

Universidade do Minho
Escola de Engenharia
Departamento de Informática

Ana Margarida Reis Maia Campos

A software to manage rehabilitation sessions with a robotic walker

October 2022



Universidade do Minho
Escola de Engenharia
Departamento de Informática

Ana Margarida Reis Maia Campos

A software to manage rehabilitation sessions with a robotic walker

Master dissertation
Integrated Master's in Informatics Engineering

Dissertation supervised by
Cristina Manuela Peixoto Santos
António Carlos da Silva Abelha

October 2022

COPYRIGHT AND TERMS OF USE FOR THIRD PARTY WORK

This dissertation reports on academic work that can be used by third parties as long as the internationally accepted standards and good practices are respected concerning copyright and related rights.

This work can thereafter be used under the terms established in the license below.

Readers needing authorization conditions not provided for in the indicated licensing should contact the author through the RepositóriUM of the University of Minho.

LICENSE GRANTED TO USERS OF THIS WORK:



CC BY-NC-ND

https://creativecommons.org/licenses/by-nc-nd/4.0/

ACKNOWLEDGEMENTS

A five-year marvellous journey ends today. I can describe it as an intense and challenging learning experience, which helped me to grow, gain knowledge, make incredible friends, and also have some fun on the way. It would not be possible without the support, help, caring and love of the amazing people who surround me during this life chapter.

First, I want to thank my mom, Isabel, and dad, Paulo, for always giving me the opportunity to follow my dreams, and for giving me unconditional support and love during all these years. This academic journey was only possible because of you, and I am extremely thankful and grateful for that.

Then, I want to thank my sister, Isabel, and my boyfriend, Gonçalo, for all their emotional strength, patience and love, and for never letting me give up.

To the rest of my family members, grandparents, uncles and cousins, thank you for all of your support during these years, with a special grating to my cousin Antero who helped me with his knowledge during this dissertation.

I am grateful for all the colleagues and friends i made during this journey. I will forever remember our study afternoons, and our funny nights.

I gratefully acknowledge the assistance and help of my advisors, Professors Cristina Santos and António Abelha. Thank you both for all the helpful guidance and motivation during the development of this dissertation.

A special thanks to João Lopes, who captivated me in choosing this theme. Thank you for your availability, guidance and patience.

Last but not least, I would like to extend my sincere thanks to the health professionals from the hospital of Braga who showed availability and agreed to participate in this project, giving suggestions and constant feedback.

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity.

I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

ABSTRACT

Cerebellar ataxia arises from damage or dysfunction that affects the cerebellum and its pathways. As a result, the motor abilities of individuals with this condition become weakened. Robotics-assisted therapy is still an emerging area, but it has several advantages that could boost the rehabilitation of these individuals. Considering this problematic, WALKit Smart Walker is being developed. Its main purpose is to improve the treatment of ataxic patients through intelligent and multidisciplinary rehabilitation sessions. Thus, it is equipped with several sensors that provide monitoring capabilities through a continuous evaluation of the end-user gait and posture.

A vast amount of data is acquired during each session by the walker sensors. For health professionals to analyse this data and have feedback on the patient's status throughout therapy, tools are needed to control, manage, and monitor sessions in a clear, practical and intuitive way. Therefore, the main goal of this dissertation is centred on implementing an effective way to store the acquired data, along with the development of software that satisfies these requirements.

To address these goals, a polyglot persistence database system, composed of a relational and a non-relational database, was implemented to store the required data while maintaining efficiency. Furthermore, a web application was developed to provide, not only to health professionals, but also to patients themselves, the management of the rehabilitation sessions with the walker. The application provides an individual and temporal analysis of the sessions through interactive graphics adapted to each patient. Additionally, it allows the management of the several patients who are/were in treatment and the addition of clinical ratting scales, which are useful to assess their motor condition and adapt therapies as needed. In this way, professionals can have a better perception of the patient's condition, and can show patients their evolution, possibly contributing to increase their motivation in therapy.

Moreover, in the context of this dissertation, the embedded software of WALKit SmartW, which allows the therapy configuration, was optimized. This software had no security mechanisms, thus the main goal was on the implementation of techniques capable of making the software secure. Additionally, other functionalities such as feedback alerts, were added to the existing application.

Throughout the development of this project, it was possible to have continuous feedback from health professionals of the Hospital of Braga. Usability tests and questionnaires were also applied, and the results were very promising, enhancing the need for a system with these characteristics. Professionals claimed the system may help in analysing the patient clinical status in an intuitive form while keeping them motivated during treatments.

KEYWORDS software, database, monitoring, smart walker, rehabilitation sessions.

RESUMO

A ataxia cerebelar surge a partir de danos ou disfunções que afetam o cerebelo e as suas vias. Como resultado, as capacidades motoras dos indivíduos que possuem esta condição ficam fragilizadas. A terapia assistida por robôs é ainda uma área em desenvolvimento, no entanto apresenta diversas vantagens que poderão agilizar os tratamentos destes indivíduos. Atendendo a esta problemática, o *WALKit SmartW* encontra-se a ser desenvolvido. O seu principal propósito é auxiliar os tratamentos de pacientes ataxicos através de sessões de reabilitação inteligentes e multidisciplinares. Para tal, é composto por um conjunto de sensores que fornecem uma monitorização e avaliação contínua da marcha e da postura do utilizador.

Uma grande quantidade de dados é adquirida ao longo de cada sessão através dos sensores. De forma a que os profissionais de saúde analisem estes dados e tenham *feedback* do estado do paciente ao longo da terapia, são necessárias ferramentas que permitam controlar, gerir e monitorizar as sessões, de forma clara, prática e intuitiva. O principal objetivo desta dissertação centra-se na implementação de uma estratégia eficiente para armazenar os dados, juntamente com o desenvolvimento de um software que satisfaça estes requisitos.

Para cumprir estes objetivos, um sistema de base de dados com persistência poliglota, composto por uma base de dados relacional e uma não relacional, foi implementado para armazenar os dados mantendo a eficiência. Além disso, uma aplicação *web* foi desenvolvida para proporcionar, não só aos profissionais de saúde, como também aos próprios pacientes, a gestão das sessões de reabilitação com o andarilho. A aplicação disponibiliza uma análise individual e temporal das sessões através de gráficos interativos adaptados a cada paciente. Adicionalmente, possibilita também a gestão dos diversos pacientes que estão/estiveram em tratamento, e a adição de escalas de classificação clínica, que são úteis para avaliar a condição motora e adaptar as terapias conforme necessário. Desta forma, os profissionais conseguem ter uma melhor perceção acerca do estado do paciente, e os pacientes podem ver a sua evolução, contribuindo para aumentar a motivação na terapia.

Ainda no contexto desta dissertação, otimizou-se a aplicação embebida no software do andarilho *WALKit*, que permite as configurações da terapia. O software era isento de qualquer mecanismo de segurança, pelo que o maior foco centrou-se na aplicação de técnicas capazes de o tornar seguro. Adicionalmente, outras funcionalidades, como alertas e configurações de algoritmos, foram adicionadas à aplicação existente.

Ao longo do desenvolvimento deste projeto, foi possível obter o *feedback* contínuo de profissionais de saúde do Hospital de Braga. Testes e questionários de usabilidade foram também aplicados, e os resusltados foram bastante promissores, reforçando a necessidade de um sistema com estas características. Os profissionais afirmaram que o sistema irá ajudar a analisar o estado do paciente de forma intuitiva, mantendo-o motivado durante os tratamentos.

PALAVRAS-CHAVE software, base de dados, monitorização, andarilho inteligente, sessões de reabilitação.

CONTENTS

Contents . . . iii

Acronyms . . . x

1 INTRODUCTION . . . 3

1.1 Motivation and Problem Statement . . . 3

1.2 Objectives . . . 4

1.3 Research Questions . . . 5

1.4 Contributions . . . 6

1.5 Dissertation Outline . . . 6

2 STATE OF THE ART . . . 8

2.1 Introduction . . . 8

2.2 Search Methodology . . . 8

2.3 Selection Strategy . . . 9

2.4 Data Extraction . . . 9

2.5 Results . . . 9

2.5.1 Databases . . . 9

2.5.2 Security and Privacy in Health Sector . . . 15

2.5.3 Software Development Life Cycle . . . 19

2.5.4 Existing Software For Patient Monitoring . . . 20

2.5.5 General Conclusions . . . 26

3 SYSTEM SPECIFICATIONS . . . 27

3.1 Introduction . . . 27

3.2 WALKit Smart Walker . . . 27

3.3 SDLC Model . . . 29

3.4 Requirements . . . 30

3.4.1 Requirement Gathering Methods . . . 30

3.4.2 Functional Requirements . . . 31

3.4.3 Non-functional Requirements . . . 32

3.5 Database Selection . . . 33

3.5.1 Relational Database . . . 34
3.5.2 Non-Relational Database . . . 35
3.6 System Global Structure . . . 37
3.7 General Conclusions . . . 38

4 WALKIT APPLICATION . . . 39
4.1 Introduction . . . 39
4.2 Design . . . 39
4.2.1 Wireframes . . . 39
4.3 Implementation . . . 40
4.3.1 Technologies . . . 40
4.3.2 Security Mechanisms . . . 41
4.3.3 Application Modifications . . . 42
4.4 Application Validation And Testing . . . 44
4.4.1 Database Storage . . . 44
4.4.2 Usability Testing . . . 45
4.4.3 Results . . . 48
4.5 General Conclusions . . . 50

5 WEB APPLICATION . . . 51
5.1 Introduction . . . 51
5.2 Application Overview . . . 51
5.2.1 Use Cases . . . 51
5.3 Design . . . 52
5.3.1 Wireframes . . . 52
5.3.2 Prototypes . . . 58
5.4 Evaluation . . . 62
5.4.1 Participants . . . 62
5.4.2 Methodology . . . 63
5.4.3 Results . . . 64
5.5 Implementation . . . 65
5.5.1 Technologies . . . 65
5.5.2 Application Architecture . . . 66
5.5.3 Backend . . . 66
5.5.4 Frontend . . . 67
5.6 Deployment . . . 77
5.7 Final Application Validation And Testing . . . 79

5.7.1 Participants . . . 79
5.7.2 Methodology . . . 79
5.7.3 Results . . . 80
5.8 General Conclusions . . . 82

6 CONCLUSIONS . . . 83
6.1 Future Work . . . 85

A REQUIREMENTS SURVEY QUESTIONNAIRE . . . 92

B WEB APPLICATION WIREFRAMES . . . 93
b.1 Patient Wireframes . . . 93

C WEB APPLICATION PROTOTYPES . . . 98
c.1 Administrator and Health Professional Prototypes . . . 98

D WEB APPLICATION RESULTS . . . 109
d.1 Patient Interfaces . . . 109

LIST OF FIGURES

Figure 1 Key-Value stores Database. Retrieved from [20]. 11
Figure 2 Column-oriented Database. Retrieved from [19]. 12
Figure 3 Document-oriented Database. Retrieved from [19]. 12
Figure 4 Graph-oriented Database. Retrieved from [20]. 13
Figure 5 Encrypted and decrypted image using AES. Retrieved from [32]. 17
Figure 6 (a) Web application interface from [44]. (b) Mobile and web application interfaces from [45]. 21
Figure 7 (a) Block Diagram with the temperature and heart rate sensors [46]. (b) Web application interface from [46]. 21
Figure 8 (a) Telerehabilitation device from [47]. (b) Real time control interface from [47]. (c) Clinician software during patient training [48]. 22
Figure 9 (a) ASBGo software application interface from [7]. (b) Mobile app (left) and Web app (right) interfaces from [49]. (c) Mobile app interface from [5]. (d) Android OS software interface from [50]. 23
Figure 10 WALKit Smart Walker and its respective sensors. 28
Figure 11 SDLC Models: (a) Waterfall Model. Based on [38]; (b) Prototyping Model. Based on [51]. 29
Figure 12 Demographic information from the requirements gathering participants: (a) Age Distribution; (b) Gender Distribution; (c) Medical Ocupation; (d) Electronic Devices Usage Distribution; (e) Usage of Technologies at Work; (f) Internet Connection. 31
Figure 13 Database Conceptual Model. 35
Figure 14 System Global Structure. 38
Figure 15 (a) Login page wireframe. (b) Sessions' information page wireframe. 40
Figure 16 Alerts Wireframes. 40
Figure 17 WALKit application login system. 41
Figure 18 CBC Mode of Operation. Retrieved from [55]. IV stands for Initialization Vector; m1,m2,m3 are the plaintext blocks; F corresponds to the block cipher; and c1,c2,c3 correspond to the ciphertext blocks. 42
Figure 19 Pop-up alerts: (a) Information Alert; (b) Error Alert. 42
Figure 20 Search Patient Menu. 43
Figure 21 Insert Patient Menu. 43
Figure 22 Sessions' Information Menu. 44
Figure 23 Grading Scale for the SUS. Retrieved from [61]. 50

Figure 24 New Application Global Use Case Diagram. 52
Figure 25 Login Page wireframe. 53
Figure 26 Home Page Wireframe. 53
Figure 27 Profile Page Wireframe. 54
Figure 28 Patients' Management Wireframes: (a) Archive Management; (b) Management of patients' undergoing therapy. 54
Figure 28 Patients' Management Wireframes: (a) Archive Management; (b) Management of patients' undergoing therapy (continued). 55
Figure 29 Patient details wireframes: (a) Patient' personal and clinical details; (b) Patient' sessions. 56
Figure 30 Sessions wireframes: (a) Session details; (b) Temporal evolution. 57
Figure 31 User Management Wireframe. 58
Figure 32 Prototypes - Patient details. 59
Figure 33 Prototypes - Temporal Evolution. 60
Figure 34 Prototypes - Individual Session Details. 62
Figure 35 Web Application Architecture. 66
Figure 36 Servers Structures: (a) Structure of the server which is connected with the MySQL database; (b) Structure of the server which is connected with the MongoDB database. 67
Figure 37 Login Menu. 68
Figure 38 Administrator Homepage. 69
Figure 39 Personal Profile Menu. 69
Figure 40 Patients Management Menu. 70
Figure 41 Insert Patient Modal. 70
Figure 42 Patient Details. 71
Figure 43 (a) TUG clinical scale modal form. (b) Fall record modal form. 72
Figure 44 Sessions menu. 72
Figure 45 Session details menu. 74
Figure 46 Temporal evolution menu. 75
Figure 47 Patients' archive menu. 76
Figure 48 User Management menu. 76
Figure 49 Pop-up Alerts. 77

LIST OF TABLES

Table 1 SQL and NoSQL features comparisons. Based on [11][14][20] 14
Table 2 Main security and privacy requirements of GDPR. Based on [8][25–27] . . . 16
Table 3 SDLC Models Comparative Analysis. Based on [38][42] 19
Table 4 Monitoring systems overview 24
Table 5 WALKit SmartW sensors' information 28
Table 6 Database Testing Results 45
Table 7 Thinking-Aloud Protocol Results 48
Table 7 Thinking-Aloud Protocol Results (continued) 49
Table 8 SUS Scores WALKit SmartW application 49
Table 9 Usability Test Results 64
Table 10 SUS Scores Prototypes 65
Table 11 Usability Test Results 81
Table 12 SUS Scores Final Application 82

LIST OF LISTINGS

3.1 MongoDB document with sensors data. 36
5.1 Docker compose file. 77

ACRONYMS

3NF Third Normal Form. 10

ABC Activities-specific Balance Confidence. 71
ACID Atomicity, Consistency, Isolation, Durability. 10, 11, 13, 14, 34, 84
AES Advanced Encryption Standard. 16, 18, 24, 41, 42, 67, 84

BASE Basic Availability, Soft State, Eventual Consistency. 13
BBS Berg Balance Scale. 4, 71

CAP Consistency, Availability, Partition Tolerance. 13
CBC Cipher Block Chaining. 42, 67, 84
CIA Confidentiality, Integrity, and availability. 15
CRUD Create, Read, Update, Delete. 4, 5, 17, 39

DBMS Database Management System. 9, 10, 14, 28, 39

GDPR General Data Protection Regulation. 8, 9, 15, 26, 34, 41

Http Hypertext Transfer Protocol. 18, 37
Https Hypertext Transfer Protocol Secure. 18, 26, 85

ICARS International Cooperative Ataxia Rating Scale. 71

JWT JSON Web Token. 67

MFA Multi-Factor Authentication. 17, 41

NoSQL Not-only-SQL. 10, 11, 13, 14, 26

OTP One Time Password. 18

PID Proportional-Integral-Derivative. 27

RAGT Robotic Assistant Gait Training. 3
RDBMS Relational Database Management System. 10, 34
ROS Robot Operating System. 4, 27, 40
RSA Rivest-Shamir-Adleman. 16
RTOS Real-Time Operating System. 27

SARA Scale for the Assessment and Rating of Ataxia. 4, 71
SDLC Software Development Life Cycle. 19, 20, 26, 29, 30, 52, 62
SFA Single-Factor Authentication. 17, 41
SQL Structured Query Language. 10, 11, 26, 69, 84
SUS Software Usability Scale. 46, 49, 63, 64, 79, 81, 83

TFA Two-Factor Authentication. 17, 18, 41
TLS Transport Layer Security. 18
TUG Time Up and Go. 4, 71, 72

VPN Virtual Private Network. 18

1

INTRODUCTION

This dissertation presents the work developed in the scope of the fifth year of the Integrated Master in Informatics Engineering. This work was developed at the Biomedical Robotic Devices Laboratory of the Center for MicroElectroMechanical Systems (CMEMS), from the University of Minho, Portugal.

This dissertation main goal was to propose and implement an efficient strategy to store data that is acquired during the use of WALKit Smart Walker, a personalised robotic walker for the rehabilitation of mobility disorders, as well as to create an user-friendly and intuitive software to monitor and manage acquired data. This software will allow health professionals to analyse, interpret, and manipulate the acquired data from the rehabilitation sessions, allowing not only to visualise the patient's evolution throughout time, but to insert and manage their clinical information. At the same time, patients will also be able to visualise their progress, contributing to increase their motivation in therapy and possibly improving their recovery.

In this first chapter the motivation and problem statement are presented, followed by the outlined goals, research questions and, finally, the document structure.

1.1 MOTIVATION AND PROBLEM STATEMENT

Cerebellar ataxia arises from damage or dysfunction that affects the cerebellum and its pathways. As a result, functional difficulties in the upper and lower limbs, lack of balance and irregular gait may appear [1]. According to [2], a possible and suitable solution to help regress this neurological condition is through well-organized rehabilitation programs. Robotic Assistant Gait Training (RAGT) [3] as a facility to treat ataxia is still limited. However, the use of RAGT in rehabilitation programs of people with ataxia can bring some advantages, such as more reproducible symmetrical gait patterns, which are very affected by ataxia given their poor motor coordination, operation by a single therapist, and decrease of the therapists' energy expenditure [3].

WALKit Smart Walker is a **robotic walker** proposed to be used in **rehabilitation sessions** of people with balance disorders, particularly **cerebellar ataxia**. This device integrates several sensors, namely RGB-D cameras, load cells, inertial and infrared sensors, which constantly analyse the end-user [4]. These **sensors** generate a **vast amount of data** in each rehabilitation session. **Healthcare professionals** must then have access to these data, in order to examine, and **conclude about the patient's condition and progress.**

Given the **large amount of data** that is created, it is essential to **store it correctly and efficiently**, always ensuring validity and preventing data loss. Data access and management must be straightforward, intuitive, and there should be no need in resorting to files system to store and handle data, as it is performed currently.

Therefore, there is a **need for a system** that can **store and display data from the smart walker's sensors** in an **efficient** and **intuitive way**, allowing healthcare professionals to manage rehabilitation sessions from any location and from different devices, such as smartphones, computers and tablets. Apart from accessing clinical and sessions' data, health professionals should also be able to manage them, *i.e.*, perform Create, Read, Update, Delete (CRUD) operations. According to [5], software for managing data acquired during these sessions, may **contribute to provide more suitable rehabilitation strategies** by health professionals, allowing them to have a **better** and an **objective evaluation** of the patients' progress throughout therapy. Clinical rating scales (*e.g.* Scale for the Assessment and Rating of Ataxia (SARA), Berg Balance Scale (BBS), Time Up and Go (TUG)) [6] also play an important role in assessing the patients' condition, thus being a relevant feature to include in the system. A **graphical interface** with patient evolution diagrams could **facilitate** healthcare professionals in **analyzing the patient's status** and, simultaneously, motivate patients during therapy [7].

WALKit SmartW presents an **embedded application** in its software, which is based on Robot Operating System (ROS), and allows to make **session's configurations** according to the patient needs. Besides from a biofeedback system, it allows performing operations associated with patients, manage therapy settings and enter several clinical scales. Nevertheless, **no specific techniques are applied to guarantee the security and anonymity of the data**. Data is stored under fictitious names, and only researchers and clinical team members know their association with real patients. Therefore, there is a **need in optimizing this application** by implementing **security techniques** to avoid problems with malicious users.

To assess the user evolution, there was already software for managing and visualizing data acquired during the use of the robotic walker [7], but these are incompatible with the current architecture of the device. Furthermore, its use implies the use of a computer, not being possible to use it with other devices, namely smartphones or tablets (Android and iOS).

Despite all benefits that a healthcare application may bring, **law and security concerns must always be a concern**. It is essential to have extra care in **protecting patients' privacy** since it can contain highly personal information and its inappropriate security may cause poor trust by its users [8]. Considering this, this proposal also aims to address this challenge by implementing security mechanisms so that data remains secure.

1.2 OBJECTIVES

Considering what was stated in the previous section, the main goals of this dissertation were to **(i)** propose and validate an efficient strategy to store not only data acquired during a rehabilitation session with WALKit SmartW, but also relevant clinical information of patients, ensuring security and the ethical standards applied to sensitive data; **(ii)** optimize and validate the WALKit SmartW application; and **(iii)** develop and validate a responsive and user-friendly web application that can let health professionals to assess and manage all data and to visualize the evolution of their patients. To achieve these goals, the following objectives were outlined:

- **Objective 1:** Identify existing methods in the literature for **(i)** data storage, **(ii)** encryption methods to ensure data security and other security measures that applications must have, **(iii)** conduct a literature review on software development life cycle to reach the best models to guide the implementation, and **(iv)** to perform a literature search on health monitoring software, considering their functionalities and graphical interfaces. This objective is addressed in Chapter 2.
- **Objective 2:** Identify technical and clinical requirements. These requirements were collected with the help of health professionals from the Hospital of Braga, who usually deal with patients undergoing gait and balance rehabilitation and who has previously worked with the robotic walker. This objective will be addressed in Chapter 3.
- **Objective 3:** Idealize and build a database schema to allow efficient storage of health records, namely the data acquired from each rehabilitation session and data related to patients and health professionals. Create a secure software to manage these data (CRUD operations), capable of providing an intuitive way to visualize historical rehabilitation sessions, as well as temporal evolution. This objective will be addressed in Chapter 3 and Chapter 5.
- **Objective 4:** Optimize the SmartW embedded software by applying security mechanisms and implementing other functionalities requested by health professionals. This objective will be addressed in Chapter 4.
- **Objective 5:** To perform the validation of the resulting applications using standard usability testing methods of literature. These tests were applied to future users of the applications, namely physiotherapists and physiatrists from the Hospital of Braga. This objective will be addressed in Chapter 4 regarding the existing application, and in Chapter 5 for the new application.

1.3 RESEARCH QUESTIONS

Considering the ultimate goals of this dissertation and the proposed objectives, namely the literature review, the deployment of the database to store the acquired data in an efficient and secure way, and the development of the software for its management and visualisation, relevant research questions were identified, as follows:

- **RQ 1:** *What are the best ways to store data relative to patients and/ or health professionals and data from the rehabilitation sessions?* This research question is related to Objective 1 and 3.
- **RQ 2:** *How can the walker data be stored efficiently while maintaining security and user's privacy?* This research question is related to Objective 2 and 3.
- **RQ 3:** *How can data collected by the walker be accessed on several electronic devices?* This research question is related to Objective 3.
- **RQ 4:** *How is the software created useful in managing the data acquired in a session with the robotic walker?* This research question is related to Objective 5.

1.4 CONTRIBUTIONS

The main contributions of this dissertation are:

- A literature review of relational and non-relational databases and their contribution in the healthcare sector;
- A literature review of the security and privacy mechanisms that applications must have to protect the users' data;
- A literature review of existing patient monitoring applications;
- A solution for the efficient storage of clinical and time series data;
- A web application for managing rehabilitation sessions with WALKit SmartW.

1.5 DISSERTATION OUTLINE

This document is organised into six chapters. A brief description of what is covered in each chapter is provided below:

- **Chapter 1** is introductory and is focused on providing a contextualization, motivation and problem statement of the project that was developed in the scope of this dissertation. The research questions are also approached, along with the primary goals to be achieved in the dissertation. Ultimately, the document structure is presented in detail.
- **Chapter 2** addresses the scientific and theoretical concepts that lay the foundations for the implementation of the several goals proposed in this dissertation. A literature review about databases is conducted in this chapter, along with the presentation of security mechanisms to follow ethical standard on sensitive personal data. Since software development is a relevant topic in this dissertation, this chapter also includes literature related to the development life cycle. Additionally, software from the healthcare sector was identified in literature, with particular emphasis on robotic walkers.
- **Chapter 3** specifies the planning phase that was followed for the optimization and development of the WALKit SmartW and web applications. This chapter starts by introducing the robotic walker, followed by the presentation of the development strategy, requirements gathering, database choice and a final analysis of the global system.
- **Chapter 4** introduces the WALKit SmartW embedded software, along with the steps followed for its optimization considering clinical and technical requirements. The software's new features are presented in detail, as well as the security mechanisms applied in order to keep it secure. Furthermore, the final application was validated and tested by health professionals, and the results are also presented in this chapter.

- **Chapter 5** introduces the development of the web application to manage the rehabilitation sessions with the WALKit SmartW. The design, implementation, deployment, and testing phases are described methodically. The resulting application and its impact on the hospital environment are also addressed.
- **Chapter 6** concludes the dissertation, summarising the work developed throughout this dissertation and by answering the outlined research questions and presenting suggestions for future work.

2

STATE OF THE ART

2.1 INTRODUCTION

The focus of this chapter is centred on a detailed literature review that supports the main goals of this dissertation. First, a detail description of how this research was organized and implemented is given. Then, the investigation results are presented and divided in four different subsections. The first one covers an important topic in any software responsible for storing and handling data, the Database (section 2.5.1). Here, the two best-known types of databases, relational and non-relational, are discussed, as well as their advantages, disadvantages, and best use. The second one (section 2.5.2), considers data security and privacy. In this sense, this section focuses, namely, on the General Data Protection Regulation (GDPR) and the security mechanisms necessary to develop a secure and reliable application. The third one (section 2.5.3) describes the different phases that should be followed in software development to enhance systems' quality. Lastly, section 2.5.4 exploits different software for health monitoring that were presented in literature, with special emphasis on robotic walkers. The different functionalities that the software presents are analysed, as well as the graphical design. At the end, a concluding section is presented (section 2.6) where a general overview of the concepts covered throughout this chapter is performed.

2.2 SEARCH METHODOLOGY

Considering the main goals of this dissertation, this research was divided in four different groups: **(i)** Databases, **(ii)** Security and Privacy, **(iii)** Software Development process, and **(iv)** Software for health monitoring. Considering these groups, and in order to facilitate the following literature, some keywords and keywords combinations were used, namely: "Databases" AND "SQL" AND "NoSQL"; "health" AND "SQL" AND "NoSQL"; "Big Data"; "software" AND "smart walker"; "health" AND "applications"; "robotic devices" AND "medical rehabilitation"; "privacy" AND "security"; "encryption"; "SDLC"; "software development"; "usability testing"; "usability" AND "testing methods"; and "web applications" AND "security". The research was conducted in the IEEE Xplore and Scopus databases.

2.3 SELECTION STRATEGY

The article selection process was based on the following inclusion criteria: **(i)** contain keywords in the title or abstract, **(ii)** allow access to the full document, **(iii)** publication date after 2010, **(iv)** written in English or Portuguese, **(v)** aim of improving the storage of patients/health professionals' data, **(vi)** efficient Big Data storage, **(vii)** description of software development process, **(viii)** usage of software in monitoring patients' health, and **(ix)** improving security in software applications.

2.4 DATA EXTRACTION

Considering the four groups presented above, the information of each article in that group was analysed by the following criteria:

- Databases - **(i)** type of database, **(ii)** advantages/ disadvantages, **(iii)** database features, **(iv)** stored data, **(v)** experimental results, and **(vi)** general conclusions;
- Security and Privacy - **(i)** principles of GDPR, **(ii)** data protection techniques, **(iii)** security and privacy mechanisms, **(iv)** encryption methods, **(v)** experimental results, and **(vi)** general conclusions;
- Software Development Life Cycle - **(i)** development phases, **(ii)** models, **(iii)** models features, **(iv)** experimental results, and **(v)** general conclusions;
- Software for monitoring patients - **(i)** software purpose, **(ii)** robotic device, **(iii)** communication with the robotic device, **(iv)** used technologies, **(v)** functionalities, **(vi)** experimental results, and **(vii)** general conclusions;

2.5 RESULTS

According to the research process described above, forty-three articles were selected to guide the present state of the art. The results obtained are introduced throughout the following sections.

2.5.1 *Databases*

The need to store data was and continues to be a major challenge. Previously, this challenge was mostly achieved through paper records due to its facilities to read and write information. Some disadvantages of this method centred on the fact that they could be easily damaged or even lost, hard to organize and insecure. However, with the constant evolution of technology, storing data has become easier and more secure, managing to overcome the disadvantages of the old methods. Nowadays, a modern way to store data is through a database. A database "*refers to a collection of data that is managed by a DBMS*" [9]. According to [10], a Database Management System (DBMS) is a software tool that allows users to store, change, and recover data. These

systems are essential and highly effective in managing and handling several sorts of data, allowing users to complete many tasks easily [11]. They are also known for replacing the file system, overcoming several limitations such as redundancy, inconsistency, and simultaneous access by multiple users [12]. Currently, there are three classes of database systems. The first and well-known class is the Relational Database Management System (RDBMS), usually known as Structured Query Language (SQL). In this type of database, data is stored in the traditional form of rows and columns, showing to be efficient for storing structured data. However, as the volume of data increases, these databases become unsuitable for storing such data, not even unstructured data. In this way, the two other classes are presented: non-relational databases and, more recently, new relational databases, popularly known as Not-only-SQL (NoSQL) and *NewSQL*, respectively. The first one, NoSQL, is more traditional and therefore more used. In contrast with the relational ones, it is schema-free and allows handling a variety of data. The second one, *NewSQL*, appeared more recently and is characterized by retaining the relational aspect of the traditional RDBMS while at the same time incorporating solutions provided by NoSQL databases [10]. In the following subsections, relational and non-relational databases will be discussed in more detail.

Relational Databases

Traditional DBMS for storage and handling data have been based on the relational model [13]. An important aspect of this model is that it follows the Atomicity, Consistency, Isolation, Durability (ACID) property. The authors of [9] and [14], clarify each one of the assets presented in ACID. As stated by them,

- *Atomicity* is interconnected with the state of the transaction, i.e. or the whole transaction execution occurs or does not occur at all, there is no halfway;
- *Consistency* implies that the confidentiality requirements must be kept during the exchange;
- *Isolation* guarantee that multiple transactions can happen simultaneously without compromising performance;
- *Durability* ensures that, once the transaction has been completed, all database updates and adjustments are made, even if there is a framework failure.

These four properties ensure the confidentiality and integrity of data in the database. Even if some failures occur, the validity of data is ensured.

