ISCTE IUL
Instituto Universitário de Lisboa

Departamento de Ciências e Tecnologias da Informação

# LeaPhysio – Games Enhanced Physical Rehabilitation

Filipe da Costa Pereira Lourenço

A Dissertation presented in partial fulfillment of the Requirements for the Degree of Master

Engenharia de Telecomunicações e Informática

Supervisor:
Dr. Octavian Postolache, Assistant Professor,
ISCTE-IUL

October 2017

## Abstract

In certain areas of the medical sciences, particularly in the field of physiotherapy, there is increasing needs on objective evaluation of the rehabilitation progress that creates new opportunities to the application of the ICT. As part of the new technologies the serious games are ones that can help the patient in the way of motivation that may help during recovery process. The dissertation describes the design, implementation and testing of *LeaPhysio – Games Enhanced Physical Rehabilitation* system. This system may be used by the physiotherapists and patients during hand and finger rehabilitation process. Using highly interactive games, developed for Leap Motion controller natural user interfaces, it's possible to assess the patient's performance throughout different physical activities. Through the games, developed in Unity 3D, with controller support, it's possible to obtain the patient performance data during the various training sessions. All information regarding training the established training plans, results and other data is stored on a remote server. The stored information can be accessed through a mobile application, developed in Android, for the physiotherapist, mobile application that can be also used for configuration of the serious games.

The use of serious games allows the patient with motor impairments, produce a dynamic interaction and a different approach in the execution of traditional exercises in recovery treatment.

The system allows an unlimited number of training sessions to be performed, which can be consulted at any time through the mobile application. In this way, it's possible to visualize historical values, and to compare the results of the different training sessions, analyzing their evolution.

**Keywords**: Serious Game; Hand Rehabilitation; Physiotherapy; Leap Motion

## Resumo

Em certas áreas das ciências médicas, nomeadamente no ramo da fisioterapia, há necessidades crescentes na avaliação objetiva do progresso da reabilitação que cria novas oportunidades para a aplicação das Tecnologias de Informação e Comunicação. Como parte das novas tecnologias, os jogos sérios são aqueles que podem ajudar o paciente no caminho da motivação e no processo de recuperação. Este sistema pode ser usado pelos fisioterapeutas e pacientes durante o processo de reabilitação de mãos e dedos. Usando jogos altamente interativos, desenvolvidos para uma interface de utilizador natural, com suporte do controlador *Leap Motion*, é possível avaliar o desempenho do paciente em diferentes atividades físicas. Através dos jogos desenvolvidos em *Unity 3D*, com suporte do controlador, é possível obter os dados de desempenho do paciente durante as várias sessões de treino. Todas as informações relativamente aos planos de treinos estabelecidos, resultados e outros dados, são armazenados em um servidor remoto. As informações armazenadas podem ser acedidas através de uma aplicação móvel, desenvolvida em *Android* e destinada ao fisioterapeuta, que também pode ser usada para configurar os jogos sérios.

O uso de jogos sérios permite ao paciente com deficiências motoras, produzir uma interação dinâmica e uma abordagem diferente na execução de exercícios tradicionais em tratamento de recuperação.

O sistema permite que um número ilimitado de sessões de treino seja realizado, podendo ser consultado a qualquer momento através da aplicação móvel. Desta forma, é possível visualizar valores históricos e comparar os resultados das diferentes sessões de treino, analisando a sua evolução.

**Palavras-chave**: Jogo Sério; Reabilitação da Mão; Fisioterapia; Leap Motion

## Acknowledgments

Firstly, I would like to express my sincere thanks to my supervisor, Prof. Doctor Octavian Postolache, for all the encouragement, availability and support given during the realization of the dissertation.

To Institute of Telecommunications at ISCTE-IUL, thanks for providing all the material and resources needed to develop this project.

Finally, a big thank you to all my friends, family, girlfriend and college colleagues for their support and constant encouragement.

# Contents

## Figures

## Tables

## List of Acronyms

| | |
|---|---|
| 3G | 3rd Generation |
| 4G | 4th Generation |
| ADT | Android Developer Tools |
| AES | Advanced Encryption Standard |
| API | Application Programming Interface |
| APP | Application |
| AR | Augmented Reality |
| CBC | Cipher Block Chaining |
| DB | Database |
| DBMS | Database Management System |
| EHR | Electronic Health Record |
| GUI | Graphical User Interface |
| HCIL | Human-Computer Interaction Lab |
| HTTP | Hypertext Transfer Protocol |
| ICT | Information and Communications Technology |
| IDE | Integrated Development Environment |
| IEEE | Institute of Electrical and Electronics Engineers |
| IR | InfraRed |
| JSON | JavaScript Object Notation |
| LAMP | Linux, Apache, MySQL and PHP |
| LMC | Leap Motion Controller |
| NFC | Near Field Communication |
| OS | Operating System |
| PANAS | Positive and Negative Affect Schedule |
| PHP | Hypertext Preprocessor |
| PPI | Pixels Per Inch |
| QUIS | Questionnaire for User Interaction Satisfaction |
| REST | Representational State Transfer |
| RFID | Radio-Frequency IDentification |
| SDK | Software Development Kit |
| SQL | Structured Query Language |
| SSH | Secure Shell |
| UI | User Interface |
| URL | Uniform Resource Locator |
| USB | Universal Serial Bus |
| VPS | Virtual Private Server |
| VR | Virtual Reality |

# Chapter 1 – Introduction

## 1. Introduction

### 1.1. Motivation and Overview

Medical sciences or health sciences are areas of study related to a life that play a fundamental and essential role in our daily. It has branches in different areas, such as human medicine, biology, nursing, psychology, nutrition, physiotherapy, among others. Over time we have witnessed an exponential evolution of these areas, obtaining results such as:

- New methods for treating diseases are designed and others improved;
- New medicines are created to cure diseases;
- Investigation of diseases in constant development;
- Significant improvements in existing resources and equipment compared to the past decade;
- Evolution of health sciences;

The health sciences mainly use some methodological principles and actions [1]:

- Initially a diagnostic phase that serves to analyze the patient's condition and perform clinical exams;
- A phase of updating, indicating medications, performing operations, physical activities, specific therapeutic exercises, among others;
- Finally, a follow-up phase consisting of an exams and tests, clinical evaluation and periodic returns.

In the world around us, there are people with physical and / or motor needs due to various reasons, whether due to an accident in the workplace, a birth problem or due to some age-related constraints. People with these needs in some cases cannot perform daily tasks independently in their daily lives, such as picking up a glass without difficulty, wearing clothes, or even showering. They are tasks easily performed by people without any physical and / or motor limitations, but for people who have these constraints they become complicated or even impossible to execute. These kinds of consequences can restrict personal activities, preventing the person from occasionally participating in the community in which they are inserted or having a negative effect on their daily work.

The use of physiotherapy methods performed in specialized clinics becomes an essential means to help people to overcome partially or totally their physical limitations. The execution of physiotherapy exercises as part of physical rehabilitation plan made possible to reduce existing physical limitations of the patient, which leads to a direct impact on the improvement of the performance of tasks carried out by the patient. In a study conducted in the United States [2], with 312 stroke survivors (with a mean age of 63 years old), only 31% performed the exercises associated with their rehabilitation plan. Failure to perform the exercises recommended by specialists in the area conducts to negative consequences for a person's life and may conduct to worse their clinical condition. However, participating in appropriate and repetitive rehabilitation exercises may help people with these limitations try to overcome some of them, but sometimes refer to lack of motivation as a hindrance to performing the exercises regularly [3]. The number of exercises in a therapy session is usually insufficient [4]. One of the solutions to this problem is personal intervention. However, it may not be economically feasible [5] for clinics, mainly due to economic constraints.

The fun that an interactive game applied to physical rehabilitation, can be a key factor to improve the patient performance and a factor in improving their physical limitations.

Through serious games that motivate people with physical limitations to perform rehabilitation exercises, it’s possible to help people improve their lives. There is a growing need to apply current new technologies in certain areas of the medical sciences, particularly in the field of physiotherapy, to achieve better patient rehabilitation outcomes, but also to create new work resources for physiotherapists. Through existing technology, it’s possible to create tools that help professionals during their consultations or create new features to aid performance and understand patients' clinical evolution over time.

By pooling the knowledge of a physiotherapist and computer scientist, it’s possible to create new working tools for physiotherapy based on Virtual Reality (VR) and Augmented Reality (AR). VR or AR scenarios, as part of physical rehabilitation gamification and natural user interface expressed by electronic sensors, creates a different way for performing rehabilitation exercises compared to traditional exercises, helping the patient during physical rehabilitation, plan with increased level of motivation. The usage of VR or AR rehabilitation frameworks made possible to collect and analyze the data associated with execution of the exercises and to present analyzed information in a simple and highly personalized format for the patient that it can be used for biofeedback or for the physiotherapists and medical doctors making decision.

## 1.2. Research Questions and Objectives

The objective of this research is to develop a system that runs serious games characterized by the usage of motion and depth controller (Leap Motion Controller [6]) and VR. Applied to a natural interface technology for the physical rehabilitation of hands and fingers for people with physical and motor limitations. This system aims to introduce metrics to assess the patient's clinical status and improve their physical limitations, complementing traditional rehabilitation systems in the physiotherapy area. The use of serious games in conjunction with the Leap Motion Controller (LMC) and VR allows the patient with physical and / or motor needs, a dynamic interaction and a different approach in the execution of traditional exercises in the treatment of hands and fingers, trying to help with the motivational issue by allowing different activities to be performed repeatedly over a pre-set period.

With this type of technology, it’s possible to create platforms for the patient to use in their rehabilitation plan. As far as the physiotherapist's point of view is concerned, it’s possible for the therapist to configure parameters related to serious games so that they are adaptable to the specific needs of each patient. The results obtained from the use of the games are stored in a data storage system in which they are statistically treated to elaborate metrics that will be used by physiotherapists to evaluate the evolution of the patient during the period of motor rehabilitation, and information reports of rehabilitation sessions on the patient may also be originated in electronic form. The resulting information can be viewed using a mobile application (developed in Android), to be used by professionals in the field of physiotherapy.

This project aims to obtain an answer to certain questions, such as:

- LeaPhysio will be a therapy tool for the physical rehabilitation of the hands and fingers?
- Will be possible to develop metrics capable of helping physiotherapists to assess the patient's progress during the period of motor rehabilitation?
- What will be the performance of the LeaPhysio system on finger detection?
- Is it possible to create highly interactive configurable and personalized serious games that will be part of the rehabilitation process and patient motivation?
- Will LMC able to respond to the needs imposed by LeaPhysio and the patients?

## 1.3. Research and Planning Method

The research method used in this project is composed of five distinct phases:

- **1st Phase → Definition of objectives** – In this phase a set of technologies and mechanisms are studied to know its impacts on the rehabilitation of patients in the physiotherapy area. These should guide the research, in which the project must comply;
- **2nd Phase → System prototype development** – At this stage the different components of the project are developed to compress with the objectives, such as:
  - System architecture LeaPhysio;
  - Serious games applied to hand and finger physiotherapy with LMC support, using Unity 3D for its development;
  - Application for mobile device, developed for Android and intended for physiotherapists, serving as support to see information;
  - Creation and structure of a database for the storage of information.
- **3rd Phase → System prototype implementation** – In a sequential structure the different components developed in the previous phase must be implemented to create a project prototype;
- **4th Phase → System prototype tests** – A set of tests and adjustments to the system is performed to obtain a final prototype;
- **5th Phase → Evaluation** – Final prototype of the project must be tested and evaluated.

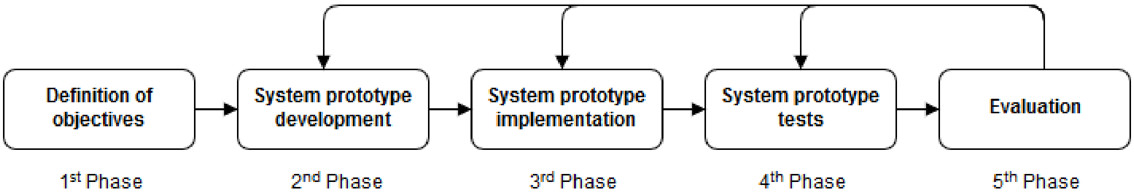


Figure 1.1 – Flow of the Research Method

The 2nd, 3rd and 4th phases are cyclical. At the end of the fifth phase, if the final prototype is in accordance with the objectives defined in the first phase, the project is finalized. If not, it's necessary to analyze the problem or improvements to be made and to return to the research phase necessary to combat difficulties between the objectives initially defined and the final prototype.

## 1.4. Structure of the dissertation

Chapter 2 includes a literature review on serious games and how the LMC architecture can be used in physical rehabilitation. Chapter 3 describes the entire LeaPhysio system and how it works. The serious games applied to physical rehabilitation are explored in Chapter 4. Chapter 5 describes the implementation of an information system for storing data, mainly expressed by database. Chapter 6 describes the support for physical therapists to create training plans for the various patients and the ability to visualize their results using a mobile application. Chapter 7 considers the experimental results and evaluation performed. Chapter 8 is concerned with conclusions and future work. Lastly, a user manual on system applications and a technical manual explaining the main functions of the applications.

# Chapter 2 – State of art

## 2. Overview

This chapter presents the research on serious games and sensors, the reported solution in the literature and the reason for using the LMC. It presents some medical applications on the market and remote data storage solutions.

### 2.1. Serious games and Sensors

The term **Serious Games** refers to digital games dedicated to various purposes that are based not only on entertainment and fun but also promote teaching, research, health, among others. Some examples:

- **VHealthCare™** [7] (see Figure 2.1) – Is a virtual environment that allows health professionals to practice their skills and learn new techniques. it's based on a set of tools that can evaluate the competence and knowledge of doctors, nurses, dentists and others;

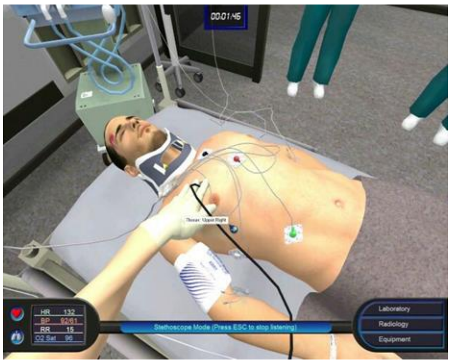


Figure 2.1 – Example of the game VHealthCare

- **The Incredible Adventures of the Amazing Food Detective** [8] (see Figure 2.2) – Is a game used to teach children to have a healthy diet.

Figure 2.2 – The Incredible Adventures of the Amazing Food Detective

In addition to the traditional use of a game with pure entertainment function, serious games open doors to a new dimension of use, for example, a help of sensors capable of detecting movements. The new generation of games and platforms available on the market have a growing role in the medical sciences, especially in the field of physiotherapy and rehabilitation. An example is the Xbox Fitness game [9], which promotes physical exercise using the Microsoft Kinect device [10]. In a study by Josef Wiemeyer and Annika Kliem [11], they analyze the impact of serious games in terms of prevention and rehabilitation. The authors describe the benefit of serious games by applying three criteria:

- **Efficiency and effectiveness** of the intervention;
- **Additional benefits**, such as psychological factors;
- **Quality of study**.

They conclude that serious games produce substantial benefits for a part of the patients and that they must be developed to be completely adaptable to the needs of the patient, and it’s necessary to evaluate some configurations of support for motivation through different social systems.

In [12], the authors evaluate a set of commercial games together with a group of physicians to identify whether computer games can help people during Stroke rehabilitation process. They have concluded that games have a lot to offer in terms of patient care, especially well-designed games with a good User Interface (UI) and adaptable to the needs, can promote the movements of the patients' members bringing benefits to a person with stroke. Challenging games where the user is challenged for a certain time to perform certain tasks to obtain a better score are considered interesting in motivational terms (see Figure 2.3).

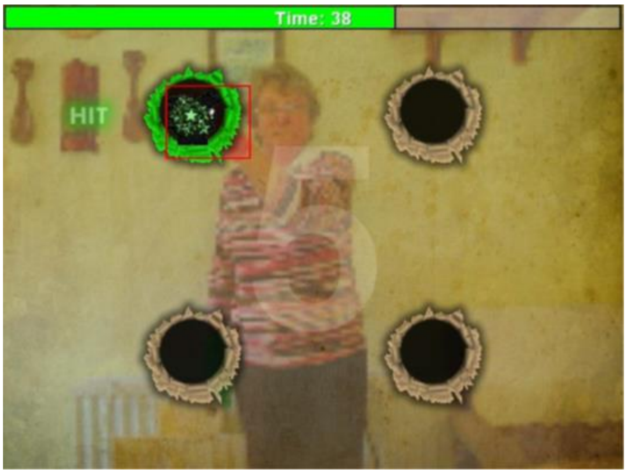


Figure 2.3 – Example of using a serious game [12]

Video games are not just for pure fun or entertainment, they can be used to help people improve their physical and / or motor skills. The virtual reality scenarios are increasingly used through games in various areas of medical sciences and the field of physiotherapy doesn't escape the rule. This new generation of tools available for use in physiotherapy and rehabilitation has seen an exponential growth over the years. Through this resource it's possible to perform a rehabilitation treatment in a more fun and perhaps motivating the patient to perform with more commitment the proposed exercises. On the other hand, there are also several games that require many sensors attached to the body, either to detect the movements performed or to perform measurements. This type of sensors can cause some wear and discomfort to the patient, and can lead to a demotivation in the use of the same [13].

Vicon [14] (not used for video games) is developer of motion capture products and services for the Life Science, Entertainment and Engineering industries. The Vicon Biomechanics and Sports Science community encompasses many applications, including research, sports performance and animal science. The analysis of movement for sports performance enhancement or injury prevention, whether clinical, research or educational, requires a fully flexible motion measurement system capable of capturing motion in all environments with minimal set-up (see Figure 2.4).

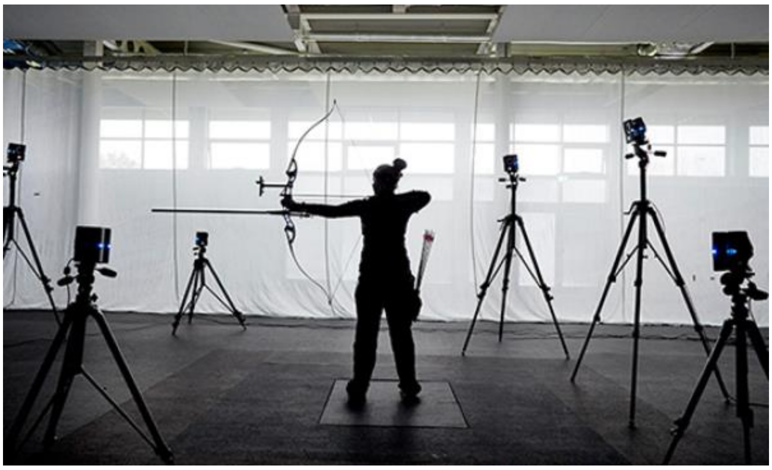


Figure 2.4 – VICON system

A project by Jack [15], a virtual reality system supported by a computation platform was used for hand rehabilitation in stroke patients. The system uses two devices to allow interaction between the user and the proposed exercises:

- CyberGlove [16], a glove that captures the movement of the hand and recreates the 3D virtual reality scenario;
- And a glove designed for virtual reality that offers force feedback.

Both gloves are somewhat expensive and may not be accessible to many people with motor limitations or by regular clinics. The use of gloves requires a special configuration in the virtual reality scenario for each patient

R. Unnikrishnan has developed a prototype [17] that focuses on the use of various industrial motion and entertainment sensors along with games to be used in physiotherapy. Using these devices, it's possible to define a set of movements with different parts of the body, so that it's possible to control the game. One of the examples given is with the video game **Tux Racer** [18] (see Figure 2.5), where **Microsoft Kinect** or **Nintendo Wii Remote** devices [19] are used to be able to control the game.

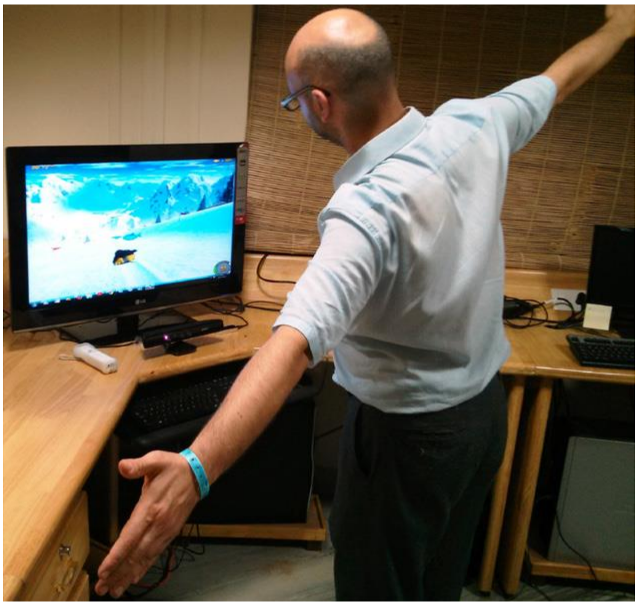


Figure 2.5 – Avatar control in Tux Racer game using sensors [17]

The 3D sensors have been identified as an effective and efficient solution for the rehabilitation area because of its potential on patient motion monitoring. We can divide the sensors into two categories:

- **Wearable** – Sensors that need to be connected to some part of the human body;
- **Non-Wearable** – Sensors that don't need to be attached to any part of the body, being placed somewhere so that it's possible to identify patient-related movements and collect data.

In [20], they note that the development of technologies over the last decade have facilitated the deployment of sensors on the level of the equipment for clinical applications, including rehabilitation. The interest of researchers and clinicians on sensorized equipments has led to a change in the field of usable technology, where new developments and designs of sensors and systems are highlighted. The advance in the field of microelectronics allows the development of small circuits and the possibility of adding many sensors to it, such as gyroscopes or accelerometers (see Figure 2.6).

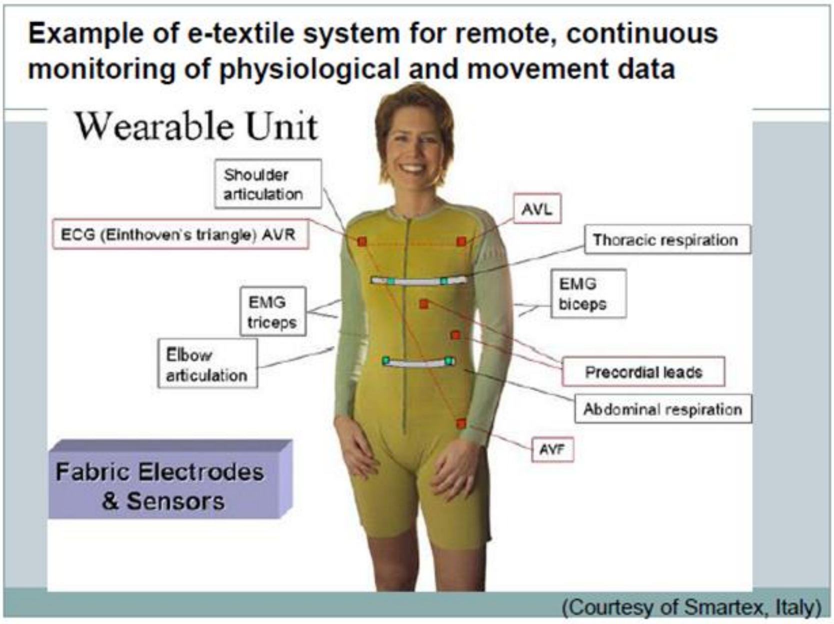

Example of e-textile system for remote, continuous monitoring of physiological and movement data
Wearable Unit
Shoulder articulation
ECG (Einthoven's triangle) AVR
AVL
Thoracic respiration
EMG triceps
EMG biceps
Elbow articulation
Precordial leads
Abdominal respiration
AVF
Fabric Electrodes & Sensors
(Courtesy of Smartex, Italy)

Figure 2.6 – Example of e-textile system for remote, continuous monitoring of physiological and movement data [20]

Sensors such as previously mentioned, for example, Microsoft Kinect, show that they are useful as tools of physical rehabilitation [17]. The fun that the games propose combined with sensing technologies, make possible the development of new training systems for physical therapy, by the psychological motivation of the patient and the extension of the training time with high quality of the performed motion in performing exercises repeatedly. The disadvantage of some motion sensors is that users need to attach them to body limbs or hold them in their hands to detect movements, which can cause discomfort and inconvenience to those who use them. Strategies involving the use of various technologies for people with motor disabilities have been developed for rehabilitation through numerous implementation, VR and serious motion-based games have been used recently for rehabilitation. Exercises range from simple goal-guided movements [21] to learning in the execution of daily tasks.

In the market, there are several motion sensors and it's necessary to analyze and identify the advantages that can bring to the LeaPhysio system, to suit your needs. Some of these sensors are:

- **Nintendo Wii Remote** (see Figure 2.7): It has been proven to have great success among different age groups when applied in rehabilitation [22]. This device has a built-in accelerometer offering position data (X, Y, Z) and a high-resolution infrared sensor in which it provides data on rotations (Yaw, Pitch, Roll). It doesn't give great precision in the detection of movements [17] [19].

Figure 2.7 – Nintendo Wii Remote

- **Microsoft Kinect V2** (see Figure 2.8): It's a video game interface with a detection of the human body. It detects changes in infrared light patterns, the coordinates of the body joints being detected. The Software Development Kit (SDK) from Microsoft provides 3D positions for 25 identified and predefined points in the human body. Kinect V1 only detects 20 points [17]. However, the spatial resolution of Kinect limits the accuracy of measuring small segment movements [23], such as fingers.

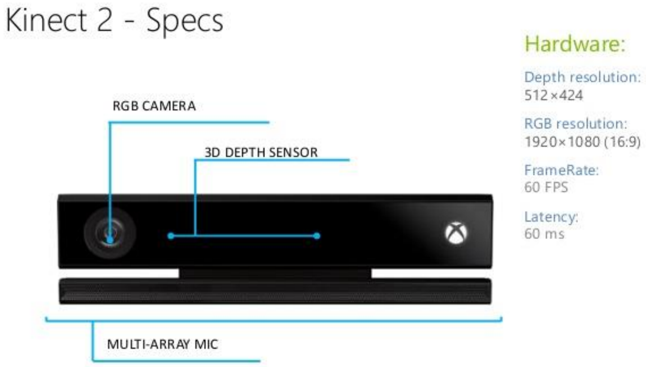


Figure 2.8 – Microsoft Kinect V2

- **Leap Motion** (see Figure 2.9): Provides high resolution in detecting small movements. It has been specially designed for hand and finger movements.

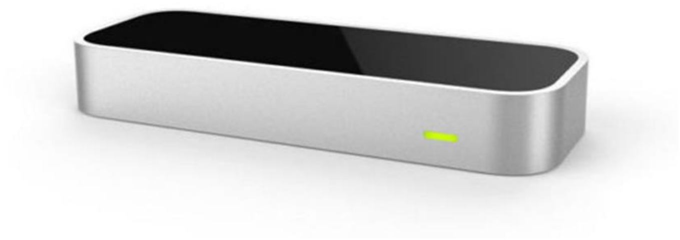


Figure 2.9 – Leap Motion Device

Since the LeaPhysio system consists of the rehabilitation of hands and fingers, there is a need to have a device that dispenses more attention to this part of the human body, the Leap Motion controller is the most viable solution.

Serious games can be developed using different technologies, such as JAVA, Unity, Blender and C#. Unity [24] is a multiplatform game engine developed by Unity Technologies and used for the development of video games for computers, consoles, mobile devices and websites using languages such as C++ and C#. The Unity is platform that is increasingly used by programmers to develop games due to its powerful game engine.

In [25], Nuno Duarte report a developed a system designed for physical rehabilitation that can be performed in clinics or at home, allowing the physiotherapist to monitor the progress of physical rehabilitation of patients. The system is divided into three components: A mobile application for Android OS devices; An Application Programming Interface (API) and a 3D video game. The last-mentioned component is developed in Unity 3D based on real life conditions along with virtual reality, supported by the Microsoft Kinect sensor for data collection. The 3D video game, can be adapted to the clinical needs of the patient (see Figure 2.10).

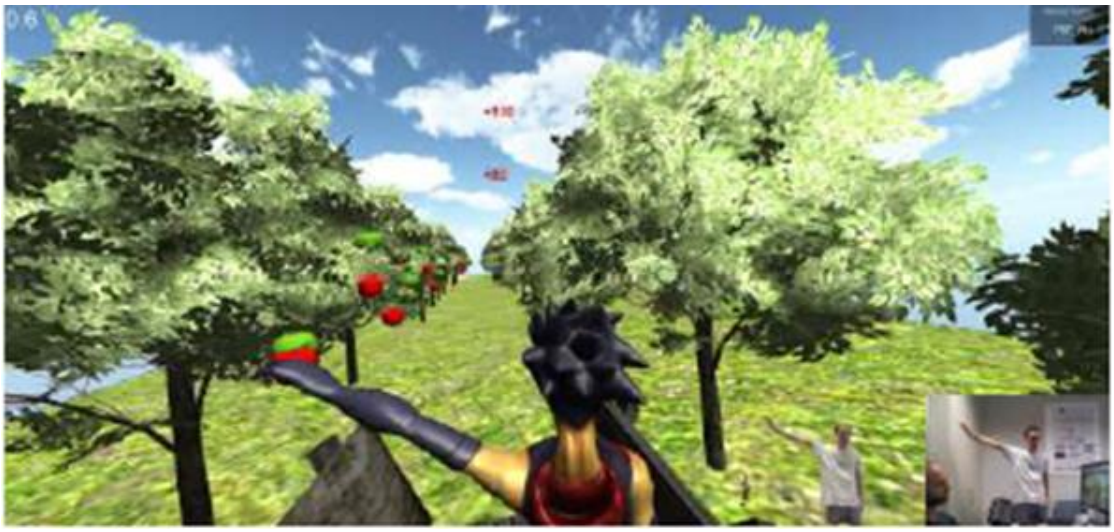


Figure 2.10 – Serious game developed in Unity 3D [25]

## 2.2. Information Storage

In healthcare system, appear increasingly needs to store the data in a DB (Database) allowing a certain amount of structured data to be collected, according with the application.

We can define the implementation of a DB in two aspects: local or remote. The local DB is when it's inserted in the device itself and remote when it's located, for example, on a server. In the case of a mobile device that uses the mobile operating system, such as Android OS, it can be used to store data from applications or even to save records created by the application. Currently smartphones contain reasonable amounts of internal memory, but there are certain applications that require a large amount of storage using remote DB's. Remote servers allow multiple devices to access data in real time, requiring an Internet connection. It's necessary to evaluate the system requirements to determine the use of a local or remote DB. Some examples of software technologies for database are:

- **SQLite** [26]: Is an Open Source service allows to create a database and incorporate in several systems. It's used on thousands of systems and uses Structured Query Language (SQL) syntax. This service is widely used in mobile applications with respect to the local database.
- **MySQL** [27]: Is a DBMS (DBMS) and uses SQL as an interface. Its features such as portability, compatibility, low demands on hardware features, ease of use among others, makes it one of the most popular in the world. Is also used in many high-profile, large-scale websites, including Facebook, Twitter and YouTube. MySQL is a central component of LAMP open-source web application software stack. LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python".

Sometimes databases store sensitive and personal data, such as contacts, addresses, names, e-mails, this information must be encrypted for security and privacy, whether at the company level or at the users.

Using a database located on a server, web services and other external services to save resources on mobile devices, open new doors for the development of distributed mobile applications, thus making an opportunity for the medical sciences, namely in the field of physiotherapy and rehabilitation. The possibility for professionals in the field to have a mobile application to consult information stored in the remote database or even to use external services dedicated to the area in question, becomes an auxiliary tool for rehabilitation consultations.

## 2.3. Android Mobile Applications

Android is a Linux-based Open Source Operating System (OS) currently being developed by Google and designed for mobile devices such as smartphones or tablets. It’s one of the most widely used mobile operating systems in the world. According to [28], in May 2017 there were 2 billion active Android OS users (values revealed by Google CEO Sundar Pichai).

Being open source software and having lots of people to use, it has stimulated a large community of developers to develop applications to meet the needs imposed by the market. There is free software for the development of applications: Android Studio, Integrated Development Environment (IDE) official; Dedicated SDK; Android Developer Tools (ADT). More and more companies develop applications for this type of platform, its low cost and possibility of customization is one of the main reasons. The Android platform has numerous advantages about applications in the medical sciences [29]. Some examples of medical applications (APP) available on Google Play:

- **Pedometer** (see Figure 2.11): “*Pedometer records the number of steps you took and displays them again along with the number of calories you burned, distance traveled, time spent, and speed per hour.*” [30];

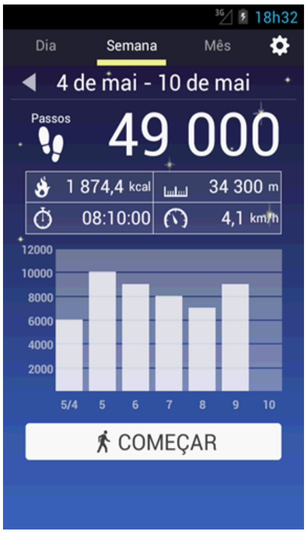


Figure 2.11 – Pedometer APP

- **Runtastic** (see Figure 2.12): "*Runtastic uses GPS to map and track sports and fitness activities such as running, jogging, biking and walking. The app rated at 4.5 tracks the progress of cardio exercises (time, distance, altitude change, calories and more) to help you create healthy lifestyle habits and achieve exercise goals.*" [31];

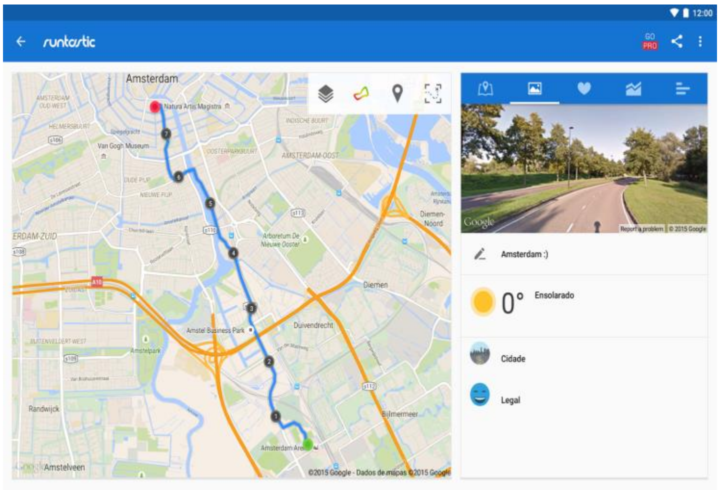


Figure 2.12 – Runtastic APP

- **Google Fit** (see Figure 2.13): "*Monitor your activity effortlessly. While walking, jogging or cycling throughout the day, the Android Wear mobile phone or watch automatically records your activity with Google Fit. Get immediate statistics. View real-time statistics of races, walks and bike rides. Google Fit records speed, pace, route, elevation, and more, so you can stay motivated and on track.*" [32].

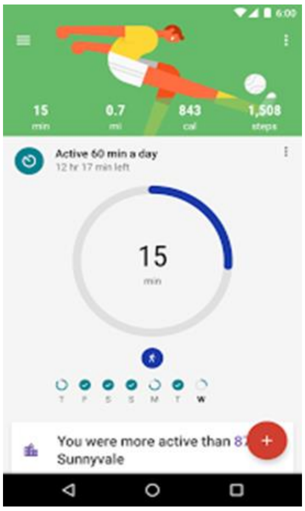
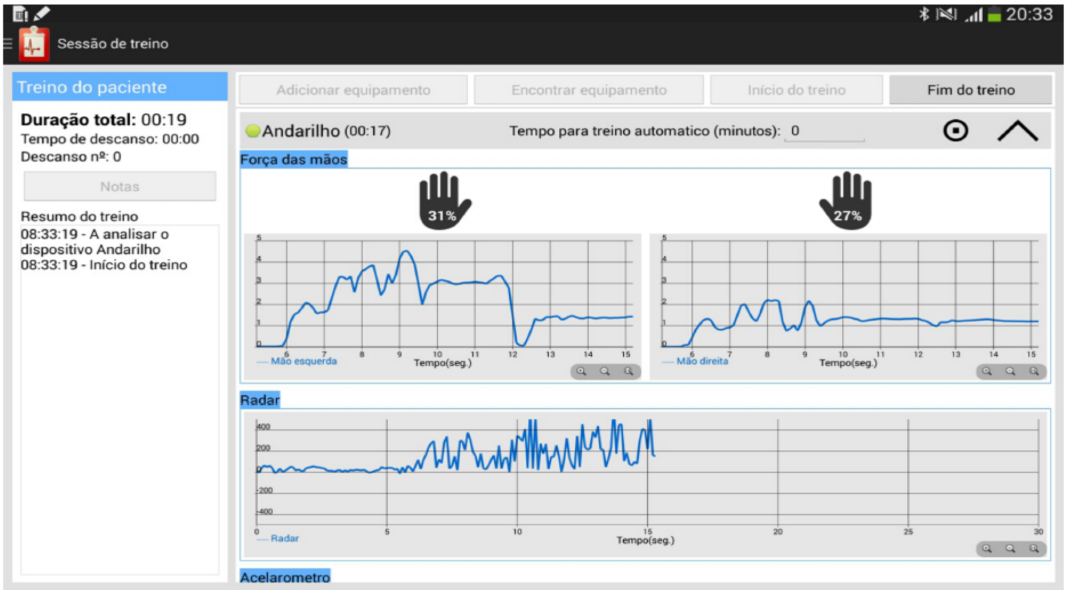


Figure 2.13 – Google Fit APP

The above applications are available to be installed by any user with an Android device.

In [33], Rúben Costa developed an application aimed at mobile devices to be used by professionals in the field of physiotherapy during the rehabilitation sessions with patients. This application is designed to allow communication with an intelligent walker composed of several types of sensors capable to extract information about force and motion during the physiotherapy sessions performed by the patient, where the information is transmitted in real time from the acquisition primary processing and Bluetooth communication module to a mobile device where additional signal processing is performed and the results are presented in personalized way to the health professional (see Figure 2.14).



Duração total: 00:19
Tempo de descanso: 00:00
Descanso nº: 0

Resumo do treino
08:33:19 - A analisar o dispositivo Andarilho
08:33:19 - Início do treino

Andarilho (00:17)
Tempo para treino automatico (minutos): 0

Figure 2.14 – mHealth APP [33]

In [34], the authors used the sensors available on Android devices to develop an application for wrist rehabilitation. With access from the device to the Internet it’s possible for the therapist to set up the exercises that he considers appropriate to the patient, the results are sent back to the therapist for this evaluation. The work also indicates that the system should self-adapt to the patient. It’s possible to create medical aid applications according to the needs of the system and that is suitable for the medium in question.

# Chapter 3 – System Description

## 3. Overview

The LeaPhysio system is a set of tools developed for the rehabilitation of hands and fingers for patients with motor activity limitations, allowing the physiotherapist to have access to the results obtained from the serious games performed by patients. Figure 3.1 illustrates the system architecture that is divided into three blocks.

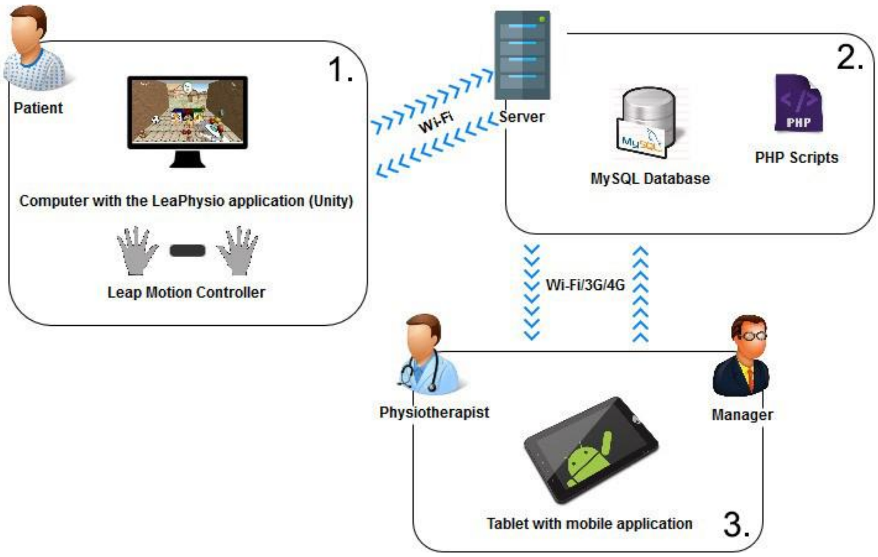


Figure 3.1 – LeaPhysio System Architecture

The first block (block "1") is dedicated to the patient allowing the access to a set of serious games characterized by natural interaction using LMC. The second block (block "2") it's mainly represented by the data server that stores the game results and game settings. The software technologies used on the server side are MySQL and PHP that allows the communication between applications. The last block (block "3") is expressed by a mobile application associated with patient data management and the health professionals data management. This chapter presents aspects related the operations associated with the LeaPhysio System blocks and the activities associated to the system users.

### 3.1. Users and Applications

The system has three different types of users: the *Manager* and *Physiotherapist* that interact with the system through mobile application that is used to manage the available resources and the Patient that uses the computer application (expressed by a serious game platform) to perform the training session imposed by the physiotherapists as part of the patient training plan.

Two types of applications were developed as parts of the LeaPhysio system:

- **Computer Application**: Developed in Unity that serves as a serious game platform for *Patients* to perform their training. The serious game platform provides a set of games, highly interactive and configurable according with the needs of patients. With the LMC, it's possible to replicate the movement of hands in the application, with a real-world interaction;
- **Mobile Application**: Developed for Android OS tablets that serves as a working tool for *Physiotherapists* that allows to manage different physiotherapy sessions and visualize results. The *Manager* also uses this application to add new physiotherapists to the system.

Both applications require Internet connection to perform database communication, either for data storage in a remote database

Additional details are mentioned for the system users:

- **Patient**: Is the main user of the LeaPhysio system, and in this case, can be mentioned that the developed system mainly corresponds to the patient under physical rehabilitation needs. The patient represents a single user of the computer application and he is the only one who performs the training created by the user's physiotherapist;
- **Physiotherapist**: Supervise the patients training and impose the training plan. Using the mobile application, the physiotherapists insert the patient data and create the training plan and analyze the training results;
- **Manager**: Presents the rights to manage the physiotherapists data using application.

## 3.2. LeaPhysio System

Figure 3.2 illustrates a time flow of the use of the LeaPhysio system by its users. The letters of A-H indicate the execution steps. The first step on system usage is carried out by *Manager* that introduce *Physiotherapists* data into the system using the mobile application. The registered *Physiotherapist* through the mobile application, can introduce the *Patients* data in the system and to include the training plan according to the patient needs. The Patient will executed the prescribed plan including serious game session using the computer application.

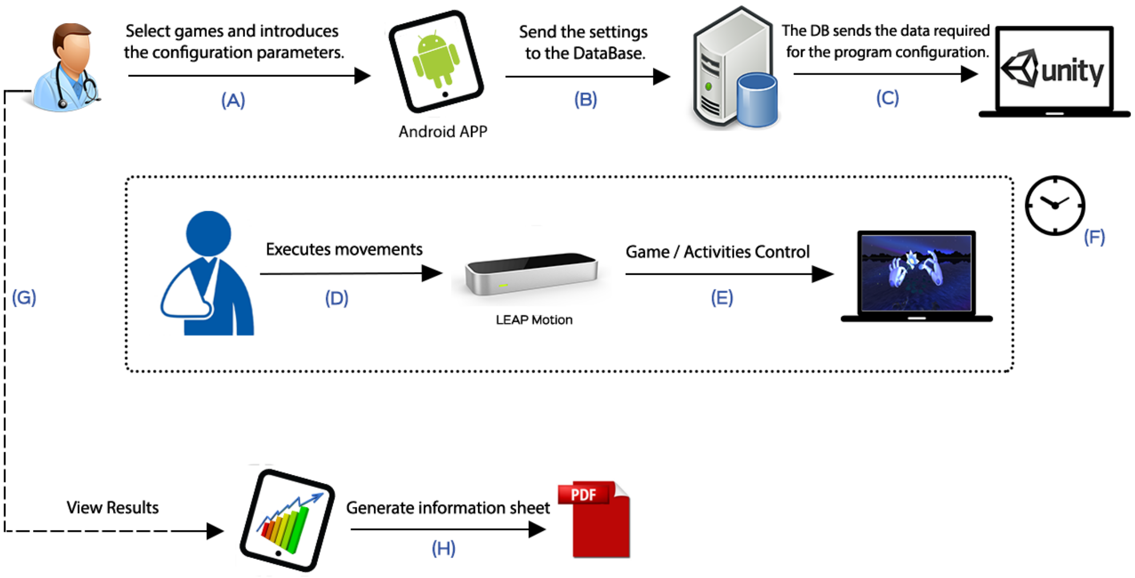

Select games and introduces the configuration parameters.
(A)
Android APP
Send the settings to the DataBase.
(B)
The DB sends the data required for the program configuration.
(C)
unity
Executes movements
(D)
LEAP Motion
Game / Activities Control
(E)
(F)
(G)
View Results
Generate information sheet
(H)
PDF

Figure 3.2 – Diagram with time flow of the LeaPhysio System

When a physiotherapist creates a new patient in the mobile application, it's recorded that the physiotherapist is responsible for creating training plans to patient, but what doesn't prevent other physiotherapists in the future to create training for this patient, being only responsible for that training, because in the absence of the physiotherapist responsible due to any circumstance there may be another physiotherapist to accompany the patient. Concluding the patients may be shared between different physiotherapists in the same clinic.

After insertion of a new patient into the system, the patient receives a LeaPhysio card that includes a QRCode. Figure 3.3 shows an example of the card, which allows the patient to log in to the computer application, showing the QRCode of the available Webcam. The use of a QRCode system was considered against others identification technologies such NFC or RFID considering the costs and a capability of this technology to be easily, interconnect with the Unity application.

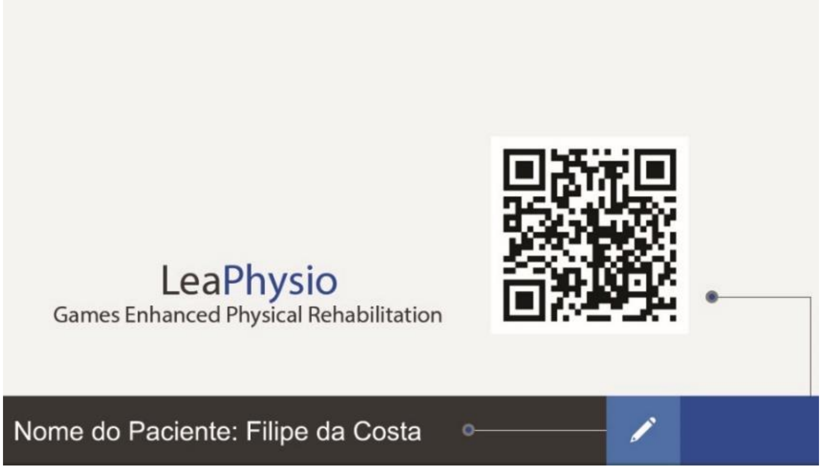

LeaPhysio
Games Enhanced Physical Rehabilitation
Nome do Paciente: Filipe da Costa

Figure 3.3 – LeaPhysio Card

After a patient is created in the system, the responsibility of the physiotherapist is to analyze the clinical status and to create an individualized training plan. The training plan setup involves the following parameters (***A***):

- **Initial Date**: Date for the patient to start the training. The initial date must be equal to or greater than the creation date;
- **Final Date**: This date is the last day (inclusive) in which the patient can perform the training. The final date must be equal to or greater than the initial date;
- **Select Game**: What is the serious game that a patient should perform;
- **Select Game Theme**: Thematic of the chosen serious game;
- **Duration of the Game**: Game time;
- **Rest time**: Interval time between each game;
- **Number of repetitions per training**: Number of times the patient should repeat the game;
- **Select hands**: The hand or hands that the patient should use (LEFT, RIGHT, BOTH);
- **Difficulty of the Game**: Level of difficulty of the game (EASY, MEDIUM, HARD);
- **Description**: Training note;

When a training is created for a patient, the mobile application sends the data to the remote database (***B***). Afterwards, the patient should access the computer application presenting the QRCode to the webcam, that will read and check the QRCode and check in the DB. If the content belongs to a patient, the patient has successfully logged into the application. The LMC needs to be connected to the computer where the application is installed so the patient can perform the training.

The computer application initiates an HTTP request through the corresponding PHP script hosted on the server so that it contacts the MySQL DB and sends the necessary data to the computer application according to the patient who is using it, using an HTTP response (***C***). If the patient had training and these trainings are between the initial and final training dates, the patient can perform the serious games configured with the parameters of the training in question (***D***). The application developed in Unity uses LMC to replicate hands and their movements in the game scenario (***E***). Each training can have a repetition or more, it means that if a certain training has, for example, two repetitions, the patient performs once the serious game with its duration, after completing, has a rest period until the serious game is executed again (***F***) and after completing his training session ends. A patient may also have more than one active training at the same time, in which case the training repetitions are intercalated, for example, two training (training X and training Y) and both with two repetitions. Initially the patient performs the game X-1 (game corresponding to training X in the first repetition), has the rest time of training Y and after this time performs the game Y-1, after rest time of training X, performs the game X-2, then rest time Y, run the Y-2 game and end the training session.

After finishing the training, the physiotherapist can use the mobile application to change any parameter in the patient's training if necessary and visualize the results obtained (***G***). The results can be of two types, results in terms of scores and results in terms of coordinates. The scores indicate, for example, how many objects the patient picked up during the game, while the coordinates refer to the hand points that the patient used during the training. The coordinates refer to values related to positions, rotations and speeds of hands and fingers. Finally, in case the physiotherapist deems it necessary, he can generate a report with information about the patient (***H***), of the exercises already performed and their respective scores. These reports can be generated for the personal use of the physiotherapist or may be delivered to the patient to the inform about his performances during the training.

The execution steps (see Figure 3.2) is repeated several times, either with other trainings or configurations. It’s important that there is a follow-up on the part of the physiotherapist, either at an early phase to explain how serious games are performed or at a later phase to assess the clinical state and evolution of the patient in performing hand movements. The following two subchapters provide information about the hardware components of the implemented system.

## 3.3. Leap Motion Controller

The LMC (see Figure 3.4) [6] is a small device (13 mm x 13 mm x 76 mm) that weights 45g. It's dual platform (Macintosh/Windows), connects to a computer via USB 3.0 connection and easy to install. The sensor is activated and ready to use when the red lights appear on the top of the device. It can detect the movement of the hands when they are on the device's field of vision, the movements being recognized by the device and translated into actions associated with the implemented VR. The LMC streams data at a variable acquisition rate of up to 120Hz.

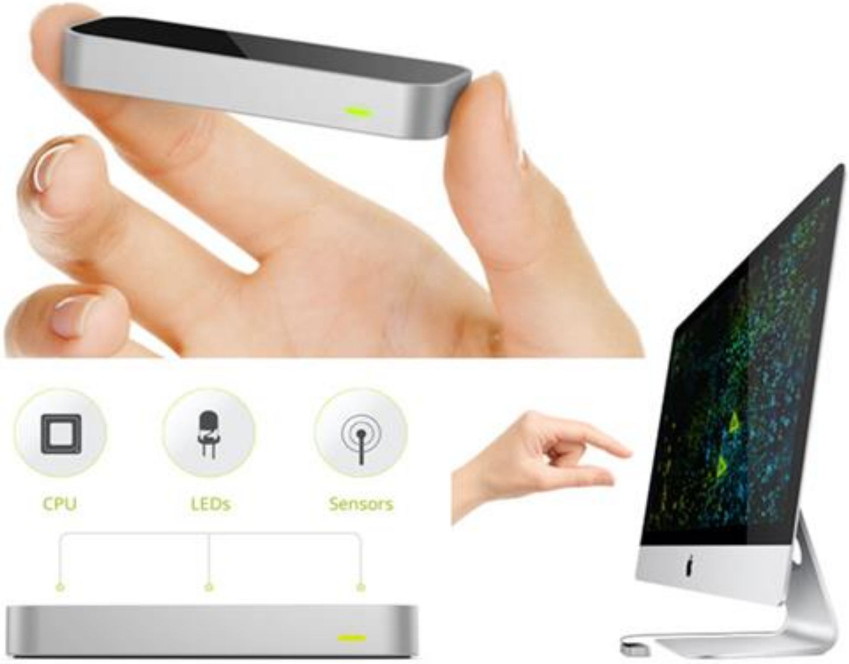


Figure 3.4 – Leap Motion Controller

The device is highly sensitive to hand small amplitude movements, due to its ability to map the entire zone over it, being able to detect small movements of the whole hand. The controller senses the upper zone and is sensitive to a range of about one meter (see Figure 3.5). It can simultaneously capture the movement of all fingers, with a sensitivity of 1/1000 mm.

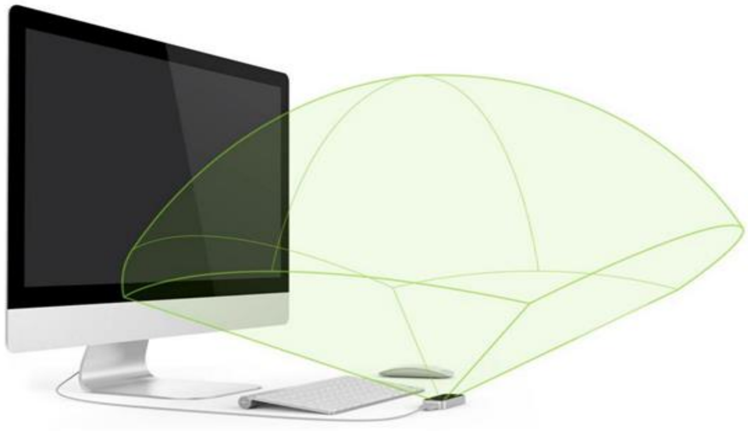


Figure 3.5 – Leap Motion interaction zone

It was launched in July 2013 [35] and presents a new way of interacting with the technology presented. It has an APP Store, known as AirSpace [36], in November 2015 the store contains about 230 applications, in which about 45% of applications are free and the rest paid. The applications cover a wide range of areas, games, educational applications, applications dedicated to music among other categories, and there is a growing need for applications dedicated to the field of physiotherapy. According to company reports, the controller uses two monochrome cameras and three IR (infrared) LEDs (wavelengths = 850nm) to rebuild the 3D scene and track the positions of the hands and fingers.

To perform the integration in the system the device SDK was considered. It provides data access through direct hand and finger mapping detection without the need to interpret detection data to locate hands and fingers, making this aspect a great potential in the development of new applications. The low cost of the device (about € 80), a simple setup along with its accuracy, indicates its potential for the evaluation of movement in a 3D space.

Independent evaluations used an industrial robotic arm with a point pen, concluding that the mean square error between the position of the point pen and the position measured by the Leap Motion system was < 0.2 mm in static tests and < 2.5 mm in dynamic tests [37]. In another evaluation, a plastic arm model was used, where it was found that the Leap Motion system was accurate to <0.5 mm in statistical conditions [38]. In addition, a recent comparison reported a considerable edge in hopping latency (850 ms) compared to Kinect (170 ms) [39].

Christianne Falcao, Ana Catarina Lemos and Marcelo Soares [40] carried out a study regarding the evaluation of the natural UI of LMC. Graphic designers with experience using Photoshop CS6 software who have never had contact with LMC participated in this study.

Participants were asked to perform certain tasks in Photoshop CS6 using LMC to be able to answer certain questions proposed by the team. At the end of the study they concluded that LMC needs adjustments in its gestural interface and in the analysis of hand movements to minimize fatigue and provide a better experience. They indicate that this is a very promising technology, but it's not ready to replace the traditional mouse and keyboard in the use of graphic software.

This technology holds great promise for the field of rehabilitation since it doesn't require the patient to wear gloves, which makes it an important factor in the therapy of patients who have undergone hand surgery. The device provides a new way of interacting between the user and the computer, the ability to bring the real world to the digital domain. The main advantages of this sensor compared to other motion sensors, also applied to rehabilitation are:

- Provides natural user interfacing and through remote sensing (no need for the device to be attached to person's body, which can cause discomfort on certain occasions);
- The SDK provided for the development of applications;
- Price.

Figure 3.6 illustrates the use of the LMC by a typical user.

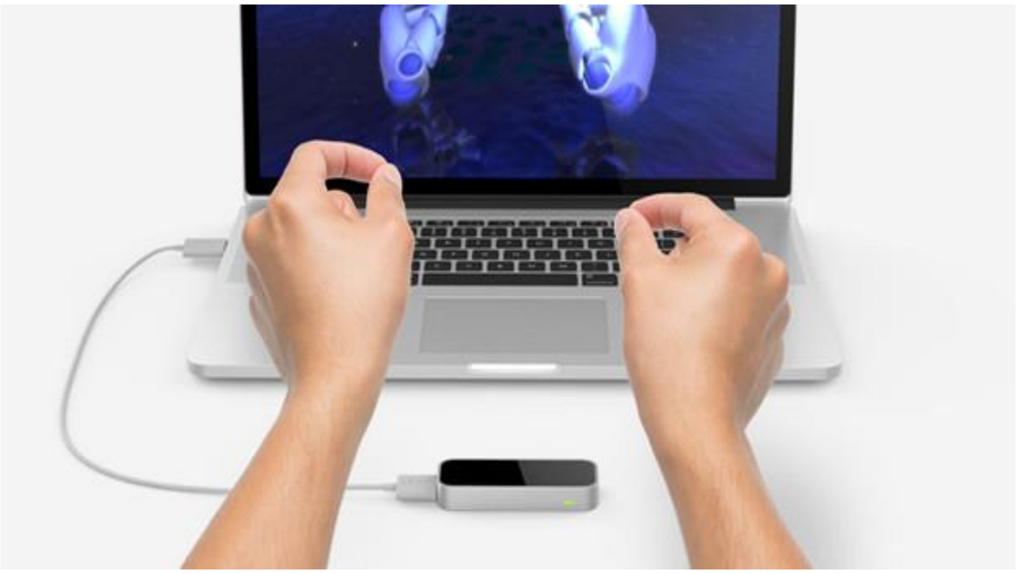


Figure 3.6 – Common usage of LMC by a user

In [41], a study was carried out on the use of LMC in the hand rehabilitation process. In the tests performed, an adjustable plate was used to place the LMC (see Figure 3.7).

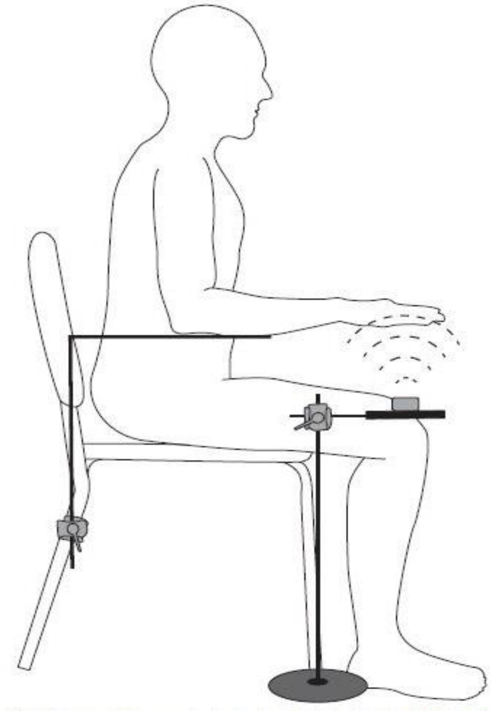


Figure 3.7 – Flexible size structure to place the Leap Motion Controller

It consists of a platform that is supported on the floor along with an adjustable plate, where the LMC is mounted. By adjusting the height of the LMC through the board, it allows it to be under the user's hands so that the controller captures the movements of the hands. This structure against the typical use of the LMC, allows the user to do no extra effort, not needing to raise the arms at a certain time for the LMC to detect the hands, thus avoiding wear and tear on the part of the limbs. For this reason, this type of structure was used for the LeaPhysio system.

## 3.4. Others Hardware Components

In addition to the LMC, there are other hardware devices that are required to use the LeaPhysio applications:

- Computer and Webcam;
- Server;
- Tablet with Android OS.

**Computer and Webcam**

The Unity application is designed for computers running the Windows OS. For its development and testing a portable computer with the following specifications was used:

| ASUS K555L | |
|---|---|
| **Windows Edition** | Windows 10 Home |
| **Screen inches** | 15.2 |
| **Processor** | Intel(R) Core(TM) i7-4510U @ 2.00GHz |
| **RAM Memory** | 8,00 GB |
| **Type of system** | 64-bit operating system, x64-based processor |
| **Video card** | NVIDIA GeForce 840M |
| **Webcam** | Logitech HD Webcam C270 |

Table 1 – Specifications of the laptop used to test and develop the LeaPhysio System

Clinics should use computers that have similar specifications as above or higher, whether desktop or laptop computer. It's recommended that there be computers with a good quality image, resolution of 1280 × 720, processor and graphics card higher than those indicated in Table 1, so that it's possible for the patient to visualize all the components of the game. About Webcam (Logitech HD Webcam C270), it's an essential component for the application, because it's through this that the patient logs into the application, showing his QRCode that is embedded in his LeaPhysio card. There are monitors in the market that already comes with a built-in Webcam that can be used by the application, but since the users of the application are patients that are sometimes very limited in terms of movements, using the webcam available on the monitor can cause discomfort, either because the patient exerts an extra effort to raise the arm to show the QRCode or even because the monitor is away from where the patient is sitting. Ideally, an external webcam should be used with the lens turned from the bottom up and available in a frame to place the QRCode at the top and log in, as shown in Figure 3.8.

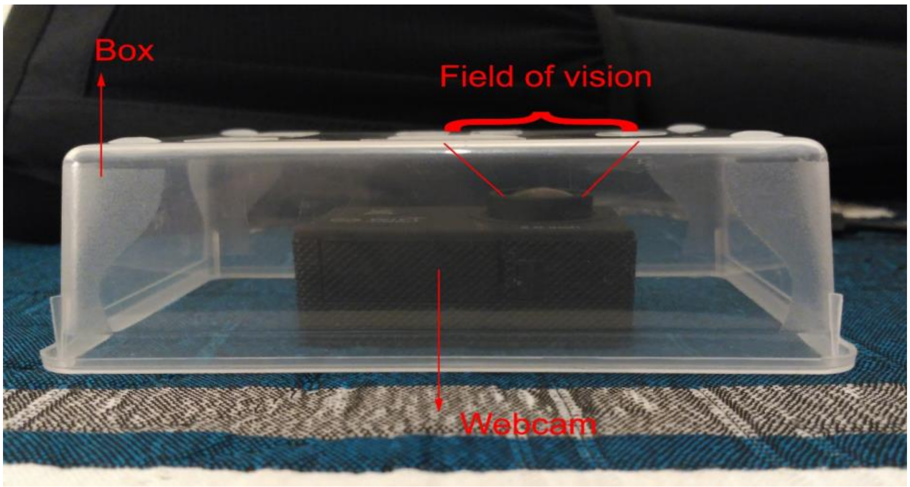


Figure 3.8 – Acrylic or plastic case for a Webcam

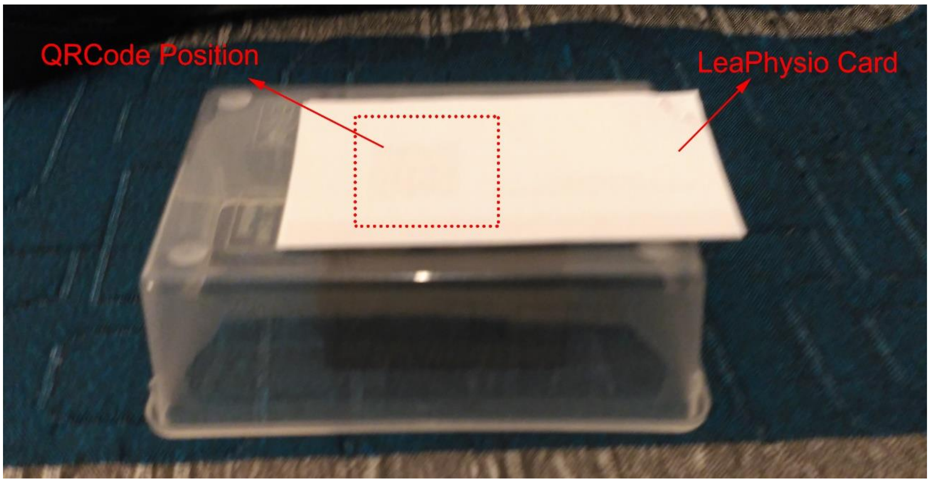


Figure 3.9 – Example of using Webcam structure with LeaPhysio card

The structure consists of an acrylic or plastic box in which the webcam is inside, with its field of view directed to the top of the box and which is connected to the computer via USB. The upper part of the box has signaled the place where the patient should place the QRCode on the LeaPhysio card. Figure 3.9 illustrates the structure as an example for using the LeaPhysio Card.

The monitor of the computer together with the box for the webcam must be housed on a table and LMC, as shown in Figure 3.10. On the other hand, the Leap Motion Controller must be placed on a stand that is lower than the table used for the monitor and webcam, as explained in the previous subchapter.

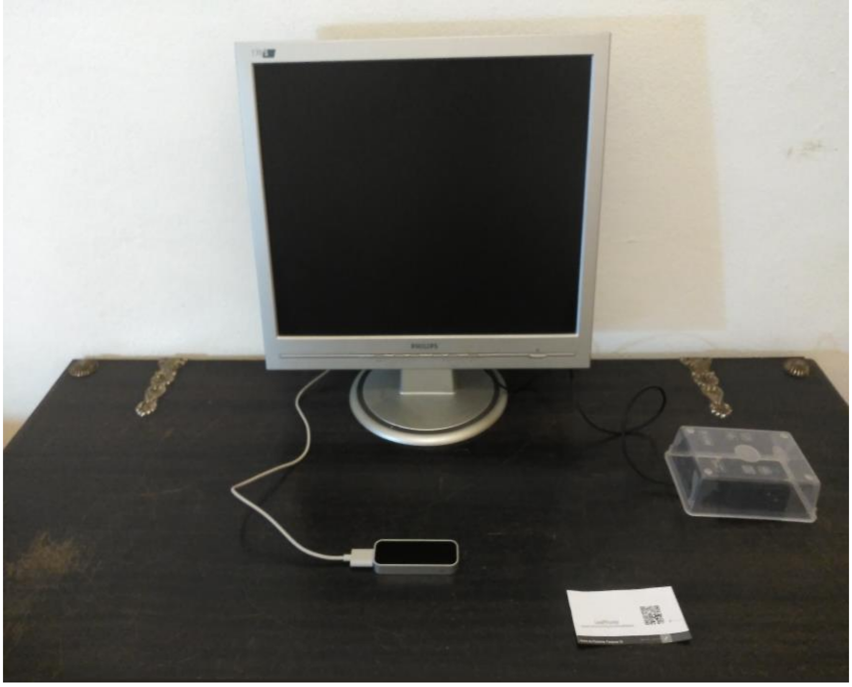


Figure 3.10 – Computer Monitor, LMC and Webcam platform on the desktop

**Server**

In this implementation was used a Virtual Private Server (VPS) consisting of a virtual machine. This server has its dedicated operating system and the user access rights, that allow him to install any software that is compatible with the operation system. The OS is Linux, where its distribution is *Debian 3.2.0-4-amd64*. On the server is hosted the MySQL DB, the Apache Web Service to perform HTTP communication between different applications, making use of PHP scripts. Also stored here are files of coordinates resulting from patient training, profile photos of various users and LeaPhysio cards.

**Android Tablet**

To use the mobile application, a device with the Android OS is required. In the developed LeaPhysio APP is possible visualize charts regarding the results of the training of the patients and due to this reason, the devices that must be used should be android tablets, allowing a better visualization of the information available to the user. The application was developed to run on tablets with version 4.0.4 (Ice Cream Sandwich) or higher, and its target is version 6.0 (Marshmallow).

Three devices were used for the development and testing of the mobile application, one with the minimum version supported by the application (Bq Maxwell Plus), one device with the target version of the application (Nexus 10) and another that was used during some tests (Sony Xperia Z2 Tablet). Figure 3.11 illustrates one of the two devices used, in which it has a seven-inch screen with a resolution of 1024x600 pixels, a total of 170 pixels per inch (ppi).

Figure 3.11 – Tablet Android Bq Maxwell Plus

Future devices used by physiotherapists and managers should be tablets with at least the minimum or higher version of Android OS. The application requires the device to have access to the internet, either through a mobile network (3G, 4G) or Wi-Fi.

