevOps capabilities where AI can provide substantial support or enhancement. Key capabilities such as test automation (C05), continuous delivery and deployment automation (C03), and proactive monitoring (C04) were recognized as areas where AI could add significant value, as identified by a majority of respondents. Test automation (C05), in particular, was highlighted by 89% of respondents, underscoring its importance in reducing manual testing efforts and increasing the speed of delivery. Continuous delivery and deployment automation (C03) and proactive monitoring (C04) were each recognized by 78% of respondents, indicating strong support for AI's role in these areas. These findings reflect a consensus on the practical areas within DevOps where AI can make substantial contributions, particularly in automating repetitive tasks, enhancing workflow efficiency, and improving monitoring and alerting systems.

The research also identified several key benefits of integrating AI with DevOps, most notably in reducing manual tasks, improving performance, supporting operational tasks, and enhancing code quality and development processes. AI's ability to automate repetitive and mundane tasks enables DevOps teams to focus on more strategic and creative aspects of development. Additionally, AI-driven tools can enhance performance by providing more accurate and real-time monitoring and analytics, supporting both incident management and proactive failure prevention. These benefits suggest that AI integration can significantly optimize DevOps processes, leading to more efficient, reliable, and agile product development.

While the potential benefits of AI integration in DevOps are substantial, the research also identified significant challenges that organizations may face. Key challenges include managing expectations and preventing over-reliance on AI, resistance to change and scepticism towards AI, and the increased initial effort and time required for training AI models and systems. These challenges underscore the importance of a balanced approach to AI integration that manages both technological and human factors. The findings highlight a critical need for organizations to foster a culture of continuous learning and adaptability while ensuring that AI is viewed as a tool to support human decision-making rather than replace it. This perspective aligns with the concerns raised by practitioners about the cultural and operational impacts of AI integration.

Looking ahead, interview findings suggest a promising but cautious future for AI integration in DevOps. Interviewees emphasized a gradual approach to AI adoption, envisioning AI as a complement to human efforts rather than a replacement. They foresee AI becoming a norm in DevOps over time, particularly in specific areas where its capabilities can maximize benefits while mitigating potential risks. This nuanced view is crucial for ensuring sustainable and effective AI adoption in DevOps environments, where AI is used strategically to enhance specific capabilities without undermining the essential human elements of creativity, judgment, and collaboration.

A comparative analysis of the SLR and interview findings reveals both convergences and divergences in understanding AI integration in DevOps. Both sources agree on AI's potential to enhance DevOps processes by automating repetitive tasks, improving decision-making, and providing predictive analytics. For instance, the SLR discusses AI's contribution to continuous monitoring, fault detection, and predictive maintenance within the DevOps pipeline, which aligns with interviewees' observations that AI can provide proactive monitoring, alert notifications, and emergency response capabilities. However, while the SLR is more focused on the technical aspects of AI integration, such as the need for modular architectures and advanced engineering tools, the interviews emphasize the cultural and practical challenges, including resistance to change, the need for training, and managing AI expectations. Moreover, the interviews highlight concerns about potential over-reliance on AI and the necessity for maintaining human oversight, which is less pronounced in the SLR findings. These differences highlight the gap between theoretical potential and practical realities, suggesting that successful AI integration requires addressing both technological advancements and human-centric concerns.

In conclusion, this research underscores the significant potential of AI to transform DevOps practices by enhancing key capabilities, improving decision-making, and supporting agile development goals. However, the integration of AI into DevOps is not without challenges, particularly in managing expectations, addressing cultural resistance, and ensuring a balanced approach that values human oversight and judgment. A successful AI-DevOps integration strategy will require a thoughtful combination of technical innovation and cultural adaptation, fostering an environment where AI complements human efforts and supports continuous improvement. As the field evolves, further research should continue to explore both the technical and human dimensions of AI in DevOps, ensuring that its integration is both effective and sustainable.

4.2. Limitations

Identified limitations of this study include the fact that the scope of the SLR was restricted to specific scientific databases, namely IEEE Xplore, ACM Digital Library, Scopus, and Web of Science. While these databases are comprehensive, they do not encompass all potentially relevant studies, particularly those published in non-traditional or emerging venues. Consequently, this limitation may have led to a narrower view of the literature on AI integration in DevOps, possibly omitting innovative perspectives or lesser-known challenges and solutions documented outside these primary databases.

Another limitation arises from the data constraints related to the year 2023. The study only includes research published up until October 2023, the point at which the SLR was concluded. As a result, any significant studies or advancements published after this date are not reflected in the analysis. Given the rapidly evolving nature of AI and DevOps integration, this temporal limitation may impact the comprehensiveness of the findings. Future research should incorporate more recent literature to maintain an up-to-date perspective on the subject.