As previously stated, relational databases are widely known as SQL, which is also the language that these databases are queried by. Therefore, a standard language is used to perform queries, allowing the manipulation of data to be done equally in all relational databases. These databases have the particularity of being suitable for complex queries [12], as well as of providing secure mechanisms that protect data [11]. Another important feature is the prevention of data redundancy and data anomalies. This is guaranteed by the application of the Third Normal Form (3NF) [15]. Although this type of database has several significant peculiarities, it also presents some limitations. RDBMS uses vertical scalability, which means it is possible to increase the software capacity by adding resources until reaching the size limit of the server. Since the size of the server is a limitation, handling a huge volume of data becomes challenging, resulting in a need for servers with extraordinary computing

capabilities, which are very expensive. Another limitation is related to the fact that relational databases need a predefined schema and can only deal with structured data, making the complexity very high and providing low flexibility [12].

Relational Databases in Health Sector

Several healthcare systems make use of a relational databases to store, organize, and process data. An example is presented in study [16], where a polyglot-persistent application resort to *PostgreSQL* to store structured data, hospital payroll and active billing data. The authors justify their choice by stating that relational databases follow the ACID properties, and allow information to remain confidential and secure [16]. Other systems that take advantages of SQL databases are presented in studies [17] and in [18]. Both are software applications whose purpose is to record and manage medical structured data, and both resort to *MySQL* database. The application of the first one (study [17]), entitled as *Health Information System for Home Visits*, has the main goal of handling information from a home visit, *i.e.* when healthcare professionals go to a patient home. The stored data includes namely: **(i)** data from the user; **(ii)** details from the home visit; **(iii)** patient information; **(iv)** chronic disease, and **(v)** appointment information. The application of the second one (study [18]), *Online Health Care*, allows patients to store their own medical data, and contact clinicians for appointments, free medical advice or prescriptions. Similarly to the other presented systems, the stored information consists of personal patient and clinician information, as well as medical prescriptions and appointments' information.

Non-relational Databases

Non-relational databases appear as a complement to relational ones, overcoming some limitations. The idea is not to replace them as both technologies can coexist [19].

Over the last few years, this type of database has dramatically grown in popularity due to its ability to handle large amounts of structured, semi-structured and unstructured data while keeping the level of performance high [13]. There are four different categories of NoSQL databases:

- ***Key-Value stores:*** According to study [12], this type of database is similar to a hash table. As the name implies, unique keys point to their corresponding data. Some of the principal features are simple schema, flexibility, availability, and high performance [12][20]. Figure 1 displays an example of a key-value store database.

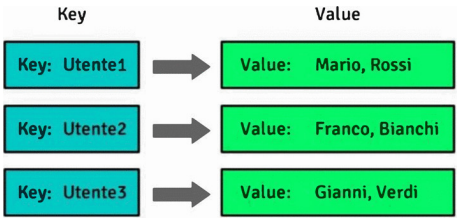


Figure 1: Key-Value stores Database. Retrieved from [20].

- ***Column-oriented:*** In this kind of database, data are stored by column rather than by row [12]. There are column families which can contain several columns containing different types of data. Usually, in a column family, columns have identical features and are rationally connected, becoming similar to a relational database [20]. These databases have multiple features, such as accessibility, faster insertion/reading of data, capacity to deal with enormous amounts of data, and an easy schema [19][20]. Figure 2 shows an example of a column-oriented database.

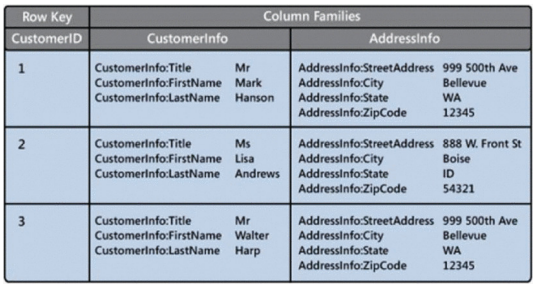


Figure 2: Column-oriented Database. Retrieved from [19].

- ***Document-oriented:*** As the name suggests, databases of this kind store and manage data in documents. It is considered an excellent system to store unstructured or semi-structured data, usually in the key-value format using JSON documents [20]. It is a good option if the intention is to store and manage large amounts of data (Big Data) and if data query is a requirement [19]. Figure 3 displays an example of a document-oriented database.

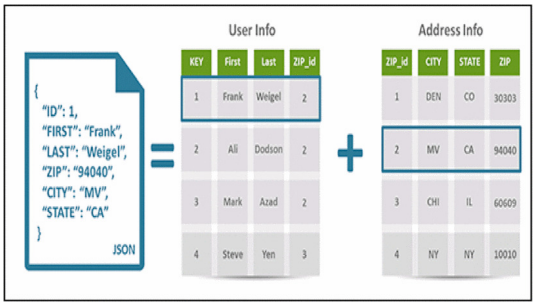


Figure 3: Document-oriented Database. Retrieved from [19].

- ***Graph-oriented:*** Here, graphs are resorted to organise the data and their relationships [12]. Graph-oriented databases are normally used when the major interest is in the data relations rather than the original data [19][20]. These systems are not design to query data [20]. Scalability and schema free are particular features of this type of database [19]. Figure 4 show an example of a graph-oriented database.

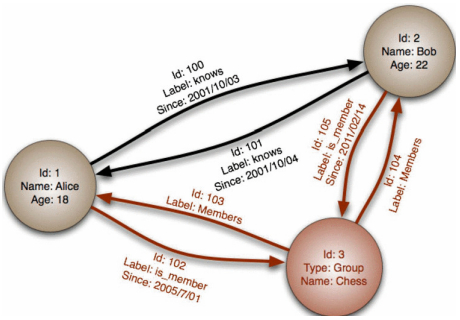


Figure 4: Graph-oriented Database. Retrieved from [20].

In contrast with relational databases, NoSQL database do not follow ACID properties. Instead, they can follow the Consistency, Availability, Partition Tolerance (CAP) theorem [12][14] and the Basic Availability, Soft State, Eventual Consistency (BASE) properties [12][20].

NoSQL databases are identified as a viable alternative to relational databases as more organisations recognise that their schema-less data model is a better method for handling large volumes of data [19]. However, the lack of data integrity and the time consumption on complex operations are two of the main limitations of these databases. Furthermore, the lack of support and management tools become also a barrier [13][19].

Non-Relational Databases in Health Sector

In the existing literature, several systems in the health sector resort to the use of non-relational databases to store and handle data. Notably, these turn out to be the best option in health systems when the requirements include the storage of a large amount of data and dealing with diverse information that is loosely structured [16]. Examples are presented in studies [21] and [22], where huge amounts of data are generated from two distinct monitoring devices: the *Lokomat* exoskeleton for gait training, and a baby's multimodal healthcare predictive system, respectively. Both use a *document-oriented* database (*MongoDB*) to store and retrieve the sensors' data. The authors of [21] state that future hospital needs can be satisfied without elevated costs due to the advantages of the chosen database, such as the simplicity to scale up/down and adding or removing workloads. Moreover, a *document-oriented* database allows storing efficiently the monitoring indicators in files of different formats, and the performance of queries based on indexes or segments [22]. Additionally, in study [22], a *key-value store* database is also used to store personal and biometric data, such as name, gender, birth date, and DNA. According to the authors, the chosen database is considered a great choice for personal data since it will allow a flexible structure associated with the individual, and the performance of queries based on different fields. Another example is presented in study [16], where a *graph-oriented* database is resorted for storing information such as blood relations among patients, and links between symptoms. In this way, doctors can trace diseases more quickly. Ultimately, the storage of health images can bring significant benefits in the hospital context as they can help health professionals reach conclusions about the patient's diagnosis [23]. The authors of [16] and [23] point to the use of *document-oriented* databases, particularity *MongoDB* and *CouchDB*, to store and manage medical images, as well as patients and doctors images.

Comparison between Relational and Non-Relational Databases

Deciding which DBMS should be used to store and handle data is a relevant topic in any software application [11]. The system requirements must be analysed and discussed in order to choose the most appropriate database. According to study [11], choosing relational databases should be considered when:

- The system deals with sensitive data and the top priorities are to ensure data integrity and consistency;
- Data comes from multiple collections and the goal is to gather them all;
- ACID properties are necessary for the system to work properly.

On the other side, NoSQL databases should be a choice when:

- The system top priorities are availability and scalability;
- Data integrity or ACID properties are not required;
- Handling large amounts of data and unstructured data is a concern;
- There is a need to change the data model multiple times.

Table 1 presents several characteristics of relational and non-relational databases in a comparative way.

Table 1: SQL and NoSQL features comparisons. Based on [11][14][20]

Database	SQL	NoSQL
Schema	Static	Dynamic
Scalability	Vertical	Horizontal
Flexibility	Very complex if there is a need to change the schema	Excellent for environments that may evolve
Query-Language	Uses a standard query language and can handle complex queries	Does not have a standardized language to query and has bad performance in complex queries
Security	Have security mechanisms	Security may not be ensured because NoSQL databases can be distributed over many servers
Complexity	High complexity because follows table structure	Can support and handle structured, semi-structured and unstructured information
Mapping Relationships	Three types of data relationship: one-to-one, one-to-many and many-to-many	Correlations between data occur during data processing. NoSQL may not be suitable for complex relations
Big Data	Not good for managing big data due the table structure	Ideal to deal with large information
Data to be Stored	Not a good fit for hierarchical data storage	Good for leveled information storage

2.5.2 *Security and Privacy in Health Sector*

In the recent past, E-health has been a target of cyber-attacks [8]. These attacks may compromise any systems' security and privacy. Therefore, it is essential to build secure software, especially in the health sector since patient's data can contain highly personal information.

One of the most meaningful aspects of security focuses on the Confidentiality, Integrity, and availability (CIA) triad. These three properties are considered the foundation of data security and privacy [24]. In order to guarantee the CIA triad in any system, some security mechanisms should be applied throughout the software development process, considering, at the same time, privacy regulations. In Europe, personal sensitive data is protected by GDPR, which means that any system which processes data must follow all the clauses and requirements expressed in it. GDPR and several security mechanisms for applications will be analysed and discussed in more detail in the next subsections.

General Data Protection Regulation

The current and rapid expansion of technology, namely, the increasing use of the Internet, social networks and digital business models, requires a greater focus on protecting personal data [25] [26]. If, on the one hand, people are lured to share more personal information without considering side effects; on the other hand, organizations are increasingly collecting more information in order to provide better services to their customers [26]. In this sense, the GDPR was created, ensuring the right to data protection for all citizens across the European Union.

The GDPR consists of a set of principles and rules applied to the processing of personal data. It allows individuals to have a better control and a understanding about how their data is being used on the Internet [27].

According to study [25], personal data is described as information, directly or indirectly, related to a data subject, such as name, identification number, location and culture or social identity of a person. Some of these, particularly data related to the physical or mental health of a person [28], are considered sensitive personal data, meaning that it is necessary to have extra care when processing this kind of information.

Therefore, it is essential to ensure the integrity and confidentiality of personal data, especially for sensitive ones, and for that it is mandatory to follow the clauses presented in the regulation. In Table 2, a summary of the main security and privacy requirements of GDPR is provided.

Table 2: Main security and privacy requirements of GDPR. Based on [8][25–27]

Principle	Description
Explicit Consent	Terms and conditions must be presented in an easily accessible, understandable, and intelligible form, with the data processing purpose attached to that consent.
Data minimization	Only data that are strictly necessary for the purpose or the functionality of the system should be collected.
Transparency	Collected data must always be transparent for data subjects.
Privacy by design and by default	Data protection and privacy implications must be considered from the beginning of the systems' design until the development process is complete.
Data subject rights	Data subjects must be able to access, modify, and delete their data, as well as to request at any time what personal information the company is using, where and how it is being used.
Retention period	Data must not be stored longer than it is required for the functionality of the system.
Data breach protection	It is mandatory to notify the competent authorities and affected individuals within a maximum of 72 hours if any data breach occurs.
Right to be forgotten	Data subjects may request to have all their data deleted.
Security measures	Mechanisms to protect the confidentiality, availability, and integrity of data must be implemented.
Penalties	If organizations do not comply with the clauses expressed in the GDPR, they may be fined.
Data portability	Data subjects have the right to request data portability between systems.

Security Mechanisms and Web Applications

In this era of the digital revolution, web applications usage is increasing in several different areas, such as banking, healthcare and e-commerce [29]. With this rise, the amount of sensitive data also increases, making the security of web systems a challenge. As previously stated, it is essential to guarantee confidentiality, integrity and availability of data [24]. It is crucial to ensure those properties by applying the security mechanisms needed, preventing systems from malicious attacks.

Cryptography is the science of protecting information from malicious users by converting it into an indecipherable form while at rest and in transmission. Cryptography algorithms are composed of two methods: **(i)** encryption, responsible for making information unreadable, and **(ii)** decryption for the reverse [30]. With the appliance of encryption algorithms, the confidentiality of data is ensured, making it impossible for malicious intruders to reveal information [24]. According to study [30], these algorithms can be classified as symmetric (private key) or asymmetric (public key). The first uses the same key both to encrypt and to decrypt. The algorithm strength depends on the size of the chosen key. The principal limitation is that the key is shared between sender and receiver. *AES*, *DES*, *3DES* and *Blowfish* are included in the symmetric type. In contrast, the second one takes advantage of two keys: a public and a private key. The public is used for encryption, and the other is used for decryption, making it possible not sharing the keys. The major limitation is that these algorithms are 1000 times slower than symmetric ones. A well-know asymmetric algorithm is Rivest-Shamir-Adleman (RSA). Several studies [30–33] indicate that Advanced Encryption Standard (AES) is the best algorithm in terms of security and performance, either for textual, binary, or even image data (Figure 5). AES is a symmetric-key algorithm, and is

characterized for having a fixed block length of 128 bits and a key size of 128, 192 or 256 bits [31]. It is composed of four operations: Sub-Bytes, Shift-Rows, Mix-Columns, and Add Round Key, and depending on the length of the chosen key, a different number of iterations is performed [32]. At the end of all iterations, the result is a cryptogram only decipherable for those who have the key, keeping confidentiality high.

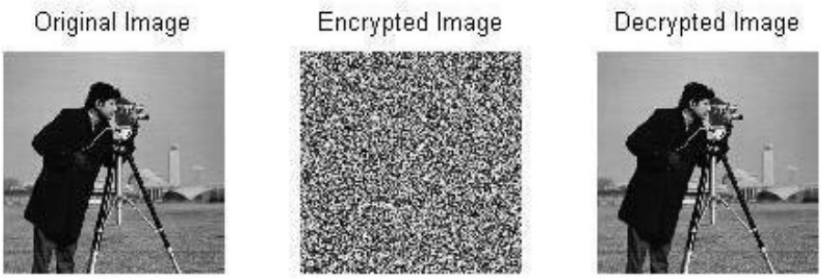


Figure 5: Encrypted and decrypted image using AES. Retrieved from [32].

A strong authentication process followed by an authorization one also plays a prominent role in protecting web applications from malicious intruders [34]. Authentication is described as the mechanism for user identification in a system before allowing them from accessing resources. Normally, this process is ensured by a login system with users' credentials. The authorization process is responsible for ensuring that only authorized users/roles have access to some or to all system resources [24]. According to the authors of [29], there are three types of authentication methods that should be considered relying on the system requirements. The first one is Single-Factor Authentication (SFA), also known as *Traditional Authentication*, and it only requires a username and password for logging into the application. In this case, users must have several precautions, particularly choosing a strong password. The second one is Two-Factor Authentication (TFA), where there are two forms of authentication. Typically, these two forms include a physical token, such as a smart card, and something memorized, for example, a PIN or a password. This type of authentication does not guarantee strong authentication nor prevent online fraud. Nevertheless, it can reduce fraud and attacks. Lastly, the Multi-Factor Authentication (MFA) include several methods for verifying users' identity, usually, something the user knows (password or pin), something the user has, such as a card, and something the user is (biometric characteristics). This method provides strong authentication, being very difficult to break.

Auditing and monitoring mechanisms may also be indispensable for tracking and identifying security events, especially in applications that take advantage of a database system. Auditing and monitoring consist of recording all actions performed by system users, such as determining the users who had logged into the system, CRUD activities performed by the users, identifying abnormal or unusual actions, among others [24]. If an incident does occur, it may be possible to identify the cause through these mechanisms, thus helping to maintain system security.

Another data security and privacy mechanism is to validate the input. This is particularly important to prevent *SQL injections*. It involves checking validity on inputted parameters based on their data types, length, presence

of patterns and null or duplicated values [35]. In this way, the insertion of malicious code in the input fields can be prevented.

Another commonly used mechanism to maintain the integrity and confidentiality of web applications focuses on using Hypertext Transfer Protocol Secure (Https) connections instead of Hypertext Transfer Protocol (Http). The Https protocol uses Transport Layer Security (TLS) as its transport layer protocol, assuring encryption of data and authenticity between server and client [34].

In order to ensure availability, the server must limit its open ports because they can be exploited by attackers, compromising the system and causing server failure [34].

Lastly, the use of firewalls, the installation of intrusion detection software, the resort to database backups, among others, are also methods that can be applied to maintain security in web applications [35].

Secure Health Applications

In literature, several health applications proposed different mechanisms to secure their data. For instance, in studies [34] and [36], data encryption and authentication methods are taken into account to protect two distinct systems: a software for storing and managing data collected from a mobile health application, and a server-client model for communication of medical images, respectively. Both use the AES algorithm for the encryption, keeping data confidentiality high. Regarding authentication, both resort to TFA: besides the typical username and password, the first system [34], requires also an RSA SecurID token to complete the authentication process; the second system [36], requires an One Time Password (OTP), i.e. using a password that is valid in a short time for only one-time use [29]. Additionally, more mechanisms are also presented in [34], including the use of the Https protocol, load balancing, and separation of identifiable data, such as name, location and identification, from non-identifiable data, in this case, health information. The use of load balancing allows the maintenance of high availability since requests are redistributed across multiple servers, keeping the intruders away as they only have access to load balancers addresses and not to the web service ones. According to the authors of [34], separating identifiable from non-identifiable information in different servers, each one with its own database and web service application, can reduce the value of data loss when only a single server is compromised. Another software solution to protect health data is proposed in [37]. The main goal is to secure platforms or web applications that deal with medical data files. As in the solutions presented above, this one also refers the AES algorithm to encrypt the medical files. The authors mention *hash* functions as another form of converting data to an unreadable state. According to them, *hash* functions can be better than encryption because the resulted block of letters and numbers cannot be converted back to the original form, being, *e.g.*, a good option for password protection in a database. Moreover, the authors suggest using a Virtual Private Network (VPN) to prevent network attacks, as it ensures confidentiality by encrypting data during communications. Lastly, the installation of antivirus programs, the usage of scanning malware programs, monitoring vulnerabilities, and the usage of backups' copies for important files and applications, are also referred as crucial mechanisms to ensure security.

2.5.3 *Software Development Life Cycle*

To develop good quality software, it is necessary to follow some procedures. Hence the Software Development Life Cycle (SDLC) arises [38].

Nowadays, SDLC is one of the most used technologies to produce powerful and high-quality systems [39]. According to study [38], it involves five major phases, which are:

- **Requirements Analysis:** in this phase, the user and system requirements are specified, including functionalities and operational constraints that will define the system [38][40].
- **Design:** the aim of this phase is to define the architecture, components, modules, interfaces, among others, that satisfy the requirements from the previous phase [41].
- **Implementation:** this phase is the coding phase, where the system requirements and design are implemented in a system that provides output [38].
- **Testing:** in this phase, the focus is on finding weaknesses or bugs, and on ensuring the functionality of the produced software [38][41].
- **Maintenance:** lastly, this phase is responsible for maintaining the software. Clients may want to change or include new features [38].

There are multiple SDLC models which help develop products with ease and in a faster way [39]. According to study [42], these models can be divided into two methodologies: heavy weight and light weight. The first one is based on sequential series of steps, meaning each one of the phases presented above only begins if the previous one is completed. Some well-known models include Waterfall, Spiral and Incremental model. In contrast, the second methodology proposes the phases execution to be done in parallel, allowing more quickly and efficient development. It encourages user participation throughout the entire SDLC, enabling the software development to be more respondent to requirements changes. Some of the most well-known models are: Prototyping, Rad and Agile. A comparison between these models is presented in Table 3:

Table 3: SDLC Models Comparative Analysis. Based on [38][42]

Methodology	**Heavy Way**			**Light Way**		
Model	**Waterfall**	**Spiral**	**Incremental**	**Prototyping**	**RAD**	**Agile**
User Involvement	In the begining	Througout the SDLC	Througout the SDLC	Througout the SDLC	Througout the SDLC	Througout the SDLC
Requirements	Complete	Incomplete	Complete	Incomplete	Incomplete	Incomplete
Frequency of Change	Minimal	High	Minimal	High	High	High
Documentation	Full	Full	Full	Low	Low	Low
Product Release	At the end	In phases	In phases	In phases	In phases	In phases
Delivery Time	Late	Early	Early	Early	Early	Early
Software Nature	Existing System	Not specified	New System	New System	Not specified	New System
Cost	Medium	High	High	Low	Low	Low

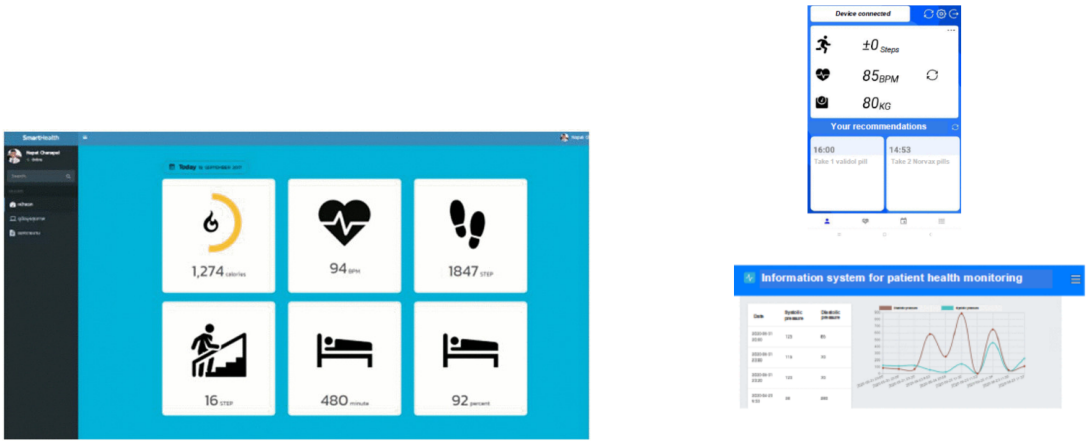
According to study [42], the model choice depends on many factors such as the project nature, involved developers, management style and support. Choosing the appropriate model for software development will allow the construction of cost-effective and efficient products, leading to user satisfaction.

Keeping user satisfaction and productivity in mind, usability should also be considered part of the SDLC. The authors of [43] state that software usability must be assessed, not only at the end but throughout the entire development cycle, thus avoiding major usability problems that can be extremely costly or even impossible to fix. The process can be part of the requirements phase, where users, their needs, expectations, and specifications are analysed, and part of the design phase, where the software's conceptual and visual design can be evaluated.

2.5.4 *Existing Software For Patient Monitoring*

The focus of this section is to present several existing solutions and studies for monitoring people health using intelligent devices. Exploring this area in the literature is significant in this context, as it allows an idea of how monitoring applications are structured, their functionalities, design and used technologies.

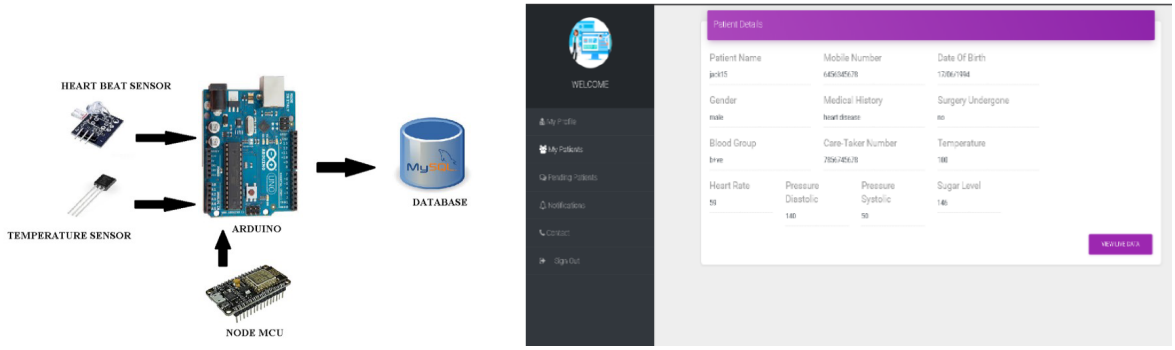
Widely known monitoring solutions include software to display and analyze users' health using wearable devices. Examples are presented in studies [44] and [45], where two distinct software applications are developed with the same purpose: monitoring individuals using smartwatches. Both devices are used to collect information such as heart rate, energy metabolism, moments of sleep, steps count, among others, and both are associated with a mobile and web application. In the first case, the solution is intended to be used during patients' daily life, providing medical recommendations or diagnoses based on data mining algorithms applied to the sensors' collected data. On the other side, in the second case, the solution is proposed to be used during and after rehabilitation sessions of multiple kinds, allowing doctors to be aware of their patients status, manage recommendations and treatment regimes. The authors of [45] state that the history of each patient health data, can be used in smart analysis to find patterns and dependencies that may improve the rehabilitation and, consequently, efficiency in treatment. Figure 6 presents the homepage from the application described in [44] and the mobile and web application screens from [45].



(a) (b)

Figure 6: (a) Web application interface from [44]. (b) Mobile and web application interfaces from [45].

Another similar system is presented in study [46], where patients' data is monitored using temperature and heart rate monitoring sensors outside of conventional clinical settings. A web application was developed to provide an automatic live monitoring system and diagnose patients' health regularly. After temperature and heart rate are measured, data are processed and stored in the database. If the values exceed a previously fixed threshold, the application sends a notification to the doctor, and an online appointment can be scheduled. Figure 7 displays the temperature heart sensors used in the study [46], and the web application interface from [46].

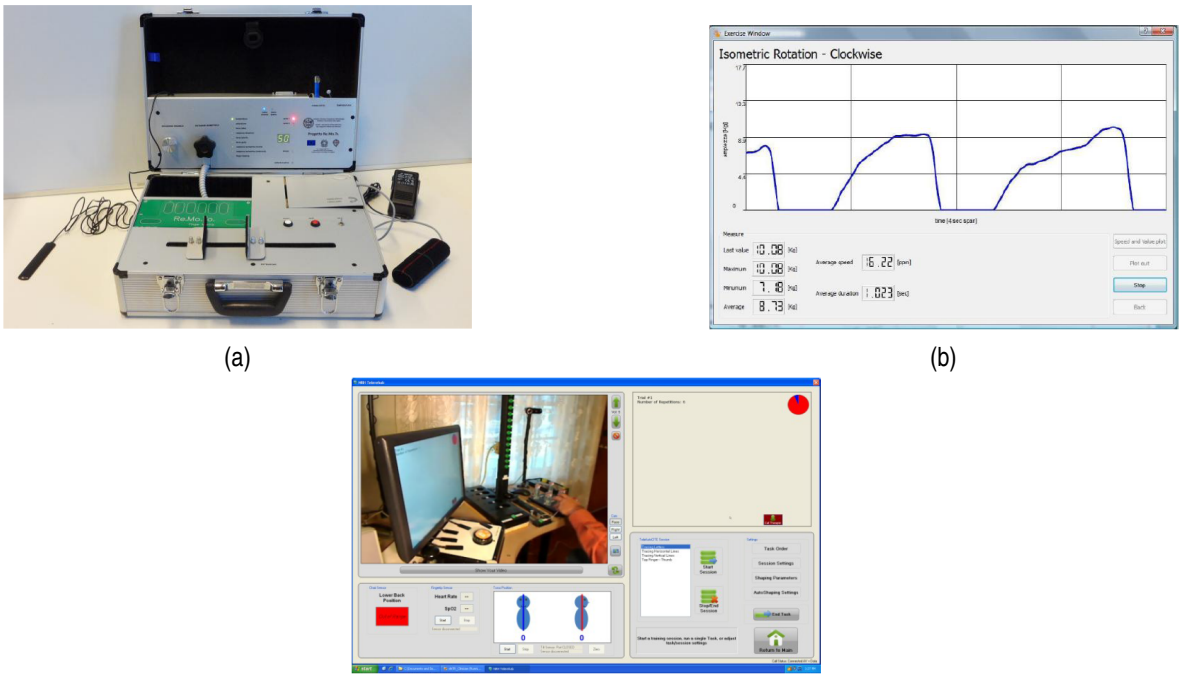


(a) (b)

Figure 7: (a) Block Diagram with the temperature and heart rate sensors [46]. (b) Web application interface from [46].

Other monitoring solutions appear associated with robotic devices used to perform therapy exercises during rehabilitation sessions. An example is presented in [47], where it is proposed a device with remote monitoring capabilities for hand kinesiotherapy sessions of rheumatic patients (Figure 8 - (a)). The developed application allow stherapists to select and monitor several hand exercises, as well as view and export final statistics to a portable file format (.csv). During the rehabilitation sessions, data collected from the device sensors is sent to the computer, and the real-time signal, graphics with the performance relative to each exercise, and other running statistics, are displayed in the interface (Figure 8 - (b)). Another example is presented in [48] and consists of a

telerehabilitation platform for stroke survivors with arm dysfunctions (Figure 8 - (c)). The main purpose of the developed solution is to facilitate in-home therapy, allowing therapists to monitor the patients' health while they are at home. The platform is composed of a table-top workstation and a desktop PC used to acquire data from the workstation sensors. The software application allows the visualization of the sensors data, videoconferencing, and remote monitoring of patient performance during exercises.



(a) (b)

(c)

Figure 8: (a) Telerehabilitation device from [47]. (b) Real time control interface from [47]. (c) Clinician software during patient training [48].

Regarding smart walkers, there are already some software applications intended to monitor the rehabilitation sessions of patients with these devices. Examples are presented in [5], [7], [49], and [50]. Despite some differences, such as the collected data and functionalities, all the proposed applications have the same purpose of displaying, and managing data generated from the walker' sensors during therapy, serving as an aid to healthcare professionals in analyzing the patient's condition and evolution. In this way, therapists can adapt treatments to obtain faster and better results [49]. Most of these data is displayed in graphics, charts, temporal plots, among others, as illustrated in Figure 9. According to [7], it contributes to an appealing interface, allowing an easy understanding of the patients' condition and evolution.



(a) (b) (c) (d)

Figure 9: (a) ASBGo software application interface from [7]. (b) Mobile app (left) and Web app (right) interfaces from [49]. (c) Mobile app interface from [5]. (d) Android OS software interface from [50].

Table 4 presents a term of comparison between the previously described solutions. The main goal is to conclude about what is generally used in the area of monitoring software, and also to determine some possible improvements towards a better solution.