# Chapter 4 – Serious Games for Physiotherapy

## 4. Overview

The term Serious Game refers to digital games dedicated to serious purposes that are based not only on entertainment and fun but also games that promote teaching, research, health, among others. This chapter introduces the computer application that was developed to be used by the patients, so that they can perform the training previously created by the physiotherapists. This chapter explores how the games were created and adapted to the individual characteristics of each patient.

### 4.1. Game Engine

The game engine is a key component when developing a video game. Is a computer program or a set of libraries, to simplify the development of video games or applications with real-time graphics. There are several game engines that can be used to develop video games in 2D or 3D, each one has certain features to help the developer in the development, for example, Unity 3D [24] and Unreal Engine [42]. After a critical analysis, was decided to use Unity 3D as a game engine, due to its great rendering capability and the features it offers. Regarding programming languages, it uses C# and JavaScript as primary languages, which makes programming easier because its syntax is like other programming languages that programmers are accustomed to using. Is a very well documented game engine with a powerful Asset Store [43] where it's possible to download scripts, objects, textures, among others to develop games.

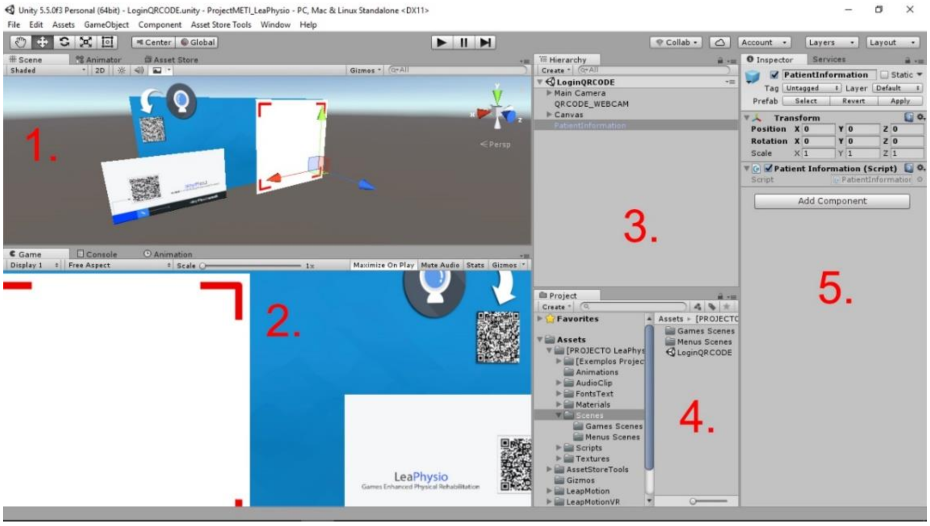
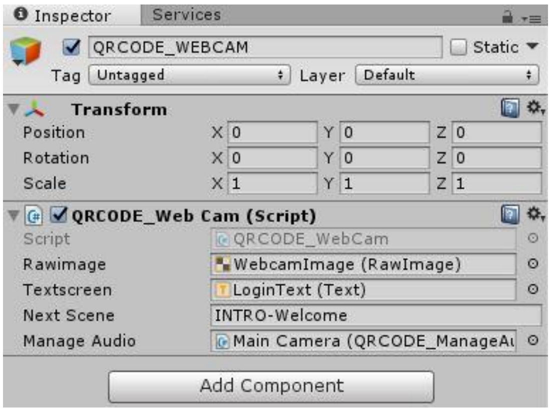

Unity 5.5.0f3 Personal (64bit) - LoginQRCODE.unity - ProjectMETI_LeaPhysio - PC, Mac & Linux Standalone <DX11>
File Edit Assets GameObject Component Asset Store Tools Window Help
Center Global
Collab Account Layers Layout
Scene Animator Asset Store
Hierarchy
LoginQRCODE
Main Camera
QRCODE_WEBCAM
Canvas
PatientInformation
Inspector Services
Transform
Position
Rotation
Scale
Patient Information (Script)
Add Component
Game Console Animation
Project
Favorites
Assets
1.
2.
3.
4.
5.

Figure 4.1 – Unity 3D Interface

Unity 3D offers a simple and intuitive interface to help the user to develop the games has a good integration with the LMC. It was some of the main arguments that led to the choice of Unity 3D as a game engine. The Unity 3D development interface can be personality and organized according to who is developing, as illustrated in Figure 4.1.

- **1. →** The Scene creator allows the programmer to interact with the objects that are present in the scenario. Allows perform rotations, translations, change its size and apply new textures;
- **2. →** When the programmer runs the game to see what has already developed is the area where you can see the scenario created in the perspective of the end user. It also has a tab named by Console that is used for debugging processes;
- **3. →** This area refers to existing objects in the scenario, where it's possible to remove or add new ones;
- **4. →** Structure with all the files belonging to the project;
- **5. →** Finally, we have the Inspector. For each object is possible to perform several operations, for example, associate scripts or create physics processes. Is very useful to visualize the various components associated with each object.

One of the main aspects is the interaction between the scenery and the objects it has. Unity 3D doesn't have any main script to control all the actions of the game, but several scripts that are behaviors and that can be associated with the various objects. After programming a script, this script must extend the *monobehaviour* class and it's ready to be associated with an object, it's only necessary to drag the script to the object, and this is associated as a component. Through the Inspector tool it's possible to view the script as a component of the object and if it has, for example, some public variables, is possible to visualize them and to define values for each of them, as illustrated in Figure 4.2.



☑ QRCODE_WEBCAM □ Static
Tag Untagged Layer Default

☑ Transform
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

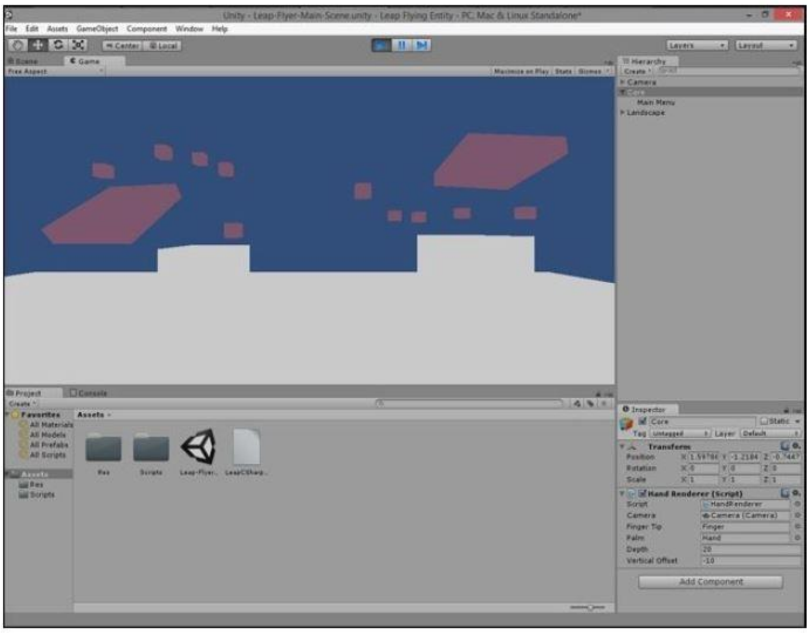
☑ QRCODE_Web Cam (Script)
Script QRCODE_WebCam
Rawimage WebcamImage (RawImage)
Textscreen LoginText (Text)
Next Scene INTRO-Welcome
Manage Audio Main Camera (QRCODE_ManageAu

Figure 4.2 – Example of a script with an object with its attributes and values

The interaction between objects can be performed in two different ways. The first consists of laws of physics, for example, two objects of equal size and different masses, the one with the greatest weight is thrown from a certain point to the position where the other object is, when colliding with the object that was standing the force exerted on it reacts and its position is altered, and there is a direct interaction between the objects. Another example can be an object that reacts to a certain light by changing its texture color. The second way is that when a given object performs an operation it sends a notification to another object so that it performs a certain function. There are different types of interactions that help make the scenario more real. The version used by Unity to develop the application was 5.5.0f3 (64 bit).

## 4.2. First tests

Initially it was necessary to understand how the communication between the Unity 3D game engine and the LMC works and for that, the book "*Mastering Leap Motion*" [44] was used, explaining the main classes of the Leap Motion API and how they can interact with Unity 3D. The book has a small tutorial, which teaches you step-by-step how to create a 3D application so that is possible to replicate hands in the game scenario, Figure 4.3 illustrates the result of the application.



Core
Tag Untagged Layer Default
Transform
Position X 1.59786 Y -1.2184 Z -0.7447
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Hand Renderer (Script)
Script HandRenderer
Camera Camera (Camera)
Finger Tip Finger
Palm Hand
Depth 20
Vertical Offset -10

Figure 4.3 – First test performed on Unity with Leap Motion controller

This tutorial was useful for understanding the interaction between the game engine and the LMC. Leap Motion Company provides a Unity 3D project that has several examples [45] to help the developer develop new applications. The project doesn't only provide examples, it also has different objects that represent hands and arms of different colors, gender and sizes, besides providing several scripts already programmed and ready to be used.

The Unity application developed for this project was built based on this sample project, to be able to take advantage of the resources provided. Figure 4.4 illustrates an example provided in the design, which consists of a room with different color boxes and sizes allowing the hands to interact with them, for example, grabbing cubes and dropping them.

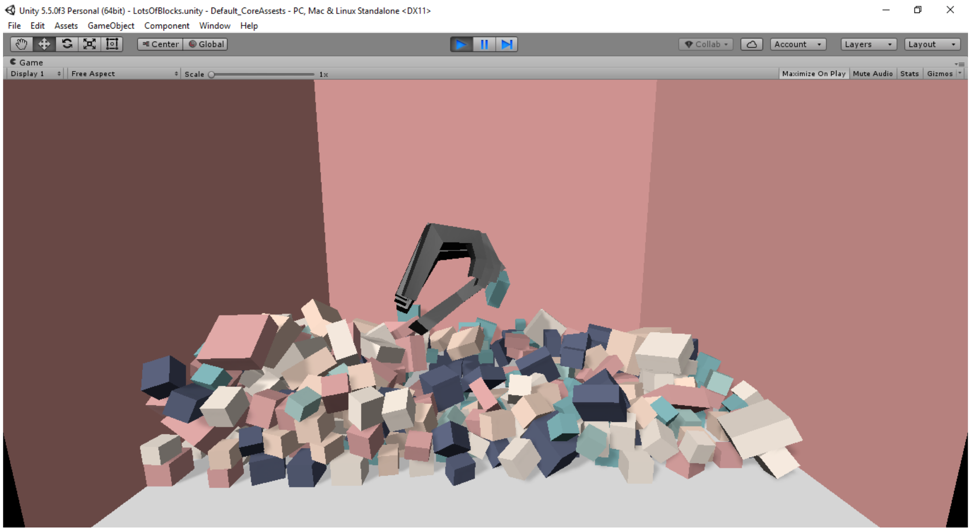


Figure 4.4 – Example of the LeapMotionCoreAssets project [45]

After exploring the examples provided, some tests were done to start developing the first game for the application. Figure 4.5 illustrates the first test to develop one of the future games of the application, using leap motion.

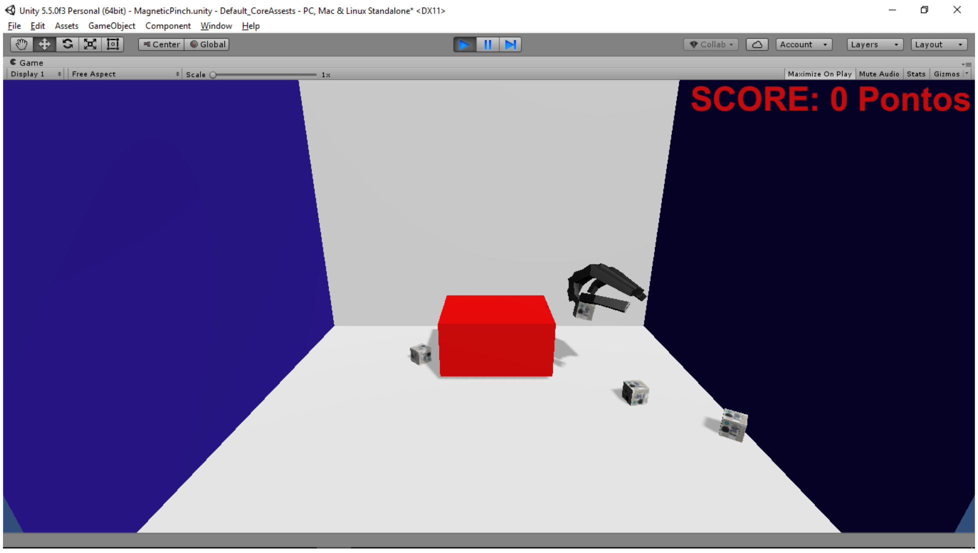


Figure 4.5 – First sketch of game development

The objective of the first test is the user can use both hands to grab the available cubes in the scenario and place them in the red box to gain points. A time counter was then added to the scenario to check how many cubes the user places in the box within a certain amount of time and a message is shown to the user indicating that he scored points when the cube is placed in the box. Figure 4.6 illustrates some scenarios created for use with the game that is intended to be developed.

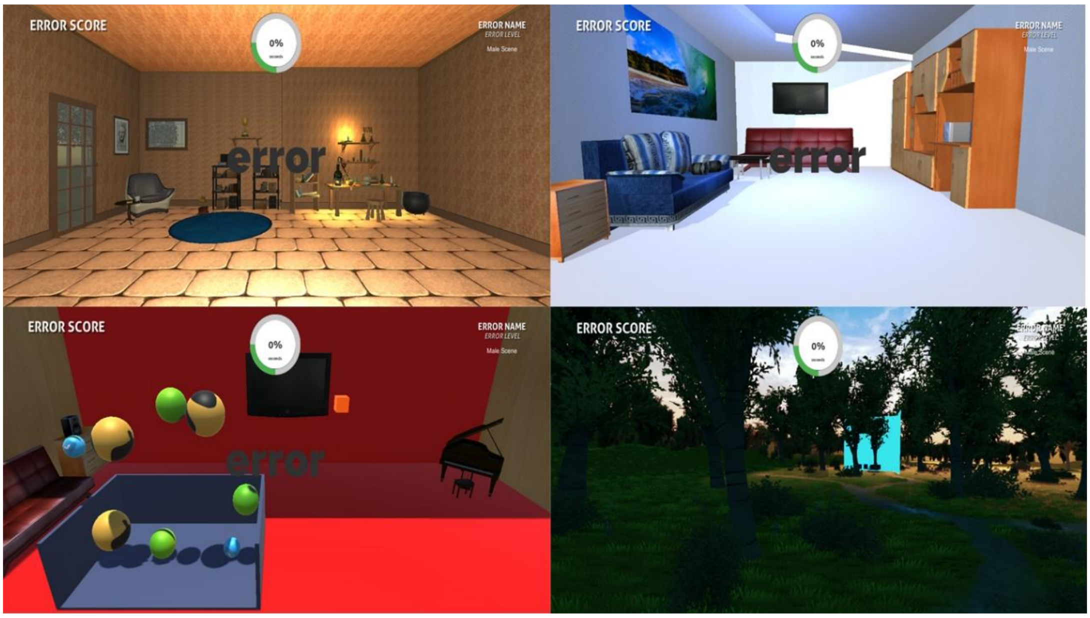


Figure 4.6 – Scenarios created in Unity

## 4.3. The Games

**CollectCube** and **PickaBall** game were developed for application. There are components that are similar in both games. The first concerns a countdown so the patient has time to prepare the hands to start playing. This time consists of five seconds for the patient to place their hands over LMC, the application will detect and replicate hands on the scene and their movements. During this time the patient cannot grasp any object, only after the time has elapsed and the playing time to start the count is that the patient is operative to grab objects. The second component is present at the top of each scenario, as shown in Figure 4.7.

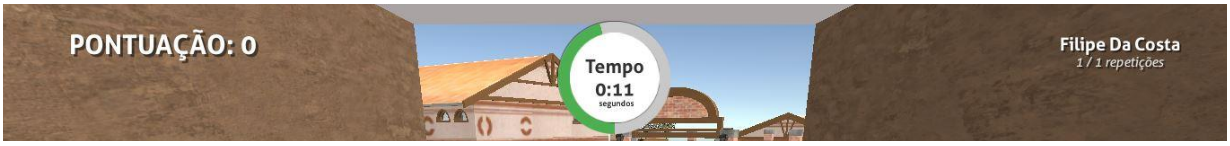


Figure 4.7 – Top component of game

In the upper left corner is the patient's score, in the center the clock with the remaining time for the game to finish and in the upper right corner the player's name and the number of the repetition of the training he is performing. Firstly, will explain the operation of the CollectCube game and subsequently the PickaBall game.

### 4.3.1. CollectCube game

The game aims is to pick up the cubes and to introduce according with the color in the corresponding box to earn points. During the game time cubes appear on the scene, for each correct cube the patient earns a point and for each failure loses a point. Figure 4.8 illustrates the CollectCube game interface.

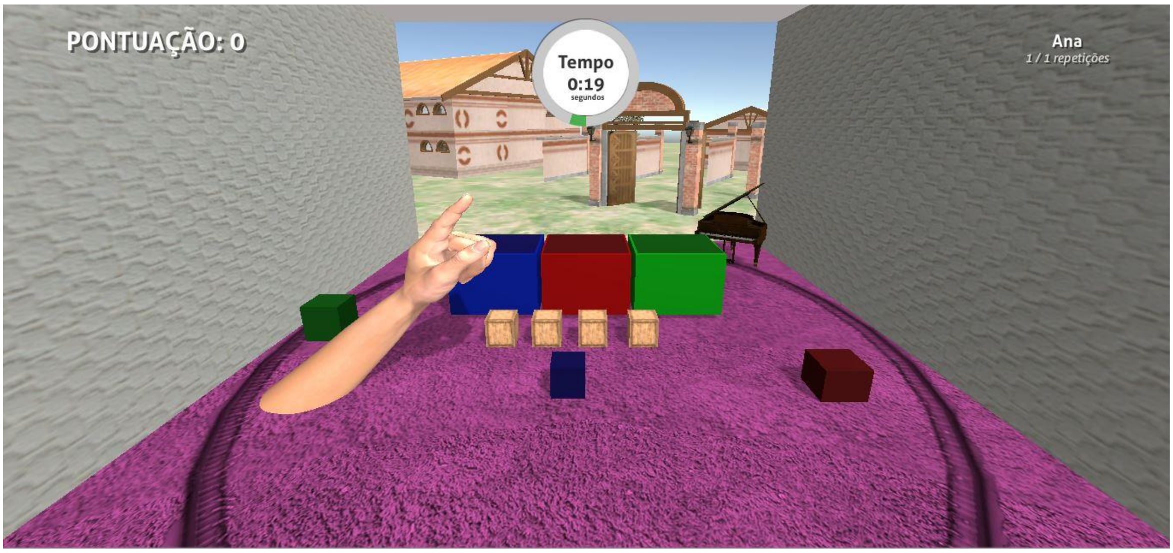


Figure 4.8 – CollectCube Game

This game has three levels of difficulty:

- **EASY**: The scenario only has a single box (green) and over time appear green cubes;
- **MEDIUM**: The scenario has two boxes (green and red) and over time randomly appear cubes with green or red color;
- **HARD**: The scenario has three boxes (green, red and blue) and over time randomly appear cubes with green, red or blue color.

In the difficulty level MEDIUM and HARD the boxes change places with each other in a pre-established period. This is for the patient to be aware of the game, always checking the position of the box when it grabs a cube. Figure 4.9 illustrates the three boxes and respective cubes.

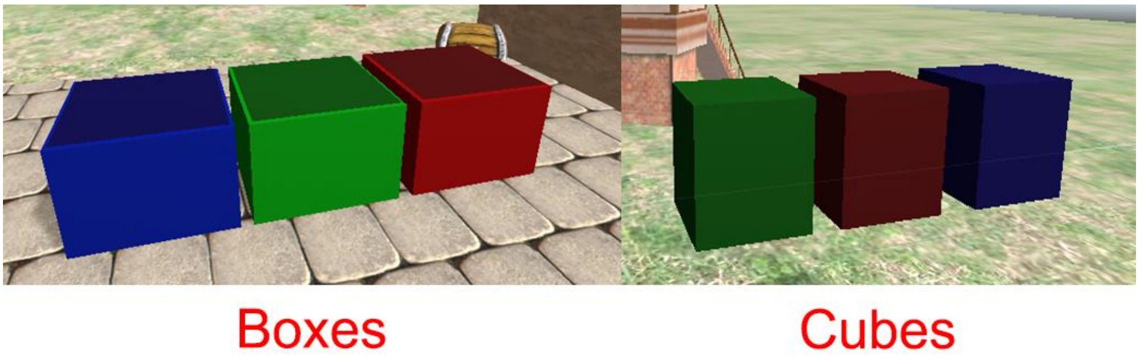


Figure 4.9 – Boxes and cubes of the CollectCube game

This game has two thematic, the COLOR thematic that is visible in figure 4.8 that was previously explained and the THEMES thematic. This thematic makes the boxes and cubes present in the COLOR thematic are replaced by other objects, but the goal of the game is the same. Figure 4.10 illustrates the game with THEMES thematic.

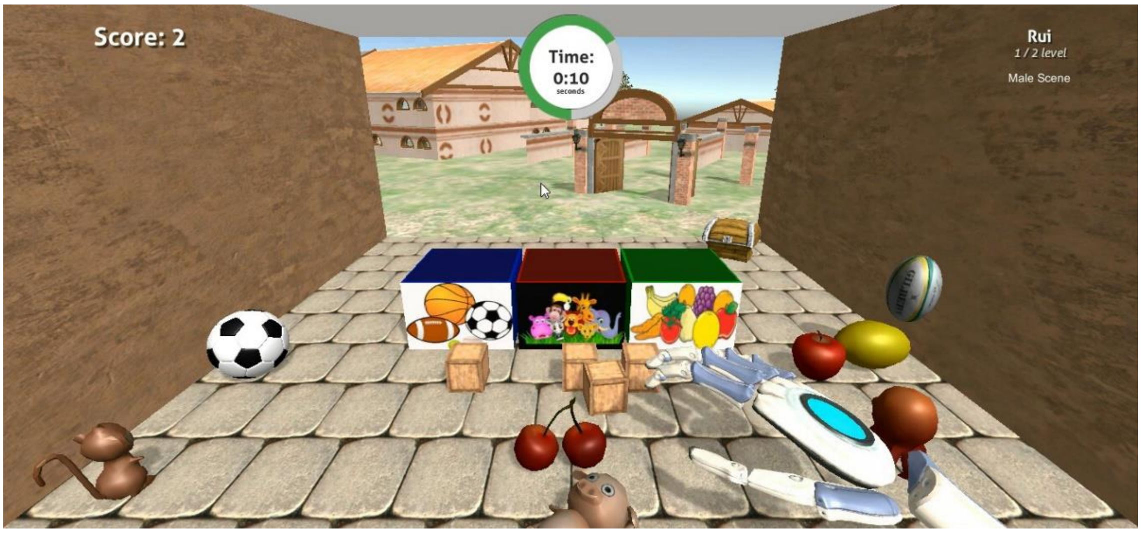


Figure 4.10 – CollectCube game with THEMES thematic

This thematic has objects and boxes with different categories. The objects can be of the type sport, fruit or animals. The patient should grab an object of a certain category and put in the corresponding box, for example, place the apple object in the fruit box. Many patients don’t only have problems with their hands and fingers, but many also have problems with cerebral palsy, which affects their ability to perform tasks. Through this thematic with categories that have objects known day-to-day, it can become useful to help patients with these problems, recognizing patterns to facilitate the execution of the proposed exercises. This thematic also has levels of difficulty with the characteristics explained above. Figure 4.11 illustrates the boxes and their objects used in the thematic.

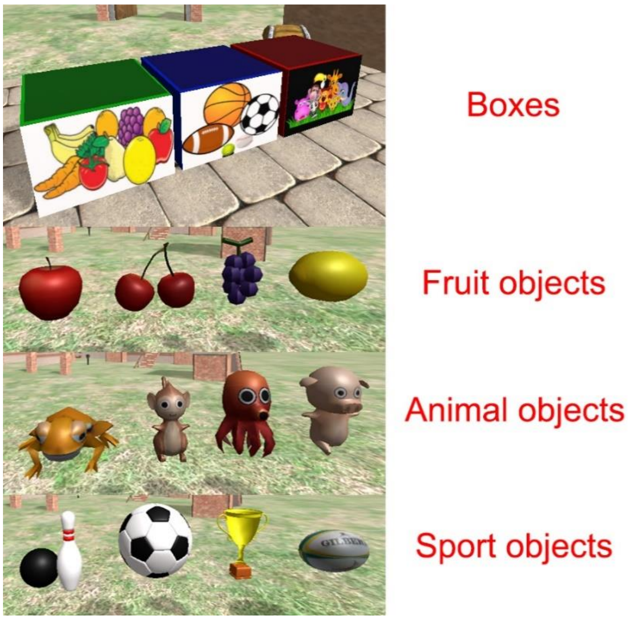


Figure 4.11 – Boxes and objects of the THEMES thematic

When the patient finishes the game, the results in terms of scores are stored in the database. Rooms different types of scores are stored:

- Game score;
- Total cubes or objects that appeared in the game;
- Total cubes or objects that have been correctly placed in their box;
- Total cubes or objects that haven't been correctly placed in their box.

### 4.3.2. PickaBall game

The game aims to get the patient to grab the available ball on the stage and pick up the cubes with the ball. The game created was an adaptation of an official Unity tutorial [46]. For each cube caught the patient gains a point A Figure 4.12 shows an example of the PickaBall game.

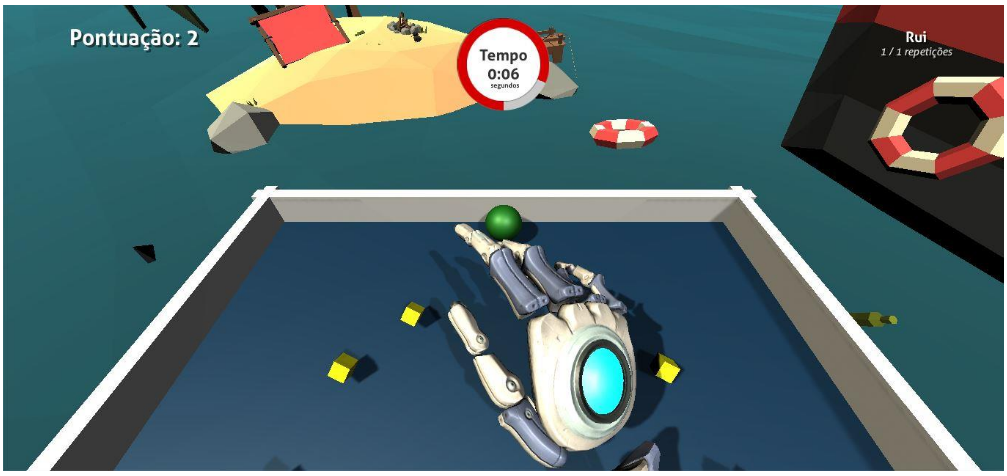


Figure 4.12 – PickaBall Game

After the patient picks up all available cubes with the ball, a coin-shaped object appears in the scenario (see Figure 4.13). The patient must pick up these coin, new cubes will appear on the game board.

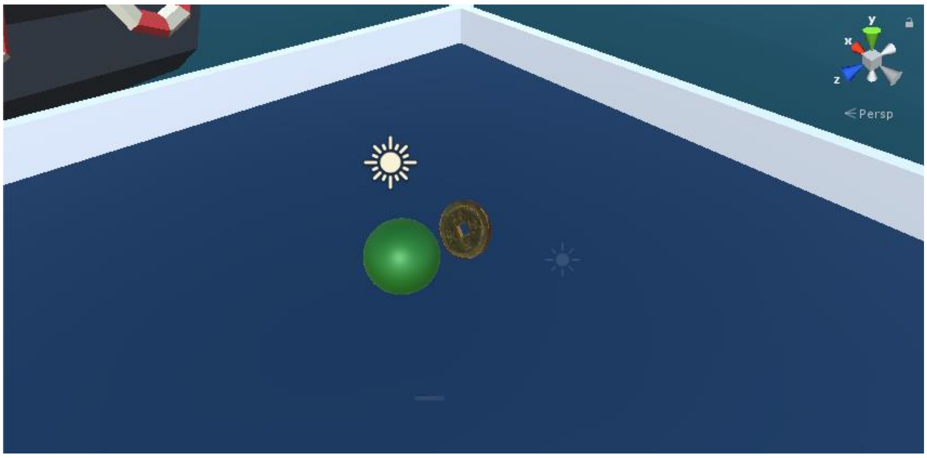


Figure 4.13 – PickaBall game coin object

The ball of the game is available for the patient to grip, but is blocked in height, that is, the patient cannot grasp the ball and move it in height, only in width and length. For each level of difficulty, the position of the cubes is different, as shown in Figure 4.14.

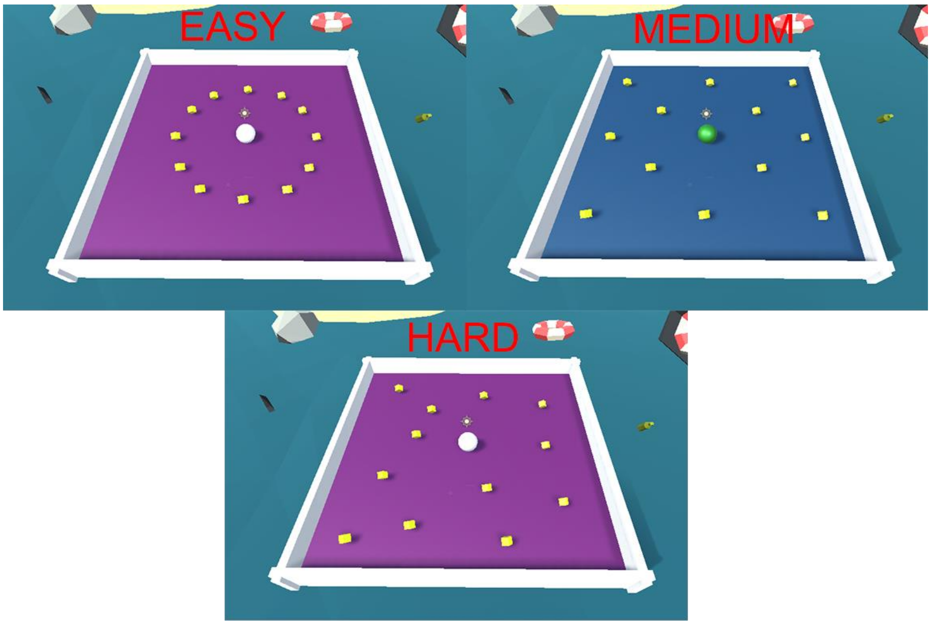


Figure 4.14 – Position of the cubes in the game PickaBall according to the level of difficulty

This game has a unique thematic, the FORMS thematic. When the patient finishes the game, the results in terms of scores are stored in the DB. Two different types of scores are stored:

- Total cubes that appeared on the board;
- Total number of cubes taken by the patient.

## 4.4. Adapting the games according to the characteristics of the patient

Not all physical limitations of the registered patients are the same and in this case, the system need to be adapted to the patient needs. Thus, the application games were developed to be adaptable to the patients needs, so that they have the best gaming experience that helps for their physical rehabilitation.

The games are adapted according to some characteristics of the patient:

- Gender (Male or Female);
- Age.

The scenario changes if the patient is male or female, for example, the floor of the male scenario will have a certain color while in the female scenario will have another, being that, these differences are applied to the textures of the objects that make up the scenario, but also certain objects can be exchanged for others. Figure 4.15 illustrates the differences of the male scenario for the female scenario of both games.

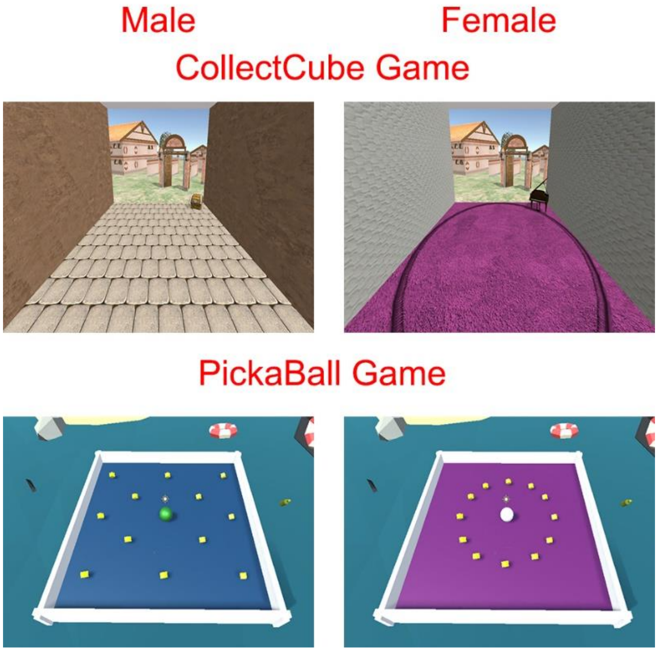


Figure 4.15 – Different game scenario depending on the gender of the patient

When the physiotherapist creates a training for the patient, the avatar hands are considered according with the patient gender or patient age. The physiotherapist will choice which hand to use in the training plan (see Figure 4.16).

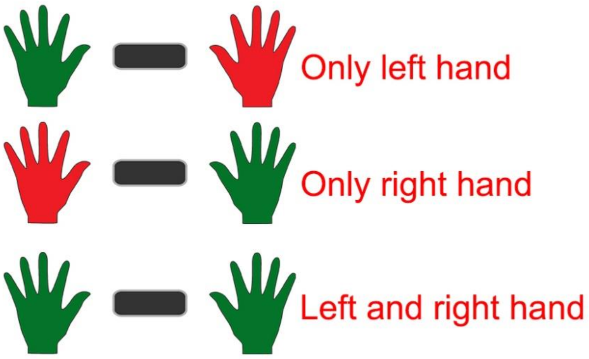


Figure 4.16 – Hand configuration for the patient to use in each training

Such an example if the physiotherapist creates a training plan where it's required a single hand (e.g. Left hand) usage during the session patient may obey when he is playing the game. If the patient decides to use the other hand (right hand) the application will not present the avatar hand in the game, and the patient may not follow the game in this case. In this way it's possible to make a prevention for the patient to execute the tasks with the hand for which it's not allowed to use.

Regarding the parameter age, this has influence in the selection of the graphical representation of the hands and fingers, for that was used three age groups to make this selection:

- From 0 years up to and including 25 years;
- From 26 years up to and including 40 years;
- From the age of 41.

Along with the gender parameter, a representation with a more feminine or masculine aspect will be returned and an aspect plus the young one the smaller the age. Figure 4.17 shows an example of the left hand for an age group of up to twenty-five years and for an age group from the age of forty-one.

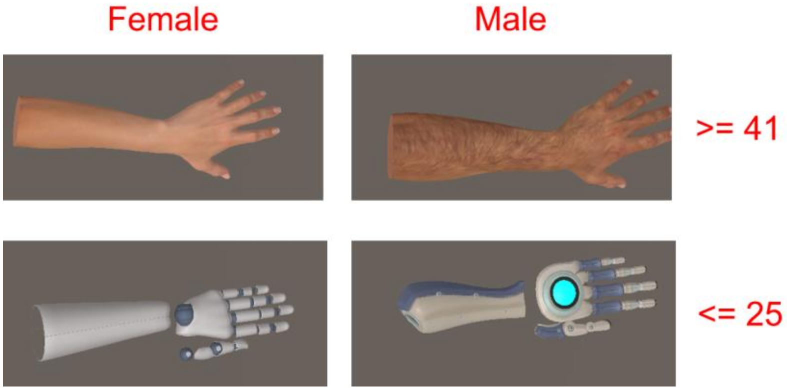


Figure 4.17 – Graphical representation of hands according gender and age

Through the settings described above it’s possible to adapt the game to the characteristics of the patient, making the game more attractive.

## 4.5. Game Feedback

One of the most important components to encourage the patient to continue playing the game is feedback. In any game whether it’s video game or a serious game, it must have a mechanism to respond to the actions performed by the user, for example, whenever the user correctly performs a goal a feedback must be given. We can categorize the feedback into two aspects, visual and auditory. With respect to the visual aspect this can be used to show a message indicating that you have performed a sequence of objectives or even to indicate how many points the patient won or lost by performing a certain task. Regarding the auditory aspect, this can be used as background music or in the case of performing a positive or negative task, for example, in the case of a positive task, a sound that conveys a sense of success is used.

Available application games use feedbacks, both visual and auditory. In the case of the CollectCube game, auditory feedback is used when the patient puts a cube in the box and the background music is used during play so as not to make the game monotonous if the patient cannot place any cube in the corresponding box. Visual feedbacks are used in several situations:

- To indicate whether the patient lost or gained points (see Figure 4.18);
- The timer present in all games informs the remaining time. This timer has a circular bar that will be filled as time progresses, initially with a green color and when there is little left to finish the game, the color changes to red. In this way, visual feedback is given to the patient through the colors (see Figure 4.19);
- In the Collectcube game, whenever the patient places a cube in any box, the box gives a visual effect so that the user notices that the cube has disappeared.

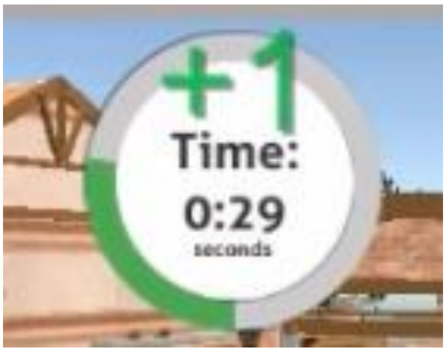


Figure 4.18 – Visual feedback when patient earns points

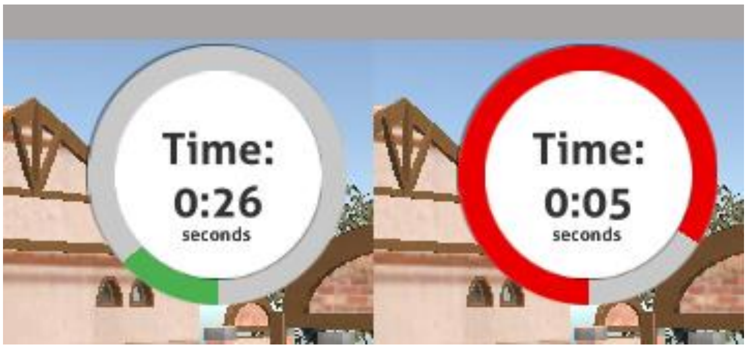


Figure 4.19 – Stopwatch with green and red color

## 4.6. Unity Application

The Unity application is expressed by a set of serious games for patients with hand and finger problems to aid in the rehabilitation process. After authentication in the application, data is collected to configure the games according to the characteristics of the patient, previously configured by the physiotherapist. Figure 4.20 illustrates the flow diagram of the LeaPhysio application.

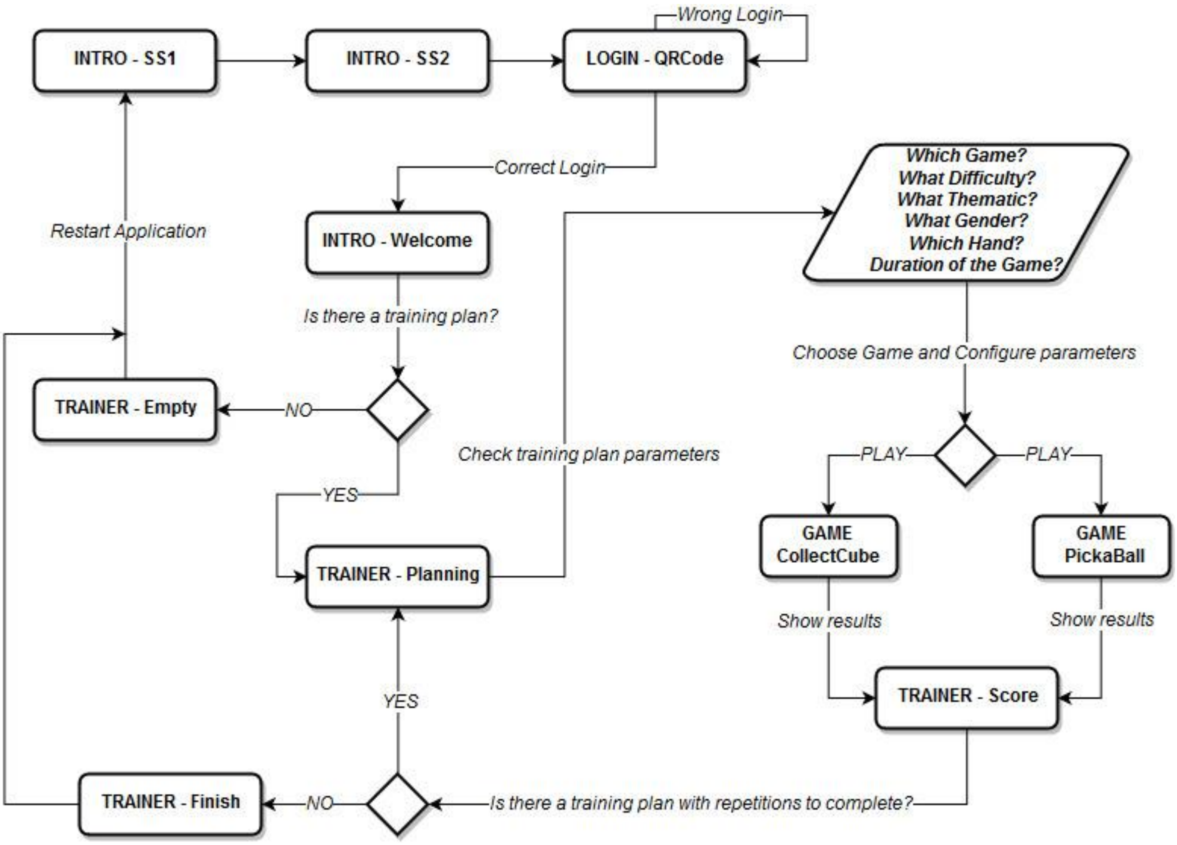


Figure 4.20 – Unity Application Diagram

The user initially sees two small splash screens ("**INTRO - SS1**" and "**INTRO - SS2**"). After the last splash screen, the patient login scenario ("**LOGIN - QRCode**") is opened, as shown in Figure 4.21.

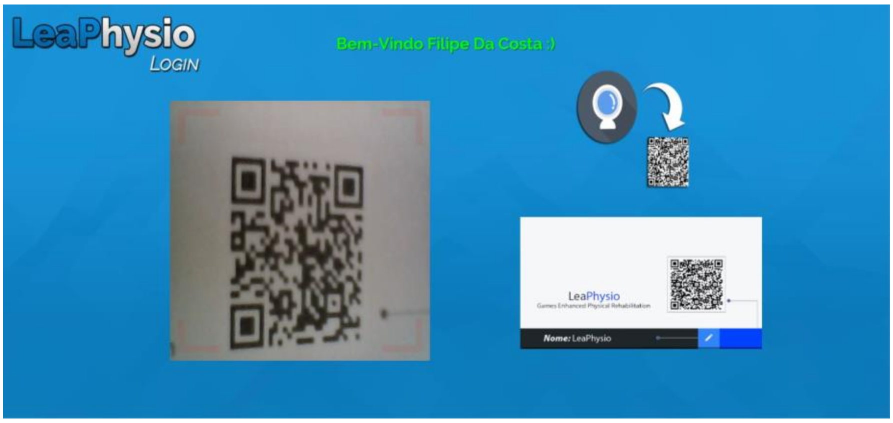
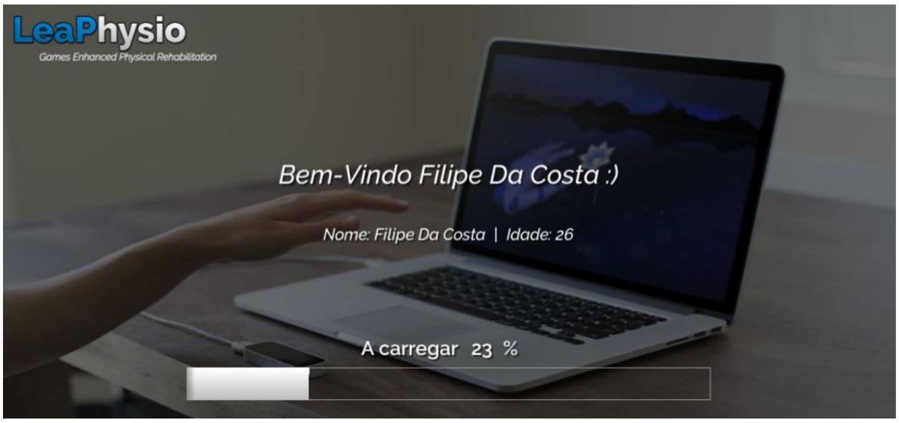

LeaPhysio
LOGIN
Bem-Vindo Filipe Da Costa :)
LeaPhysio
Games Enhanced Physical Rehabilitation
Nome: LeaPhysio

Figure 4.21 – Scene for authentication in the Unity application

As explained in the previous chapter, the developed application uses QRCode's for patients log in. This option was chosen against the traditional authentication system, for example, identification number and password, for a simple reason, this application was designed and developed to be used by people with impairments on the level of hands and fingers, typing numbers and letters in a computer keyboard to login may cause some discomfort or can be also impossible. Thus, the QRCode that is present in the LeaPhysio card and that is unique for each patient, is what allows to perform the authentication in the application. The QR Code stores encrypted information based on a code and the patient identification number. Thus, the patient authenticates to the system successfully a new scenario is shown ("**INTRO-WELCOME**"), see Figure 4.22. This scenario is used to collect information from the remote database, through HTTP communication through the PHP scripts, about the patient's training.



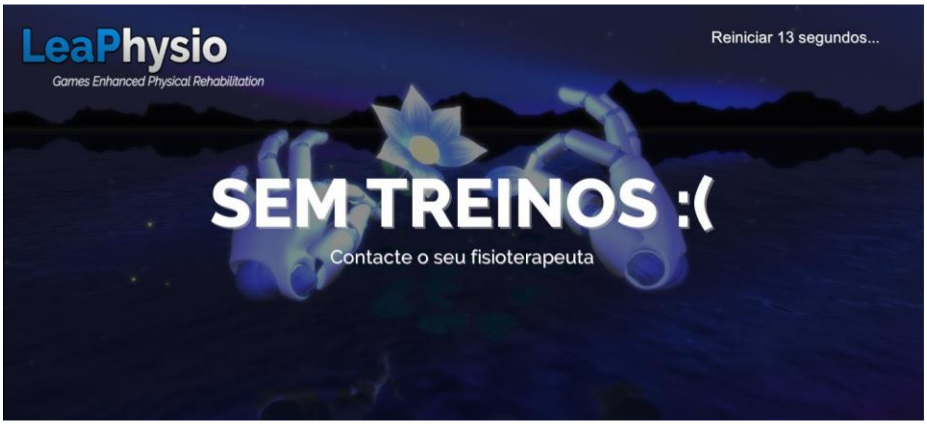
Bem-Vindo Filipe Da Costa :)

Nome: Filipe Da Costa | Idade: 26

A carregar 23 %

Figure 4.22 – Application Welcome Scene

After this information collection has been performed the application will check patient the active training session associated to the current identified, if not has a new scenario ("**TRAINER - EMPTY**") indicating that the patient doesn’t have any training to be performed and application is restarted after the pre-set time. In Figure 4.23 is illustrated this scenario.



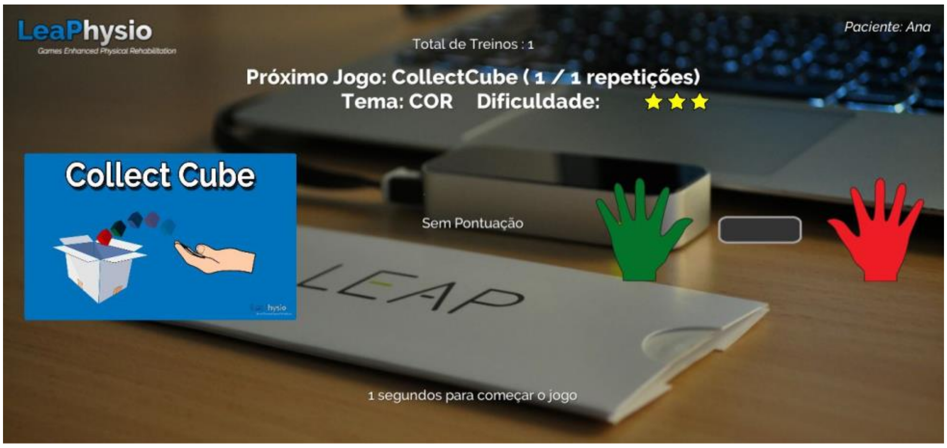
Reiniciar 13 segundos...

SEM TREINOS :(

Contacte o seu fisioterapeuta

Figure 4.23 – There are no active trainings

If for identified patient exist a training session to perform, the scenario for the settings ("**TRAINER-PLANNING"**) is shown, as shown in Figure 4.24.



Paciente: Ana

Total de Treinos : 1

Próximo Jogo: CollectCube ( 1 / 1 repetições)
Tema: COR Dificuldade: ★★★

Sem Pontuação

1 segundos para começar o jogo

Figure 4.24 – Scene to give information about next game that will perform

The planned training session includes:

- the total number of trainings to be performed;
- name of next game;
- how many times does the game should be repeated and how many times has the patient already performed;
- score of the last game performed, for example, patient X has already played game Y for the first time, the second time the score made the first time of game Y is indicated;
- the difficulty of each game (one star for easy difficulty, two for medium difficulty and three for difficult difficulty);
- which hand is imposed to be used during the game (LEFT, RIGHT, BOTH).

This scenario is also shown after the end of each game if the patient still had training or some repetition to perform. The game starts after a certain time predefined by the physiotherapist (the rest time). After the rest time is over, the application automatically selects the corresponding game and configures it according to the following parameters:

- Difficulty;
- Thematic;
- Gender;
- Hand;
- Duration of the game.

After the game setup the game is presented to the patient that will perform the proposed exercises, such as CollectCube or PickaBall game that were previously recommended by physiotherapists. After the playing time is over, a new scenario is shown to the user ("**TRAINER - SCORE**") that indicates the game score, as shown in Figure 4.25.

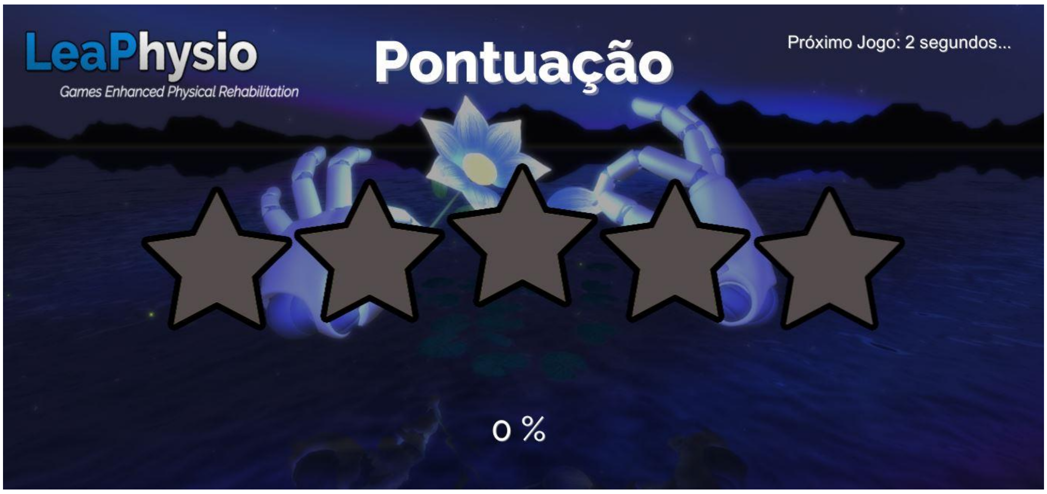

LeaPhysio
Games Enhanced Physical Rehabilitation
Pontuação
Próximo Jogo: 2 segundos...
0 %

Figure 4.25 – Lats game score

The score is shown through stars thus giving a more appealing aspect to the visualization of the result. It’s also in this scenario that the resulting scores are entered into the DB and the file with the coordinates of the finger joints maybe are stored on the server. After a pre-set time, the application checks if there is still some training that has repetitions to do, if it exists, then the application presented the **TRAINER - PLANNING** scenario, otherwise the patient doesn’t have any more training and his rehabilitation session is finished, in which a new scenario ("**TRAINER - FINISH**") is displayed indicating that the session has ended, as illustrated in Figure 4.26.

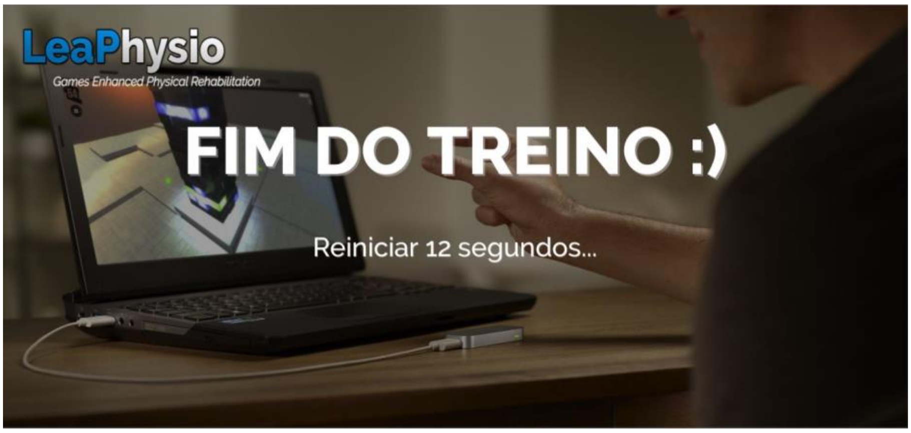


Figure 4.26 – End of training session

After a pre-set time, the application is restarted to its home state ("INTRO - SS1").

The physiotherapist during the application has at his disposal a tool to close the program. It consists of the ESC key of the keyboard or a double-click with the right mouse button and serves in case of an anomaly in the program the physiotherapist can finish or in case the patient is performing some training in which the physiotherapist verifies that isn't indicated for the patient. The application was designed so that the patient doesn't interact with the computer through mouse and keyboard, but only through leap motion.

If for some reason the LMC disconnects from the computer when the patient is playing, the application displays a warning indicating that the controller is not connected, the playing time is stopped (see Figure 4.27). When the LMC is reconnected, the warning disappears, the playing time is activated again.

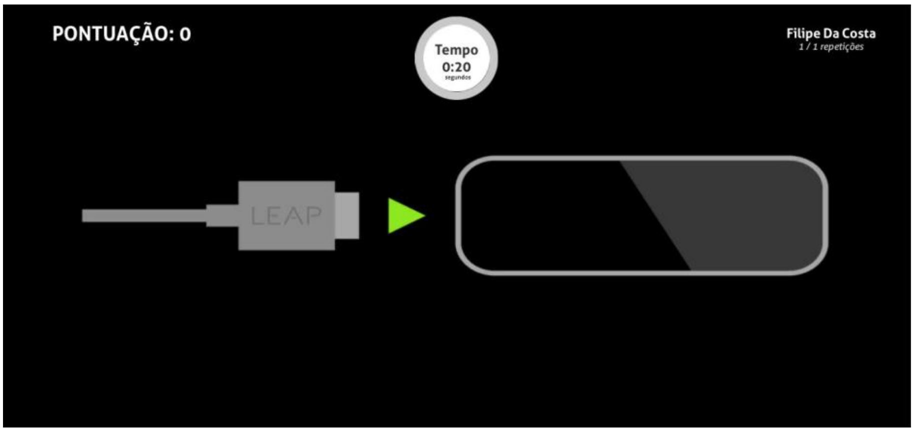


Figure 4.27 – Leap Motion controller disconnect

## 4.7. Data collection

During rehabilitation sessions, data are collected and stored to create an Electronic Health Record, so they are accessible through the mobile application, serving as future analysis for physiotherapists. The LMC is a powerful tool for removing finger and hand data. Through Figure 4.28 it's possible to visualize a representation of the hand with the different possible points that the leap motion detects.

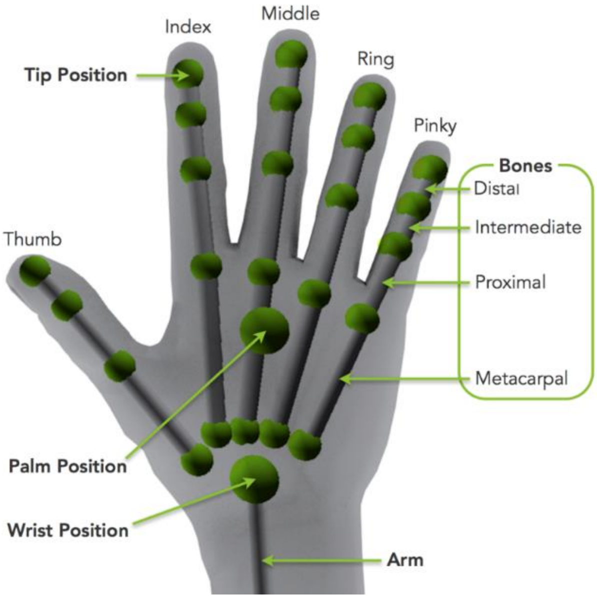


Figure 4.28 – Hand points that the Leap Motion controller detects

Through the points detected by the leap motion it’s possible to extract data, such as positions, rotations or even speeds, as shown in Figure 4.29.

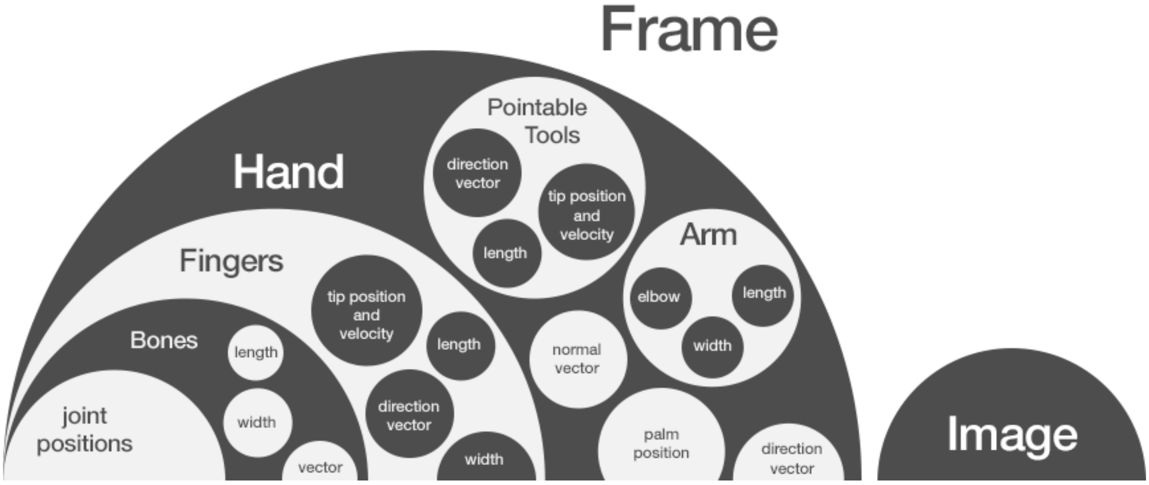


Figure 4.29 – Data provided by Leap Motion controller

An analysis of the points and respective values required for physiotherapists was carried out. Figure 4.30 illustrates which values will be stored.

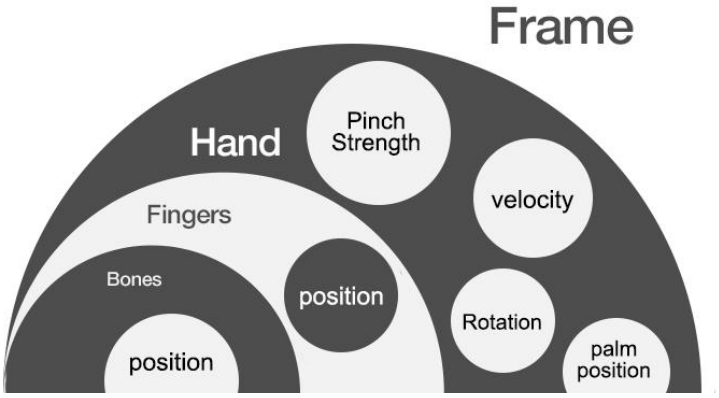


Figure 4.30 – Data collected by the application

In the **Hand** Component:

- *PinchStrength* – Indicates values between 0 and 1, where 0 corresponds to open hand and 1 to the closed hand;
- *Velocity (cm / s)* – Hand speed, with X, Y and Z coordinates;
- *Rotation (degress)* – Rotation of the hand, with PITCH YAW ROLL coordinates (see Figure 4.31);

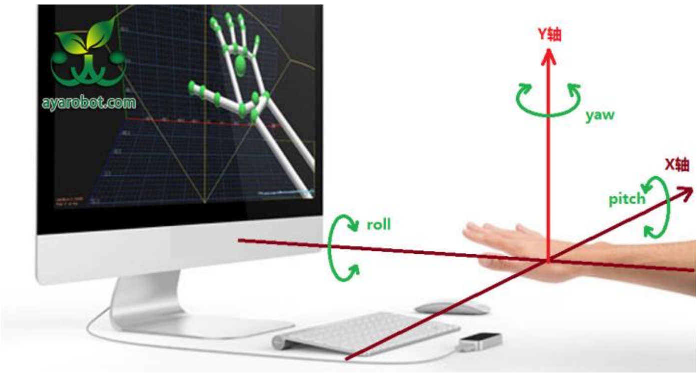


Figure 4.31 – Rotations made by the user's hand

- *Palm Position (cm)* – Hand position in centimeters in relation to the leap motion, with X Y Z coordinates.

In the **Fingers** Component:

The five fingers (THUMB, INDEX MIDDLE, RING, PINKY).

- *Position (cm)* – Position of the finger in centimeters in relation to the leap motion, with coordinates X Y Z. The position considered is at the tip of each finger.

In the **Bones** Component:

The four bones (METACARPAL, PROXIMAL, INTERMEDIATE, DISTAL) are recorded for each finger, as shown in Figure 4.32.

- *Position (cm)* – Position of each bone in centimeters in relation to the leap motion, with coordinates X Y Z. The position considered is in the center of each bone. The collection of information from the fingers is a fundamental part of the origin of the system.

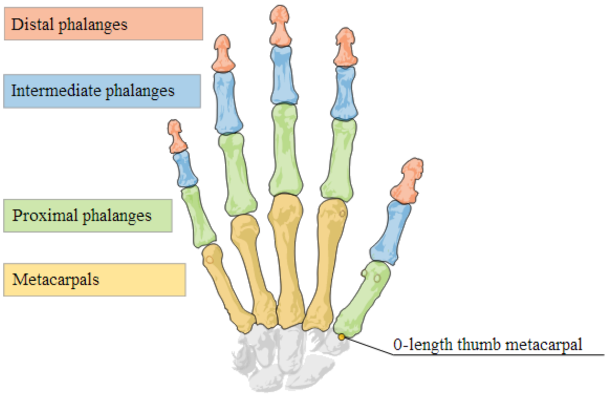


Figure 4.32 – Bones of the hand

After the analysis it was necessary to understand where it was going to store the information resulting from the execution of the training. The first hypothesis would be to create tables in the database to store the information, while the second hypothesis would be to create a file with a certain structure and save that file on the server. The second hypothesis was chosen due to the following factors:

- Due to the large amount of data produced by the games, the database would be under loaded with information, which could affect the rest of the system related search tasks;
- Saving the data to files that are stored on the server makes it possible to download the file for analysis in another program, for example, using MATLAB.

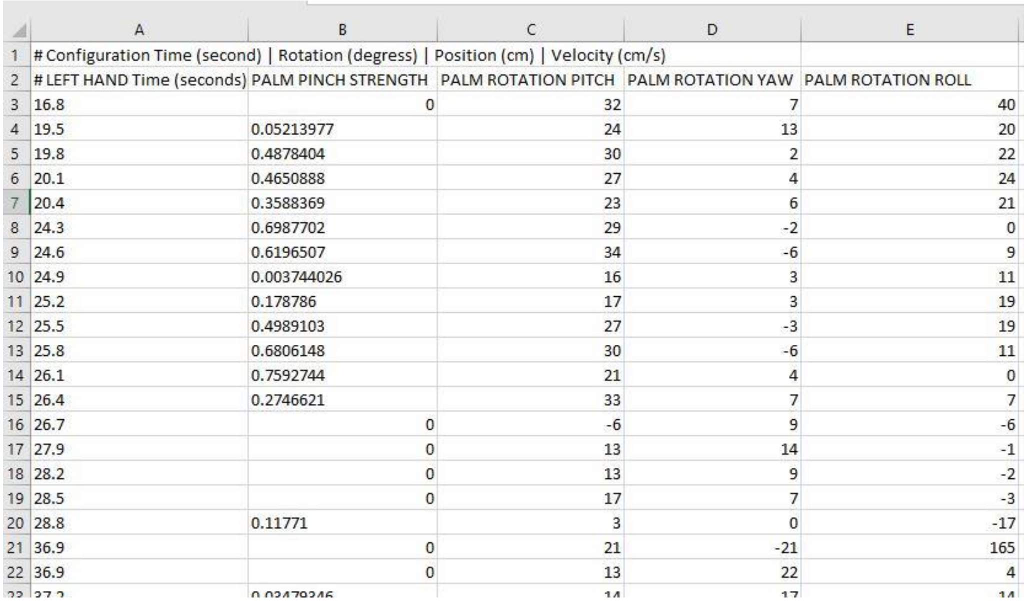
The data is stored in a .CSV format to facilitate future analysis.

A study concerning the leap motion optimal sampling period was carried out. Initially the sampling rate was 2 samples/s and after some tests it was possible to verify that the time resolution was not appropriate and some of the relevant information being lost. Then the sampling rate was increased to 3 samples/s.

Specific organization of the data was considered and following described:

- First, the LEFT hand data is stored and then the RIGHT hand data. When a hand has no information (because it was not considered for the game in question), the file is recorded as *"## [Hand Name] HAND WITHOUT INFORMATION*";
- For each hand, data is divided into two categories. The first category, stores the following data:
    - Time;
    - Palm Pinch;
    - Palm Rotation Pitch;
    - Palm Rotation Yaw;
    - Palm Rotation Roll.

The second category refers to the data that relate to positions and speeds, the palm of the hand, fingers and their bones. This part is also subdivided into three other parts, that is, first all the values corresponding to the X axis, then the Y axis, and finally the Z axis are shown. For example, with respect to the X axis, the order starts by represents the value of the palm position, then the velocity value, the position of the finger, then the position of the finger bones, followed by the position of the next finger and so on. Figure 4.33 shows an example of the file structure when it's opened by the Microsoft Excel program.



| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | # Configuration Time (second) \| Rotation (degress) \| Position (cm) \| Velocity (cm/s) | | | | |
| 2 | # LEFT HAND Time (seconds) | PALM PINCH STRENGTH | PALM ROTATION PITCH | PALM ROTATION YAW | PALM ROTATION ROLL |
| 3 | 16.8 | 0 | 32 | 7 | 40 |
| 4 | 19.5 | 0.05213977 | 24 | 13 | 20 |
| 5 | 19.8 | 0.4878404 | 30 | 2 | 22 |
| 6 | 20.1 | 0.4650888 | 27 | 4 | 24 |
| 7 | 20.4 | 0.3588369 | 23 | 6 | 21 |
| 8 | 24.3 | 0.6987702 | 29 | -2 | 0 |
| 9 | 24.6 | 0.6196507 | 34 | -6 | 9 |
| 10 | 24.9 | 0.003744026 | 16 | 3 | 11 |
| 11 | 25.2 | 0.178786 | 17 | 3 | 19 |
| 12 | 25.5 | 0.4989103 | 27 | -3 | 19 |
| 13 | 25.8 | 0.6806148 | 30 | -6 | 11 |
| 14 | 26.1 | 0.7592744 | 21 | 4 | 0 |
| 15 | 26.4 | 0.2746621 | 33 | 7 | 7 |
| 16 | 26.7 | 0 | -6 | 9 | -6 |
| 17 | 27.9 | 0 | 13 | 14 | -1 |
| 18 | 28.2 | 0 | 13 | 9 | -2 |
| 19 | 28.5 | 0 | 17 | 7 | -3 |
| 20 | 28.8 | 0.11771 | 3 | 0 | -17 |
| 21 | 36.9 | 0 | 21 | -21 | 165 |
| 22 | 36.9 | 0 | 13 | 22 | 4 |
| 23 | 37.2 | 0.03479346 | 14 | 17 | 14 |

Figure 4.33 – Example of data from a file with coordinates referring to the patient's hands and fingers

The data to be stored in independent files provides an extra analysis tool so that physiotherapists can perform other types of analysis through the mobile application. The programming to create the file with the structure is described in Appendix C.