Furthermore, the geographical and organizational diversity of the interviewees poses a limitation. Interviews were conducted exclusively within a single Portuguese technology company specializing in software development, which operates with a flat, agile organizational structure emphasizing collaboration across diverse projects. While this setting provides valuable insights, it may not fully capture the experiences of different organizations, particularly those with varying sizes, geographical locations, or organizational hierarchies. This context-specific focus may limit the generalizability of the findings to other sectors or companies with distinct dynamics.

Additionally, although the study reached saturation by the eighth interview, with a noticeable decline in new insights from Interview 6 onward, the sample size remains relatively small. This limitation may affect the generalizability of the findings. Moreover, the interviewees were predominantly professionals actively engaged in DevOps, which could introduce a bias toward practical and immediate concerns over theoretical or long-term considerations.

4.3. Future Work

The findings of this study provide a foundation for future research to further investigate the integration of AI within DevOps. Future studies should explore whether the key areas identified in this research, such as test automation, continuous delivery, and proactive monitoring, remain critical as both AI technologies and DevOps practices continue to evolve. In particular, it will be important to assess how AI can enhance these specific capabilities and to develop frameworks that enable the standardized integration of AI across various DevOps workflows while ensuring that human oversight remains a priority.

Additionally, the scope of this research was limited to a single Portuguese technology company, and further studies could expand this by exploring AI-DevOps integration across different industries, organizational structures, and geographical regions. Understanding how organizational factors such as company size, hierarchy, and regional practices influence AI adoption in DevOps could offer valuable insights. A key question is whether the cultural and operational challenges identified, such as resistance to change and over-reliance on AI, are universally present or more context-dependent. Broadening the scope would also provide a clearer picture of how different sectors utilize AI within their DevOps processes.

At the organizational level, a long-term assessment of AI-driven DevOps processes is necessary. While this research focused on the immediate benefits of AI, such as reducing manual tasks and improving operational efficiency, future work should examine whether these benefits are sustainable over time. An important question for future studies is: which metrics can effectively track the performance of AI in DevOps, and which require human interpretation? The development of such metrics will help ensure that organizations maintain an appropriate balance between AI-driven automation and human involvement in decision-making.

This study also faced limitations in its temporal scope, as it included literature and data available only up until October 2023. With AI and DevOps technologies rapidly advancing, future research must continually update the body of knowledge by incorporating the latest developments. This will help organizations stay at the forefront of AI integration, allowing them to refine their DevOps practices in response to new tools and technologies.

Finally, ethical considerations related to AI integration in DevOps require further attention. While this research briefly touched upon concerns such as over-reliance on AI and the importance of human oversight, future studies should focus more extensively on these issues. A particularly valuable area for research is the development of ethical guidelines that can ensure AI enhances rather than replaces human decision-making. Additionally, investigating how organizations can maintain transparency, accountability, and creativity in AI-driven DevOps environments would support responsible AI adoption.

References

A. Di Stefano, A. Di Stefano, G. Morana y D. Zito, <<Prometheus and AIOps for the orchestration of Cloud-native applications in Ananke>>, 2021 IEEE 30th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE) - Volume 0, Issue 0, pp. 27-32 - published 2021-01-01

A. Dagnino, M. Kolomycki y A. Kucheria, <<MAP: Design, Development, Deployment, and Maintenance of Industrie 4.0 AI Applications>>, 2022 IEEE Eighth International Conference on Big Data Computing Service and Applications (BigDataService) - Volume 0, Issue 0, pp. 108-113 - published 2022-01-01

C. Vuppalapati, A. Ilapakurti, K. Chillara, S. Kedari y V. Mamidi, <<Automating Tiny ML Intelligent Sensors DevOPS Using Microsoft Azure>>, 2020 IEEE International Conference on Big Data (Big Data) - Volume 0, Issue 0, pp. 2375-2384 - published 2020-01-01

A. Alnafessah, A. U. Gias, R. Wang, L. Zhu, G. Casale y A. Filieri, <<Quality-Aware DevOps Research: Where Do We Stand?>>, IEEE Access - Volume 9, Issue 0, pp. 44476-44489 - published 2021-01-01

Z. Wang, M. Shi y C. Li, <<An Intelligent DevOps Platform Research and Design Based on Machine Learning>>, 2020 Eighth International Conference on Advanced Cloud and Big Data (CBD) - Volume 0, Issue 0, pp. 42-47 - published 2020-01-01