Table 4: Monitoring systems overview

Solution	Smart Device	Type of Software	Collected data from the device	Database	Functionalities	Programming languages/ Frameworks	Communication between Server, Client and Smart Device	Security Mechanisms	Testing Methods
Smart Health Monitoring System [44]	SmartWatch	Android and Web application	· Energy metabolism · Hear rate · Moments of sleep	MySQL	Mobile and Web app: · Allow the visualization of the collected data · Show basic health recommendations Web app: · Allow recording information · Users can change their profile · Administrator can add, delete, modify all members and make health reports to the users	PHP, Java, Android and Bootstrap	Internet is used for the communication between server and client. The server runs the Fitbit API which is responsible for the communication with the smart device.	· Authorization and Authentication via a Login system	-
Patient Health Monitoring Information System [45]	SmartWatch	Mobile and Web application	· Heart rate · Step count · Blood pressure	MySQL	Web app: · Allow the visualization of the collected data · Show doctor recommendations and treatment regimes · Charts with heart rate, step count data and their changes over time · Doctors can manage recommendations, treatments, and patient's threshold levels Mobile app: · Displays the collected data, along with recommendations and treatment regimes · Users can manage their profile · Users can also record systolic and diastolic pressure values	Node js with JavaScript Express library, React js (for the web app), and React Native with Redux (for the mobile app)	Bluetooth is used for data collection from the smartwatch. Internet is used for server and client communication.	· Authorization and Authentication via a Login system	-
Quarcare [46]	Temperature and heart rate sensors	Web application	· Temperature · Heart rate	-	· Manage patient/doctor data · Ermergency ping to a doctor · Online appointments schedulling · Online prescriptions · Videocall consultation · Telemedicine	HTML, CSS, JavaScript, and PHP	Internet connections.	-	-
Remote Monitoring of Hand Rehabilitation [47]	Kinesiotherapy device	Desktop application	· Force [torque, extension] · Repetitions duration · Number of repetitions · Rotation angle and speed · Exercise duration · Number of touches per second · Number and duration of correct/wrong sequences	SQLite	Therapists can: · Select exercises and the related parameters · Start/stop exercises · Monitor exercise execution · View/export final statistics	C++ and Qt framework	At the end of the session, the statical summary is sent to the remote server via a TCP/IP connection with the embedded GSM/GPRS module. A Bluetooth adapter is used to allow the therapist to control the device.	-	Patients express their satisfaction by fulfilling the QUEST questionnaire.
Telerehabilitation platform for home--based automated therapy of arm function [48]	teleAutoCITE	Desktop application	· Palm height · Heart rate · Blood-oxygen · Torso position · etc.	-	· Videoconferencing · Therapists can control the therapy · Remote monitoring of patient performance · Live display of data from patient sensors	C#.NET 2007, HDDK Developers Kit (for audio and video)	Bluetooth is used to collect some data from the device.	· Videoconfere-ncing is secured under the AES encryption algorithm.	A pilot test was made with 3 patients.

Solution	Smart Device	Type of Software	Collected data from the device	Database	Functionalities	Programming languages/ Frameworks	Communication between Server, Client and Smart Device	Security Mechanisms	Testing Methods
ASBGo database [7]	Smart walker ASBGo	Desktop application	· Average width · Total distance · Session duration · Average Speed · Gait distance discrepancy · Duration of the right/left foot ahead from the contrary foot · Right/left strike · Standard deviation · Average foot angle · Standard deviation of the angles	MySQL	· Creation/ removal of patients · Creation/ removal of sessions · Session configuration · Graphs with different statistics · Temporal patient evolution	Visual Studio 2013 with DevExpress tool	-	-	-
PhysioApp [49]	Smart walker	Android and Web application	· Orientation · Number of steps performed · Bilateral walker elevation · Balance · Elevation · Pressure	MySQL	Administrators can: · Manage physiotherapists · See own data Physiotherapists can: · See own data · Manage patient's profile · Start live sessions · Manage patient's workouts Patients can: · See own sessions and workouts	C programming language and PHP scripts	The communication between server, client and database occurs via WiFi, and the transmission of data to the mobile device is ensured by using Bluetooth.	· Access control	-
m-Health [5]	Smart walker and smart crutches	Android application	· Training duration · Hand Force · Heart rate · Information about the lower limb motion · Gait pattern information	MySQL	· Manage current and historical data of each rehabilitation session · Evolution tendency of multiple metrics related to gait	Android development studio and Android SDK	The communication between smart and mobile devices occur via Bluetooth, and WiFi is used for the connection between server and mobile app.	-	-
Smart Walker for Pervasive HealthCare [50]	Smart walker	Android application	· Average velocity · Training session duration · Total traveled distance · Acceleration	SQLite	· User profile · Synchronization with the server · Charts with training sessions data · Changing session preferences	Java and Android SDK	Bluetooth is used for the communication with the smart walker, and WiFi for the synchronization process between the mobile phone and server.	-	-

Performing a critical analysis on Table 4, it is possible to identify some common points present in literature, including:

- The majority of the solutions are Web or Android applications;
- Relational databases were the most resorted for storing all types of data, including sensors' data;
- Most software interfaces display graphs, charts, and temporal patient evolution plots;
- For communication between server, client and intelligent device, the most used technologies are Bluetooth and WiFi;
- Only a few of the solutions have implemented security mechanisms;
- Software validation was only performed in two of the presented applications, namely [47][48].

2.5.5 *General Conclusions*

Considering the previous sections of this chapter, some general conclusions about databases, security, development life cycle, and software for monitoring patients can be drawn.

As far as databases are concerned, relational and non-relational databases have different focuses and characteristics, advantages and disadvantages. In this way, the choice between these two types must be made considering the system requirements and the best way to meet them. Regarding healthcare databases, it is notable that there is more recurrence to NoSQL databases when dealing with Big Data, namely data resulting from sensors or machines. On the other side, relational databases are usually used when the focus is on storing and handling data from patients, healthcare professionals, appointments, among others.

Concerning security, in the development of a software application for the health sector, it is crucial to resort to security mechanisms to protect the patients' privacy, as health data is considered extremely sensitive [28]. The most used security measures in web applications include encryption of data, access control, input validation, Https protocol, among others. It is also important to follow established regulations to protect citizens' data. In the European Union, the clauses and rules that must be followed are present in the GDPR.

For software development to be done easily and more effectively, a good practice is to follow a SDLC model. There are multiple models that can be considered. Choosing the most appropriate model is important and can simplify the software construction, resulting in cost-effective and high-quality products, leading to user satisfaction.

Finally, regarding existing monitoring applications, some drawbacks can be solved to construct better solutions. For instance, a large part of the presented applications uses Android technology, which may be a limitation related to the available devices. In addition, the literature systems resort to SQL databases to store the data acquired from the sensors of the smart devices, which can be a drawback in terms of performance and availability. Furthermore, security mechanisms and tests for system validation do not often appear, which may bring insecurity to users. Nevertheless, most of the presented solutions concluded that an application to monitor rehabilitation sessions is beneficial, contributing to a better perception of the patient's condition throughout therapy.

3

SYSTEM SPECIFICATIONS

3.1 INTRODUCTION

This next chapter specifies the planning and modulation phase that was necessary to: **(i)** optimize the WALKit SmartW embedded software by including security measures to protect patients sensitive data; **(ii)** propose an efficient strategy for the storage of data acquired during rehabilitation sessions with the device; and **(iii)** develop a user-friendly and intuitive responsive application that allows both patients and clinicians to verify the patient's evolution throughout the rehabilitation program and perform the management of rehabilitation sessions considering relevant clinical information. To accomplish this, this chapter presents a **(i)** a brief introduction to WALKit SmartW (Section 3.2), **(ii)** the followed SDLC models (Section 3.3), **(iii)** technical and clinical requirements gathering (Section 3.4), **(iv)** the database choice (Section 3.5), and **(v)** an analysis of the system's global structure (Section 3.6).

3.2 WALKIT SMART WALKER

WALKit Smart Walker, illustrated in Figure 10, is being developed at the BiRDLab from University of Minho, and its main goal is to assist the rehabilitation of ataxic patients, by providing personalized mobility assistance oriented to the patient's needs, and fostering human-robot interaction and collaboration through biofeedback systems and gait and balance monitorization algotithms.

The walker includes several embedded sensors on its mechanical structure (Figure 10), including two RGB-D cameras, two load cells, one infrared sensor, one inertial measurement unit (that integates an accelerometer and a gyroscope in the same board), nine ultrasonic sensors, two encoders, and a laser range finder. WALKit SmartW presents a modular and hierarchical architecture divided into high and low-level controls. The low-level control runs a Real-Time Operating System (RTOS) on an STM32F4 Discovery. This board is used to read most of the low-level sensors, *i.e.* the inertial sensor, load cells, infrared sensor, encoders, and ultrasonic sensors, and to control the device in terms of linear speed considering the response of a Proportional-Integral-Derivative (PID) controller. Regarding the high-level control, this runs ROS in a minicomputer, and it is considered the device's Central Control Unit. It is responsible for communicating with the low-level board and acquiring data from the two RGB-D cameras and the laser range finder.

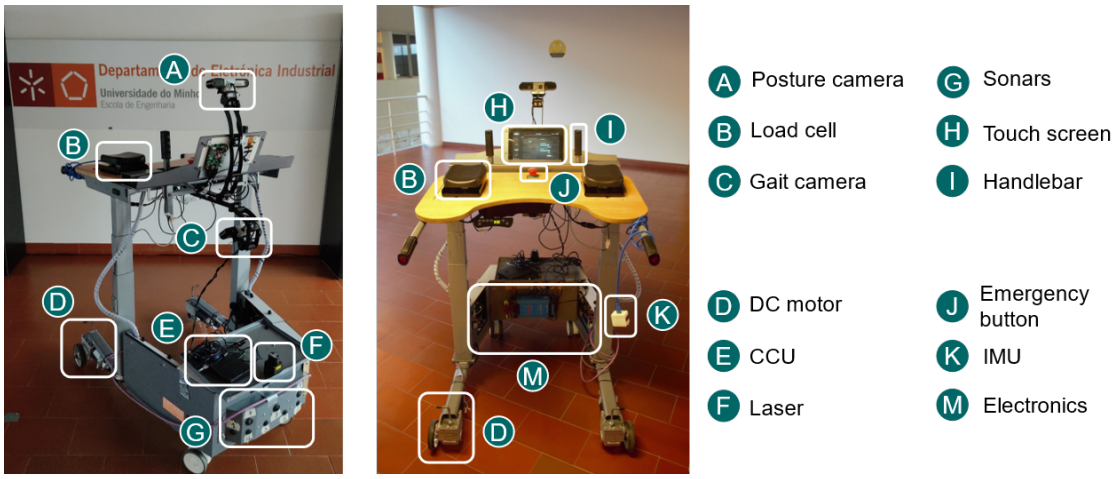


Figure 10: WALKit Smart Walker and its respective sensors.

Table 5 presents some information regarding the sensors that make up the walker's architecture, namely: **(i)** the type of sensor; **(ii)** the number of units in the device; **(iii)** a brief description of their utility; and **(iv)** some technical specifications, namely the range, resolution, sampling frequency and unit.

Table 5: WALKit SmartW sensors' information

Type of sensor	Sensor	No. of sensors	Description	Resolution	Range	Sampling frequency	Units
RGB sensor	Orbbec Astra	2	3-channel image to assess gait and torso's posture.	640 x 480p	0-255	30 *fps*	N.A.
Depth sensor	Orbecc Astra	2	Distance between the image plane and the corresponding object in the RGB image.	640 x 480p	0.6-8	30 *fps*	m
Infrared	Sharo GP2Y0A21	1	Distance of the user from the walker.	N.R.	0.1-0.8	100 Hz[1]	m
Load cells	CZL635	2	Measures the load the user is applying on the device.	N.R.	0-50	100 Hz[1]	kg
Acceleration	MPU-6050	1	Measures torso acceleration in *x*, *y*, and *z* axes.	N.R.	± 8	100 Hz[1]	*g*
Gyroscope	MPU-6050	1	Measures torso angular velocity in *x*, *y*, and *z* axes.	N.R.	± 2000	100 Hz[1]	deg/s
Ultrasonic sensor	MB1210 XL MaxSonar-EZ1	9	Frontal distance from obstacles.	1	0-765	100 Hz[1]	cm

N.A. - Not Applicable; N.R. - Not Refereed; *fps* - frames per second; Hz-s^{-1}; $p - pixels$

1 It could reach 200 Hz in the new architecture under development.

During rehabilitation sessions, WALKit SmartW sensors are constantly analysing and collecting information about the end-user, thus resulting in a considerable amount of data that must be processed and stored correctly. This data becomes bigger as the sampling frequency increases, which can reach to 200 Hz, i.e., 200 samples per second, for the device's new architecture under development. Therefore, there is the need to implement a DBMS capable of storing these data efficiently. Furthermore, in addition to the walker-embedded application responsible for promoting assistance during sessions, there is a also the need for an application which provides a way for clinicians and technicians to access and manage the data acquired from the sensors. This will be

particularly helpful to consult and update the end-users' clinical information, but also to assess their evolution throughout therapy and adapt the rehabilitation program accordingly.

3.3 SDLC MODEL

To achieve software quality SDLC models can be used, allowing the application development to be easier and quicker.

As discussed in the literature review (Chapter 2), the model that fits with existing systems is the traditional Waterfall approach [38], as illustrated in Figure 11 (a). For that reason, and since users already had contact with the application and knew exactly what to change (requirements well known), this was the model followed to continue developing the WALKit SmartW application. According to [51], this model has the advantages of being user-friendly, requiring minimal planning overhead, and being

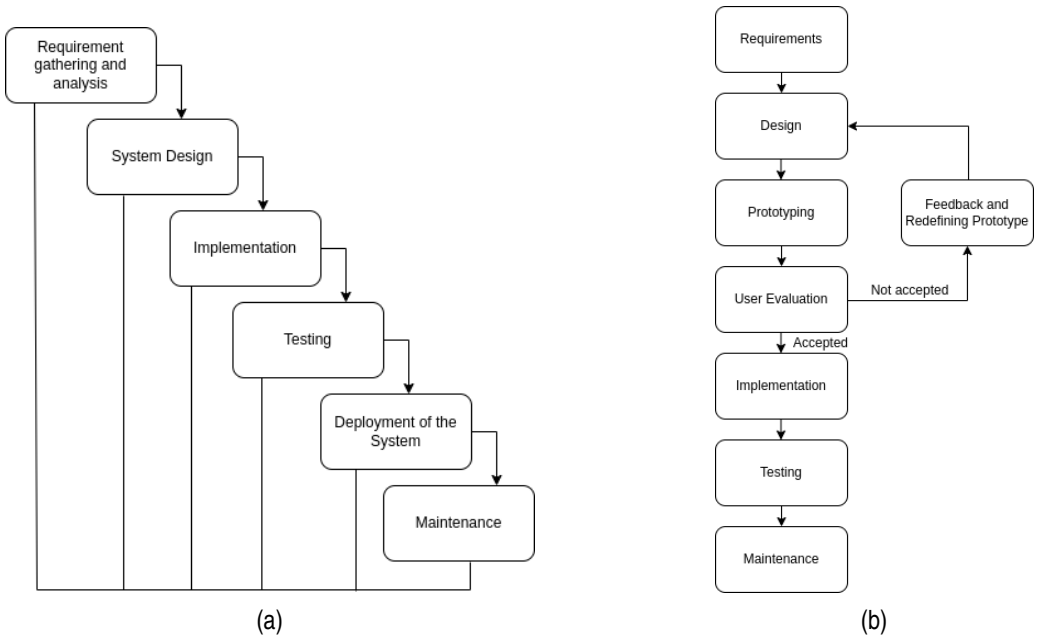


Figure 11: SDLC Models: (a) Waterfall Model. Based on [38]; (b) Prototyping Model. Based on [51].

easy to understand, implement and manage. Nevertheless, some of the main throwbacks include inflexibility, late problems discovery, and tough going back to phases. As Figure 11 (a) illustrates, the waterfall model is a linear sequential model composed by six phases: **(i)** requirements gathering, **(ii)** system design, **(iii)** implementation, **(iv)** testing, **(v)** deployment and **(vi)** maintenance [38]. The requirements phase is presented in this chapter, while the other phases are described in detail in Chapter 4.

Regarding the web application, the followed SDLC model was the Prototyping model. In this case, as seen in Figure 11 (b), the model is composed by seven steps which are: **(i)** requirements, **(ii)** design, **(iii)** prototyping, **(iv)** user evaluation, **(v)**implementation, **(vi)** testing and **(vii)** maintenance. The purpose of this model is to develop a prototype system that includes the basic requirements, and then evaluate the resulting prototype, verifying if it

meets the established requirements. If not, another version should be develop until reaching the user satisfaction [42]. The main advantages of this model include the end-user involvement during all cycle, early error detection, and faster feedback, resulting in improved solutions [51]. In this chapter the requirements phase is described, and the other phases are explained in Chapter 5.

3.4 REQUIREMENTS

According to both SDLC models, and to construct efficient applications that satisfies all needs, a collection of functional and non-functional requirements was performed and considered throughout the development process. The following subsections will present the entire requirements gathering and formation process.

3.4.1 *Requirement Gathering Methods*

The method of collecting requirements took place through in-person and online meetings with health professionals. A *google forms* questionnaire was also used (Appendix A). This method enabled greater participation by health professionals since it was difficult to contact directly with some of them due to schedule restrictions. A total of seven healthcare professionals were involved in the initial requirements gathering phase, with five of them actively participating in the entire development process. Based on the questions made on the online questionnaire, the subsequent demographic information was acquired:

- The range of health professionals' age is between 27 and 44 years, with an average age of 33 $\pm$ 6 years. Figure 12 (a) represents the participants' age distribution;
- All health professionals were female, as shown in Figure 12 (b);
- One health professional is a physiotherapist, two are physiatrists, and the remaining are residents of the Physical Medicine and Rehabilitation Service (Figure 12 (c));
- Regarding technologies, all health professionals often use computers and smartphones, four also use a tablet, and one mentioned the use of a kindle, as Figure 12 (d) shows ;
- Five health professionals always resort to technologies at work, and two contact with software applications often (Figure 12 (e));
- Five health professionals classify the hospital internet connection as median, one classifies it as strong, and the other as weak, as shown in Figure 12 (f).

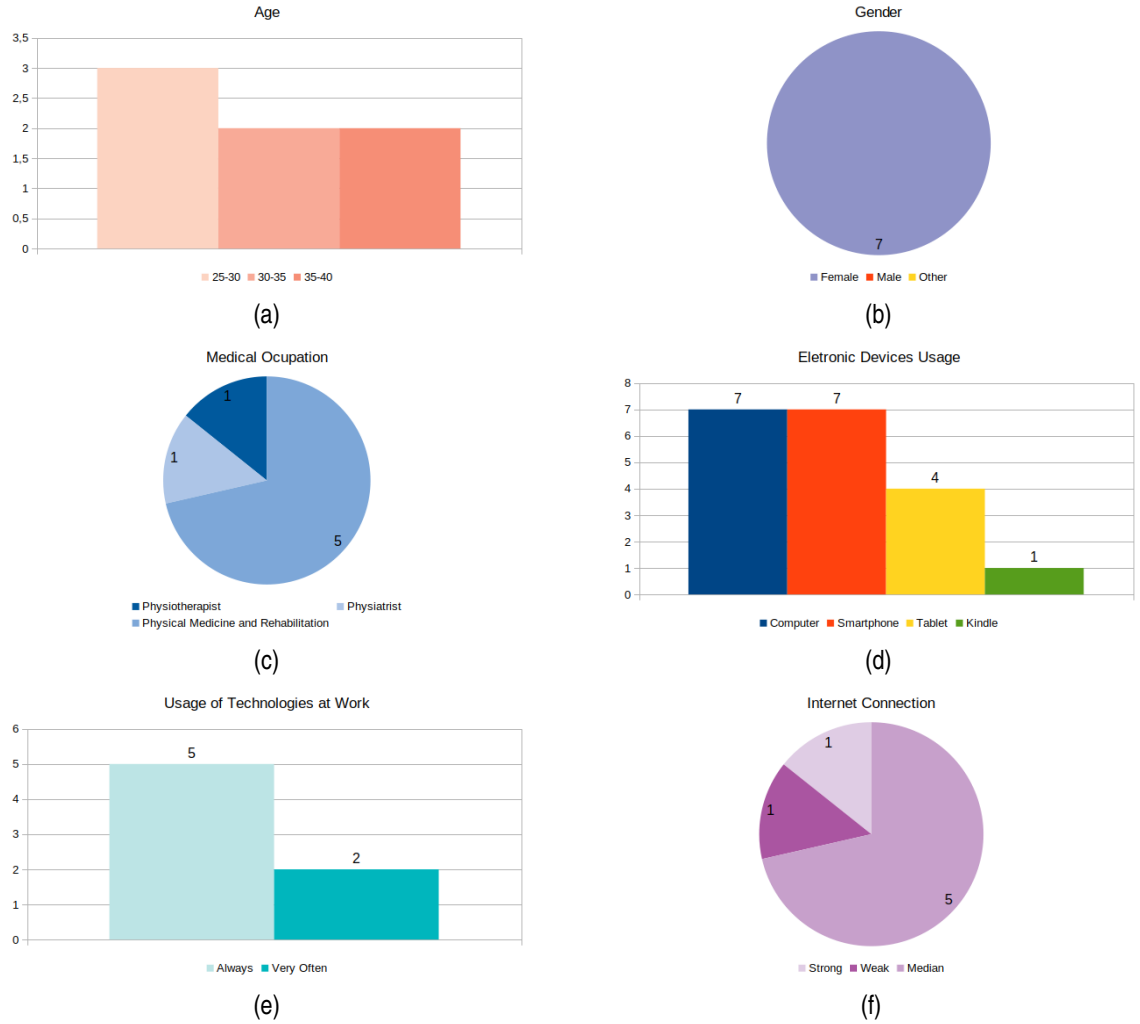


Figure 12: Demographic information from the requirements gathering participants: (a) Age Distribution; (b) Gender Distribution; (c) Medical Ocupation; (d) Electronic Devices Usage Distribution; (e) Usage of Technologies at Work; (f) Internet Connection.

Despite the demographic questions, others were asked to understand what features, suggestions, and ideas are needed to build a valuable and high-quality application that fulfil their work requirements. The answer to those questions, along with online and in-person meetings with the health professionals and the WALKit team, gave rise to the functional and non-functional requirements presented in the following subsections.

3.4.2 *Functional Requirements*

WALKit embedded Application

Regarding the WALKit embedded application, the following clinical and technical requirements were considered:

- Adding anthropometric data - the system must allow the storage and visualization of the end-user anthropometric data, such as body mass, ankle height, pelvis height, individual height and BMI, which are relevant to characterize the population;
- Pop-up alerts - the system must inform users of the success/unsuccess of their actions;
- Rehabilitation sessions' information - users must be able to visualize the details of therapies, namely the session configurations (*e.g.* chosen algorithms, chosen gait speed) and some clinical remarks that the clinician considers relevant in the context of the therapy.

Web Application

Regarding the web application, the following clinical and technical requirements were considered:

- Manage patient's data - system must allow the visualization, insertion, edition, and removal of patients;
- Sessions management - the system must allow the visualization and removal of patients' rehabilitation sessions;
- Archive management - the system must have an archive where it can store the information of patients who are not currently undergoing therapy, but who may return to therapy in the future. Additionally, it must allow the visualization of this information, remove or readmit the patients.
- Charts with patient evolution - the system must have progress charts with the patient's evolution throughout the rehabilitation sessions;
- Download sensors' data - the users must be able to download files in CSV format relating to the sensors values acquired in each rehabilitation session;
- Download session details - the users must be able to download a PDF file with the session information;
- Patient profile - the system should contain a dedicated profile for patients so that they can access exclusively to their clinical information and view their progress in therapy;
- Clinical evaluation scales - the system should allow the completion of clinical scales questionnaires/tests, as well as show their evolution over time;
- Patient's falls - as a clinical requirement, the user must be able to record the daily number of patient falls.

3.4.3 *Non-functional Requirements*

The following non-functional requirements were organized by the classification scheme proposed by Sommerville [52], which divides the requirements into three different categories: **(i)** product requirements, **(ii)** organisational requirements, and **(iii)** external requirements.

WALKit embedded Application

Product Requirements:

- The system must support data storage at a minimum frequency of 200 Hz;
- The system new features must follow the current design.

Organizational Requirements:

- The system must be developed in C++ and QML programming languages, which are the basis of the current application.

External Requirements:

- The system must not disclose any personal information about patients except to healthcare professionals who have access to the system;
- The system must follow the ethical requirements applied to sensitive data.

New Application

Product Requirements:

- The system must be easy and intuitive to use;
- The system must be able to be used on different electronic devices such as computers, smartphones and tablets;
- The system must be able to be used outside hospital area.

External Requirements:

- The system must not disclose any personal information about patients except to healthcare professionals who have access to the system;
- The system must follow the ethical requirements applied to sensitive data.

3.5 DATABASE SELECTION

The system embedded in the walker had already associated a relational database, *MySQL*. The main functionality was to store data related to patients, namely personal and anthropometric details, clinical evaluation, and technical data from rehabilitation sessions (date and time of the session and observations). However, considering the requirements of both applications, there was a need to change the existing system so that it can support more data.

The new data to be stored consists of patient, session configurations, and user data, such as username and password. The data acquired by the different sensors during the rehabilitation sessions should also be stored efficiently, since it was being stored in files from which access was difficult.

Considering all these data and the literature review, a polyglot persistence database system was implemented. Polyglot persistence consists of using a combination of multiple databases in a way that enhances performance [53]. Thus, the system will combine a relational and a non-relational database.

As discussed in the literature review (Chapter 2), relational databases are characterized by ACID properties [9][14], internal security mechanisms (making them ideal for dealing with sensitive data [11]), and the ability to deal with complex queries [12]. For these reasons, this type of database was chosen to store all structured data, namely patient, session configuration, clinical scales and users data. The storage of sensors' data is set aside as these databases are not ideal for supporting huge volumes of unstructured data [12]. Therefore, a non-relational database, which, as studied before, are the best option for dealing with Big Data while keeping the level of performance high [13], was chosen to store raw and processed sensors' data. Only data that is necessary for the correct functioning of the system is stored, thus following the principle of Data Minimization present in the GDPR regulation [25].

In the following subsections, the chosen databases are explained in more detail.

3.5.1 *Relational Database*

Taking into consideration the requirements and the data to be stored, a conceptual model was created. As shown in Figure 13, the model is composed by eleven tables, each of which represents an entity with its associated attributes. In general, each patient, characterized by personal and anthropometric data, fall records, and clinical evaluations, has one or several rehabilitation sessions in association. Each rehabilitation session is identified by its primary details and by the various algorithm settings. Lastly, each patient has also a system user account.

The chosen RDBMS, which allowed the implementation of the idealized model, was *MySQL* database, one of the most common in the SQL universe. It is a well-known open-source database [14] characterized by exceptional performance, reliability and ease of use [54], which proved to be a valuable option for the system.

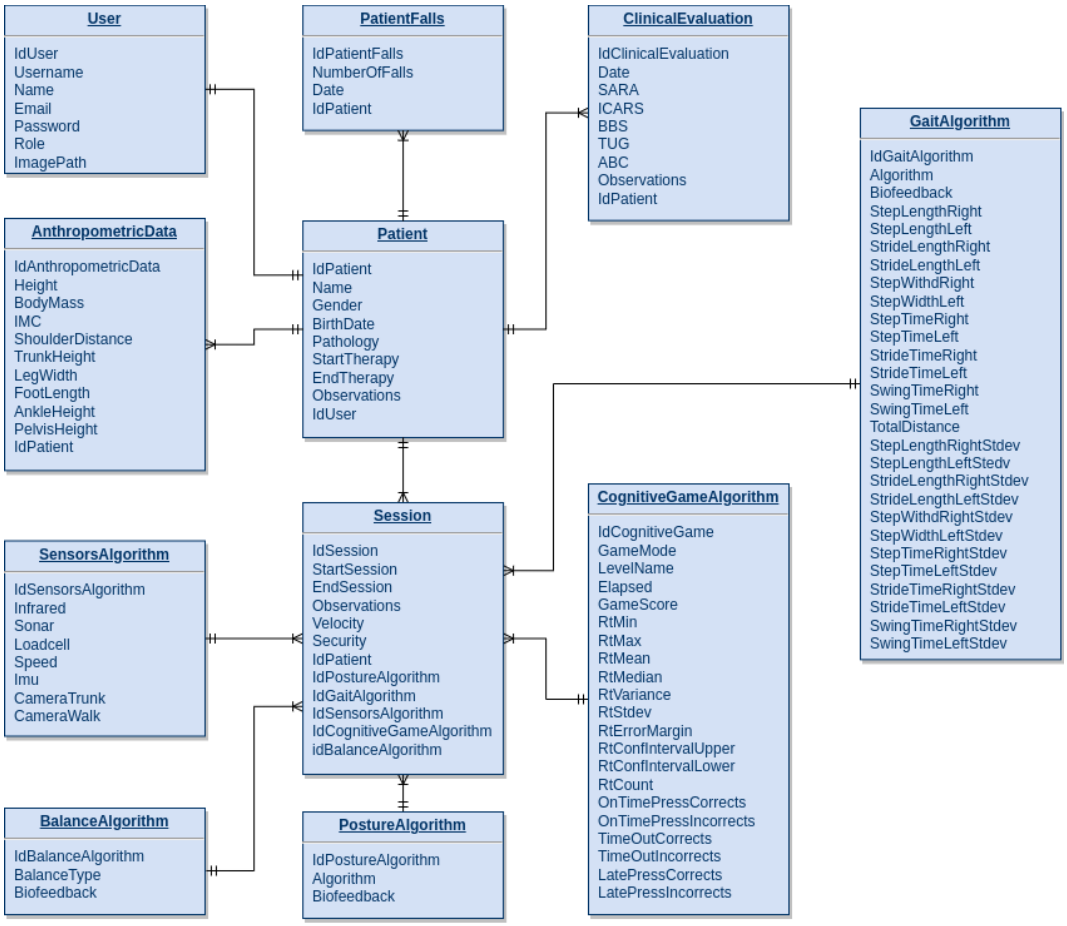


Figure 13: Database Conceptual Model.

3.5.2 *Non-Relational Database*

As explained in Section 3.2, the acquisition of sensors' data occurs according to a sampling frequency, for example, frequency of 100 Hz. Therefore, it can be considered time series data since the measurements are taken at time intervals from the source. Moreover, the main intention is to store and query this great amount of data without compromising the system performance. Considering this and the four types of non-relational databases studied in the literature review, a document-oriented database, *MongoDB*, was chosen.

MongoDB is an open-source, schema-free database that allows data to be stored in BSON (Binary JSON) files. It has its own query language, which also brings high flexibility [22]. *MongoDB* in version 5.0 provides time series collections, witch efficiently stores time series data while reducing the disk usage at the same time, being optimal for saving sensors' data. Therefore, the system's non-relational database is composed of a time series collection, which is responsible for storing the sensors' raw and processed data in every rehabilitation session. Each document is composed by the following fields:

- *timestamp*: represents the date of each document;
- *sessionId*: the session identification;
- *patientId*: identification of the patient who is performing the rehabilitation session;
- *data_raw*: the data acquired from each sensor in the respective timestamp;
- *data_processed*: the data processed in the respective timestamp.

The following JSON document represents an example of the data structure at each timestamp. If during a rehabilitation session, data is being collected at a sampling frequency of 100 Hz, one hundred documents, similar to this one, will be stored per second.


```
{
    "_id": {
        "$oid": "62ecf17ab266dbfd5f0cbb89"
    },
    "timestamp": {
        "$date": "2022-08-05T11:31:17.982Z"
    },
    "sessionId": "1",
    "patientId": "UM0000",
    "data_raw": {
        "sonar": [
            187.380859375,
            117.54948425292969,
            150.137451171875,
            80.30608367919922,
            109.4024887084961,
            359.631591796875,
            112.89405822753906,
            112.89405822753906,
            54.70124435424805
        ],
        "pot": {
            "pot_left": 0.0068426188081502914,
            "pot_right": 0.0772237777709961
        },
        "infrared": 24,
        "loadcell": {
            "loadcell_left": 10.69792366027832,
            "loadcell_right": 15.093088150024414
        },
        "battery": 0.9999998807907104,
        "imu_torso": {
            "temp": 29.58293914794922,
            "gyroX": -1.8902440071105957,
```

```
            "gyroY": -0.8536585569381714,
            "gyroZ": 0.24390244483947754,
            "accelX": -0.0283203125,
            "accelY": -0.4501953125,
            "accelZ": -0.934814453125
        },
        "speed": {
            "motor_left_speed": 0,
            "motor_right_speed": 0
        },
        "button": {
            "button_left": 0,
            "button_right": 0
        }
    },
    "data_processed": {
        "pitch_raw": 64.14798736572266,
        "roll_raw": -86.70124816894531,
        "pitch_calib": 3.2813899517059326,
        "roll_calib": -1.6656301021575928
    }
}
```


Listing 3.1: MongoDB document with sensors data.