# Chapter 5 – Server Side

## 5. Overview

This chapter is intended to describe LeaPhysio's Server Side and how communications are performed between the server and existing applications (Unity and Android).

### 5.1. Server

The Server is responsible for storing all the information regarding clinic patients, training and results, and the main objective is that all information is centralized in one place and can be accessed remotely by the system applications. By storing this data, it’s possible to analyze and see the information in the future, either with the mobile application destined for physiotherapists and the manager, but also for Unity application for the patients, so the application can collect information and configure the games destined for the patient.

To enable the exchange of information between the LeaPhysio applications and the server, a Web Services called Representational State Transfer (REST) is used. Web Servers are computers responsible for accepting HTTP requests from clients and serving them with HTTP responses. One of the most popular and used in the world, is the Apache server (free software). The process starts with a connection between the terminal, where the Web Server is installed and the client terminal, for example, a browser or an application, and the Web Server must be always available and operational, because it’s not possible to predict when a connection will be made between the two terminals. After the connection is made the customer's request is processed (based on your security restrictions and requested information) and a response is returned from the server. These Web Server can also run programs or scripts to interact with the client. Content sent by the Web Server in response to an HTTP request can have two types:

- **Static Pages**: if the content comes directly from an existing file on the server, that is, regardless of which client accessing the information will always be the same;
- **Dynamic Pages**: if the content is dynamically created by another script program or API called by the server. The request, once received, is processed by the Web Server that will dynamically create the content that will then be sent to the client.

Dynamic pages have the advantage that they can be programmed using some programming language such as PHP, JAVA, C# and others. Through these it’s possible to program them to access a DB and present personalized information, which is sent to the client. The LeaPhysio system uses this type of pages to process the customer's request, access the DB and send the information, through PHP scripts. PHP is a server-side scripting language designed primarily for web development but also used as a general-purpose programming language.

For this purpose, an Architecture system, called LAMP (see Figure 5.1) was used. It's a combination of free and open source software and is one of the most common configurations for webservers. The term LAMP comes from the combination of the following technologies:

- **L**inux: Operation System;
- **A**pache: Web Server (HTTP) software;
- **M**ySQL: Database component;
- **P**HP: Programming language.

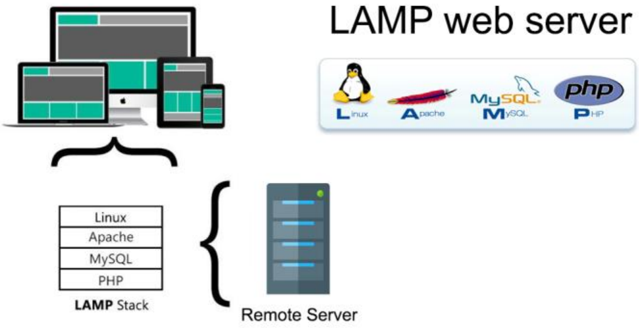


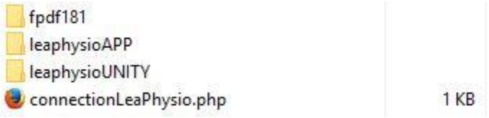
Figure 5.1 – LAMP Architecture System

All components of this architecture are freeware, minus the server that was provided by the dissertation advisor. The configuration needed to implement the architecture was done through the command line, using a SSH (Secure Shell) connection.

After all the settings have been made, the following contents have been developed:

- Creation of the database, to store the necessary information;
- PHP scripts to return content to clients;
- Organization of the folders on the server.

The first two topics will be described in the following sub-chapters. Regarding the organization of the folders on the server, Figure 5.2 presents the files organization:

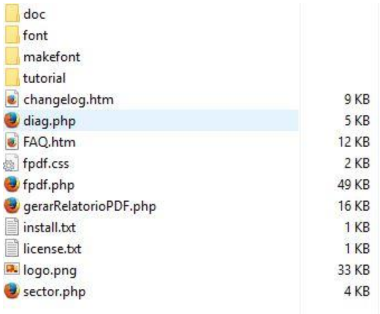


| | |
|---|---|
| fpdf181 | |
| leaphysioAPP | |
| leaphysioUNITY | |
| connectionLeaPhysio.php | 1 KB |

Figure 5.2 – Main folder organization on the server

- ***fpdf181***: this folder contains the scripts needed to generate clinical reports about patients. The *FDPF Library PDF generator* was used. The Figure 5.3 demonstrates the contents of this folder

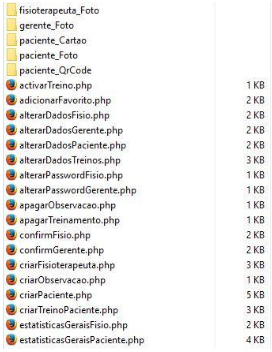
"*FPDF is a PHP class which allows to generate PDF files with pure PHP, that is to say without using the PDFlib library. F from FPDF stands for Free: you may use it for any kind of usage and modify it to suit your needs.*"



| | |
|---|---|
| doc | |
| font | |
| makefont | |
| tutorial | |
| changelog.htm | 9 KB |
| diag.php | 5 KB |
| FAQ.htm | 12 KB |
| fpdf.css | 2 KB |
| fpdf.php | 49 KB |
| gerarRelatorioPDF.php | 16 KB |
| install.txt | 1 KB |
| license.txt | 1 KB |
| logo.png | 33 KB |
| sector.php | 4 KB |

Figure 5.3 – Content of the fdpf181 folder on the server

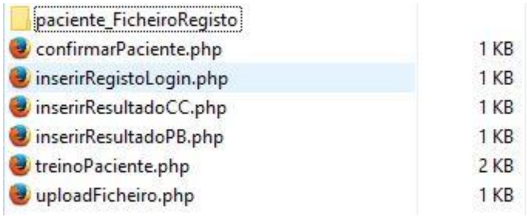
- ***leaphysioAPP***: contains PHP files dedicated to the mobile application (see Figure 5.4). It also stores photographs of patients, physiotherapists and managers, patients' LeaPhysio cards and their QRCode's.



| | |
|---|---|
| fisioterapeuta_Foto | |
| gerente_Foto | |
| paciente_Cartao | |
| paciente_Foto | |
| paciente_QrCode | |
| activarTreino.php | 1 KB |
| adicionarFavorito.php | 2 KB |
| alterarDadosFisio.php | 2 KB |
| alterarDadosGerente.php | 2 KB |
| alterarDadosPaciente.php | 2 KB |
| alterarDadosTreinos.php | 3 KB |
| alterarPasswordFisio.php | 1 KB |
| alterarPasswordGerente.php | 1 KB |
| apagarObservacao.php | 1 KB |
| apagarTreinamento.php | 1 KB |
| confirmFisio.php | 2 KB |
| confirmGerente.php | 2 KB |
| criarFisioterapeuta.php | 3 KB |
| criarObservacao.php | 1 KB |
| criarPaciente.php | 5 KB |
| criarTreinoPaciente.php | 3 KB |
| estatisticasGeraisFisio.php | 2 KB |
| estatisticasGeraisPaciente.php | 4 KB |

Figure 5.4 – Content of the leaphysioAPP folder on the server

- ***leaphysioUNITY***: contains PHP files dedicated to the Unity application (see Figure 5.5). It also stores the coordinate files of the hands resulting from the training performed by the patients.



| | |
|---|---|
| paciente_FicheiroRegisto | |
| confirmarPaciente.php | 1 KB |
| inserirRegistoLogin.php | 1 KB |
| inserirResultadoCC.php | 1 KB |
| inserirResultadoPB.php | 1 KB |
| treinoPaciente.php | 2 KB |
| uploadFicheiro.php | 1 KB |

Figure 5.5 – Content of the leaphysioUNITY folder on the server

- ***connectionLeaPhysio.php***: This file is used to connect to the database. It's the only file that is common to the two existing applications.

## 5.2. Database

The DB stores most of the data used and created by the applications, making it one of the main components of LeaPhysio, and its design and implementation was done ensure the correct operation of the system. The entire database structure was developed in the MySQL Workbench program [47]. This program has several tools to create a database through its visual presentation. The database was developed in MySQL. This DBMS is one of the most popular in the world, and an essential system in LAMP architecture. Initially a system needs analysis was performed to find out what kind of data the database should store. After completion, the database was exported to a file, using the MySQL Workbench, to later import that file to the server through the phpMyAdmin tool [48]. In Figure 5.6 it's possible to observe the final diagram of the database.

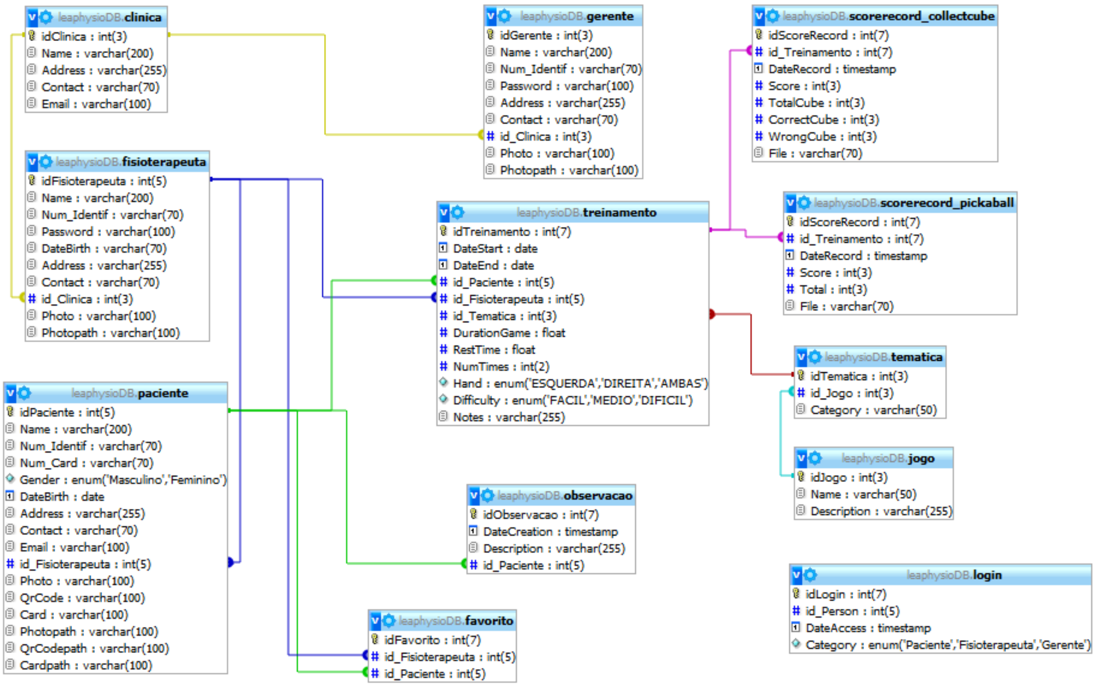


Figure 5.6 – Diagram Database

The database consists of twelve tables:

- **clinica**: this table stores the data for the physiotherapy clinic. Fields such as the name of the clinic, address, contact and e-mail are present in this table. The primary key is idClinica.

- **gerente**: stores the data for the clinic manager. It’s in this table that the credentials (identification number and password) are present for the manager to enter the mobile application. Fields such as name, address and contact are present in this table. The table stores the name of the manager's photo, through the Photo field, which is stored on the server. The Photopath field indicates the path to the folder where the manager's photo is hosted. Contains a foreign key (id_Clinica) that matches the clinical table, thus indicating which clinic the manager belongs to. The primary key is idGerente.
- **fisioterapeuta**: stores the data concerning the physiotherapists of the clinic. It’s in this table that the credentials (identification number and password) are present for the physiotherapist to enter the mobile application. Fields such as name, address, date of birth and contact are present in this table. The table stores the name of the physiotherapist's photograph, through the Photo field, which is stored on the server. The Photopath field indicates the path to the folder where the photograph is housed. Contains a foreign key (id_Clinica) that matches the clinical table, thus indicating which clinic the physiotherapist belongs to. The primary key is idFisioterapeuta.
- **paciente**: stores the data for patients in the clinic. It’s in this table that the patient's credentials are present, through the Num_Card field. This field is automatically generated by the mobile application and this is the content that is inserted in the QrCode of the LeaPhysio card, being an essential field to check the authentication in the Unity application. Fields such as the name, identification number, gender, address, contact, and e-mail are present in this table. The Photo, QRCode, and Card fields indicate the name of the photograph of each component that is stored on the server. The Photopath, QrCodepath, and Cardpath fields indicate the path to the path where the photograph is lodged. Contains a foreign key (id_Fisioterapeuta) that matches the physiotherapist table, indicating which physiotherapist is responsible for that patient. The primary key is idPaciente.
- **observacao**: stores the data regarding the observations created by the physiotherapists for the patients. This table has a field with the creation date of the observation and its description. Contains a foreign key (id_Paciente) that matches the patient table, thus indicating which patient belongs to the observation. The primary key is idObservacao.
- **favorito**: each physiotherapist using the mobile application can mark a patient as a favorite. It’s in this table that this information is stored. Contains two foreign keys, id_Fisioterapeuta and id_Paciente, which correspond to the physiotherapist and clinic

table, respectively, thus indicating that a patient is signaled as a favorite for a physiotherapist. The primary key is idFavorito.

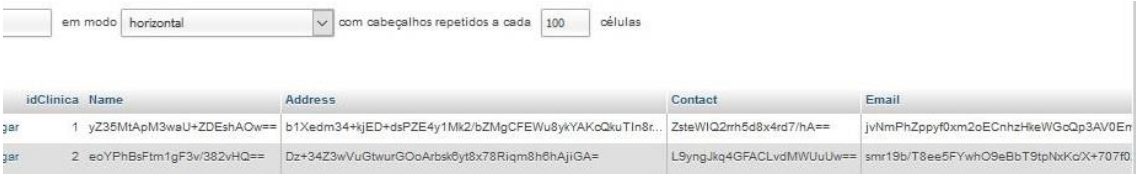
- **treinamento**: it’s in this table that the configurations relative to each training are stored, being an essential table for the patients to execute the training created by the physiotherapists. The DateStart field indicates the start date for which this workout is available. The DateEnd field indicates the end date of the workout, for example, the date by which the patient can perform the training. The DurationGame field indicates the time the patient must play each game. The RestTime field indicates the rest interval between two games. The NumTimes field indicates how many times that game must be repeated. The Hand field indicates which hand to use (LEFT, RIGHT or BOTH). The Difficulty field indicates the difficulty of the game (EASY, MEDIUM or HARD). The Notes field allows the physiotherapist responsible for the training to write some observation. This table contains three foreign keys, the Patient id to indicate to which patient this training belongs, the id_Fisioterapeuta that indicates which physiotherapist is responsible for the training and the id_Tematica that indicates the theme of the game. Through this last foreign key, it’s possible to know which game the theme belongs to. The primary key is idTreinamento.
- **login**: whenever a system intervener (patient, manager or physiotherapist) logs in to LeaPhysio applications, their date of entry is stored in the database. The id_Person field is intended to store the id of the patient, manager or physiotherapist tables. The DateAccess field stores the date that the login was performed, while the Category field indicates which category (Patient, Manager, or Physiotherapist). The primary key is idLogin.
- **jogo**: is where the names and descriptions of the different existing games for the Unity application are stored. The primary key is idJogo.
- **tematica**: each game has at least one theme, which makes the objects found during each game different. This table is intended to store the names of the existing themes, through the Category field. It has a foreign key (id_Jogo) that matches the game table, thus indicating which game belongs to a certain theme. The primary key is idTematica.
- **scorerecord_pickaball**: this table is intended to store the results coming from the workouts performed by each patient when the game is PickaBall. The DateRecord field stores the date that the result was entered the table. The Score and Total field

refers to the scores performed by the patient. The File field indicates the name of the file where the coordinates of the hands taken during the game performed by the patient are contained. It has a foreign key that matches the training table, thus indicating that a certain result comes from a certain training, which training is performed by a patient. The primary key is idScoreRecord.

- **scorerecord_collectcube**: this table is intended to store the results of the training performed by each patient when the game is CollectCube. The DateRecord field stores the date that the result was entered the table. The Score, TotalCube, CorrectCube and WrongCube fields report to patient scores. The File field indicates the name of the file where the coordinates of the hands taken during the game performed by the patient are contained. It has a foreign key that matches the training table, thus indicating that certain results come from certain training, which training is performed by a patient. The primary key is idScoreRecord.

It was decided that the profile photos referring to the interveners, QRCode's and LeaPhysio cards should not be stored directly in the database, due to their size limitation for these fields and not to load the database with this information, taking advantage of the server and storing in this (e.g. in the patient table, there is a field that stores the link for LeaPhysio card). The coordinates of the hands from the training performed by the patients are also not stored directly in the database, but in a file hosted on the server. It's mainly due to the enormous amount of data collected in each game performed by the patient, that is, the database in the short term would be overloaded which would make slow access to other information. For this it was decided to store this information in isolated files with a .CSV structure for easy access of the mobile application. During the design of the database, the type and size of the fields of each table were also considered, optimizing the table.

As the database stores sensitive information, that is, it contains passwords and personal data of each user, it was considered that the most important fields should be encrypted. In Figure 5.7 it's possible to visualize the data of the clinic table.



| | idClinica | Name | Address | Contact | Email |
|---|---|---|---|---|---|
| gar | 1 | yZ35MtApM3waU+ZDEshAOw== | b1Xedm34+kjED+dsPZE4y1Mk2/bZMgCFEWu8ykYAKcQkuTIn8r... | ZsteWIQ2rrh5d8x4rd7/hA== | jvNmPhZppyf0xm2oECnhzHkeWGcQp3AV0Em |
| gar | 2 | eoYPhBsFtm1gF3v/382vHQ== | Dz+34Z3wVuGtwurGOoArbsk6yt8x78Riqm8h6hAjiGA= | L9yngJkq4GFACLvdMWUuUw== | smr19b/T8ee5FYwhO9eBbT9tpNxKc/X+707f0 |

Figure 5.7 – Encrypted data from the clinical table

The mobile application deals with the part of encryption and decryption. In Chapter 6, referring to the mobile application, this aspect is treated in more detail.

The following code describes the developed structure to create the *observacao* table.

```
--
-- Estrutura da tabela `observacao`
--

CREATE TABLE IF NOT EXISTS `observacao` (
`idObservacao` int(7) NOT NULL AUTO_INCREMENT,
`DateCreation` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
`Description` varchar(255) NOT NULL,
`id_Paciente` int(5) NOT NULL,
PRIMARY KEY (`idObservacao`),
KEY `id_Paciente_obs_idx` (`id_Paciente`)
) ENGINE=InnoDB  DEFAULT CHARSET=utf8 AUTO_INCREMENT=22 ;
```

In the code it's possible to verify that all the fields of the *observacao* table are mandatory and that primary key of the table is idObservacao.

## 5.3. PHP Scripts

The use of PHP scripts is required to perform communication between the server and the applications (Unity and Android). These files are stored on the server, and when it's necessary to perform some search operation, change data or even erase data, the corresponding file is invoked using a HTTP (Hypertext Transfer Protocol) connection. The HTTP is designed to enable communications between clients and servers and acts as a request-response protocol between client and server (see Figure 5.8), for example, a client submits an HTTP request to the server and it returns a response to the client. This response contains information about the request and may also contain the requested content.

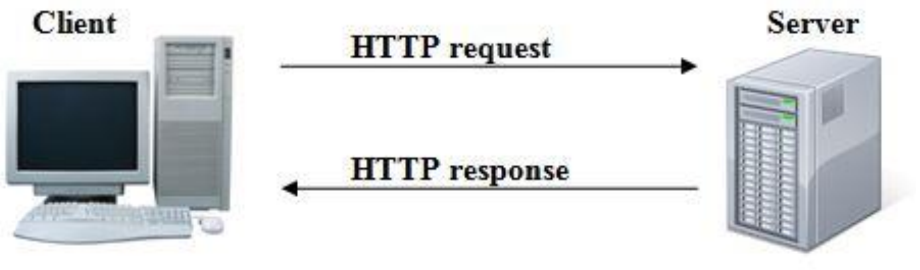


Figure 5.8 – Request-Response Protocol

The two most usual methods for a request-response between client and server are: **GET** and **POST**.

- **GET** – Requests data from a specified resource;
- **POST** – Submits data to be processed to a specified resource.

The GET method passes the variables through the URL.

- /iscte/test.php**?name1=value1&name2=value2**

Variables are embedded in the URL, which makes it useful to share, for example, a page of a product, only having to copy the URL. This method has restrictions in terms of variable size and should never be used when dealing with sensitive data.

- POST /iscte/test.php HTTP/1.1
  Host: iscte.pt
  **name1=value1&name2=value2**

Through this method the variables are not passed through the URL, making it impossible to see the variables sent. On the other hand, it has no restrictions in terms of data size, ideal, for example, for sending files. The GET method is relatively faster than the POST method, since the latter needs to take some time to encapsulate the information in the message. The main difference is mainly in the visibility of the data. Both applications deal with sensitive information about patients and their training plans, that is, this information should always be the best protected. Then the POST method was chosen to protect the information, but also because it doesn’t have any restriction in terms of data size, making it possible to send photos or files to the LeaPhysio server.

Regarding PHP scripts, we can catalog the files in two categories:

- Those that are used by the Unity application. They are scripts that have as main function to look for necessary information about the patients and training plans to be able to configure the game scenarios, so that they are prepared to the needs of each training and to record information on the results, either at the level of scores or coordinates fingers and hands;
- The ones that are used by the Android application. They are scripts for searching patient information, creating new patients, creating training plans, changing training parameters, erasing drills or even visualizing results.

All programming could only have been done in a single PHP script, but due to the number of existing scripts and for better understanding, organization and detection of possible errors, it was decided to perform separate scripts. The *connectionLeaPhysio.php* file is used by all PHP scripts, which is a fundamental file for connecting to the database.

In the Unity application there are some important files for the proper functioning of the system:

- ***confirmarPaciente.php***: check if the patient is valid in the system. This script is invoked when the patient in the Unity application shows their QRCode of the LeaPhysio card in the Webcam. The application reads the information and the script tries to check if this information belongs to any patient. If it exists, returns the necessary information to follow the process of the application, otherwise, an error message is sent;
- ***treinoPaciente.php***: using the Patient id, this script tries to gather information about the patient's active training, and it's through the information provided by the script that the Unity application configures the games for the patient;
- ***uploadFicheiro.php***: at the end of each game, the Unity application creates a file with a certain structure referring to the various coordinates of the hands that the patient used. This script is meant to send that same file to the cloud and store it in its place.

In the mobile application there are some important files for the proper functioning of the system:

- ***meusPacientes.php***: returns information about patients, in which a physiotherapist is responsible.
- ***verTreinosporPaciente.php*: returns the data of all existing trainings for a given patient;
- ***criarTreinoPaciente.php***: allows create a training for a patient. This script is used by the physiotherapist in the application and the information is sent by the POST method of the HTTP connection. The script returns a positive result if no error has occurred. The code structure of PHP scripts for the most part is similar. Initially, checks that serve to verify that the request method is what is indicated in the file and that the arguments that are sent through the request method exist. If there are no errors, the connection to the DB is started, SQL query is created and later executed, returning the results found and through them to make decisions, for example in case of not finding

some result write a message to indicate the same, or otherwise, save the information in a structure to be written.

We'll look in more detail at the script code used by the Android application.

```
1 <?php
2 if ($_SERVER['REQUEST_METHOD'] == 'POST') {
3   if (isset($_POST['name'])) {
4       require_once("../connectionLeaPhysio.php");
5       error_reporting(0);
6
7       $name = strip_tags(trim($_POST['name']));
8
9       $sql = "SELECT clinica.Name, clinica.Address, clinica.Contact,
clinica.Email FROM clinica WHERE clinica.Name='$name'";
10
11      $result = mysqli_query($conn, $sql);
12      if (mysqli_num_rows($result) > 0) {
13          $arr     = array();
14          $row     = mysqli_fetch_array($result);
15          $nameC   = $row['Name'];
16          $address = $row['Address'];
17          $contact = $row['Contact'];
18          $email   = $row['Email'];
19
20          $arr[] = array( 'Name' => "$nameC", 'Address' =>
"$address",
21              'Contact' => "$contact", 'Email' => "$email" );
22          echo '{"clinic":' . json_encode($arr) . '}';
23          ;
24      } else {
25          echo 'Error';
26      }
27
28      mysqli_close($conn);
29  }
30}
```

Line number 2 checks if the request method is of type POST, if it’s then continuing the rest of the script process. Line number 3 verifies that the POST method contains the name of the variables that it’s waiting to receive and that they are not empty. If there are no errors, the script invokes the *connectionLeaPhysio.php* file to make a connection to the database. Line 7 is intended to store the value of the variable sent by POST to a local variable. In line 9 is built the SQL query to search the DB, it will look for information about the clinic. Line 11 executes the query, while line 12 checks whether the result returned by the query execution is greater than 0, if it’s less than or equal, is written to an error message through the echo function. If it’s greater than 0, the resulting information, stored in variables, is read for later storage in an array. At the end the array is converted to a JSON format and is written on the page, and you can see the result. This result is never visible to the user directly, because the user doesn't execute the script, for example, in a browser, it's the Android application in this case that executes and reads the information written by the PHP script, thus showing the result to the user. Finally, the script closes the connection to the DB.

Patient reports are created through PHP files, hosted on the server. The *FDPF Library PDF generator* was used. The report has three chapters:

- Chapter with general information;
- Chapter about trainings;
- Chapter with results.

About programming, the code was structured to be easy to add new chapters in the future. The following is an excerpt from the file *exportPDF.php*.

```
    function ChapterNumber1($num_identif, $name_Fisio){
        $this->ChapterHeader("I", utf8_decode('Informações Gerais'));
        $this->SetFont('Helvetica', '', 10);
        $date = date('Y-m-d H:i:s');
        $this->Cell(0, 5, utf8_decode('Data do Relatório: '. $date), 0,
1, 'R');
        $this->GeneralInformation($num_identif, $name_Fisio);
        $this->Cell(0, 5, '', 0, 1);
    }
    function ChapterNumber2($num_identif){
        $this->ChapterHeader("II", utf8_decode('Treinos'));
        $this->SetFont('Helvetica', '', 10);
        $this->GeneralStatistics($num_identif);
        $this->Cell(0, 5, '', 0, 1);
        $this->SetFont('Helvetica', 'B', 10);
        $this->Cell(0, 5, utf8_decode('Treinos Activos'), 0, 1, 'C');
        $this->SetFont('Helvetica', '', 10);
        $this->Cell(0, 5, '', 0, 1);
        $this->ActiveTrainings($num_identif);
        $this->Cell(0, 5, '', 0, 1);
    }
if($_SERVER['REQUEST_METHOD'] == 'POST'){
    if (isset($_POST['num_identif']) && isset($_POST['name_Fisio'])){
        error_reporting(0);
        $num_identif = strip_tags(trim($_POST['num_identif']));
        $name_Fisio = strip_tags(trim($_POST['name_Fisio']));

        $pdf = new PDF();
        $pdf->AliasNbPages();
        $pdf->AddPage();
        $pdf->ChapterNumber1($num_identif, $name_Fisio);
        $pdf->ChapterNumber2($num_identif);
        $pdf->ChapterNumber3($num_identif);

        $pdf->Output();
```

Appendix D contains an example of a patient report. More information about programming PHP scripts is available in Appendix C.

# Chapter 6 – Mobile Application

## 6. Overview

This chapter describes the mobile application of the LeaPhysio System, how the initial evaluation took place, what are the main functions in the application for users and how it was implemented and what resources it uses.

### 6.1. Initial Evaluation

The application was developed as a working tool to allow the physiotherapist to manage the information during the physiotherapy sessions, but also for future analyzes on the results obtained by the patients. The need to have a well-designed and intuitive application, serves to help the physiotherapist, improving interaction between machine-man. Initially a decision was made about the final platform to use the application. This decision was made between two technologies:

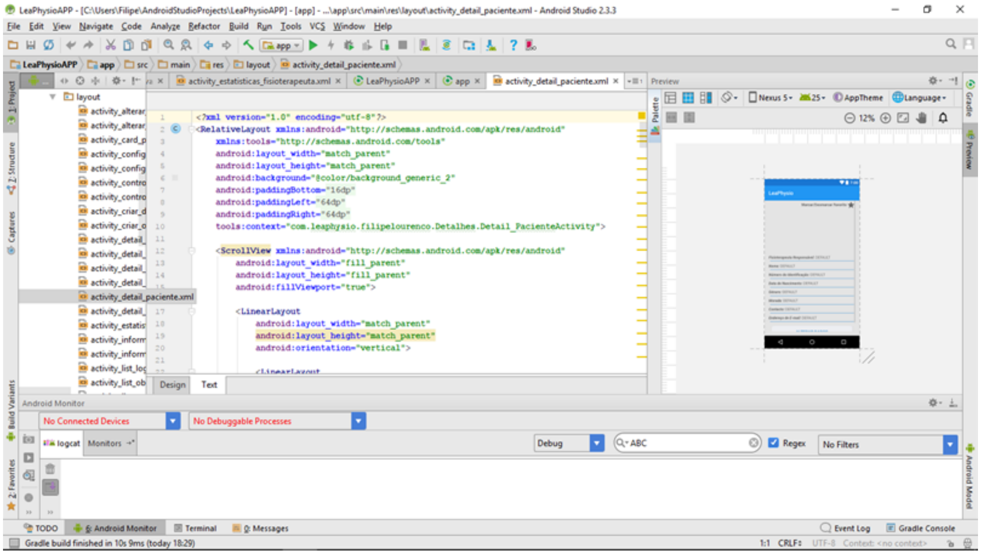
- Computer application;
- Mobile application;
- Web application.

The decision fells on the mobile application. The physiotherapist always in constant movement to assist his patients, which in the case of a computer application would oblige the physiotherapist to move to a certain place where the software would be installed, taking the attention of patients and causing some discomfort. The mobile application presents the advantages of mobile device that can follow the physiotherapists. After making the decision which application is to be used on mobile devices, another question was considered, what should be the development platform (see Figure 6.1).

- Native application for iOS;
- Android application for Android OS;
- Hybrid application, developed in Xamarin.

Figure 6.1 – Platforms for mobile application development

One of the great advantages of having a hybrid application is that it can be used on more than one platform, for example iOS and Android, while a native application can optimize the application for the platform that is intended for it, taking advantage of the resources. In the end the choice fell on a native application for Android mobile devices. The only function of the mobile device is to use the application, and for this, the device doesn’t need to have the most advanced specifications in the world of technology. There are thousands of Android mobile devices for sale on the market and in monetary terms, an Android mobile device is cheaper than an iOS device, which allows the clinic to be able to purchase several devices at a low price, which would not happen with iOS devices. In terms of mobile devices, the Android operating system is present in Phones and Tablets. Here the choice focused on Tablets, due to the need to visualize results in graphs and large amounts of information, either on training or patient data. The application was developed using Android Studio 2.3.3 software (see Figure 6.2). Android Studio is the official IDE for Google's Android operating system. For its development in terms of programming was used JAVA.



LeaPhysioAPP - [C:\Users\Filipe\AndroidStudioProjects\LeaPhysioAPP] - [app] - ...\app\src\main\res\layout\activity_detail_paciente.xml - Android Studio 2.3.3

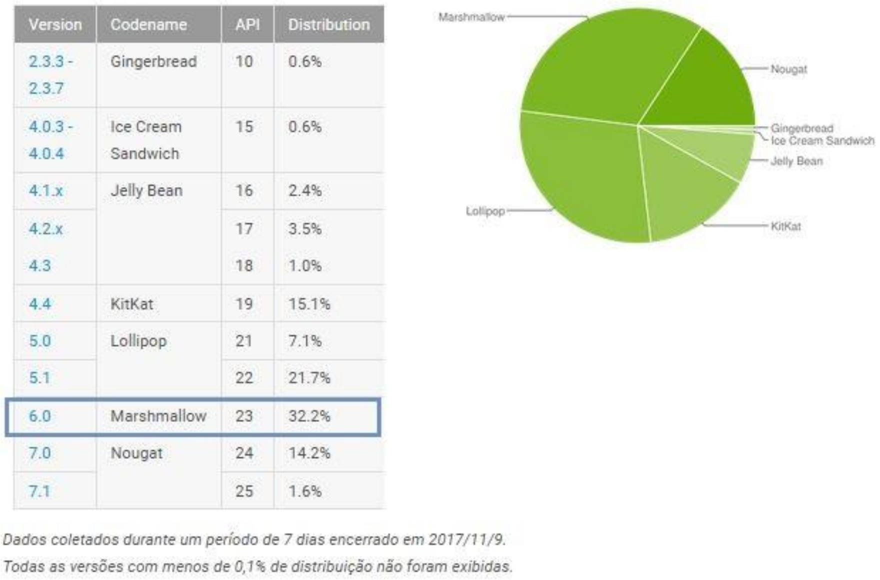
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/background_generic_2"
    android:paddingBottom="16dp"
    android:paddingLeft="64dp"
    android:paddingRight="64dp"
    tools:context="com.leaphysio.filipelourenco.Detalhes.Detail_PacienteActivity">

    <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:fillViewport="true">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:orientation="vertical">

            <LinearLayout
```

No Connected Devices
No Debuggable Processes

Gradle build finished in 10s 9ms (today 18:29)

Figure 6.2 – Android Studio

Through the official Android website for developers [49], Google provides data on the relative number of devices with a certain version of the Android platform (see Figure 6.3).



| Version | Codename | API | Distribution |
|---|---|---|---|
| 2.3.3 - 2.3.7 | Gingerbread | 10 | 0.6% |
| 4.0.3 - 4.0.4 | Ice Cream Sandwich | 15 | 0.6% |
| 4.1.x | Jelly Bean | 16 | 2.4% |
| 4.2.x | | 17 | 3.5% |
| 4.3 | | 18 | 1.0% |
| 4.4 | KitKat | 19 | 15.1% |
| 5.0 | Lollipop | 21 | 7.1% |
| 5.1 | | 22 | 21.7% |
| 6.0 | Marshmallow | 23 | 32.2% |
| 7.0 | Nougat | 24 | 14.2% |
| 7.1 | | 25 | 1.6% |

*Dados coletados durante um período de 7 dias encerrado em 2017/11/9.*
*Todas as versões com menos de 0,1% de distribuição não foram exibidas.*

Figure 6.3 – Data on the relative number of devices with a certain version of the Android platform

Most devices already support version 6.0. So, the proposed application was developed to run on devices with version 4.0.4 (Ice Cream Sandwich) or higher, and its target is version 6.0 (Marshmallow). To run the application, the device must have access to the internet, either through a mobile network (3G, 4G) or Wi-Fi.

## 6.2. Structure

The mobile application can be used by two users:

- The physiotherapist to perform queries regarding patient data, create training plans and register patients or to visualize results.
- The manager to register physical therapists in the system or to view information about them.

It was designed to serve as an interface between the users and the system, providing the necessary resources for its use. Figure 6.4 illustrates the functionalities that the Physiotherapist has at his disposal in the application.

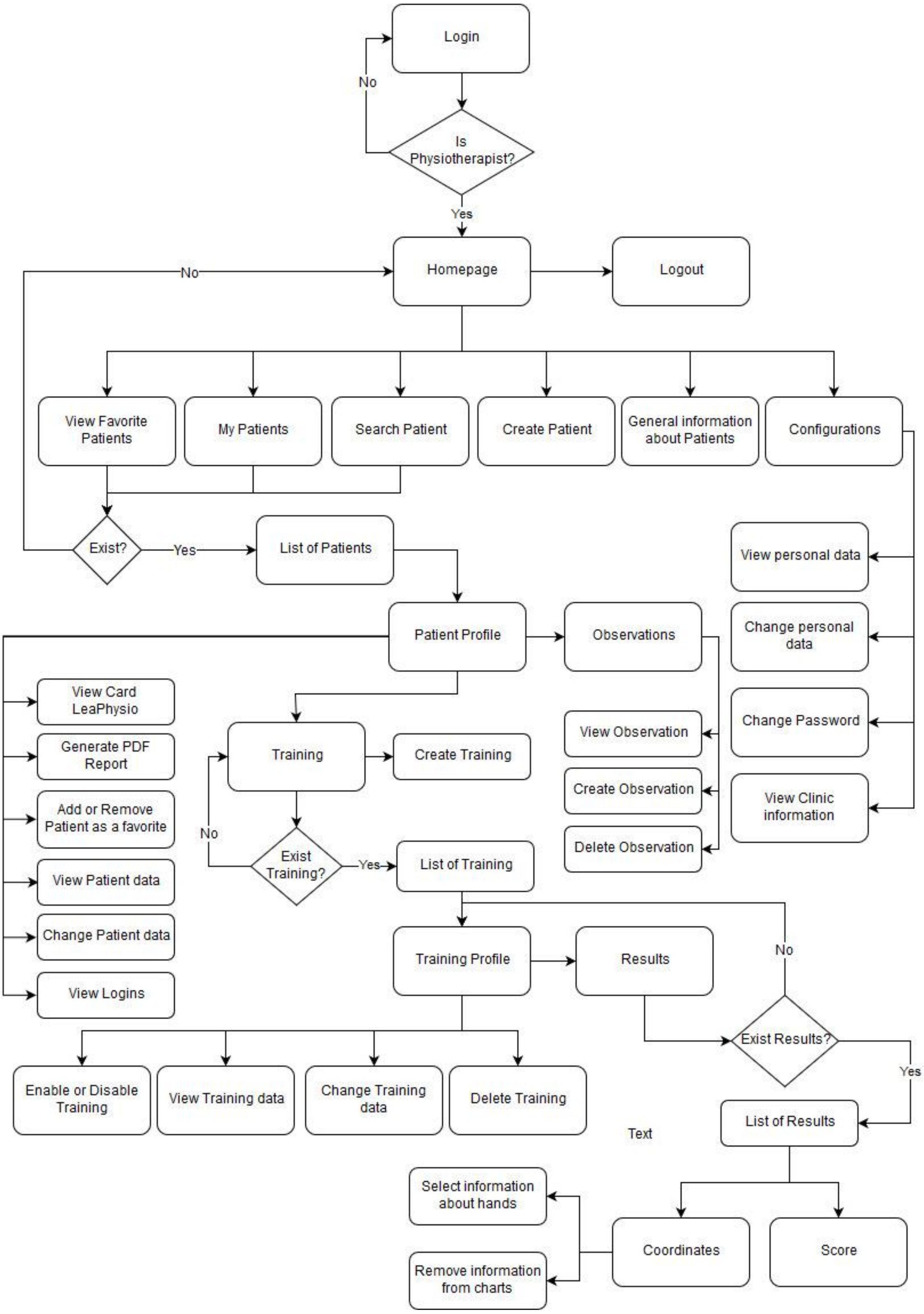


Figure 6.4 – Structure of the functionalities of the physiotherapist in the mobile application

The physiotherapist after successfully logging in, is given a set of options, for example, to visualize their patients. If the physiotherapist has a patient for you, a list is shown, allowing you to access your profile. The patient profile is an essential component that allows access to a diverse set of functionalities, either to visualize the LeaPhysio card or even to visualize the training if it exists. In the training profile it's possible to visualize the settings and activate or deactivate the training, but also, to visualize the results coming from the execution of the exercises by the patient. The results are divided into two types, scores and coordinates. The scores type refers to the patient's scores in the game, for example, total of certain cubes. The type coordinates, indicates the positions of the various points of the hands used.

Figure 6.5 illustrates the features that the manager has at his disposal in the application.

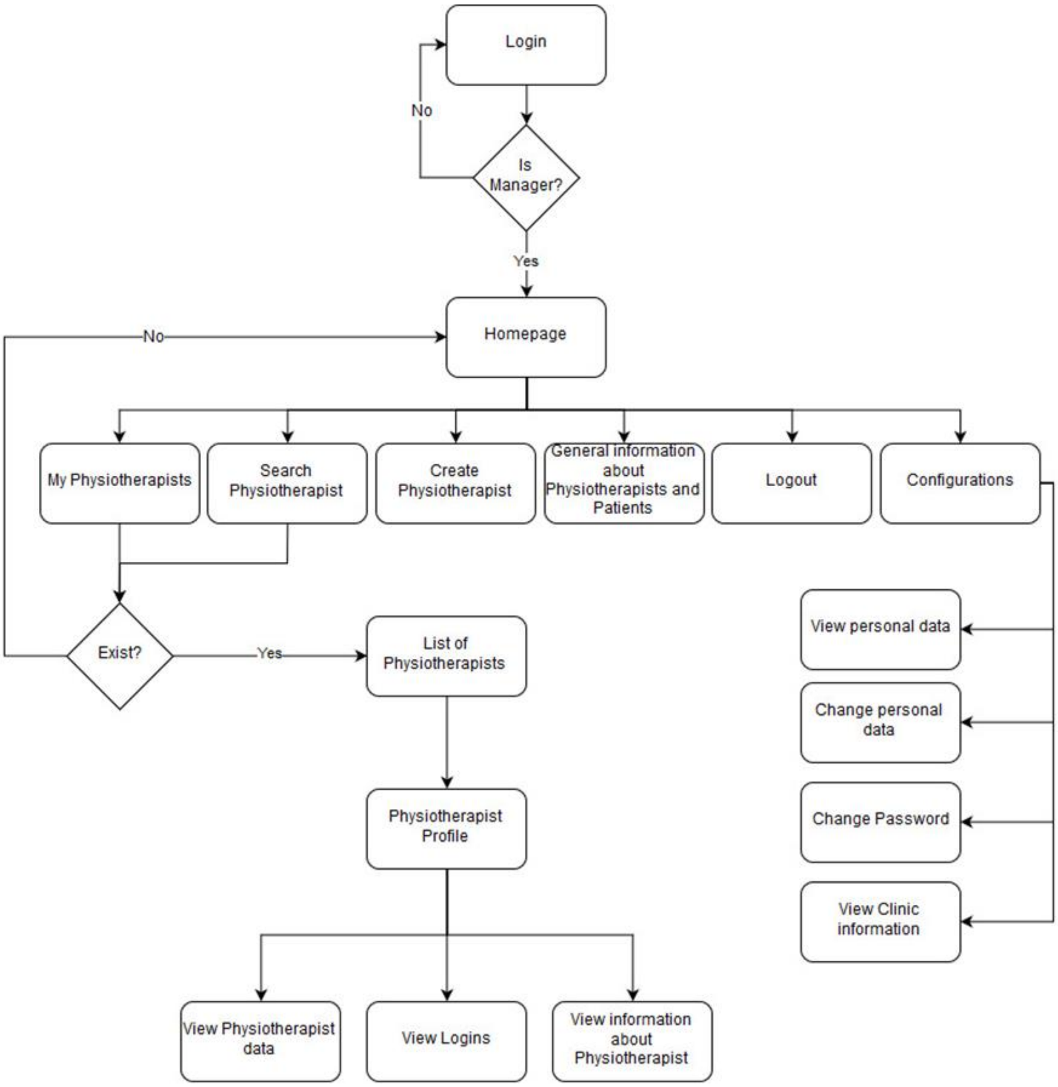


Figure 6.5 – Structure of the functionalities of the manager in the mobile application

The manager can search the physiotherapists for their identification number. In the profile of the physiotherapist, is possible visualize their data and the names of the patients for which the physiotherapist is responsible. In terms of functionalities, they are less compared to physiotherapists, because the physiotherapist "controls" the patients, who are the focus of the system and where the main functions are centered.

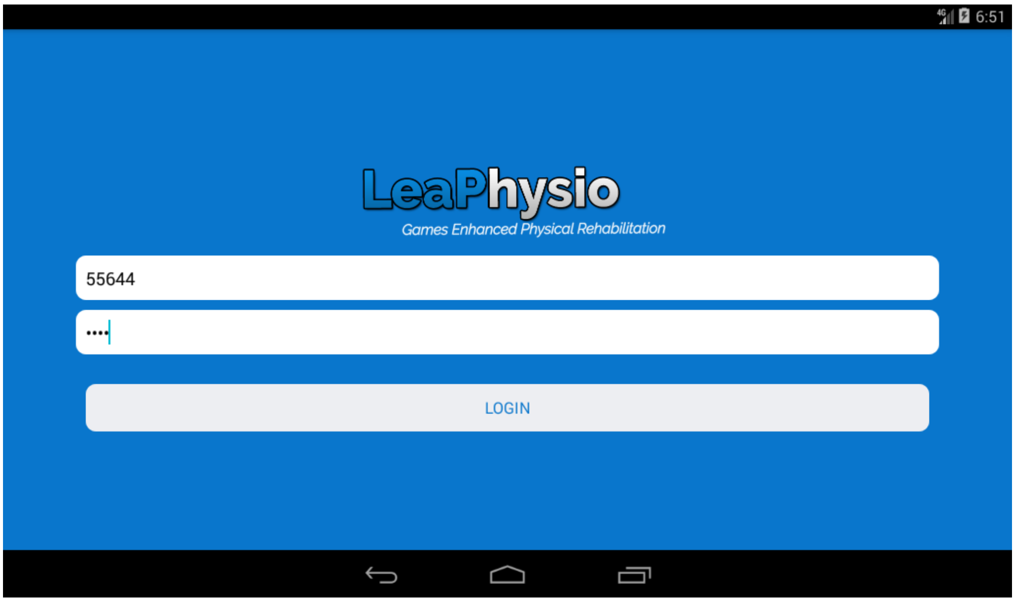
## 6.3. Design and Implementation

The interface is one of the most important things to keep in mind when developing a mobile application. An application without a simple and intuitive interface becomes less appealing to its end users. The navigation must be consistent to create the best possible user experience.

For an application to have a good user experience, the standards set by Android for the creation and development of the Graphical User Interface (GUI) have been followed and respected. The application was developed from the beginning, starting with the logo (see Figure 6.6).

Figure 6.6 – LeaPhysio Mobile Application Logo

When the application is started, the login activity is launched (see Figure 6.7). The user performs the authentication by entering their identification number and password in the corresponding text areas. In the following figure it's possible to verify the filled text areas.

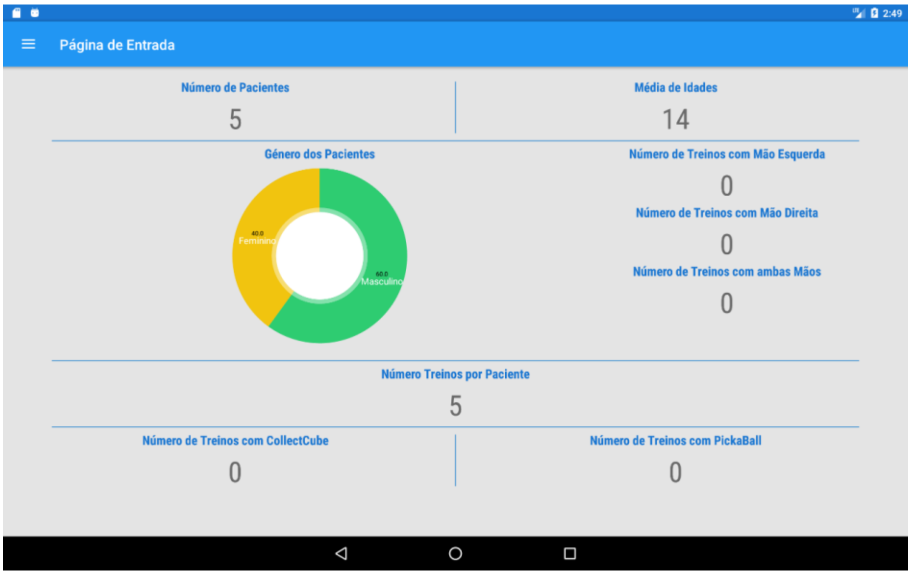


55644

••••

Figure 6.7 – LeaPhysio Mobile Application Login Activity Layout

The Physiotherapist's, homepage activity is launched after the authentication succeeds (see Figure 6.8). The user has access to information about patients, for example, how many patients are considered and how many training tasks are specified.



Número de Pacientes 5

Média de Idades 14

Género dos Pacientes

Número de Treinos com Mão Esquerda 0

Número de Treinos com Mão Direita 0

Número de Treinos com ambas Mãos 0

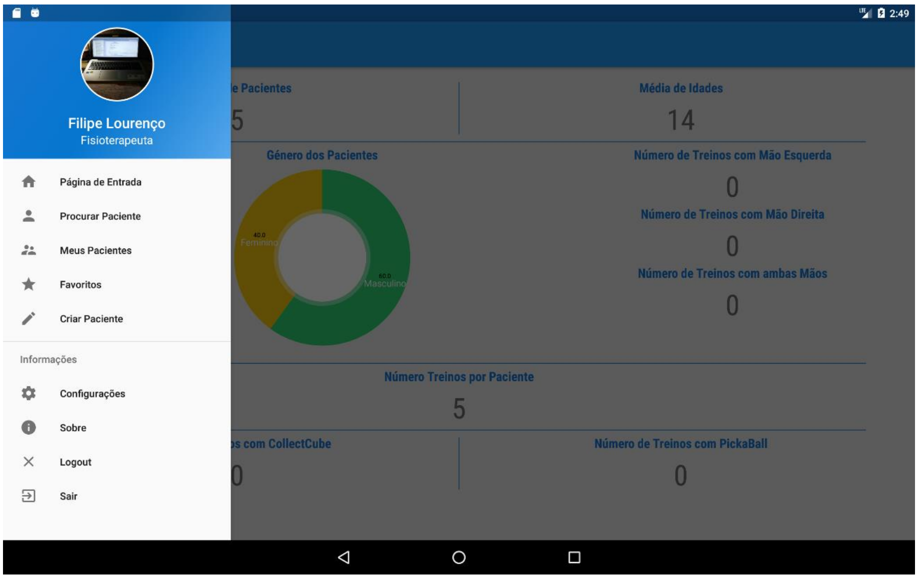
Número Treinos por Paciente 5

Número de Treinos com CollectCube 0

Número de Treinos com PickaBall 0

Figure 6.8 – LeaPhysio Mobile Application Homepage Activity Layout

In the upper left corner, there is a button to access the navigation menu, which has the main and initial options of the application (see Figure 6.9). The user can create new patients or search for patients, either through their name, identification number or QRCode present in LeaPhysio cards.



e Pacientes 5

Média de Idades 14

Género dos Pacientes

Número de Treinos com Mão Esquerda 0

Número de Treinos com Mão Direita 0

Número de Treinos com ambas Mãos 0

Número Treinos por Paciente 5

os com CollectCube 0

Número de Treinos com PickaBall 0

Figure 6.9 – LeaPhysio Mobile Application Navigation Drawer Layout

The Figure 6.10 shows the layout for the patient profile. The physiotherapist can perform various operations for a considered patient. Actions such view LeaPhysio card, generate PDF reports with information of the exercises that were already done, visualize data, view trainings, observations and create new trainings are also available for physiotherapists.

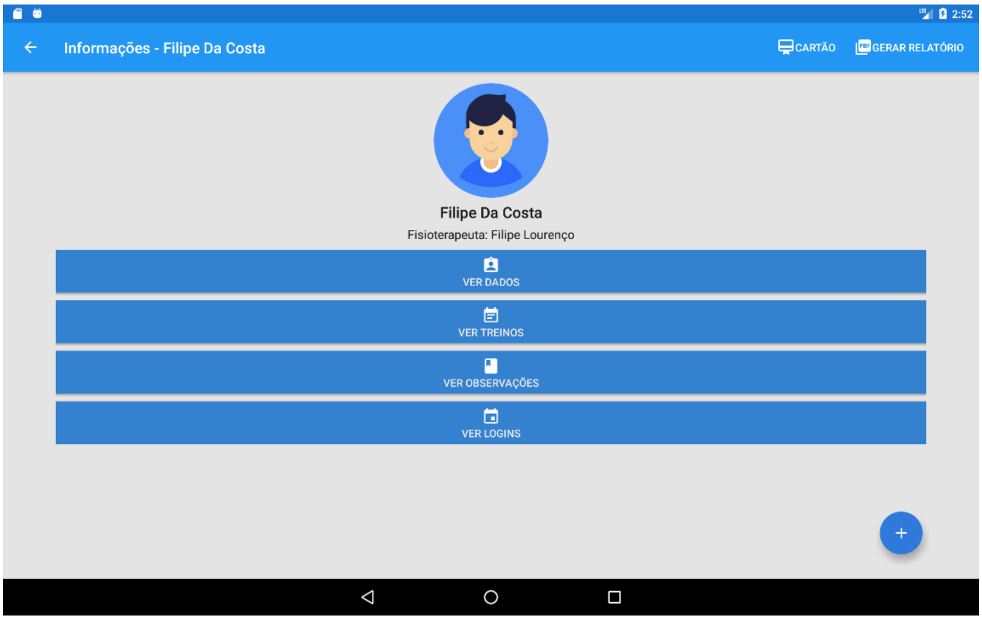
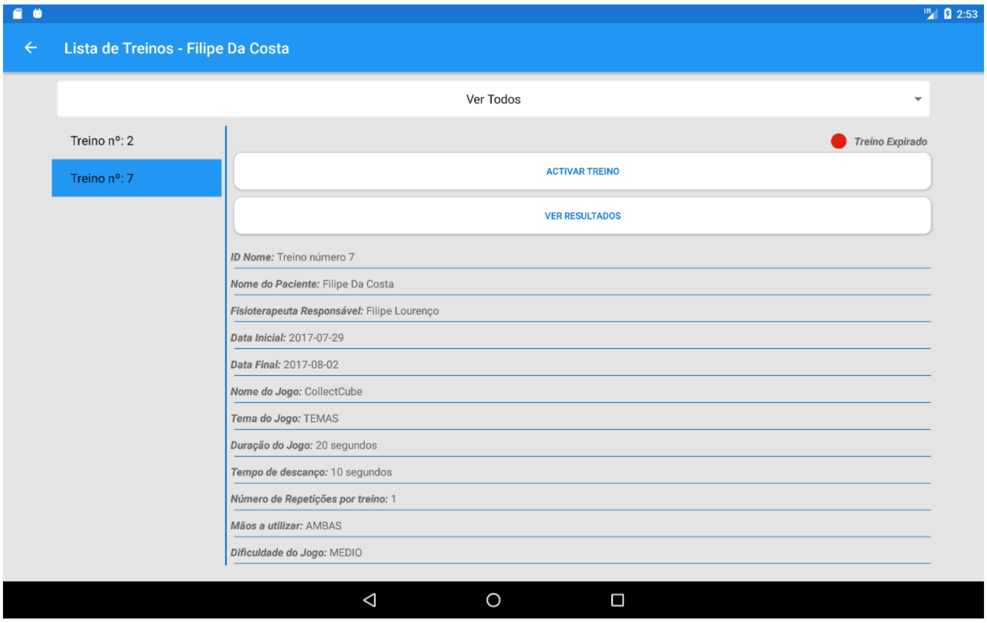


Figure 6.10 – LeaPhysio Mobile Application Patient Profile Activity Layout

Clicking on the "*Ver Treinos*" button opens a new activity that relates to training (see Figure 6.11). Is possible visualize the differences settings for each training, which start and end date each training, how many reps, which game and which hands to use are just some data that are visible. Can also check whether a training is active or expired, change existing settings, view results, and sort the existing trainings, for example by time order or view only the active trainings.



***ID Nome:*** Treino número 7

***Nome do Paciente:*** Filipe Da Costa

***Fisioterapeuta Responsável:*** Filipe Lourenço

***Data Inicial:*** 2017-07-29

***Data Final:*** 2017-08-02

***Nome do Jogo:*** CollectCube

***Tema do Jogo:*** TEMAS

***Duração do Jogo:*** 20 segundos

***Tempo de descanço:*** 10 segundos

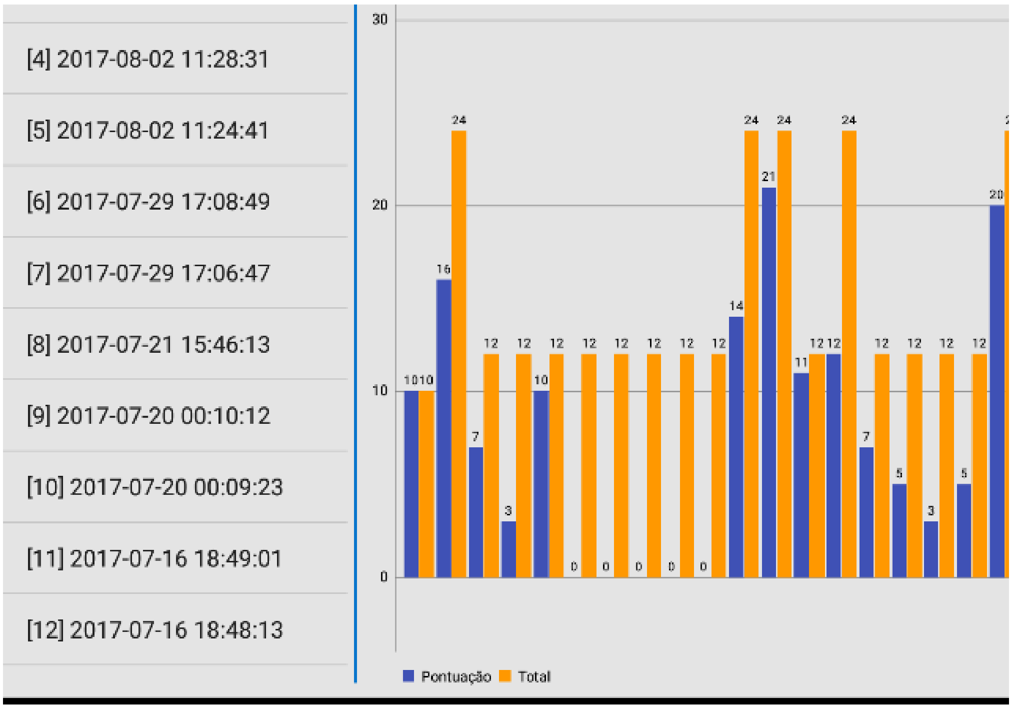
***Número de Repetições por treino:*** 1

***Mãos a utilizar:*** AMBAS

***Dificuldade do Jogo:*** MEDIO

Figure 6.11 – LeaPhysio Mobile Application Training Patient Profile Activity Layout

When selecting a training and clicking on the "*Ver Resultados*" button a new activity is opened if there are results from the execution of the training by the patient. The results are divided in score and coordinates. When activity related to results is opened, the type of result that is initially visible is the score. These are represented by bar graphs and the variables of each graph may change depending on the game (CollectCube or PickaBall Game) that the training has. Figure 6.12 shows results for a given training with the PickaBall Game.



[4] 2017-08-02 11:28:31

[5] 2017-08-02 11:24:41

[6] 2017-07-29 17:08:49

[7] 2017-07-29 17:06:47

[8] 2017-07-21 15:46:13

[9] 2017-07-20 00:10:12

[10] 2017-07-20 00:09:23

[11] 2017-07-16 18:49:01

[12] 2017-07-16 18:48:13

Figure 6.12 – LeaPhysio Mobile Application Score Results Activity Layout with all results

The activity layout consists of a list of results and a corresponding graph. The first element of the list, given by the name "*[0] Ver Todos*", encompasses all the results is a single chart, allowing an overview of the scores. The second element in the list is "*[1] Total*", averaging all variables (see Figure 6.13). Regardless of the game to which the results belong, these two elements are always present.

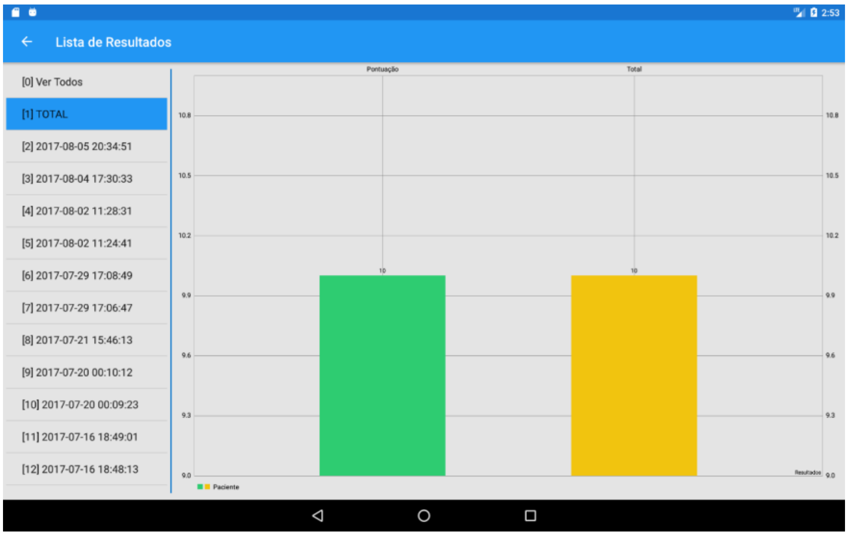


Figure 6.13 – LeaPhysio Mobile Application Score Results Activity Layout with average values

From the third element of the list, we can find the individual results obtained by executing the training plans. The name of the list element is represented by the date and time of the execution of the exercise by the patient and it’s through these that it’s possible to visualize the results in terms of coordinates. Looking at Figure 6.14 is possible verify the top of the graph there are two elements, a label indicating the name of the file that has the coordinates of the hands and a button to access the results.

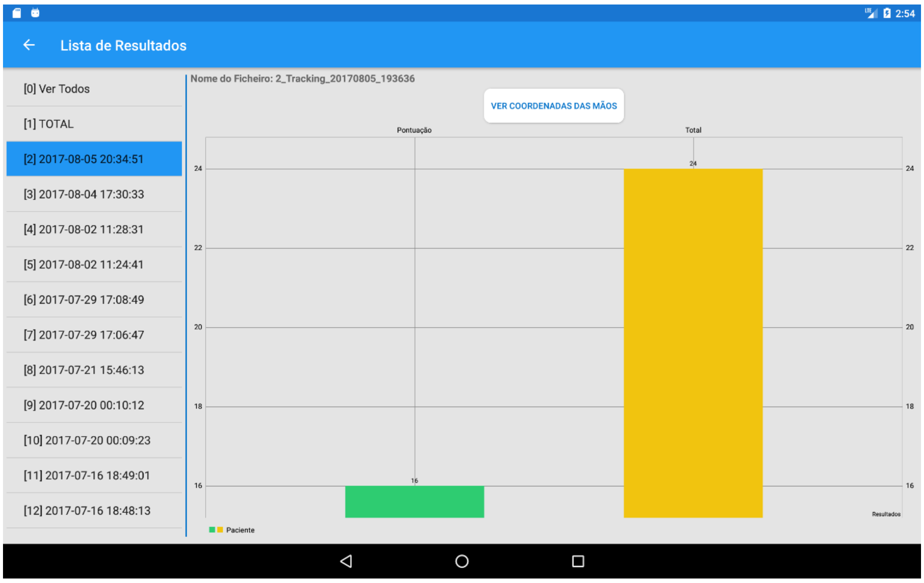


Figure 6.14 – LeaPhysio Mobile Application Score Results Activity Layout with individual result

When click the "*Ver coordenadas das mãos*" button, the coordinates activity is launched (see Figure 6.15). It consists of two panels, an upper one to view the selected data, select the points of interest for the clinical analysis or deselect all the data in the charts. The other panel concerns the representation of the selected points, these are presented using the line chart. To select data, click the "*Selecionar Dados*" button.

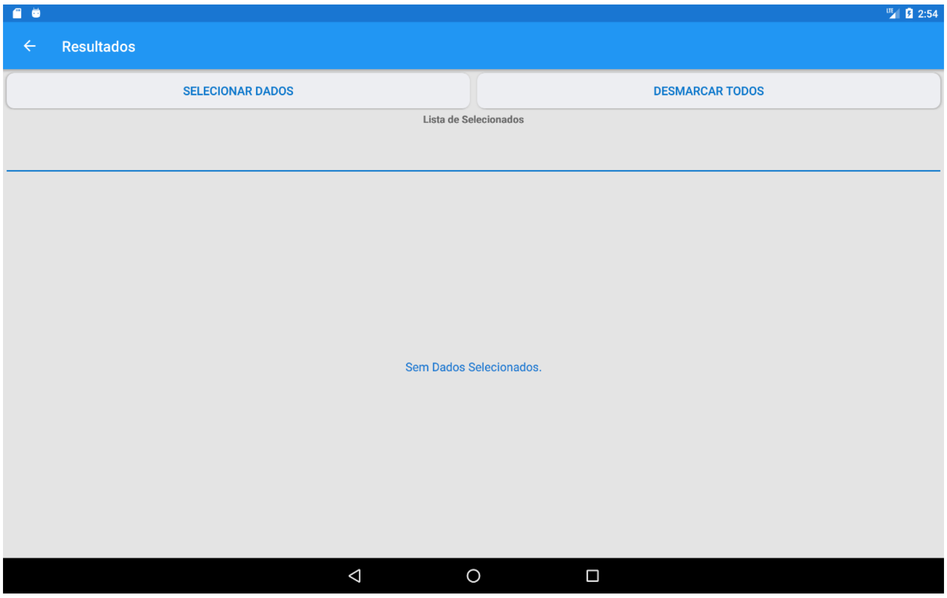


Figure 6.15 – LeaPhysio Mobile Application Coordinates Results Activity Layout

If there are selected data for different hands, they are presented in two graphs. In Figure 6.16 is possible verify the appearance of only having data selected for a hand and the aspect of having data for both.

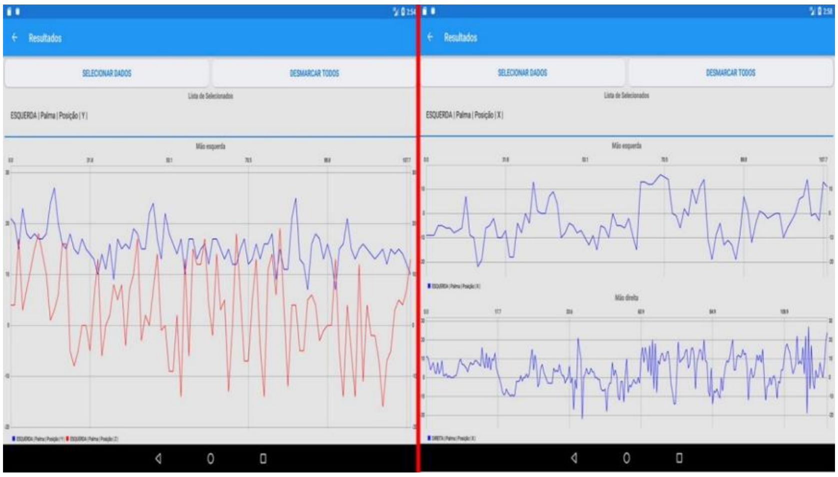


Figure 6.16 – LeaPhysio Mobile Application Coordinates Results Activity Layout with one and two charts

To remove a line from a chart, it’s necessary to search for that information in the list of selected data and press the element to remove. Zoom-in and zoom-out are allowed in the charts.

Figure 6.17 illustrates hierarchically the possible coordinates available for selection.

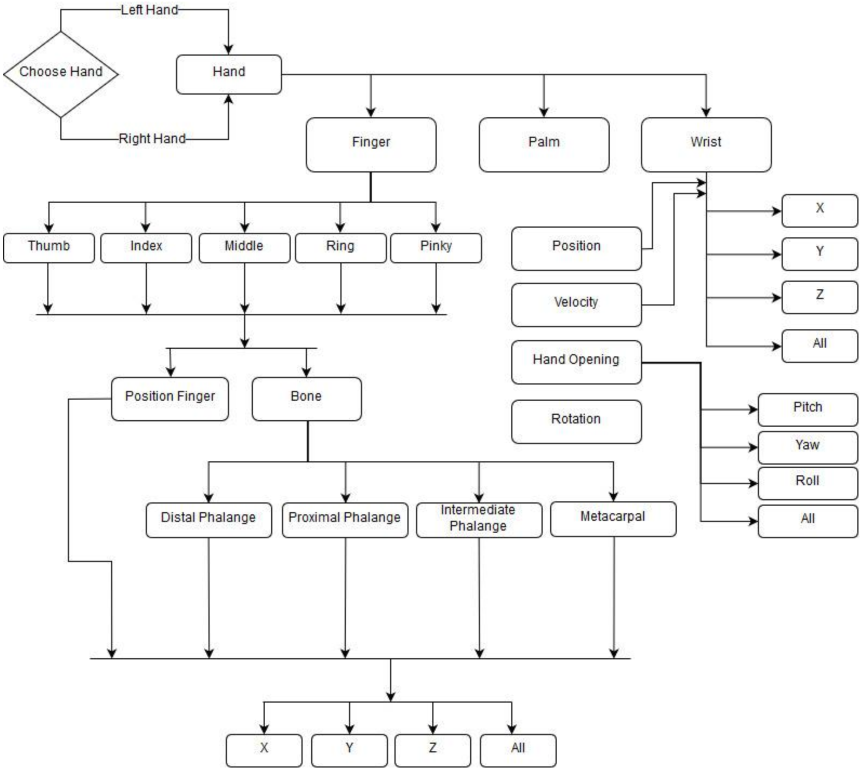


Figure 6.17 – Hierarchical diagram with the options to select coordinate data in the mobile application

The Appendix B, it's intended for the user manual of the mobile application, here it's described in greater detail the functionalities in the application, both for the Physiotherapist and for the Manager.

## 6.4. Application Main Features

**Application Authentication** – The *Physiotherapist* or *Manager* needs to be registered in the DB to use the application. Only these users can use the mobile application and need to use their identification number and password to sign in. New managers can only be added to the system manually, need to be created directly in the database (when the system is delivered to a clinic, the number of managers must be indicated to be added before the system is delivered).

**Search for Patient through QRCode** – The physiotherapist can search for a patient through the QRCode inserted in the patient's LeaPhysio card, to access the profile. The *Zxing library* [50] is used for this purpose. This library allows the reading of different types of barcodes and configurable in the visual aspects for the end user. Figure 6.18 shows an example of QRCode scan of the LeaPhysio card.

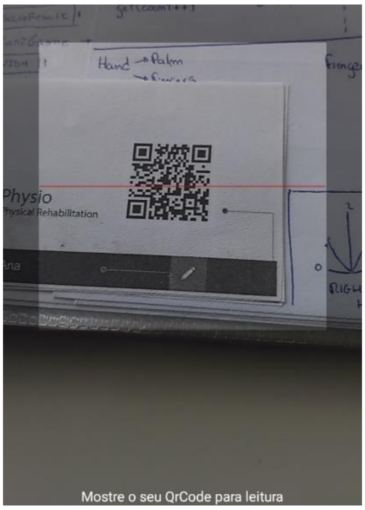


Figure 6.18 – Reading QRCode in mobile application

**Charts** – Patient training results are represented by line charts and bars. For this purpose, the *MPAndroidChart* library [51] is used. This library allows access to various types of graphics and configure them. The mobile application uses bar graphs to represent the results for the scores and line graphs for the results for the coordinates of the hands. These graphs allow the user to zoom-in and zoom-out for better observation of the data. A pie chart is also used to represent the percentage of patients in terms of their gender, male or female. Figure 6.19 illustrates some charts used in application.

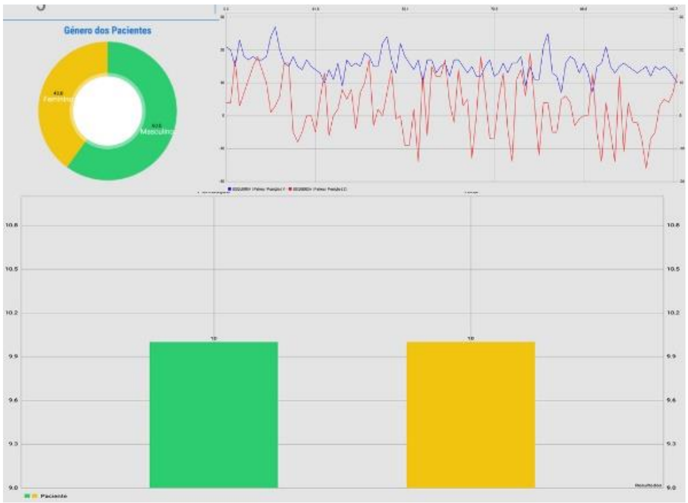


Figure 6.19 – MPAndroidChart library charts [51]

**Circular Profile Images** – To make the application more appealing to users, the profile images of the actors (patient, physiotherapist or manager) instead of having a square shape, was used a circular shape. For this, the library *CircleImageView* [52] was used. Figure 6.10 shows an example of a circular image.

**Encryption and Data Decryption** – Several fields in the database tables are encrypted to ensure protection of sensitive data, for example, the home address of the patient. The mobile application uses the Encryption library [53] to encrypt and decrypt data, for example, the mobile application wants to search the database for information about a specific clinic using the corresponding PHP script over an HTTP connection. If no anomaly occurs, the script executes the search in the database and returns the result obtained in a JSON format with the encrypted data. The mobile application reads this information and decrypts the data using the previously defined keys. Finally, the decryption information is presented to the user. To insert a record in the database, the application encrypts the information and executes the corresponding PHP script, sending the information through the POST method and is the script that tries to insert the encrypted data into the database. This library uses by default the Advanced Encryption Standard (AES) algorithm in Cipher Block Chaining (CBC) mode. Figure 6.20 illustrates the CBC encryption and decryption schemes.

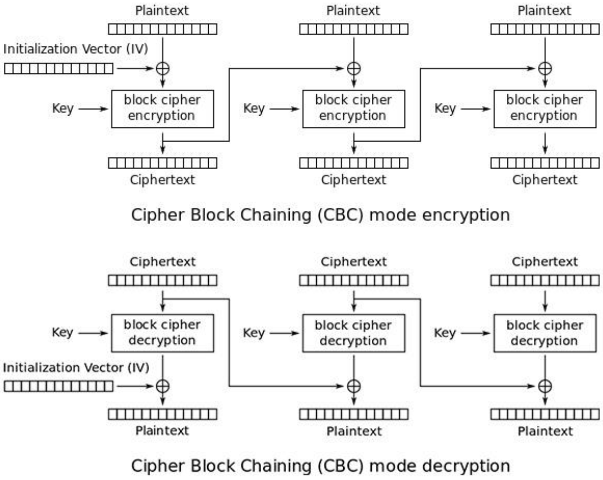
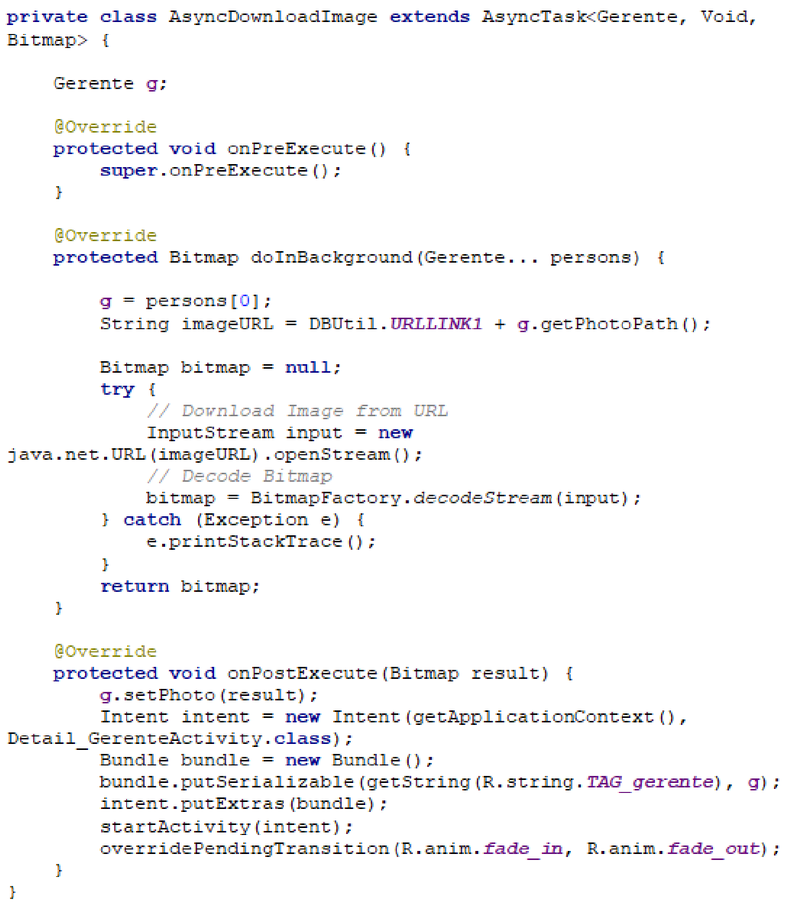


Figure 6.20 – CBC mode encryption and decryption

**HTTP Connection** – The application communicates with the database through HTTP requests. To perform this type of connection is used AsyncTask. This class is used to facilitate the processing of background operations and to use the data in the User Interface (UI). The following code illustrates an example of using an AsyncTask to download a Manager profile image.