Bruneliere, H., Muttillo, V., Eramo, R., Berardinelli, L., Gómez, A., Bagnato, A., Sadovykh, A. y Cicchetti, A., <<AIDOaRt: AI-augmented Automation for DevOps, a model-based framework for continuous development in Cyber–Physical Systems>>, Microprocessors and Microsystems - Volume 94, Issue 0, pp. - published 2022-01-01

K. Matsumoto, <<Conceptual Framework for Next-Generation Software Ecosystems>>, 2021 IEEE/ACIS 22nd International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD) - Volume 0, Issue 0, pp. 218-223 - published 2021-01-01

T. Granlund, A. Kopponen, V. Stirbu, L. Myllyaho y T. Mikkonen, <<MLOps Challenges in Multi-Organization Setup: Experiences from Two Real-World Cases>>, 2021 IEEE/ACM 1st Workshop on AI Engineering - Software Engineering for AI (WAIN) - Volume 0, Issue 0, pp. 82-88 - published 2021-01-01

L. Yang y D. Rossi, <<Quality Monitoring and Assessment of Deployed Deep Learning Models for Network AIOps>>, IEEE Network - Volume 35, Issue 6, pp. 84-90 - published 2021-01-01

Borg, M, <<Pipeline Infrastructure Required to Meet the Requirements on AI>>, IEEE SOFTWARE - Volume 40, Issue 1, pp. 18-22 - published 2023-01-01

Moens, P., Andriessen, B., Sebrechts, M., Volckaert, B. y Van Hoecke, S., <<Edge Anomaly Detection Framework for AIOps in Cloud and IoT>>, International Conference on Cloud Computing and Services Science, CLOSER - Proceedings - Volume 2023, Issue 0, pp. 204-211 - published 2023-01-01

Abdallah Atouani, Jörg Christian Kirchhof, Evgeny Kusmenko y Bernhard Rumpe, <<Artifact and reference models for generative machine learning frameworks and build systems>>, Proceedings of the 20th ACM SIGPLAN International Conference on Generative Programming: Concepts and Experiences - Volume 0, Issue 0, pp. 55–68 - published 2021-01-01

P. Thantharate, <<IntelligentMonitor: Empowering DevOps Environments with Advanced Monitoring and Observability>>, 2023 International Conference on Information Technology (ICIT) - Volume 0, Issue 0, pp. 800-805 - published 2023-01-01

B. Saunders, J. Sim, T. Kingstone, S. Baker, J. Waterfield, B. Bartlam, H. Burroughs, and C. Jinks, <<Saturation in qualitative research: exploring its conceptualization and operationalization>>, Quality and Quantity - Volume 52, Issue 4, pp. 1893–1907 - published 2018-07-01

T. Power, D. Jackson, B. Carter, and R. Weaver, <<Misunderstood as mothers: women's stories of being hospitalized for illness in the postpartum period>>, Journal of Advanced Nursing - Volume 71, Issue 2, p. 372 - published 2015-02-01

M. P. Grady, <<Qualitative and action research: A practitioner handbook>>, Phi Delta Kappa International - published 1998-01-01

R. Amaro, R. Pereira, and M. Silva, <<Capabilities and Practices in DevOps: A Multivocal Literature Review>>, IEEE Transactions on Software Engineering, Volume 49, Issue 2, pp. 883- 901 - published 2023-01-01

APPENDIX A

Interview Guideline

Introduction

My name is Afonso Figueiredo and I, in collaboration with Ruben Filipe Pereira and Miguel Rodrigues Silva, am conducting research as part of the Master's program in Computer Science and Management at ISCTE to explore the integration of DevOps and Artificial Intelligence (AI).

The purpose of this research is to gain insights into how DevOps practices can be enhanced through the integration of AI technologies, and how this integration can contribute to agile product development.

In this research, we are focusing on the concept of AIOps, which stands for Artificial Intelligence for IT Operations, and it refers to the use of AI and ML techniques to address DevOps challenges. It involves integrating AI/ML into the DevOps pipeline to automate operational tasks, extract insights from data, and enhance continuous deployment and operations management. AIOps utilizes data science and computational techniques to automate routine tasks and uses inference models to extract actionable insights from data.

Interviewed Information

Please provide some descriptive information about yourself:

Name	
Role/Position in the company	
Years of experience in DevOps	
Years of experience in AI	
How do you assess your knowledge in DevOps (on a scale of 0-5, 5 being the highest)	
How do you assess your knowledge in AI (on a scale of 0-5, 5 being the highest)	

Discussion Questions

1. In your opinion, considering the capabilities/practices listed where do you see AI being useful in the context of DevOps?
2. What benefits do you identify in integrating AI with DevOps?
3. What challenges do you foresee in this integration?
4. What are your future prospects for this integration?