3.6 SYSTEM GLOBAL STRUCTURE

In order to comply all the imposed requirements, it was decided to develop a responsive web application as the new complementary application to the WALKit SmartW system. In this way, it becomes possible to use it on different electronic devices and outside the hospital area, allowing patients to access their progress from home.

Considering all the above, the system as a whole is composed by two databases, *MySQL* and *MongoDB*, the walker and its embedded application, and a web application, as it is illustrated in Figure 14.

Users can interact with both applications. Regarding the walker application, the access between it and the relational database is made directly, returning and displaying the results that the user requested on the WALKit SmartW computer screen. It also sends the sensor's raw and processed data to *MongoDB*. Regarding the web application, the user can access it through different electronic devices (smartphone, tablet, or computer). The required information reaches the devices through Http responses from the server, which is connected with both databases.

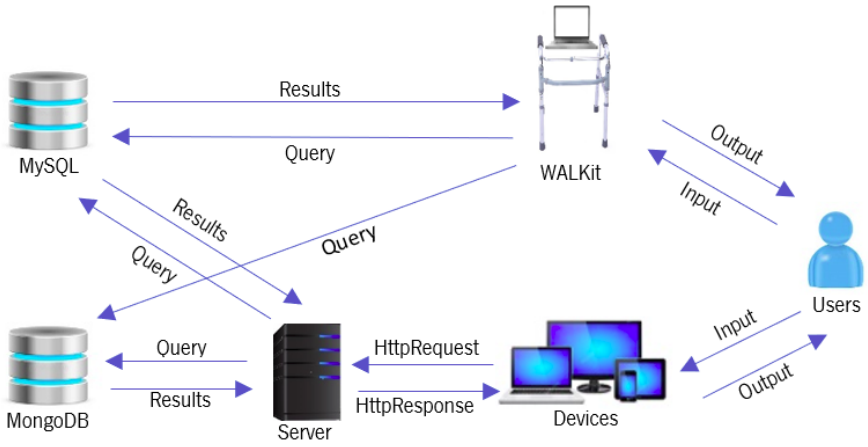


Figure 14: System Global Structure.

3.7 GENERAL CONCLUSIONS

This chapter presented the planning and modulation phase for the optimization and development of the applications. To build software with greater ease, the development life cycle phases were outlined according to two different models: the Waterfall and Prototyping models, for the SmartW embedded application and the Web application, respectively. The first phase of each model consisted on the requirements gathering, which was presented in this chapter according to the ideas and suggestions of the healthcare professionals who volunteered to participate in this study. With these, the next steps for building the applications were structured. Considering the collected requirements, the data storage was planned, and a polyglot persistence database system was implemented, allowing to enhance the overall performance. Therefore, the system global structure is composed by two databases, a relational database (*MySQL*) and a non-relational database (*MongoDB*), to allow efficient data storage.

4

WALKIT APPLICATION

4.1 INTRODUCTION

As mentioned before, WALKit SmartW endows an embedded application that allows health professionals to perform basic CRUD operations regarding the patients' clinical information and allows for therapy configuration considering the patients clinical needs. However, this application needs optimization since it presents some drawbacks, namely security issues regarding sensitive personal data and the lack of feedback to users regarding their operations in the software. Considering the requirements presented in Section 3.4, this optimization focused on adding new features and increasing the system's overall security.

This chapter presents the design and implementation process that was carried out to optimize the SmartW embedded application. Furthermore, it presents the validation of the DBMS and the SmartW application with end-users from Hospital of Braga.

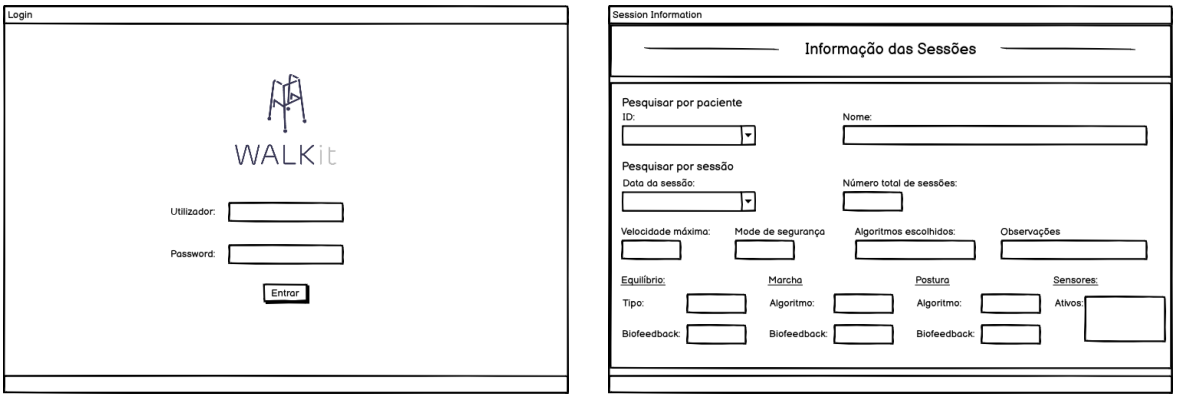
4.2 DESIGN

The system design corresponds to the second phase in the Waterfall model [38]. The purpose of this phase is defining the architecture and the different interfaces that fulfil the requirements presented in Section 3.4. Considering this, wireframes were created to serve as a guidance for the optimization of the robotic walker embedded application.

4.2.1 *Wireframes*

Balsamiq was the chosen tool for designing the different user interfaces. This tool allows the wireframe creation using simple components, similar to paper design, being easy to perform modifications if needed.

For security purposes, a login page was designed, as seen in Figure 15. The user must insert his username and password to have authorization and access the system. Furthermore, Figure 15 also shows the session information design. To check the session details, namely configurations, the user must search for the required patient and for the solicited session date.



(a) (b)

Figure 15: (a) Login page wireframe. (b) Sessions' information page wireframe.

Figure 16 shows the designed alerts created to warn the users of the failure/success and information regarding their actions.

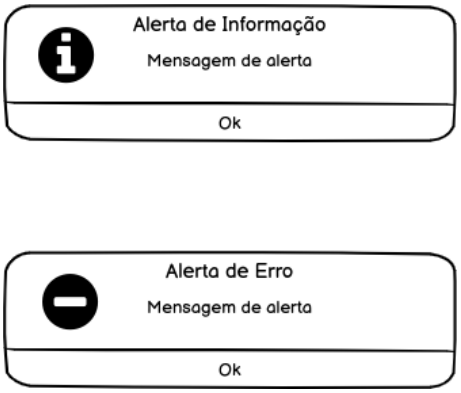


Figure 16: Alerts Wireframes.

4.3 IMPLEMENTATION

4.3.1 *Technologies*

The programming language used was C++ for the backend and the QT framework for the development of the graphical interfaces. Furthermore, it was used ROS, which is a set of software libraries to build robot applications. These technologies were used since they are the basis of the robotic walker software.

4.3.2 *Security Mechanisms*

Since the application deals with sensitive data, and keeping in mind the principles in GDPR, it is essential to have security mechanisms that ensure data confidentiality, integrity, and availability. In this way, patients and healthcare professionals may feel more comfortable while using the application.

The walker's embedded application was exempt from any security. Anyone who had access to the robotic walker was able to access patient data. The only mechanism that was being used was to insert fictitious codes/-names instead of the patients' identification and original name. This mechanism was inefficient, and the association between real-fictitious may become confusing for health professionals.

Considering all the above and what was studied in the literature review, several security mechanisms were implemented. Firstly, a login system with user credentials, similar to the wireframe, was developed. Only authorized health professionals are now capable of entering the system. Thus, achieving authentication and authorization. In this case, is being use SFA [30], in which is only required username and password for logging in. TFA and MFA were excluded at the moment, because the application is still under development, and the access must be kept simple. Figure 17 depicts the developed application login system.

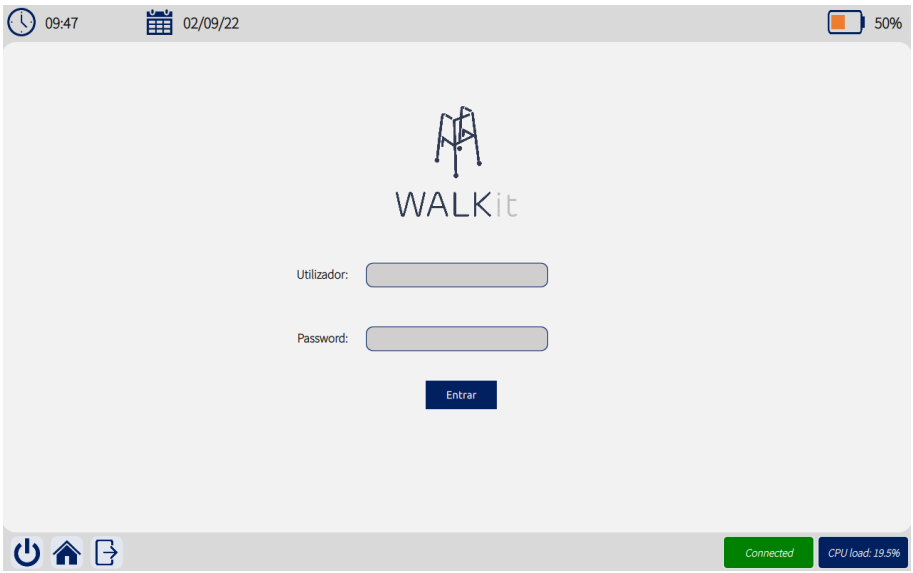


Figure 17: WALKit application login system.

Secondly, considering that a login system was created, it was essential to keep the users' passwords safe. To do so, *Argon2* was used. *Argon2* is a cryptography hashing algorithm, most recommended for password hashing. This means that the password is converted to an unreadable state, and it cannot be converted back to the original form [37]. The password hash is kept in the database, keeping the user's raw password secure.

Thirdly, patients' and clinicians' data must be protected from malicious users. Data should remain indecipherable while at rest and in transmission. Thus, a cryptography algorithm was applied. As discussed in Chapter 2, AES is the best algorithm in terms of security and performance and, for that, was the chosen cryptography algo-

rithm. AES has five different modes of operation. In this case, it was used the Cipher Block Chaining (CBC) [55] since it is considered a secure mode when using a random initialization vector in each encryption. In this mode, the plaintext (text to be encrypted) is divided by blocks, and each block is XORed with the previous ciphertext block, and then encrypted. The first block is XORed with the initialization vector. The result is an unreadable text called cryptogram. Figure 18 illustrates the CBC mode of operation. To be kept in a secure way, the CBC cryptography key, is saved as an environment variable in an *.env* file.

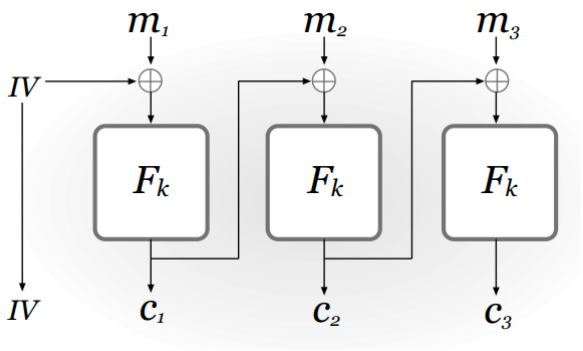


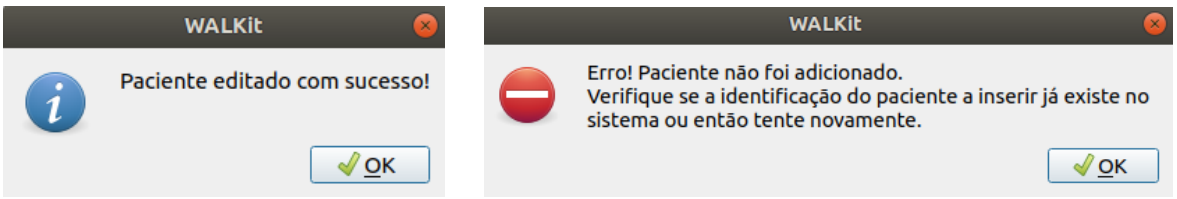
Figure 18: CBC Mode of Operation. Retrieved from [55]. IV stands for Initialization Vector; m1,m2,m3 are the plaintext blocks; F corresponds to the block cipher; and c1,c2,c3 correspond to the ciphertext blocks.

Another security mechanism applied was input validation. All input fields are validated according to the expected type, size, null values, or presence of pattern. By performing this, the insertion of malicious code is prevented. Additionally, data also becomes correctly stored and follows always the same structure.

4.3.3 *Application Modifications*

In addition to the implementation of security mechanisms and, following the requirements, new application features were also developed.

The first implemented feature was pop-up alerts. Previously, users could not know about the success or failure of their actions. Now, each action performed by the user, *e.g.* inserting or updating a patient's information, is followed by a success or error message, as illustrated in Figure 19 (a) and 19 (b).



(a) (b)

Figure 19: Pop-up alerts: (a) Information Alert; (b) Error Alert.

Some patients' personal information, namely height, pelvis height, and ankle height were also requested to be added as input fields to the previous implementation. Figures 20 and 21 shows the insert and search patient menus with the modifications that were introduced. It is noteworthy that the system design was kept, and only the necessary fields were added.

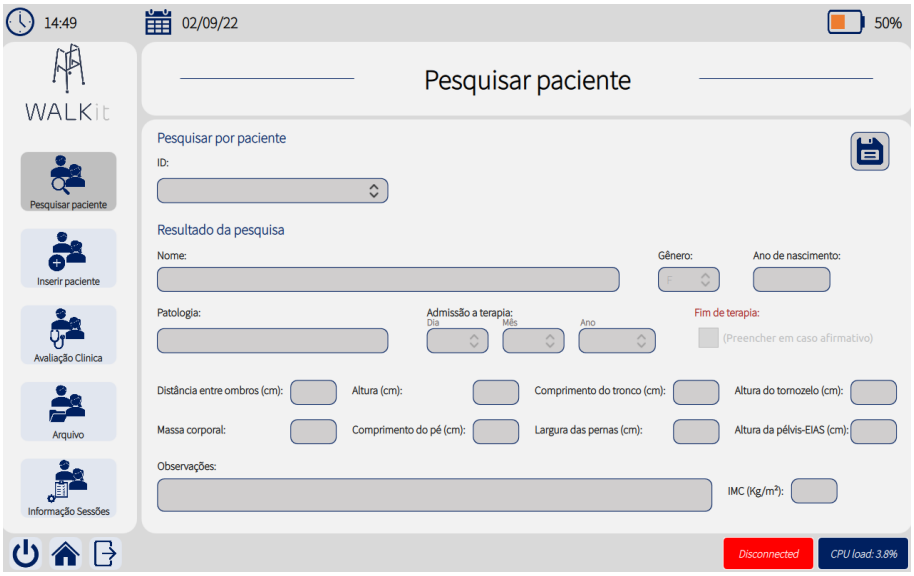


Figure 20: Search Patient Menu.



Figure 21: Insert Patient Menu.

Lastly, a menu with historical sessions' configurations was developed. This menu allows health professionals to verify quickly the settings of previous sessions that may be useful to delineate the following sessions. Figure 22 illustrates this menu.



Figure 22: Sessions' Information Menu.

4.4 APPLICATION VALIDATION AND TESTING

The satisfaction, trust and acceptance of users must always be high, so it becomes crucial to evaluate the quality of the application based on different parameters. This section is focused on describing the testing methods used to validate the application. Firstly, since this application was also responsible for storing the sensors' data, database testing was made to check if the huge amount of data was being stored correctly, while maintaining the system quality. Then, a usability test was performed to consider if the application interfaces satisfied all the end-user requirements.

4.4.1 *Database Storage*

One of the biggest challenges in this application development was the storage of the data collected during each rehabilitation session. That is because the amount of data acquired is large considering the high sampling frequency, and there is always a possibility of data loss at this stage or even program failures due to computational overload. Therefore, tests were carried out to find the ideal solution for data storage. The test methodology and results are reported throughout this section.

Methodology

The tests focused on verifying if there was any program failure or data loss during data acquisition. For that, a simulation of rehabilitation sessions of both thirty and sixty minutes was performed, acquiring data at a

sampling frequency of both 100 Hz and 200 Hz. Additionally, the CPU was overloaded to verify the effectiveness of the proposed solution under extreme conditions.

Multiple ways of storing data were submitted to this methodology, such as to consider the use or not of threads, and data being stored in *MongoDB* right after being acquired by the sensors or only after a few seconds, being firstly stored locally in a buffer.

Results

The best-founded solution was the creation of a thread that, every five seconds, would insert the data acquired in those five seconds into the database. The other attempts were automatically excluded because data loss was significant or there was some program failure.

Table 6 presents the above solution results. The maximum data loss was verified for a session of sixty minutes in which the robotic walker acquired data at a sampling frequency of 200 Hz. This loss was 0.04%, which corresponds to 288 samples of the 720 000 that were expected. Still, it is noteworthy that the sampling frequency is not constant, it presents some fluctuations around the theoretical value, which can explain the data loss.

Table 6: Database Testing Results

Time \HZ	100			200		
30 min	1º	Expected	180 000	1º	Expected	360 000
		Obtained	179 950		Obtained	359 907
		Lost Samples(%)	0.0278		Lost Samples(%)	0.0258
	2º	Expected	180 000	2º	Expected	360 000
		Obtained	179 960		Obtained	359 951
		Lost Samples(%)	0.0222		Lost Samples(%)	0.0136
	3º	Expected	180 000	3º	Expected	360 000
		Obtained	179 929		Obtained	359 932
		Lost Samples(%)	0.0394		Lost Samples(%)	0.0189
60 min	1º	Expected	360 000	1º	Expected	720 000
		Obtained	359 915		Obtained	719 782
		Lost Samples(%)	0.0236		Lost Samples(%)	0.0303
	2º	Expected	360 000	2º	Expected	720 000
		Obtained	359 912		Obtained	719 712
		Lost Samples(%)	0.0244		Lost Samples(%)	0.04
	3º	Expected	360 000	3º	Expected	720 000
		Obtained	359 901		Obtained	719 817
		Lost Samples(%)	0.0275		Lost Samples(%)	0.0254

4.4.2 *Usability Testing*

Usability issues can be discovered by applying testing methods to end-users. Therefore, a set of tasks were delivered to future users of the application so that they could identify usability problems. The test was applied

not only to the new features, but to the entire application. Throughout this section, it is described in detail how the entire usability process took place.

Participants

Six health professionals, who were also involved in the requirements gathering phase (see Section 3.4.1), volunteered to participate in the usability experiment. Each participant gave their informed consent to participate in this evaluation. Of the six participants:

- All were female;
- Five of them currently work at the Hospital of Braga, and one work at the Hospital of Guimarães;
- One was a physiotherapist, two were physiatrists, and the others three were residents in the Physical Medicine and Rehabilitation Service;
- All are used to dealing with technologies daily, including at work;

Methodology

Since health professionals showed availability to perform the test in person, the chosen usability testing method was the *Think Aloud Protocol*. In this method, the participants are encouraged to verbalize their thoughts while performing the predefined tasks. They can explain their approach of attempting to complete the task, clarifying difficulties and consequently helping in finding the system problems [56][57]. Additionally, the computer's screen was recorded to evaluate the end-user's interaction with the application.

Moreover, it was decided to apply a standardized usability questionnaire at the end of the procedure, namely the Software Usability Scale (SUS) [58]. This is a simple ten-item open-source questionnaire, with a global view of subjective assessments of usability, being considered a valuable evaluation tool for its strength and reliability [59]. Since the participants' native language is Portuguese, a translation of the original SUS [60] was applied. In this questionnaire, the odd items advantage from higher levels of agreement, whereas the even items advantage from lower scores. The scoring system varies between 0-100, and it is calculated by subtracting the scale position by one in each odd item and, for the pair items, the score contribution is five minus the scale position. To get the final score, the sum between the odd and even contributions is performed and multiplied by 2.5 [58]. The described formula is presented below, where Qx represents the scale position for item number x:

$$FinalScore = [(Q1 - 1) + (Q3 - 1) + (Q5 - 1) + (Q7 - 1) + (Q9 - 1) + (5 - Q2) +$$

$$(5 - Q4) + (5 - Q6) + (5 - Q8) + (5 - Q9) + (5 - Q10)] * 2.5$$

Outlined Tasks

For the participants to have contact with all the application features, and have a complete navigation through it, ten tasks were outlined:

- **Task 1 -** Access the system with the following credentials:
  `username`: teste
  `password`: teste
- **Task 2 -** Insert the patient “Pedro Santos” who is a fictitious patient and has the following fictitious personal and clinical data:
  `ID`: HY78U
  `Gender`: Male
  `Year of birth`: 1966
  `Pathology`: Hemorrhagic stroke
  `Admission to therapy`: 29/06/2021
  `Shoulder distance`: 42 cm
  `Body mass`: 83 kg
  `Height`: 176 cm
  `Foot length`: 40 cm
  `Pelvis height`: 97 cm
  `Leg width`: 55 cm
  `Ankle height`: 15 cm
  Save the patient.
- **Task 3 -** Search for the patient whose ID is “HU87T”, insert the observation “New drug being administered since May.” and save the changes.
- **Task 4 -** Search for the patient whose ID is "HK33D", end the patient therapy and save.
- **Task 5 -** In the archive, search for the patient whose identification is “HP67F”, restart the patient therapy and save.
- **Task 6 -** Insert the following clinical evaluation to the patient whose identification is "HN55N":
  `SARA`: 32
  `TUG`: 2.9
  `BBS`: 74
  Save the clinical evaluation.
- **Task 7 -** Check the session information on May 6th at 14:19:47 for patient “HO99Y”.
- **Task 8 -** Return to the main menu. Set up a new rehabilitation session for patient “HW777” with the following configurations:

`Control`: Manual - maximum velocity of 3 km/h and security mode activated
`Balance`: Dynamic and biofeedback activated
`Sensors`: Use the pre-defined ones
Proceed to session start point.

- **Task 9 -** On the session page, access the balance biofeedback and assign the following threshold values:
  `Anterior limit`: 6º
  `Posterior limit`: -3º
  `Lateral limits`: 4º
- **Task 10 -** Start the rehabilitation session and reset the anterior limit in the biofeedback to 11º.

4.4.3 *Results*

Thinking-Aloud Protocol

Table 7 presents the results of the thinking-aloud protocol applied to health professionals. The table is divided by tasks. Each task has the participants' difficulties and suggestions, as well as the average execution time.

By analysing Table 7, several difficulties and suggestions were pointed out by the participants. The task with the longest duration was number two, which was expected since it was necessary to enter multiple data related to the patient.

Table 7: Thinking-Aloud Protocol Results

Tasks	Difficulties	Suggestions	Average Duration
1	· All participants performed this task directly without any difficulties.	· No suggestions.	9.20 $\pm$ 2. 37 seconds
2	· Two participants initially went to "Create Session" menu, later realizing it should be on "Patients" menu; · Four of the participants were hesitant to click on the save button. They claimed the icon was not intuitive enough; · One participant was unsure whether, on the therapy admission date, the input fields were in day/month/year or month/day/year format;	· One participant suggested changing the name "Height of the pelvis" to "Height of the pelvis (EIAS)" so that it would be clear to what height it was referring.	151.20 $\pm$ 0.65 seconds
3	· All participants performed this task directly without any difficulties.	· Two participants suggested that the dropdown for the patient search should also allow writing. This way the search could be done more quickly.	87.00 $\pm$ 19.94 seconds
4	· Two participants took considerable time finding the end-therapy button; · The other four participants performed the task directly without any difficulties.	Two participants suggested highlighting the end-therapy button.	36.82 $\pm$ 26.27 seconds
5	· Two participants, instead of going to the archive menu, searched for the patient as in task 3. They quickly realized the archive was not there and proceeded with the task successfully; · The other four participants performed the task directly without any difficulties.	· No suggestions.	28.58 $\pm$ 12.55 seconds

Table 7: Thinking-Aloud Protocol Results (continued)

Tasks	Difficulties	Suggestions	Average Duration
6	· All participants performed this task directly without any difficulties.	· No suggestions.	38.88 ± 9.94 seconds
7	· One participant initially went to the "Search Patient" menu instead of the "Session Information" one; · The other five participants performed the task directly without any difficulties.	· No suggestions.	38.48 ± 14.29 seconds
8	· Five participants had difficulties handling the algorithms' settings buttons; · Only one participant performed the task directly without any difficulties.	· Two participants suggested changing the settings button. · One participant suggested changing the balance settings from "Walking" to "Dynamic" and "Stopped" to "Static".	140.40 ± 70.20 seconds
9	· All participants performed this task directly without any difficulties.	· No suggestions.	37.70 ± 7.45 seconds
10	· All participants performed this task directly without any difficulties.	· No suggestions.	30.78 ± 16.12 seconds

SUS

After the participants had performed the defined tasks, it was delivered to them the SUS usability questionnaire. Table 8 presents a heat map with the participants' positive and negative reactions, and the individual and average scores obtained through answers. The lowest and highest scores were 57.5 and 92.5, respectively, with an average score of 84.17 ± 12.39.

Table 8: SUS Scores WALKit SmartW application

		Partipants Answers					
Questions	Questions	A	B	C	D	E	F
1	I think that I would like to use this system.	5	5	5	5	3	5
2	I found the system unnecessarily complex.	2	2	1	2	2	2
3	I thought the system was easy to use.	4	5	4	4	3	5
4	I think that I would need the support of a technical person to be able to use this system.	1	1	2	1	2	2
5	I found the various functions in the system were well integrated.	5	3	5	5	3	5
6	I thought there was too much inconsistency in this system.	1	1	1	2	3	2
7	I would imagine that most people would learn to use this system very quickly.	4	5	5	2	4	5
8	I found the system very cumbersome to use.	1	1	1	2	3	1
9	I felt very confident using the system.	4	5	4	5	3	4
10	I needed to learn a lot of things before I could get going with this system.	1	1	1	1	3	1
Score by participant		90.0	92.5	92.5	82.5	57.5	90.0

Average Score:	84.17 ± 12.39

In [61] is presented a study focused on the interpretation and grade of the scores obtained in the questionnaire. Figure 23 illustrates the grading scale.

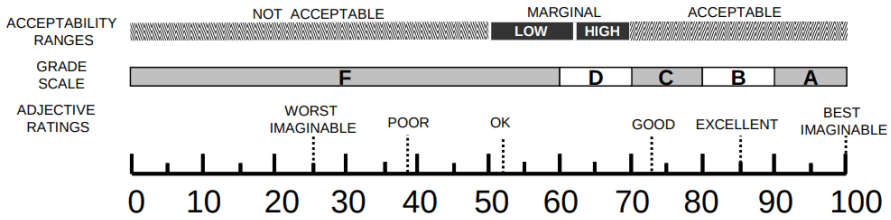


Figure 23: Grading Scale for the SUS. Retrieved from [61].

Considering the grading scale of Figure 23, as the average score is 84.17, it implies that the degree of usability for the application is excellent. Furthermore, by the analysis of Table 8, five of the six participants showed significant agreement on items number one and ten, enhancing the need for a system with these features, and showing great learnability by new users.

The participants' feedback after completing the questionnaire was also extremely positive. They reported that the application was, in general, very intuitive to use. Still, some slight modifications to the application were suggested, namely:

- The save patient button, which was changed to a diskette, which is more intuitive;
- An indication of the action was introduced for each button explaining what is their purpose;
- A caption Day/Month/Year was introduced on each date input fields;
- "Height of the pelvis" was modified to "Height of the pelvis (EIAS)", which is the abbreviation of "Espinha Ilíaca Ântero-Superior";
- All search lists have been changed to allow for written search;
- The end-therapy button was highlighted;
- In the balance algorithm settings, the term "Walking" was changed to "Dynamic" and "Stopped" to "Static".

4.5 GENERAL CONCLUSIONS

This chapter presented the development process that was carried out to optimize the existing application of WALKit SmartW. The application needed to be improved with new features and with security measures required to be enforced to keep patients' data secure, bringing satisfaction and trust to users. The design, implementation, and testing stages were applied in this development, resulting in a useful application that fulfilled all the requirements and necessities, being considered excellent in terms of usability. Database testing was made to ensure the acquired data was being saved correctly. It was verified that data was being stored properly with minimal data loss.

5

WEB APPLICATION

5.1 INTRODUCTION

This chapter will cover the design, implementation and testing stages for the development of the WALKit SmartW Web Application according to the Prototyping model. As stated previously, this application is intended to provide health professionals a way to control and manage rehabilitation sessions using the robotic walker in a clear, practical and intuitive way. The application development process considered the requirements presented in Section 3.4.

5.2 APPLICATION OVERVIEW

As previously mentioned in this document, the main purpose of this web application is to display the data acquired from the robotic walker sensors in an efficient, intuitive, and user-friendly way, making it possible for health professionals to have a better perception about the patients' progress throughout therapy. Moreover, it should also allow managing the patients' personal and clinical information in a secure environment.

Considering this and based on the requirements presented in Section 3.4, the different functionalities of the application were structured and formalized in a global use case diagram.

5.2.1 *Use Cases*

Figure 24 presents the possible actions throughout the system by three different users: health professionals, administrators and patients. In the case of the healthcare professional user, the actions include: **(i)** add, edit, delete, archive and unarchive a patient, **(ii)** add clinical evaluation scales, **(iii)** add patient fall, **(iv)** check rehabilitation sessions details, **(v)** check rehabilitation sessions temporal evolution, **(vi)** delete sessions, **(vii)** download rehabilitation session data and report, and **(viii)** manage personal profile. Regarding the Administrator user, he can perform the same actions as the health professional user and additionally can manage (*i.e.*, add, edit and delete) users. Lastly, the user Patient can only manage his profile and check his rehabilitation sessions details and evolution.

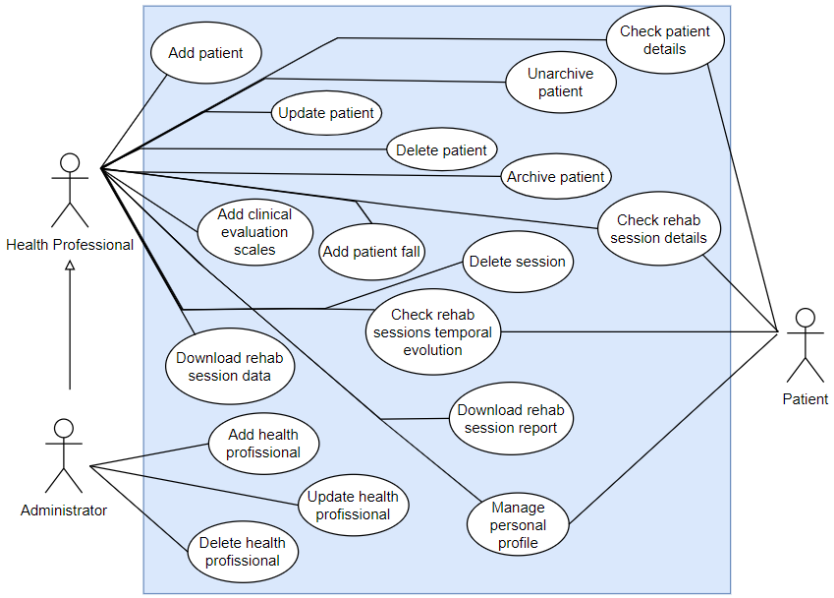


Figure 24: New Application Global Use Case Diagram.