```
private class AsyncDownloadImage extends AsyncTask<Gerente, Void,
Bitmap> {

    Gerente g;

    @Override
    protected void onPreExecute() {
        super.onPreExecute();
    }

    @Override
    protected Bitmap doInBackground(Gerente... persons) {

        g = persons[0];
        String imageURL = DBUtil.URLLINK1 + g.getPhotoPath();

        Bitmap bitmap = null;
        try {
            // Download Image from URL
            InputStream input = new
java.net.URL(imageURL).openStream();
            // Decode Bitmap
            bitmap = BitmapFactory.decodeStream(input);
        } catch (Exception e) {
            e.printStackTrace();
        }
        return bitmap;
    }

    @Override
    protected void onPostExecute(Bitmap result) {
        g.setPhoto(result);
        Intent intent = new Intent(getApplicationContext(),
Detail_GerenteActivity.class);
        Bundle bundle = new Bundle();
        bundle.putSerializable(getString(R.string.TAG_gerente), g);
        intent.putExtras(bundle);
        startActivity(intent);
        overridePendingTransition(R.anim.fade_in, R.anim.fade_out);
    }
}
```

Figure 6.21 illustrates the steps in an AsyncTask. In the application only the onPreExecute, theInBackground and onPostExecute methods are used.

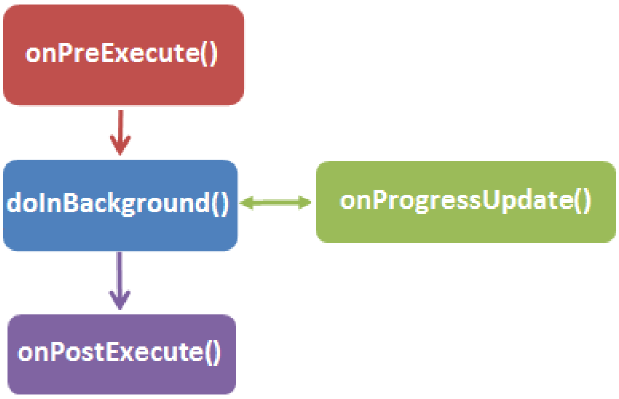


Figure 6.21 – AsyncTask method sequence

- The onPreExecute method always runs before the Thread starts, and is where messages are displayed to the user if necessary.
- The required method is doInBackground. This is responsible for all the heavy processing as it runs on a separate Thread. This is where the information is processed,

and the HTTP request is executed, and the user screen is not locked. The result is passed by parameter to the onPostExecute method.

- The onPostExecute method is the one that receives the return from doInBackground and is called using a Handler. Its execution takes place on the same Thread as the UI and it’s here that the final decision regarding the result obtained from the doInBackground is processed.

Appendix C explains in more detail not only the AsyncTask but also the main functions of the programming application.

# Chapter 7 – Results

## 7. Overview

This chapter demonstrates the experimental results associated with LeaPhysio system tests. Considering the ethical issue related with human research the tests were performed in a first phase with physiotherapists and healthy volunteers, the second phase will be carried out with patient in motor rehabilitation program in the framework of the research project financed by *Fundação para a Ciência e a Tecnologia*, TailorPhy – Smart Sensors and Tailored Environments in Physiotherapy, after research approval by Ethical Committee. The performed tests were conducted to get results and try to understand people's feedback when playing games in the Unity application or using the mobile application and to have information that can improve the serious game quality for future adoption by the physiotherapy clinics.

### 7.1. Results

In an intermediate phase of the development of the Unity application with the CollectCube some test sessions were conducted with people and the obtained results were reported in a publication indexed IEEEXplore [54]. This article compares the efficiency of a real environment with a virtual environment using the CollectCube game for the virtual scenario and the game replicated for the real scenario with real cubes and RFID technologies for cubes automatic counting and identification during the real game.

Several testing sessions were carried out in laboratory with a number of four healthy volunteers: three males and one female to evaluate the performance of the serious games for virtual reality and real scenarios. The healthy volunteers have played the serious games developed for both scenarios in different training sessions. The first session was focused on the serious game characterized by the virtual reality scenario. To evaluate the training effectiveness a temperature measurement procedure were performed at the end of each training session, the temperature being measured using a thermographic camera E60 from Flir. The volunteers played sessions of 30s, 1min, 1.5min and 2 min.

In order to extract the correlation between game score, and effort expressed by intensive muscle activity the scores for each game session was stored and later analyzed. The evolution of the temperature on the hand skin level, before and after two minutes training session based on serious game for virtual reality scenario, is presented in Figure 7.1.

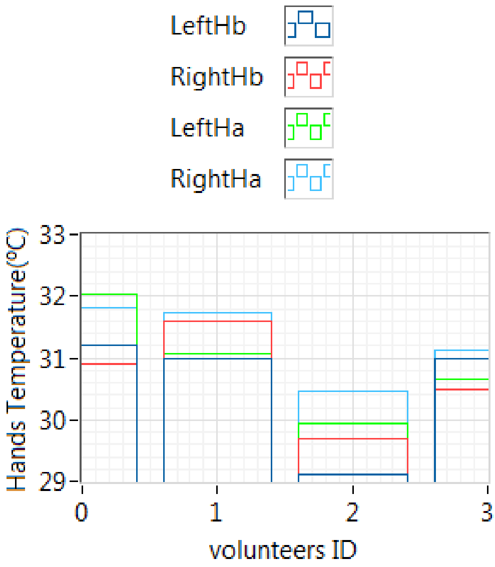


Figure 7.1 – Average temperature evolution of hand skin temperature before and after 2 min training

In Figure 7.1 it can be observed that the game practicing induced temperature variations on the level of hands' skin. The relation between learning and game scores is represented in Figure 7.2. It can be observed that some low performance was occurred for 1min training, while for 1.5min and 2min the players increase their performances according with their knowledge about this new game.

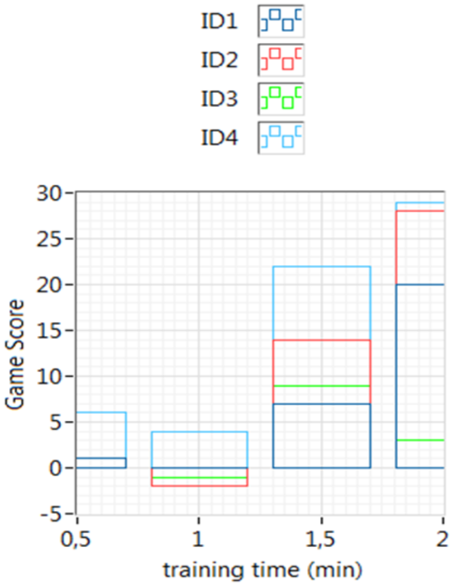


Figure 7.2 – Score Evolution for different training durations (IDi volunteer ID)

The comparison between the obtained scores by the participants when they are playing in virtual reality scenario or real scenario was carried out. The volunteers obtained higher scores at the beginning on serious game in real scenario while latter the obtained scores are near each other (ID3 volunteer: 5 for 1min serious game for VR scenario, 3 for 1 min serious game for real scenario).

Serious game effectiveness evaluation was also performed by comparing the temperature variation on the hands' skin when the user performs the serious game for real and for virtual scenario. The results obtained in one of the cased is presented in Figure 7.3, where it can be observed that for real scenario the values of average temperature after 1 min playing time is in general higher than obtained for VR scenario. The reduced number of tests doesn't permit to take final conclusion about the relation between the intensities of muscular activities registered during the two types of serious game for virtual and real scenario. It's expected that the user may play easily the video game than the real object based game.

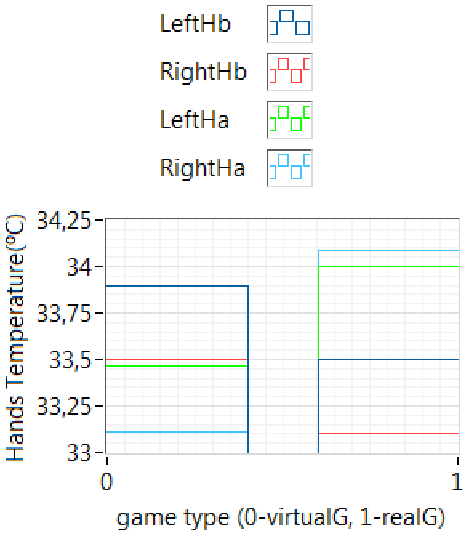


Figure 7.3 – Hand' Average temperature evolution for virtual training game and real training game (LeftHb, LeftHa- temperatures of left hand before and after training, RightHb, RightHa – temperatures of right hand after) for one volunteer

Preliminary results on usability tests of developed application were obtained after experimental tests organized at ISCTE were initiated. The tests were realized in the framework of TailorPhy research project and involved 8 software application evaluation by therapists and healthy volunteers. The LeaPhysio application was evaluated by 8 participants - 4 female occupational therapists (age range 29-31 years) and 4 healthy volunteers (1 female and 3 men, age range 25-48 years). All participants played the CollectCube game from the Unity application.

In the Unity application, the participant needed to login to the system using the LeaPhysio card, then play the CollectCube game with the COLOR thematic using two hands with a three-minute game time. At the end, hand and finger data were collected by the application for further analysis. Subsequently, the participant was asked to respond to a questionnaire.

The Android application was exclusively tested only with physiotherapists. A set of planned as following:

- Search for a Patient;
- View all training sessions and select active training;
- View the results of the hand joints;
- Select any existing data type, for example, X position of the palm of the hand.

Subsequently, the physiotherapist was asked to respond to an inquiry.

### 7.1.1. Hand Motion Analysis Results

At the end of the usability tests, the data was processed. The following graphs presents some of the data collected by the Unity application and processed in Microsoft Excel. The data collected are always described in relation to the time, the considered tests period being expressed by 3 minutes = 180 seconds.

Figure 7.4 represents the charts for the serious game played with both hands for two randomly chosen participants, with values referring to Palm Pinch Strength over time (seconds).

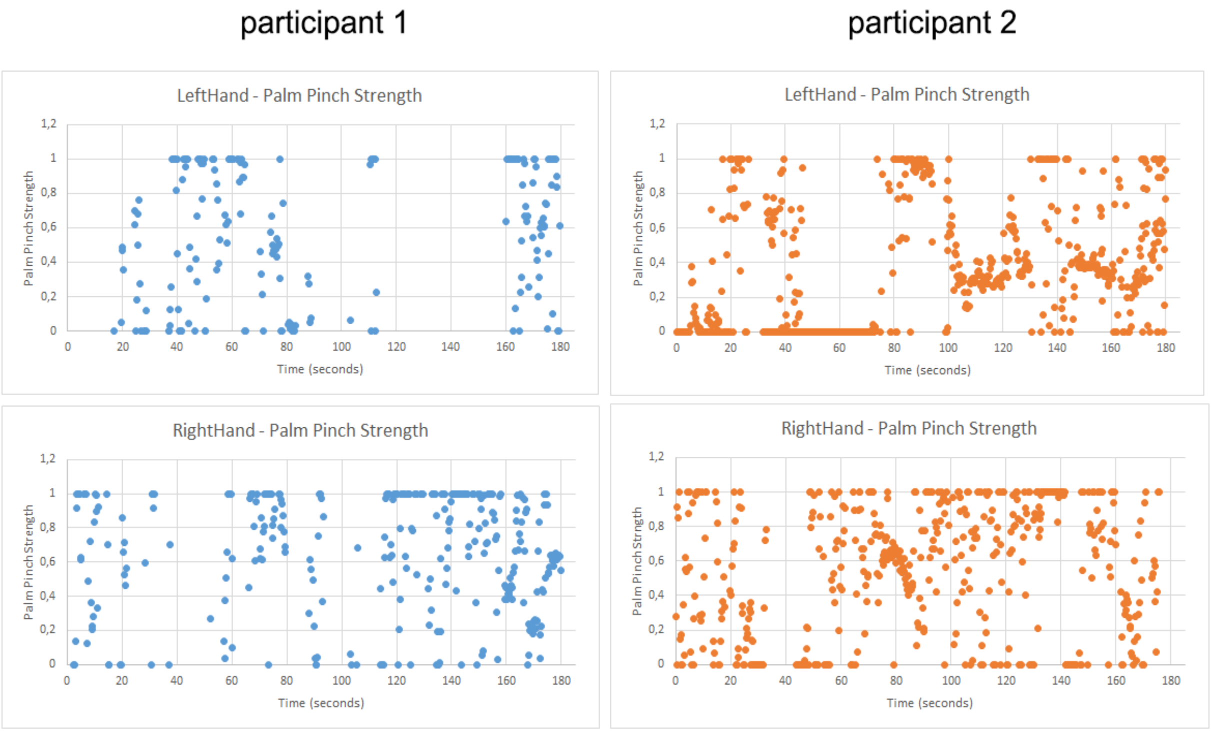


Figure 7.4 – Left and right hand charts of two participants, with values of Palm Pinch Strength

- *PinchStrength* – Indicates values between 0 and 1, where 0 corresponds to open hand and 1 to the closed hand;

The participant1 and the participant2 were with their right hand more often closed in relation to the left hand (the participant1, for a value of 1, has 29% in the left hand and 61% in the right hand), thus indicating that the right hand of both was more used to try to catch the game cubes. The right hand of both participants, have more values 1 than the left hand. It's also possible to verify that the participant2 has more data in the charts, the participant1 in certain parts of the game did not have his hands on the LMC detection field (participant2 used the hands for an average time of 160 seconds and participant1 for 77 seconds)

Figure 7.5 represents the charts associated with both hands usage of two chosen participants, with values referring to the Z position of the palm. The Z position refers to the depth axis, for example, during the game the participants need to move the hand in depth (Z axis) to place the cubes in the respective boxes.

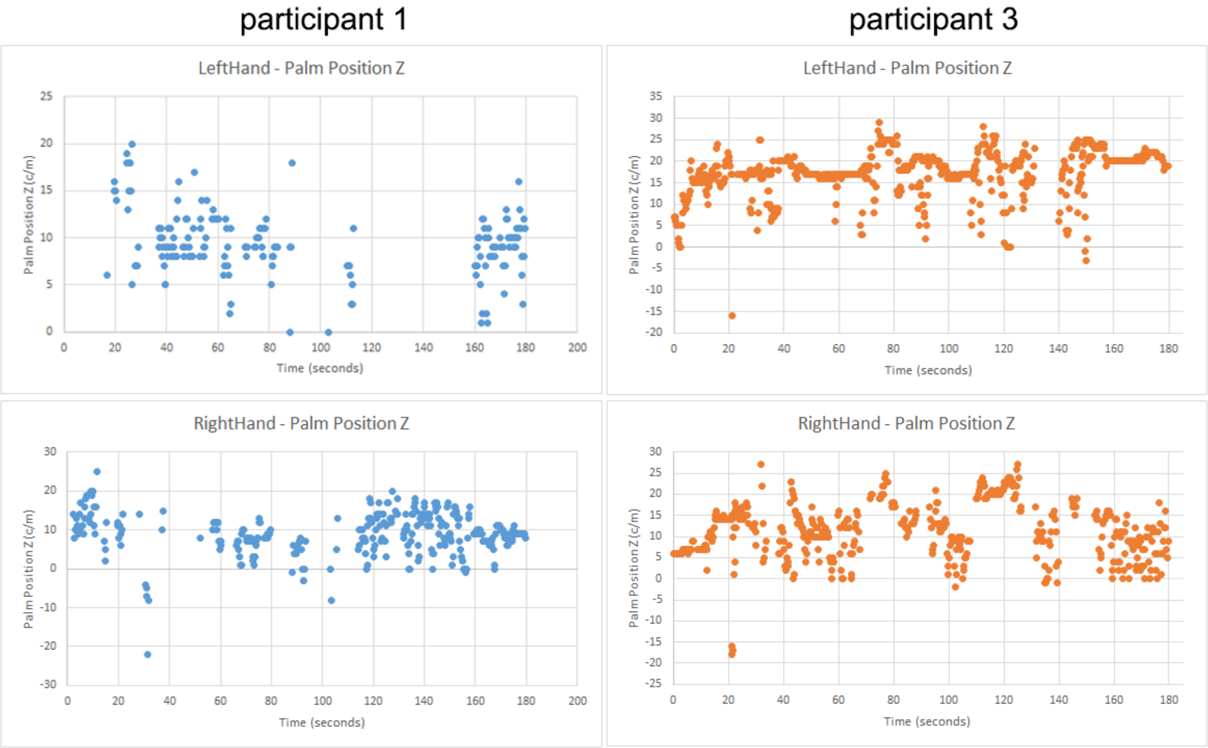


Figure 7.5 – Left and right hand charts of two participants, with values of Palm Position Z

The values for the position are expressed in cm. A value equal to 0 means that the participant has a hand over the LMC, values greater than 0 means that the hand is moved to the front (towards the boxes) and negative values means that the hand is moved to the back part. The participant3 obtained a better score for values greater than 0, compared with the participant1, meaning that the hands of participant3 was closer to the boxes. The participant1 used more the right hand, while the participant3 had a similar performance in both hands. The participant1 in the left hand has the most negative values over time.

Figure 7.6 represents the graphs of two chosen participants with values for palm velocity on the X axis. The velocity may indicate the participant's reaction to moving his or her hand to place a cube in the box. The units displayed for the velocity are cm/s.

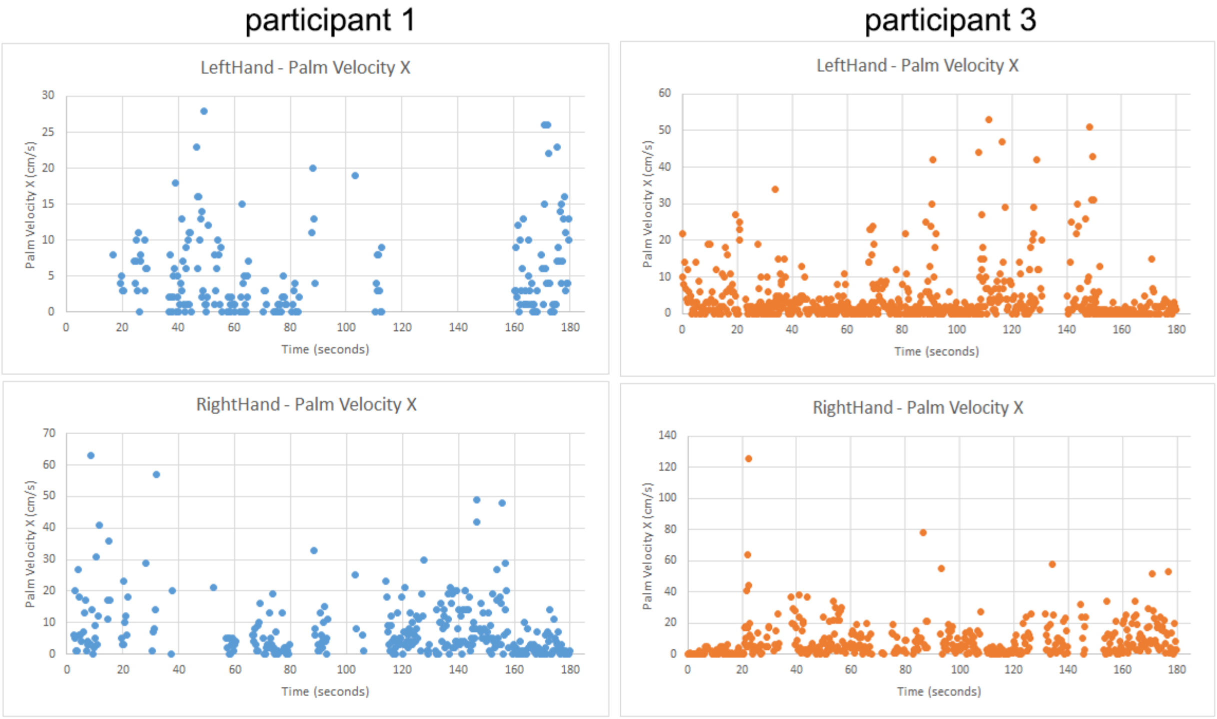


Figure 7.6 – Left and right hand charts of two participants, with values of Palm Velocity X

Both participants had similar velocities, values ranging from 0 to 20 cm/s. It’s also possible to verify that the participant3 was the one that obtained a greater velocity, about 125 cm/s, using the right hand.

Figure 7.7 represents the charts for two chosen participants, with values referring to the rotation of the hand on the Roll axis. Roll values are represented in degrees. Positive values indicate the hand was rotated counterclockwise, negative values indicate the hand was rotated clockwise. It’s possible to verify for both participants, that the left hand values are mostly positive, while the right hand values are negative. The human being is more inclined to turn the left hand counterclockwise and the right hand clockwise. If the recorded data doesn't comply with this "rule", it may indicate that the participant has difficulty to performing the hand rotation in the natural sense, for example, participant 1 has 94.3% of the values above zero in the left hand and 98.1% for negative values for the right hand, which indicates that it’s in accordance with the “rule”.

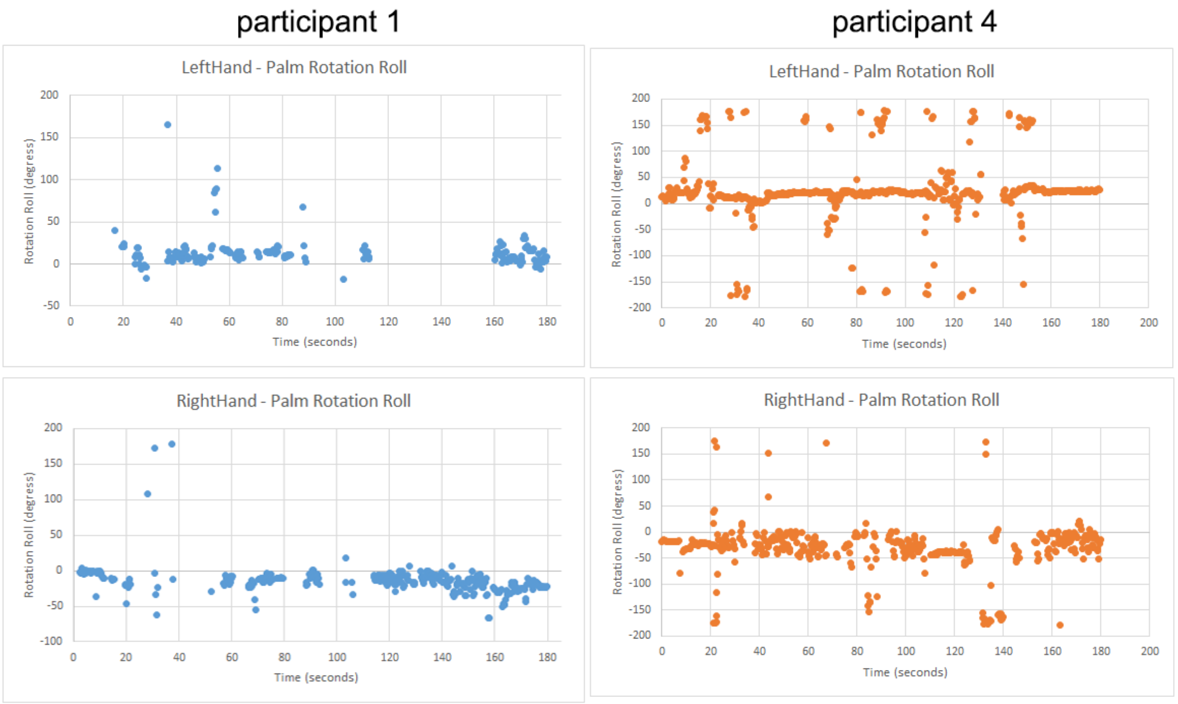


Figure 7.7 – Left and right hand charts of two participants, with values of Palm Rotation Roll

The previous charts have been rendered in Microsoft Excel, but also can be visualized on the mobile application. Figure 7.8 illustrates a comparison between the data processed in Excel and by mobile application.

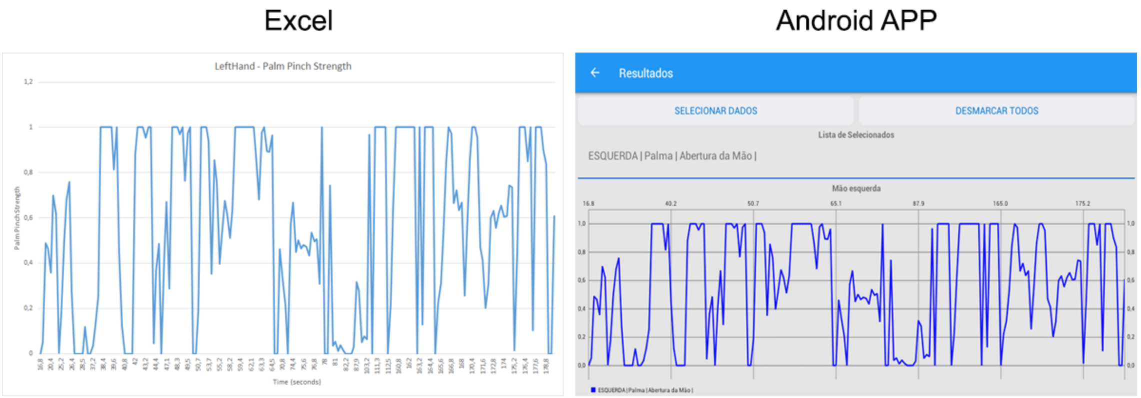


Figure 7.8 – Results processed in Excel and mobile application

### 7.1.2. QUIS Results

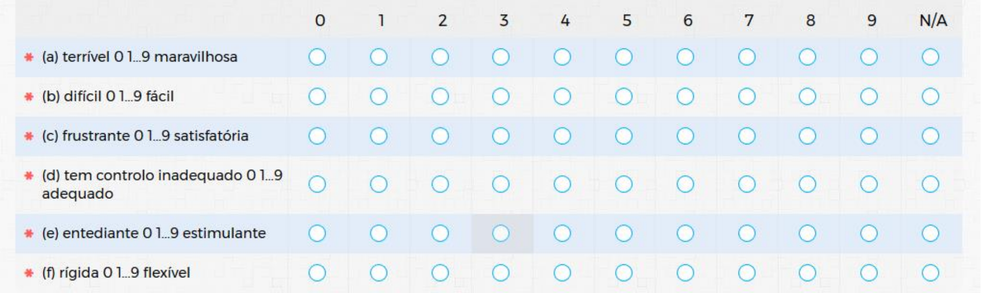
Evaluation of the developed software was realized. The usability tests were conducted using a Portuguese translation of the Questionnaire for User Interaction Satisfaction (QUIS) [55] [56]. This questionnaire developed by a multi-disciplinary team of researchers in the Human-Computer Interaction Lab (HCIL) at the University of Maryland at College Park, assess users' subjective satisfaction with specific aspects of the human-computer interface [57]. The QUIS is highly reliable across many types of interfaces. The QUIS questionnaire was included in the questionnaire developed by the team of research project, financed by *Fundação para a Ciência e a Tecnologia*, TailorPhy – Smart Sensors and Tailored Environment for Physiotherapy. The questions of TailorPhy questionnaire address the emotion that the application produced (by using validated Portuguese questionnaire of PANAS-VRP, Positive and Negative Affect Schedule [58]), the arousal, and the socio-demographic data. The questionnaire for therapists and the questionnaire applied to other type of participants in usability tests (healthy volunteers or physiotherapy patients) has 23 questions among these 13 (those related to QUIS, PANAS-VRP, and arousal scale) are equal in both questionnaires. The tests of developed hand rehabilitation game were realized with 8 participants (4 therapists and 4 healthy volunteers). In the preliminary results, the participants indicate that the application obtained a positive evaluation.

During the questionnaire participants were asked to list the three most positive and negative aspects of the application (Question 12 & 13). Some answers were selected.

- **Most positive aspects:**
  - "*Image code input is a fantastic idea*";
  - "*The concept of a game more directed to the motricity of the hand is very interesting and unusual*";
  - "*Constant muscle activation during application*";
- **Most negative aspects:**
  - "*Lack of some explanation of operation and use*";
  - "*Inaccessible points in the lower areas, mainly*";
  - "*It requires more cognitive skills, some users may feel frustrated more easily*".

The following charts reflect the participants' results on some questions in the questionnaire. Each question has several statements where the participant had to select an option from 0 to 9 that best reflects their opinion.

- **Question 6: Overall reactions to the software**



| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | N/A |
|---|---|---|---|---|---|---|---|---|---|---|---|
| * (a) terrível 0 1…9 maravilhosa | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (b) difícil 0 1…9 fácil | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (c) frustrante 0 1…9 satisfatória | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (d) tem controlo inadequado 0 1…9 adequado | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (e) entediante 0 1…9 estimulante | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (f) rígida 0 1…9 flexível | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |

Figure 7.9 – QUIS Question 6: Overall reactions to the software

Through the chart (see Figure 7.10) is possible to verify that all the affirmations obtained a good score, being the stimulating (*e*) affirmation that obtained better result, thus the participants consider the serious game as stimulant.

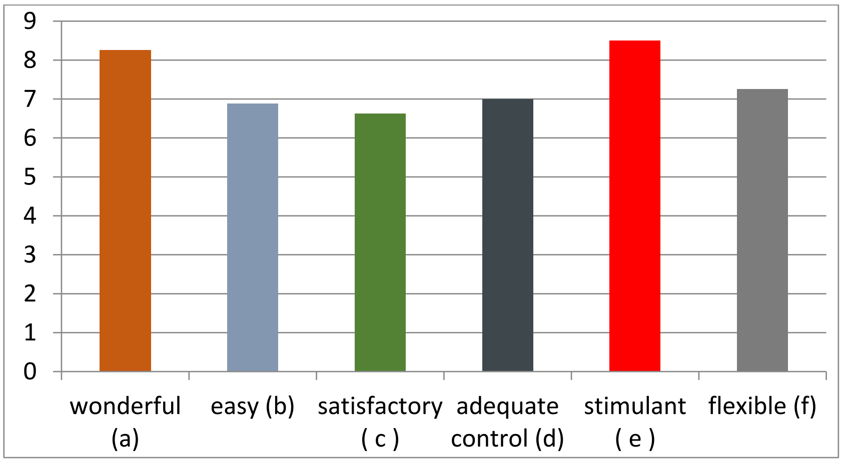
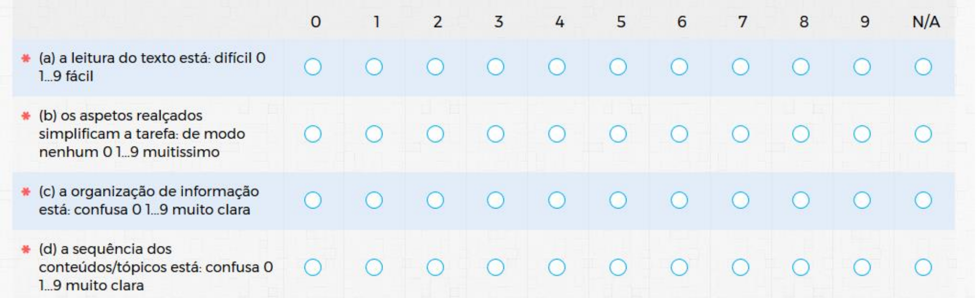


Figure 7.10 – QUIS Question 6 result

- **Question 7: Screen**



| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | N/A |
|---|---|---|---|---|---|---|---|---|---|---|---|
| * (a) a leitura do texto está: difícil 0 1…9 fácil | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (b) os aspetos realçados simplificam a tarefa: de modo nenhum 0 1…9 muitissimo | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (c) a organização de informação está: confusa 0 1…9 muito clara | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (d) a sequência dos conteúdos/tópicos está: confusa 0 1…9 muito clara | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |

Figure 7.11 – QUIS Question 7: Screen

Through the chart (see Figure 7.12) is possible to verify that all affirmations obtained a good score, with an average score of eight. The sequence of the contents (*d*) according to the participants, is very clear and the information is organized (*c*).

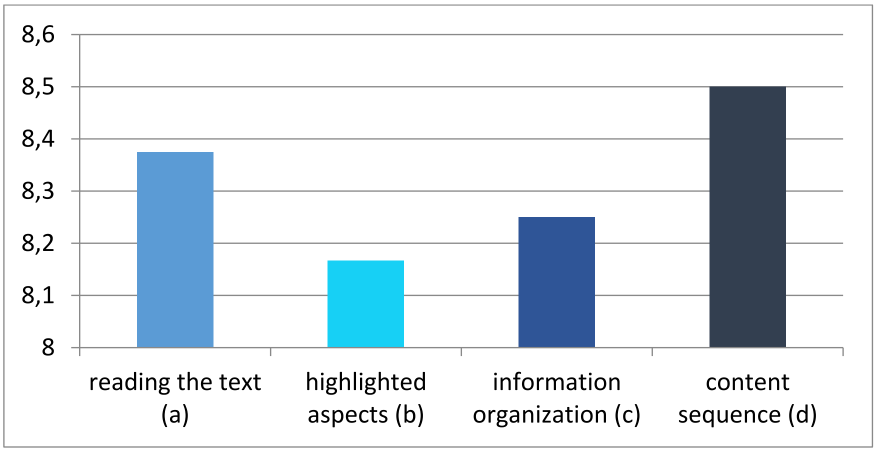
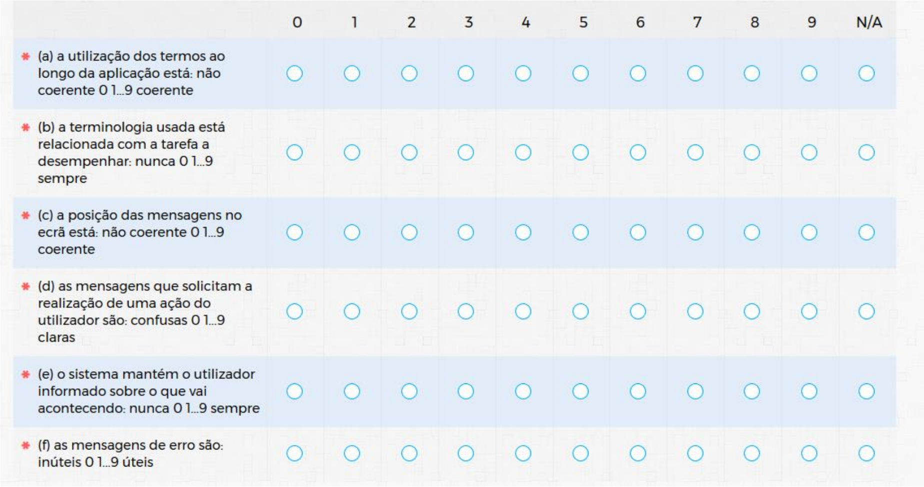


Figure 7.12 – QUIS Question 7 result

- **Question 8: Terminology and system information**



| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | N/A |
|---|---|---|---|---|---|---|---|---|---|---|---|
| * (a) a utilização dos termos ao longo da aplicação está: não coerente 0 1...9 coerente | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (b) a terminologia usada está relacionada com a tarefa a desempenhar: nunca 0 1...9 sempre | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (c) a posição das mensagens no ecrã está: não coerente 0 1...9 coerente | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (d) as mensagens que solicitam a realização de uma ação do utilizador são: confusas 0 1...9 claras | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (e) o sistema mantém o utilizador informado sobre o que vai acontecendo: nunca 0 1...9 sempre | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (f) as mensagens de erro são: inúteis 0 1...9 úteis | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |

Figure 7.13 – QUIS Question 8: Terminology and system information

Through the chart (see Figure 7.14), no affirmation has scored less than eight values. The participants consider that the position of messages on the screen are consistent (*c*) and that the terminology used is almost always related to the task to be performed (*d*).

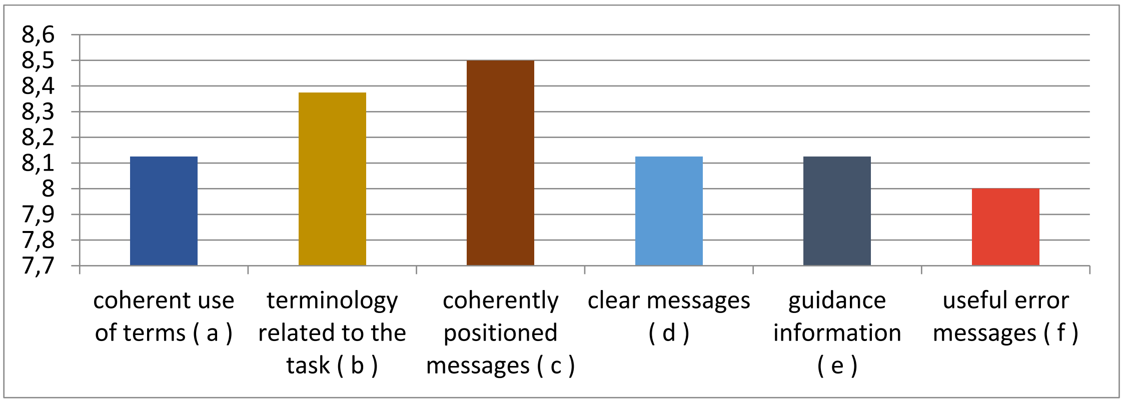
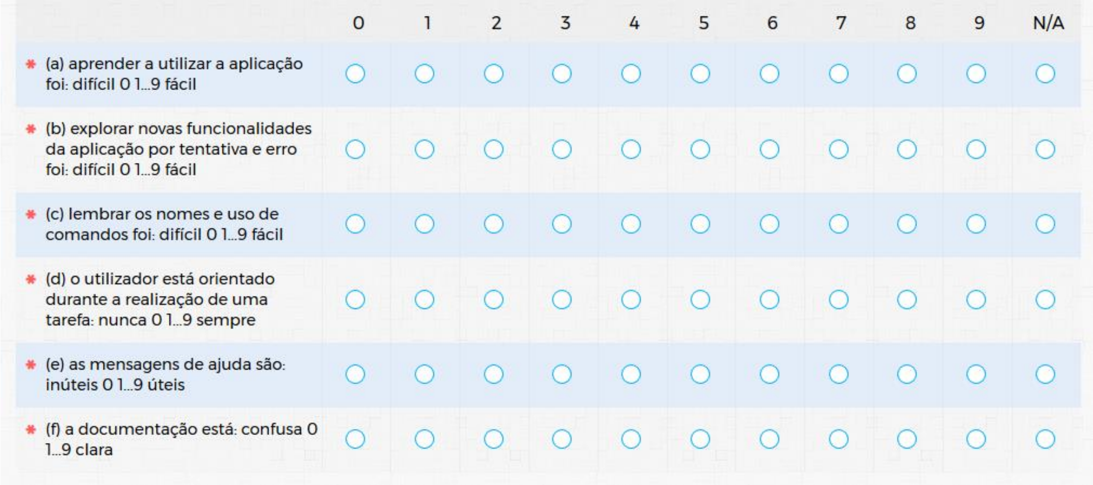


Figure 7.14 – QUIS Question 8 result

- **Question 9: Learning**



| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | N/A |
|---|---|---|---|---|---|---|---|---|---|---|---|
| * (a) aprender a utilizar a aplicação foi: difícil 0 1...9 fácil | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (b) explorar novas funcionalidades da aplicação por tentativa e erro foi: difícil 0 1...9 fácil | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (c) lembrar os nomes e uso de comandos foi: difícil 0 1...9 fácil | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (d) o utilizador está orientado durante a realização de uma tarefa: nunca 0 1...9 sempre | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (e) as mensagens de ajuda são: inúteis 0 1...9 úteis | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (f) a documentação está: confusa 0 1...9 clara | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |

Figure 7.15 – QUIS Question 9: Learning

Through the chart (see Figure 7.16), it's possible view that factors such as easy to explore (*b*) and always guidance (*d*) were obtained a lower score when compared with the others. The participants reported that the application is easy to remember (*c*) and it's not difficult to memorize the commands of use.

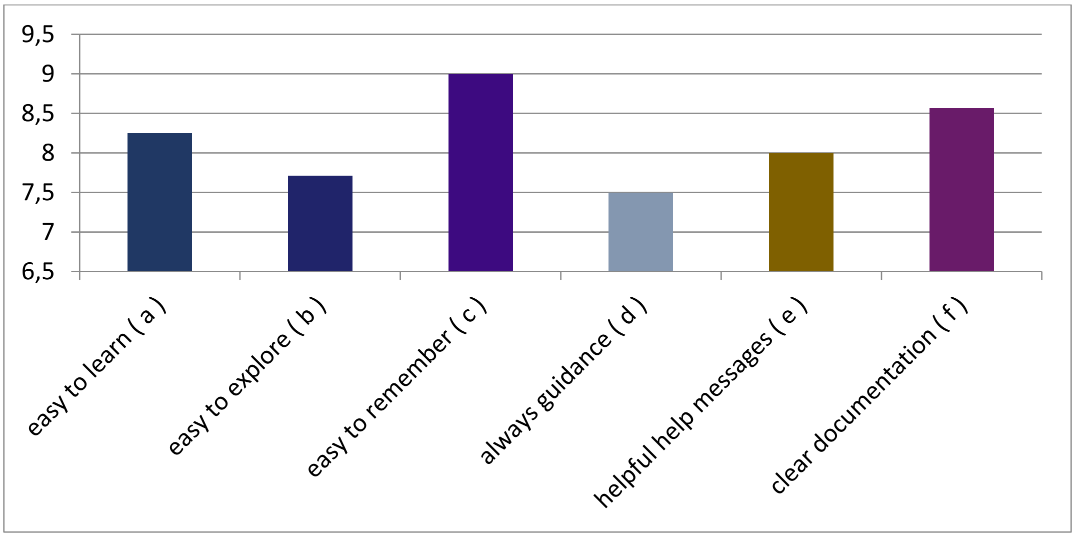
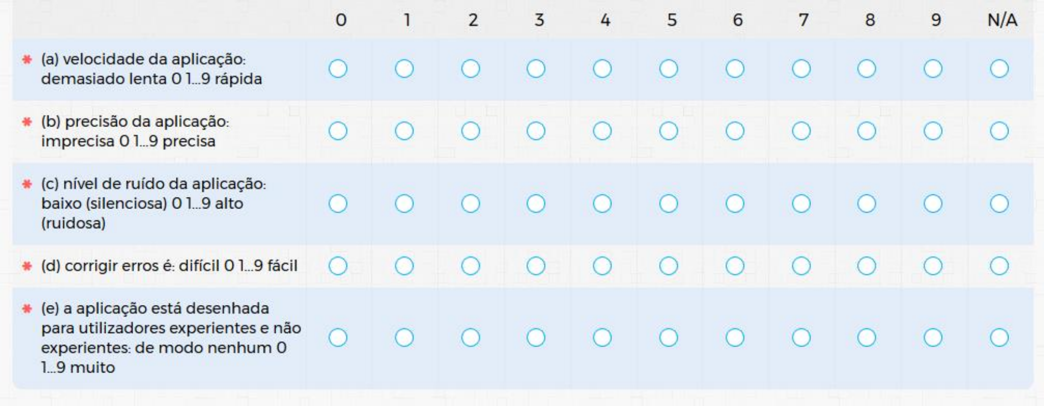


Figure 7.16 – QUIS Question 9 result

- **Question 10: System capabilities**



| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | N/A |
|---|---|---|---|---|---|---|---|---|---|---|---|
| * (a) velocidade da aplicação: demasiado lenta 0 1...9 rápida | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (b) precisão da aplicação: imprecisa 0 1...9 precisa | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (c) nível de ruído da aplicação: baixo (silenciosa) 0 1...9 alto (ruidosa) | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (d) corrigir erros é: difícil 0 1...9 fácil | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (e) a aplicação está desenhada para utilizadores experientes e não experientes: de modo nenhum 0 1...9 muito | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |

Figure 7.17 – QUIS Question 10: System capabilities

Participants were mentioned that the velocity (*a*) of the application is too fast, with a value of 7.75 (in a scale from 1 to 10) (see Figure 7.18). They were also mentioned that the application was not thoroughly designed for experienced users (*e*).

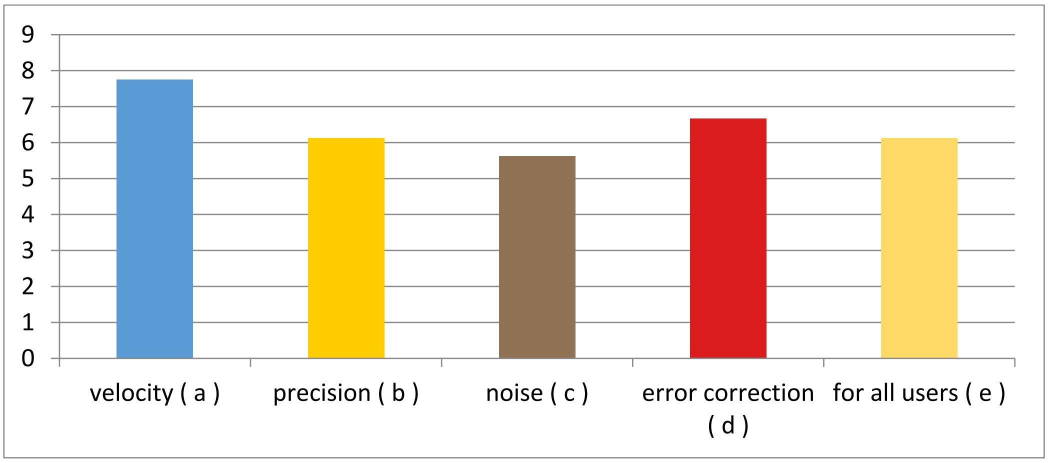
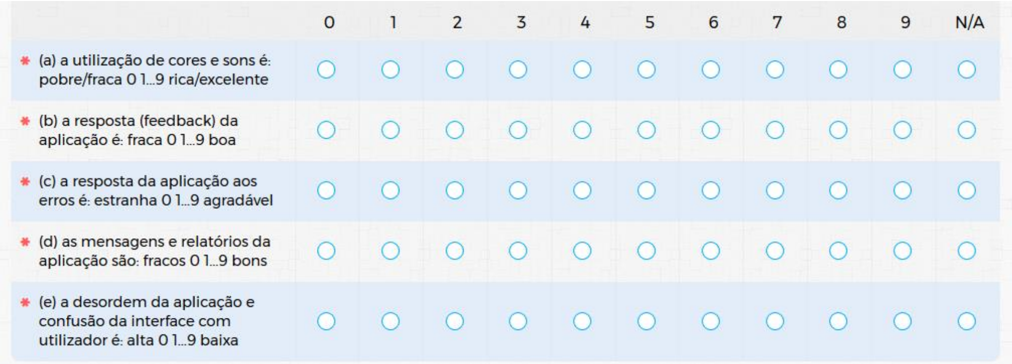


Figure 7.18 – QUIS Question 10 result

- **Question 11: Usability and user interface**



| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | N/A |
|---|---|---|---|---|---|---|---|---|---|---|---|
| * (a) a utilização de cores e sons é: pobre/fraca 0 1...9 rica/excelente | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (b) a resposta (feedback) da aplicação é: fraca 0 1...9 boa | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (c) a resposta da aplicação aos erros é: estranha 0 1...9 agradável | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (d) as mensagens e relatórios da aplicação são: fracos 0 1...9 bons | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| * (e) a desordem da aplicação e confusão da interface com utilizador é: alta 0 1...9 baixa | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |

Figure 7.19 – QUIS Question 11: Usability and user interface

Through the chart (see Figure 7.20), participants find that the use of the interface color and the sound level (*a*) is rich. The participants give a good feedback (*b*) and most participants appreciated the developed interface (*e*).

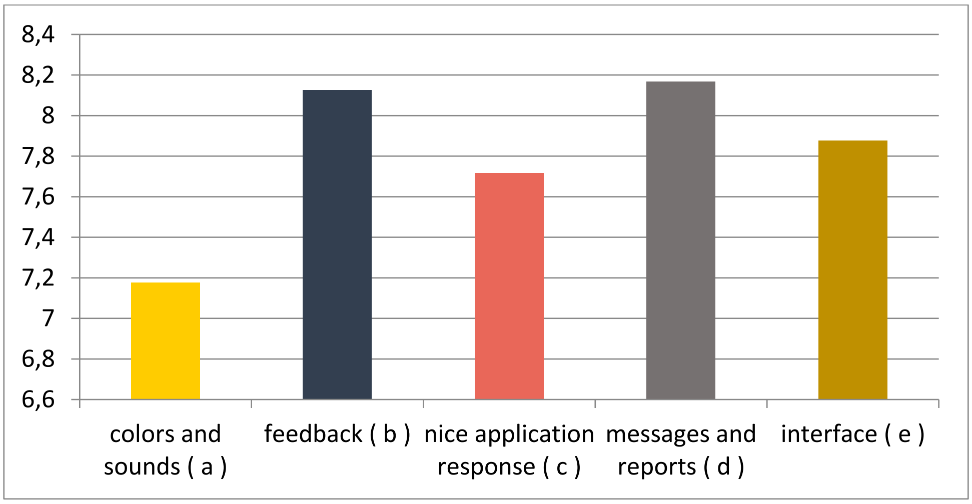


Figure 7.20 – QUIS Question 11 result

# Chapter 8 – Conclusions and Future Work

## 8. Overview

This chapter presents the conclusions and future work of the project that was presented in the previous chapters.

### 8.1. Conclusions

In this dissertation is presented a system that was developed to be used in the physiotherapy area, namely motor rehabilitation of hands and fingers. The designed technology is aiming to help professionals in this area to improve their work by effective monitoring of patients and objective assessment of the results of physical therapy sessions. The system provides two main tools, an application that allow patients to perform therapy exercises by playing the games and another for the physiotherapists to visualize the results and manage the patient's. All the most relevant information as scores performed in the trainings or hand coordinates, are stored on a remote server, allowing communication between the two applications.

Regarding the development phase of the project, a study was initially undertaken to understand how serious games should be created and adapted to the needs and preferences of each patient. Subsequently, the serious games were developed to respond to this needs and preferences. The serious game was developed through the Unity 3D game engine and the Leap Motion Controller was considered as the natural user interface for user serious game interaction. This controller detects hands and their movement and transmit the information to the computer through USB connection. Based on the provided SDK the hand and finger information such finger coordinates, hand speeds, hand positions, finger translations or rotation among others may be retrieved. Patient data, training plan, and results (scores and coordinates) are stored on a remote server. Most of the data is stored in the MySQL database, while files, for example, LeaPhysio QRCode cards or coordinate files are stored on the server disk. Communication between applications and database is performed through the developed PHP scripts. These scripts read and write to the database efficiently and are also used to generate Electronic Health Record (ERH). To complement the work, an Android application was developed to allow the physiotherapists and clinic manager to manage the different resources. The physiotherapist has the function of creating patients, creating and editing training plans, but also allow the visualization of the different results for future analysis. The clinic manager has the main function of registering the physiotherapists in the system.

The LMC prove to be a very useful controller for extracting information about the hands of the user in unobtrusive way and its usage may be fundamental for patients during the hand rehabilitation process. Throughout the process of developing serious games, different tests were conducted with healthy volunteers including physiotherapists in order to extract information about the usability and also to receive suggestions that can be used to improve the serious games scenario and the whole LeaPhysio system. It was possible to verify that the games had a greater adhesion to younger age group. Whenever they played it was possible to visualize their enthusiasm, causing people to try again to increase the score in the game, and also for novel experience that LMC provides in Virtual Reality scenarios.

## 8.2. Future work

It was a long project, full of challenges and still has many parts that can be improved in the future. Thus, such future work, can be mentioned new features and improvements that can be done to improve the capabilities of the system as following:

- Perform an extended study on regarding the selection of the movements that are more appropriate for finger and hand rehabilitation. Through the study, will be performed new games with different themes, allowing a greater variety of training plans;
- Create a virtual assistant for the Unity application. An assistant who teaches the patient how to play a game, when it comes to the first time, or when not performing motion in appropriate mode according with the game goals;
- Make serious games even more adaptable to patient characteristics increasing the game personalization;
- Perform an improved, extensive and adapted ERH according to each patient and the needs of physiotherapists;
- On the mobile application level, develop new algorithms to automatically give information if the patient is having improvements in performed movements, correlating data from previous rehabilitation sessions and informing if the games are taking effect.

Despite all the improvements that can be made to the project, the solution already implemented represents a valid and ready solution to be used in any physiotherapy clinics to assist people in the motor rehabilitation of the hands and fingers, but also for the physiotherapists to have a tool to analyze the results of rehabilitation sessions in objective way.

## References

Here are described the references used in the dissertation.

[1] "Ciências da saúde," [Online]. Available: https://pt.wikipedia.org/wiki/Ciências_da_saúde. [Accessed September 2017].

[2] M. Shaughnessy and B. Resnick, "Testing a Model of Post-Stroke Exercise Behaviour," *Rehabilitation Nursing,* vol. 31, pp. 15-21, 2006.

[3] D. Lloyd-Jones, R. Adams, T. Brown, M. Carnethon, S. Dai and G. D. Simone, "Heart disease and stroke statistics—2010 update: A report from the American Heart Association," Circulation, 2010.

[4] C. Lang, J. MacDonald and C. Gnip, "Counting repetitions: An observational study of outpatient therapy for people with hemiparesis post-stroke," *Journal of Neurologic Physical Therapy,* vol. 30, pp. 209-222, 2006.

[5] Y.-J. Chang, S.-F. Chen and J.-D. Huang, "A Kinect-based system for physical rehabilitation: A pilot study for young," *Research in Developmental Disabilities,* vol. 32, pp. 2566-2570, 2011.

[6] "Leap Motion," Leap Motion, [Online]. Available: https://www.leapmotion.com/. [Accessed September 2017].

[7] "BreakAway Games - Serious Games Developer," BreakAway Games, [Online]. Available: http://www.breakawaygames.com/. [Accessed September 2017].

[8] "The Incredible Adventures of the Amazing Food Detective," Kaiser Permanente, [Online]. Available: https://healthy.kaiserpermanente.org/static/health/en-us/landing_pages/afd/index.htm. [Accessed September 2017].

[9] "Xbox Fitness," Xbox, [Online]. Available: http://www.xbox.com/pt-PT/xbox-one/games/xbox-fitness?xr=shellnav. [Accessed July 2017].

[10] "Kinect for Xbox One | Xbox," Xbox, [Online]. Available: https://www.xbox.com/en-US/xbox-one/accessories/kinect. [Accessed September 2017].

[11] J. Wiemeyer and A. Kliem, "Serious games in prevention and rehabilitation - a new panacea for elderly people?," *European Group for Research into Elderly and Physical Activity,* vol. 9, pp. 41-50, 2012.

[12] B. J. W, M. M. D. J and C. D. K, "Optimising engagement for stroke rehabilitation using serious," *The Visual Computer,* vol. 25, pp. 1085-1099, 2009.

[13] J. Deutsch, D. Robbins, J. Morrison and P. G. Bowlby, "Wii-Based Compared to Standard of Care Balance," *Virtual Rehabilitation,* pp. 117-120, 2009.

[14] "Motion Capture for Biomechanics and Sports Science | VICON," VICON, [Online]. Available: https://www.vicon.com/motion-capture/biomechanics-and-sport. [Accessed October 2017].

[15] D. Jack, R. Boian, A. Merians, S. V. Adamovich and M. Tremaine, "Virtual Reality-Based Exercise Program for Stroke Rehabilitation," *Proceedings of the 4th international ACM conference on assistive technologies* , pp. 56-63, 2000.

[16] "CyberGlove Systems LLC," CyberGlove Systems, [Online]. Available: http://www.cyberglovesystems.com/. [Accessed September 2017].

[17] R. Unnikrishnan, K. Moawad and R. R. Bhavani, "A Physiotherapy Toolkit using Video Games and Motion Tracking Technologies," *Global Humanitarian Technology Conference: South Asia Satellite (GHTC-SAS) IIE,* pp. 90-95, 2013.

[18] "Tux Racer," Sunspire Studios, [Online]. Available: http://tuxracer.sourceforge.net/. [Accessed July 2017].

[19] "Wii Remote (Accessory)," Giant Bomb, [Online]. Available: https://www.giantbomb.com/wii-remote/3000-13/. [Accessed July 2017].

[20] M. Rodgers, "Wearable Sensors & Systems with Application in Rehabilitation," [Online]. Available: http://www.medicalautomation.org/wp-content/uploads/2012/11/Rodgers_Sensors_hopkins.pdf. [Accessed July 2017].

[21] M. McNeill, L. Pokluda, S. McDonough and J. Crosbie, "Following stroke," *IEEE International Conference on Man and Cybernetics,* p. 2783–2789, 2004.

[22] J. Decker, H. Li, D. Losowyj and V. Prakash, "Wiihabilitation: Rehabilitation of Wrist Flexion and Extension Using a Wiimote-Based Game System," *Research Journal,* 2009.

[23] K. Khoshelham and S. O. Elberink, "Accuracy and Resolution of Kinect Depth Data for Indoor Mapping Applications," *Sensors,* vol. 12, pp. 437-1454, 2012.

[24] "Unity - Game Engine," Unity, [Online]. Available: https://unity3d.com/pt/. [Accessed July 2017].

[25] N. M. S. Duarte, "KSGphysio – Kinect Serious Game for Physiotherapy," Departamento de Ciências e Tecnologias da Informação, ISCTE-IUL, 2014.

[26] "SQLite Home Page," [Online]. Available: http://www.sqlite.org/. [Accessed July 2017].

[27] "MySQL," Oracle, [Online]. Available: https://www.mysql.com/. [Accessed July 2017].

[28] "Google announces over 2 billion monthly active devices on Android," The Verge, [Online]. Available: https://www.theverge.com/2017/5/17/15654454/android-reaches-2-billion-monthly-active-users. [Accessed October 2017].

[29] "The Advantages Of Using Embedded Android In Medical Device Development," Hughes Systique Corporation, [Online]. Available: https://hsc.com/Blog/The-Advantages-Of-Using-Embedded-Android-In-Medical-Device-Development-1. [Accessed July 2017].

[30] I. Technologies, "Pedometer," Google Play, [Online]. Available: https://play.google.com/store/apps/details?id=com.tayu.tau.pedometer. [Accessed July 2017].

[31] Runtastic, "Runtastic," Google Play, [Online]. Available: https://play.google.com/store/apps/details?id=com.runtastic.android. [Accessed July 2017].

[32] G. LLC, "Google Fit," Google Play, [Online]. Available: https://play.google.com/store/apps/details?id=com.google.android.apps.fitness. [Accessed July 2017].

[33] R. T. L. Costa, "mHealth – Dispositivos inteligentes e computação mobile com aplicação na fisioterapia," Departamento de Ciências e Tecnologias da Informação, ISCTE-IUL, 2014.

[34] D. Maggiorini, C. E. Palazzi and D. Deponti, "DroidGlove: An Android-Based Application for Wrist Rehabilitation," *ICUMT '09 International Conference,* pp. 1-7, 2009.

[35] "Leap Motion Controller Ship Date Delayed Until July 22, Due To A Need For A Larger, Longer Beta Test," TechCrunch, [Online]. Available: https://techcrunch.com/2013/04/25/leap-motion-controller-ship-date-delayed-until-july-22-due-to-a-need-for-a-larger-longer-beta-test/. [Accessed September 2017].

[36] "AirSpace," Leap Motion, [Online]. Available: https://apps.leapmotion.com/. [Accessed July 2017].

[37] F. Weichert, D. Bachmann, B. Rudak and D. Fisseler, "Analysis of the Accuracy and Robustness of the Leap," *Sensors,* vol. 13, pp. 6380-6393, 2013.

[38] J. Guna, G. Jakus, M. Pogačnik, S. Tomažič and J. Sodnik, "An Analysis of the Precision and Reliability of the Leap Motion Sensor and Its Suitability for Static and Dynamic Tracking," *Sensors*, vol. 14, pp. 3702-3720, 2014.

[39] A. M. Brown, W. Stuerlinger and M. J. E. Filho, "The Performance of un-instrumented in-air pointing," *Proc. of the 2014 Graphics Interface Conf. Canadian Information Processing Society,* pp. 59-66, 2014.

[40] C. Falcao, A. C. Lemos and M. Soares, "Evaluation of Natural User Interface: A Usability Study Based on The Leap Motion Device," *Procedia Manufacturing,* vol. 3, pp. 5490-5495, 2015.

[41] A. H. Smeragliuolo, N. J. Hill, L. Disla and D. Putrino, "Validation of the Leap Motion Controller using markered motion capture technology," *Journal of Biomechanics,* vol. 49, pp. 1742-1750, 2016.

[42] "Game Engine Technology by Unreal," Unreal Engine, [Online]. Available: https://www.unrealengine.com/en-US/what-is-unreal-engine-4 . [Accessed October 2017].

[43] "Asset Store," Unity, [Online]. Available: https://unity3d.com/pt/asset-store. [Accessed October 2017].

[44] B. Sanders, Mastering Leap Motion, Birmingham: PACKT Publishing, 2014.

[45] L. Motion, "LeapMotionCoreAssers," GitHub, [Online]. Available: https://github.com/leapmotion/LeapMotionCoreAssets. [Accessed July 2017].

[46] "Roll-a-ball tutorial," Unity, [Online]. Available: https://unity3d.com/pt/learn/tutorials/projects/roll-ball-tutorial. [Accessed July 2017].

[47] "MySQL Workbench," MySQL, [Online]. Available: https://www.mysql.com/products/workbench/. [Accessed July 2017].

[48] "phpMyAdmin," [Online]. Available: https://www.phpmyadmin.net/. [Accessed July 2017].

[49] "Dashboards Android Developer," Google, [Online]. Available: https://developer.android.com/about/dashboards/index.html . [Accessed September 2017].

[50] JourneyApps, "Zxing-Android-Embedded," GitHub, [Online]. Available: https://github.com/journeyapps/zxing-android-embedded. [Accessed September 2017].

[51] P. Jahoda, "MPAndroidChart," GitHub, [Online]. Available: https://github.com/PhilJay/MPAndroidChart. [Accessed September 2017].

[52] H. Dodenhof, "CircleImageView," GitHub, [Online]. Available: https://github.com/hdodenhof/CircleImageView. [Accessed September 2017].

[53] Simbiose, "Encryption," GitHub, [Online]. Available: https://github.com/simbiose/Encryption. [Accessed September 2017].

[54] O. Postolache, F. Lourenço, J. M. D. Pereira and P. Girão, "Serious game for physical rehabilitation: Measuring the effectiveness of virtual and real training environments," in *IEEE International Instrumentation and Measurement Technology Conference (I2MTC),* Turin, Italy, 2017.

[55] U. o. Maryland, "QUIS: The Questionnaire for User Interaction Satisfaction," Human-Computer Interaction Lab, [Online]. Available: http://www.cs.umd.edu/hcil/quis/. [Accessed October 2017].

[56] H. P. repository, "Questionnaire for User Interface Satisfaction (QUIS)," HP Repository, [Online]. Available: https://ext.eurocontrol.int/ehp/?q=node/1611. [Accessed October 2017].

[57] J. P. Chin, V. A. Diehl and K. L. Norman, "Development of a Tool Measuring User Satisfaction of the Human-Computer Interface," *Proceeding of SIGHI Conference on Human Factors in Computing Systems, CHI'88,* pp. 213-218, 1988.

[58] I. C. Galinha, C. R. Pereira and F. Esteves, "Versão reduzida da escala portuguesa de afeto positivo e negativo – PANAS-VRP: Análise fatorial confirmatória e invariância temporal," *Revista PSICOLOGIA*, vol. 28(1), pp. 53-65, 2014.

# Appendix A – Articles

Article: **Serious Game for Physical Rehabilitation: Measuring the Effectiveness of Virtual and Real Training Environments**

This article has been accepted and presented in I2MTC 2017 May 22-25, 2017, Politecnico di Torino, Torino, Italy, 2017 IEEE International Instrumentation and Measurement Technology Conference.

I²MTC™ 2017 IEEE International Instrumentation and Measurement Technology Conference

CALL FOR PAPERS

May 22-25, 2017 | Politecnico di Torino | Torino, Italy

"Man is the measure of all things"
– *Protagoras*

POLITECNICO DI TORINO

I²MTC 2017 spans research, development and applications in the field of instrumentation and measurement science and technology. This includes Industrial Tracks, where research merges with practical applications in industrial technology used every day. The Conference fosters the exchange of know-how between industry and academia.

Paper contests will include a Conference Best Paper Award and Student Best Poster Awards. In addition to papers, the conference will also have Tutorials and Exhibits covering the entire range of instrumentation and measurement technology. The Conference focuses on all aspects of instrumentation and measurement science and technology-research, development and applications. The program topics include:

- Advances in Instrumentation and Measurement Developments and Techniques
- Data Acquisition Systems and Real-Time Measurements Internet of Things
- Energy and Power Systems
- Image Processing and Computational Intelligence Techniques
- Measurement and Instrumentation for Industrial Applications, Industry 4.0
- Measurement Applications and Software
- Measurement of Electric and Magnetic Quantities
- Measurement Systems and Theory
- Measurement, Instrumentation, and Methodologies for Medical, Biomedical, and Healthcare Systems Environment
- Non-invasive Measurement Techniques and Instrumentation
- Robotics, Control, Mechanical, and Material Measurements
- Sensors, Actuators, Transducers, and Sensor Fusion
- Signal Processing Techniques
- Techniques related to Instrumentation
- Virtual Measurement Systems

IEEE

IEEE INSTRUMENTATION & MEASUREMENT SOCIETY®

2017.imtc.ieee-ims.org

# Serious Game for Physical Rehabilitation: Measuring the Effectiveness of Virtual and Real Training Environments

[1.2]Octavian Postolache, [2]Filipe Lourenço
1. Instituto de Telecomunicações, Lisbon, Portugal
2. ISCTE-IUL, Lisbon, Portugal
Email: opostolache@lx.it.pt

[1.3.]J.M. Dias Pereira, [1.4]Pedro Silva Girão
3. ESTSetúbal/IPS - LabIM, Setúbal, Portugal
4. DEEC/Instituto Superior Técnico, Lisbon, Portugal

***Abstract*— Recent advances in low-cost natural user interfaces such as Microsoft Kinect and Leap Motion controller allow the Virtual Reality implementation of 3D serious games for, posture, upper limb and lower limb rehabilitation purposes. However, it is very important to compare the results obtained by the users that train in virtual and real environments. This paper presents a virtual reality serious game for upper limb rehabilitation using a natural user interface expressed by Leap Motion controller. One of the developed virtual reality serious game for rehabilitation is converted to a real scenario with the same elements and rules and the same aims of physical rehabilitation. In order to extract appropriate information from the serious game based on real objects a RFID technology was used together with software components developed in LabVIEW. The evaluation of hand muscles' activity during the training session is based on the usage of thermography that permits to measure in an unobtrusive way the distribution of the temperature on the hands' level. Based on analysis of thermographic images obtained before and after serious game practice, the level of activity of specific muscles associated with training for virtual and real scenario is extracted. Experimental results that are also included in the paper underline the effectiveness of the proposed method for the comparison of the training in two kinds of scenarios virtual and real.**

***Keywords*—**natural user interface, leap motion controller, serious game, RFID, thermography

## I. Introduction

The physical rehabilitation for the patients that suffered accidents or stroke events requires intensive intervention aimed at mainly improving motor abilities. Moreover, the periods associated with rehabilitation are usually long and the costs are expensive. One of the possible solutions to minimize these drawbacks is to perform remotely supervised self-training at home or to use system based supervised self-training at home. This last solution can contribute to reduce significantly the rehabilitation costs and also to shorten the time of rehabilitation processes. Self-training based on classical equipment's always requires the clinical professional supervision taking into account the usage of classical rehabilitation equipment that not deliver any information about the rehabilitation process. However in the last decade the healthcare community has been demonstrating a great interest for IT in physical rehabilitation that include the usage of serious games (Therapeutic games or ”Theragames”). The concept of Serious Games refers to the use of computer games without the main purpose of providing pure entertainment [1]. Healthcare rehabilitation is emerging as a leading target area for serious games, presenting new patient, health caregiver, and public expectations [2]. Several Theragames based projects have appeared which shows a wide general interest in improving and sustaining this technology. Among the Theragames, the Exergames [3] are a form of physical activity that requires the user to move at least a part of the body, the hands in our case, in order to interact and get the best experience the game, usually in virtual reality (VR) scenario.

The core idea of VR-based rehabilitation is to use sensing devices to capture and quantitatively assess the movements of patients under treatment to track their progress more accurately [4]. Two of the sensing devices that can be also considered as natural user interface (NUI) [5] are the Microsoft Kinect sensor and the Leap Motion Controller [6]. Referring Kinect there are several implementations that use Microsoft Kinect to control the games, providing feedback to patients, and even as a measuring tool [7][8]. According to the experience of Chang et al [4] for example, that integration of VR technology with exergames provides more motivation and engagement to patients while they are in rehabilitation activities. Important experience in the serious game for the rehabilitation is reported by our team [9][10][11]. The metrics values and the generated reports associated with implemented serious game are very useful for rehabilitation outcome evaluation. Nevertheless, the question of the usage of VR scenario instead the real scenario made the physiotherapist to question the VR efficiency in rehabilitation. In this context the present work provide a serious game based solution for physical rehabilitation, while the same serious game is implemented for VR and real scenario. A set of technologies including Leap Motion 3D camera, RFID are used to interact with two scenarios together the medical infrared thermography that permit to extract information in unobtrusive mode about the skin temperature on the hand level. The present approach is based on the usage of a thermography camera that provides non-intrusive procedure to measure the temperature without physical contact with the patient. Referring physical therapy, it can be mentioned that during training the skin temperature increases due to an increase in the blood flow. Thus, the measurement of the skin temperature, before and after the training sessions based on serious games, can give important information regarding the training session effectiveness.

The paper is organized as follows: section two presents the proposed serious game in two type of implementation including the main elements of the developed hardware and software components; section three is dedicated to the thermography as a method to measure the skin temperature on the hand level; section four is dedicated to the uncertainty sources; section five includes preliminary experimental results and the last section, section six, draws the conclusions.

## II. Serious Game for Physiotherapy Setup

The described serious game was developed for rehabilitation of the finger and hand motion and was designed and implemented for virtual reality (VR) scenario and for real scenario with real objects. In the case of VR scenario the interaction between the user under training and the game is performed using the Leap Motion natural user interface, while in the case of real scenario the user manipulates a set of colored cubes characterized by identification capabilities based on RFID technology.

### A. *Leap Motion Controller*

The Leap Motion Controller (see Fig. 1) [12] can be connected to a computer using a USB cable. The heart of the device consists of two cameras and three infrared LEDs. This device can track the motion of both hands and all 10 fingers with up to 1/100 mm accuracy and no visible latency within its field of vision. The Leap Motion Controller's viewing range is 60 cm above the device and can reach 80 cm with the Orion beta software [13]

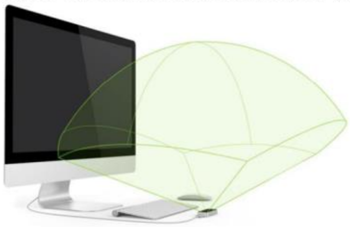


Fig. 1. Leap motion controller detection zone

This technology holds great promise for the rehabilitation field, since it does not require the patient to wear gloves (example, force glove feedback), which makes an important factor in patients who had gone through hand surgery. The device provides a new way to interact with user and the computer, with the ability to bring the real world to the digital domain.

### B. *Serious Game Setup: Virtual Reality Scenario*

The game "Collect Color Cube" was developed by the authors using Unity and C# scripts the interface between the user and VR scenario is expressed by the Leap Motion controller connected by a computer that runs the VR serious game. The Leap Motion captures the movements of the user's hands and fingers, and represents them as part of the virtual game scenario for physical rehabilitation (Fig. 2). In the implemented serious game the goal is to place the colored cubes in the collecting box that presents the same color as the captured cubes in order to earn points. If the user associated the cubes with different color collecting box the player loses points. It has a certain established time and the boxes will change the color during the game. A common setup was 4s to 10s the period between color changes. At the same time the physiotherapist can impose the serious game duration according to the capabilities of the user under training.

The game has to be adaptable to the player needs to take better advantage of the gaming experience, to do so at the beginning of each game are set some parameters such as:

- Name;
- Age;
- Gender;
- What hands to use during the game (left, right, both);
- What is the rest time between two games?
- What is the duration of the game?

Depending on the parameters set, the game will adapt to the player, for example, along with Gender and Age will define how the hand will be represented during the game. A younger player will have a more robot aspect then an older player group. This group would have a closer representation of reality, and this representation is different in male and female genders.

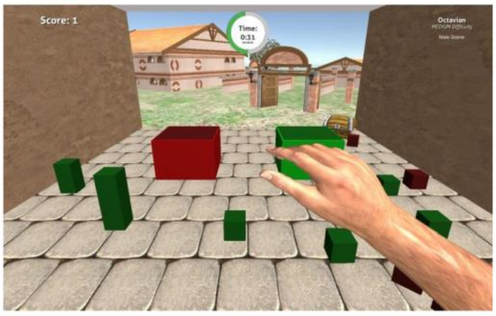


Fig. 2. "Collect Color Cube" Serious Game: Virtual Reality Scenario

### C. *Serious Game Setup in Real Scenario*

The "Collect Color Cube" virtual reality serious game was translated into a real scenario where the virtual objects (cubes and boxes) are expressed by red and green cubes with RFID tags (125Hz disk tags) used for identification of individual cubes that can be red or green. The implemented technology may be used to extend the diversity of the object color in a next level of the implemented serious game. The cubes are disposed on a table and the user will catch and introduce in the red and green box the box color being signaled by red and green LEDs disposed on the box level (Fig. 3).

The color associated with boxes alternatively changes in order to impose the same challenges that characterize the virtual reality serious game. Thus in serious game real scenario from time to time (e.g. 5s time interval) the LEDs array are receiving the switch-on or switch-off commands from the LED driver that is a part of RFID LF reader unit (1024 Phidget Read-Write) characterized by 125kHz embedded antenna (Fig. 4).

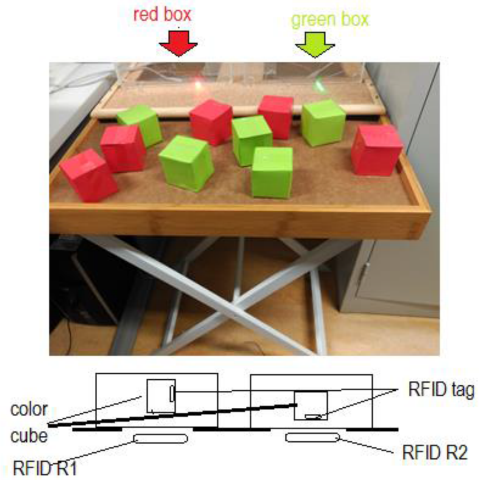


Fig. 3. Serious game setup for real scenario including 2 RFID readers associated with two boxes and RFID tags for each color cube

The PhidgetRFID supports the following protocols; EM4100, ISO11785 FDX-B, and PhidgetTag. The PhidgetTag protocol simply stores up to 24 ASCII characters to the tag, eliminating the necessity for a table of corresponding tag numbers and names.

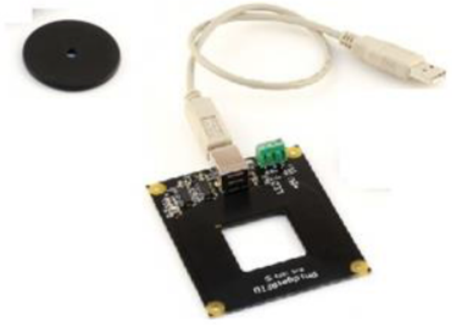


Fig. 4. LF RFID reader embedded in the box and LF tags embedded in the red and green color cubs for cubs 'identification.

Taking into account that passive tags require a strong RF field to operate, their effective range is limited to a space volume in close proximity to the RFID reader (less than 7.5 cm). The distance over which the RFID tag is usable is affected by the orientation of the reader and tag in respect to each other. For this reason a freely motion, based on gravity, in the color cube can be assure less failure detection events of the cubes that are introduced by the user. Taking into account that the serious game implementation in real scenario implies limitations related with the number of objects and the game space, in the preliminary implementation of the game a set of 10 cubes (5 red and 5 green) were considered. Table 1 the RFID tags that were used for cubes identification. Table 1- Serious Game' Cubes identification

| No/ | RED Cube Tag | GREEN Cube Tag |
|---|---|---|
| 1 | 0107ee57be | 0107ee579b |
| 2 | 01068dbf16 | 0107ee66b2 |
| 3 | 0107ee5928 | 0107ee8bce |
| 4 | 0107ee5691 | 0107ee56cf |
| 5 | 01023c5297 | 0107ee8ca2 |

The control of the LEDs through the RFID LED driver and the RFID readers' control (RFID R1, RFID R2 – Fig. 3) is carried out by a software developed in LabVIEW using RFID Phidget driver. The software GUI is presented in Fig.5 and is expressed by two TABs one for RFID Readers and LEDs control and one is associated with data storage and visualization.

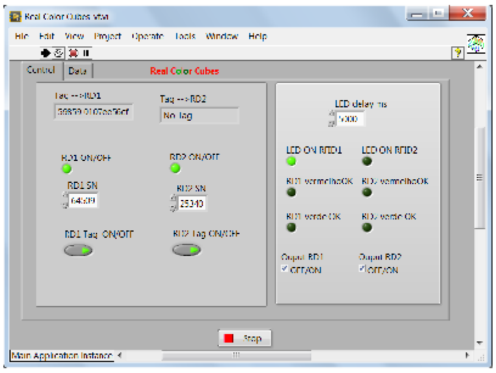


Fig. 5. Read Color Cubes – GUI including RFID reader controls and LEDs control associated with the implemented serious game for real scenario.

Similarly to the "Collect Color Cube" for virtual scenario, in the real scenario case a time delay between the "box color change" events was implemented, being this time interval defined by the physiotherapist in the serious game configuration phase. The values of "LED delay" control that appears on the GUI level were considered between 4s and 10s. Considering the LEDs color associated with the box and the cube color detected through the embedded RFID tag, the correct and incorrect actions of the user are signaled and stored in a file of training. The physiotherapist may perform an objective evaluation of the rehabilitation process creating new scenarios for the rehabilitation sessions by modifying cubes size, color, texture, weight, and can also perform modification for cognitive rehabilitation sessions [14].

The physical activity developed during serious game based physical rehabilitation can be evaluated in unobtrusive way using thermography.

## III. Thermography

Thermography is a non-radiating and contact-free technology to monitor physiological functions related to skin temperature control that made this technology efficient in the case of muscular activity assessment during the rehabilitation period. One of the currently application of medical infrared thermography is for athletes that are exposed to physical stress during training that can originate "minor traumas". Thus, the usage of medical infrared thermography can avoid injuries [15] and also make an evaluation of the rehabilitation program outcome through the evolution in time of the injuries' region temperature [16].

The novelty of the present approach is related to the usage of thermography for unobtrusive evaluation of muscle activity during serious game. Thus a comparison between the user results obtained for a serious game performed in virtual scenario, or when the user perform the same serious game in real scenario. At the same time the evaluation of muscular activity, that originate variation of the temperature on the skin level, during successive sessions characterized by different duration is considered.

To evaluate the effectiveness of the training for specific *Theragames* a thermographic camera FLIR E60 is used to capture the skin thermographic images of the hand and finger regions that can be associated with the localization of the muscles that perform intensive activity during the considered serious game as the muscles that are presented in Fig. 6. Referring to the thermographic camera specifications it can provide 320x240 thermographic images for a frame rate of 60Hz. The thermographic sensitivity is better than 0.05ºC and the accuracy is 2% of reading for an extended measurement range between -20°C and 650°C.

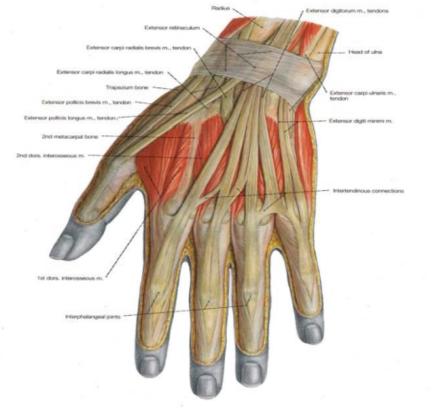

Radius
Extensor digitorum m., tendons
Extensor retinaculum
Extensor carpi radialis brevis m., tendon
Head of ulna
Extensor carpi radialis longus m., tendon
Trapezium bone
Extensor carpi ulnaris m., tendon
Extensor pollicis brevis m., tendon
Extensor pollicis longus m., tendon
Extensor digiti minimi m.
2nd metacarpal bone
2nd dors. interosseous m.
Intertendinous connections
1st dors. interosseous m.
Interphalangeal joints

Fig. 6. Hand muscle that are used during the "Collect Color Cubes" serious game in VR and real scenario

Using the acquired thermographic images that provide information about skin temperature distribution caused by the muscle activity, o the temperature before and after training was extracted. The temperature measurement is based on the region selection of the thermographic image (Fig. 7). The temperatures for the selected region (minimum, maximum and average temperature) are obtained using the FLIR Tools+ thermography image analysis toolkit [17].

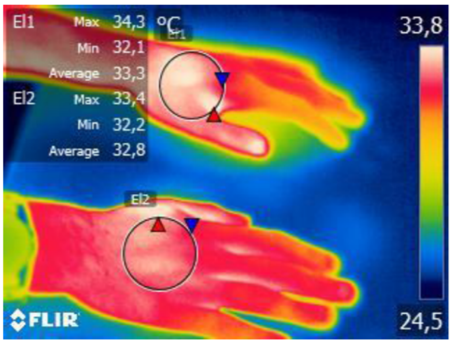


Fig. 7. Thermographic images associated with temperature distribution after 1 min game training using "Colored Cubes" serious game for virtual scenario.

## IV. Uncertainty Sources

The usage, in the present work, of different technologies such as infrared emitters and detectors associated with Leap Motion Controller, RFID for real game objects identification and the thermography for unobtrusive measurement of hand skin temperature requires a discussion about the involved sources of uncertainties and about the possibilities to evaluate the uncertainties values.

The usage of RFID in real serious game cube identification implies the introducing in the system of uncertainties that are proper of this type of technology. One of the uncertainty sources is represented by tag-reader location. In the present case, the position of the LF RFID was established in order to assure optimal reading of the tags however the usage of a single RFID tag on the level of the cube made relative position tag – RFID reader antenna difficult to predict during the game. One of the solutions, and the cheaper one, is to use cubes with geometrical characteristics that made maximum distance between tag and reader antenna small enough for successful reading.

Referring to the thermographic camera that is essential for physical training effectiveness evaluation the sources of uncertainties according with FLIR thermographic camera manufacturer [18] are: emissivity, reflected ambient temperature, transmittance, atmosphere temperature, camera response and calibrator (blackbody) temperature accuracy. Taking into account the laboratory requirements to extract the quantitative values of all of the above mentioned sources of uncertainties in the present work was not considered this kind of approach, however the FLIR E60 calibration procedures was followed for the used thermographic camera in order to guarantee the specified accuracy values. At the same time for objective evaluation of real and virtual game effectiveness it is important to figure out the differences between the measured temperature for VR gaming scenario and for Real gaming scenario, and not the absolute values of temperatures for each of the considered cases.

## V. Results and Discussions

Several laboratory testing sessions were carried out with a number of four volunteers: three males and one female to evaluate the performance of the serious games for virtual reality and real scenarios. The healthy volunteers were participated in two game training sessions. The first session was focused on the serious game characterized by the virtual reality scenario. The experimental procedure is essentially based on the temperature measurement of the hands' skin before and after the game session. The volunteers played sessions of 30s, 1min, 1.5min and 2 min and a set of thermographic images were acquired and off-line analyzed using the FLIR Tools+ and a circular temperature estimation region as it is presented in Fig. 7. At the same time in order to extract the correlation between game score, and effort expressed by intensive muscle activity the scores for each game session was stored and later analyzed. The evolution of the temperature on the hand skin level, before and after two minutes training based on serious game for virtual reality scenario, is presented in Fig. 8.

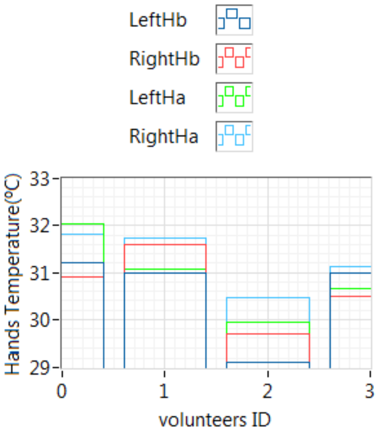


Fig. 8. Average temperature evolution of hand skin temperature before and after 2 min training

In Fig. 8 it can be observed that in some of the cases that the game practicing originates temperature variations on the level of hands' skin. However the number of tests and the fact that part of the user are using only one of their hand to catch red or green cubes and put in collecting box requires new additional tests that are running now with an extended number of participants. The relation between learning and game scores are represented in Fig. 9. It can be observed that some particular low performance was occurred for 1min training, while for 1.5min and 2min the players increase their performances according with their knowledge about this new game.

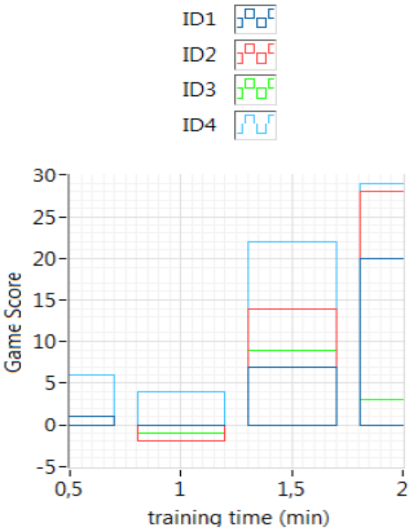


Fig. 9. Score Evolution for different training durations (IDi volunteer ID)

The comparison between the obtained scores by the participants when they are playing in virtual reality scenario or real scenario was carried out. The volunteers obtained better scores at the beginning on serious game in real scenario while latter the obtained scores are near each other (ID3 volunteer: 5 for 1min serious game for VR scenario, 3 for 1 min serious game for real scenario).

An important part of the serious game effectiveness evaluation was based on the comparison of the temperature variation on the hands' skin when the user performs the serious game for real and for virtual scenario. The tests represent an ongoing work involving an extended number of volunteers. The results obtained in one of the cased is presented in Fig. 10, where it can be observed that for real scenario the values of average temperature after 1 min playing time is in general higher than obtained for VR scenario. The reduced number of tests does not permit to decide about the type of serious game that is more appropriate for rehabilitation regarding the intensity of muscle activity.

## VI. Conclusions

A serious game "Collect Color Cube" was designed and implemented for virtual reality scenario including virtual object and user hand avatar and for real scenario where the user interact directly with real objects. Important work was done regarding the serious game interaction based on Leap Motion considering the game personalization according to the patient needs.

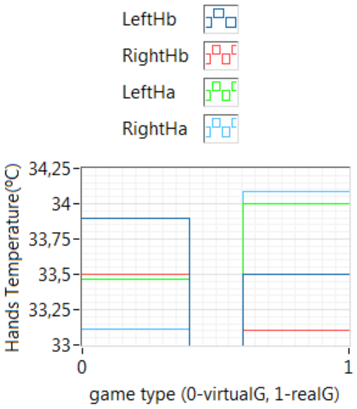


Fig. 10. Hand' Average temperature evolution for virtual training game and real training game (LeftHb, LeftHa- temperatures of left hand before and after training, RightHb, RightHa – temperatures of right hand after) for one volunteer

The serious game for real scenario was materialized using an innovative solution that permit the object identification and automatic recording of the number of cubes captured and collected on the boxes. Based on RFID technology and appropriate software, the same functionalities as for VR game were implemented. Thermography usage for unobtrusive evaluation of the muscle activity during serious game session proves to be one of the promissory solution taking into account that no contact with the user body is required. However, additional thermographic camera calibration concerns must be addressed taking into account the importance of temperature measurement accuracy before and after the game sessions. Future work involves additional tests with an extended number of users that are under rehabilitation processes. An extended approach concerning the sources of uncertainty that affect the system accuracy will be considered for the future developments..

## Acknowledgement

The work was supported by Fundação para a Ciência e Tecnologia project: PTDC/DTP-DES/6776/2014 and Instituto de Telecomunicacões and IT-IUL.

## References

[1] Rego, P., Moreira, P.M., Reis, L.P., "Serious games for rehabilitation: A survey and a classification towards a taxonomy," Proc. 5th Iberian Conference on Information Systems and Technologies (CISTI), IEEE Press, New York, 1-6, 2010.

[2] Watters, C., Oore, S., Shepherd, M., Abouzied, A., Cox, A., Kellar, M., Kharrazi, H., Liu F., Otley, A.," Extending the Use of Games in Health Care", Proc. of HICSS, 2006.

[3] Sinclair, J., Hingston, P., & Masek, M., "Considerations for the design of exergames," Proc. of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia, 2007.

[4] Chang, C.-Y., Lange, B., Zhang, M., Koenig, S., Requejo, P., Somboon, N., Sawchuk, AA, Rizzo, AA, "Towards pervasive physical rehabilitation using Microsoft Kinect, Pervasive Computing Technologies for Healthcare (PervasiveHealth)," 2012 6th International Conference on, 21-24 May, 159-162, 2012.

[5] Madeira, R.N., Costa, L., Postolache, O., "PhysioMate - Pervasive physical rehabilitation based on NUI and gamification," Proc. 2014 International Conference and Exposition on Electrical and Power Engineering (EPE), 16-18 Oct., 612-616, 2014.

[6] Mapari Rajesh B, Kharat G., "Real time human pose recognition using leap motion sensor," IEEE International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), 2015.

[7] Clark, R.A., Pua, Y.-H., Bryant, A.L. and Hunt, M.A., "Validity of the Microsoft Kinect for providing lateral trunk lean feedback during gait retraining," Gait Posture, 38, 4, 1064–1066, 2013.

[8] Chang, Y.J., Chen, S.-F., Huang, J.-D., "A Kinect-based system for physical rehabilitation: A pilot study for young adults with motor disabilities," Research in Developmental Disabilities, 32 (6), 2566–2570, 2011.

[9] Postolache O., Girão P.M., Dias Pereira J. M., Postolache G., "Postural Balance Analysis using Force Platform for K-Theragame users," IEEE International Symp. on Medical Measurements and Applications - MeMeA, Benevento, Italy, Vol. 1, pp. 1 - 6, May, 2016.

[10] O. Postolache, "Physical Rehabilitation Assessment based on Smart Training Equipment and Mobile APPs," IEEE EMBS Romanian Chapter E-Health and Bioengineering Conference EHB, Iasi, Romania, Vol. 1, pp. 1 - 6, November, 2015.

[11] O. Postolache, F.C. Cary, N.D Duarte, P.M. Girão, "Physiotherapy Assessment based on Kinect and Mobile APPs," International Conf. on Information, Intelligence Systems and Applications - IISA, Corfu, Greece, Vol. 1, pp. 1 - 6, July, 2015.

[12] Colgan A, "How Does the Leap Motion Controller Work?" on-line at: http://blog.leapmotion.com/hardware-to-software-how-does-the-leap-motion-controller-work/, 2014.

[13] Leap Motion Developer, "Orion beta", on line at: https://developer.leapmotion.com/orion, 2016.

[14] Rocha R., Luís Paulo Reis, Paula Alexandra Rego, "Serious games for cognitive rehabilitation: Forms of interaction and social dimension," 10th Iberian Conference on Information Systems and Technologies (CISTI), 2015.

[15] B.B. Lahiri, S. Bagavathiappan, T. Jayakumar, John Philip, "Medical applications of infrared thermography: A review," Infrared Physics & Technology, Volume 55, Issue 4, July 2012, pp. 221–235.

[16] C. Hildebrandt, C. Raschner, K. Ammer, "An Overview of Recent Application of Medical Infrared Thermography in Sports Medicine in Austria," Sensors (Basel). Vol. 10, issue. 5, pp. 4700–4715, 2010.

[17] Flir, Flir Tools/ Tools+ User manual", https://www.ivytools.com/v/flir-manuals/software/flir-tools-manual.pdf, 2016.

[18] Flir, "Infrared Camera Accuracy and Uncertainty in Plain Language", on-line at: http://www.flir.com/science/blog/details/?ID=74935, 2017.

Article: **Wrist and Hand Rehabilitation Software Platform Based on Leap Motion Controller**

This article has been accepted and presented in SEIA 2017 3rd edition September 20-22, 2017, Moscow, Russia, 2017 IFSA International Conference on Sensors and Electronic Instrumentation Advances SEIA.

3rd International Conference on Sensors and Electronic Instrumentation Advances (SEIA' 2017)

20-22 September 2017, Moscow, Russia

Welcome to Moscow !

*3rd International Conference on Sensors and Electronic Instrumentation Advances (SEIA' 2017),*
*20-22 September 2017, Moscow, Russia*

# Wrist and Hand Rehabilitation Software Platform Based on Leap Motion Controller

**G. Postolache [1], F. Lourenço [2], A. Antunes [1], and O. Postolache [1,2]**
[1] Instituto de Telecomunições, Lisbon, Portugal
[2] Instituto Universitário de Lisboa, ISCTE-IUL, Lisbon, Portugal
E-mail: opostolache@lx.it.pt

**Summary:** A software platform for upper limbs movements' assessment based on Leap Motion Controller (LMC) was developed. It allows measurements of clinically proved effective hand and finger exercises. The amplitude and time interval for each different movement, frequency of different movements, asymmetry of bilateral movements, standard deviation of signal amplitude, Poincaré plots are represented in the software platform. A serious game *Collect Color Cube*, was developed using Unity, C# scrips, and signals from LMC related to movements of the user's hands and fingers.

**Keywords:** Leap Motion Controller, wrist and hand rehabilitation, serious games.

## 1. Introduction

The core of expertise, practice, education and research in physiotherapy is the assessment, prevention and treatment of movement disorders as well as functional autonomy promotion. Quantifying the overall effects of the physiotherapy is important both for health professionals (i.e., for optimization of therapy interventions and objective monitoring of recovery), as well as for patients, which by receiving digital information on many aspects of movements (e.g., hand velocity or joint angles) are physically and cognitively challenged, in an environment that promote therapeutic engagement.

Various wearable and unobtrusive devices have been developed for assessment of neuromusculoskeletal and movement related functions (i.e., for joint mobility, muscle tone or strength) and/or activity limitation (i.e., difficulties that an individual may have in executing activities). Devices based on inertial sensors (e.g., accelerometer, gyroscope, inertial measurements unit) [1], sensors for electromiography [2] or mecanomiography [3], or unobtrusive sensors based on microwave radar [4] and infrared technology [5] were described in the last decades, as useful instruments for diagnosis as well as for in-clinic or remote monitoring of progress in treatment of neuromusculoskeletal impairments.

The Leap Motion Controller (LMC), was publicly presented for the first time in 2013. It is a new type of device based on infrared technology that by his small dimension, low price and millimetre accuracy promise development of myriad applications for motor rehabilitation environment. Study on LMC efficacy in characterization of essential tremor was recently presented [6]. Use of LMC in conjunction with virtual reality (VR) has been shown being a useful tool for rehabilitation of children with cerebral palsy [7] and for stroke patients [8]. Also, several serious game for upper limb training (VirtualRehab) were developed by VirtualWare [9]. The goal of our work was to developed a platform for arm and fingers exercise monitoring, including an extended number of exercises that may be monitored during training in motor rehabilitation sessions.