5.3 DESIGN

Following the SDLC Prototyping model, the design phase has two stages: a simple design that gives a brief idea of the system, and a prototype, which is a working model of the required application. The designed wireframes and prototypes used to guide this development are presented in detail in this section.

5.3.1 *Wireframes*

As an initial stage of the development process, wireframes were created using the *Balsamiq* tool. These wireframes were designed for computer screens, but all the logic behind them, was planned to be responsive to various electronic devices.

Three sets of wireframes were created, corresponding to the three types of users. In the case of administrators and healthcare professionals, the wireframes are identical, only changing the user management menu since this functionality is only allowed by administrators. On the other hand, the patient's wireframes suffer some changes, as patients should only have access to their information. Nevertheless, the login page (Figure 25) is the same for the three types of users and is responsible for accessing to the system.

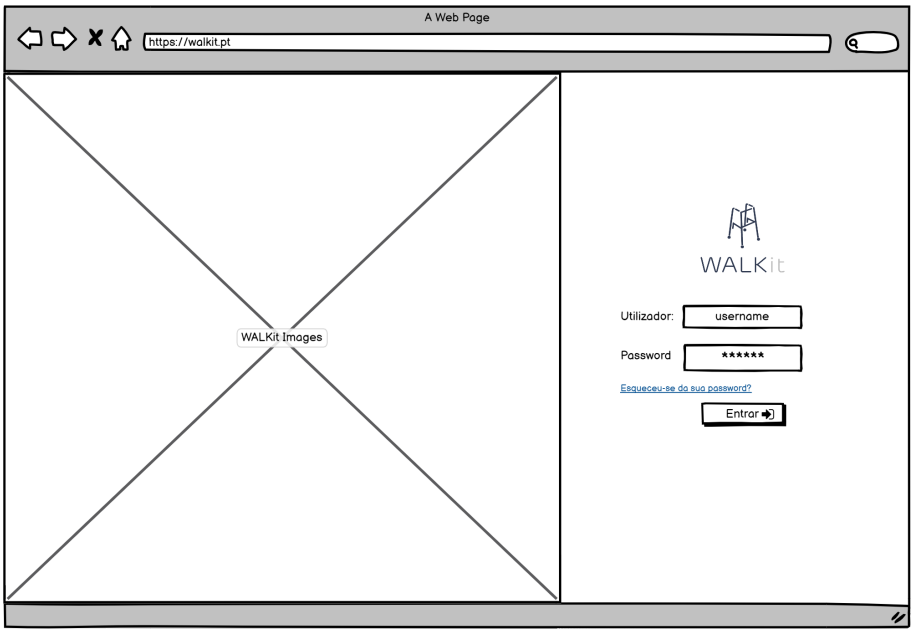


Figure 25: Login Page wireframe.

Administrator and Health Professional Wireframes

One of the main goals was to provide fast, valuable and intuitive navigation, so the design was made to ensure simplicity throughout all the menus. Figure 26 shows the home page. The home page is composed of some statistics regarding patients, sessions and professionals, and acts as a starting point for all other menus.

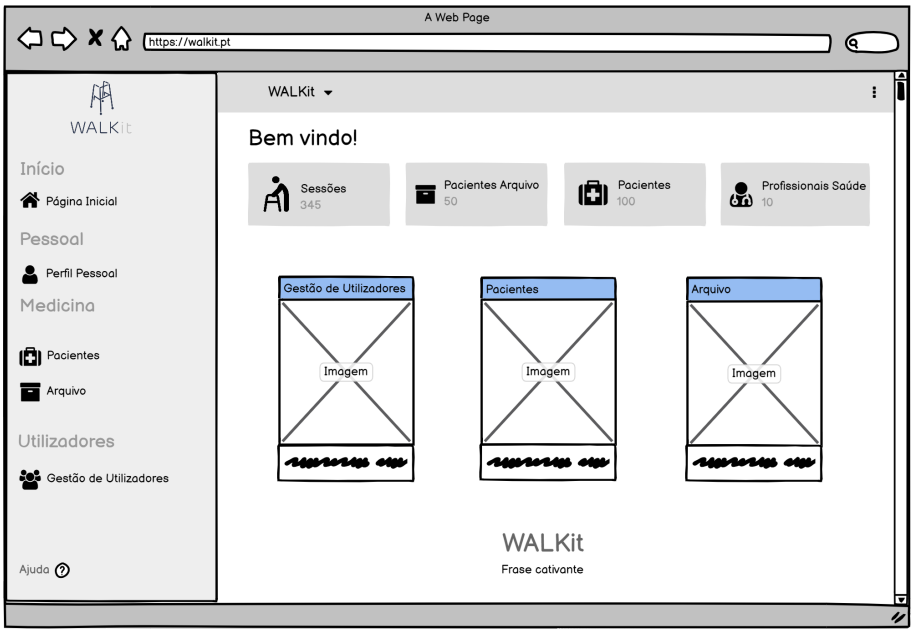


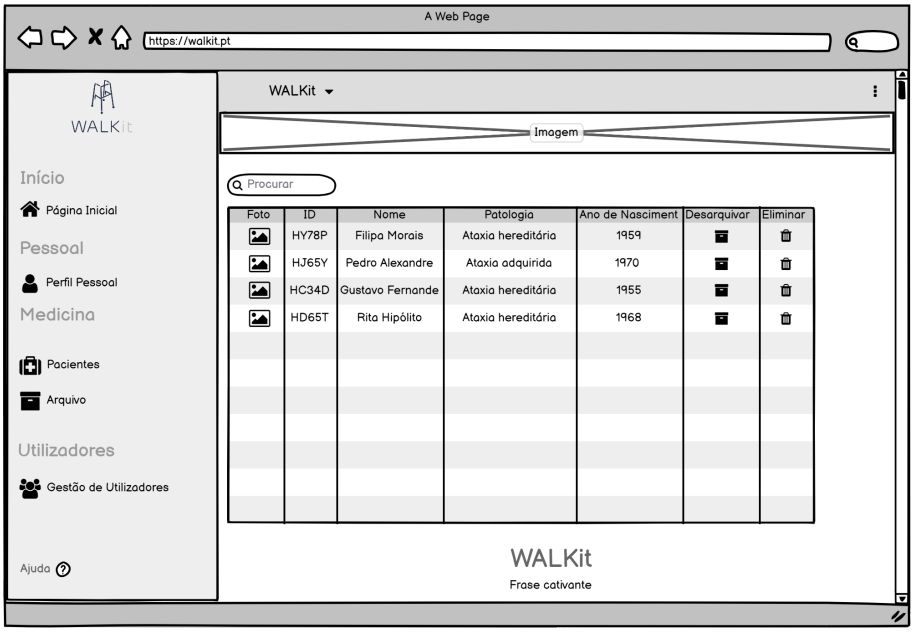
Figure 26: Home Page Wireframe.

Figure 27 illustrates the profile page. It was designed to allow users to check or update their personal data.



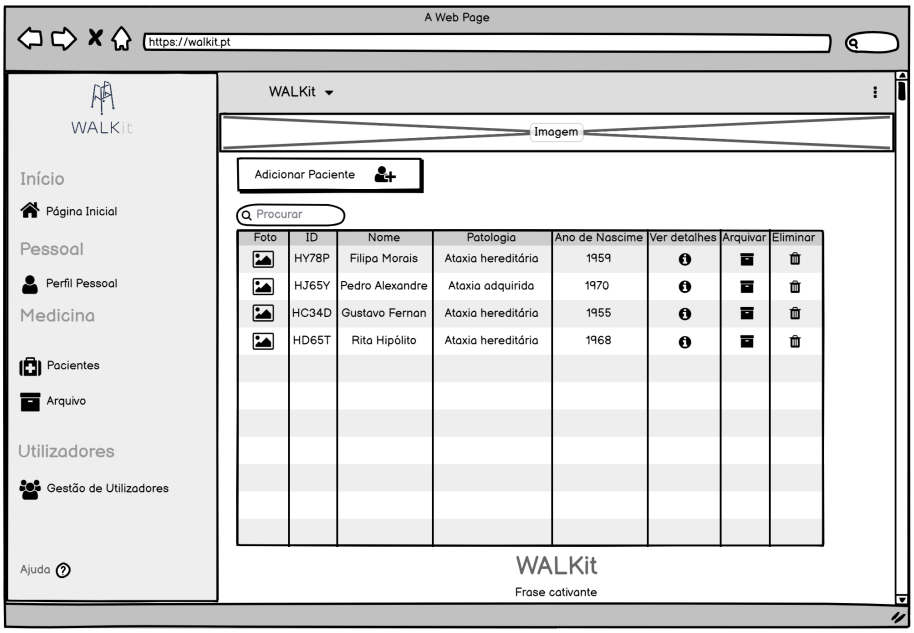
Figure 27: Profile Page Wireframe.

For patients' management, two menus were created: one to deal with patients who had already finalized treatments (archive), and the other responsible for managing patients undergoing therapy.



(a)

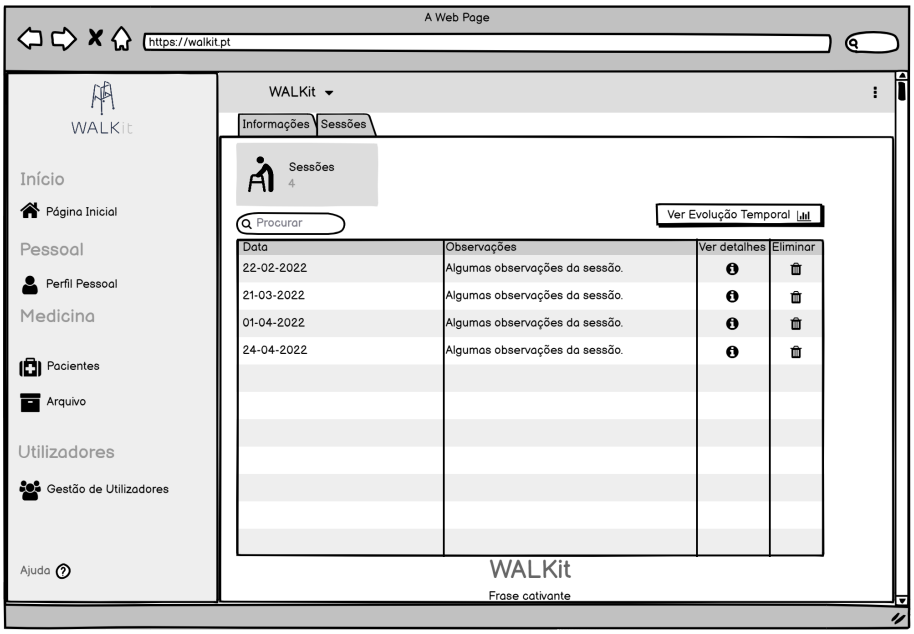
Figure 28: Patients' Management Wireframes: (a) Archive Management; (b) Management of patients' undergoing therapy.



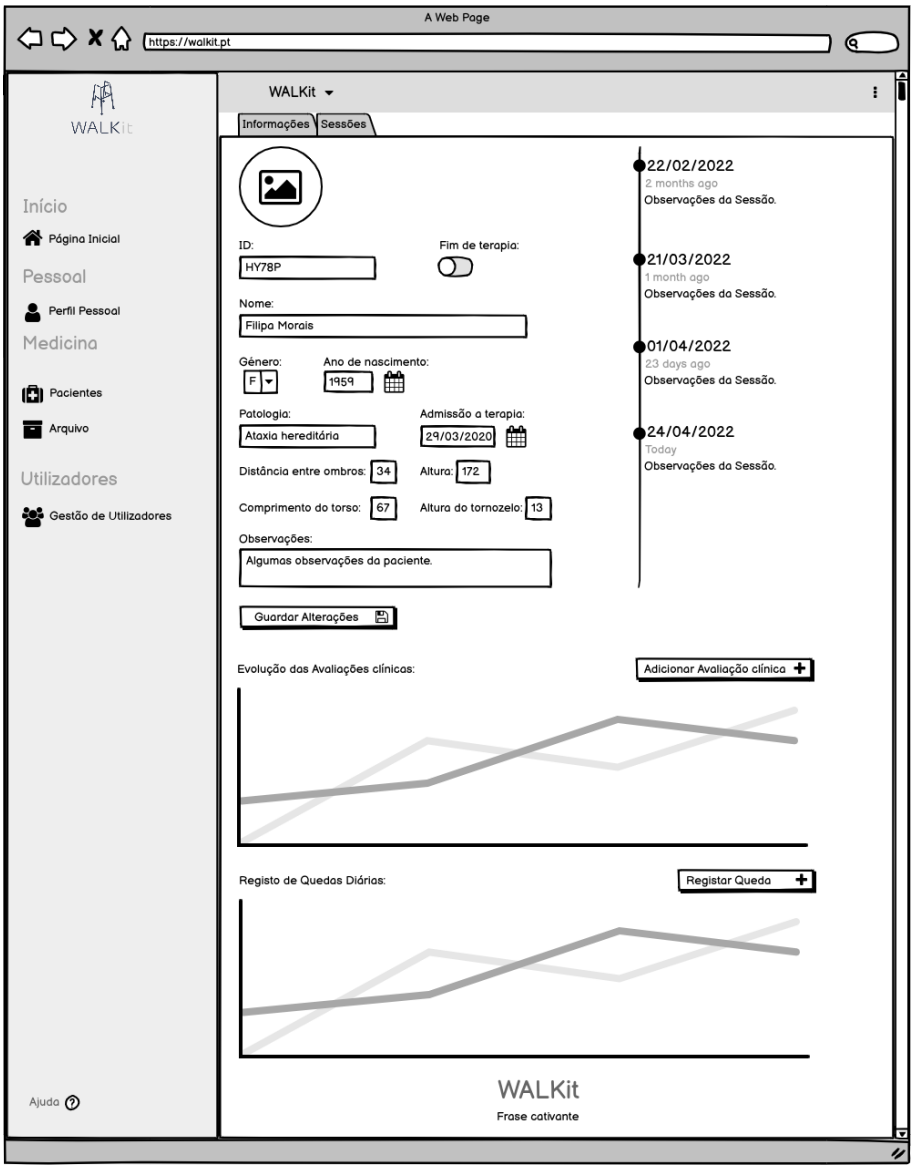
(b)

Figure 28: Patients' Management Wireframes: (a) Archive Management; (b) Management of patients' undergoing therapy (continued).

As shown in Figure 28, both menus were designed with tables, providing better organization and promoting a more effective search. Each row represents a patient, and allows to archive/unarchive, delete or view the details.



(a)



(b)

Figure 29: Patient details wireframes: (a) Patient' personal and clinical details; (b) Patient' sessions.

The patient details focus on his personal data, including anthropometric data, clinical scales and fall records, and on his rehabilitation sessions. Figure 29 shows the designed pages for displaying this information.

For displaying the data acquired from the walker sensors, two menus were idealized: one representative of each individual session, and the other with the evolution over time. Figure 30 shows both menus. Cards were used to display the session configurations, allowing a more appealing view, and charts were drawn to allow a more practical and intuitive analysis of the patient progress.

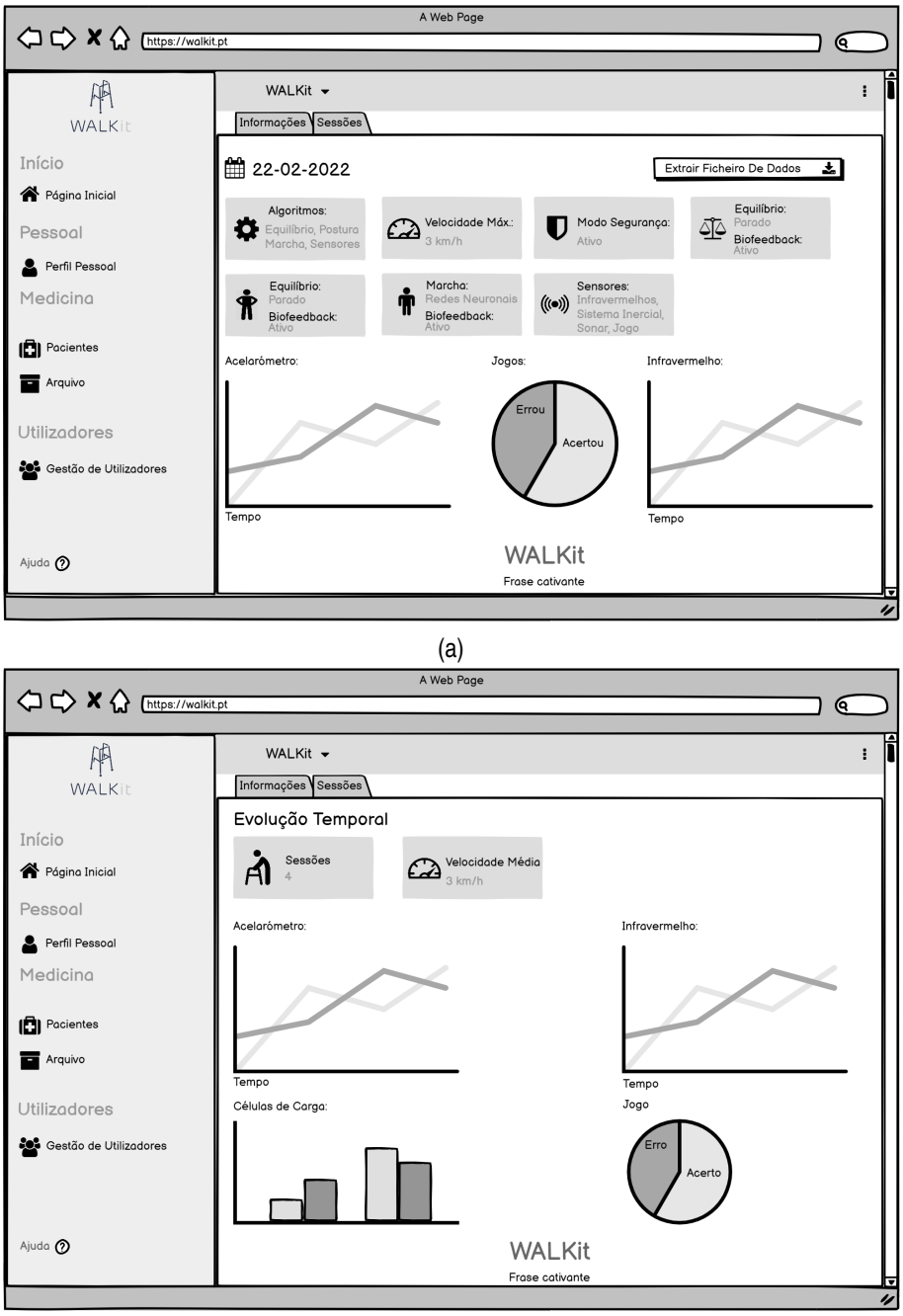


Figure 30: Sessions wireframes: (a) Session details; (b) Temporal evolution.

Lastly, the user management menu (only available for administrators) was designed to allow the insertion, update or removal of the system users (Figure 31). As it is possible to see throughout the wireframes, the left side of each page has a drawer menu, which was designed to ensure a faster and more efficient navigation.

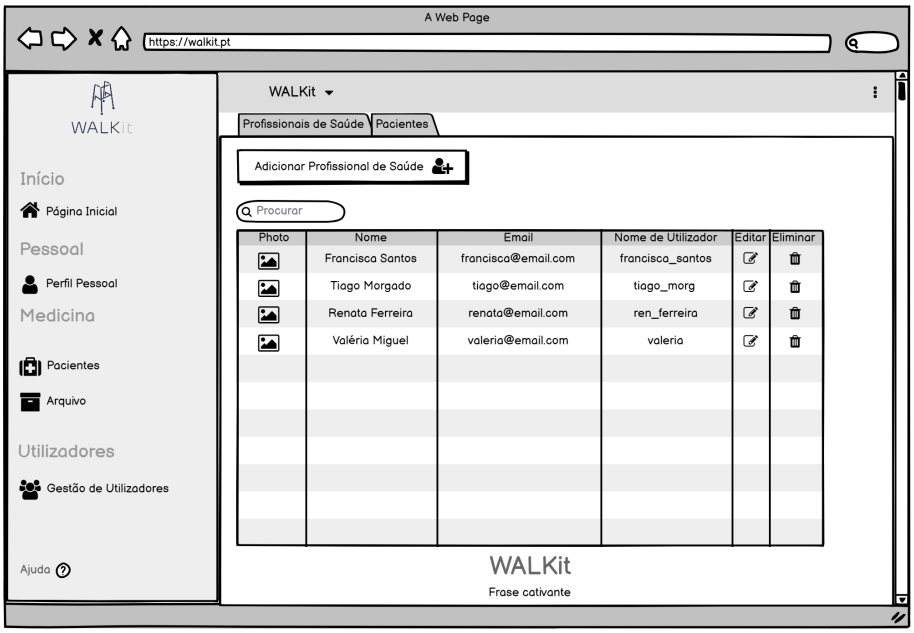


Figure 31: User Management Wireframe.

Patient Wireframes

The main idea behind the patient profile is to give patients the opportunity to analyse their progress, keeping them motivated during therapy. Each patient must only have access to their own information. Therefore the wireframes are restricted to the homepage, the personal information and the rehabilitation sessions pages. The created wireframes are very similar to the ones presented above and can be seen in more detail in the Appendix B.

Results

After completing this phase, the wireframes were presented to two healthcare professionals who were willing to give their opinion. They approved the design and were very positive about the outcome, turning the green light for the prototyping phase.

5.3.2 *Prototypes*

Keeping the wireframes in mind, prototypes were created to make the product close to the final application. All the visual details, including colors, icons, dialogue boxes, images, among others, are covered in this phase, allowing future users to experience how the application will look and behave.

To design the prototypes, the *Adobe XD* was used. This tool allowed the customization of the visual components with more precision and the creation of transitions through the menus, making it possible to idealize the users' flow using the application. As for the wireframes, three sets of prototypes were prepared.

The three menus which provide more information are the page with patient clinical data (Figure 32), temporal evolution (Figure 33) and individual session details (Figure 34). The others are very similar to the wireframes and can be consulted in the Appendix C. Note that the data shown is for visualization only and does not correspond to actual data from any patient or rehabilitation session.

The data displayed in the prototypes is fictitious.

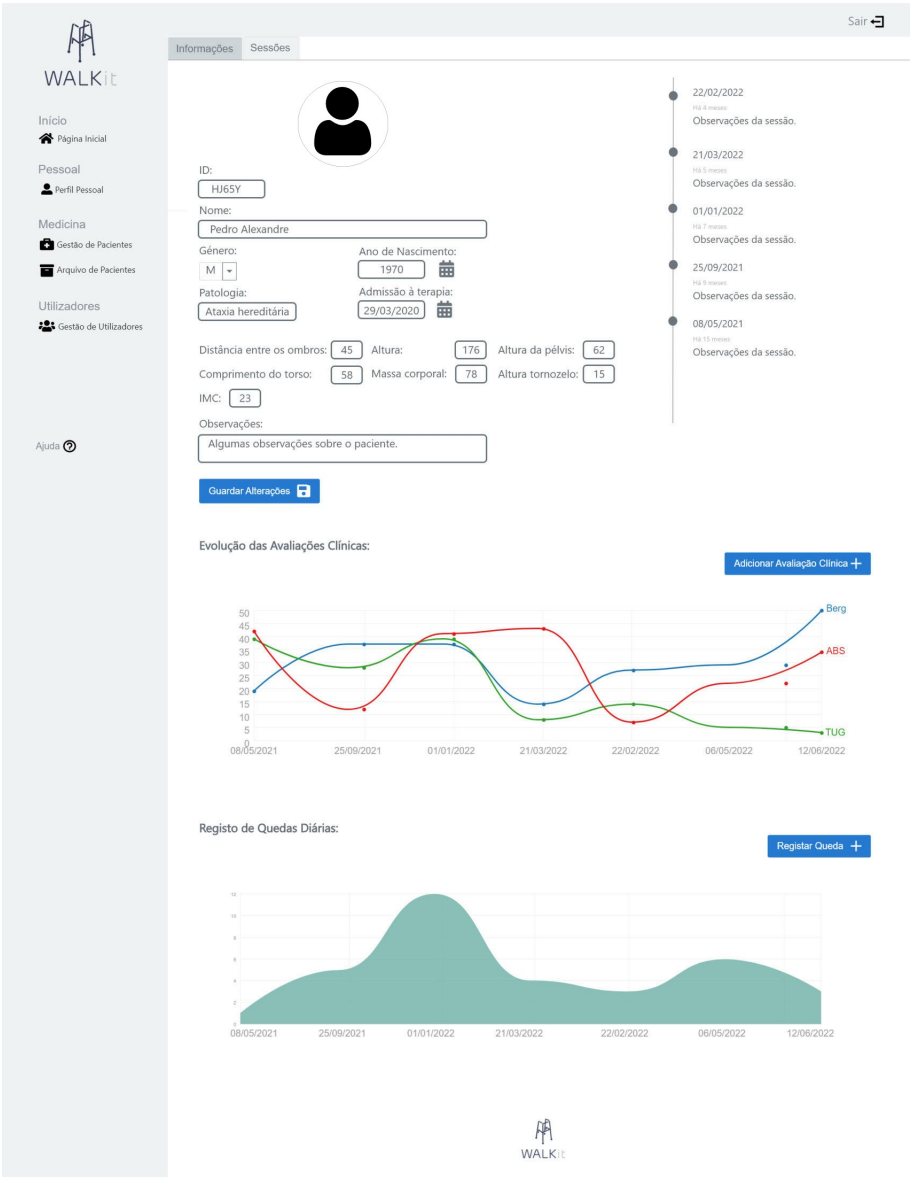


Figure 32: Prototypes - Patient details.

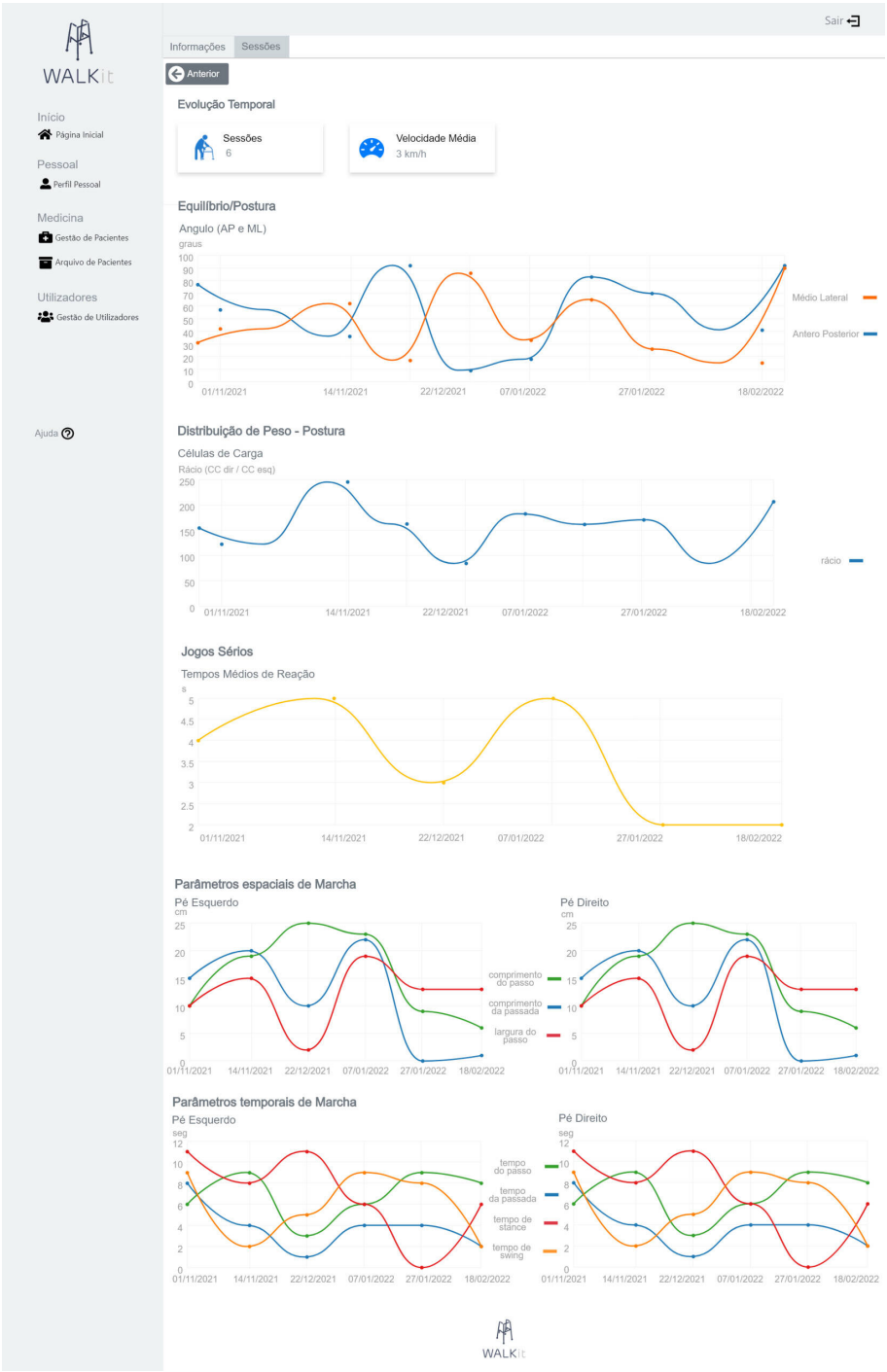


Figure 33: Prototypes - Temporal Evolution.

As seen in the prototypes images, a design choice was the light theme so that a clear and faster page interpretation could be achieved. Moreover, charts are presented in a colorful mode to provide a user-friendly analysis experience.

It is also important to mention that the charts presented in the prototypes are the ones defined with the medical team to analyse the patient progress. They are mainly intended to monitor the patients' posture, balance and gait during each rehabilitation session, and to display a temporal evolution. Additionally, metrics from serious games embedded in the SmartW software are also presented in a pie chart form.

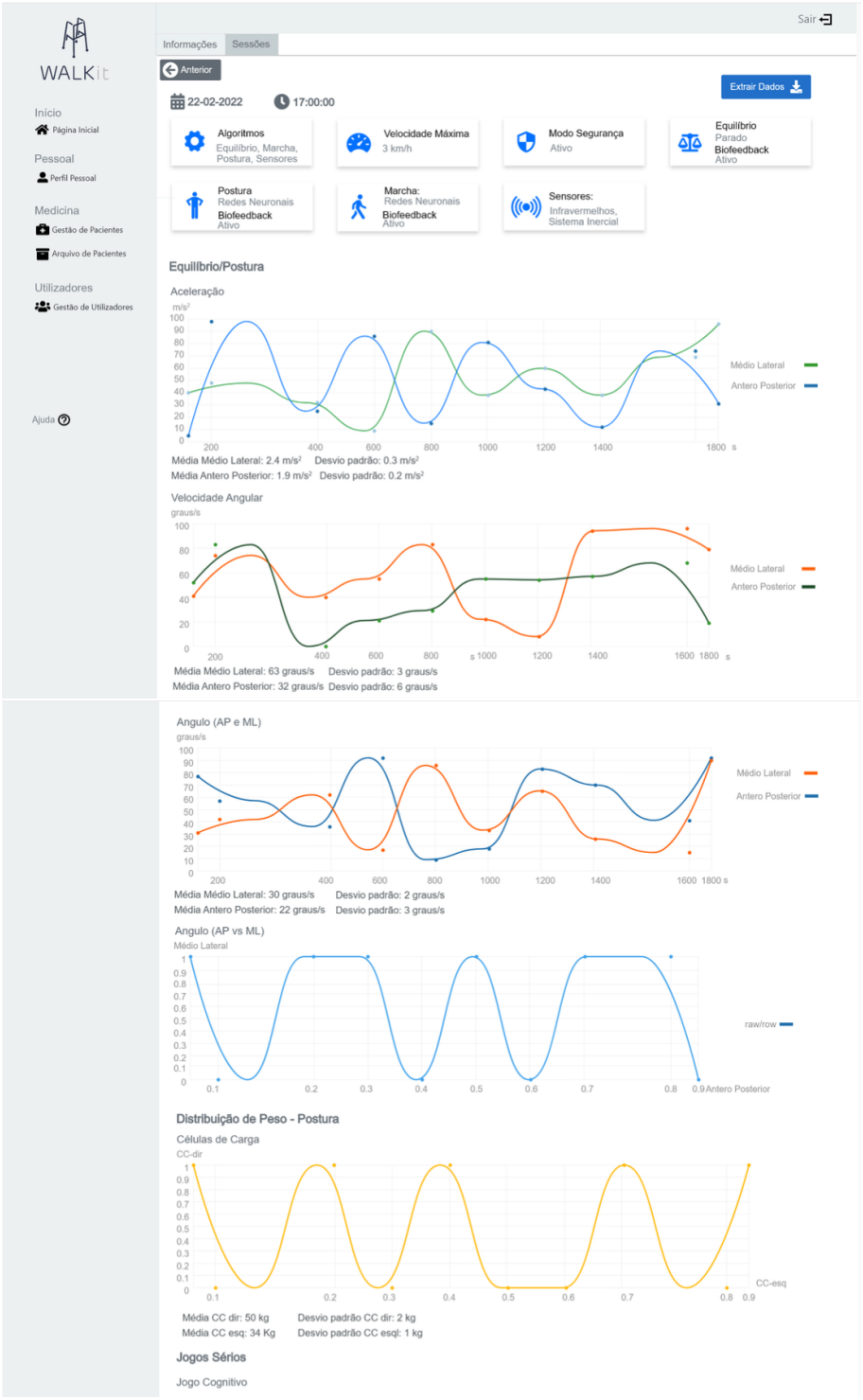




Figure 34: Prototypes - Individual Session Details.

5.4 EVALUATION

After the prototyping phase, and following the SDLC Prototyping model, the user evaluation was required to check if the designed system fulfilled all the expectations and needs. Therefore, and to avoid major usability problems that can be very costly, usability testes were applied in this phase.

5.4.1 *Participants*

The participants for the prototype usability test were expected to be healthcare professionals and patients undergoing therapy. Five health professionals, accepted voluntarily to participate. Unfortunately, it was not possible to have patients participating in the experiment because, at the time, there were no patients undergoing treatments with the robotic walker. Each participant gave their informed consent to participate in this evaluation.

Of the five participants:

- All were female;
- All currently work at the Hospital of Braga;
- One was a physiotherapist, one was a physiatrist, and the others three were residents in the Physical Medicine and Rehabilitation Service;
- All are used to dealing with technologies daily, including at work;

5.4.2 *Methodology*

The methodology used was similar to the one applied in the WALKit SmartW embedded application: first, a usability testing method was applied, and then a standardized usability questionnaire was given to the participants. Nevertheless, some health professionals were not able to take the test in person due to schedule restrictions. Therefore, two testing methods were resorted: the Think Aloud Protocol [56][57] and Remote Usability Testing [57]. The first one was performed by two participants who had availability to do in-person, and the second one by the other three participants via online meeting. At the end all participants fulfilled the SUS questionnaire [58][60].

Each participant interacted with the prototypes using the *Maze* tool, which is an online platform that allows to perform usability tests, in which one task is considered a mission to overcome. This tool registers the users' paths, clicks, and the duration of each task, providing an overall performance of the usability test. Additionally, this tool allows tests to be accomplished in a remote environment.

Outlined Tasks

Eight tasks were outlined for the usability test to allow a full experience with the prototypes, as follows:

- **Task 1 -** Imagine entering the system with your credentials to access its different features. Please show how to do it.
- **Task 2 -** Imagine that it is necessary to insert a new user so that he also has access to the system. Please show how to do it.
- **Task 3 -** Imagine that it is necessary to add a patient to the system, including their personal and anthropometric data. Please show how to do it.
- **Task 4 -** Imagine that you want to access the patient "Pedro Alexandre", check his information/details and add a clinical scale (ABC). Please show how to do it.
- **Task 5 -** Imagine that you want to record the number of falls the patient has had on a given day. Please show how to do it.
- **Task 6 -** Imagine that you want to check the patient's sessions and access their temporal evolution. Please show how to do it.
- **Task 7 -** Imagine that you want to check the session on 22/02/2022 for the same patient. Please show how to do it.
- **Task 8 -** Imagine that you want to check which patients are stored in the system. Please show how to do it.

5.4.3 *Results*

Thinking-Aloud Protocol and Remote Usability Testing

Table 9 shows the results of the thinking-aloud protocol and the remote usability testing applied to health professionals. The table is divided by task. Each task is composed by the difficulties and suggestions of the participants, as well as the average execution time.

Table 9: Usability Test Results

Tasks	Difficulties	Suggestions	Average Duration
1	· All the participants performed this task directly without any difficulties.	· No suggestions	19.55 $\pm$ 4.77 seconds
2	· One participant instead of going to the user management page went to the patients' page, later realizing it was not there; · The other participants completed the task directly without any difficulties.	· No suggestions	23.16 $\pm$ 19.09 seconds
3	· All the participants performed this task directly without any difficulties.	· No suggestions	13.5 $\pm$ 5.23 seconds
4	· One participant misclicked the page and went back to the user management page, claiming it was a lack of attention; · The other participants completed the task directly without any difficulties.	· No suggestions	57.59 $\pm$ 37.40 seconds
5	· One participant thought the number of falls would be recorded in the patient's observations, later realizing there was a specific zone for that purpose; · The other participants completed the task directly without any difficulties.	· No suggestions	19.97 $\pm$ 13.02 seconds
6	· All the participants performed this task directly without any difficulties.	· No suggestions	29.00 $\pm$ 19.29 seconds
7	· All the participants performed this task directly without any difficulties.	· No suggestions	32.70 $\pm$ 20.82 seconds
8	· All the participants performed this task directly without any difficulties.	· No suggestions	9.52 $\pm$ 8.28 seconds

SUS

After the usability test, the participants fulfilled the SUS usability questionnaire. Since some of the tests were online, a *google form* was created with the standardized questionnaire.

Table 10 presents a heat map with the individual answers and the average score. The lowest and highest scores where 90 and 97.5 respectively, with an average of 94 $\pm$ 3.

Table 10: SUS Scores Prototypes

		Partipants Answers				
Questions	Questions	A	B	C	D	E
1	I think that I would like to use this system.	5	5	5	5	5
2	I found the system unnecessarily complex.	2	1	1	2	1
3	I thought the system was easy to use.	5	5	5	4	5
4	I think that I would need the support of a technical person to be able to use this system.	1	1	1	1	1
5	I found the various functions in the system were well integrated.	4	5	5	5	5
6	I thought there was too much inconsistency in this system.	1	1	1	1	1
7	I would imagine that most people would learn to use this system very quickly.	5	4	3	4	4
8	I found the system very cumbersome to use.	1	1	1	1	1
9	I felt very confident using the system.	4	5	5	4	5
10	I needed to learn a lot of things before I could get going with this system.	1	1	2	1	1
Score by participant		92.5	97.5	92.5	90.0	97.5

Average Score:	94.0 ± 3.0

Considering the grading scale presented in Figure 23, as the average score is 94, it means the degree of usability for the prototypes is between the "excellent" and "best imaginable" range. Furthermore, it is possible to conclude about the positivity of the participants, since the degree of agreement was mostly high. The feedback was very optimistic, and no suggestions were given. The patients' prototypes were also shown the health professionals and the feedback was also promising.

5.5 IMPLEMENTATION

Since the feedback obtained from health professionals about the the prototype's evaluation was very positive, and no changes or suggestions were requested, it was possible to start the implementation phase and turn the prototypes into a real application. The following subsections will detail the technologies used to develop the web application, as well as information regarding both backend and frontend development. Additionally, illustrations are given regarding the final look of the web application.

5.5.1 *Technologies*

Since the development is focused on a Web application, the *React* library was chosen for the interface creation, with the aid of the *Material UI* for page styling. *React* is an open-source JavaScript library used for building user interfaces in web environments. It allows the creation and management of all the visual components and enables the creation of responsive design, making it possible to address the application on different devices. For the backend development it was decided to use *NodeJS* with the *Express* framework. It is also a JavaScript library which provides features for building web applications. The asynchronous behind this framework increases the program efficiency, which can be useful while handling large amounts of data. Moreover, *React* and *NodeJS* are extremely used for developing web applications, having a large community of developers, and, consequently, pro-

viding a greater support. In addition to these technologies, *Nginx* and *Docker* were resorted for the deployment phase.

5.5.2 *Application Architecture*

The application follows a client-server architecture with requests managed through HTTP in the JSON format. The backend, which was developed in *NodeJS*, is the server side and is composed of two different microservices. The services are independent of one another, and each one communicates with its own database. In this way, an easier development, management, and maintenance can be achieved since each microservice is autonomous. Moreover, the clinical data is separated from the personal patients' data, reducing the value of data loss if a single service is compromised. The frontend, which was developed in *React*, is the client side and is responsible for providing an intuitive user interface for accessing and managing the users' requests. Since different servers are used, the *Nginx* is used as a reverse proxy, redirecting the client requests to the appropriate backend server and simultaneously protecting the web server's identity from the user. Lastly, each component has its own docker image, allowing a faster and more reliable deployment. Figure 35 is representative of the described application architecture.

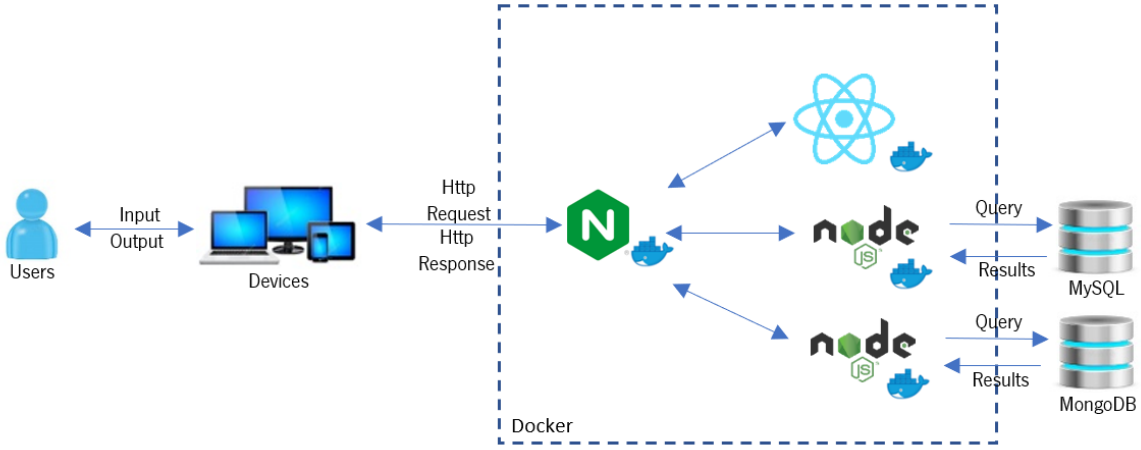


Figure 35: Web Application Architecture.

5.5.3 *Backend*

As previously stated, the backend is composed of two independent microservices. Each one is responsible for handling client requests by communicating with the appropriate database. The servers structure consists of the following:

- *Routes*: responsible for forward the client HTTP requests to the appropriate controller functions;

- *Controllers*: where the functions to get the requested data from the models or directly from the database are defined;
- *Models*: only used in the microservice which connects with the *Mongo* database, and is responsible for defining the database structure through different files;
- *Middleware*: responsible for the database connection and user authorization.

Figure 36 is representative of the servers' structure. The left one represents the server, which is connected with the *MySQL* database and is responsible for dealing with requests regarding users, patients, and session configurations. The right one is combined with *MongoDB* and is mainly responsible for preparing the required sensor data to appear on the evolution charts.

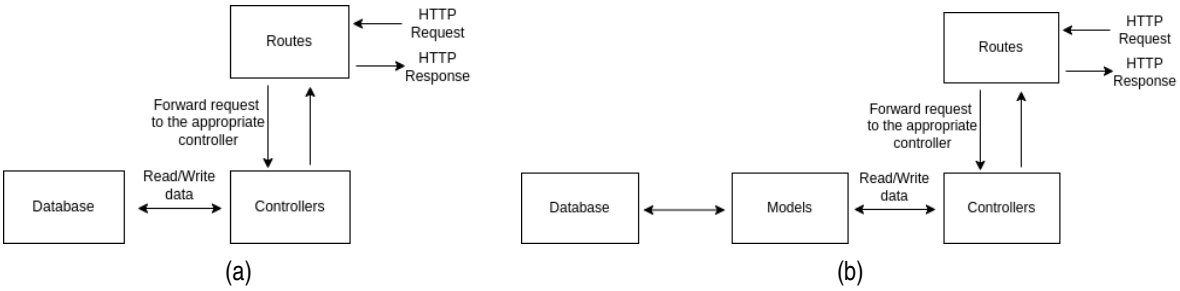


Figure 36: Servers Structures: (a) Structure of the server which is connected with the *MySQL* database; (b) Structure of the server which is connected with the *MongoDB* database.

To ensure the system security, the backend provides mechanisms to prevent abusive and malicious behavior. Similar to the WALKit SmartW application, *Argon2* is used for password hashing, *i.e.*, keeping the password save in an unreadable state. Moreover, AES in CBC mode is also used for data encryption and decryption. The authorization and authentication are ensured in every client request with the help of a JSON Web Token (JWT), keeping the data transmission secure. If a user succeeds in logging into the system, a JWT is generated and included in each subsequent request. The backend is responsible for guaranteeing that it is valid and, if so, returns the information requested by the user.

5.5.4 *Frontend*

The frontend is responsible for providing an intuitive and user-friendly interface for accessing the patients' clinical information and evolution throughout the rehabilitation sessions. The design was totally based on the prototypes, assuring the user trust and satisfaction. Throughout this section, it will be possible to see the different interface menus resulting from this web application development. It is noteworthy that all the information presented in the application figures is simulated and does not represent clinical data of any patient or from rehabilitation sessions.

At first, when the user launches the application, a login menu is displayed, and the user credentials are required, *i.e.*, the username and password. If the login credentials match those stored in the database, the login

is successful, allowing the user to have access to the system's features. Figure 37 illustrates the login menu in a desktop screen.



Figure 37: Login Menu.

Considering the user is an administrator, the first page after signed in into the application is the homepage and is displayed in Figure 38. This page provides some statistics regarding sessions, patients, and health professionals and allows users to access the other menus, which includes:

- ***Perfil Pessoal***: the personal profile, where the user can have access and update his personal information;
- ***Gestão de Pacientes***: patients management, which allows the user to manage all the patients undergoing therapy, including their rehabilitation sessions;
- ***Arquivo***: archive, which allows the user to manage the patients who had already finished treatments;
- ***Gestão de Utilizadores***: users management, where the user can manage all the system users, whether they are patients or health professionals.

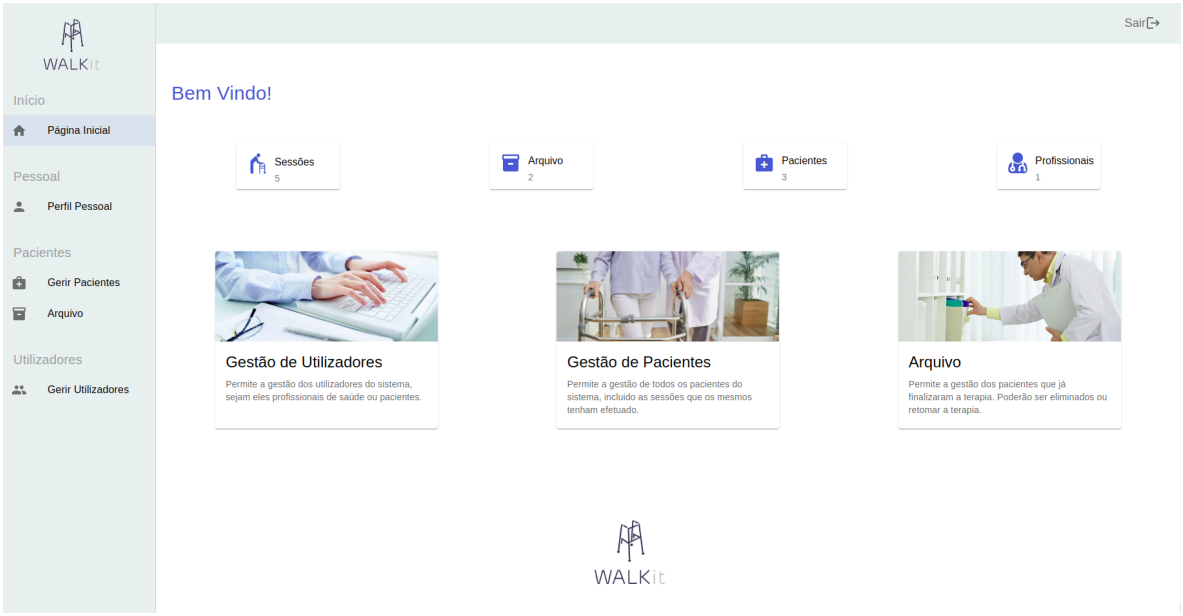


Figure 38: Administrator Homepage.

The personal profile menu, displayed in Figure 39, allows users to change their username, name, email, password and picture. As a security measure, if a user wants to update his profile, he has to confirm his password. Moreover, the input fields are always validated, not only the ones in the profile page, but along the application, preventing SQL injections.

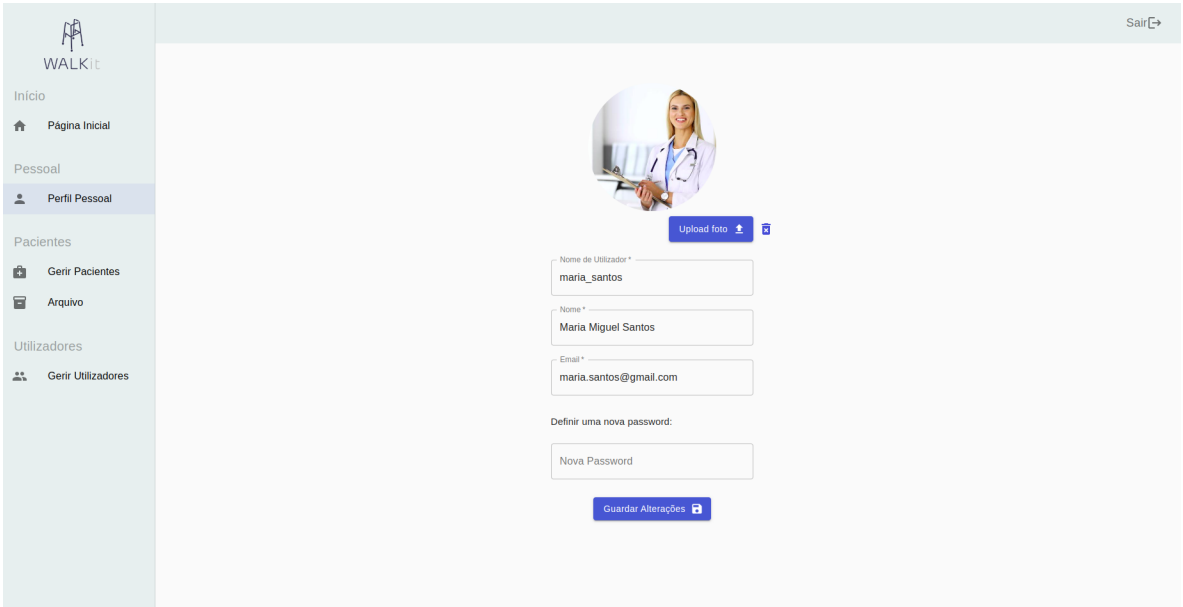


Figure 39: Personal Profile Menu.

The patients management menu (Figure 40) is composed by a table with patients undergoing therapy, which displays their main information and allows to perform some actions, including deleting, archiving or checking

their clinical details. In this menu, it is also possible to insert a new patient who will start therapy with the robotic walker. For that, it is necessary to insert his clinical information in the form presented in Figure 41. Furthermore, a patient account is automatically created, which allows them to check about their sessions and information.

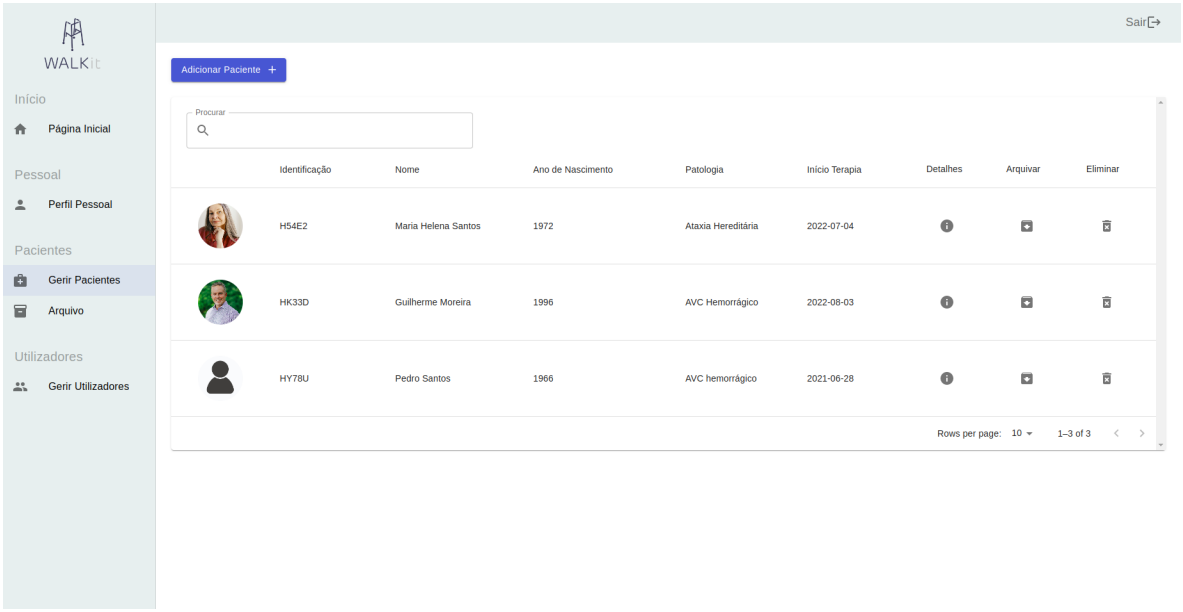


Figure 40: Patients Management Menu.

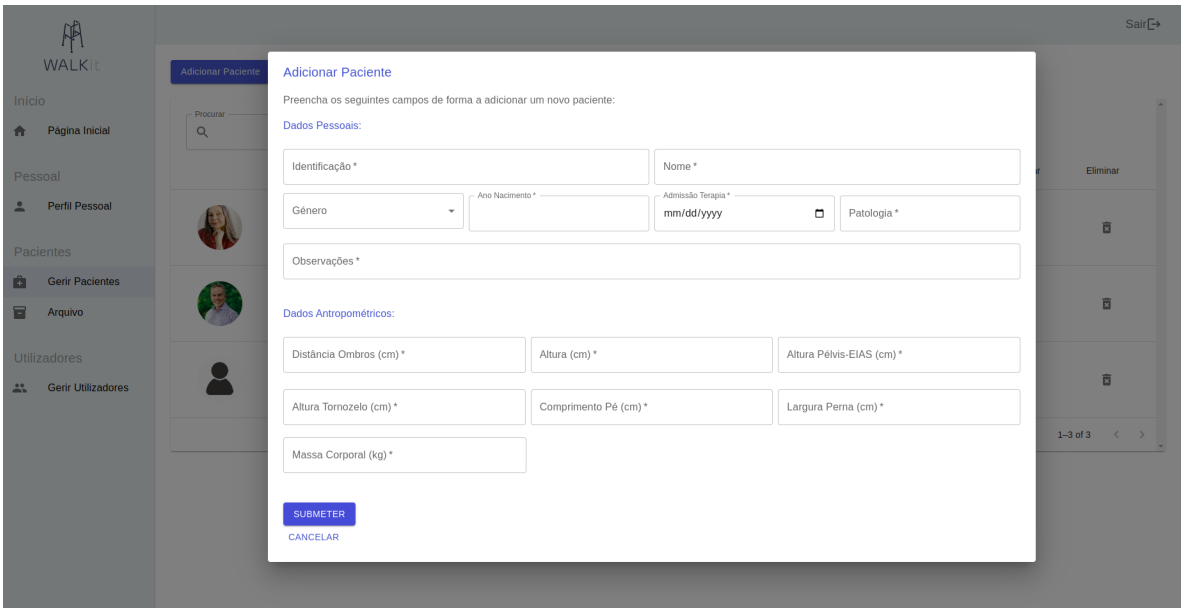


Figure 41: Insert Patient Modal.

Clicking on the details button, it is possible to visualize the patient's personal and anthropometric data, followed by a line chart with clinical rating scales and an area chart with the patient falls evolution. The patient's data can

be updated, if needed, but only by a health professional, and clinical scales or fall records can be added. Figure 42 shows the described menu.

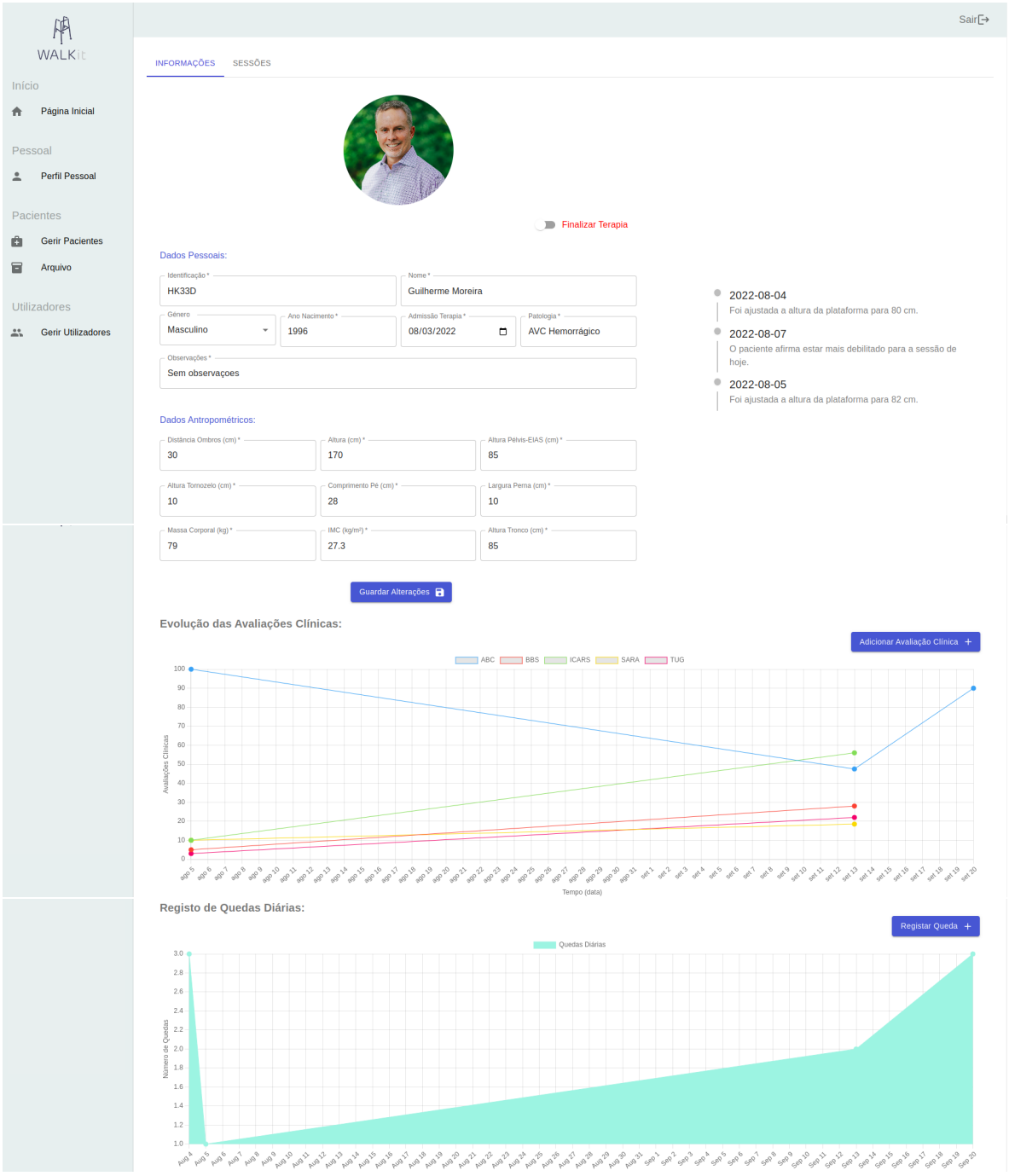
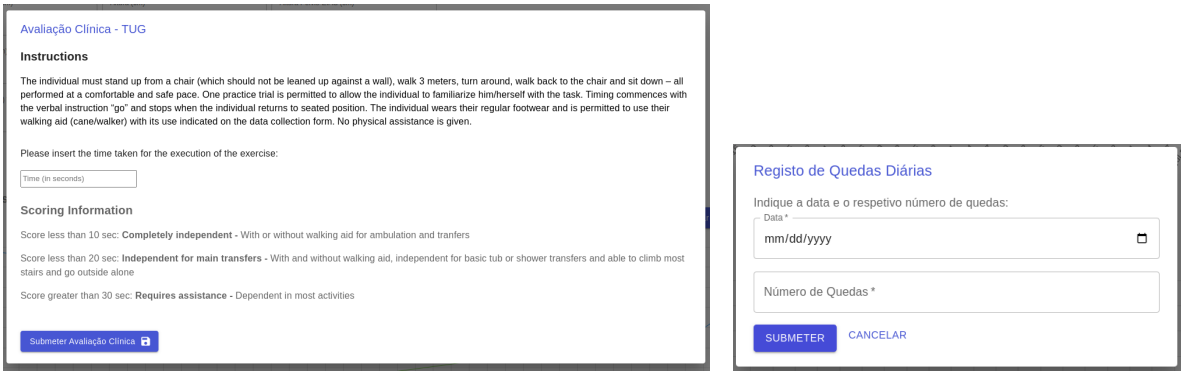


Figure 42: Patient Details.

Five clinical rating scales are being considered, each one with its completion form: Activities-specific Balance Confidence (ABC), BBS, SARA, International Cooperative Ataxia Rating Scale (ICARS) and TUG. After selecting the required scale, a dialog form appears with the different questions, and a score is calculated depending on

the answers. In the case of the fall records, the date and number of falls must be filled in the form. Figure 43 displays the TUG and the falls' registration forms.



(a) (b)

Figure 43: (a) TUG clinical scale modal form. (b) Fall record modal form.

Moving to the sessions tab (Figure 44), a list of the patient sessions is displayed along with some information, namely date, time and observations. It is possible to check the details of each session, delete a session, or check the temporal evolution throughout therapy.