## 2. Methods

### 2.1. Leap Motion Controller

Leap Motion Controller is a small, rectangular device (13mmx13mmx76mm) that weights 45g. The LMC consists of three IR (Infrared Light) emitters and two IR cameras [10]. It streams data at a variable acquisition rate, that varies between mean value of less than 40 Hz [11] to up to 120 Hz [12], under both static and dynamic conditions [11]. LMC works with both Windows and Macintosh operating system and connects to a computer via a USB 3.0 connection. It has a full-functioning Software Developer Kit (SDK). The controller itself is accesses and programmed through Application Programming Interfaces (APIs), with support for a variety of programming languages, ranging from C++ to Python [13].

Positive value of the vertical y-axis increase upwards. The effective range of the controller extends from approximately 25 to 600 millimeters above the device (Fig. 1).

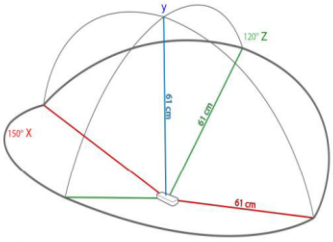


**Fig. 1**. Leap Motion Controller interaction area.

The accuracy and robustness of the LMC were evaluated [10-12,14]. During static setup a deviation between a desired 3D position and the average measured positions below 0.2mm was obtained and of 1.2mm for dynamic setups. The repeatability had an average of less than 0.17mm. Standard deviation was below 0.7mm per axis when moving to discrete positions on a path [10].

### 2.2. Metrics

When the user's hand is detected by the LMC sensor, the discrete position and orientation of the palm and fingers are recorded and processsed. The developed software allows representation of amplitude and time interval for each different movement, frequency of different movements, asymmetry of bilateral movements, standard deviation of signal amplitude, Poincaré plots.

Asymmetry of movement was estimated using ratio index (RI) and symmetry index (SI). Thus, assuming that $X_R < X_L$, where $X_R$ and $X_L$ are the values of the upper limb movements' duration for the right and left limbs, the factors were calculated as follows:

$$RI = \left(1 - \frac{X_R}{X_L}\right) 100[\%] \qquad (1)$$

The factor indicates which of the variables has the highest value, and as such creates asymmetries. The value of RI = 0 indicates full symmetry, while RI ≥ 100% indicates asymmetry.

$$SI = \frac{|X_L - X_R|}{0.5 \cdot (X_L + X_R)} \cdot 100[\%] \qquad (2)$$

The value of SI = 0 indicates full symmetry, while SI ≥ 100% indicates its asymmetry.

Metrics were implemented for the following type of movements: Thumb Extension/Flexion; Thumb Touch by each four Fingertips; Finger Stretch; Fingers Abduction; Finger Lift; Hand/Finger Tendon Glide; Claw Stretch; Make a Fist/Release and Spread the Finger; Wrist Flexion/Extension; Wrist Supination/Pronation; Wrist Ulnar/Radial Deviation; Diadochokinesis; Trace Geometrical Forms (e.g., circle, spiral, zigzag, triangle, square, or rectangle).

During the experiments, video recording of hands movements was realized with an action camera (Ricoh Action Cam WG-M2). Video images were analysed in order to evaluate the LMC precision detection at different movements' frequencies. Tests were realized with 6 healthy participants, mean age and range 36 (24-49). Instructions of participants were realized for the movements be realized into LMC area of interaction, that research studies have shown better accuracy for movements detection [10-12].

### 2.3. Serious Game

The serious game *Collect Color Cube* was developed using Unity and C# scrips. The LMC captures the movements of the user's hands and fingers, and represents them as part of the virtual game scenario for upper limbs motor rehabilitation. The goal of the game is to use claw stretch and fingers abduction exercises to place the coloured cubes in the collecting box that presents the same colour as the captured cubes. Points were obtained taking into account the number of cubes correctly placed in the boxes. The game allows configuration of time for exercises and type of hands that can be used (i.e., left or right hand, robotic hands, man or woman hand). The boxes can change the colour during the game. Also, the user can choose to collect fruits, vegetables, or pets. The therapist may change the setting of the game creating new scenarios for the rehabilitation sessions by changing the size of objects that should be collected, their colour and texture as well as the color of walls that limits the virtual space of game.

## 3. Discussion and Conclusions

A software platform for upper limbs movements assessment based on LMC was developed. It allows measurements of clinically proved effective arm, hand and fingers exercises. In Fig. 2 a spiral form traced by movement captured by LMC are represented. Can be observed the difficulties that even the healthy subjects have on realizing the task. The standard deviation of movements amplitude can be used as indicator of severity of impairments and progress in therapy.

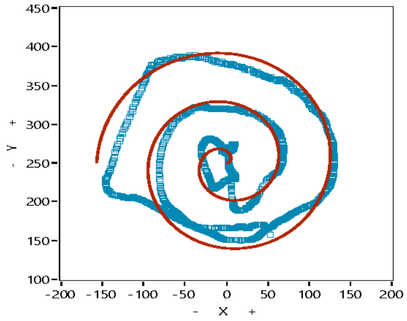


**Fig. 2.** Traced spiral using Leap Motion Controller detection of movements.

Examples of signals acquired during wrist supination/pronation and wrist flexion/extension are represented in Fig. 3 and Fig. 4. The user obtained feedback on the movement by observing the virtual hand movement (in this case a robotic hand representation) and LMC signal. In Fig. 3 a slow movement of hand supination and pronation was realized, during 1 minute, by following the superior part of trace of a circle. The geometric forms were chosen for exercise realization, in order to provide meaningful feedback for user on deviation from optimal movements as well as for graphical comparison of movements realized in different sessions or by different patients. By observing different characteristics of LMC signal important information on progress of rehabilitation can be obtained. In Fig. 3 the image bellow represent the X projection of palm signal where assessment of time for each movement and frequency of movements can be easily carried out. The initial hand supination movement was realized more slowly than the following pronation movement. In Fig. 4 wrist flexion and extention was realized by asking the user to follow the zig zag trace. The LMC Y projection of palm signal, relevant for analyzis of the wrist fexion and extension movements, are represented in the bellow image of Fig. 4, where information on frequency of the movements and time of each movement can be analysed.

In Fig. 5, an image of the developed serious game that used LMC for movements detection is represented. The image may be an important visual feedback for the patient with movement impairments (e.g., he/she can observe that the impaired limb have more difficulties on realizing the task – more cubs are presented on affected side).

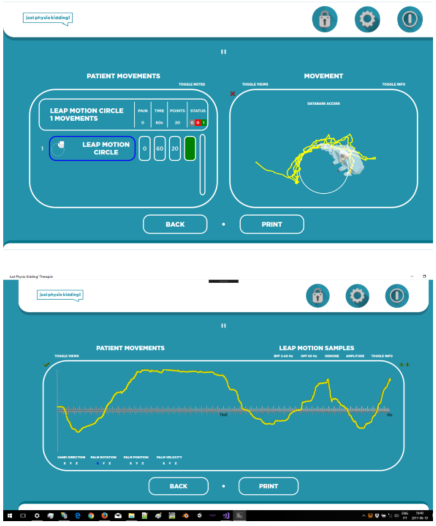


**Fig. 3.** Representation of LMC signal during palm supination and pronation.

LMC did not have the features of a mouse for tracing geometrical forms. As previously shown [14-15] users took a longer time to perform simple tasks, and indicated muscle fatigue and tired arms. Fortunately, these are the most important requirements for software development for upper limbs movements' rehabilitation. The patients cannot realize the movements with high or normal velocity and need to strengthen the muscles involved in those movements' training.

The developed software platform by providing objective data on type, intensity and frequency of movements during motor rehabilitation interventions, may help therapists and patients in selection of appropriate movements or activities that offer meaningful and motivating exercise environment.

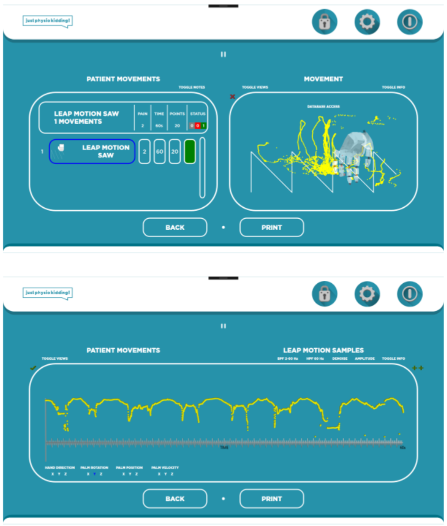


**Fig. 4.** Representation of LMC signal during wrist flexion and extension.

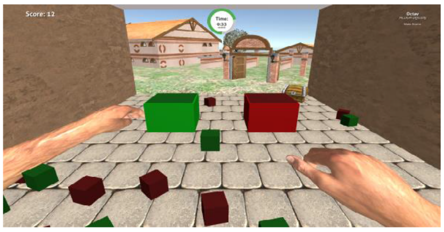


**Fig. 5.** Image from *Collect Color Cube* game.

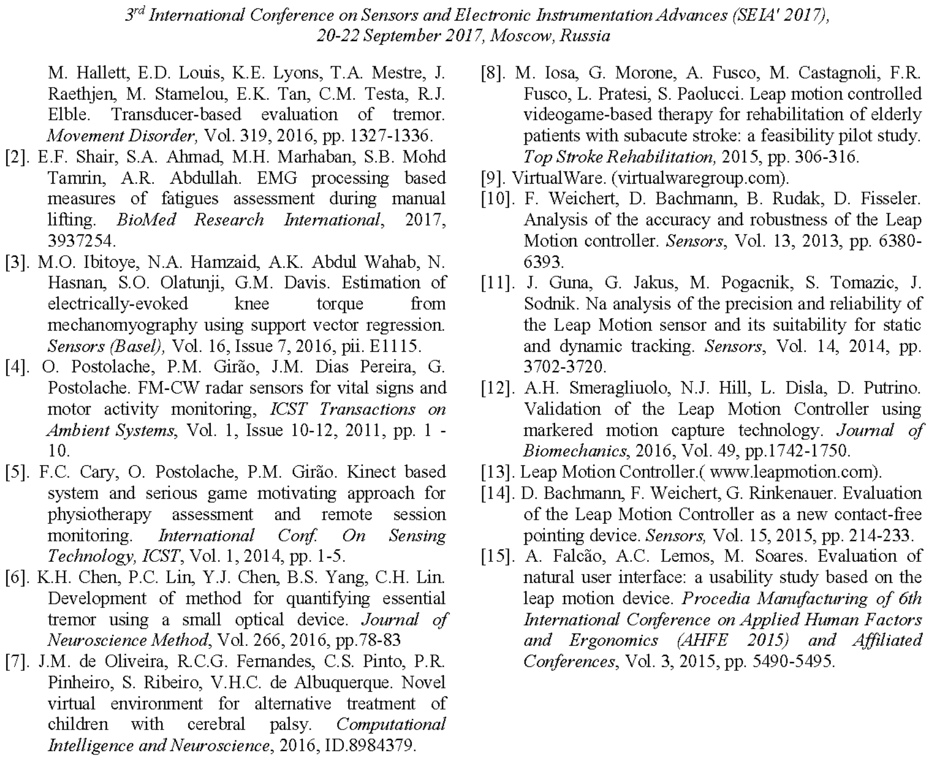
Future research should be realized on algorithms and low cost sensors, that may be used in conjunction with LMC to compensate the errors caused by uncertainty in sampling frequency and also on strategies that may reduce the impact on perceived constrains and discomfort related with narrow LMC interaction space.

## Acknowledgements

The work was supported by Fundação para a Ciência e a Tecnologia, project PTDC/DTT-DES/6776/2014 and Instituto de Telecomunicações.

## References

[1]. A. Haubenberger, G. Abbruzzese, P.G. Bain, N. Bajaj, J. Benito-Leon, K.P. Bhatia, G. Deuschl, M.J. Forjaz,



*3rd International Conference on Sensors and Electronic Instrumentation Advances (SEIA' 2017), 20-22 September 2017, Moscow, Russia*

M. Hallett, E.D. Louis, K.E. Lyons, T.A. Mestre, J. Raethjen, M. Stamelou, E.K. Tan, C.M. Testa, R.J. Elble. Transducer-based evaluation of tremor. *Movement Disorder,* Vol. 319, 2016, pp. 1327-1336.

[2]. E.F. Shair, S.A. Ahmad, M.H. Marhaban, S.B. Mohd Tamrin, A.R. Abdullah. EMG processing based measures of fatigues assessment during manual lifting. *BioMed Research International*, 2017, 3937254.

[3]. M.O. Ibitoye, N.A. Hamzaid, A.K. Abdul Wahab, N. Hasnan, S.O. Olatunji, G.M. Davis. Estimation of electrically-evoked knee torque from mechanomyography using support vector regression. *Sensors (Basel),* Vol. 16, Issue 7, 2016, pii. E1115.

[4]. O. Postolache, P.M. Girão, J.M. Dias Pereira, G. Postolache. FM-CW radar sensors for vital signs and motor activity monitoring, *ICST Transactions on Ambient Systems*, Vol. 1, Issue 10-12, 2011, pp. 1 - 10.

[5]. F.C. Cary, O. Postolache, P.M. Girão. Kinect based system and serious game motivating approach for physiotherapy assessment and remote session monitoring. *International Conf. On Sensing Technology, ICST*, Vol. 1, 2014, pp. 1-5.

[6]. K.H. Chen, P.C. Lin, Y.J. Chen, B.S. Yang, C.H. Lin. Development of method for quantifying essential tremor using a small optical device. *Journal of Neuroscience Method*, Vol. 266, 2016, pp.78-83

[7]. J.M. de Oliveira, R.C.G. Fernandes, C.S. Pinto, P.R. Pinheiro, S. Ribeiro, V.H.C. de Albuquerque. Novel virtual environment for alternative treatment of children with cerebral palsy. *Computational Intelligence and Neuroscience*, 2016, ID.8984379.

[8]. M. Iosa, G. Morone, A. Fusco, M. Castagnoli, F.R. Fusco, L. Pratesi, S. Paolucci. Leap motion controlled videogame-based therapy for rehabilitation of elderly patients with subacute stroke: a feasibility pilot study. *Top Stroke Rehabilitation,* 2015, pp. 306-316.

[9]. VirtualWare. (virtualwaregroup.com).

[10]. F. Weichert, D. Bachmann, B. Rudak, D. Fisseler. Analysis of the accuracy and robustness of the Leap Motion controller. *Sensors*, Vol. 13, 2013, pp. 6380-6393.

[11]. J. Guna, G. Jakus, M. Pogacnik, S. Tomazic, J. Sodnik. Na analysis of the precision and reliability of the Leap Motion sensor and its suitability for static and dynamic tracking. *Sensors*, Vol. 14, 2014, pp. 3702-3720.

[12]. A.H. Smeragliuolo, N.J. Hill, L. Disla, D. Putrino. Validation of the Leap Motion Controller using markered motion capture technology. *Journal of Biomechanics*, 2016, Vol. 49, pp.1742-1750.

[13]. Leap Motion Controller.( www.leapmotion.com).

[14]. D. Bachmann, F. Weichert, G. Rinkenauer. Evaluation of the Leap Motion Controller as a new contact-free pointing device. *Sensors,* Vol. 15, 2015, pp. 214-233.

[15]. A. Falcão, A.C. Lemos, M. Soares. Evaluation of natural user interface: a usability study based on the leap motion device. *Procedia Manufacturing of 6th International Conference on Applied Human Factors and Ergonomics (AHFE 2015) and Affiliated Conferences*, Vol. 3, 2015, pp. 5490-5495.

## Appendix B – User Manual

ISCTE IUL
Instituto Universitário de Lisboa

Departamento de Ciências e Tecnologias de Informação

# User Manual

LeaPhysio – Games Enhanced Physical Rehabilitation

Filipe da Costa Pereira Lourenço

Advisor:
Doctor Octavian Postolache, Assistant Professor
ISCTE-IUL

**October 2017**

## Figures

## Contents

## User Manual

This manual aims to present the features available in the LeaPhysio project applications and explain how they work. Chapter 1 is intended to demonstrate the functionalities of the LeaPhysio application for the computer, which is intended for patients in the physiotherapy clinic. Chapter 2 demonstrates the functionalities of the mobile application, which serves as a physiotherapist for the management of patients. This application is also intended for the manager of the clinic, making it possible to manage their physiotherapists. Chapter 3 is intended to explain the method of installation of the abovementioned applications.

# Chapter 1 – LeaPhysio Computer Application

## 1. Overview and Application Description

The LeaPhysio Computer Application is intended for patients in physiotherapy clinics, but this manual isn't intended for this type of actor, it should be the responsibility of the physiotherapist to be aware of this manual, to teach and explain to the patient what to do.
At first, the user encounters two splash screens (see Figure 1.1).

ISCTE IUL
Instituto Universitário de Lisboa
LeaPhysio
Games Enhanced Physical Rehabilitation

Figure 1.1 – Splash screens

After both are finished, a new window is displayed (see Figure 1.2). This window is required for the patient to log in so that they can perform their respective training.

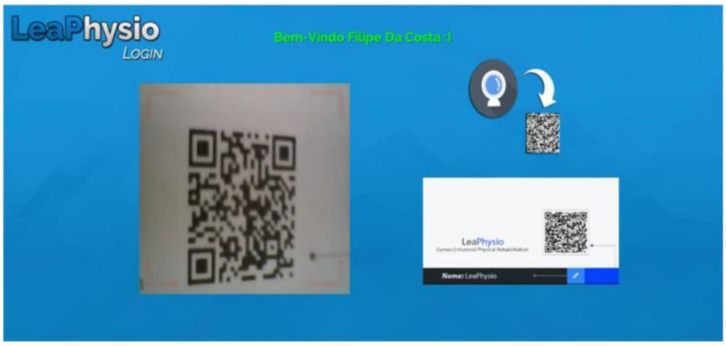
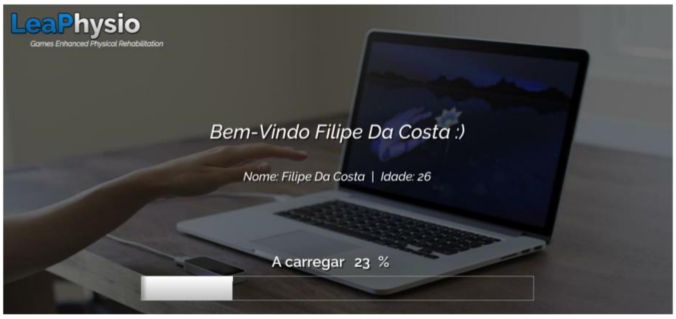


Figure 1.2 – Login QRCode window

The patient (with the help of the physiotherapist) shows in the Web Cam the QRCode in LeaPhysio card. The application reads the information contained in the QRCode and identifies whether the user is valid. If not, a negative sound is released, and a message is displayed in the window that the user is invalid. Otherwise, an affirmative sound and a message that the login to the system was successful was released.

After successfully entering the system, a new window appears with information about the patient (see Figure 1.3). The patient must wait until the information loading is finished.



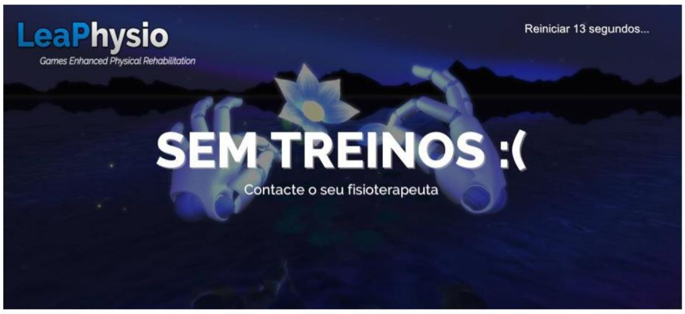
Bem-Vindo Filipe Da Costa :)

Nome: Filipe Da Costa | Idade: 26

A carregar 23 %

Figure 1.3 – Welcome window

If the patient did not have active training, this information is given to the patient through a new window (see Figure 1.4) And after a certain pre-established time the application is restarted for the first splash screen.



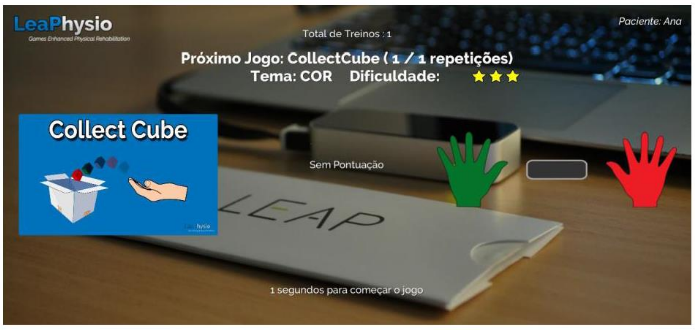
Reiniciar 13 segundos...

SEM TREINOS :(

Contacte o seu fisioterapeuta

Figure 1.4 – Empty window

If the patient has active training, a window on the training is presented (see Figure 1.5).



Paciente: Ana

Total de Treinos : 1

Próximo Jogo: CollectCube ( 1 / 1 repetições)

Tema: COR Dificuldade: ★★★

Sem Pontuação

1 segundos para começar o jogo

Figure 1.5 – Planning Trainer window

This window indicates various information:

- What's the Game?
- What is the difficulty?
- What is the thematic of the game?
- What is the gender of the patient?
- Hands?
- How long is the game?

After finishing the time to start the game, the application configures the next game. There are two games available, **CollectCube game** and **PickaBall game**.

**CollectCube game**

The game aims to have the patient place the cubes in the corresponding box to earn points. During the game time cubes appear on the scene, for each correct cube the patient earns a point and for each failure loses a point. Figure 1.6 illustrates an example of the CollectCube game.

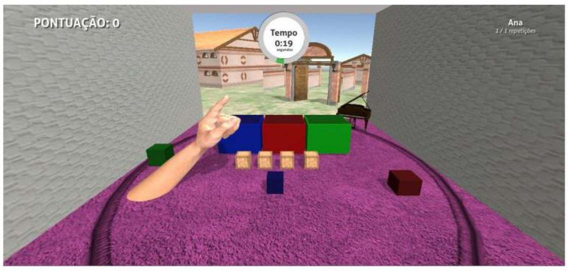


Figure 1.6 – CollectCube game

This game has three levels of difficulty:

- **EASY**: The scenario only has a single box (green) and over time appear green cubes;
- **MEDIUM**: The scenario has two boxes (green and red) and over time randomly appear cubes with green or red color;
- **HARD**: The scenario has three boxes (green, red and blue) and over time randomly appear cubes with green, red or blue color.

In the difficulty level MEDIUM and HARD the boxes change places with each other in a pre-established period. This is for the patient to be aware of the game, always checking the position of the box when it grabs a cube. Figure 1.7 illustrates the three boxes and respective cubes.

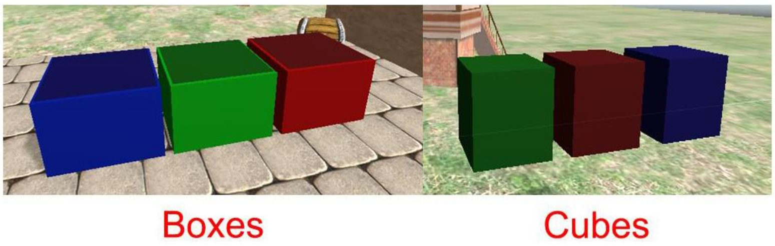


Figure 1.7 – Boxes and cubes of the CollectCube game

This game has two thematic, the COLOR thematic that is visible in Figure 1.6 that was previously explained and the THEMES thematic. This thematic makes the boxes and cubes present in the COLOR thematic are replaced by other objects, but the goal of the game is the same. Figure 1.8 illustrates the game with THEMES thematic.

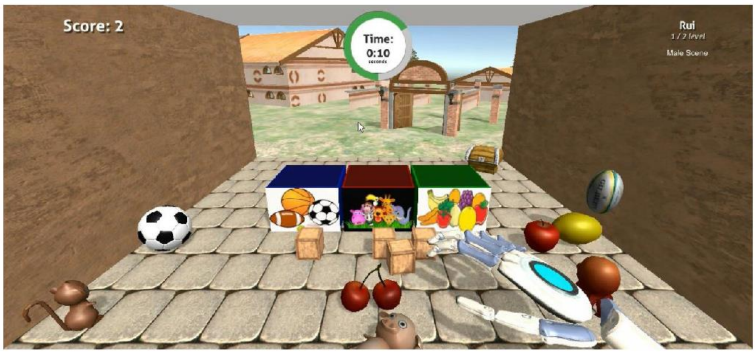


Figure 1.8 – CollectCube game with THEMES thematic

This thematic has objects and boxes with different themes. The objects can be of the type sport, fruit or animals. The patient should grab an object of a certain category and put in the corresponding box, for example, place the apple object in the fruit box. Many patients don’t only have problems with their hands and fingers, but many also have problems with cerebral palsy, which affects their ability to perform tasks. Through this thematic with categories that have objects known day-to-day, it can become useful to help patients with these problems, recognizing patterns to facilitate the execution of the proposed exercises. This thematic also has levels of difficulty with the characteristics explained above. Figure 1.9 illustrates the boxes and their objects used in the thematic.

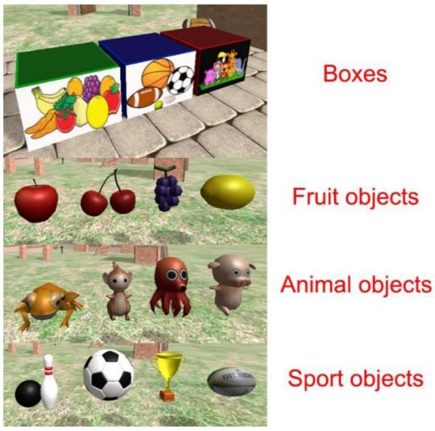


Figure 1.9 – Boxes and objects of the THEMES thematic

When the patient finishes the game, the results in terms of scores are stored in the database. Rooms different types of scores are stored:

- Game score;
- Total cubes or objects that appeared in the game;
- Total cubes or objects that have been correctly placed in their box;
- Total cubes or objects that haven't been correctly placed in their box.

**PickaBall game**

The game aims to get the patient to grab the available ball on the stage and pick up the cubes with the ball. For each cube caught the patient gains a point A Figure 1.10 shows an example of the PickaBall game.

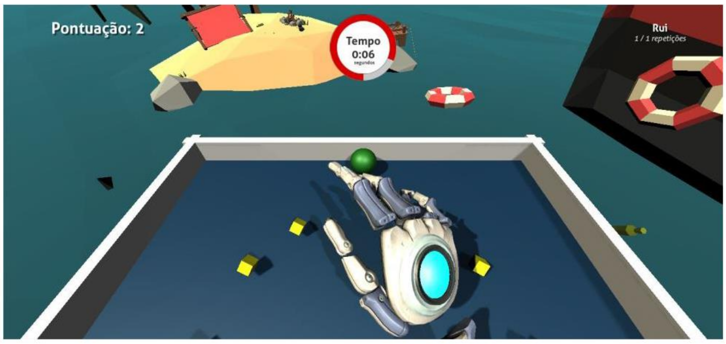


Figure 1.10 – PickaBall Game

After the patient picks up all available cubes with the ball, a coin-shaped object appears in the scenario (see Figure 1.11). The patient must pick up these coin, new cubes will appear on the game board.

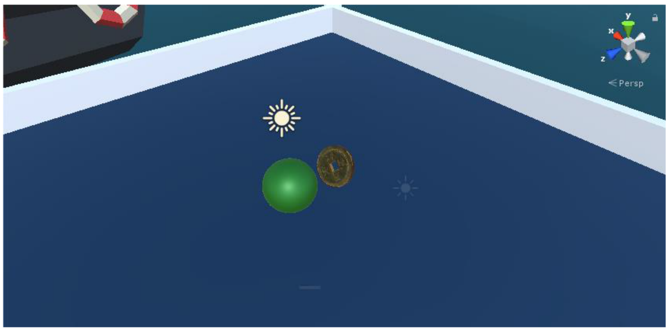


Figure 1.11 – PickaBall game coin object

The ball of the game is available for the patient to grip, but is blocked in height, that is, the patient cannot grasp the ball and move it in height, only in width and length. For each level of difficulty, the position of the cubes is different, as shown in Figure 1.11.

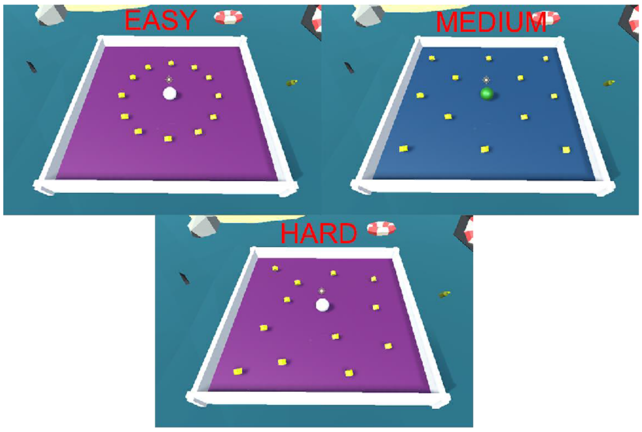


Figure 1.12 – Position of the cubes in the game PickaBall according of the level of difficulty

This game has a unique thematic, the FORMS thematic. When the patient finishes the game, the results in terms of scores are stored in the database. Two different types of scores are stored:

- Total cubes that appeared on the board;
- Total number of cubes taken by the patient.

After finishing the game, a window with the patient's score is shown (see Figure 1.13).

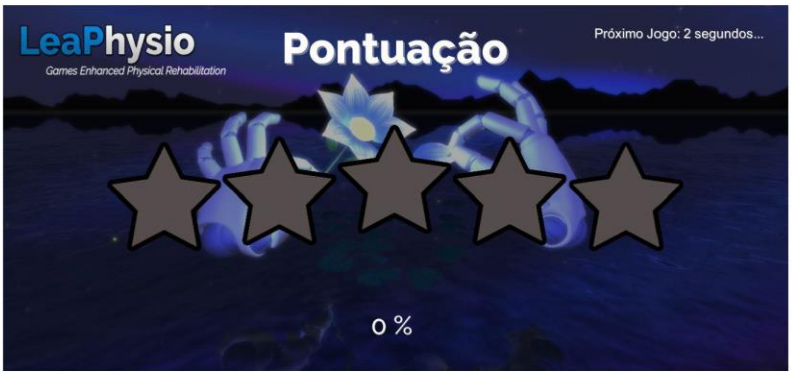

LeaPhysio
Games Enhanced Physical Rehabilitation
Pontuação
Próximo Jogo: 2 segundos...
0 %

Figure 1.13 – Last game score window

The application analyzes if there is any more training plan with repetitions to run. If it exists, the window corresponding to the training (see Figure 1.5) with the information for the following game is displayed again. Otherwise, a new window is displayed (see Figure 1.14) with the information that the execution of all the training has ended. After a pre-set time, the application is restarted for the first splash screen (see Figure 1.1).

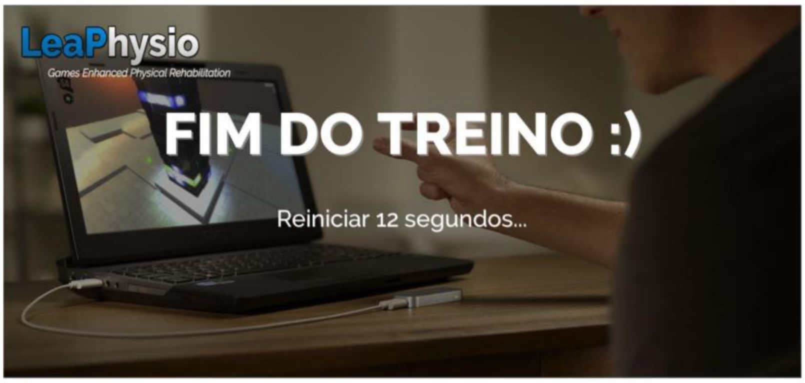


Figure 1.14 – End of training window

If the physiotherapist wishes to restart the application, for example due to an anomaly or to verify that the training isn't the most suitable for the patient, exist the ESC key to exit the application. This method is available in any application window.

# Chapter 2 – LeaPhysio Mobile Application

## 2. Overview

There are two types of users in this application, **physiotherapist** and **manager**. Each participant has access to a restricted area in the application, with different functionalities, physiotherapists have functionalities for patient management, which includes the possibility of creating and managing new training plans, visualizing results and creating new patients in the system, while the manager has features to manage and create new physiotherapists in the system. For each user to have access to his restricted area in the application, enter his credentials (identification number and password), enter the identification number in the first field, the password in the second field and click the Login button. Figure 2.1 represents the area where the user enters the credentials.

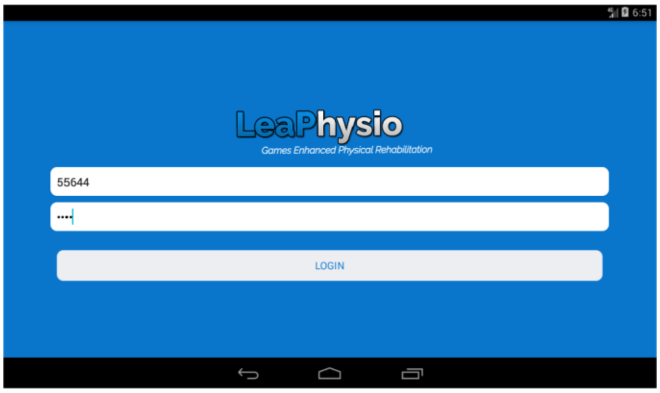
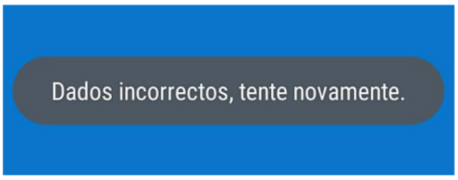


Figure 2.1 – Mobile application login screen

If the credentials are incorrect, that is, the application does not find any user with these credentials on the system, the application displays a message indicating that the credentials are incorrect (see Figure 2.2).



Dados incorrectos, tente novamente.

Figure 2.2 – Warning message for wrong credentials

If credentials are valid, a welcome message is displayed to the user (see Figure 2.3), with access to a workspace according to their privileges.

Bem-Vindo Fisioterapeuta

OR

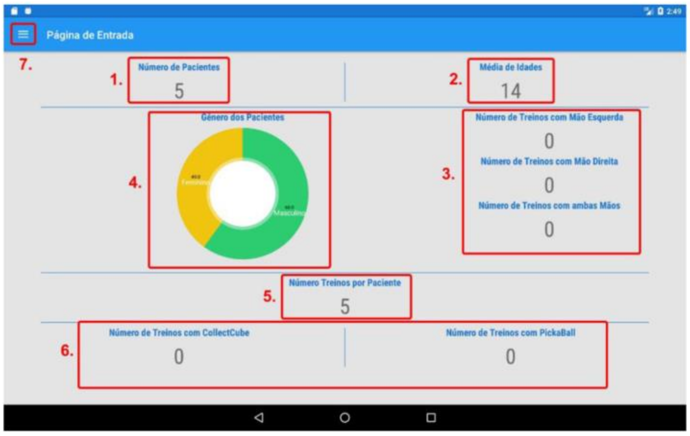
Bem-Vindo Gerente

Figure 2.3 – Welcome message

The application requires an internet connection to work properly, whether via Wi-Fi or mobile network (3G / 4G). As both users have different features in the application and for a better understanding of them, they are divided into two sub-chapters. Subchapter 2.1 is intended for the physiotherapist, while subchapter 2.2 is intended for the manager.

## 2.1. Physiotherapist Manual

This subchapter is intended for the user with Physiotherapist privileges. After successful login, the application shows the corresponding homepage (see Figure 2.4).



Página de Entrada

Número de Pacientes
5

Média de Idades
14

Género dos Pacientes

Número de Treinos com Mão Esquerda
0
Número de Treinos com Mão Direita
0
Número de Treinos com ambas Mãos
0

Número Treinos por Paciente
5

Número de Treinos com CollectCube
0

Número de Treinos com PickaBall
0

Figure 2.4 – Homepage of the physiotherapist in the mobile application

The Homepage has some information and functionalities:

- **1. →** ***Número de Pacientes***: indicates the number of patients belonging to the user;
- **2. →** ***Média de Idades***: mean age of patients;
- **3. →** ***Número de Treinos por Mão***: indicates the number of active trainings with the left hand, right hand or both;

- **4. →** ***Género dos Pacientes***: circular chart to represent the gender of the patients (Female or Male);
- **5. →** ***Número de Treinos por Paciente***: average of trainings (active or expired) per patient;
- **6. →** ***Número de Treinos por Jogo***: number of active training for existing games (CollectCube Game and PickaBall Game);
- **7. →** ***Botão de Menu***: Button that opens the main application menu. It contains several functionalities for the Manager (see Figure 2.5).

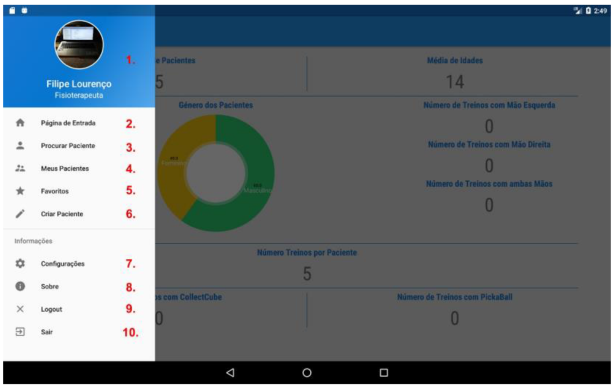


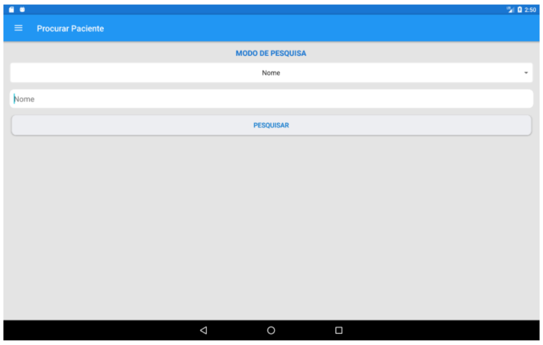
Figure 2.5 – Physiotherapist main application menu

The Menu has some functionalities and actions:

- **1. →** ***Cabeçalho***: it is possible to view the user's profile picture, its name and its user privilege (in this case it is Physiotherapist);
- **2. →** ***Página de Entrada***: returns the application entry page (Homepage);
- **3. →** ***Procurar Paciente***: search for patients in the system;
- **4. →** ***Meus Pacientes***: view patients belonging to the user;
- 5. **→** ***Favoritos***: view the patients marked as favorite;
- **6. →** ***Criar Paciente***: create a user with patient privileges;
- 7. **→** ***Configurações***: feature set to manage the user account;
- **8. →** ***Sobre***: information about the application;
- **9. →** ***Logout***: Log out of the application;
- **10. →** ***Sair***: allows the user to exit the application. It does not end the session, that is, the next time you start the application, the application will use previously entered credentials to log in.

### 2.1.1. Search Patient

Select the Menu Button and select the option "*Procurar Paciente*". Then the application opens a new window to allow the search of a Patient (see Figure 2.6).



MODO DE PESQUISA

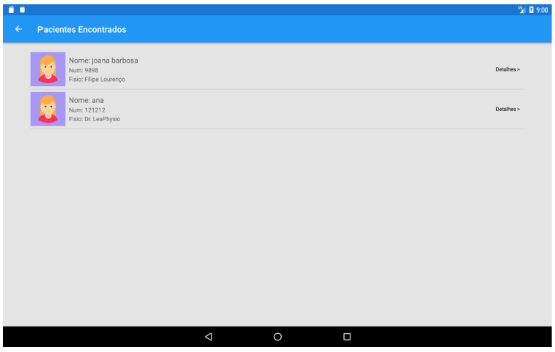
Nome

Nome

Figure 2.6 – Search patient

The user chooses the search mode in the first field. There are 3 search modes:

- ***Nome***: the user enters the name of the patient in the second field. You can enter the full or partial name, for example, by typing the name "Ana" in the second field and submit the search, the application will search all patients who have the word "Ana" in their name, presenting a list with results see Figure 2.7;
- ***Número de Identificação***: the user enters the patient identification number in the second field. The application upon finding the patient with this identification number, opens the patient's profile;
- ***Leitor de QRCode***: the application opens a new window to allow the user to display the patient's LeaPhysio card (see Figure 2.8). The application when finding the QrCode, will look for the patient respecitvo and open the profile.



Nome: joana barbosa
Num: 9898
Fisio: Filipe Lourenço

Nome: ana
Num: 121212
Fisio: Dr. LeaPhysio

Figure 2.7 – Patient list

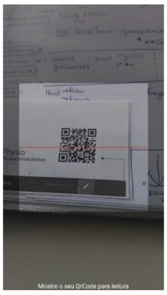


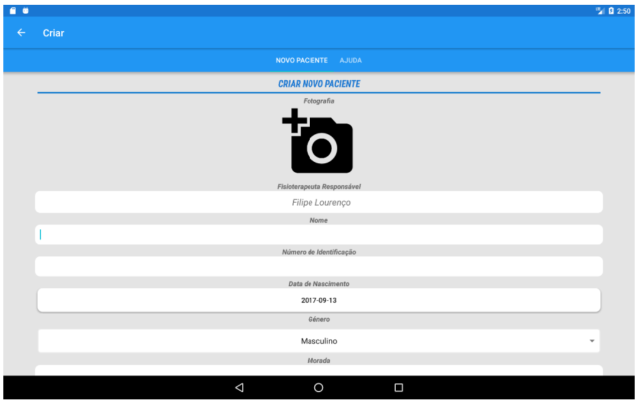
Figure 2.8 – QRCode reader in application

### 2.1.2. View my Patients

The user accesses the Menu Button and select the option “*Meus Pacientes*”. If there is no Patient associated with the Physiotherapist, the Physiotherapist doesn’t have any patient at charge, a message with the information that no result has been found is shown. Otherwise, a list of Patients is shown (see Figure 2.7). Clicking on the desired list component opens the Patient profile (see Figure 2.17).

### 2.1.3. Create a Patient

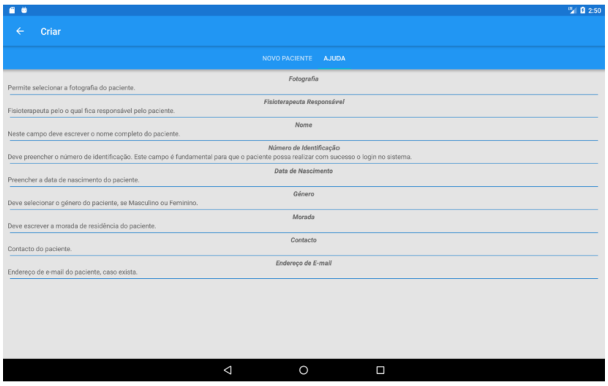
The user accesses the Menu Button and select the option “*Criar Paciente*”. Clicking the button displays a new window with fields to fill in to register a new Patient in the system (see Figure 2.9).



Fisioterapeuta Responsável: Filipe Lourenço
Nome:
Número de Identificação:
Data de Nascimento: 2017-09-13
Género: Masculino
Morada:

Figure 2.9 – Create new patient

In the upper part, the user has two tabs. The "*Novo Paciente*" tab refers to the window that corresponds to the user record, while the "*Ajuda*" tab (see Figure 2.10) serves to help the user fill in the fields for the record.



***Fotografia***
Permite selecionar a fotografia do paciente.

***Fisioterapeuta Responsável***
Fisioterapeuta pelo o qual fica responsável pelo paciente.

***Nome***
Neste campo deve escrever o nome completo do paciente.

***Número de Identificação***
Deve preencher o número de identificação. Este campo é fundamental para que o paciente possa realizar com sucesso o login no sistema.

***Data de Nascimento***
Preencher a data de nascimento do paciente.

***Género***
Deve selecionar o género do paciente, se Masculino ou Feminino.

***Morada***
Deve escrever a morada de residência do paciente.

***Contacto***
Contacto do paciente.

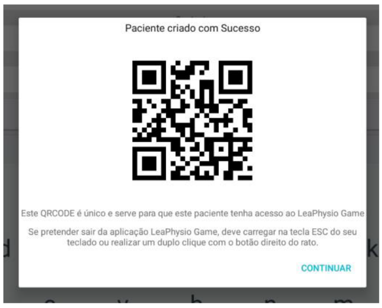
***Endereço de E-mail***
Endereço de e-mail do paciente, caso exista.

Figure 2.10 – Help panel to create new patient

In the "*Novo Paciente*" tab, the user needs to complete the following fields:

- ***Fotografia***: When clicking on the image, the user has two options, add a new photo or remove photo. Regarding the first option, the Camera application of the device is opened so that the user takes a picture of the Patient. In the second option, you can remove the previously added photo. It is not mandatory to take a photograph to the Physiotherapist, if you do not add and submit the registration, a photograph is added by default;
- ***Fisioterapeuta Responsável***: Name of Physiotherapist Responsible. This field cannot be changed;
- ***Nome***: Patient's Name;
- ***Número de Identificação***: Identification number. This field is essential so that the Patient can successfully login to the application through his LeaPhysio card. The system doesn't support two Patients with the same identification number;
- ***Data de Nascimento***: Date of birth of the patient. By clicking the date button, can select the correct date;
- ***Género***: Patient Gender (Male or Female);
- ***Morada***: Patient's home address;
- ***Contacto***: Patient Contact;
- ***Endereço de E-mail***: Patient's E-mail Address. This field isn't required.

If the fields aren't filled, the user is shown a message with the information to fill in the fields. If the fields are filled and there is no anomaly (for example, not connected to the internet), the patient is successfully created in the system (see figure 2.11). The QRCode displayed in the message is the same as the LeaPhysio card intended for the patient.



Paciente criado com Sucesso

Este QRCODE é único e serve para que este paciente tenha acesso ao LeaPhysio Game

Se pretender sair da aplicação LeaPhysio Game, deve carregar na tecla ESC do seu teclado ou realizar um duplo clique com o botão direito do rato.

Figure 2.11 – Successfully created patient

### 2.1.4. Settings

The user accesses the Menu Button and select the option "*Configurações*". Clicking the button displays a new window with the following options (see Figure 2.12):

- **1**. → ***Minhas Informações***: user information (Physiotherapist);
- **2**. → ***Alterar Password***: change the password;
- **3.** → ***Reportar Erro***: view information about how the user should report an error or bug in the application.

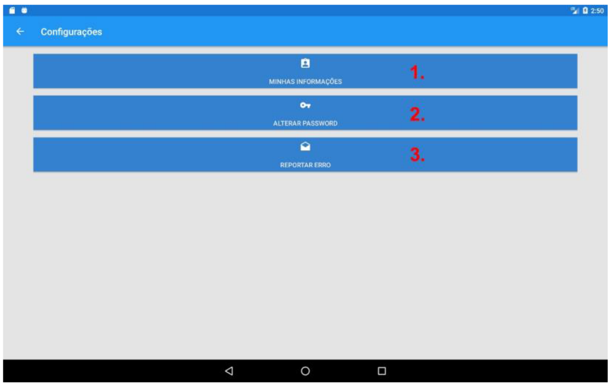
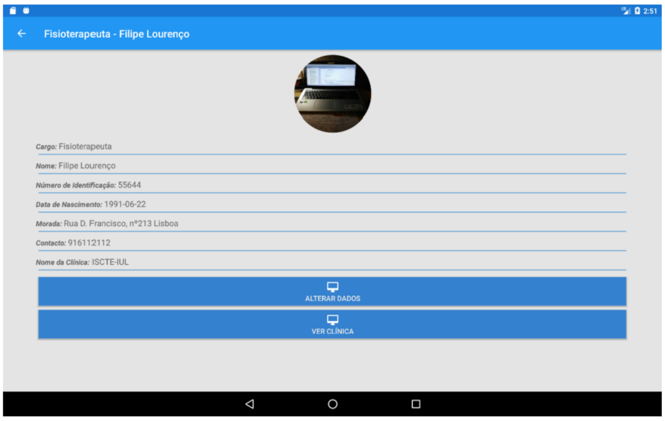


Figure 2.12 – User settings (Physiotherapist)

### 2.1.5. Viewing and Changing User Information

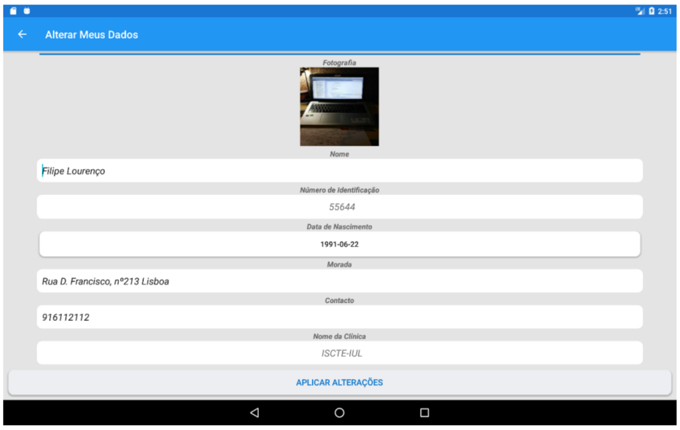
To view the user information, click the "*Minhas Informações*" button in the "*Configurações*" menu. Next, a new window (see Figure 2.13) is displayed with the user information.



Cargo: Fisioterapeuta
Nome: Filipe Lourenço
Número de Identificação: 55644
Data de Nascimento: 1991-06-22
Morada: Rua D. Francisco, nº213 Lisboa
Contacto: 916112112
Nome da Clínica: ISCTE-IUL

Figure 2.13 – Information about the physiotherapist

To change personal data, click the “*Alterar Dados*” button. A new window for editing personal data is displayed (see Figure 2.14).



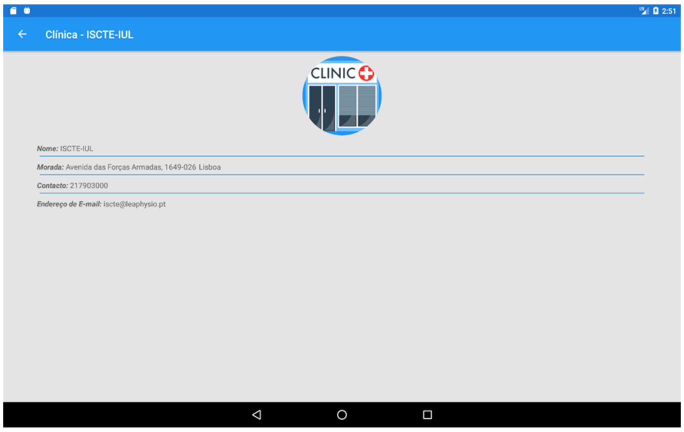
Nome: Filipe Lourenço
Número de Identificação: 55644
Data de Nascimento: 1991-06-22
Morada: Rua D. Francisco, nº213 Lisboa
Contacto: 916112112
Nome da Clínica: ISCTE-IUL

Figure 2.14 – Change physiotherapist data

The user clicks the “*Aplicar Alterações*” button. If the fields aren’t filled in, a message with the information to fill in the fields is shown. If no malfunction occurs (for example, no connected to the internet), the changes are successful.

### 2.1.6. View Clinic Information

Click the "*Minhas Informações*" button and then the "*Ver Clínica*" button in the Settings. Next, a new window (see Figure 2.15) is displayed with clinic information.



**Nome:** ISCTE-IUL

**Morada:** Avenida das Forças Armadas, 1649-026 Lisboa

**Contacto:** 217903000

**Endereço de E-mail:** iscte@leaphysio.pt

Figure 2.15 – Clinic information

### 2.1.7. Change Password

Click the "*Alterar Password*" button in the Settings. A new window is displayed (see Figure 2.16) to change the password.

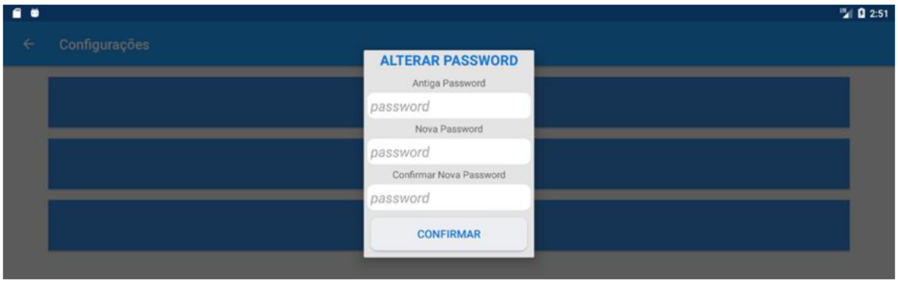
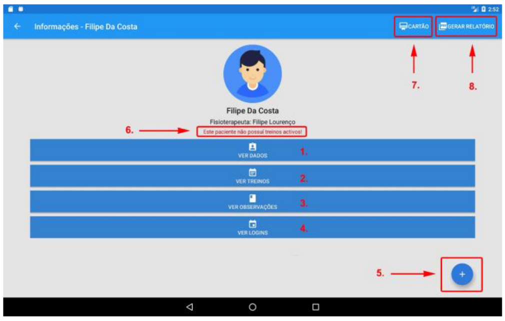


Figure 2.16 – Change user password

After completing all the fields correctly, clicking the "*Confirmar*" button and if no malfunction occurs, a message confirming that the password has been successfully changed is displayed.

### 2.1.8. Patient Profile

In the patient profile, the user has access to the following functionalities (see Figure 2.17):



Filipe Da Costa

Fisioterapeuta: Filipe Lourenço

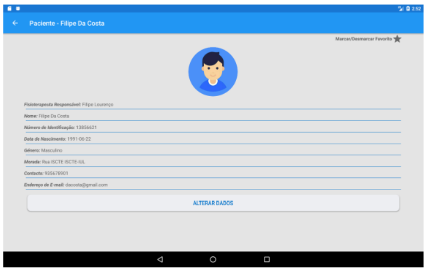
Este paciente não possui treinos activos!

Figure 2.17 – Patient profile

- **1. → *Ver Dados***: displays the personal data of the Patient;
- **2. → *Ver Treinos***: view the patient's trainings;
- **3. → *Ver Observações*:** allows visualizing the patient's observations. It is also possible to remove observations;
- **4. → *Ver Logins*:** visualize the logins performed in the application intended for patients;
- **5. → *Botão Criar*:** create new observations and trainings;
- **6. → *Mensagem de aviso*:** Indicates that the patient doesn't have any active training;
- **7. → *Ver Cartão*:** LeaPhysio card;
- **8. → *Gerar Relatório*:** generate a report with information about the patient.

### 2.1.9. View Patient Information

To view Patient information, click the "*Ver Dados*" button. Next, a new window appears (see Figure 2.18) with information about the patient.



***Fisioterapeuta Responsável:*** Filipe Lourenço

***Nome:*** Filipe Da Costa

***Número de Identificação:*** 13856621

***Data de Nascimento:*** 1991-06-22

***Género:*** Masculino

***Morada:*** Rua ISCTE ISCTE-IUL

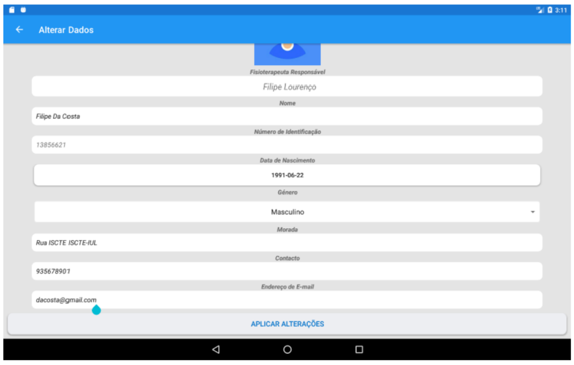
***Contacto:*** 935678901

***Endereço de E-mail:*** dacosta@gmail.com

Figure 2.18 – Patient information

### 2.1.10. Changing Patient Information

To change the patient's personal data, click the "*Alterar Dados*" button in the patient profile. A new window appears for editing personal data (see Figure 2.19).



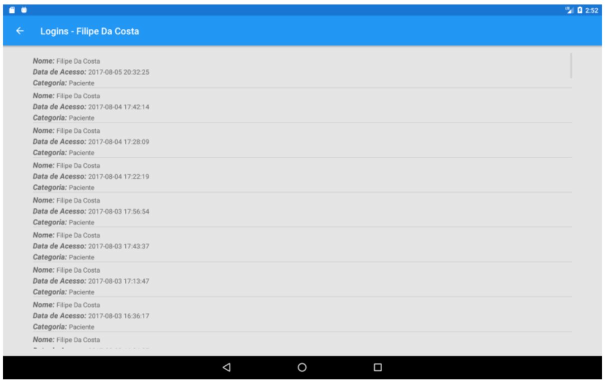
Fisioterapeuta Responsável
Filipe Lourenço
Nome
Filipe Da Costa
Número de Identificação
13856621
Data de Nascimento
1991-06-22
Género
Masculino
Morada
Rua ISCTE ISCTE-IUL
Contacto
935678901
Endereço de E-mail
dacosta@gmail.com

Figure 2.19 – Change patient information

After changing the desired fields, the user clicks the "*Aplicar Alterações*" button. If no anomaly occurs (for example, not connected to the internet), the changes are performed successfully.

### 2.1.11. View Patient Logins

To view patient logins, click the "*Ver Logins*" button in the patient profile. A new window appears (see Figure 2.20).



**Nome:** Filipe Da Costa
**Data de Acesso:** 2017-08-05 20:32:25
**Categoria:** Paciente

**Nome:** Filipe Da Costa
**Data de Acesso:** 2017-08-04 17:42:14
**Categoria:** Paciente

**Nome:** Filipe Da Costa
**Data de Acesso:** 2017-08-04 17:28:09
**Categoria:** Paciente

**Nome:** Filipe Da Costa
**Data de Acesso:** 2017-08-04 17:22:19
**Categoria:** Paciente

**Nome:** Filipe Da Costa
**Data de Acesso:** 2017-08-03 17:56:54
**Categoria:** Paciente

**Nome:** Filipe Da Costa
**Data de Acesso:** 2017-08-03 17:43:37
**Categoria:** Paciente

**Nome:** Filipe Da Costa
**Data de Acesso:** 2017-08-03 17:13:47
**Categoria:** Paciente

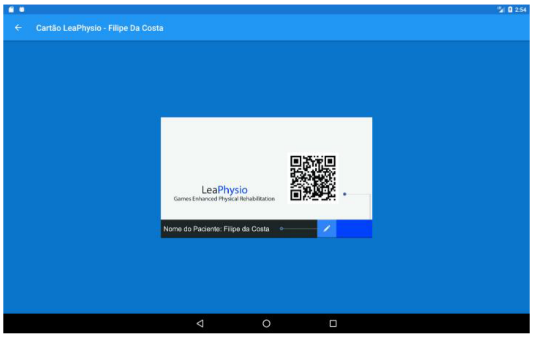
**Nome:** Filipe Da Costa
**Data de Acesso:** 2017-08-03 16:36:17
**Categoria:** Paciente

**Nome:** Filipe Da Costa

Figure 2.20 – Patient logins

### 2.1.12. View LeaPhysio Card

To view Patient's LeaPhysio card, click the View Card button in the patient profile. A new window appears (see Figure 2.21).


2:54
Cartão LeaPhysio - Filipe Da Costa
LeaPhysio
Games Enhanced Physical Rehabilitation

Nome do Paciente: Filipe da Costa

Figure 2.21 – LeaPhysio Card

### 2.1.13. Generate Report

To generate a report with information about the Patient, click the “Gerar Relátorio” button. This report is created and stored on the Android device in the ***LeaPhysioPDF*** folder in .PDF format. The file name contains the following format.

- **[Patient Name]_[YYYYMMDD]_[HHMMSS].pdf**
- Example: **Filipe Da Costa_20170914_131012.pdf**

After being stored on the device, the user can open the document, having an application installed on the device to open .pdf documents (see Figure 2.22).

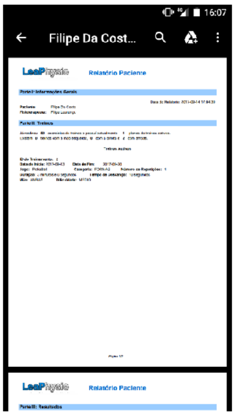


Figure 2.22 – Patient Report

### 2.1.14. View Favorites

The user accesses the Menu Button and select the Favorites option. If there is no Patient Signed as Favorite, a message is displayed with the information that no result has been found. If there are favorites, a list of patients is shown (see Figure 2.7).

### 2.1.15. Add and Remove Favorite

In the Patient profile, the user has a button in the upper right corner to mark or unmark a patient as a favorite. When the star is filled it means that the patient is marked as a favorite, otherwise it isn't a favorite for the user (see Figure 2.23).

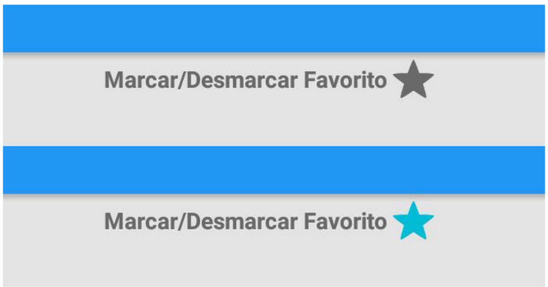
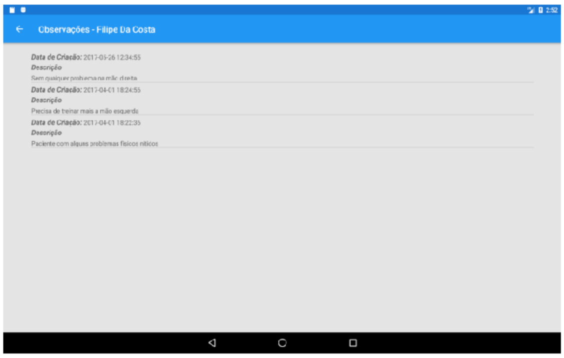


Figure 2.23 – Mark and unmark a patient as a favorite

### 2.1.16. Observation Patient

To view the patient's observations, click the "*Ver Observações*" button in the Patient profile. A new window appears (see Figure 2.24).


Observações - Filipe Da Costa

*Data de Criação:* 2017-05-26 12:34:55
*Descrição*
Sem qualquer problema na mão direita

*Data de Criação:* 2017-04-01 18:24:55
*Descrição*
Precisa de treinar mais a mão esquerda

*Data de Criação:* 2017-04-01 18:22:35
*Descrição*
Paciente com alguns problemas fisicos niticos

Figure 2.24 – List of patient observations

To remove a note, the user accesses the patient's observation list (see Figure 2.24) and press the note to delete. After pressing, a message appears prompting the user to delete the observation.

To create an observation, access the Patient profile (see Figure 2.17), click the "*Criar*" button (lower right corner) and select New Observation. A new window is opened (see figure 2.25), where the user should write the observation. After finish, click the "*Criar Nova Observação*" button to submit it to the system.

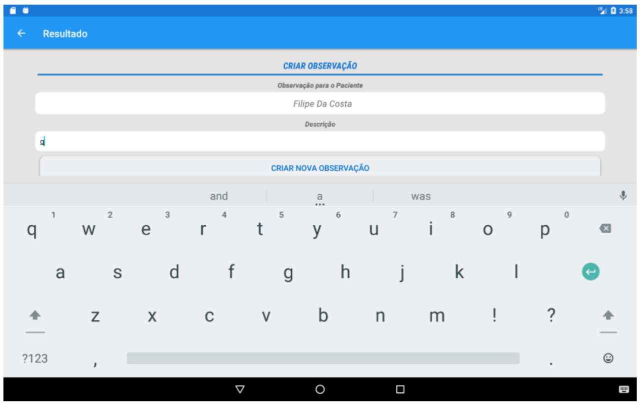
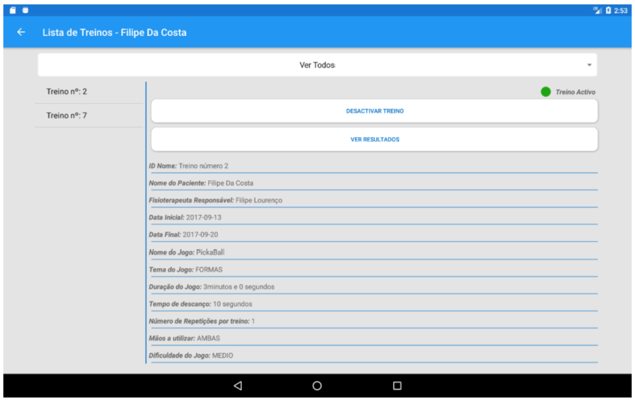


Figure 2.25 – Create a new observation for the patient

## 2.1.17. View Trainer

To view the patient's trainings, access the Patient profile (see Figure 2.17) and click on the " *Ver Treinos*" button. A new window is opened with the patient's trainings (see Figure 2.26).



*ID Nome:* Treino número 2

*Nome do Paciente:* Filipe Da Costa

*Fisioterapeuta Responsável:* Filipe Lourenço

*Data Inicial:* 2017-09-13

*Data Final:* 2017-09-20

*Nome do Jogo:* PickaBall

*Tema do Jogo:* FORMAS

*Duração do Jogo:* 3minutos e 0 segundos

*Tempo de descanço:* 10 segundos

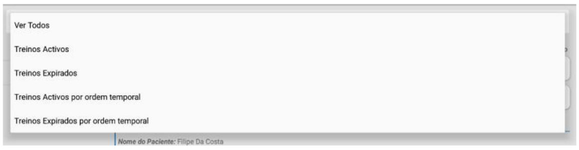
*Número de Repetições por treino:* 1

*Mãos a utilizar:* AMBAS

*Dificuldade do Jogo:* MEDIO

Figure 2.26 – Patient trainings

At the top of the screen there is a list to sort the trainings (see Figure 2.27), with the following options:


Ver Todos
Treinos Activos
Treinos Expirados
Treinos Activos por ordem temporal
Treinos Expirados por ordem temporal

Nome do Paciente: Filipe Da Costa

Figure 2.27 – Options for viewing patient training

- ***Ver Todos*:** shows in the list all existing trainings (active or expired);
- ***Treinos Ativos*:** shows only the active trainings;
- ***Treinos expirados*:** shows only the expired trainings;
- ***Treinos Ativos por ordem temporal*:** shows active trainings in descending order (from the most recent to the oldest);
- ***Treinos expirados por ordem temporal*:** shows the expired trainings in descending order (from the most recent to the oldest).

The user can see if the training is active or expired (see Figure 2.28). If it is active, a button appears to deactivate the training, otherwise a button to activate it.

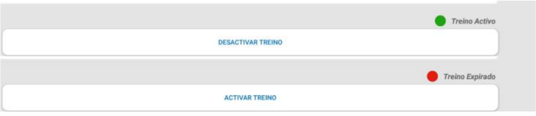


Figure 2.28 – Training state

The user also has the possibility to change training data and erase trainings.

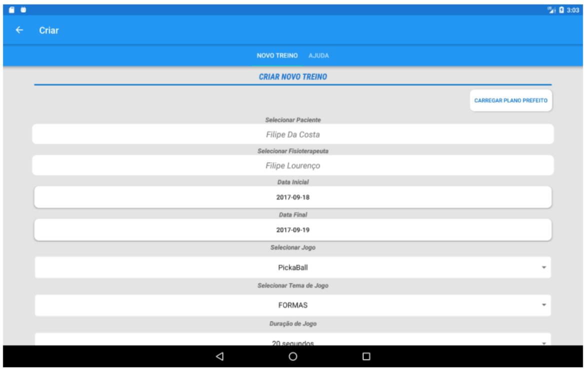
### 2.1.18. Enable and Disable Training

To disable or activate a workout, accesses the Patient profile (see Figure 2.17), click the "*Ver Treinos*" button. A new window opens with patient training (see Figure 2.26). When a training is active, there is the option to deactivate the training, available through the "*Desativar Treino*" button.

When a training is disabled, the user has the option to activate this training, using the "*Ativar Treino*" button.

### 2.1.19. New Trainer

To create a new training, the user accesses the Patient profile (see Figure 2.17), click the "*Criar*" button (lower right corner) and select New Training. A new window is opened (see Figure 2.29).



CRIAR NOVO TREINO

CARREGAR PLANO PREFEITO

Selecionar Paciente
Filipe Da Costa

Selecionar Fisioterapeuta
Filipe Lourenço

Data Inicial
2017-09-18

Data Final
2017-09-19

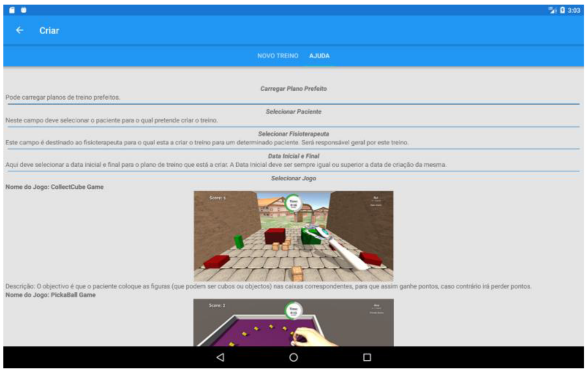
Selecionar Jogo
PickaBall

Selecionar Tema de Jogo
FORMAS

Duração de Jogo
20 segundos

Figure 2.29 – Create new training

At the top, there are two tabs. The "*Novo Treino*" tab refers to the window to create a new workout for the patient, while the "*Ajuda*" tab (see Figure 2.30) serves to help the user fill in the fields for registration. Contains information about available games, themes and different levels of difficulty.



Carregar Plano Prefeito
Pode carregar planos de treino prefeitos.

Selecionar Paciente
Neste campo deve selecionar o paciente para o qual pretende criar o treino.

Selecionar Fisioterapeuta
Este campo é destinado ao fisioterapeuta para o qual esta a criar o treino para um determinado paciente. Será responsável geral por este treino.

Data Inicial e Final
Aqui deve selecionar a data inicial e final para o plano de treino que está a criar. A Data Inicial deve ser sempre igual ou superior a data de criação da mesma.

Selecionar Jogo
Nome do Jogo: CollectCube Game

Descrição: O objectivo é que o paciente coloque as figuras (que podem ser cubos ou objectos) nas caixas correspondentes, para que assim ganhe pontos, caso contrário irá perder pontos.
Nome do Jogo: PickaBall Game

Figure 2.30 – Help tab to create new training

After completing all the fields, the user clicks the "*Criar Novo Treino*" button. If the fields are all filled in and no anomaly occurs (example: not connected to the internet), training is successfully created in the system.

### 2.1.20. Delete Trainer

To delete a training, the user accesses the Patient profile (see Figure 2.17), click on the "Ver Treinos" button, select the workout to delete and click the "Apagar Treinos" button.

### 2.1.21. Change Training data

To change the training data, access the Patient profile (see Figure 2.17), click on the "*Ver Treinos*" button, select the training to change the data and click the "*Alterar Dados*" button. A new window for changing the data is displayed. After changing the required data, click the "*Alterar Dados*" button.

### 2.1.22. Results Scores

To view results in terms of scores, access the Patient profile (see Figure 2.17), click on the "*Ver Treinos*" button, choose the training you want and click the "*Ver Resultados*" button. A new window is shown with the results (see figure 2.31), if there are no results a warning message is shown to the user.

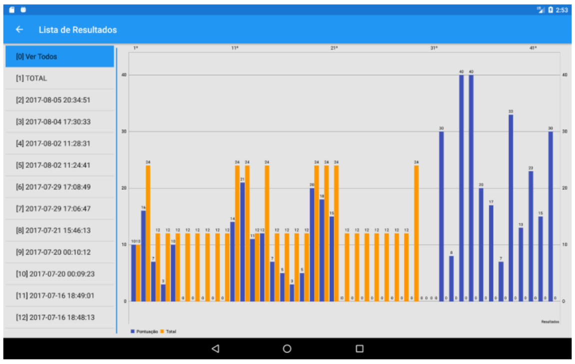
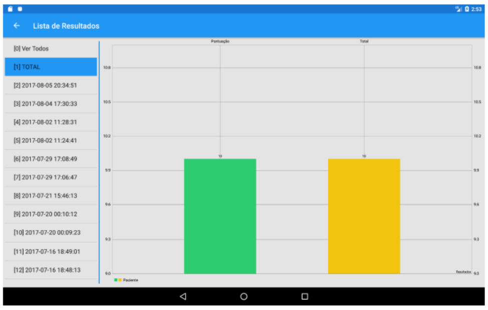


Figure 2.31 – Score Result Chart

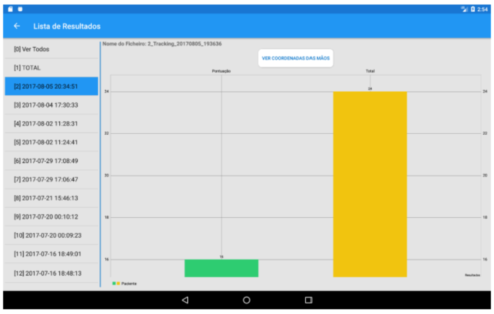
It is possible to observe three structures of results:

- ***Ver Todos***: joins all results into a single graph (see Figure 2.31);
- ***Total***: average of the patient's results (see Figure 2.32);
- ***Resultados individuais***: individual results performed by the patient, indicating the date and time (see Figure 2.33).



[0] Ver Todos
[1] TOTAL
[2] 2017-08-05 20:34:51
[3] 2017-08-04 17:30:33
[4] 2017-08-02 11:28:31
[5] 2017-08-02 11:24:41
[6] 2017-07-29 17:08:49
[7] 2017-07-29 17:06:47
[8] 2017-07-21 15:46:13
[9] 2017-07-20 00:10:12
[10] 2017-07-20 00:09:23
[11] 2017-07-16 18:49:01
[12] 2017-07-16 18:48:13

Figure 2.32 – All Score Results Chart



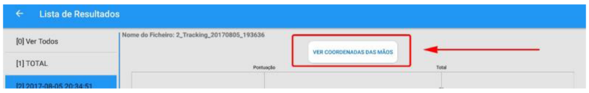
Nome do Ficheiro: 2_Tracking_20170805_193636

[0] Ver Todos
[1] TOTAL
[2] 2017-08-05 20:34:51
[3] 2017-08-04 17:30:33
[4] 2017-08-02 11:28:31
[5] 2017-08-02 11:24:41
[6] 2017-07-29 17:08:49
[7] 2017-07-29 17:06:47
[8] 2017-07-21 15:46:13
[9] 2017-07-20 00:10:12
[10] 2017-07-20 00:09:23
[11] 2017-07-16 18:49:01
[12] 2017-07-16 18:48:13

Figure 2.33 – Individual Score Result

### 2.1.23. Results Coordinates

In the Results List, select an individual training result and click the "*Ver Coordenadas das Mãos*" button (see Figure 2.34).

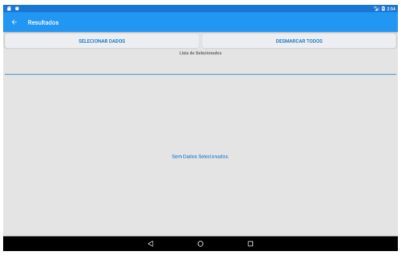


Nome do Ficheiro: 2_Tracking_20170805_193636

[0] Ver Todos
[1] TOTAL
[2] 2017-08-05 20:34:51

Figure 2.34 – Button to view coordinate results

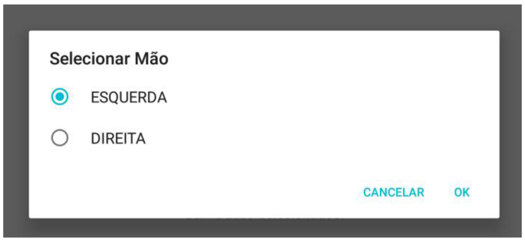
A new window opens for the user to select the data they want to see (see Figure 2.35).



Sem Dados Selecionados.

Figure 2.35 – Coordinate Results Window

To view information, click the "*Selecionar Dados*" button and choose which data you want (see Figure 2.36).



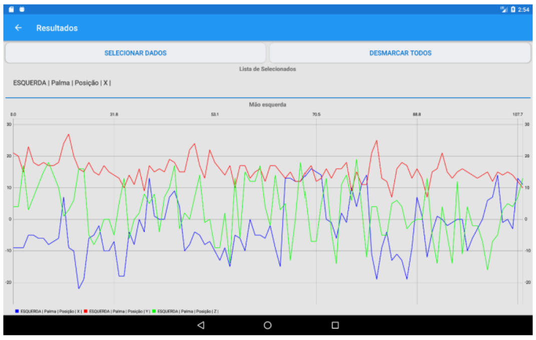
Selecionar Mão

● ESQUERDA

○ DIREITA

Figure 2.36 – Menu to select hands

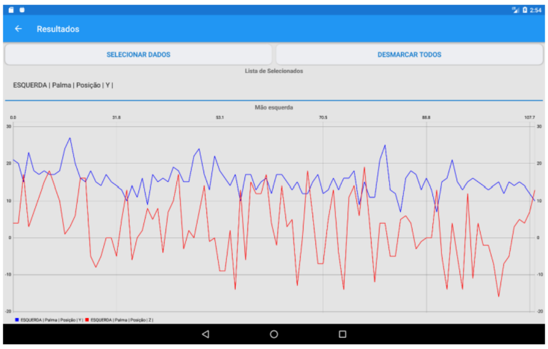
After selecting the data, they can be viewed graphically (see Figure 2.37).



ESQUERDA | Palma | Posição | X |

Figure 2.37 – Coordinates results window with a single chart

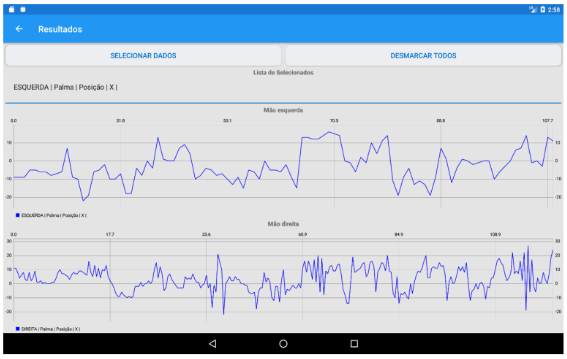
To delete all the information in the graph, click the "*Desmarcar Todos*" button. To remove a row from the chart, under "*Lista de Selecionados*" and choose which data want to remove, tap finger on the corresponding row (see Figure 2.38).



ESQUERDA | Palma | Posição | Y |

Figure 2.38 – Updated chart lines

If the user selects data from different hands, the information is shown in different charts, one for the left hand and one for the right hand (see Figure 2.39).



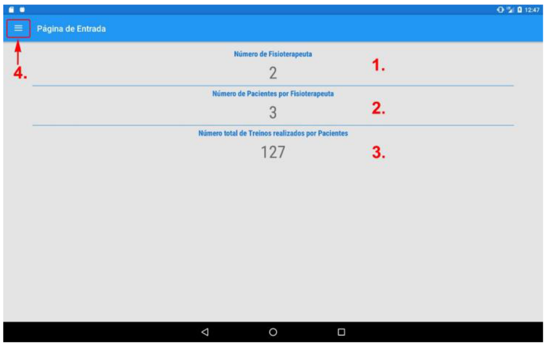
Lista de Selecionados

ESQUERDA | Palma | Posição | X |

Figure 2.39 – Coordinates results window with two charts

## 2.2. Manager Manual

This subchapter is intended for the user with Manager privileges. After entering the credentials successfully in the application, the homepage appears (see Figure 2.40).



Número de Fisioterapeuta
2

Número de Pacientes por Fisioterapeuta
3

Número total de Treinos realizados por Pacientes
127

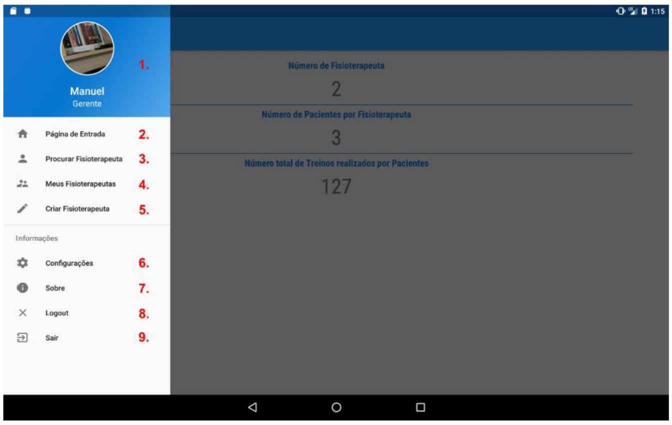
Figure 2.40 – Manager Homepage

On the Homepage there are some information / functionalities available:

- **1.** → ***Número de Fisioterapeuta***: indicates the number of Physiotherapists that exist in the clinic, which the Manager belongs to;
- **2.** → ***Número de Pacientes por Fisioterapeuta***: indicates the average number of Patients to which a Physiotherapist is responsible;
- 3. → ***Número Total de Treinos realizados por Pacientes***: indicates the total number of workouts performed by all Clinic Patients, for example, a training plan containing

two sets set up, each with two repetitions, after being performed by the Patient, 4 repetitions are counted and added to this count;

- **4. →** ***Botão de Menu***: Button that opens the main application menu. It contains several functionalities for the Manager (see Figure 2.41).



Número de Fisioterapeuta
2
Número de Pacientes por Fisioterapeuta
3
Número total de Treinos realizados por Pacientes
127

Figure 2.41 – Manager main application menu

- **1. →** ***Cabeçalho***: the profile picture of the user, their name and their user privilege (Manager);
- **2. →** ***Página de Entrada***: returns the application entry page (Homepage);
- **3. →** ***Procurar Fisioterapeuta***: search for a Physiotherapist in the system;
- **4. →** ***Meus Fisioterapeutas***: visualize Physiotherapists belonging to the user;
- **5. →** ***Criar Fisioterapeuta***: create a user with Physiotherapist privileges;
- **6. →** ***Configurações***: set of features to manage the user's account;
- **7. →** ***Sobre***: information about the application;
- **8. →** ***Logout***: log out of the application;
- **9. →** ***Sair***: allows the user to exit the application. Does not end the session, the next time you start the application, the application will use previously entered credentials.

### 2.2.1. Search Physiotherapist

The user accesses the Menu Button and select the option "*Procurar Fisioterapeuta*". The application opens a new window to allow the search of a specific Physiotherapist (see Figure 2.42).

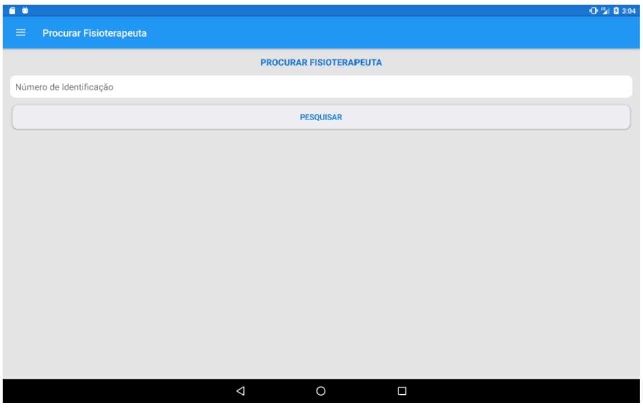
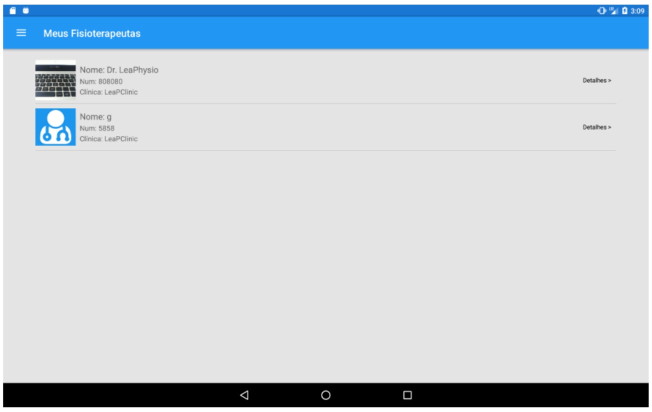


Figure 2.42 – Search physiotherapist

The user enters the Physiotherapist's Identification Number in the first text box and clicks the "*Procurar*" button. If there is no Physiotherapist with the identification number that was entered, a message with the information that no result was found is shown. If there is a Physiotherapist with the Identification Number entered, your profile is loaded (see Figure 2.46).

### 2.2.2. View my Physiotherapists

Access the menu button and select the "*Meus Fisioterapeutas*" option. If there is a Physiotherapist associated with the Manager, a list of Physiotherapists is shown (see Figure 2.2.43).



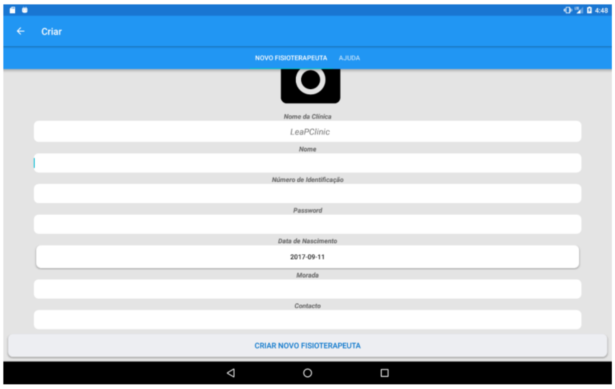
Nome: Dr. LeaPhysio
Num: 808080
Clínica: LeaPClinic

Nome: g
Num: 5858
Clínica: LeaPClinic

Figure 2.43 – Physiotherapist list

### 2.2.3. Create a Physiotherapist

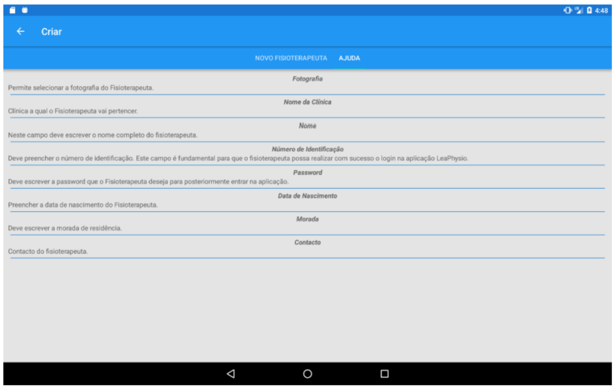
Access the menu button and select the "*Criar Fisioterapeuta*" option. Clicking on the button shows a new window with fields to fill to register a new Physiotherapist in the system (see Figure 2.44).



Nome da Clínica
LeaPClinic
Nome
Número de Identificação
Password
Data de Nascimento
2017-09-11
Morada
Contacto

Figure 2.44 – Create Physiotherapist window

At the top, the user has two tabs. The "*Novo Fisioterapeuta*" tab refers to the window that corresponds to the creation, the "*Ajuda*" tab (see Figure 2.45) serves to help the user fill in the fields for registration.



Fotografia
Permite selecionar a fotografia do Fisioterapeuta.
Nome da Clínica
Clínica a qual o Fisioterapeuta vai pertencer.
Nome
Neste campo deve escrever o nome completo do fisioterapeuta.
Número de Identificação
Deve preencher o número de identificação. Este campo é fundamental para que o fisioterapeuta possa realizar com sucesso o login na aplicação LeaPhysio.
Password
Deve escrever a password que o Fisioterapeuta deseja para posteriormente entrar na aplicação.
Data de Nascimento
Preencher a data de nascimento do Fisioterapeuta.
Morada
Deve escrever a morada de residência.
Contacto
Contacto do fisioterapeuta.

Figure 2.45 – Help panel to create new physiotherapist

After completing all the fields, the user clicks on the "*Criar Novo Fisioterapeuta*" button. If there is no anomaly (for example, not connected to the internet), the physiotherapist is successfully created in the system.

### 2.2.4. Physiotherapist Profile

In the physiotherapist profile, the user has access to the following functionalities (see Figure 2.46):

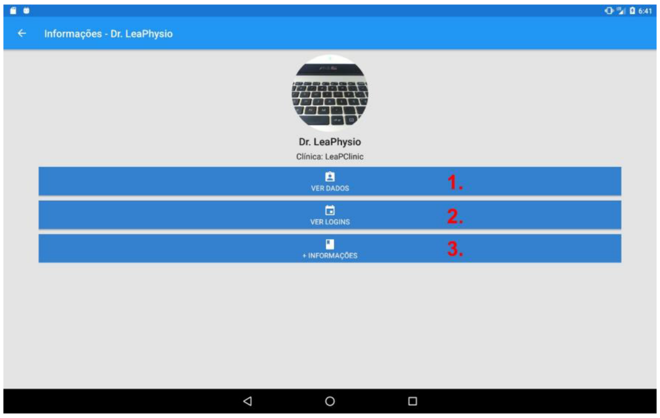
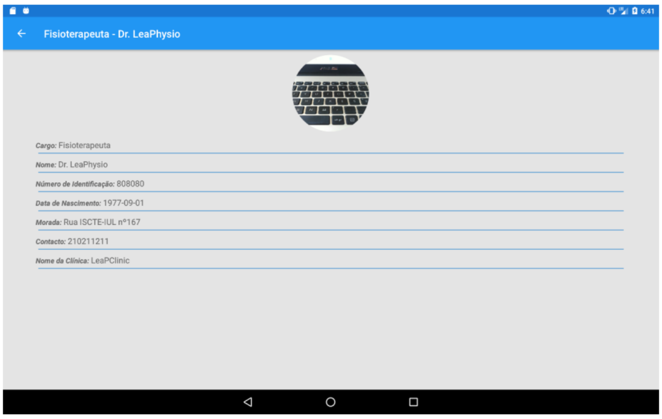


Figure 2.46 – Physiotherapist Profile

- **1. →** ***Ver Dados***: view the personal data of the Physiotherapist;
- **2. →** ***Ver Logins*****:** view the Physiotherapist logins in the application;
- **3. →** + ***Informações*****:** show more information about Physiotherapist.

### 2.2.5. View Physiotherapist Information

Click the "*Ver Dados*" button in the physiotherapist profile (see Figure 2.47). A new window opens with the data.



*Cargo:* Fisioterapeuta

*Nome:* Dr. LeaPhysio

*Número de Identificação:* 808080

*Data de Nascimento:* 1977-09-01

*Morada:* Rua ISCTE-IUL nº167

*Contacto:* 210211211

*Nome da Clínica:* LeaPClinic

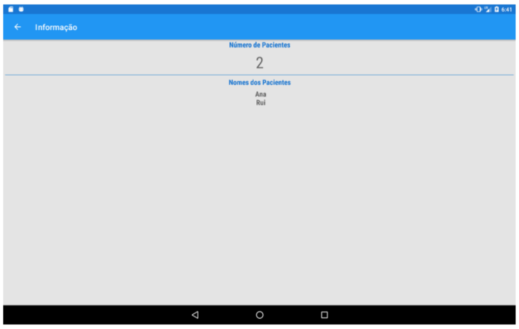
Figure 2.47 – Physiotherapist information

### 2.2.6. View Physiotherapist Logins

Click the “*Ver Logins*” button in the physiotherapist profile. A new window opens with the logins performed by the physiotherapist in the mobile application.

### 2.2.7. View More Information about Physiotherapist

Click the “+ *Informações*” button in the physiotherapist profile. A new window opens with the information (see Figure 2.48).



Número de Pacientes
2

Nomes dos Pacientes
Ana
Rui

Figure 2.48 – More Information about Physiotherapist

### 2.2.8. Settings

Search the Menu Button and select “*Configurações*” option. A new window appears with the following options (see figure 2.49):

- **1.** → ***Minhas Informações***: view user information (Manager);
- **2.** → ***Alterar Password***: change the password;
- 3. → ***Ver Clinica***: view the clinic data.

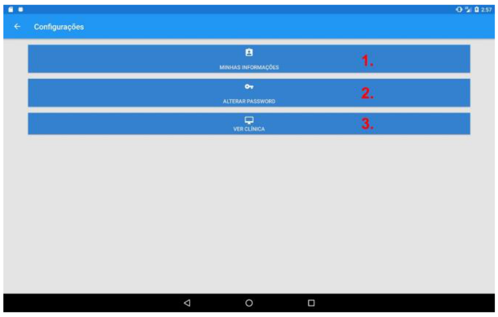
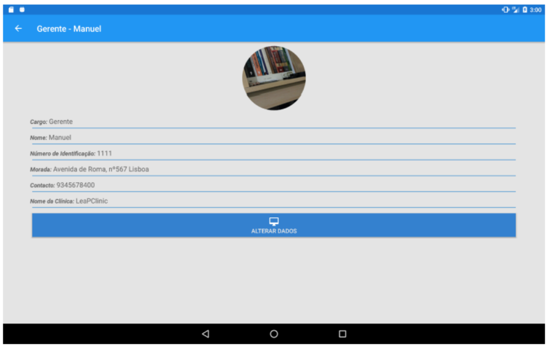


Figure 2.49 – User Settings (Manager)

### 2.2.9. Viewing and Changing Manager Information

Click the “*Minhas Informações*” button. A new window appears with the information (see Figure 2.50).



Cargo: Gerente

Nome: Manuel

Número de Identificação: 1111

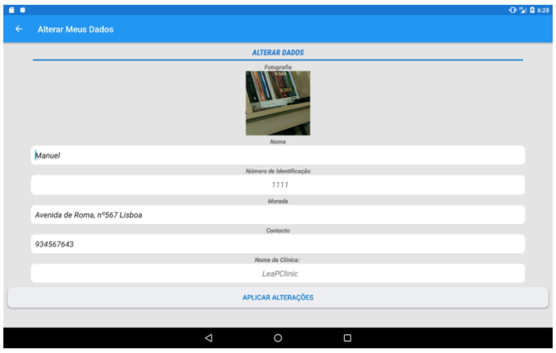
Morada: Avenida de Roma, nº567 Lisboa

Contacto: 9345678400

Nome da Clínica: LeaPClinic

Figure 2.50 – Information about the manager

For change personal data it is necessary to click the “*Alterar Dados*” button. A new window for editing personal data is displayed (see figure 2.51).



Nome: Manuel

Número de Identificação: 1111

Morada: Avenida de Roma, nº567 Lisboa

Contacto: 934567643

Nome da Clínica: LeaPClinic

Figure 2.51 – Change manager data

### 2.2.10. Change Manager Password

Need to click on the “*Alterar Password*” button, fill in the fields and click the “*Confirmar*” button. If no anomaly happened the password was changed.

### 2.2.11. View Clinic Information

Need to click on the “*Ver Clínica*” button. A new window appears with the information about clinic.

# Chapter 3 – Installation Manual

## 3. Overview

This chapter explains how to install the two LeaPhysio applications. First, how to install the application for the computer and then the mobile application.

Should accesses the pen drive with the contents of LeaPhysio, as shown in Figure 3.1.

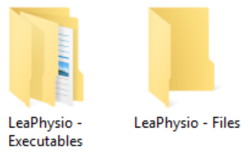


Figure 3.1 – Pen drive main folders

There are two folders, for this Chapter we will only use the "**LeaPhysio – Executables**" folder.

## 3.1. Computer Application Installation

Open the "**LeaPhysio – Computer Application**" folder (see Figure 3.2).

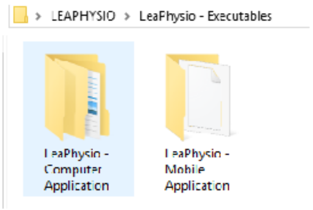

LEAPHYSIO > LeaPhysio - Executables
LeaPhysio - Computer Application
LeaPhysio - Mobile Application

Figure 3.2 – Contents of the "*LeaPhysio - Executables*" folder

When you open the folder, we find two folders (see Figure 3.3):

- ***Executable – Computer Application***: LeaPhysio application executable;
- ***Executable – Leap Motion Controller***: executable to install leap motion on computer.

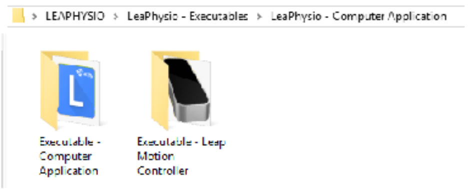
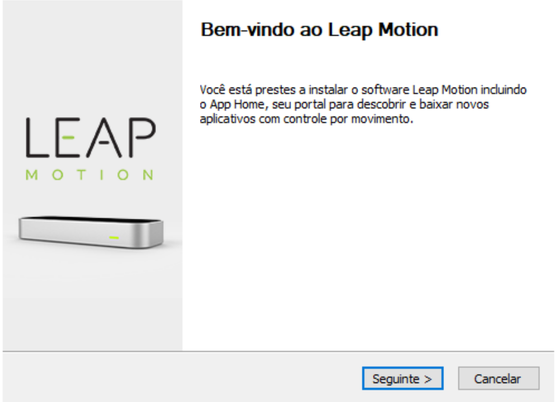

LEAPHYSIO > LeaPhysio - Executables > LeaPhysio - Computer Application
Executable - Computer Application
Executable - Leap Motion Controller

Figure 3.3 – Contents of the "*LeaPhysio – Computer Application*" folder

To install the executables, need a computer with the Windows operating system. Open the "**Executable - Leap Motion Controller**" folder and install the **Leap_Motion_Setup_Win_3.1.3.exe** on your computer, as shown in Figure 3.4.

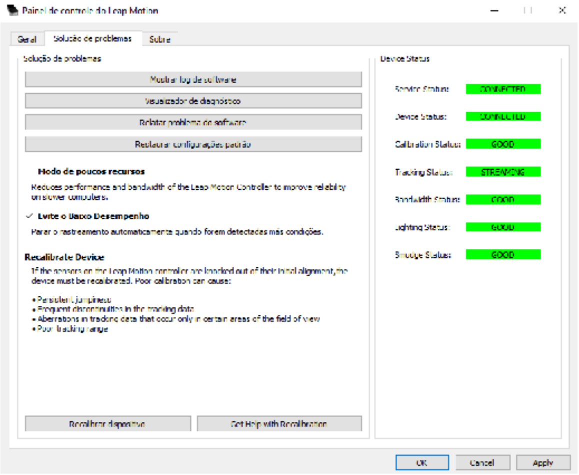


**Bem-vindo ao Leap Motion**

Você está prestes a instalar o software Leap Motion incluindo o App Home, seu portal para descobrir e baixar novos aplicativos com controle por movimento.

Figure 3.4 – Leap motion application installation

After successfully installing the executable, it is possible to verify the icon on the desktop. Connect the Leap Motion controller to the computer through a USB port. The Leap Motion should emit a green light on the front and as red lights on the top should be active. Open the Leap Motion Control Panel and check that the device status is green, as shown in Figure 3.5.



Painel de controle do Leap Motion
Geral
Solução de problemas
Sobre
Solução de problemas
Mostrar log de software
Visualizador de diagnóstico
Relatar problema do software
Restaurar configurações padrão
Recalibrar dispositivo
Get Help with Recalibration
Device Status
OK
Cancel
Apply

**Modo de poucos recursos**

Reduces performance and bandwidth of the Leap Motion Controller to improve reliability on slower computers.

✓ **Evite o Baixo Desempenho**

Parar o rastreamento automaticamente quando forem detectadas más condições.

**Recalibrate Device**

If the sensors on the Leap Motion controller are knocked out of their initial alignment,the device must be recalibrated. Poor calibration can cause:

- Persistent jumpiness
- Frequent discontinuities in the tracking data
- Aberrations in tracking data that occur only in certain areas of the field of view
- Poor tracking range

Service Status: CONNECTED
Device Status: CONNECTED
Calibration Status: GOOD
Tracking Status: STREAMING
Bandwidth Status: GOOD
Lighting Status: GOOD
Smudge Status: GOOD

Figure 3.5 – Leap Motion Control Panel

After this step, open the “**Executable – Computer Application**” folder. In the folder there is an executable (LeaPhysioGame) and a folder (LeaPhysioGame_Data), as illustrated in Figure 3.6.

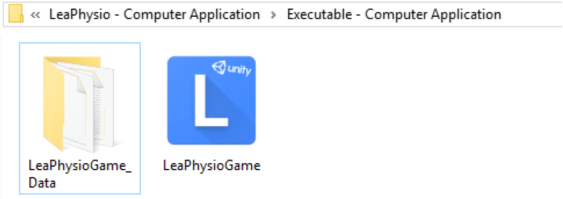

« LeaPhysio - Computer Application › Executable - Computer Application
LeaPhysioGame_ Data
LeaPhysioGame

Figure 3.6 – Contents of the “*Executable – Computer Application*” folder

Should copy both files to the desktop of computer. These files should stay always together for the application to work. After these steps the application is ready to use. To run the application, use the LeaPhysioGame, as illustrated in Figure 3.7.

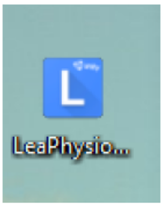


Figure 3.7 – Computer application icon

### 3.2. Mobile Application Installation

Open the “**LeaPhysio – Mobile Application**” folder (see Figure 3.2). In the folder there is an APK file (see Figure 3.8) that must be installed on a mobile device to use the LeaPhysio mobile application. Should download this file to your device.

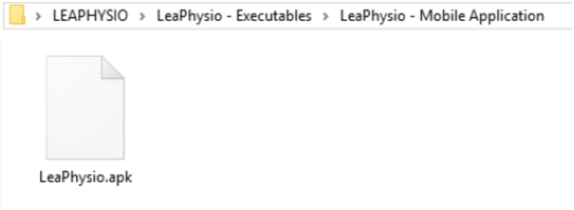

› LEAPHYSIO › LeaPhysio - Executables › LeaPhysio - Mobile Application
LeaPhysio.apk

Figure 3.8 – Contents of the “*LeaPhysio – Mobile Application*” folder

The application was developed to run on devices with version 4.0.4 (Ice Cream Sandwich) or higher, and its target is version 6.0 (Marshmallow). The mobile device where the application is to be installed should have the "Unknown Sources" option active:

- Go to "Settings" (see Figure 3.9).

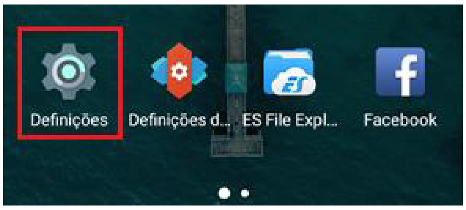


Figure 3.9 – Device settings

- Tap "Security" (see Figure 3.10).

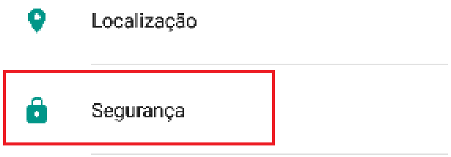
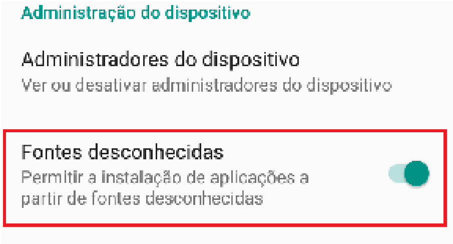


Figure 3.10 – Device Settings Security Option

- Check the "Unknown sources" box (see Figure 3.10).


Administração do dispositivo

Administradores do dispositivo
Ver ou desativar administradores do dispositivo

Fontes desconhecidas
Permitir a instalação de aplicações a partir de fontes desconhecidas

Figure 3.11 – Unknown Source option

To manually install the APK file:

- Tap "File manager" (see Figure 3.12).

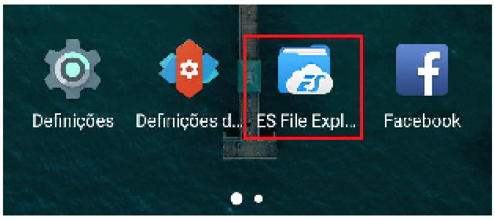


Figure 3.12 – Device File Manager

- Use the File manager to locate the APK application and tap the APK file to open it (see Figure 3.13).

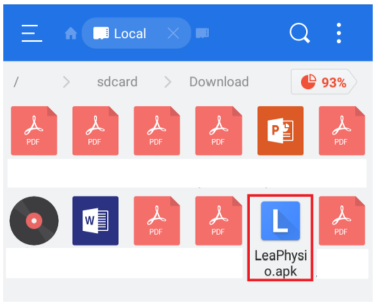
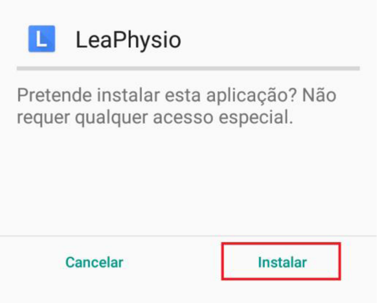


Figure 3.13 – LeaPhysio.apk

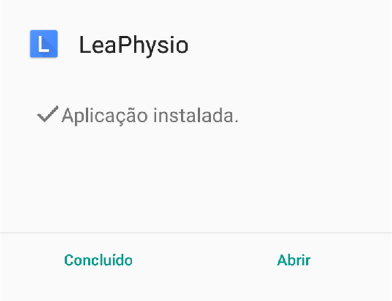
- Tap "Install" to complete the installation (see Figure 3.14).


LeaPhysio

Pretende instalar esta aplicação? Não requer qualquer acesso especial.

Figure 3.14 – Install LeaPhysio application

After installing the application is ready to be used by physiotherapists and clinic manager (see Figure 3.15).


LeaPhysio

✓Aplicação instalada.

Figure 3.15 – Installation Complete

# Appendix C – Technical Manual

Departamento de Ciências e Tecnologias de Informação

# Technical Manual

LeaPhysio – Games Enhanced Physical Rehabilitation

Filipe da Costa Pereira Lourenço

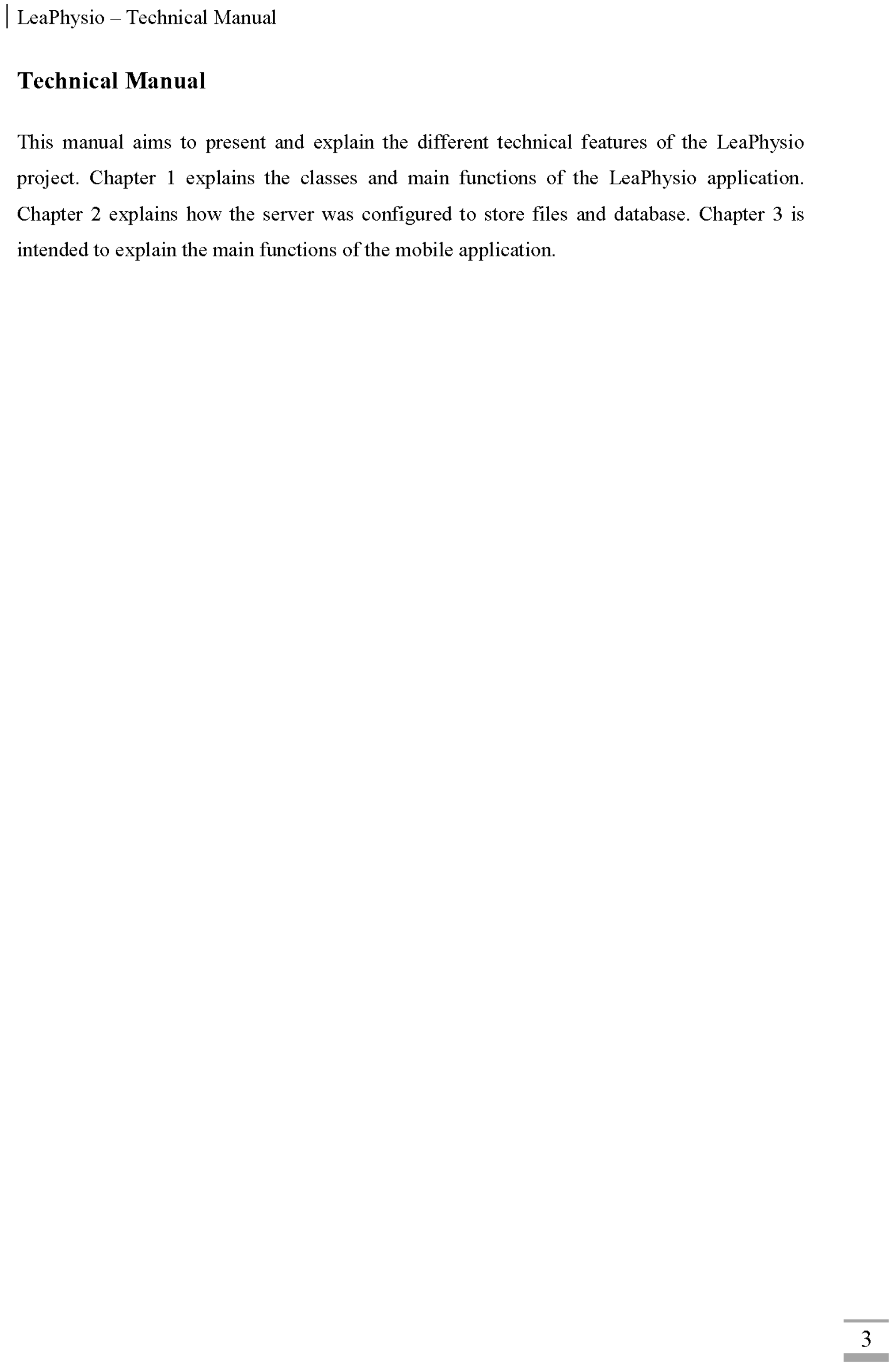
Advisor:
Doctor Octavian Postolache, Assistant Professor
ISCTE-IUL

**October 2017**

## Figures

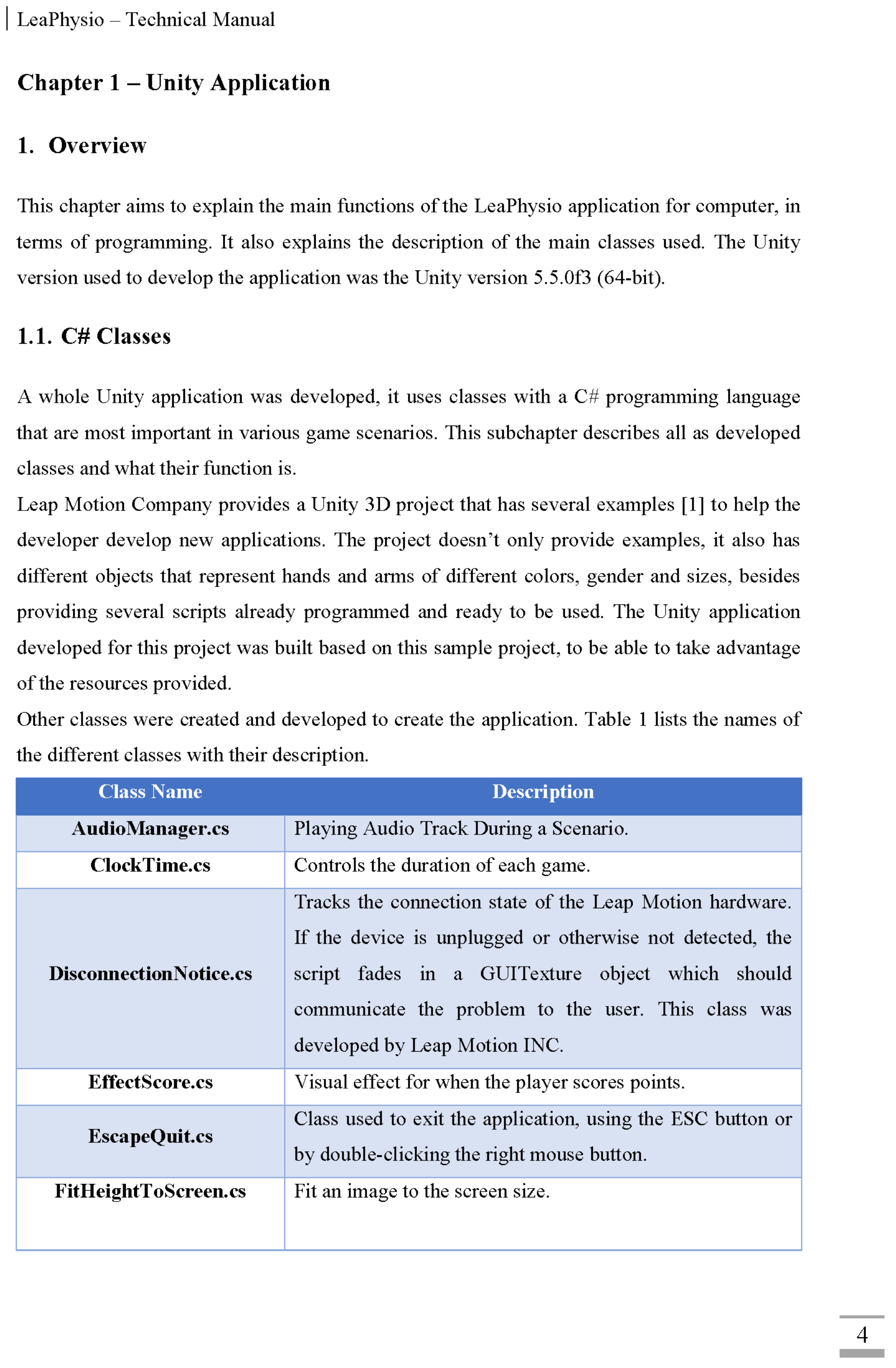
## Tables

## Contents



## Technical Manual

This manual aims to present and explain the different technical features of the LeaPhysio project. Chapter 1 explains the classes and main functions of the LeaPhysio application. Chapter 2 explains how the server was configured to store files and database. Chapter 3 is intended to explain the main functions of the mobile application.



## Chapter 1 – Unity Application

### 1. Overview

This chapter aims to explain the main functions of the LeaPhysio application for computer, in terms of programming. It also explains the description of the main classes used. The Unity version used to develop the application was the Unity version 5.5.0f3 (64-bit).

### 1.1. C# Classes

A whole Unity application was developed, it uses classes with a C# programming language that are most important in various game scenarios. This subchapter describes all as developed classes and what their function is.

Leap Motion Company provides a Unity 3D project that has several examples [1] to help the developer develop new applications. The project doesn't only provide examples, it also has different objects that represent hands and arms of different colors, gender and sizes, besides providing several scripts already programmed and ready to be used. The Unity application developed for this project was built based on this sample project, to be able to take advantage of the resources provided.

Other classes were created and developed to create the application. Table 1 lists the names of the different classes with their description.

| Class Name | Description |
|---|---|
| **AudioManager.cs** | Playing Audio Track During a Scenario. |
| **ClockTime.cs** | Controls the duration of each game. |
| **DisconnectionNotice.cs** | Tracks the connection state of the Leap Motion hardware. If the device is unplugged or otherwise not detected, the script fades in a GUITexture object which should communicate the problem to the user. This class was developed by Leap Motion INC. |
| **EffectScore.cs** | Visual effect for when the player scores points. |
| **EscapeQuit.cs** | Class used to exit the application, using the ESC button or by double-clicking the right mouse button. |
| **FitHeightToScreen.cs** | Fit an image to the screen size. |

| | |
|---|---|
| **GameProcess.cs** | Used for the management of the training plan scenario, this is what indicates the next training to be performed and with what settings. |
| **GrabbableObject.cs** | Allows an object to be grabbed. This class was developed by Leap Motion INC. |
| **GrabbingHand.cs** | Leap Motion hand script that detects pinches and grabs the closest rigidbody. This class was developed by Leap Motion INC. |
| **HandController.cs** | Main class to replicate hand movement in the game scenario. This class was developed by Leap Motion INC. |
| **HideMouse.cs** | Class used to hide the mouse cursor. |
| **Information_Loading.cs** | Used to verify if patient has active trainings. |
| **iTweenPath.cs** | Apply a motion to the game camera. |
| **MoveCameraEffect_CC.cs** | Apply the movement in the Camera using the iTween, at the beginning of the CollectCube game. |
| **MoveCameraEffect_PB.cs** | Apply the motion in the Camera using the iTween, at the beginning of the PickaBall game. |
| **MyHandSelection.cs** | Serves to choose which hand drawing will appear in the scenario. |
| **MyObjectDetector.cs** | Checks the collision of objects in the CollectCube game. |
| **PatientInformation.cs** | Singleton Class to store patient information after successfully completing Login to the system. Is used in different scenarios. |
| **PlayerController.cs** | This class represents the player for the PickaBall game |
| **ProcessControl.cs** | Used to manage all the decisions of the beginning of the game and at the end, whether there is a next game and save information of each result in the database. |
| **QRCODE_ManageAudio,cs** | It serves to activate the 'WrongLogin' & 'CorrectLogin' audio tracks and select random audio to play as background music. |
| **QRCODE_ScanEffect.cs** | Creates a visual effect of 'turn on' and 'off' at the corners of the Webcam reading screen, used to read the QRCode. |
| **QRCODE_Webcam.cs** | Class used to manage, process and process all information |

| | |
|---|---|
| | coming from patient's QRCode. |
| **RestartApplication.cs** | Serves to restart the LeaPhysio application. Used for when there is no training or end of training performed by the patient. |
| **ResultLastGame.cs** | Shows the patient's score in the last game. |
| **RigidHand.cs** | Represents the structure of the hand in the game scenario. class was developed by Leap Motion INC. |
| **Rotator.cs** | Applies a rotation transform to the object. |
| **SceneFadeInOut.cs** | Used to create a 'FadeToBlack' or 'FadetoWhite' effect on an object or texture. |
| **SpawnPickUps.cs** | Used in the PickaBall game, it indicates what type of SHAPE should appear on the board depending on the difficulty of the game. |
| **SpawnPosition.cs** | Spawn of cubes and boxes in CollectCube game. |
| **Thematic.cs** | Indicates what type of game theme is being used. |
| **TimetoStartGame.cs** | Used at the start of the game, while the game time doesn't start there is a pre-set time for the player to visualize the hands. |

Table 1 - Names of classes used in the Unity application and their description

Many of the classes described above are present in different application scenarios. In the Unity application each scenario is represented by the name of Scene. The following tables indicate which classes are used in each Scene.

- **INTRO-SplashScreen_1 Scene**

Figure 1.1 – INTRO-SplashScreen_1 Scene

| Classes Used | | |
|---|---|---|
| HideMouse.cs | EscapeQuit.cs | SceneFadeInOut.cs |

Table 2 – Classes used in INTRO-SplashScreen_1 Scene

- **INTRO-SplashScreen_2 Scene**

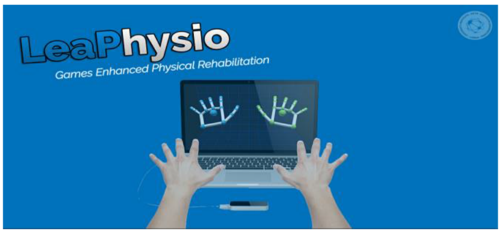


Figure 1.2 – INTRO-SplashScreen_2 Scene

| Classes Used | | |
|---|---|---|
| HideMouse.cs | EscapeQuit.cs | SceneFadeInOut.cs |

Table 3 – Classes used in INTRO-SplashScreen_2 Scene

- **LoginQRCODE Scene**

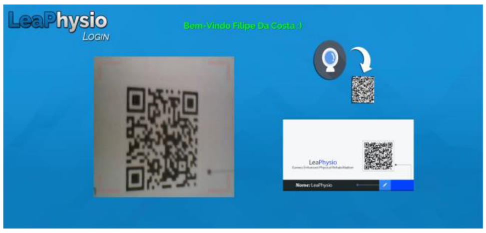
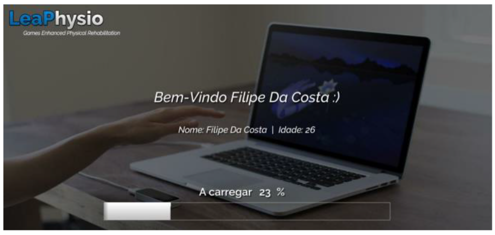


Figure 1.3 – LoginQRCODE Scene

| Classes Used | | |
|---|---|---|
| QRCODE ManageAudio.cs | QRCODE Webcam.cs | QRCODE ScanEffect.cs |
| PatientInformation.cs | HideMouse.cs | EscapeQuit.cs |

Table 4 – Classes used in LoginQRCODE Scene

- **INTRO-WELCOME Scene**



Bem-Vindo Filipe Da Costa :)

Nome: Filipe Da Costa | Idade: 26

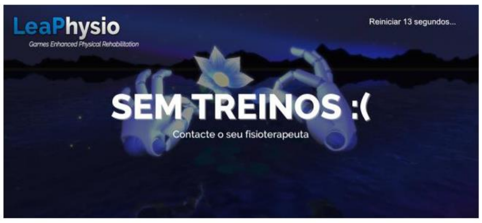
A carregar 23 %

Figure 1.4 – INTRO-WELCOME Scene

| Classes Used | | |
|---|---|---|
| HideMouse.cs | EscapeQuit.cs | Information_Loading.cs |
| PatientInformation.cs | | |

Table 5 – Classes used in INTRO-WELCOME Scene

- **Trainer-Empty Scene**



Reiniciar 13 segundos...

SEM TREINOS :(

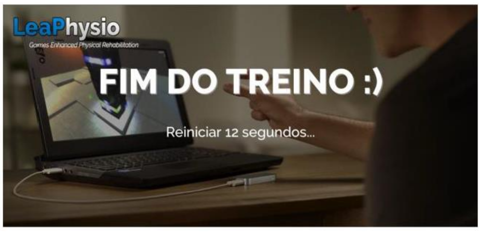
Contacte o seu fisioterapeuta

Figure 1.5 – Trainer-Empty Scene

| Classes Used | | |
|---|---|---|
| HideMouse.cs | EscapeQuit.cs | RestartApplication.cs |
| AudioManager.cs | | |

Table 6 – Classes used in Trainer-Empty Scene

- **Trainer-Finish Scene**



FIM DO TREINO :)

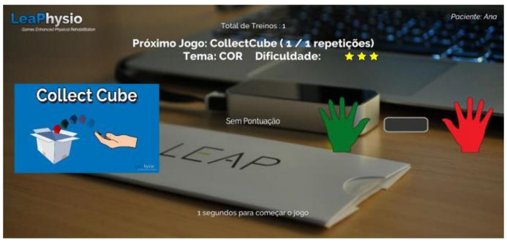
Reiniciar 12 segundos...

Figure 1.6 – Trainer-Finish Scene

| Classes Used | | |
|---|---|---|
| HideMouse.cs | EscapeQuit.cs | RestartApplication.cs |
| AudioManager.cs | PatientInformation.cs | |

Table 7 – Classes used in Trainer-Finish Scene

- **Trainer-Planning Scene**



Paciente: Ana

Total de Treinos : 1

Próximo Jogo: CollectCube ( 1 / 1 repetições)
Tema: COR Dificuldade: ★★★

Sem Pontuação

1 segundos para começar o jogo

Figure 1.7 – Trainer-Planning Scene

| Classes Used | | |
|---|---|---|
| HideMouse.cs | EscapeQuit.cs | GameProcess.cs |
| AudioManager.cs | PatientInformation.cs | |

Table 8 – Classes used in Trainer-Planning Scene

- **Trainer-ScoreResult Scene**

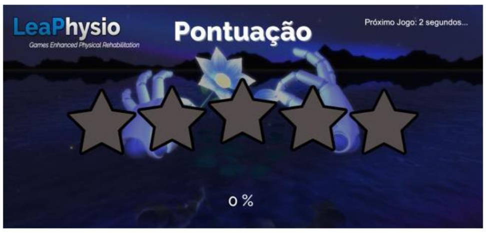

LeaPhysio
Games Enhanced Physical Rehabilitation
Pontuação
Próximo Jogo: 2 segundos...
0 %

Figure 1.8 – Trainer-ScoreResult Scene

| Classes Used | | |
|---|---|---|
| HideMouse.cs | EscapeQuit.cs | ResultLastGame.cs |
| AudioManager.cs | PatientInformation.cs | |

Table 9 – Classes used in Trainer-ScoreResult Scene

- **CollectCube_M & CollectCube_F Scene**

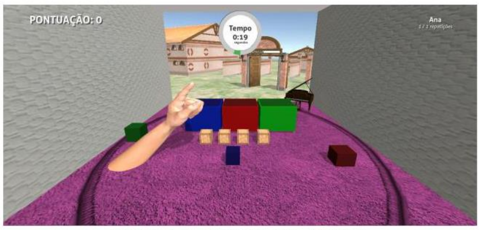


Figure 1.9 – CollectCube_F Scene

| Classes Used | | |
|---|---|---|
| MoveCameraEffect_CC.cs | iTweenPath.cs | HideMouse.cs |
| EscapeQuit.cs | Thematic.cs | ClockTime.cs |
| PatientInformation.cs | AudioManager.cs | ProcessControl.cs |
| TimetoStartGame.cs | EffectScore.cs | MyHandSelection.cs |
| HandController.cs | SpawnPosition.cs | MyObjectDetector.cs |
| GrabbableObject.cs | RigidHand.cs | GrabbingHand.cs |
| DisconnectionNotice.cs | FitHeightToScreen.cs | |

Table 10 – Classes used in CollectCube_M & CollectCube_F Scene

- **PickaBall_M & PickaBall_F Scene**

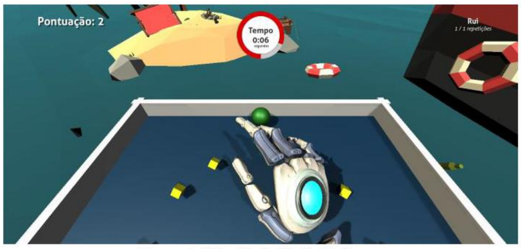


Figure 1.10 – PickaBall_M Scene

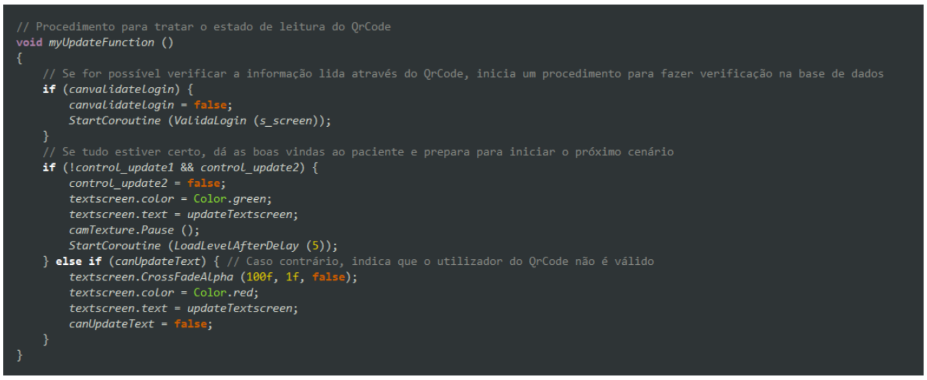
| Classes Used | | |
|---|---|---|
| MoveCameraEffect_PB.cs | iTweenPath.cs | HideMouse.cs |
| EscapeQuit.cs | Thematic.cs | ClockTime.cs |
| PatientInformation.cs | AudioManager.cs | ProcessControl.cs |
| TimetoStartGame.cs | ScoreResult.cs | HandController.cs |
| MyHandSelection.cs | SpawnPickUps.cs | Rotator.cs |
| RigidHand.cs | GrabbingHand.cs | DisconnectionNotice.cs |
| FitHeightToScreen.cs | | |

Table 11 – Classes used in PickaBall_M & PickaBall_F Scene

## 1.2. Programming

This subchapter explains some of the main functions used by the application. In each class the code is commented on for a better understanding of the lines of code.

The *myUpdateFunction()* function is in the QRCODE_Webcam class. Checks the status for reading the QRCode when the patient tries to enter the application. The following code illustrates this procedure (see Figure 1.11).

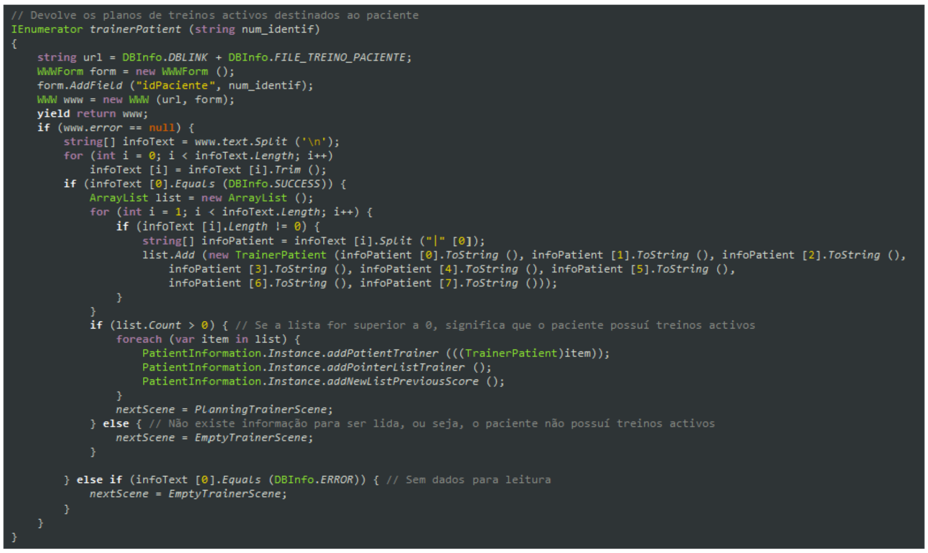