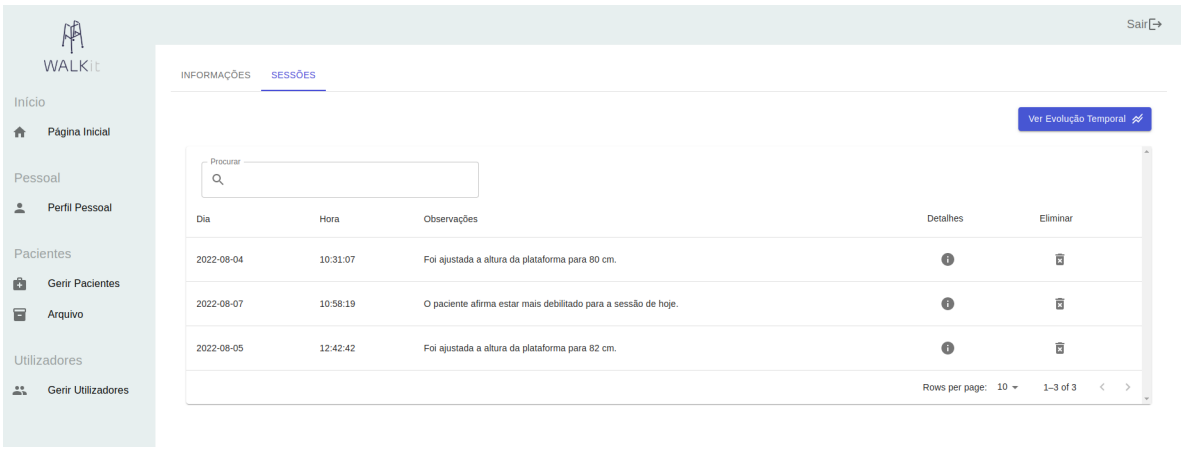


Figure 44: Sessions menu.

Clicking on the details button, a page with the session details appears. As shown in Figure 45, the page begins with cards which contain information about the session's configurations, namely the chosen algorithms, security mode, sensors, among others. Depending on the settings, the charts will change accordingly. For example, if in a session, the posture algorithm was not chosen, the charts with posture data will not appear on the session details. Each chart is constructed with the data acquired by the sensors, allowing clinicians to have a better perception of the patient's evolution during the session. The button on the upper right allows the download of a *csv* file with the sensors' data or a *pdf* with a copy of the page.

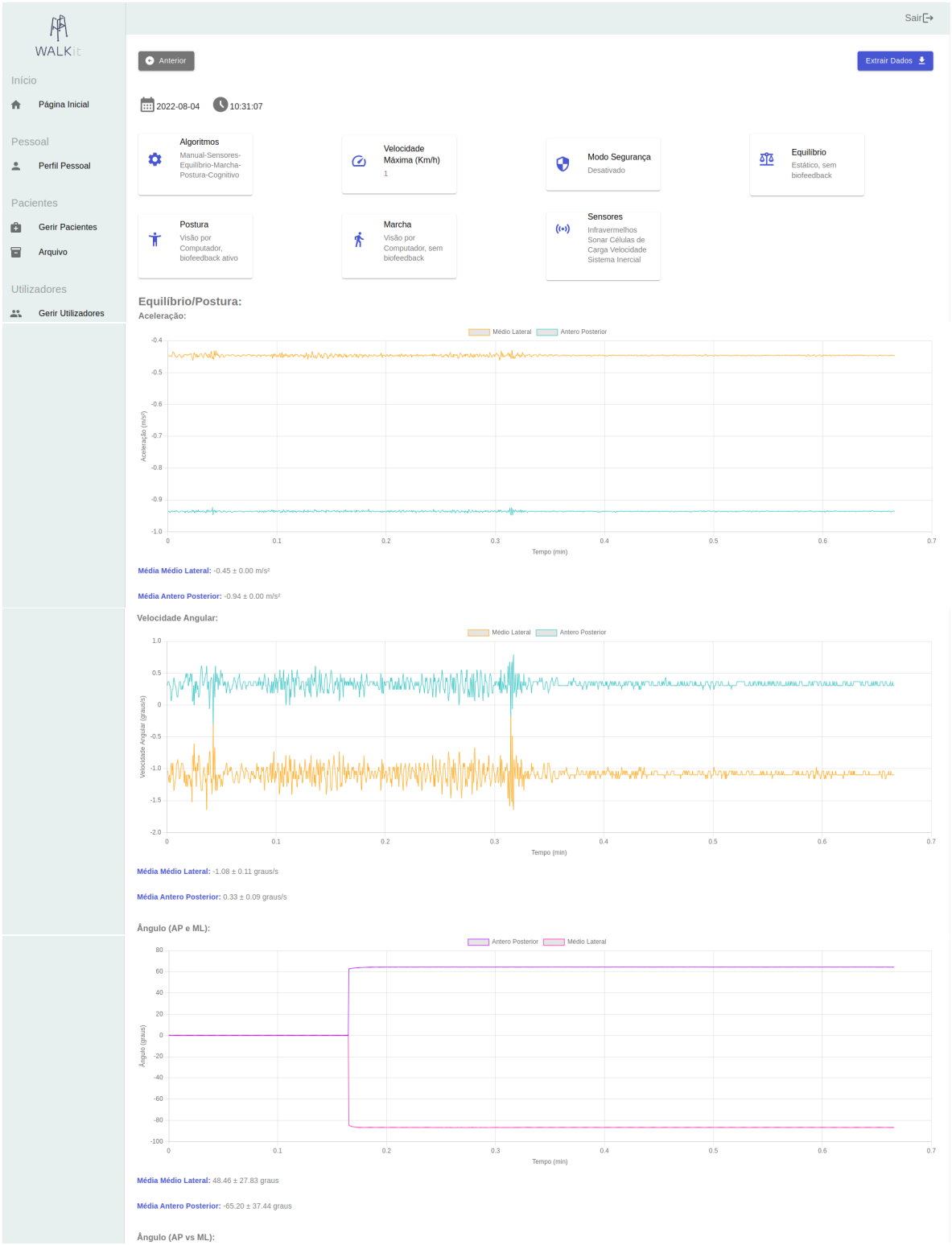

WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Pacientes
Gerir Pacientes
Arquivo
Utilizadores
Gerir Utilizadores
Sair
Anterior
Extrair Dados
2022-08-04
10:31:07
Algoritmos
Manual-Sensores-Equilíbrio-Marcha-Postura-Cognitivo
Velocidade Máxima (Km/h)
1
Modo Segurança
Desativado
Equilíbrio
Estático, sem biofeedback
Postura
Visão por Computador, biofeedback ativo
Marcha
Visão por Computador, sem biofeedback
Sensores
Infravermelhos
Sonar Células de Carga Velocidade
Sistema Inercial
Equilíbrio/Postura:
Aceleração:
Médio Lateral
Antero Posterior
Aceleração (m/s²)
Tempo (min)
Média Médio Lateral: -0.45 ± 0.00 m/s²
Média Antero Posterior: -0.94 ± 0.00 m/s²
Velocidade Angular:
Médio Lateral
Antero Posterior
Velocidade Angular (graus/s)
Tempo (min)
Média Médio Lateral: -1.08 ± 0.11 graus/s
Média Antero Posterior: 0.33 ± 0.09 graus/s
Ângulo (AP e ML):
Antero Posterior
Médio Lateral
Ângulo (graus)
Tempo (min)
Média Médio Lateral: 48.46 ± 27.83 graus
Média Antero Posterior: -65.20 ± 37.44 graus
Ângulo (AP vs ML):



Figure 45: Session details menu.

The temporal evolution menu (Figure 46) is composed of multiple charts representing the patient's evolution throughout all the sessions. The sensors' mean values along with the respective standard deviation were calculated for each session, allowing the clinicians to conclude about the patient's condition throughout the entire therapy and motivating the patients with their evolution over time. It is also important to mention that the charts provide zoom and panning controls, allowing a more detailed analysis.

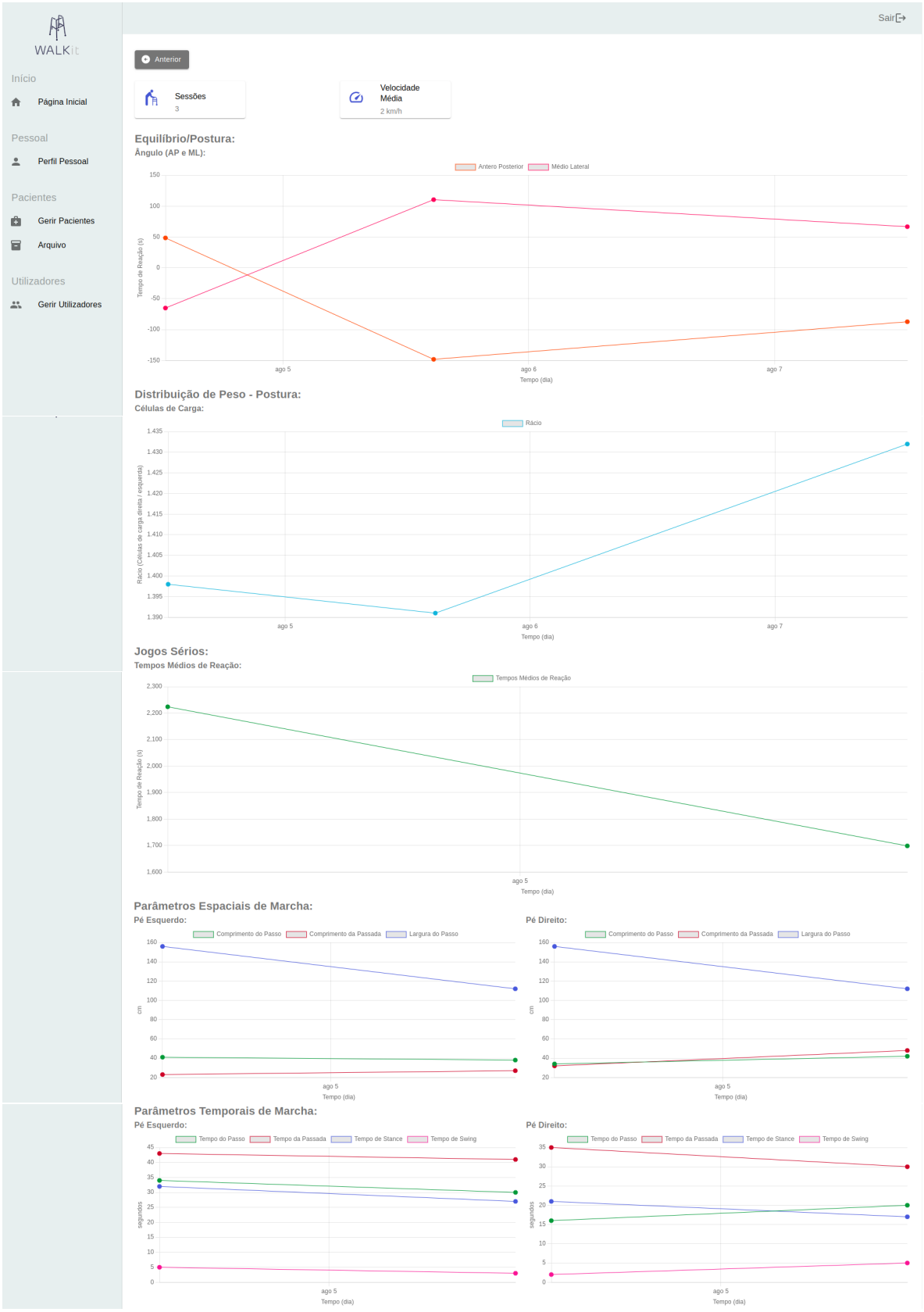


Figure 46: Temporal evolution menu.

The archive, displayed in Figure 47, is composed by a table with patients who had already concluded therapy. It is possible to delete those patients, or to retrieve them to therapy.

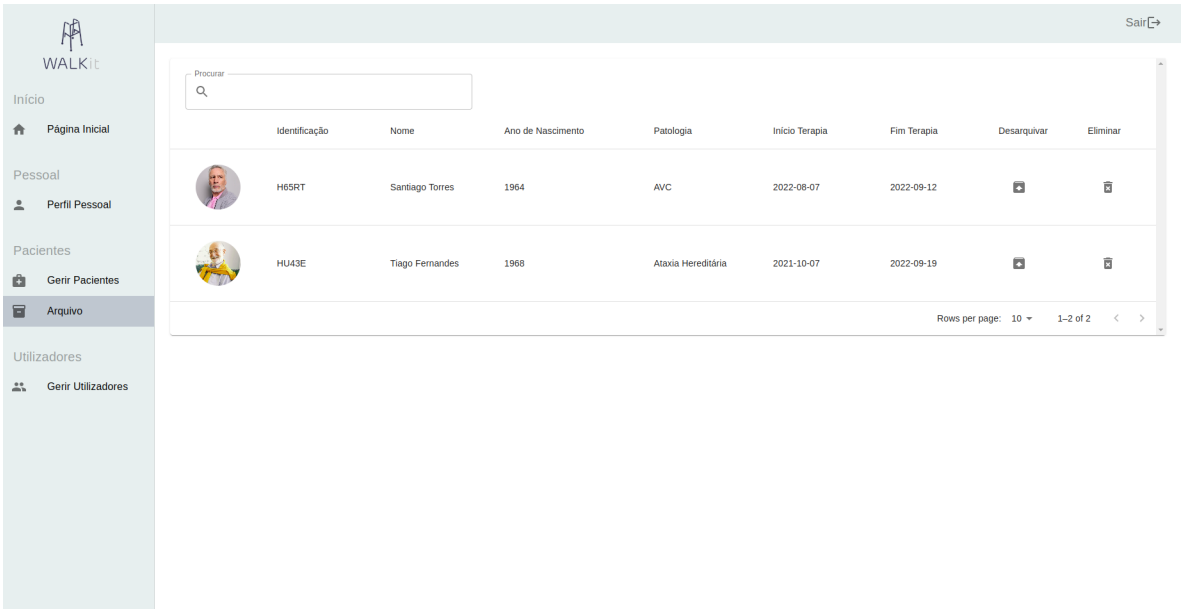


Figure 47: Patients' archive menu.

Lastly, the user management menu, presented in Figure 48, allows the insertion, update or removal of the system users by the administrator.

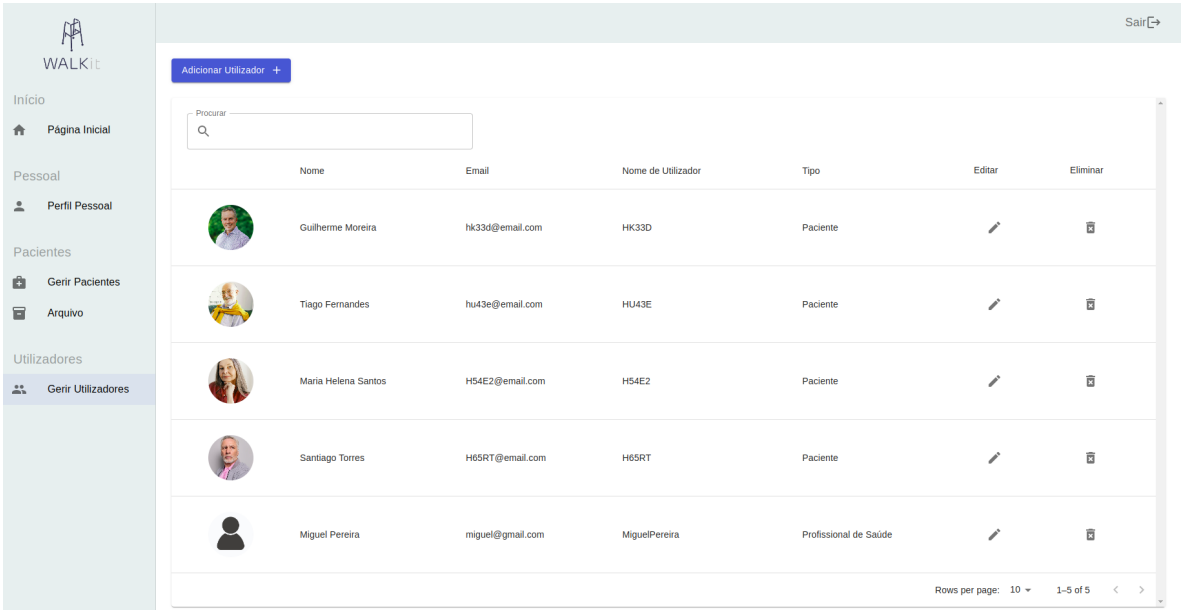


Figure 48: User Management menu.

It is also important to mention that along the application navigation, pop-up alerts are displayed, providing users with information about their actions. Examples are presented in Figure 49, where an alert is asking for permission to delete an user, and other is informing about the success of updating a patient.

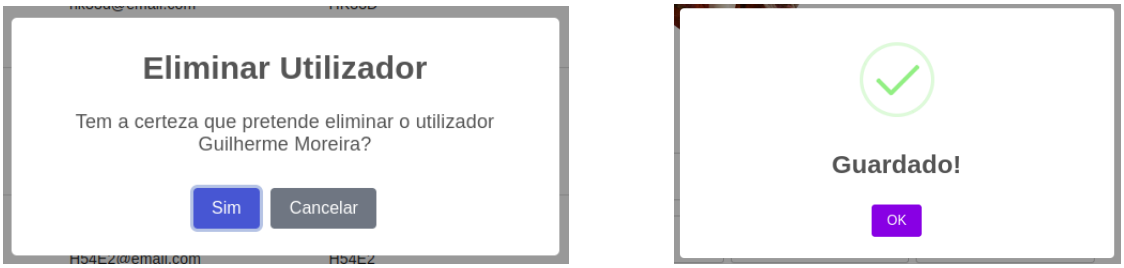


Figure 49: Pop-up Alerts.

The administrator web interfaces were the ones presented during this section since the admin is the most complete of the users. The health professional user interfaces are identical, only changing the user management menu since they do not have permission. The patient interfaces are quite similar, with just a few adjustments, and can be seen in the appendix D.

5.6 DEPLOYMENT

As seen in Figure 35, each system component has its own docker image. Thus, the resulting application can be deployed using *Docker* in any environment.

A docker-compose file was created to allow running all the containers as a single service without the need to start each one separately. The `docker-compose.yml` file is presented below. Each service was defined, *i.e.*, both the backend microservices and the frontend. Each one has its docker image and ports exposure, making the containers reachable internally and externally in the local machine. The `restart:always` was defined, allowing the service to restart automatically if any failure occur. Moreover, it was necessary to define `extra_hosts` for allowing the communication with the local databases.

By running the `docker-compose up -d -build` command, the deployment is completed and it is possible to access the web application via the defined port.

```
version: '3.5'

services:
  backend_mysql:
      build:
        context: ./walkit_webapp_backend
        dockerfile: Dockerfile
      container_name: backend_mysql
      restart: always
      ports:
        - 8082:8082
```

```
      stdin_open: true
      tty: true
      extra_hosts:
        - "host.docker.internal:host-gateway"
      networks:
        main:
          aliases:
            - backend_mysql
  backend_mongodb:
    build:
      context: ./walkit_webapp_backend_mongoDB
      dockerfile: Dockerfile
    extra_hosts:
      - "mongoservice:172.17.0.1"
    container_name: backend_mongodb
    restart: always
    ports:
      - 8081:8081
    stdin_open: true
    tty: true
    networks:
      main:
        aliases:
          - backend_mongodb
  frontend:
    build:
      context: ./walkit_webapp_react
      dockerfile: Dockerfile
    container_name: frontend
    restart: always
    depends_on:
      - backend_mysql
      - backend_mongodb
    ports:
      - 80:81
    networks:
      main:
        aliases:
          - frontend
networks:
  main:
```

Listing 5.1: Docker compose file.

5.7 FINAL APPLICATION VALIDATION AND TESTING

To validate the resulting application, and similarly to the prototyping phase, usability tests were applied to health professionals. In this way, it was possible to check the intuitiveness of the application and if it fulfilled the clinical and technical requirements.

5.7.1 *Participants*

Four healthcare professionals accepted to participate in the usability test. Each participant gave their informed consent to participate in this evaluation. Of the four participants:

- All were female;
- All currently work at the Hospital of Braga;
- One was a physiotherapist, one was a physiatrist, and the other two were residents in the Physical Medicine and Rehabilitation Service;
- All are used to dealing with technologies daily, including at work;

5.7.2 *Methodology*

The methodology used was the same to the one applied in the prototype usability test: the Thinking Aloud Protocol [56][57] (applied to two participants), the Remote Usability Testing [57] (also applied to two participants) and the SUS questionnaire [58][60] (applied to all participants).

Since the application was not online and was only available on one machine, the *Team Viewer* tool was used, allowing the participants to have contact with the system.

Tasks

Ten tasks were outlined to allow a full navigation and evaluation through the web application:

- **Task 1 -** Access the system with the following credentials:
  `username`: teste
  `password`: teste
- **Task 2 -** Insert the patient “Pedro Santos” who has the subsequent personal and clinical data:
  `ID`: HY78U
  `Gender`: Male
  `Year of birth`: 1966
  `Pathology`: Hemorrhagic stroke

`Admission to therapy`: 29/06/2021
`Shoulder distance`: 30cm
`Body mass`: 83kg
`Height`: 176cm
`Foot length`: 28cm
`Pelvis height`: 97cm
`Leg width`: 15cm
`Ankle height`: 9cm
Save the patient.

- **Task 3 -** Finalize the therapy (archive) the patient identified by "HU43E".
- **Task 4 -** Access the patient identified by "HK33D", check his information/details and add the clinical evaluation ABC (evaluate with random numbers and save).
- **Task 5 -** For the same patient, record the number of falls (3 falls) that he had today.
- **Task 6 -** For the same patient, check the patient's sessions and access their temporal evolution.
- **Task 7 -** For the same patient, check the session on 2022/08/04, and download a *csv* file with the sensors acquired data of the session.
- **Task 8 -** Access the archive and delete the patient identified by "HTY61".
- **Task 9 -** Add a new user named "Miguel Pereira", so that he can also have access to the website:
  `Type`: Health Professional
  `Email`: miguel@email.com
  `Username`: miguelPereira
- **Task 10 -** Lastly, go to the personal profile page, change the name, save the modifications and log out of the system.

5.7.3 *Results*

Thinking-Aloud Protocol and Remote Usability Testing

Table 11 shows the healthcare professionals difficulties and suggestions while performing the usability test. In general, the participants did not show great difficulties while executing the tasks, although a limitation was identified regarding the date format being in *month/day/year*. The task that took the longest to complete was the second one, which was expected since it was necessary to enter multiple data related to the patient.

Table 11: Usability Test Results

Tasks	Difficulties	Suggestions	Average Duration
1	· All participants performed this task directly without any difficulties.	· No suggestions.	2.72 $\pm$ 1.82 seconds
2	· Two participants were struggling with the admission date because the date was in month/day/year format instead of day/month/year; · The others performed this task directly without any difficulties.	· All the participants suggested changing the date format; · One participant suggested changing the "body mass" label to "weight".	131.00 $\pm$ 30.51 seconds
3	· All participants performed this task directly without any difficulties.	· No suggestions.	15.03 $\pm$ 3.61 seconds
4	· All participants performed this task directly without any difficulties.	· One participant suggested searching for the patient by clicking on the table row instead of the "details" button.	65.31 $\pm$ 15.90 seconds
5	· All participants performed this task directly without any difficulties.	· No suggestions.	22.20 $\pm$ 5.21 seconds
6	· All participants performed this task directly without any difficulties.	· No suggestions.	18.80 $\pm$ 9.80 seconds
7	· All participants performed this task directly without any difficulties.	· No suggestions.	18.80 $\pm$ 5.78 seconds seconds
8	· Two participants though the archive was in the same menu as the patients who are still under therapy; · The others performed this task directly without any difficulties.	· No suggestions.	30.56 $\pm$ 17.69 seconds
9	· All participants performed this task directly without any difficulties.	· No suggestions.	48.25 $\pm$ 28.09 seconds
10	· All participants performed this task directly without any difficulties.	· No suggestions.	29.78 $\pm$ 11.49 seconds

SUS

After the usability test, each participant fulfilled the SUS questionnaire. Table 12 shows a heat map with the participants' answers and final and average scores. The lowest and higher scores were 87.5 and 100, respectively, with an average of 95.625 $\pm$ 4.8.

Considering the grading scale presented in Figure 23, the degree of usability for the resulting web application is between the "excellent" and "best imaginable" range. Moreover, from the heat map analysis, it is possible to deduce the participants' high satisfaction, which is a positive result.

The final feedback was extremely positive. The participants claim the application was very intuitive, dynamic, and practical, and that fulfilled all the clinical requirements.

Table 12: SUS Scores Final Application

		Partipants Answers			
Questions	Questions	A	B	C	D
1	I think that I would like to use this system.	5	5	5	5
2	I found the system unnecessarily complex.	1	2	1	1
3	I thought the system was easy to use.	5	4	5	5
4	I think that I would need the support of a technical person to be able to use this system.	1	1	1	1
5	I found the various functions in the system were well integrated.	5	4	5	5
6	I thought there was too much inconsistency in this system.	1	1	1	1
7	I would imagine that most people would learn to use this system very quickly.	5	4	4	4
8	I found the system very cumbersome to use.	1	1	1	1
9	I felt very confident using the system.	5	4	5	5
10	I needed to learn a lot of things before I could get going with this system.	1	1	1	1
Score by participant		100	87.5	97.5	97.5

Average Score:	95.625 ± 4.8

Based on the usability test results and to improve the application's acceptance, three changes were made according to the suggestions of the medical staff:

- The date format was changed to *day/month/year*;
- The label "body mass" was changed to "weight". Although the term "body mass" is more accurate, the clinician argued that it could be misleading with body mass index, which is often used in the clinical context;
- The selection of a patient can also be made by clicking in the table row.

5.8 GENERAL CONCLUSIONS

This chapter presented the development process carried out to develop the WALKit SmartW new web application, that allow the management of rehabilitation sessions with the robotic walker. The prototyping model was followed to develop the application, and each stage, including the design, prototyping, user evaluation, implementation, testing, and deployment phases, were achieved successfully. The feedback from the health professionals was extremely positive, and they concluded that the application is valuable and could fully motivate the patients throughout treatments.

6

CONCLUSIONS

This chapter presents the conclusions regarding the work that has been developed to achieve the objectives outlined for the dissertation and suggests possible improvements as a part of future work.

As introduced previously, this dissertation's main goals included finding an efficient and effective way to store the data acquired during a rehabilitation session, along with the clinical information of patients, and the development of an application capable of providing an intuitive way to manage the rehabilitation sessions and to access the clinical information and evolution of patients undergoing rehabilitation with the robotic walker. Furthermore, the optimization of the walker's embedded application was also considered in this dissertation.

To achieve the first goal, *i.e.*, finding an effective way to store data, it was decided to create a polyglot persistence database system. A relational and a non-relational databases were combined in a way to enhance performance, maximizing the efficiency. The *MongoDB* database is responsible for storing the data acquired during the rehabilitation sessions in a time-series collection, while the *MySQL* database stores the structured data, namely patients' clinical information, session configuration, and specific metrics of the algorithms.

Regarding the optimization of the walker's application, and based on the requirements gathered with the health professionals, new features were implemented, namely feedback alerts, session's configuration forms, and a menu with the sessions' information. Furthermore, this application was exempt from any security mechanisms, so a crucial step was the security enforcement to keep the patients' data safe. Health professionals were recruited to validate and test the application usability, and the results of the usability questionnaire were in the range of good to excellent, with an average SUS score of 84.17 ± 12.39.

Regarding the software to manage the rehabilitation sessions, it was decided to develop a responsive web application, allowing to be accessed by multiple devices. The development process had continuous support from the medical team. The result was an intuitive and user-friendly software, with multiple charts representing the patients' data acquired with the robotic walker, allowing health professionals to assess their evolution throughout therapy. In a similar way to the walker's embedded software, health professionals were recruited to test and validate the web application, and the results were in the range of excellent to best imaginable, with an average SUS score of 95.63 ± 4.8.

Ultimately, after achieving the main goals of this dissertation and considering the developed work, it is now possible to answer the Research Questions (RQ) presented in Chapter 1.

RQ 1: *What are the best ways to store data relative to patients and/ or health professionals and data from the rehabilitation sessions?*

This RQ was answered in Chapters 2 and 3. After a literature review about data storage, where the characteristics of relational and non-relational databases were explored, it was possible to draw some conclusions about the best ways of storing health data. According to the literature review, health applications generally resort to SQL databases to store data related to patients, clinicians, appointments, among other structured data. They provide multiple features including, the ACID properties, security mechanisms for data protection, prevention of data redundancy, and performance of complex queries. On the other hand, non-relational databases are used in health systems to store large amounts of structured or loosely structured data, for example, data acquired from sensors, images, or blood relations. This type of database provides high availability and scalability. A combination of both technologies can be configured together to achieve better performance. Considering this, a polyglot persistence system was proposed and implemented in this dissertation. The system combines a relational (*MySQL*) and non-relational database (*MongoDB*) to enhance the performance. *MySQL* is used to store data related to patients and/or health professionals, and *MongoDB* is responsible for storing the sensors' data acquired during the rehabilitation sessions with the robotic walker.

RQ 2: *How can the walker data be stored efficiently while maintaining security and user's privacy?*

This question was answered in Chapters 2, 4 and 5. In the literature review, it was pointed out multiple security measures that health applications may have to protect the patient's sensitive data. Within those measures, some were implemented to protect the data while stored in the database and in transmission. A login page was implemented in both applications, ensuring the authentication and authorization of the users. Moreover, the patient's personal data is encrypted using the AES CBC mode, being in an unreadable state while stored and during transmission. To avoid SQL injections, all the input files, in both applications, are constantly validated. Lastly, the patient's data details were separated from the data acquired from the sensors in different servers. This is also considered a security measure since if a single service is compromised, there is less change in connecting the clinical data acquired during sessions with the respective patient, maintaining the anonymization between the two servers.

RQ 3: *How can data collected by the walker be accessed on several electronic devices?*

This question was answered in Chapter 3. Since one of the requirements was to access and manage the data acquired from the walker sensors on several devices, it was decided to develop a responsive web application. In this way, health professionals and patients can monitor the rehabilitation sessions from a computer, smartphone or tablet, and from any location with an internet connection. Currently, the databases are local, but in the future, the intention is to migrate them to the cloud, allowing this approach to be possible.

RQ 4: *How is the software created useful in managing the data acquired in a session with the robotic walker?*

This question was addressed in Chapter 5. The software was validated and tested by the medical team who works with the robotic walker. The feedback received was extremely positive. In general, they thought the software was very useful and practical, and that it fulfilled the clinical needs and requirements. Additionally, they claimed that the charts with the patients evolution will help in the analysis of the patient status in an intuitive form and, by showing them their evolution, it may help them become more motivated during treatments.

6.1 FUTURE WORK

Several ideas for future work and application improvements were raised during this work. Firstly, for the users to have a better experience while using the web application, lazy loading could be implemented on the charts. Since the charts handle big volumes of data, lazy loading would provide a faster component rendering because sessions' data could be splitted in different time blocks and only rendered when needed. Moreover, during the web application development multiple charts libraries were tested, and the ones with better performance results needed to be paid. A future opportunity could be changing the chart library to a better one. To improve the system's security, the Https protocol could be implemented, along with auditing and monitoring mechanisms. In this way, the security during data transmission would be assured, and the user actions would be recorded, allowing to track and identify security events more easily. The use of cloud environments to store data will also be part of the future work. This could make it possible to access data from different locations, apart from several advantages such as storage capability, data backups and security. Lastly, having feedback from the patients would also be important in the web application assessment.