```
// Procedimento para tratar o estado de leitura do QrCode
void myUpdateFunction ()
{
    // Se for possível verificar a informação lida através do QrCode, inicia um procedimento para fazer verificação na base de dados
    if (canvalidatelogin) {
        canvalidatelogin = false;
        StartCoroutine (ValidaLogin (s_screen));
    }
    // Se tudo estiver certo, dá as boas vindas ao paciente e prepara para iniciar o próximo cenário
    if (!control_update1 && control_update2) {
        control_update2 = false;
        textscreen.color = Color.green;
        textscreen.text = updateTextscreen;
        camTexture.Pause ();
        StartCoroutine (LoadLevelAfterDelay (5));
    } else if (canUpdateText) { // Caso contrário, indica que o utilizador do QrCode não é válido
        textscreen.CrossFadeAlpha (100f, 1f, false);
        textscreen.color = Color.red;
        textscreen.text = updateTextscreen;
        canUpdateText = false;
    }
}
```

Figure 1.11 – *myUpdateFunction()* function

The first *if* checks whether it is possible to read the information through the QRCode, if so, then initiates a *Coroutine* to verify that the QRCode content is valid. The second *if* tries to verify that the result from the QRCode reading is valid, if everything is right, then welcomes the patient and prepares to start the next scenario, otherwise indicates that the user who is trying to enter the application does not have a valid QRCode. The application uses the *Zxing.dll library* to decode the information contained in the QRCode.

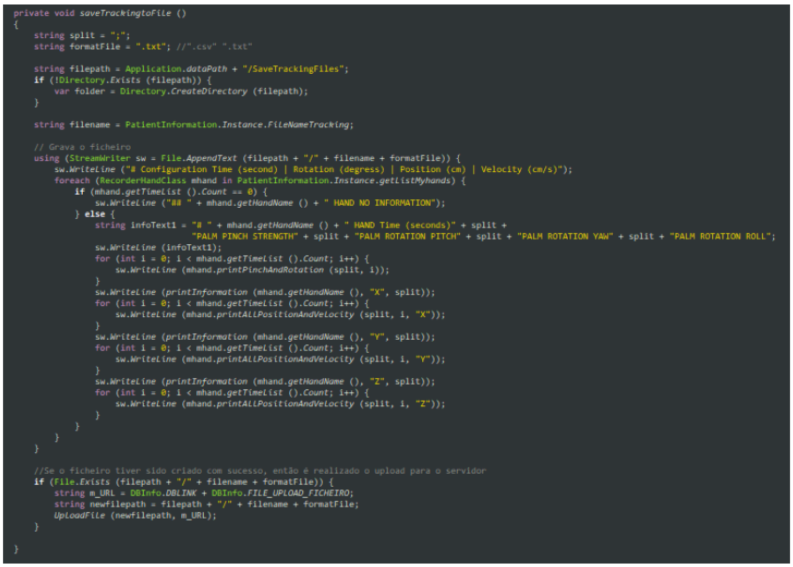
The *IEnumerator trainerPatient()* is in the Information_Loading class and tries to execute the corresponding PHP script so that it returns the active training plans for the patient who entered the application. The following code illustrates this procedure (see Figure 1.12).



```
// Devolve os planos de treinos activos destinados ao paciente
IEnumerator trainerPatient (string num_identif)
{
    string url = DBInfo.DBLINK + DBInfo.FILE_TREINO_PACIENTE;
    WWWForm form = new WWWForm ();
    form.AddField ("idPaciente", num_identif);
    WWW www = new WWW (url, form);
    yield return www;
    if (www.error == null) {
        string[] infoText = www.text.Split ('\n');
        for (int i = 0; i < infoText.Length; i++)
            infoText [i] = infoText [i].Trim ();
        if (infoText [0].Equals (DBInfo.SUCCESS)) {
            ArrayList list = new ArrayList ();
            for (int i = 1; i < infoText.Length; i++) {
                if (infoText [i].Length != 0) {
                    string[] infoPatient = infoText [i].Split ("|" [0]);
                    list.Add (new TrainerPatient (infoPatient [0].ToString (), infoPatient [1].ToString (), infoPatient [2].ToString (),
                        infoPatient [3].ToString (), infoPatient [4].ToString (), infoPatient [5].ToString (),
                        infoPatient [6].ToString (), infoPatient [7].ToString ()));
                }
            }
            if (list.Count > 0) { // Se a lista for superior a 0, significa que o paciente possuí treinos activos
                foreach (var item in list) {
                    PatientInformation.Instance.addPatientTrainer (((TrainerPatient)item));
                    PatientInformation.Instance.addPointerListTrainer ();
                    PatientInformation.Instance.addNewListPreviousScore ();
                }
                nextScene = PlanningTrainerScene;
            } else { // Não existe informação para ser lida, ou seja, o paciente não possuí treinos activos
                nextScene = EmptyTrainerScene;
            }

        } else if (infoText [0].Equals (DBInfo.ERROR)) { // Sem dados para leitura
            nextScene = EmptyTrainerScene;
        }
    }
}
```

Figure 1.12 – *trainerPatient()* function

At first the procedure creates the necessary fields to execute the PHP script and waits for a response by the result. If the application gets an affirmative answer, either because the connection was successful and exists results, then read the information obtained, create training plans and associate to PatientInformation, which works as a Singleton class, so that it is possible to save information whenever the game scenario changes.

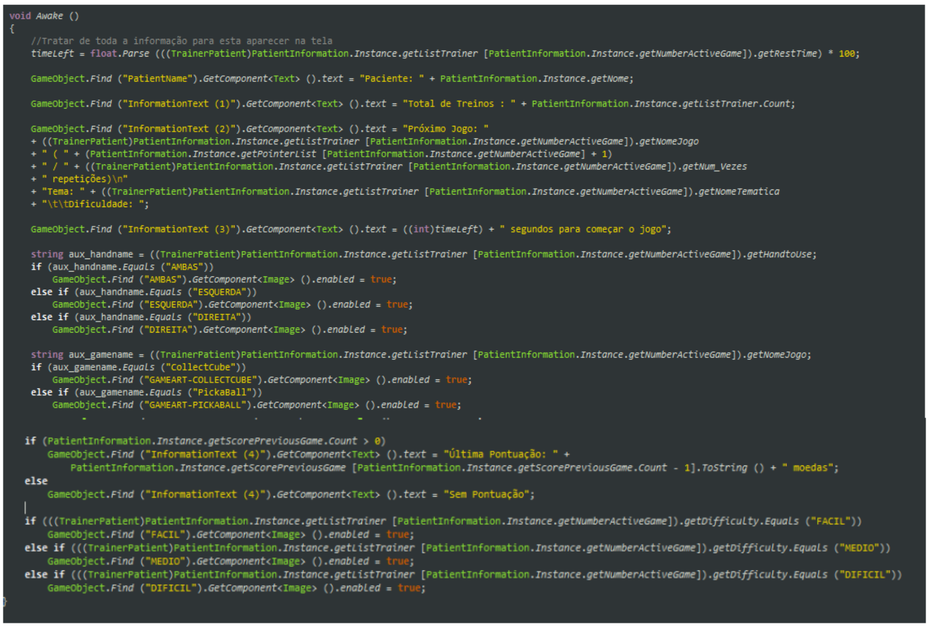


```
private void saveTrackingtoFile ()
{
    string split = ";";
    string formatFile = ".txt"; //".csv" ".txt"

    string filepath = Application.dataPath + "/SaveTrackingFiles";
    if (!Directory.Exists (filepath)) {
        var folder = Directory.CreateDirectory (filepath);
    }

    string filename = PatientInformation.Instance.FileNameTracking;

    // Grava o ficheiro
    using (StreamWriter sw = File.AppendText (filepath + "/" + filename + formatFile)) {
        sw.WriteLine ("# Configuration Time (second) | Rotation (degress) | Position (cm) | Velocity (cm/s)");
        foreach (RecorderHandClass mhand in PatientInformation.Instance.getListMyhands) {
            if (mhand.getTimeList ().Count == 0) {
                sw.WriteLine ("## " + mhand.getHandName () + " HAND NO INFORMATION");
            } else {
                string infoText1 = "# " + mhand.getHandName () + " HAND Time (seconds)" + split +
                                   "PALM PINCH STRENGTH" + split + "PALM ROTATION PITCH" + split + "PALM ROTATION YAW" + split + "PALM ROTATION ROLL";
                sw.WriteLine (infoText1);
                for (int i = 0; i < mhand.getTimeList ().Count; i++) {
                    sw.WriteLine (mhand.printPinchAndRotation (split, i));
                }
                sw.WriteLine (printInformation (mhand.getHandName (), "X", split));
                for (int i = 0; i < mhand.getTimeList ().Count; i++) {
                    sw.WriteLine (mhand.printALLPositionAndVelocity (split, i, "X"));
                }
                sw.WriteLine (printInformation (mhand.getHandName (), "Y", split));
                for (int i = 0; i < mhand.getTimeList ().Count; i++) {
                    sw.WriteLine (mhand.printALLPositionAndVelocity (split, i, "Y"));
                }
                sw.WriteLine (printInformation (mhand.getHandName (), "Z", split));
                for (int i = 0; i < mhand.getTimeList ().Count; i++) {
                    sw.WriteLine (mhand.printALLPositionAndVelocity (split, i, "Z"));
                }
            }
        }
    }

    //Se o ficheiro tiver sido criado com sucesso, então é realizado o upload para o servidor
    if (File.Exists (filepath + "/" + filename + formatFile)) {
        string m_URL = DBInfo.DBLINK + DBInfo.FILE_UPLOAD_FICHEIRO;
        string newfilepath = filepath + "/" + filename + formatFile;
        UploadFile (newfilepath, m_URL);
    }

}
```

Figure 1.13 – *saveTrackingtoFile()* function

The *saveTrackingtoFile()* function is in the ResultLastGame class. Creates a data structure to support the coordinates of hands taken in the game that the patient performed. This structure is saved to a file, which is sent to the server. See Figure 1.13.

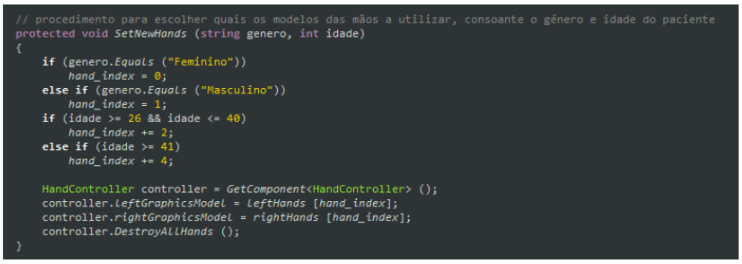
The Awake () procedure is to handle information before the scenario is created. The following code illustrates *Awake()* from the GameProcess class (see Figure 1.14) and is used to retrieve collected information about the patient and their training and to populate the Trainer-Planning scenario with data.



```
void Awake ()
{
    //Tratar de toda a informação para esta aparecer na tela
    timeLeft = float.Parse (((TrainerPatient)PatientInformation.Instance.getListTrainer [PatientInformation.Instance.getNumberActiveGame]).getRestTime) * 100;

    GameObject.Find ("PatientName").GetComponent<Text> ().text = "Paciente: " + PatientInformation.Instance.getNome;

    GameObject.Find ("InformationText (1)").GetComponent<Text> ().text = "Total de Treinos : " + PatientInformation.Instance.getListTrainer.Count;

    GameObject.Find ("InformationText (2)").GetComponent<Text> ().text = "Próximo Jogo: "
    + ((TrainerPatient)PatientInformation.Instance.getListTrainer [PatientInformation.Instance.getNumberActiveGame]).getNomeJogo
    + " ( " + (PatientInformation.Instance.getPointerList [PatientInformation.Instance.getNumberActiveGame] + 1)
    + " / " + ((TrainerPatient)PatientInformation.Instance.getListTrainer [PatientInformation.Instance.getNumberActiveGame]).getNum_Vezes
    + " repetições)\n"
    + "Tema: " + ((TrainerPatient)PatientInformation.Instance.getListTrainer [PatientInformation.Instance.getNumberActiveGame]).getNomeTematica
    + "\t\tDificuldade: ";

    GameObject.Find ("InformationText (3)").GetComponent<Text> ().text = ((int)timeLeft) + " segundos para começar o jogo";

    string aux_handname = ((TrainerPatient)PatientInformation.Instance.getListTrainer [PatientInformation.Instance.getNumberActiveGame]).getHandtoUse;
    if (aux_handname.Equals ("AMBAS"))
        GameObject.Find ("AMBAS").GetComponent<Image> ().enabled = true;
    else if (aux_handname.Equals ("ESQUERDA"))
        GameObject.Find ("ESQUERDA").GetComponent<Image> ().enabled = true;
    else if (aux_handname.Equals ("DIREITA"))
        GameObject.Find ("DIREITA").GetComponent<Image> ().enabled = true;

    string aux_gamename = ((TrainerPatient)PatientInformation.Instance.getListTrainer [PatientInformation.Instance.getNumberActiveGame]).getNomeJogo;
    if (aux_gamename.Equals ("CollectCube"))
        GameObject.Find ("GAMEART-COLLECTCUBE").GetComponent<Image> ().enabled = true;
    else if (aux_gamename.Equals ("PickaBall"))
        GameObject.Find ("GAMEART-PICKABALL").GetComponent<Image> ().enabled = true;

   if (PatientInformation.Instance.getScorePreviousGame.Count > 0)
       GameObject.Find ("InformationText (4)").GetComponent<Text> ().text = "Última Pontuação: " +
           PatientInformation.Instance.getScorePreviousGame [PatientInformation.Instance.getScorePreviousGame.Count - 1].ToString () + " moedas";
   else
       GameObject.Find ("InformationText (4)").GetComponent<Text> ().text = "Sem Pontuação";
   |
   if (((TrainerPatient)PatientInformation.Instance.getListTrainer [PatientInformation.Instance.getNumberActiveGame]).getDifficulty.Equals ("FACIL"))
       GameObject.Find ("FACIL").GetComponent<Image> ().enabled = true;
   else if (((TrainerPatient)PatientInformation.Instance.getListTrainer [PatientInformation.Instance.getNumberActiveGame]).getDifficulty.Equals ("MEDIO"))
       GameObject.Find ("MEDIO").GetComponent<Image> ().enabled = true;
   else if (((TrainerPatient)PatientInformation.Instance.getListTrainer [PatientInformation.Instance.getNumberActiveGame]).getDifficulty.Equals ("DIFICIL"))
       GameObject.Find ("DIFICIL").GetComponent<Image> ().enabled = true;
}
```

Figure 1.14 – *Awake()* function of GameProcess class

The setNewHands () function is in the MyHandSelection class and used during each game. It serves to choose which models of hands to use for graphic representation, depending on the gender and age of the patient. The following code illustrates this procedure (see Figure 1.15).

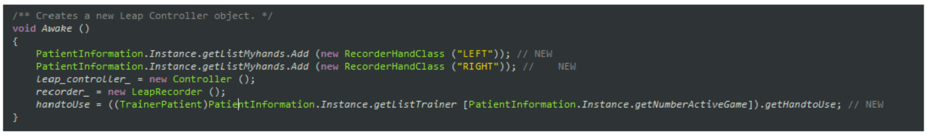


```
// procedimento para escolher quais os modelos das mãos a utilizar, consoante o género e idade do paciente
protected void SetNewHands (string genero, int idade)
{
    if (genero.Equals ("Feminino"))
        hand_index = 0;
    else if (genero.Equals ("Masculino"))
        hand_index = 1;
    if (idade >= 26 && idade <= 40)
        hand_index += 2;
    else if (idade >= 41)
        hand_index += 4;

    HandController controller = GetComponent<HandController> ();
    controller.leftGraphicsModel = leftHands [hand_index];
    controller.rightGraphicsModel = rightHands [hand_index];
    controller.DestroyAllHands ();
}
```

Figure 1.15 – *setNewHands()* function

Through gender and age an index is taken that serves as a position for the array of existing hand models. After this step the HandController component is used to change the graphic models of the patient's hands.

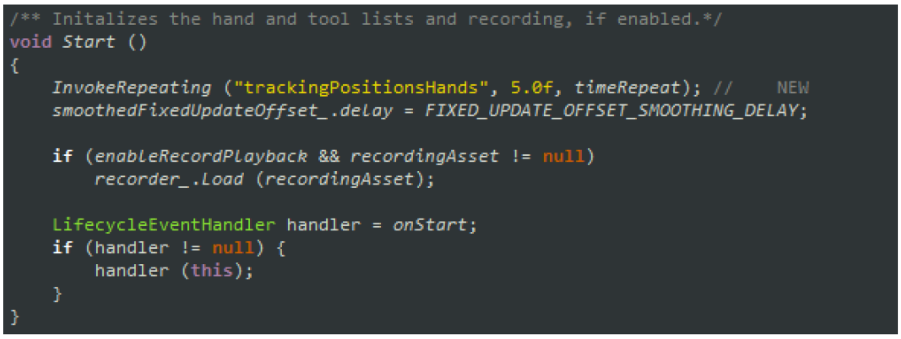
The HandController class is a class created by *Leap Motion INC* but changes were made to make it adaptable to games. When the HandController class is in games, it is necessary in the *Awake()* procedure to create objects of type RecorderHandClass to store the coordinates of the hands. These objects are stored in the instance of the PatientInformation class (see Figure 1.16).



```
/** Creates a new Leap Controller object. */
void Awake ()
{
    PatientInformation.Instance.getListMyhands.Add (new RecorderHandClass ("LEFT")); // NEW
    PatientInformation.Instance.getListMyhands.Add (new RecorderHandClass ("RIGHT")); //    NEW
    leap_controller_ = new Controller ();
    recorder_ = new LeapRecorder ();
    handtoUse = ((TrainerPatient)PatientInformation.Instance.getListTrainer [PatientInformation.Instance.getNumberActiveGame]).getHandtoUse; // NEW
}
```

Figure 1.16 – *Awake()* function of HandController class

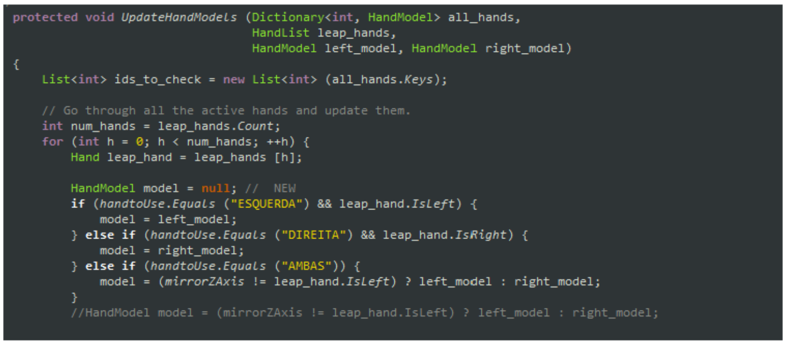
When the *Start()* method (see Figure 1.17) is invoked a procedure is initialized through the *InvokeRepeating* method. This method causes a procedure to be called X at X time (timeRepeat = 3 seconds) and what time it will initialize (5 seconds). This boot time was chosen to match the wait time before starting the game. The function that runs from X in X time is the *trackingPositionsHands* and serves to collect coordinates referring to the hands.



```
/** Initalizes the hand and tool lists and recording, if enabled.*/
void Start ()
{
    InvokeRepeating ("trackingPositionsHands", 5.0f, timeRepeat); //    NEW
    smoothedFixedUpdateOffset_.delay = FIXED_UPDATE_OFFSET_SMOOTHING_DELAY;

    if (enableRecordPlayback && recordingAsset != null)
        recorder_.Load (recordingAsset);

    LifecycleEventHandler handler = onStart;
    if (handler != null) {
        handler (this);
    }
}
```

Figure 1.17 – *Start()* method of HandController class

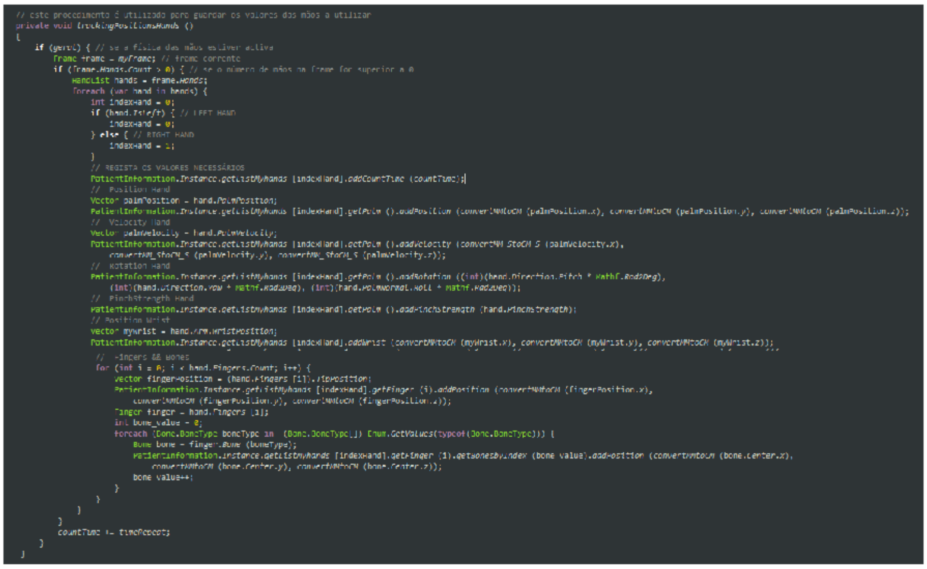
In the *UpdateHandModels()* function (see Figure 1.18) it was necessary to make changes so that the game doesn't always have the representation of both hands, but only the hands indicated when the physiotherapist created the training plan.



```
protected void UpdateHandModels (Dictionary<int, HandModel> all_hands,
                                 HandList leap_hands,
                                 HandModel left_model, HandModel right_model)
{
    List<int> ids_to_check = new List<int> (all_hands.Keys);

    // Go through all the active hands and update them.
    int num_hands = leap_hands.Count;
    for (int h = 0; h < num_hands; ++h) {
        Hand leap_hand = leap_hands [h];

        HandModel model = null; //  NEW
        if (handtoUse.Equals ("ESQUERDA") && leap_hand.IsLeft) {
            model = left_model;
        } else if (handtoUse.Equals ("DIREITA") && leap_hand.IsRight) {
            model = right_model;
        } else if (handtoUse.Equals ("AMBAS")) {
            model = (mirrorZAxis != leap_hand.IsLeft) ? left_model : right_model;
        }
        //HandModel model = (mirrorZAxis != leap_hand.IsLeft) ? left_model : right_model;
```

Figure 1.18 – *UpdateHandModels()* function

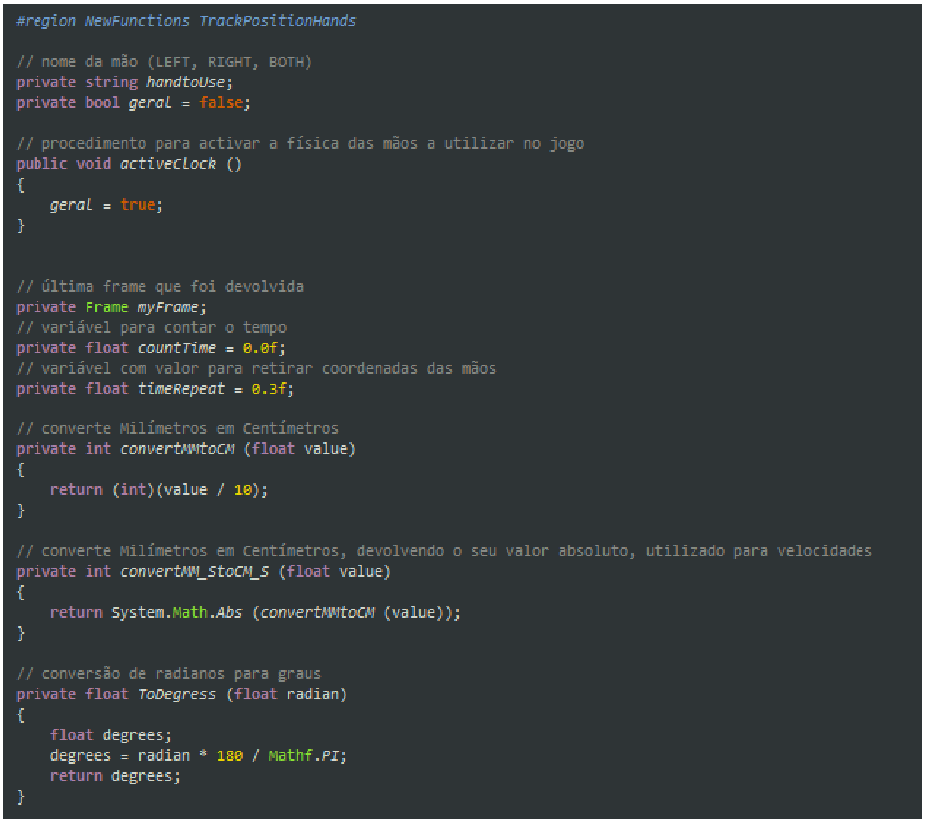
The *trackingPositionsHands()* function (see Figure 1.19) is used to store the values of the hands to use. Initially check if I have any active hands, if yes, will use the current frame, check which hands have graphical representation and record the different coordinates, for example, position in X, Y and Z for wrist or Yaw, Pitch Raw to the palm of the hand.



```
// este procedimento é utilizado para guardar os valores das mãos a utilizar
private void trackingPositionsHands ()
{
    if (gerul) { // se a física das mãos estiver activa
        Frame frame = myFrame; // frame corrente
        if (frame.Hands.Count > 0) { // se o número de mãos na frame for superior a 0
            HandList hands = frame.Hands;
            foreach (var hand in hands) {
                int indexHand = 0;
                if (hand.IsLeft) { // LEFT HAND
                    indexHand = 0;
                } else { // RIGHT HAND
                    indexHand = 1;
                }
                // REGISTA OS VALORES NECESSÁRIOS
                PatientInformation.Instance.getListMyhands [indexHand].addCountTime (countTime);
                //  Position Hand
                Vector palmPosition = hand.PalmPosition;
                PatientInformation.Instance.getListMyhands [indexHand].getPalm ().addPosition (convertMMtoCM (palmPosition.x), convertMMtoCM (palmPosition.y), convertMMtoCM (palmPosition.z));
                //  Velocity Hand
                Vector palmVelocity = hand.PalmVelocity;
                PatientInformation.Instance.getListMyhands [indexHand].getPalm ().addVelocity (convertMM_StoCM_S (palmVelocity.x),
                    convertMM_StoCM_S (palmVelocity.y), convertMM_StoCM_S (palmVelocity.z));
                //  Rotation Hand
                PatientInformation.Instance.getListMyhands [indexHand].getPalm ().addRotation ((int)(hand.Direction.Pitch * Mathf.Rad2Deg),
                    (int)(hand.Direction.Yaw * Mathf.Rad2Deg), (int)(hand.PalmNormal.Roll * Mathf.Rad2Deg));
                //  PinchStrength Hand
                PatientInformation.Instance.getListMyhands [indexHand].getPalm ().addPinchStrength (hand.PinchStrength);
                // Position Wrist
                Vector myWrist = hand.Arm.WristPosition;
                PatientInformation.Instance.getListMyhands [indexHand].addWrist (convertMMtoCM (myWrist.x), convertMMtoCM (myWrist.y), convertMMtoCM (myWrist.z));
                 //  Fingers && Bones
                 for (int i = 0; i < hand.Fingers.Count; i++) {
                    Vector fingerPosition = (hand.Fingers [i]).TipPosition;
                    PatientInformation.Instance.getListMyhands [indexHand].getFinger (i).addPosition (convertMMtoCM (fingerPosition.x),
                        convertMMtoCM (fingerPosition.y), convertMMtoCM (fingerPosition.z));
                    Finger finger = hand.Fingers [i];
                    int bone_value = 0;
                    foreach (Bone.BoneType boneType in  (Bone.BoneType[]) Enum.GetValues(typeof(Bone.BoneType))) {
                        Bone bone = finger.Bone (boneType);
                        PatientInformation.Instance.getListMyhands [indexHand].getFinger (i).getBonesByIndex (bone_value).addPosition (convertMMtoCM (bone.Center.x),
                            convertMMtoCM (bone.Center.y), convertMMtoCM (bone.Center.z));
                        bone_value++;
                    }
                 }
            }
        }
        countTime += timeRepeat;
    }
}
```

Figure 1.19 – *trackingPositionHands()* function

The following code illustrates the variables and auxiliary methods that were created and used in the procedures previously created (see Figure 1.20).



```
#region NewFunctions TrackPositionHands

// nome da mão (LEFT, RIGHT, BOTH)
private string handtoUse;
private bool geral = false;

// procedimento para activar a física das mãos a utilizar no jogo
public void activeClock ()
{
    geral = true;
}


// última frame que foi devolvida
private Frame myFrame;
// variável para contar o tempo
private float countTime = 0.0f;
// variável com valor para retirar coordenadas das mãos
private float timeRepeat = 0.3f;

// converte Milímetros em Centímetros
private int convertMMtoCM (float value)
{
    return (int)(value / 10);
}

// converte Milímetros em Centímetros, devolvendo o seu valor absoluto, utilizado para velocidades
private int convertMM_StoCM_S (float value)
{
    return System.Math.Abs (convertMMtoCM (value));
}

// conversão de radianos para graus
private float ToDegress (float radian)
{
    float degrees;
    degrees = radian * 180 / Mathf.PI;
    return degrees;
}
```

Figure 1.20 - Variables and auxiliary methods of the HandController class

## Chapter 2 – Server Side

### 2. Overview

This chapter explains all the settings made to allow HTTP communications between different applications (Unity and Android) through PHP scripts. Explains the settings made to host files and MySQL database.

The Server is responsible for storing all the information regarding clinic patients, training and results, and the main objective is that all information is centralized in one place and can be accessed remotely by the system applications. By storing this data, it is possible to analyze and see the information in the future, either with the mobile application destined for physiotherapists and the manager, but also for Unity application for the patients, so the application can collect information and configure the games destined for the patient.

To enable the exchange of information between the LeaPhysio applications and the server, a Web Services called Representational State Transfer (REST) is used. Web Servers are computers responsible for accepting HTTP requests from clients and serving them with HTTP responses. One of the most popular and used in the world, is the Apache server (free software).

For this purpose, an Architecture system, called LAMP (see Figure 2.1) was used. It is a combination of free and open source software and is one of the most common configurations for webservers. The term LAMP comes from the combination of the following technologies:

- **L**inux: Operation System;
- **A**pache: Web Server (HTTP) software;
- **M**ySQL: Database component;
- **P**HP: Programming language.

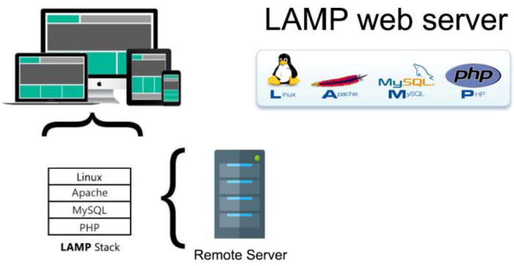


Figure 2.1 – LAMP Architecture System

All components of this architecture are freeware, minus the server that was provided by the dissertation advisor. The configuration needed to implement the architecture was done through the command line, using a SSH (Secure Shell) connection.

## 2.1. Configurations

The dissertation supervisor made available a Virtual Private Server (VPS) consisting of a virtual machine. This server has its dedicated operating system where your user has superuser access, allowing you to install any software that is compatible with the system. The operating system is Linux, where its distribution is Debian 3.2.0-4-amd64.

To have a server with LAMP architecture implemented it is necessary to perform a set of configurations. Configure the server as a WebServer to ensure that it can perform HTTP communications. Executes the following command:

```
sudo apt-get install apache2
```

The root directory of Apache is in:

```
/var/www/
```

Install MySQL to support the database in the future. MySQL is an open-source platform that allows relational database management, which uses the SQL language. MySQL is a popular tool and a central component for the smooth operation of the LeaPhysio system. Executes the following command:

```
sudo apt-get install mysql-server
```

The following commands are used to install version 5 of PHP and the MySQL libraries so that PHP has access to the MySQL database:

```
sudo apt-get install php5
sudo apt-get install php5-mysql
```

For a better management of the system database was installed phpMyAdmin, in which it is necessary to use the following command:

```
sudo apt-get install phpmyadmin
```

After the installation is complete it is necessary to move phpMyAdmin to the Apache root directory so that it can be accessed by the browser. Executes the following command:

```
cp -R /usr/share/phpmyadmin /var/www/phpmyadmin
```

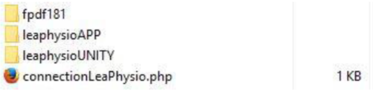
It is already possible to access through the browser in https://[IP]/phpmyadmin/.

## 2.2. Organization of folders and files on the server

After all the settings have been made, the following contents have been developed:

- Creation of the database, to store the necessary information;
- PHP scripts to return content to clients;
- Organization of the folders on the server.

The first two topics will be developed in the following sub-chapters. Regarding the organization of the folders on the server, in Figure 2.2 presents the files organization:

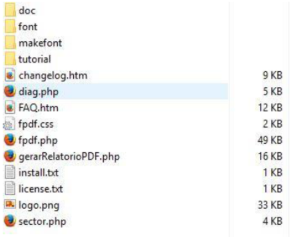


| | |
|---|---|
| fpdf181 | |
| leaphysioAPP | |
| leaphysioUNITY | |
| connectionLeaPhysio.php | 1 KB |

Figure 2.2 – Main folder organization on the server

- ***fpdf181***: this folder contains the scripts needed to generate clinical reports about patients. The *FDPF Library PDF generator* was used. The Figure 2.3 demonstrates the contents of this folder

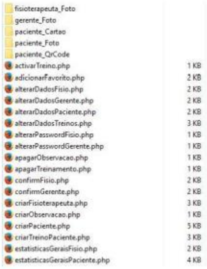
*"FPDF is a PHP class which allows to generate PDF files with pure PHP, that is to say without using the PDFlib library. F from FPDF stands for Free: you may use it for any kind of usage and modify it to suit your needs."*



| | |
|---|---|
| doc | |
| font | |
| makefont | |
| tutorial | |
| changelog.htm | 9 KB |
| diag.php | 5 KB |
| FAQ.htm | 12 KB |
| fpdf.css | 2 KB |
| fpdf.php | 49 KB |
| gerarRelatorioPDF.php | 16 KB |
| install.txt | 1 KB |
| license.txt | 1 KB |
| logo.png | 33 KB |
| sector.php | 4 KB |

Figure 2.3 – Content of the fdpf181 folder on the server

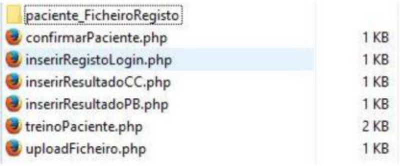
- ***leaphysioAPP***: contains PHP files dedicated to the mobile application (see Figure 2.4). It also stores photographs of patients, physiotherapists and managers, patients' LeaPhysio cards and their QRCode's.



| | |
|---|---|
| fisioterapeuta_Foto | |
| gerente_Foto | |
| paciente_Cartao | |
| paciente_Foto | |
| paciente_QrCode | |
| activarTreino.php | 1 KB |
| adicionarFavorito.php | 2 KB |
| alterarDadosFisio.php | 2 KB |
| alterarDadosGerente.php | 2 KB |
| alterarDadosPaciente.php | 2 KB |
| alterarDadosTreinos.php | 3 KB |
| alterarPasswordFisio.php | 1 KB |
| alterarPasswordGerente.php | 1 KB |
| apagarObservacao.php | 1 KB |
| apagarTreinamento.php | 1 KB |
| confirmFisio.php | 2 KB |
| confirmGerente.php | 2 KB |
| criarFisioterapeuta.php | 3 KB |
| criarObservacao.php | 1 KB |
| criarPaciente.php | 5 KB |
| criarTreinoPaciente.php | 3 KB |
| estatisticasGeraisFisio.php | 2 KB |
| estatisticasGeraisPaciente.php | 4 KB |

Figure 2.4 – Content of the leaphysioAPP folder on the server

- ***leaphysioUNITY***: contains PHP files dedicated to the Unity application (see Figure 2.5). It also stores the coordinate files of the hands resulting from the training performed by the patients.



| | |
|---|---|
| paciente_FicheiroRegisto | |
| confirmarPaciente.php | 1 KB |
| inserirRegistoLogin.php | 1 KB |
| inserirResultadoCC.php | 1 KB |
| inserirResultadoPB.php | 1 KB |
| treinoPaciente.php | 2 KB |
| uploadFicheiro.php | 1 KB |

Figure 2.5 – Content of the leaphysioUNITY folder on the server

- ***connectionLeaPhysio.php***: This file is used to connect to the database. It is the only file that is common to the two existing applications.

### 2.3. Database

The entire database structure was developed in the MySQL Workbench program. This program has several tools to create a database through its visual presentation. The database was developed in MySQL. This DBMS is one of the most popular in the world, and an essential system in LAMP architecture. Initially a system needs analysis was performed to find out what kind of data the database should store. After completion, the database was exported to a file, using the MySQL Workbench, to later import that file to the server through the phpMyAdmin tool. In Figure 2.6 it is possible to observe the final diagram of the database.

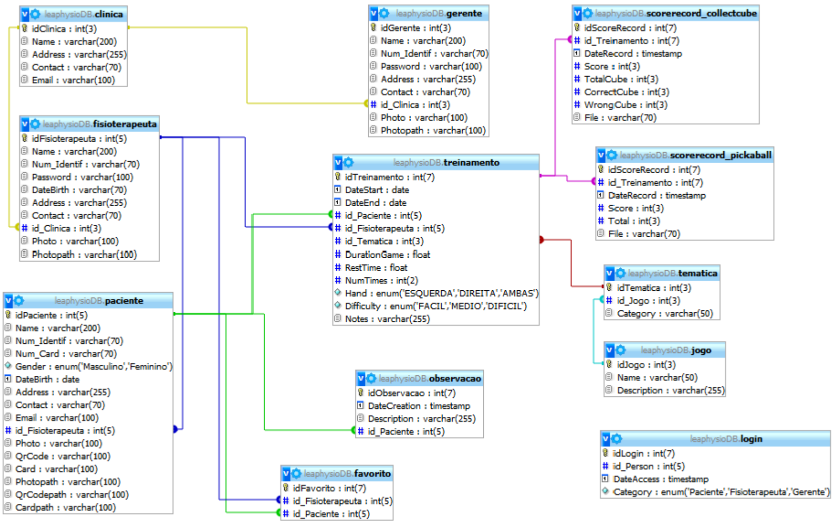
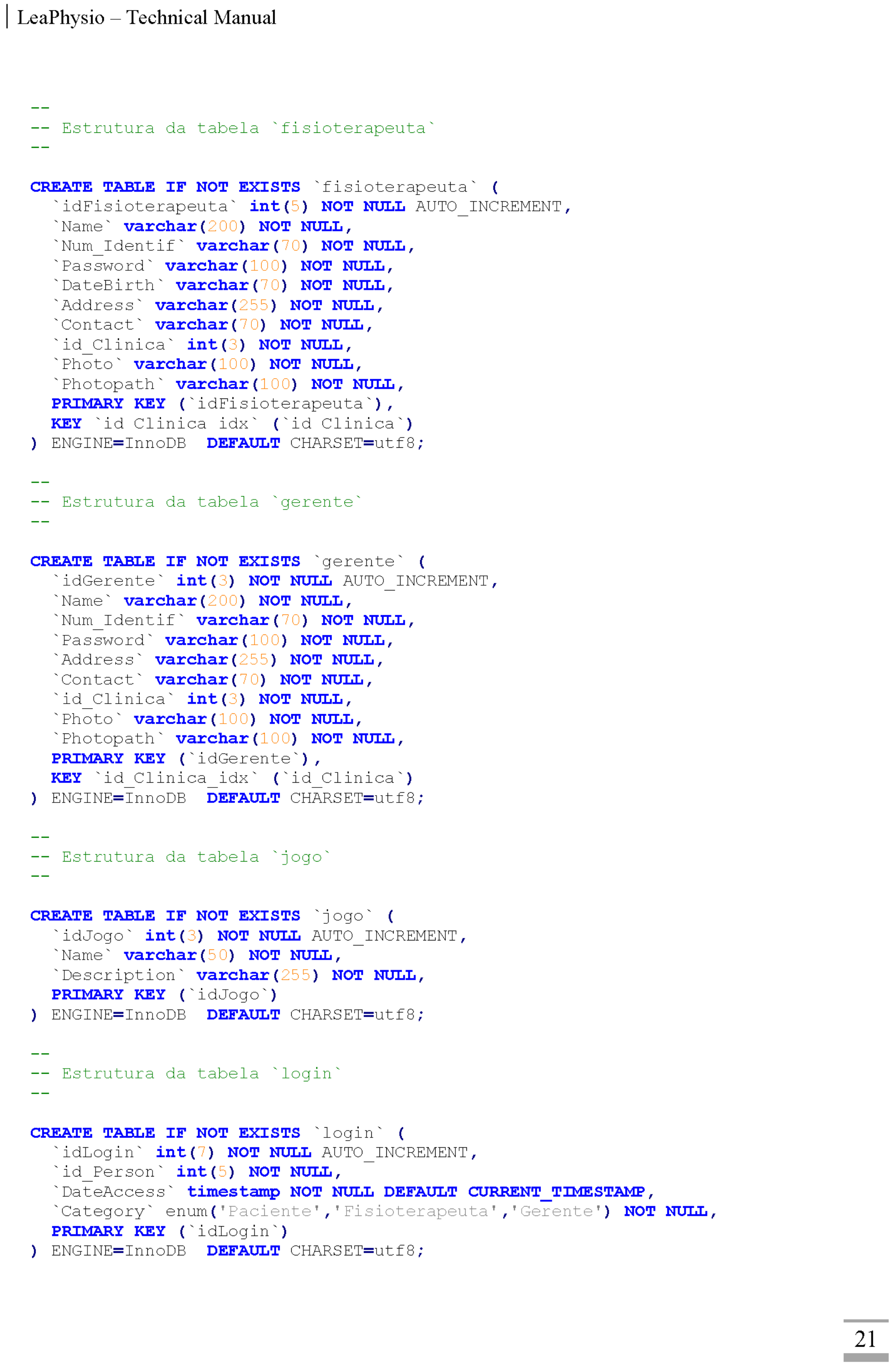


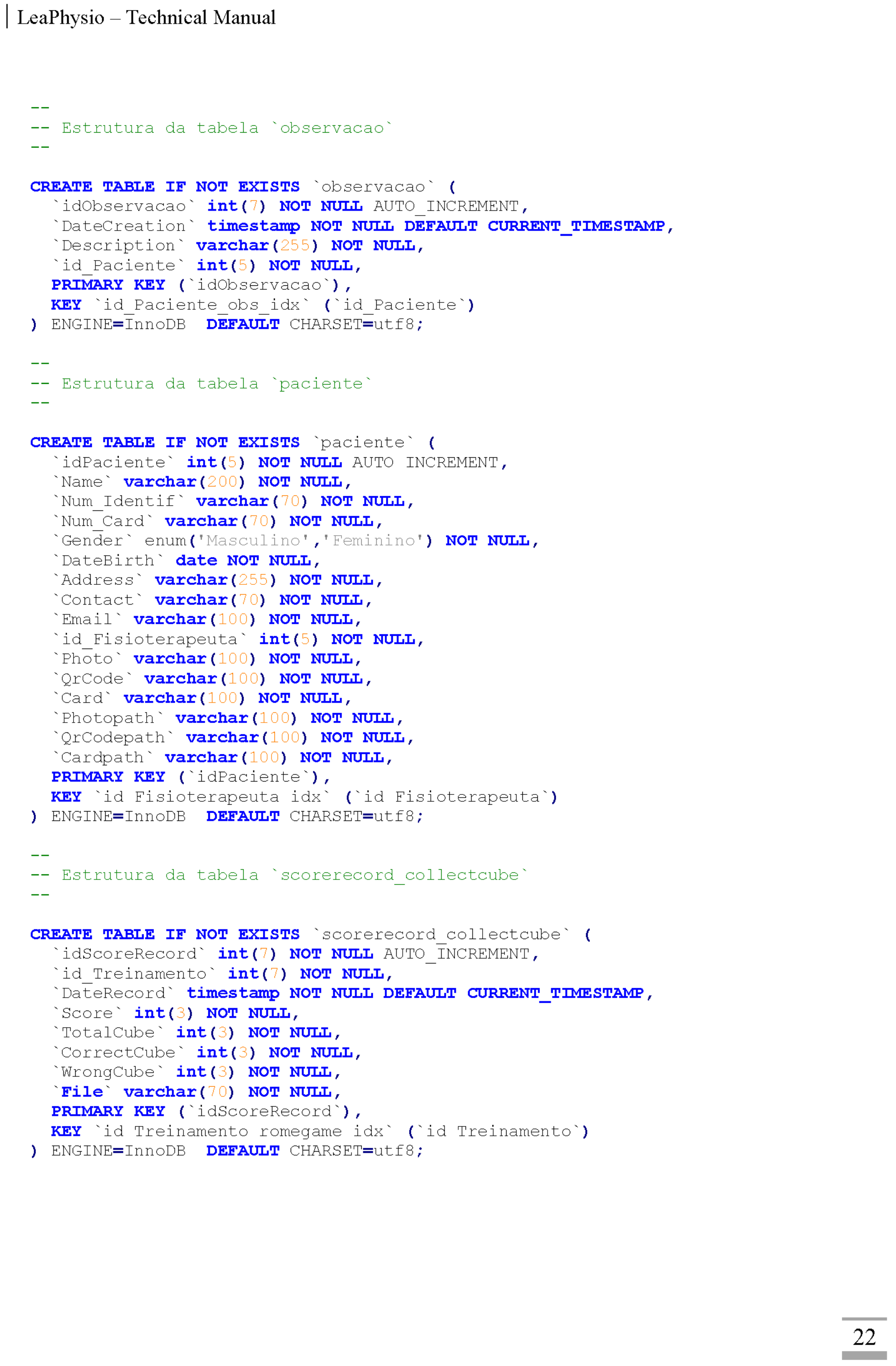
Figure 2.6 – Diagram Database

The database and the above tables were created in the Database with the following script in SQL:

```
--
-- Estrutura da tabela `clinica`
--

CREATE TABLE IF NOT EXISTS `clinica` (
  `idClinica` int(3) NOT NULL AUTO_INCREMENT,
  `Name` varchar(200) NOT NULL,
  `Address` varchar(255) NOT NULL,
  `Contact` varchar(70) NOT NULL,
  `Email` varchar(100) NOT NULL,
  PRIMARY KEY (`idClinica`)
) ENGINE=InnoDB  DEFAULT CHARSET=utf8;

--
-- Estrutura da tabela `favorito`
--

CREATE TABLE IF NOT EXISTS `favorito` (
  `idFavorito` int(7) NOT NULL AUTO_INCREMENT,
  `id_Fisioterapeuta` int(5) NOT NULL,
  `id_Paciente` int(5) NOT NULL,
  PRIMARY KEY (`idFavorito`),
  KEY `id_Paciente_idx` (`id_Paciente`),
  KEY `id_Fisioterapeuta_idx` (`id_Fisioterapeuta`)
) ENGINE=InnoDB  DEFAULT CHARSET=utf8;
```



```
--
-- Estrutura da tabela `fisioterapeuta`
--

CREATE TABLE IF NOT EXISTS `fisioterapeuta` (
  `idFisioterapeuta` int(5) NOT NULL AUTO_INCREMENT,
  `Name` varchar(200) NOT NULL,
  `Num_Identif` varchar(70) NOT NULL,
  `Password` varchar(100) NOT NULL,
  `DateBirth` varchar(70) NOT NULL,
  `Address` varchar(255) NOT NULL,
  `Contact` varchar(70) NOT NULL,
  `id_Clinica` int(3) NOT NULL,
  `Photo` varchar(100) NOT NULL,
  `Photopath` varchar(100) NOT NULL,
  PRIMARY KEY (`idFisioterapeuta`),
  KEY `id Clinica idx` (`id Clinica`)
) ENGINE=InnoDB  DEFAULT CHARSET=utf8;

--
-- Estrutura da tabela `gerente`
--

CREATE TABLE IF NOT EXISTS `gerente` (
  `idGerente` int(3) NOT NULL AUTO_INCREMENT,
  `Name` varchar(200) NOT NULL,
  `Num_Identif` varchar(70) NOT NULL,
  `Password` varchar(100) NOT NULL,
  `Address` varchar(255) NOT NULL,
  `Contact` varchar(70) NOT NULL,
  `id_Clinica` int(3) NOT NULL,
  `Photo` varchar(100) NOT NULL,
  `Photopath` varchar(100) NOT NULL,
  PRIMARY KEY (`idGerente`),
  KEY `id_Clinica_idx` (`id_Clinica`)
) ENGINE=InnoDB  DEFAULT CHARSET=utf8;

--
-- Estrutura da tabela `jogo`
--

CREATE TABLE IF NOT EXISTS `jogo` (
  `idJogo` int(3) NOT NULL AUTO_INCREMENT,
  `Name` varchar(50) NOT NULL,
  `Description` varchar(255) NOT NULL,
  PRIMARY KEY (`idJogo`)
) ENGINE=InnoDB  DEFAULT CHARSET=utf8;

--
-- Estrutura da tabela `login`
--

CREATE TABLE IF NOT EXISTS `login` (
  `idLogin` int(7) NOT NULL AUTO_INCREMENT,
  `id_Person` int(5) NOT NULL,
  `DateAccess` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `Category` enum('Paciente','Fisioterapeuta','Gerente') NOT NULL,
  PRIMARY KEY (`idLogin`)
) ENGINE=InnoDB  DEFAULT CHARSET=utf8;
```



LeaPhysio – Technical Manual

```
--
-- Estrutura da tabela `observacao`
--

CREATE TABLE IF NOT EXISTS `observacao` (
  `idObservacao` int(7) NOT NULL AUTO_INCREMENT,
  `DateCreation` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `Description` varchar(255) NOT NULL,
  `id_Paciente` int(5) NOT NULL,
  PRIMARY KEY (`idObservacao`),
  KEY `id_Paciente_obs_idx` (`id_Paciente`)
) ENGINE=InnoDB  DEFAULT CHARSET=utf8;

--
-- Estrutura da tabela `paciente`
--

CREATE TABLE IF NOT EXISTS `paciente` (
  `idPaciente` int(5) NOT NULL AUTO INCREMENT,
  `Name` varchar(200) NOT NULL,
  `Num_Identif` varchar(70) NOT NULL,
  `Num_Card` varchar(70) NOT NULL,
  `Gender` enum('Masculino','Feminino') NOT NULL,
  `DateBirth` date NOT NULL,
  `Address` varchar(255) NOT NULL,
  `Contact` varchar(70) NOT NULL,
  `Email` varchar(100) NOT NULL,
  `id_Fisioterapeuta` int(5) NOT NULL,
  `Photo` varchar(100) NOT NULL,
  `QrCode` varchar(100) NOT NULL,
  `Card` varchar(100) NOT NULL,
  `Photopath` varchar(100) NOT NULL,
  `QrCodepath` varchar(100) NOT NULL,
  `Cardpath` varchar(100) NOT NULL,
  PRIMARY KEY (`idPaciente`),
  KEY `id Fisioterapeuta idx` (`id Fisioterapeuta`)
) ENGINE=InnoDB  DEFAULT CHARSET=utf8;

--
-- Estrutura da tabela `scorerecord_collectcube`
--

CREATE TABLE IF NOT EXISTS `scorerecord_collectcube` (
  `idScoreRecord` int(7) NOT NULL AUTO_INCREMENT,
  `id_Treinamento` int(7) NOT NULL,
  `DateRecord` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `Score` int(3) NOT NULL,
  `TotalCube` int(3) NOT NULL,
  `CorrectCube` int(3) NOT NULL,
  `WrongCube` int(3) NOT NULL,
  `File` varchar(70) NOT NULL,
  PRIMARY KEY (`idScoreRecord`),
  KEY `id Treinamento romegame idx` (`id Treinamento`)
) ENGINE=InnoDB  DEFAULT CHARSET=utf8;
```

22

```
--
-- Estrutura da tabela `tematica`
--

CREATE TABLE IF NOT EXISTS `tematica` (
  `idTematica` int(3) NOT NULL AUTO_INCREMENT,
  `id_Jogo` int(3) NOT NULL,
  `Category` varchar(50) NOT NULL,
  PRIMARY KEY (`idTematica`),
  KEY `id_Jogo_tematica_idx` (`id_Jogo`)
) ENGINE=InnoDB  DEFAULT CHARSET=utf8;

--
-- Estrutura da tabela `treinamento`
--

CREATE TABLE IF NOT EXISTS `treinamento` (
  `idTreinamento` int(7) NOT NULL AUTO_INCREMENT,
  `DateStart` date NOT NULL,
  `DateEnd` date NOT NULL,
  `id_Paciente` int(5) NOT NULL,
  `id_Fisioterapeuta` int(5) NOT NULL,
  `id_Tematica` int(3) NOT NULL,
  `DurationGame` float NOT NULL,
  `RestTime` float NOT NULL,
  `NumTimes` int(2) NOT NULL,
  `Hand` enum('ESQUERDA','DIREITA','AMBAS') NOT NULL,
  `Difficulty` enum('FACIL','MEDIO','DIFICIL') NOT NULL,
  `Notes` varchar(255) NOT NULL,
  PRIMARY KEY (`idTreinamento`),
  KEY `Id_Paciente_trainer_idx` (`id_Paciente`),
  KEY `Id_Tematica_trainer_idx` (`id_Tematica`),
  KEY `Id_Fisioterapeuta_trainer_idx` (`id_Fisioterapeuta`)
) ENGINE=InnoDB  DEFAULT CHARSET=utf8;
```

### 2.4. PHP Scripts

The use of PHP scripts is required to perform communication between the server and the applications (Unity and Android). These files are stored on the server, and when it is necessary to perform some search operation, change data or even erase data, the corresponding file is invoked using a HTTP connection.

Regarding PHP scripts, we can catalog the files in two categories:

- Those that are used by the Unity application. They are scripts that have as main function to look for necessary information about the patients and training plans to be able to configure the game scenarios, so that they are prepared to the needs of each training and to record information on the results, either at the level of scores or coordinates fingers and hands;
- The ones that are used by the Android application. They are scripts for searching patient information, creating new patients, creating training plans, changing training parameters, erasing drills or even visualizing results.

All programming could only have been done in a single PHP script, but due to the number of existing scripts and for better understanding, organization and detection of possible errors, it was decided to perform separate scripts. The connectionLeaPhysio.php file is used by all PHP scripts, which is a fundamental file for connecting to the database.

- ***confirmarPaciente.php***: check if the patient is valid in the system. This script is invoked when the patient in the Unity application shows their QRCode of the LeaPhysio card in the Webcam. The application reads the information and the script tries to check if this information belongs to any patient. If it exists, returns the necessary information to follow the process of the application, otherwise, an error message is sent;
- ***treinoPaciente.php***: using the Patient id, this script tries to gather information about the patient's active training, and it is through the information provided by the script that the Unity application configures the games for the patient;
- ***criarTreinoPaciente.php***: allows create a training for a patient. This script is used by the physiotherapist in the application and the information is sent by the POST method of the HTTP connection. The script returns a positive result if no error has occurred. The code structure of PHP scripts for the most part is similar. Initially, checks that serve to verify that the request method is what is indicated in the file and that the arguments that are sent through the request method exist. If there are no errors, the connection to the DB is started, SQL query is created and later executed, returning the results found and through them to make decisions, for example in case of not finding some result write a message to indicate the same, or otherwise, save the information in a structure to be written.

We'll look in more detail at the script code used by the Android application.

```
1 <?php
2 if ($_SERVER['REQUEST_METHOD'] == 'POST') {
3   if (isset($ POST['name'])) {
4       require_once("../connectionLeaPhysio.php");
5       error_reporting(0);
6
7       $name = strip_tags(trim($_POST['name']));
8
9       $sql = "SELECT clinica.Name, clinica.Address, clinica.Contact,
clinica.Email FROM clinica WHERE clinica.Name='$name'";
10
              11       $result = mysqli_query($conn, $sql);
12      if (mysqli_num_rows($result) > 0) {
13          $arr     = array();
14          $row     = mysqli_fetch_array($result);
15          $nameC   = $row['Name'];
16          $address = $row['Address'];
17          $contact = $row['Contact'];
18          $email   = $row['Email'];
19
20          $arr[] = array( 'Name' => "$nameC", 'Address' =>
"$address",
21              'Contact' => "$contact", 'Email' => "$email" );
22          echo '{"clinic":' . json_encode($arr) . '}';
23          ;
24      } else {
25          echo 'Error';
26      }
27
28      mysqli_close($conn);
29  }
30}
31?>
```

Line number 2 checks if the request method is of type POST, if it is then continuing the rest of the script process. Line number 3 verifies that the POST method contains the name of the variables that it is waiting to receive and that they are not empty. If there are no errors, the script invokes the *connectionLeaPhysio.php* file to make a connection to the database. Line 7 is intended to store the value of the variable sent by POST to a local variable. In line 9 is built the SQL query to search the database, it will look for information about the clinic. Line 11 executes the query, while line 12 checks whether the result returned by the query execution is greater than 0, if it is less than or equal, is written to an error message through the echo function. If it is greater than 0, the resulting information, stored in variables, is read for later storage in an array. At the end the array is converted to a JSON format and is written on the page, and you can see the result. This result is never visible to the user directly, because the user doesn’t execute the script, for example, in a browser, it is the Android application in this case that executes and reads the information written by the PHP script, thus showing the result to the user. Finally, the script closes the connection to the DB.

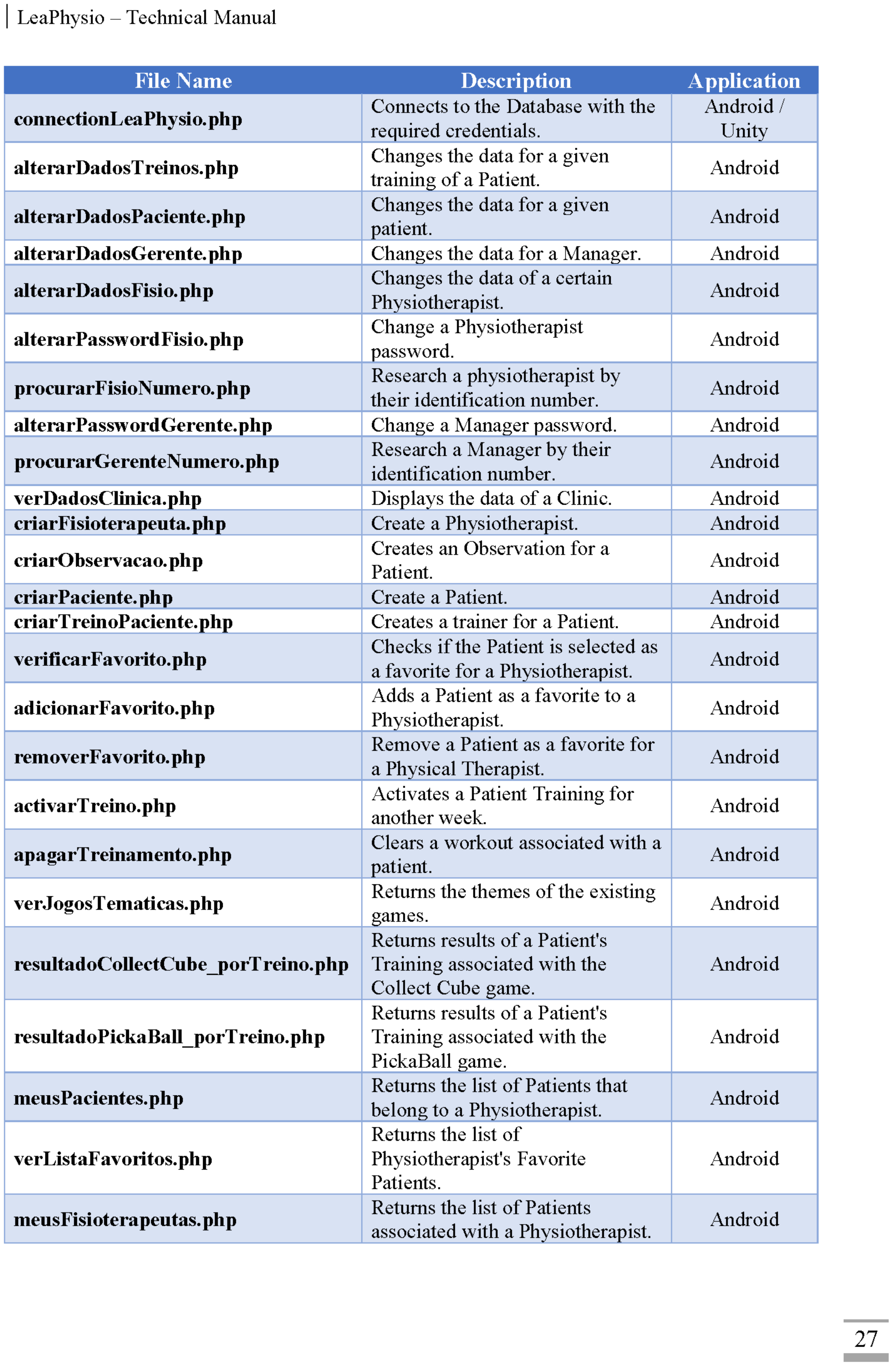
Patient reports are created through PHP files, hosted on the server. The *FDPF Library PDF generator* was used. The report has three chapters:

- Chapter with general information;
- Chapter about trainings;
- Chapter with results.

About programming, the code was structured to be easy to add new chapters in the future. The following is an excerpt from the file *exportPDF.php*.

```
    function ChapterNumber1($num_identif, $name_Fisio){
        $this->ChapterHeader("I", utf8_decode('Informações Gerais'));
        $this->SetFont('Helvetica', '', 10);
        $date = date('Y-m-d H:i:s');
        $this->Cell(0, 5, utf8_decode('Data do Relatório: '. $date), 0,
1, 'R');
        $this->GeneralInformation($num_identif, $name_Fisio);
        $this->Cell(0, 5, '', 0, 1);
    }
    function ChapterNumber2($num_identif){
        $this->ChapterHeader("II", utf8_decode('Treinos'));
        $this->SetFont('Helvetica', '', 10);
        $this->GeneralStatistics($num_identif);
        $this->Cell(0, 5, '', 0, 1);
        $this->SetFont('Helvetica', 'B', 10);
        $this->Cell(0, 5, utf8_decode('Treinos Activos'), 0, 1, 'C');
        $this->SetFont('Helvetica', '', 10);
        $this->Cell(0, 5, '', 0, 1);
        $this->ActiveTrainings($num_identif);
        $this->Cell(0, 5, '', 0, 1);
    }
if($_SERVER['REQUEST_METHOD'] == 'POST'){
    if (isset($_POST['num_identif']) && isset($_POST['name_Fisio'])){
        error_reporting(0);
        $num_identif = strip_tags(trim($_POST['num_identif']));
        $name_Fisio = strip_tags(trim($_POST['name_Fisio']));

        $pdf = new PDF();
        $pdf->AliasNbPages();
        $pdf->AddPage();
        $pdf->ChapterNumber1($num_identif, $name_Fisio);
        $pdf->ChapterNumber2($num_identif);
        $pdf->ChapterNumber3($num_identif);

        $pdf->Output();
    }
}
```

The following table describes the PHP modules and which applications are used.



LeaPhysio – Technical Manual

| File Name | Description | Application |
|---|---|---|
| **connectionLeaPhysio.php** | Connects to the Database with the required credentials. | Android / Unity |
| **alterarDadosTreinos.php** | Changes the data for a given training of a Patient. | Android |
| **alterarDadosPaciente.php** | Changes the data for a given patient. | Android |
| **alterarDadosGerente.php** | Changes the data for a Manager. | Android |
| **alterarDadosFisio.php** | Changes the data of a certain Physiotherapist. | Android |
| **alterarPasswordFisio.php** | Change a Physiotherapist password. | Android |
| **procurarFisioNumero.php** | Research a physiotherapist by their identification number. | Android |
| **alterarPasswordGerente.php** | Change a Manager password. | Android |
| **procurarGerenteNumero.php** | Research a Manager by their identification number. | Android |
| **verDadosClinica.php** | Displays the data of a Clinic. | Android |
| **criarFisioterapeuta.php** | Create a Physiotherapist. | Android |
| **criarObservacao.php** | Creates an Observation for a Patient. | Android |
| **criarPaciente.php** | Create a Patient. | Android |
| **criarTreinoPaciente.php** | Creates a trainer for a Patient. | Android |
| **verificarFavorito.php** | Checks if the Patient is selected as a favorite for a Physiotherapist. | Android |
| **adicionarFavorito.php** | Adds a Patient as a favorite to a Physiotherapist. | Android |
| **removerFavorito.php** | Remove a Patient as a favorite for a Physical Therapist. | Android |
| **activarTreino.php** | Activates a Patient Training for another week. | Android |
| **apagarTreinamento.php** | Clears a workout associated with a patient. | Android |
| **verJogosTematicas.php** | Returns the themes of the existing games. | Android |
| **resultadoCollectCube_porTreino.php** | Returns results of a Patient's Training associated with the Collect Cube game. | Android |
| **resultadoPickaBall_porTreino.php** | Returns results of a Patient's Training associated with the PickaBall game. | Android |
| **meusPacientes.php** | Returns the list of Patients that belong to a Physiotherapist. | Android |
| **verListaFavoritos.php** | Returns the list of Physiotherapist's Favorite Patients. | Android |
| **meusFisioterapeutas.php** | Returns the list of Patients associated with a Physiotherapist. | Android |

27

| | | |
|---|---|---|
| **estatisticasGeraisPaciente.php** | General Statistics on Patients Associated with a Physical Therapist. | Android |
| **confirmFisio.php** | Checks if the credentials entered correspond to a Physical Therapist, if so, returns the necessary information about the Physical Therapist. | Android |
| **confirmGerente.php** | Checks if the credentials entered correspond to a Clinic Manager, if yes, returns the necessary information about the Clinic Manager. | Android |
| **registarLogin.php** | Registers the access entry in the application by the user, either Physiotherapist or Manager. | Android |
| **apagarObservacao.php** | Clears one associated with a Patient. | Android |
| **procurarPacienteNome.php** | Find a Patient by name. | Android |
| **procurarPacienteNumero.php** | Look for a Patient by their identification number. | Android |
| **verTreinosporPaciente.php** | Returns the Trainings associated with a Patient. | Android |
| **verObservacao.php** | Returns the list of Observations about a Patient. | Android |
| **verLoginsPacienteporNum.php** | Returns the list of Logins performed by a Patient in the LeaPhysio Game application. | Android |
| **informacaoTreinos.php** | Returns information about how many active trainers there are associated with a Patient. | Android |
| **estatisticasGeraisFisio.php** | General Statistics on Physiotherapists associated with a clinic, which is the same as the Manager. | Android |
| **procurarPacienteCartao.php** | Finding a Patient through your LeaPhysio Card. | Android |
| **verLoginsFisioporNum.php** | Returns a list of logins performed by a Physiotherapist. | Android |
| **verNomesPacientes.php** | Returns a list of Patient names that belongs to a Physiotherapist. | Android |
| **gerarRelatorioPDF.php** | Generates a PDF report for a given Patient. | Android |
| **confirmarPaciente.php** | Confirms whether the patient exists in the system. | Unity |
| **inserirRegistoLogin.php** | Records the Patient's entry into the System. | Unity |
| **inserirResultadoCC.php** | Add result of the CollectCube game. | Unity |

| | | |
|---|---|---|
| **inserirResultadoPB.php** | Add result of the PickaBall game. | Unity |
| **treinoPaciente.php** | List the patient's active trainers. | Unity |
| **uploadFicheiro.php** | Uploads the coordinate file to the server. | Unity |

Table 12 – PHP Scripts

# Chapter 3 – Mobile Application

## 3. Overview

This chapter aims to explain the main functions of the mobile application, in terms of programming. It also explains the description of the main classes used. The Android Studio version used to develop the application was the version 2.3.3.

The proposed application was developed to run on devices with version 4.0.4 (Ice Cream Sandwich) or higher, and its target is version 6.0 (Marshmallow). To run the application, the device must have access to the internet, either through a mobile network (3G, 4G) or Wi-Fi.

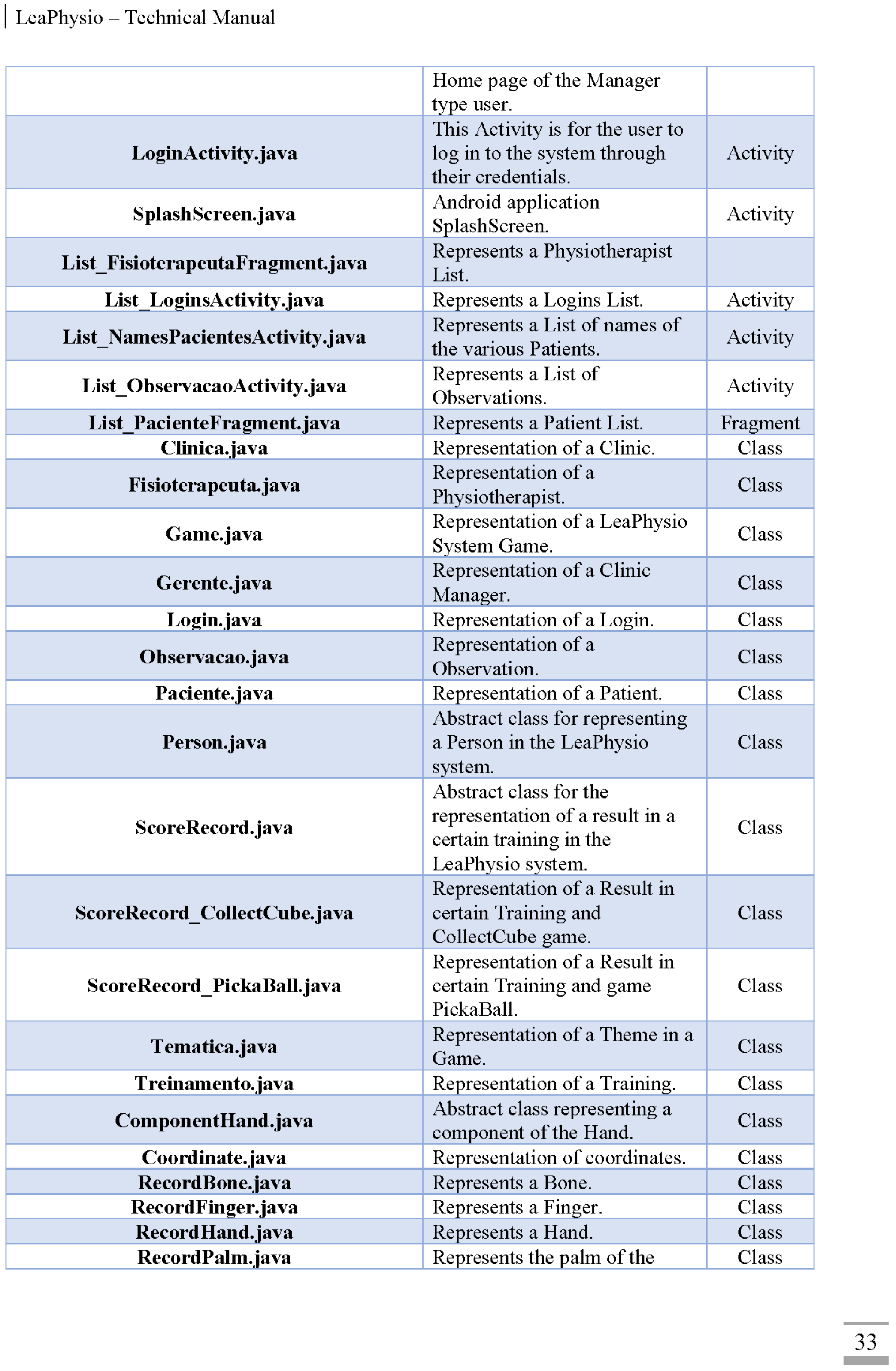
## 3.1. Java Classes

Activity is an application component that provides a screen with which users can interact to do something, such as dialing a number on the phone, taking a photo, sending an email, or viewing a map. Each activity receives a window that displays the user interface. An Activity can also support multiple Fragments, for example, to change part of the layout when an action happens. Table 13 shows the classes created to develop the mobile application, description and respective category. The classes were developed in Java.

| Class Name | Description | Category |
|---|---|---|
| **Help_Card_PacienteActivity.java** | This Activity serves to view a patient's LeaPhysio Card. | Activity |
| **Help_FisioterapeutaFragment.java** | Fragment for the visualization of Helps when a new Physiotherapist is created. Utilized by the Manager. | Fragment |
| **Help_PacienteFragment.java** | Fragment for the visualization of Helps when a new Patient is created. Utilized by the Manager. | Fragment |
| **Help_SelecionarFragment.java** | Fragment for displaying information related to field selection. | Fragment |
| **Help_SemTreinosFragment.java** | Fragment for information visualization when there is no Training. | Fragment |
| **Help_TreinoFragment.java** | Fragment for the visualization of Helps when a new Training is created. Utilized by the Manager. | Fragment |
| **Alterar_FisioterapeutaActivity.java** | Activity to change data related | Activity |

| | | |
|---|---|---|
| | to a Physiotherapist. | |
| **Alterar_GerenteActivity.java** | Activity for the change of data related to a Manager. | Activity |
| **Alterar_PacienteActivity.java** | Activity for changing patient data. | Activity |
| **Alterar_TreinoActivity.java** | Activity to change data related to a Training. | Activity |
| **DBUtil.java** | Class with information of Server URLs and PHP scripts | Class |
| **EncryptDecryptUtil.java** | Class intended for encryption and decryption of data. This class uses AES algorithm in CBC mode. | Class |
| **JsonUtil.java** | Class for converting JSON data to intended objects | Class |
| **Config_FisioterapeutaActivity.java** | Activity for the Physiotherapist settings menu. | Activity |
| **Config_GerenteActivity.java** | Activity for the manager settings menu. | Activity |
| **Criar_FisioterapeutaFragment.java** | Fragment destined to the creation of a Physiotherapist in the system. Used by the Manager type user. | Fragment |
| **Criar_ModelActivity.java** | Activity to select which 'Create' Fragment to call to perform the task. | Activity |
| **Criar_ObservacaoActivity.java** | Activity for creating a Patient Observation. | Activity |
| **Criar_PacienteFragment.java** | Fragment intended for the creation of a Patient in the system. Used by the user of the Physiotherapist type. | Class |
| **Criar_PagerAdapter.java** | Create a PagerAdapter according to the 'Create' mode that is being used (Create_Patient, Create_Physiotherapist, Create_Trainer). This joins the intended 'Create' mode with the corresponding 'Help' in different ones, configured with the needs imposed by the application. | Class |
| **Criar_TreinoFragment.java** | Fragment designed to create a Patient Training. Used by the user of the Physiotherapist type. | Fragment |
| **PageAdapter.java** | Abstract class to represent the number of separators to be | Class |

| | | |
|---|---|---|
| | used in a Fragment. | |
| **Detail_ClinicaActivity.java** | This Activity shows the details (for example, name, address, contact) of a Clinic. | Activity |
| **Detail_Fisioterapeuta_GerenteActivity.java** | This Activity shows the details (for example, name, address, contact) of a Physiotherapist, but only what is necessary, since this Activity is intended for the Manager type user and not the Physiotherapist type. | Activity |
| **Detail_FisioterapeutaActivity.java** | This Activity shows the details (for example, name, address, contact) of a Physiotherapist. | Activity |
| **Detail_GerenteActivity.java** | This Activity shows the details (for example, name, address, contact) of a Manager. | Activity |
| **Detail_InfoFisioterapeutaActivity.java** | This Activity shows more information (statistics) about the Physiotherapist. | Activity |
| **Detail_PacienteActivity.java** | This Activity shows the details (for example, name, address, contact) of a Patient. | Activity |
| **Detail_TreinosPacienteActivity.java** | This Activity shows the menu of a Patient Training. List all trainings. | Activity |
| **Detail_TreinosPacienteFragment.java** | This Fragment shows the details of a Training. | Fragment |
| **Information_FisioterapeutaActivity.java** | This Activity represents the profile page of a Physiotherapist. | Activity |
| **Information_PacienteActivity.java** | This Activity represents a Patient's profile page. | Activity |
| **ItemList_FisioterapeutaAdapter.java** | Represents an item for Physiotherapist Lists. | Class |
| **ItemList_LoginAdapter.java** | Represents an item for Login Lists. | Class |
| **ItemList_ObservacaoAdapter.java** | Represents an item for Observation Lists. | Class |
| **ItemList_PacienteAdapter.java** | Represents an item for Patient Lists. | Class |
| **ControlPanel_FisioterapeutaActivity.java** | Homepage for a Physiotherapist type user. | Activity |
| **ControlPanel_GerenteActivity.java** | Home page for a Manager type user. | Activity |
| **FrontPage_FisioterapeutaFragment.java** | This Fragment is used on the Home page of the Physiotherapist type user. | Fragment |
| **FrontPage_GerenteFragment.java** | This Fragment is used on the | |



| | | |
|---|---|---|
| | Home page of the Manager type user. | |
| **LoginActivity.java** | This Activity is for the user to log in to the system through their credentials. | Activity |
| **SplashScreen.java** | Android application SplashScreen. | Activity |
| **List_FisioterapeutaFragment.java** | Represents a Physiotherapist List. | |
| **List_LoginsActivity.java** | Represents a Logins List. | Activity |
| **List_NamesPacientesActivity.java** | Represents a List of names of the various Patients. | Activity |
| **List_ObservacaoActivity.java** | Represents a List of Observations. | Activity |
| **List_PacienteFragment.java** | Represents a Patient List. | Fragment |
| **Clinica.java** | Representation of a Clinic. | Class |
| **Fisioterapeuta.java** | Representation of a Physiotherapist. | Class |
| **Game.java** | Representation of a LeaPhysio System Game. | Class |
| **Gerente.java** | Representation of a Clinic Manager. | Class |
| **Login.java** | Representation of a Login. | Class |
| **Observacao.java** | Representation of a Observation. | Class |
| **Paciente.java** | Representation of a Patient. | Class |
| **Person.java** | Abstract class for representing a Person in the LeaPhysio system. | Class |
| **ScoreRecord.java** | Abstract class for the representation of a result in a certain training in the LeaPhysio system. | Class |
| **ScoreRecord_CollectCube.java** | Representation of a Result in certain Training and CollectCube game. | Class |
| **ScoreRecord_PickaBall.java** | Representation of a Result in certain Training and game PickaBall. | Class |
| **Tematica.java** | Representation of a Theme in a Game. | Class |
| **Treinamento.java** | Representation of a Training. | Class |
| **ComponentHand.java** | Abstract class representing a component of the Hand. | Class |
| **Coordinate.java** | Representation of coordinates. | Class |
| **RecordBone.java** | Represents a Bone. | Class |
| **RecordFinger.java** | Represents a Finger. | Class |
| **RecordHand.java** | Represents a Hand. | Class |
| **RecordPalm.java** | Represents the palm of the | Class |

| | | |
|---|---|---|
| | hand. | |
| **RecordWrist.java** | Represents the wrist of the hand. | Class |
| **StructToGraph.java** | Representing a data structure for graphing, stores the time values and the coordinates in which they occurred. | Class |
| **Procurar_FisioterapeutaFragment.java** | Fragment for representation of Physiotherapist Search options. Used by the user of the Manager type. | Fragment |
| **Procurar_PacienteFragment.java** | Fragment for representation of Patient Search options. Used by the Physiotherapist type user. | Fragment |
| **Resultados_TrackingActivity.java** | This Activity represents the certain functions for displaying graphs relative to the Patient's hands. | Activity |
| **Resultados_TrackingGraphFragment.java** | This Fragment allows the visualization of graphs relative to the Patient's hands. It is used by Activity Resultados_TrackingActivity. | Fragment |
| **Resultados_TreinoActivity.java** | This Activity represents certain functions for visualizing the results from the various Patient Trainings. | Activity |
| **Resultados_TreinoFragment.java** | This Fragment allows the visualization of a certain individual result in terms of punctuation for a Training. | Fragment |
| **Resultados_TreinoTodosFragment.java** | This Fragment allows you to view all the results of a given Training on the same chart. It is used by Activity Resultados_TreinoActivity. | Fragment |

Table 13 – Names of classes used in the mobile application and their description

### 3.2. Application Main Features

**Search for Patient through QRCode** – The physiotherapist can search for a patient through the QRCode inserted in the patient's LeaPhysio card, to access the profile. The *Zxing library* is used for this purpose. This library allows the reading of different types of barcodes and configurable in the visual aspects for the end user.

**Charts** – Patient training results are represented by line charts and bars. For this purpose, the *MPAndroidChart* library is used. This library allows access to various types of graphics and configure them. The mobile application uses bar graphs to represent the results for the scores and line graphs for the results for the coordinates of the hands. These graphs allow the user to zoom-in and zoom-out for better observation of the data. A pie chart is also used to represent the percentage of patients in terms of their gender, male or female.

**Circular Profile Images** – To make the application more appealing to users, the profile images of the actors (patient, physiotherapist or manager) instead of having a square shape, was used a circular shape. For this, the library *CircleImageView* was used.

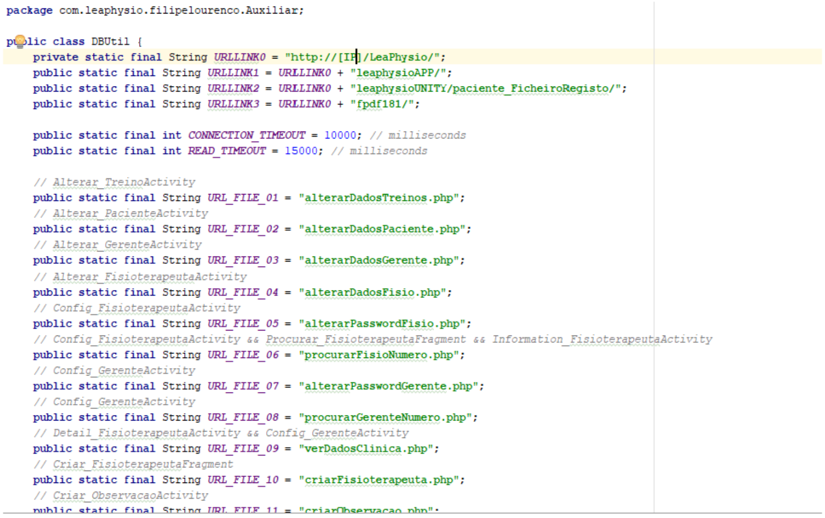
**Encryption and Data Decryption** – Several fields in the database tables are encrypted to ensure protection of sensitive data, for example, the home address of the patient. The mobile application uses the Encryption library to encrypt and decrypt data, for example, the mobile application wants to search the database for information about a specific clinic using the corresponding PHP script over an HTTP connection. If no anomaly occurs, the script executes the search in the database and returns the result obtained in a JSON format with the encrypted data. The mobile application reads this information and decrypts the data using the previously defined keys. Finally, the decryption information is presented to the user. To insert a record in the database, the application encrypts the information and executes the corresponding PHP script, sending the information through the POST method and is the script that tries to insert the encrypted data into the database. This library uses by default the Advanced Encryption Standard algorithm in Cipher Block Chaining mode.

**HTTP Connection** – The application communicates with the database through HTTP requests. To perform this type of connection is used AsyncTask. This class is used to facilitate the processing of background operations and to use the data in the User Interface.

## 3.3. Programming

This subchapter explains some of the main functions used by the application. In each class the code is commented on for a better understanding of the lines of code.

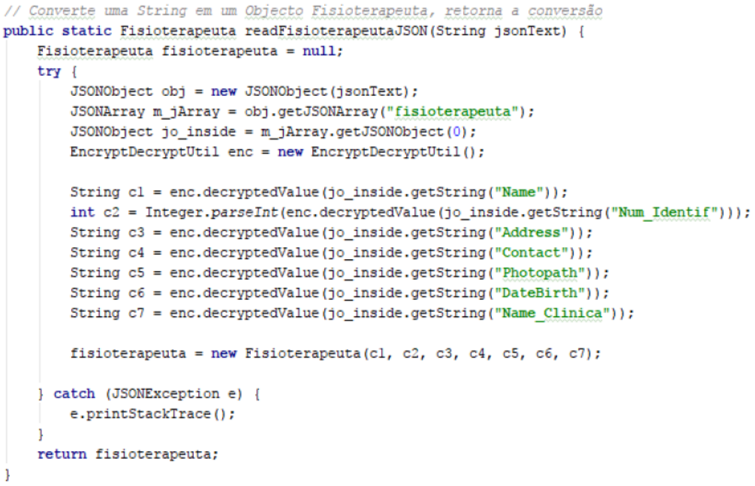
The DBUtil class saves the IP address of the server and the name of the PHP scripts to be used in the future to execute HTTP requests. Figure 3.1 illustrates the class code with its IP address and file names. It is in this class that future files must be added.



```java
package com.leaphysio.filipelourenco.Auxiliar;

public class DBUtil {
    private static final String URLLINK0 = "http://[IP]/LeaPhysio/";
    public static final String URLLINK1 = URLLINK0 + "leaphysioAPP/";
    public static final String URLLINK2 = URLLINK0 + "leaphysioUNITY/paciente_FicheiroRegisto/";
    public static final String URLLINK3 = URLLINK0 + "fpdf181/";

    public static final int CONNECTION_TIMEOUT = 10000; // milliseconds
    public static final int READ_TIMEOUT = 15000; // milliseconds

    // Alterar_TreinoActivity
    public static final String URL_FILE_01 = "alterarDadosTreinos.php";
    // Alterar_PacienteActivity
    public static final String URL_FILE_02 = "alterarDadosPaciente.php";
    // Alterar_GerenteActivity
    public static final String URL_FILE_03 = "alterarDadosGerente.php";
    // Alterar_FisioterapeutaActivity
    public static final String URL_FILE_04 = "alterarDadosFisio.php";
    // Config_FisioterapeutaActivity
    public static final String URL_FILE_05 = "alterarPasswordFisio.php";
    // Config_FisioterapeutaActivity && Procurar_FisioterapeutaFragment && Information_FisioterapeutaActivity
    public static final String URL_FILE_06 = "procurarFisioNumero.php";
    // Config_GerenteActivity
    public static final String URL_FILE_07 = "alterarPasswordGerente.php";
    // Config_GerenteActivity
    public static final String URL_FILE_08 = "procurarGerenteNumero.php";
    // Detail_FisioterapeutaActivity && Config_GerenteActivity
    public static final String URL_FILE_09 = "verDadosClinica.php";
    // Criar_FisioterapeutaFragment
    public static final String URL_FILE_10 = "criarFisioterapeuta.php";
    // Criar_ObservacaoActivity
    public static final String URL_FILE_11 = "criarObservacao.php";
```

Figure 3.1 – DBUtil class

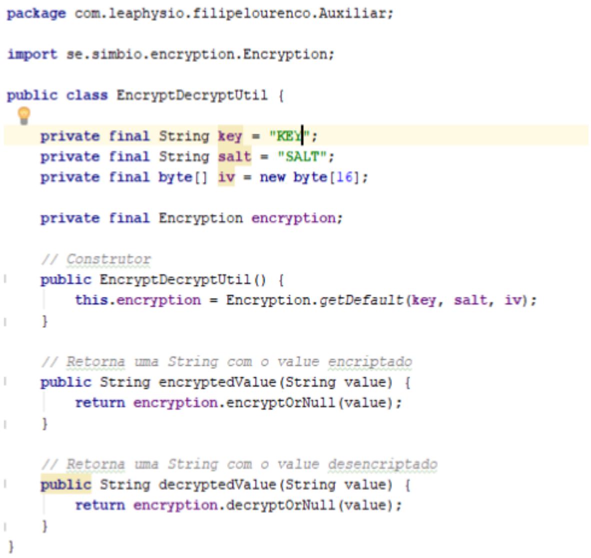
The results of PHP scripts are returned is JSON format, where it is necessary to extract their values, decrypt and create an instance of the corresponding class. Figure 3.2 indicates the code to convert a physiotherapist into JSON format for an object of the Fisioterapeuta class, using the *readFisioterapeuta()* function.



```java
// Converte uma String em um Objecto Fisioterapeuta, retorna a conversão
public static Fisioterapeuta readFisioterapeutaJSON(String jsonText) {
    Fisioterapeuta fisioterapeuta = null;
    try {
        JSONObject obj = new JSONObject(jsonText);
        JSONArray m_jArray = obj.getJSONArray("fisioterapeuta");
        JSONObject jo_inside = m_jArray.getJSONObject(0);
        EncryptDecryptUtil enc = new EncryptDecryptUtil();

        String c1 = enc.decryptedValue(jo_inside.getString("Name"));
        int c2 = Integer.parseInt(enc.decryptedValue(jo_inside.getString("Num_Identif")));
        String c3 = enc.decryptedValue(jo_inside.getString("Address"));
        String c4 = enc.decryptedValue(jo_inside.getString("Contact"));
        String c5 = enc.decryptedValue(jo_inside.getString("Photopath"));
        String c6 = enc.decryptedValue(jo_inside.getString("DateBirth"));
        String c7 = enc.decryptedValue(jo_inside.getString("Name_Clinica"));

        fisioterapeuta = new Fisioterapeuta(c1, c2, c3, c4, c5, c6, c7);

    } catch (JSONException e) {
        e.printStackTrace();
    }
    return fisioterapeuta;
}
```

Figure 3.2 – *readFisioterapeutaJSON()* function

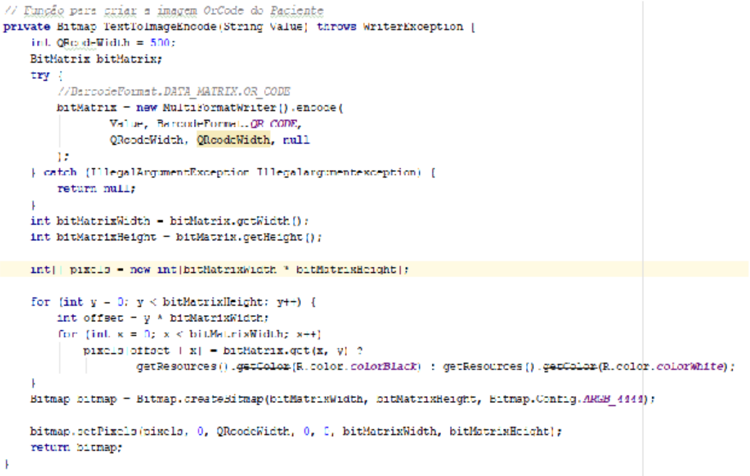
Sending encrypted data to be stored in the database and receiving data that is possible to decrypt is a key part of the mobile application. The EncrypDecryptUtil class has functions to encrypt and decrypt String's, for example, all functions of the JSONUtil class use the *decryptedValue()* function.



```
package com.leaphysio.filipelourenco.Auxiliar;

import se.simbio.encryption.Encryption;

public class EncryptDecryptUtil {

    private final String key = "KEY";
    private final String salt = "SALT";
    private final byte[] iv = new byte[16];

    private final Encryption encryption;

    // Construtor
    public EncryptDecryptUtil() {
        this.encryption = Encryption.getDefault(key, salt, iv);
    }

    // Retorna uma String com o value encriptado
    public String encryptedValue(String value) {
        return encryption.encryptOrNull(value);
    }

    // Retorna uma String com o value desencriptado
    public String decryptedValue(String value) {
        return encryption.decryptOrNull(value);
    }
}
```

Figure 3.3 – *EncryptDecryptUtil()* function

Figure 3.4 illustrates the TextToImageEncode () function. After the physiotherapist has created a patient in the system, this function is used to create the patient's QRCode.



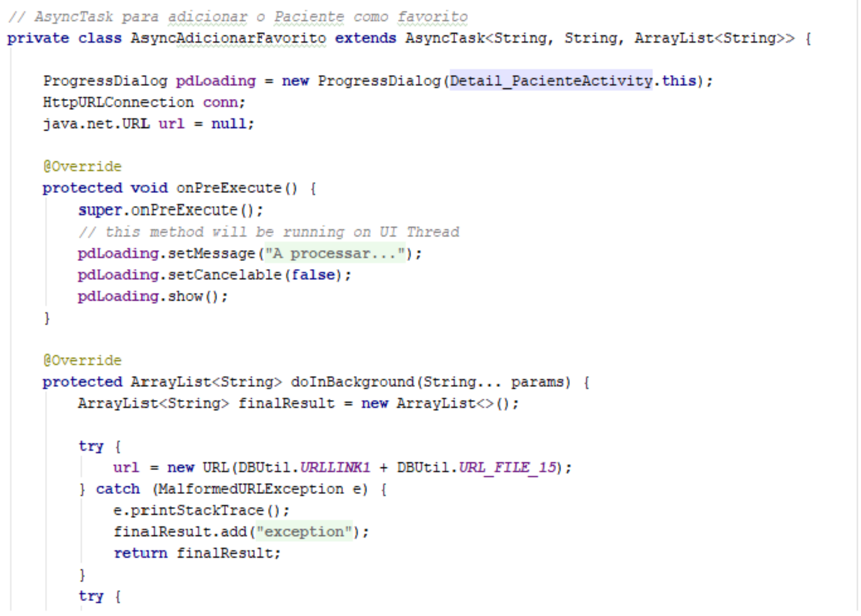
```
// Função para criar a imagem QrCode do Paciente
private Bitmap TextToImageEncode(String Value) throws WriterException {
    int QRcodeWidth = 500;
    BitMatrix bitMatrix;
    try {
        //BarcodeFormat.DATA_MATRIX.QR_CODE
        bitMatrix = new MultiFormatWriter().encode(
                Value, BarcodeFormat.QR_CODE,
                QRcodeWidth, QRcodeWidth, null
        );
    } catch (IllegalArgumentException Illegalargumentexception) {
        return null;
    }
    int bitMatrixWidth = bitMatrix.getWidth();
    int bitMatrixHeight = bitMatrix.getHeight();

    int[] pixels = new int[bitMatrixWidth * bitMatrixHeight];

    for (int y = 0; y < bitMatrixHeight; y++) {
        int offset = y * bitMatrixWidth;
        for (int x = 0; x < bitMatrixWidth; x++)
            pixels[offset + x] = bitMatrix.get(x, y) ?
                    getResources().getColor(R.color.colorBlack) : getResources().getColor(R.color.colorWhite);
    }
    Bitmap bitmap = Bitmap.createBitmap(bitMatrixWidth, bitMatrixHeight, Bitmap.Config.ARGB_4444);

    bitmap.setPixels(pixels, 0, QRcodeWidth, 0, 0, bitMatrixWidth, bitMatrixHeight);
    return bitmap;
}
```

Figure 3.4 – *TextToImageEncode()* function

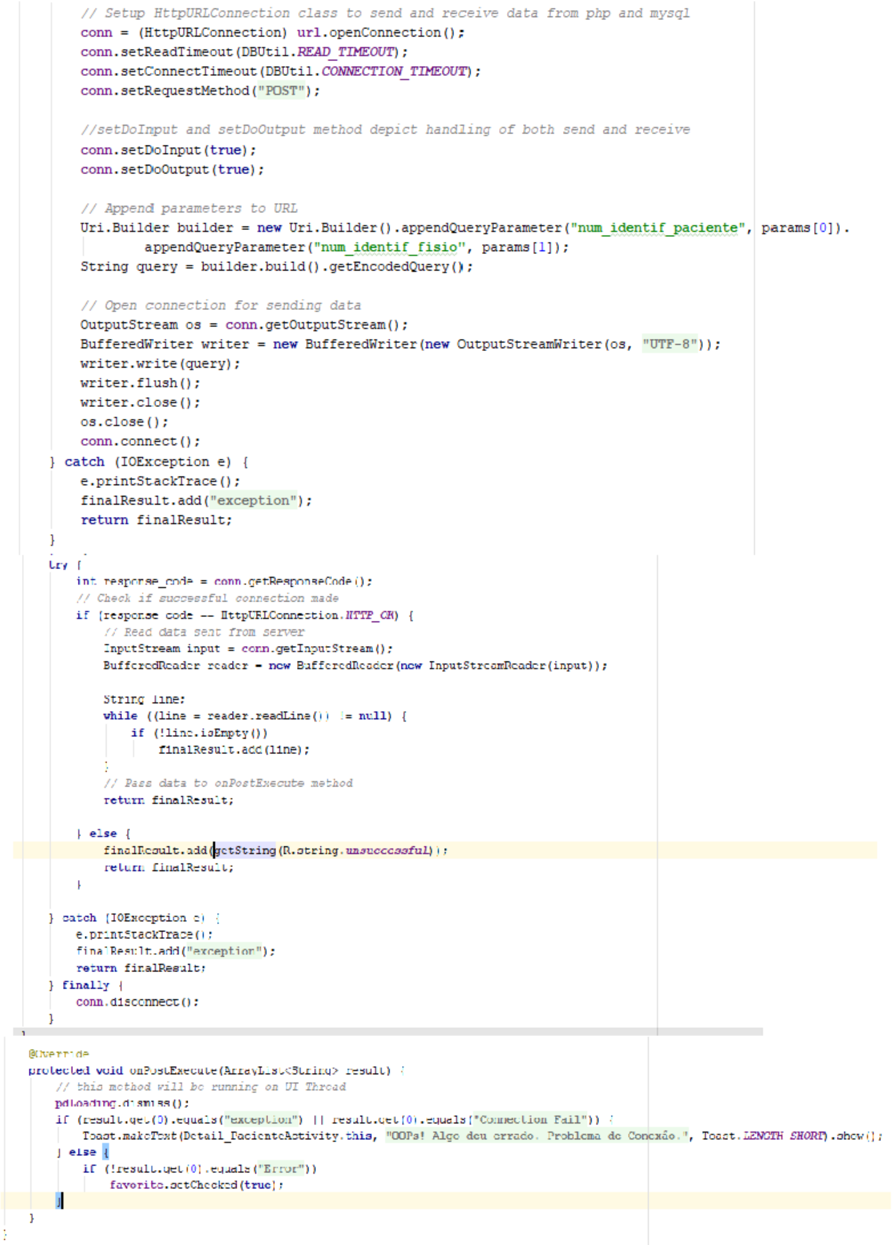
In the mobile classes application that extends *AsyncTask* are used to perform the HTTP request and are waiting for their results to execute task completion. In the application only the *onPreExecute*, *theInBackground* and *onPostExecute* methods are used.

- The *onPreExecute* method always runs before the Thread starts, and is where messages are displayed to the user if necessary.
- The required method is *doInBackground*. This is responsible for all the heavy processing as it runs on a separate Thread. This is where the information is processed, and the HTTP request is executed, and the user screen is not locked. The result is passed by parameter to the *onPostExecute* method.
- The on*PostExecute* method is the one that receives the return from *doInBackground* and is called using a Handler. Its execution takes place on the same Thread as the UI and it is here that the final decision regarding the result obtained from the *doInBackground* is processed.

Figure 3.5 represents the code for the AsyncAdicionarFavorito class that is used to add a patient as a favorite. All *AsyncTask* classes are commented out, as shown in the following figure.



```
// AsyncTask para adicionar o Paciente como favorito
private class AsyncAdicionarFavorito extends AsyncTask<String, String, ArrayList<String>> {

    ProgressDialog pdLoading = new ProgressDialog(Detail_PacienteActivity.this);
    HttpURLConnection conn;
    java.net.URL url = null;

    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        // this method will be running on UI Thread
        pdLoading.setMessage("A processar...");
        pdLoading.setCancelable(false);
        pdLoading.show();
    }

    @Override
    protected ArrayList<String> doInBackground(String... params) {
        ArrayList<String> finalResult = new ArrayList<>();

        try {
            url = new URL(DBUtil.URLLINK1 + DBUtil.URL_FILE_15);
        } catch (MalformedURLException e) {
            e.printStackTrace();
            finalResult.add("exception");
            return finalResult;
        }
        try {
```



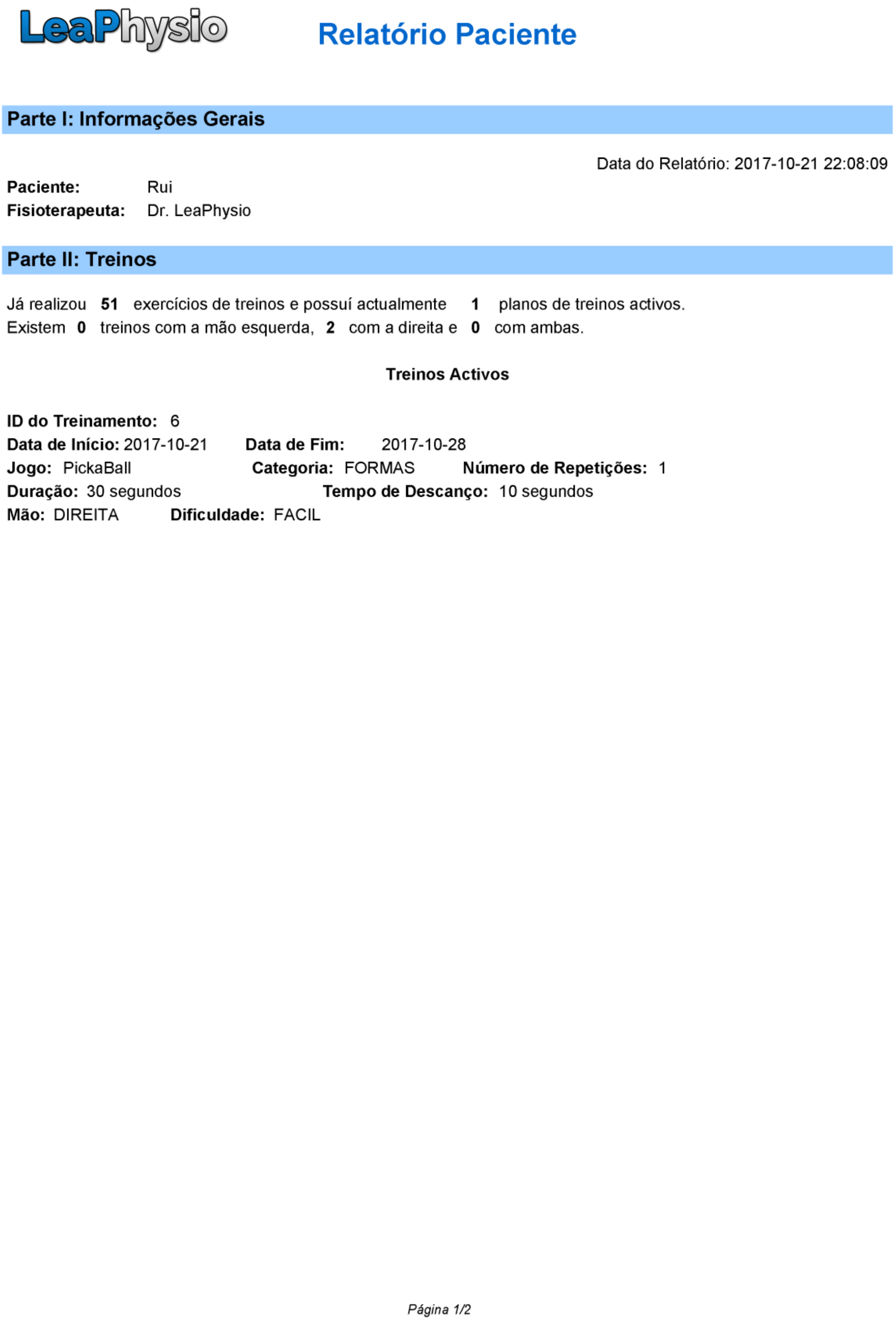
```
        // Setup HttpURLConnection class to send and receive data from php and mysql
        conn = (HttpURLConnection) url.openConnection();
        conn.setReadTimeout(DBUtil.READ_TIMEOUT);
        conn.setConnectTimeout(DBUtil.CONNECTION_TIMEOUT);
        conn.setRequestMethod("POST");

        //setDoInput and setDoOutput method depict handling of both send and receive
        conn.setDoInput(true);
        conn.setDoOutput(true);

        // Append parameters to URL
        Uri.Builder builder = new Uri.Builder().appendQueryParameter("num_identif_paciente", params[0]).
                appendQueryParameter("num_identif_fisio", params[1]);
        String query = builder.build().getEncodedQuery();

        // Open connection for sending data
        OutputStream os = conn.getOutputStream();
        BufferedWriter writer = new BufferedWriter(new OutputStreamWriter(os, "UTF-8"));
        writer.write(query);
        writer.flush();
        writer.close();
        os.close();
        conn.connect();
    } catch (IOException e) {
        e.printStackTrace();
        finalResult.add("exception");
        return finalResult;
    }
    try {
        int response_code = conn.getResponseCode();
        // Check if successful connection made
        if (response_code == HttpURLConnection.HTTP_OK) {
            // Read data sent from server
            InputStream input = conn.getInputStream();
            BufferedReader reader = new BufferedReader(new InputStreamReader(input));

            String line;
            while ((line = reader.readLine()) != null) {
                if (!line.isEmpty())
                    finalResult.add(line);
            }
            // Pass data to onPostExecute method
            return finalResult;

        } else {
            finalResult.add(getString(R.string.unsuccessful));
            return finalResult;
        }

    } catch (IOException e) {
        e.printStackTrace();
        finalResult.add("exception");
        return finalResult;
    } finally {
        conn.disconnect();
    }
}
@Override
protected void onPostExecute(ArrayList<String> result) {
    // this method will be running on UI Thread
    pdLoading.dismiss();
    if (result.get(0).equals("exception") || result.get(0).equals("Connection Fail")) {
        Toast.makeText(Detail_PacienteActivity.this, "OOPs! Algo deu errado. Problema de Conexão.", Toast.LENGTH_SHORT).show();
    } else {
        if (!result.get(0).equals("Error"))
            favorito.setChecked(true);
    }
}
}
```

Figure 3.5 – AsyncAdicionarFavorito class

## References

[1] L. Motion, "LeapMotionCoreAssers," GitHub, [Online]. Available: https://github.com/leapmotion/LeapMotionCoreAssets. [Accessed July 2017].

# Appendix D – Example of Patient Related



LeaPhysio

## Relatório Paciente

### Parte I: Informações Gerais

Data do Relatório: 2017-10-21 22:08:09

**Paciente:** Rui
**Fisioterapeuta:** Dr. LeaPhysio

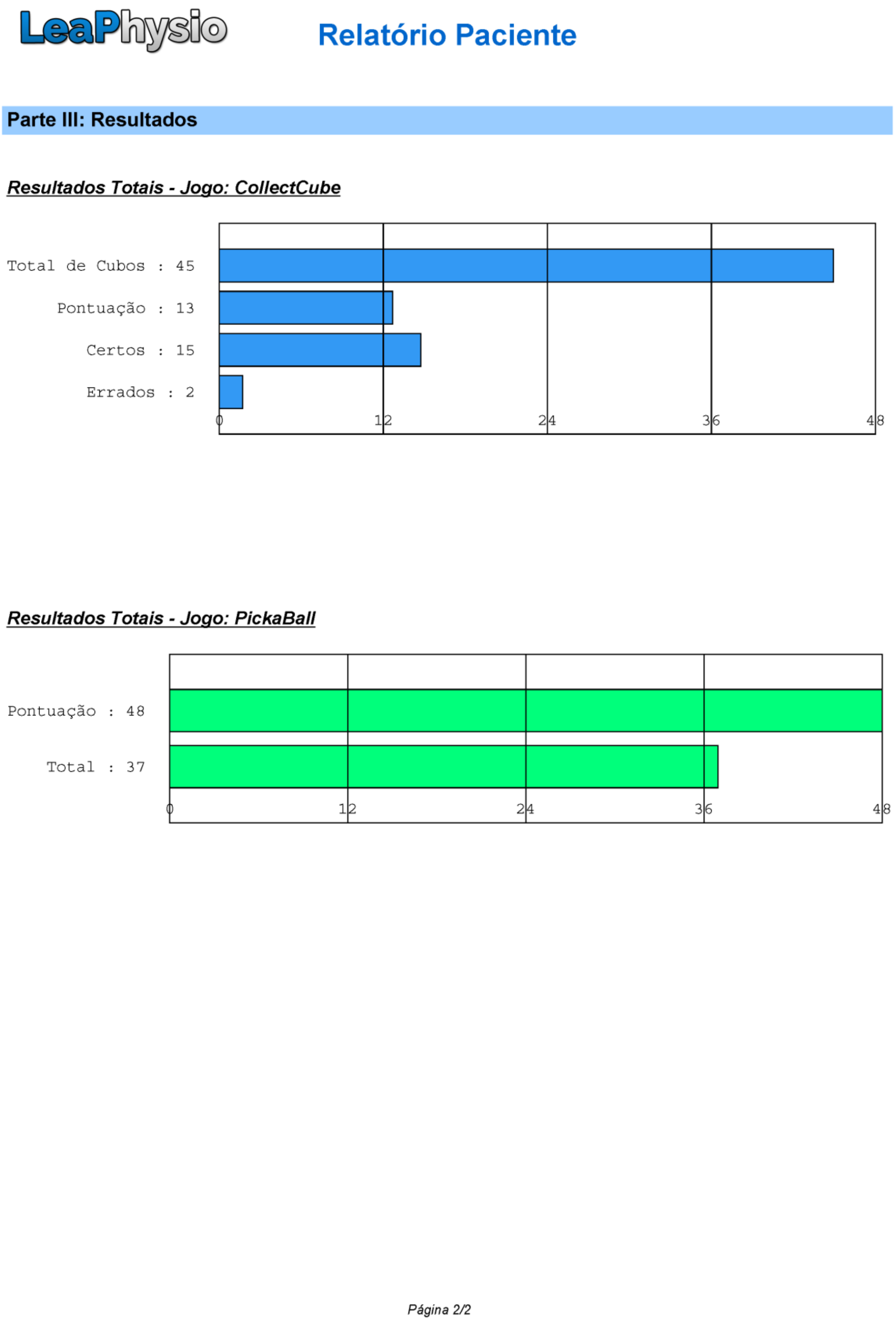
### Parte II: Treinos

Já realizou **51** exercícios de treinos e possuí actualmente **1** planos de treinos activos.
Existem **0** treinos com a mão esquerda, **2** com a direita e **0** com ambas.

**Treinos Activos**

**ID do Treinamento:** 6
**Data de Início:** 2017-10-21 **Data de Fim:** 2017-10-28
**Jogo:** PickaBall **Categoria:** FORMAS **Número de Repetições:** 1
**Duração:** 30 segundos **Tempo de Descanço:** 10 segundos
**Mão:** DIREITA **Dificuldade:** FACIL

*Página 1/2*



LeaPhysio

Relatório Paciente

**Parte III: Resultados**

***Resultados Totais - Jogo: CollectCube***

***Resultados Totais - Jogo: PickaBall***

*Página 2/2*