BIBLIOGRAPHY

[1] Jon Marsden and Chris Harris. Cerebellar ataxia: pathophysiology and rehabilitation. *Clinical Rehabilitation*, 25(3):195–216, 2011. doi: 10.1177/0269215510382495. URL https://doi.org/10.1177/0269215510382495. PMID: 21321055.

[2] A. Matsushima, Y. Maruyama, N. Mizukami, M. Tetsuya, M. Hashimoto, and K. Yoshida. Gait training with a wearable curara® robot for cerebellar ataxia: a single-arm study. *BioMedical Engineering Online*, 20(1), 2021. URL www.scopus.com.

[3] Marcia Belas dos Santos, Clarissa Barros de Oliveira, Arly Dos Santos, Cristhiane Garabello Pires, Viviana Dylewski, and Ricardo Mario Arida. A comparative study of conventional physiotherapy versus robot-assisted gait training associated to physiotherapy in individuals with ataxia after stroke. *Behavioural neurology*, 2018, 2018.

[4] João M. Lopes, João André, António Pereira, Manuel Palermo, Nuno Ribeiro, João Cerqueira, and Cristina P. Santos. Asbgo: A smart walker for ataxic gait and posture assessment. In *Robotic Technologies in Biomedical and Healthcare Engineering*. CRC Press, 2021.

[5] Octavian Postolache. Physical rehabilitation assessment based on smart training equipment and mobile apps. In *2015 E-Health and Bioengineering Conference (EHB)*, pages 1–6, 2015. doi: 10.1109/EHB.2015.7391530.

[6] Stanley Winser, Catherine Smith, Leigh Hale, Leica Claydon, and Susan Whitney. Balance outcome measures in cerebellar ataxia: A delphi survey. *Disability and rehabilitation*, 37, 04 2014. doi: 10.3109/09638288.2014.913709.

[7] Inês Caetano, Joana Alves, Ricardo Varzim, and Cristina P. Santos. Development of a data base to gait analysis and data organization in asbgo smart walker. In *2017 IEEE 5th Portuguese Meeting on Bioengineering (ENBENG)*, pages 1–4, 2017. doi: 10.1109/ENBENG.2017.7889485.

[8] Uzma Mustafa, Eckhard Pflugel, and Nada Philip. A novel privacy framework for secure m-health applications: The case of the gdpr. In *2019 IEEE 12th International Conference on Global Security, Safety and Sustainability (ICGS3)*, pages 1–9, 2019. doi: 10.1109/ICGS3.2019.8688019.

[9] Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom. *Database Systems: The Complete Book*. Prentice Hall Press, USA, 2 edition, 2008. ISBN 9780131873254.

[10] Haleemunnisa Fatima and Kumud Wasnik. Comparison of sql, nosql and newsql databases for internet of things. In *2016 IEEE Bombay Section Symposium (IBSS)*, pages 1–6, 2016. doi: 10.1109/IBSS.2016.7940198.

[11] Kosovare Sahatqija, Jaumin Ajdari, Xhemal Zenuni, Bujar Raufi, and Florije Ismaili. Comparison between relational and nosql databases. In *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 0216–0221, 2018. doi: 10.23919/MIPRO.2018.8400041.

[12] Tariq N. Khasawneh, Mahmoud H. AL-Sahlee, and Ali A. Safia. Sql, newsql, and nosql databases: A comparative survey. In *2020 11th International Conference on Information and Communication Systems (ICICS)*, pages 013–021, 2020. doi: 10.1109/ICICS49469.2020.239513.

[13] Yishan Li and Sathiamoorthy Manoharan. A performance comparison of sql and nosql databases. In *2013 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM)*, pages 15–19, 2013. doi: 10.1109/PACRIM.2013.6625441.

[14] Sowndarya Palanisamy and P. SuvithaVani. A survey on rdbms and nosql databases mysql vs mongodb. In *2020 International Conference on Computer Communication and Informatics (ICCCI)*, pages 1–7, 2020. doi: 10.1109/ICCCI48352.2020.9104047.

[15] Feng Guo. The design and optimization of database. In *Journal of Physics: Conference Series*, volume 1087. IOP Publishing, 2018.

[16] Karamjit Kaur and Rinkle Rani. Managing data in healthcare information systems: Many models, one solution. *Computer*, 48(3):52–59, 2015. doi: 10.1109/MC.2015.77.

[17] Anuchit Nirapai, Jittarporn Somchue, Pongsiri Chaikot, Nipa Kimsungnoen, Manas Sangworasil, Ratchanee Saosuwan, Anantasak Wongkamhang, and Nipa Kimsungnoen. Health information system for home visits. In *2019 4th Technology Innovation Management and Engineering Science International Conference (TIMES-iCON)*, pages 1–4, 2019. doi: 10.1109/TIMES-iCON47539.2019.9024480.

[18] Fayezah Anjum, Abu Saleh Mohammed Shoaib, Abdullah Ibne Hossain, and Mohammad Monirujjaman Khan. Online health care. In *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 580–583, 2018. doi: 10.1109/CCWC.2018.8301617.

[19] Jagdev Bhogal and Imran Choksi. Handling big data using nosql. In *2015 IEEE 29th International Conference on Advanced Information Networking and Applications Workshops*, pages 393–398, 2015. doi: 10.1109/WAINA.2015.19.

[20] Rashid Zafar, Eiad Yafi, Megat F. Zuhairi, and Hassan Dao. Big data: The nosql and rdbms review. In *2016 International Conference on Information and Communication Technology (ICICTM)*, pages 120–126, 2016. doi: 10.1109/ICICTM.2016.7890788.

[21] Lorenzo Carnevale, Rocco Salvatore Calabrò, Antonio Celesti, Antonino Leo, Maria Fazio, Placido Bramanti, and Massimo Villari. Toward improving robotic-assisted gait training: Can big data analysis help us? *IEEE Internet of Things Journal*, 6(2):1419–1426, 2019. doi: 10.1109/JIOT.2018.2855937.

[22] João Rala Cordeiro and Octavian Postolache. Big data storage for a health predictive system. In *2018 International Symposium in Sensing and Instrumentation in IoT Era (ISSI)*, pages 1–6, 2018. doi: 10.1109/ISSI.2018.8538285.

[23] Luís A. Bastião Silva, Louis Beroud, Carlos Costa, and José Luis Oliveira. Medical imaging archiving: A comparison between several nosql solutions. In *IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*, pages 65–68, 2014. doi: 10.1109/BHI.2014.6864305.

[24] G. Dumindu Samaraweera and J. Morris Chang. Security and privacy implications on database systems in big data era: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 33(1):239–258, 2021. doi: 10.1109/TKDE.2019.2929794.

[25] Dimitra Georgiou and Costas Lambrinoudakis. Compatibility of a security policy for a cloud-based healthcare system with the eu general data protection regulation (gdpr). *Information*, 11(12), 2020. ISSN 2078-2489. doi: 10.3390/info11120586. URL https://www.mdpi.com/2078-2489/11/12/586.

[26] Isabel Maria Lopes and Pedro Oliveira. Implementation of the general data protection regulation: A survey in health clinics. In *2018 13th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–6, 2018. doi: 10.23919/CISTI.2018.8399156.

[27] Joana Muchagata and Ana Ferreira. Translating gdpr into the mhealth practice. In *2018 International Carnahan Conference on Security Technology (ICCST)*, pages 1–5, 2018. doi: 10.1109/CCST.2018.8585546.

[28] Chiara Braghin, Stelvio Cimato, and Alessio Della Libera. Are mhealth apps secure? a case study. In *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, volume 02, pages 335–340, 2018. doi: 10.1109/COMPSAC.2018.10253.

[29] Sreenivasa Rao Basavala, Narendra Kumar, and Alok Agarrwal. Authentication: An overview, its types and integration with web and mobile applications. In *2012 2nd IEEE International Conference on Parallel, Distributed and Grid Computing*, pages 398–401, 2012. doi: 10.1109/PDGC.2012.6449853.

[30] Madhumita Panda. Performance analysis of encryption algorithms for security. In *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, pages 278–284, 2016. doi: 10.1109/SCOPES.2016.7955835.

[31] Qi Zhang and Qun Ding. Digital image encryption based on advanced encryption standard (aes). In *2015 Fifth International Conference on Instrumentation and Measurement, Computer, Communication and Control (IMCCC)*, pages 1218–1221, 2015. doi: 10.1109/IMCCC.2015.261.

[32] Avinash Ray, Anjali Potnis, Prashant Dwivedy, Shahbaz Soofi, and Uday Bhade. Comparative study of aes, rsa, genetic, affine transform with xor operation, and watermarking for image encryption. In *2017 International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE)*, pages 274–278, 2017. doi: 10.1109/RISE.2017.8378166.

[33] Achmad Solichin, Mochammad Andika Putra, and Khaeri Diniari. Restful web service optimization with compression and encryption algorithm. In *2018 International Seminar on Application for Technology of Information and Communication*, pages 333–337, 2018. doi: 10.1109/ISEMANTIC.2018.8549727.

[34] Muhammad J. Alibasa, Marlon R. Santos, Nick Glozier, Samuel B. Harvey, and Rafael A. Calvo. Designing a secure architecture for m-health applications. In *2017 IEEE Life Sciences Conference (LSC)*, pages 91–94, 2017. doi: 10.1109/LSC.2017.8268151.

[35] Divyani Yadav, Deeksha Gupta, Dhananjay Singh, Devendra Kumar, and Upasana Sharma. Vulnerabilities and security of web applications. In *2018 4th International Conference on Computing Communication and Automation (ICCCA)*, pages 1–5, 2018. doi: 10.1109/CCAA.2018.8777558.

[36] Sundararaman Rajagopalan, Siva Janakiraman, Amirtharajan Rengarajan, Sivaraman Rethinam, Sridevi Arumugham, and Gowtham Saravanan. Iot framework for secure medical image transmission. In *2018 International Conference on Computer Communication and Informatics (ICCCI)*, pages 1–5, 2018. doi: 10.1109/ICCCI.2018.8441284.

[37] Antoanela Naaji and Marius Popescu. Algorithms for securing medical web pages and files. In *2020 24th International Conference on Circuits, Systems, Communications and Computers (CSCC)*, pages 11–17, 2020. doi: 10.1109/CSCC49995.2020.00011.

[38] T. Saravanan, Sumit Jha, Gautam Sabharwal, and Shubham Narayan. Comparative analysis of software life cycle models. In *2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, pages 906–909, 2020. doi: 10.1109/ICACCCN51052.2020.9362931.

[39] Shilpa Sen, Mayank Patel, and Ajay Kumar Sharma. Software development life cycle performance analysis. In Rajeev Mathur, C. P. Gupta, Vaibhav Katewa, Dharm Singh Jat, and Neha Yadav, editors, *Emerging Trends in Data Driven Computing and Communications*, pages 311–319, Singapore, 2021. Springer Singapore. ISBN 978-981-16-3915-9.

[40] Vittorio Cortellessa, Antinisca Marco, and Paola Inverardi. *Software Lifecycle and Performance Analysis*, pages 65–77. 05 2011. ISBN 978-3-642-13620-7. doi: 10.1007/978-3-642-13621-4_4.

[41] Omar Sabri and Fahad Alfifi. Integrating knowledge life cycle within software development process to produce a quality software product. In *2017 International Conference on Engineering and Technology (ICET)*, pages 1–7, 2017. doi: 10.1109/ICEngTechnol.2017.8308172.

[42] Mohamed A. Ben-Zahia and Ibrahim Jaluta. Criteria for selecting software development models. In *2014 Global Summit on Computer Information Technology (GSCIT)*, pages 1–6, 2014. doi: 10.1109/GSCIT.2014.6970099.

[43] X. Ferre, N. Juristo, H. Windl, and L. Constantine. Usability basics for software developers. *IEEE Software*, 18(1):22–29, 2001. doi: 10.1109/52.903160.

[44] Siriwan Kajornkasirat, Napat Chanapai, and Benjawan Hnusuwan. Smart health monitoring system with iot. In *2018 IEEE Symposium on Computer Applications Industrial Electronics (ISCAIE)*, pages 206–211, 2018. doi: 10.1109/ISCAIE.2018.8405471.

[45] S.Yu. Zavozkin. Patient health monitoring information system. In *2020 International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon)*, pages 1–5, 2020. doi: 10.1109/FarEastCon50210.2020.9271545.

[46] C Visvesvaran, B Maruthi Shankar, S Kaviya, P Kaviya, K Monika, and I J John Bharath Kumar. Quarcare - iot based patient health monitoring system. In *2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC)*, pages 870–874, 2021. doi: 10.1109/ICESC51422.2021.9532652.

[47] Danilo Pani, Gianluca Barabino, Alessia Dessì, Iosto Tradori, Matteo Piga, Alessandro Mathieu, and Luigi Raffo. A device for local or remote monitoring of hand rehabilitation sessions for rheumatic patients. *IEEE Journal of Translational Engineering in Health and Medicine*, 2:1–11, 2014. doi: 10.1109/JTEHM.2014.2299274.

[48] David. M. Brennan, Peter S. Lum, Gitendra Uswatte, Edward Taub, Brendan M. Gilmore, and Joydip Barman. A telerehabilitation platform for home-based automated therapy of arm function. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 1819–1822, 2011. doi: 10.1109/IEMBS.2011.6090518.

[49] Carlos Nave, Yongsheng Yang, Vítor Viegas, and Octavian Postolache. Physical rehabilitation based on smart walker. In *2018 12th International Conference on Sensing Technology (ICST)*, pages 388–393, 2018. doi: 10.1109/ICSensT.2018.8603660.

[50] O. Postolache, P. Girão, J.M. Dias Pereira, J. Pincho, C. Moura, and G. Postolache. Smart walker for pervasive healthcare. In *2011 Fifth International Conference on Sensing Technology*, pages 482–487, 2011. doi: 10.1109/ICSensT.2011.6137027.

[51] Apar Garg, Rohit Kumar Kaliyar, and Anurag Goswami. Pdrsd-a systematic review on plan-driven sdlc models for software development. In *2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS)*, volume 1, pages 739–744, 2022. doi: 10.1109/ICACCS54159.2022.9785261.

[52] Ricardo J. Machado João M. Fernandes. *Requirements in Engineering Projects*. Springer Cham, 1 edition, 2016.

[53] Srikrishna Prasad and S B Avinash. Application of polyglot persistence to enhance performance of the energy data management systems. In *2014 International Conference on Advances in Electronics Computers and Communications*, pages 1–6, 2014. doi: 10.1109/ICAECC.2014.7002444.

[54] Kodrat Iman Satoto, R. Rizal Isnanto, Rinta Kridalukmana, and Kurniawan Teguh Martono. Optimizing mysql database system on information systems research, publications and community service. In *2016 3rd International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, pages 1–5, 2016. doi: 10.1109/ICITACEE.2016.7892476.

[55] Jonathan Katz and Yehuda Lindell. *Introduction to modern cryptography*. CRC press, 2020.

[56] Freddy Paz, Freddy A. Paz, Daniela Villanueva, and José Antonio Pow-Sang. Heuristic evaluation as a complement to usability testing: A case study in web domain. In *2015 12th International Conference on Information Technology - New Generations*, pages 546–551, 2015. doi: 10.1109/ITNG.2015.92.

[57] Muhammad Junaid Aamir and Arshad Mansoor. Testing web application from usability perspective. In *2013 3rd IEEE International Conference on Computer, Control and Communication (IC4)*, pages 1–7, 2013. doi: 10.1109/IC4.2013.6653765.

[58] Jeff Sauro and James R Lewis. *Quantifying the user experience: Practical statistics for user research*. Morgan Kaufmann, 2016.

[59] John Brooke. Sus: A quick and dirty usability scale. *Usability Eval. Ind.*, 189, 11 1995.

[60] Ana Isabel Martins, Ana Filipa Rosa, Alexandra Queirós, Anabela Silva, and Nelson Pacheco Rocha. European portuguese validation of the system usability scale (sus). *Procedia Computer Science*, 67: 293–300, 2015. ISSN 1877-0509. doi: https://doi.org/10.1016/j.procs.2015.09.273. URL https://www.sciencedirect.com/science/article/pii/S1877050915031191. Proceedings of the 6th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion.

[61] Aaron Bangor, Philip Kortum, and James Miller. Determining what individual sus scores mean: Adding an adjective rating scale. *Journal of usability studies*, 4(3):114–123, 2009.

A

REQUIREMENTS SURVEY QUESTIONNAIRE

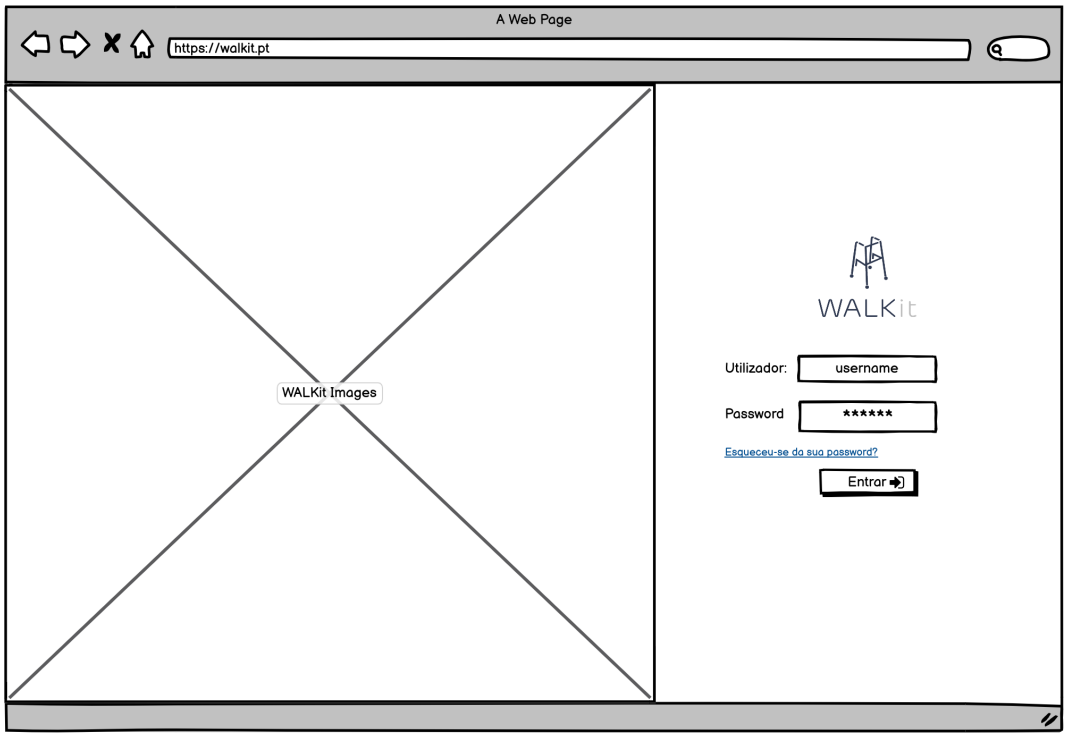
Ao longo deste questionário classifique, quando necessário, as questões de 1 a 4, sendo 1-"Não será útil", 2-"Será pouco útil", 3-"Será útil", 4-"Será muito útil".

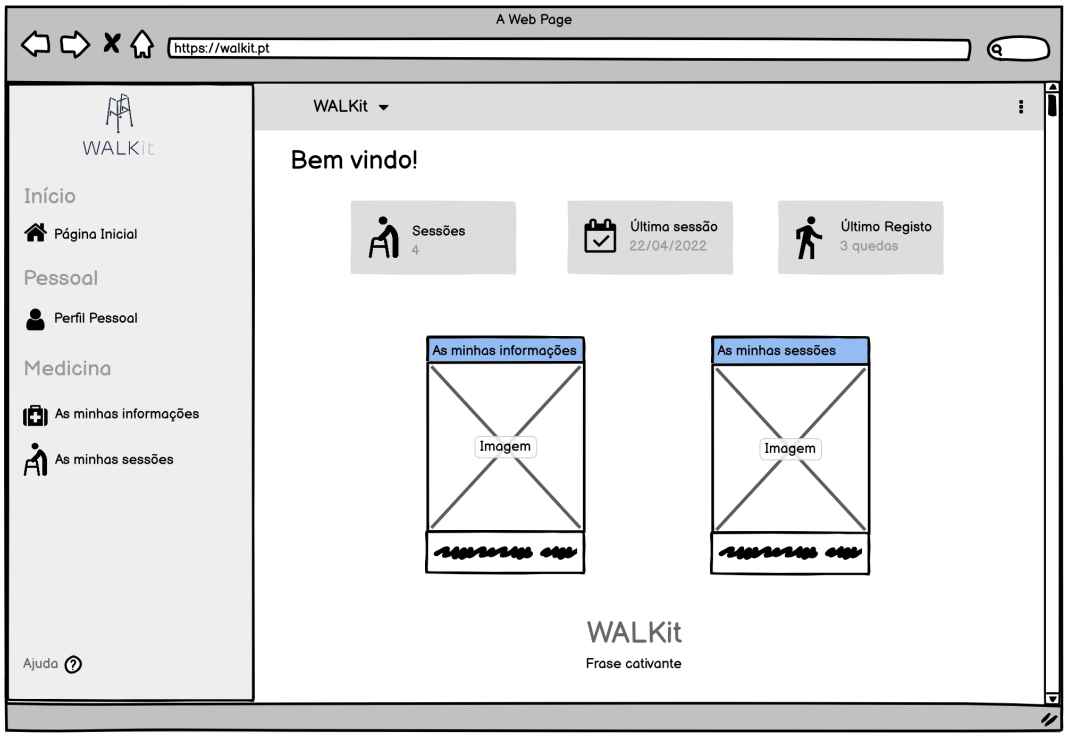
1. Qual a sua opinião relativamente à aplicação proposta?
2. Considera a ideia de colocar gráficos com a evolução do paciente útil?
3. Considera a modulação da aplicação em diferentes dispositivos eletrónicos útil? Isto é, ser adaptável para computadores, smartphones, tablets, entre outros.
4. Qual a sua opinião relativamente à aplicação poder ser utilizada fora do contexto hospitalar? Por exemplo, verificar o progresso dos pacientes em casa.
5. Considera útil a existência de um perfil dedicado ao paciente para que este consiga analisar o seu progresso?
6. Crê que seja relevante que a aplicação permita marcar e desmarcar sessões de reabilitação no andarilho, por exemplo, incluindo um calendário de utilização?
7. Considera útil a inclusão de questionários/testes clínicos de avaliação (e.g., SARA, ABC, TUG) na aplicação?
8. Na sua opinião, quais as informações clínicas que considera importante incluir na aplicação para melhor descrever os pacientes que usam o andarilho?
9. Na sua opinião, o que poderia ser incluído na aplicação que seria útil/vantajoso para a sua atividade profissional?

B

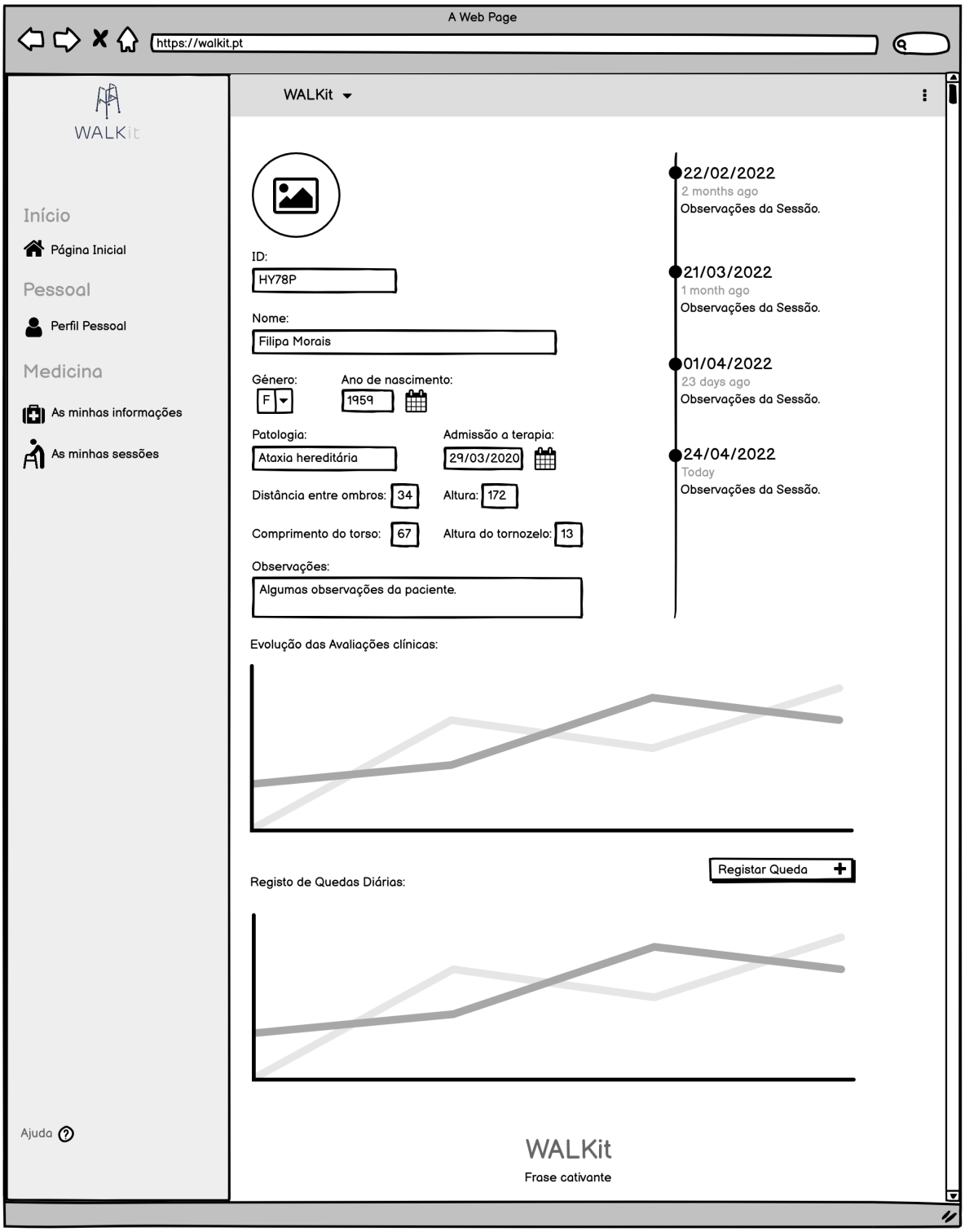
WEB APPLICATION WIREFRAMES

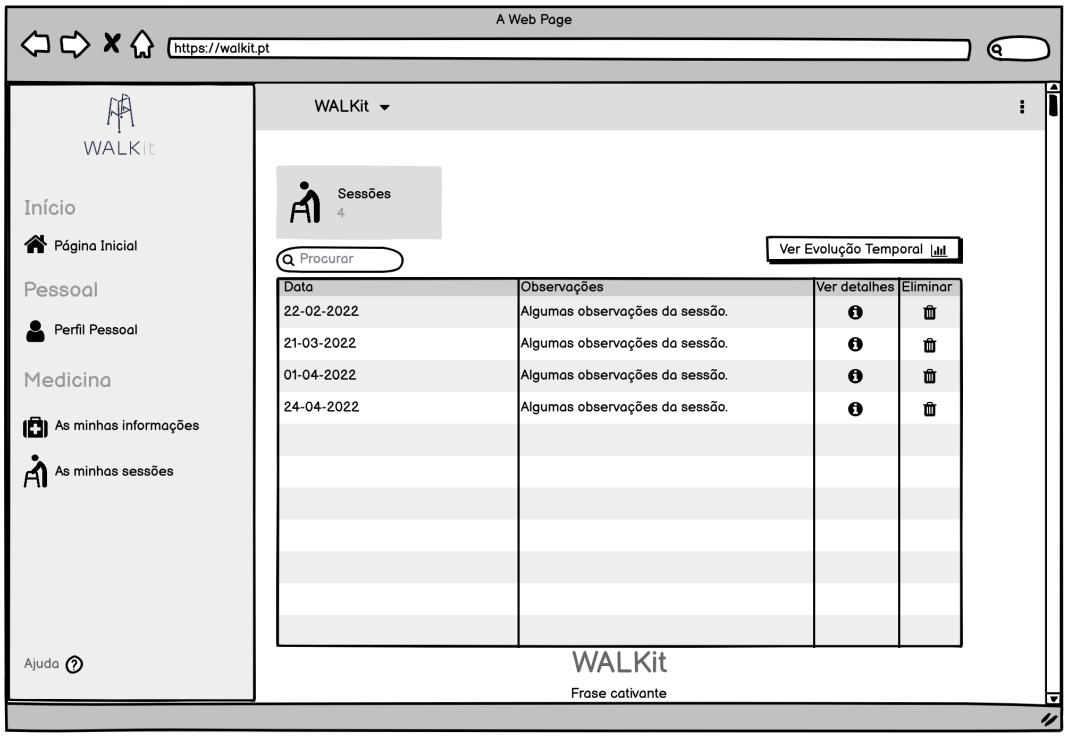
B.1 PATIENT WIREFRAMES

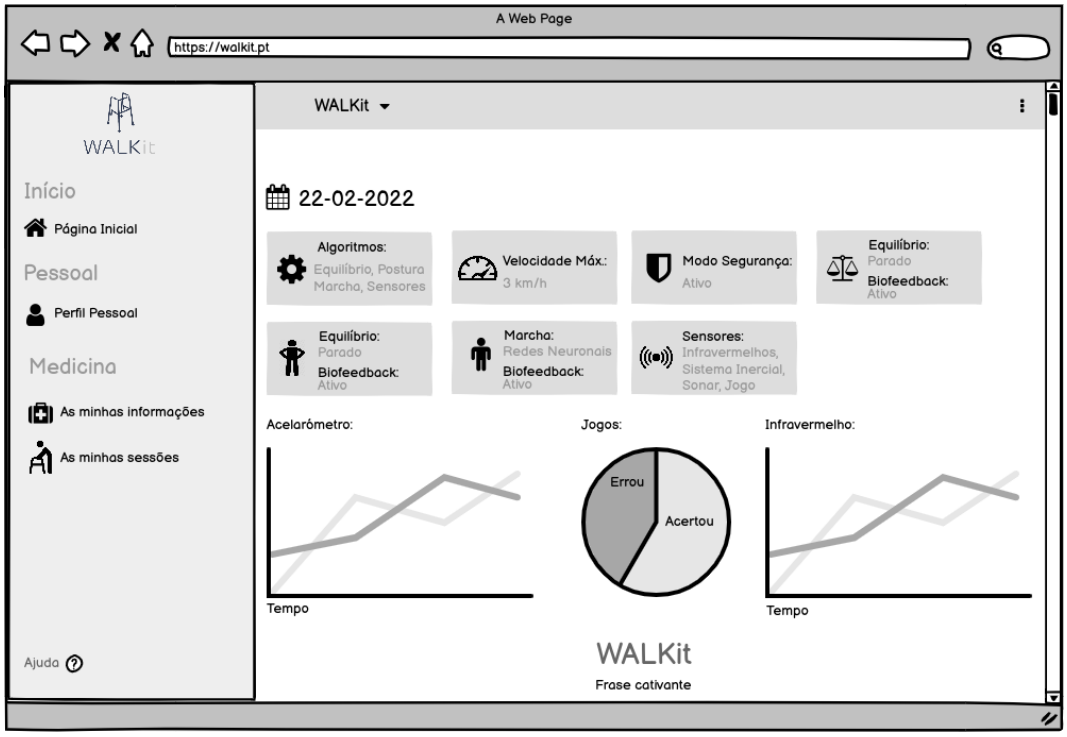


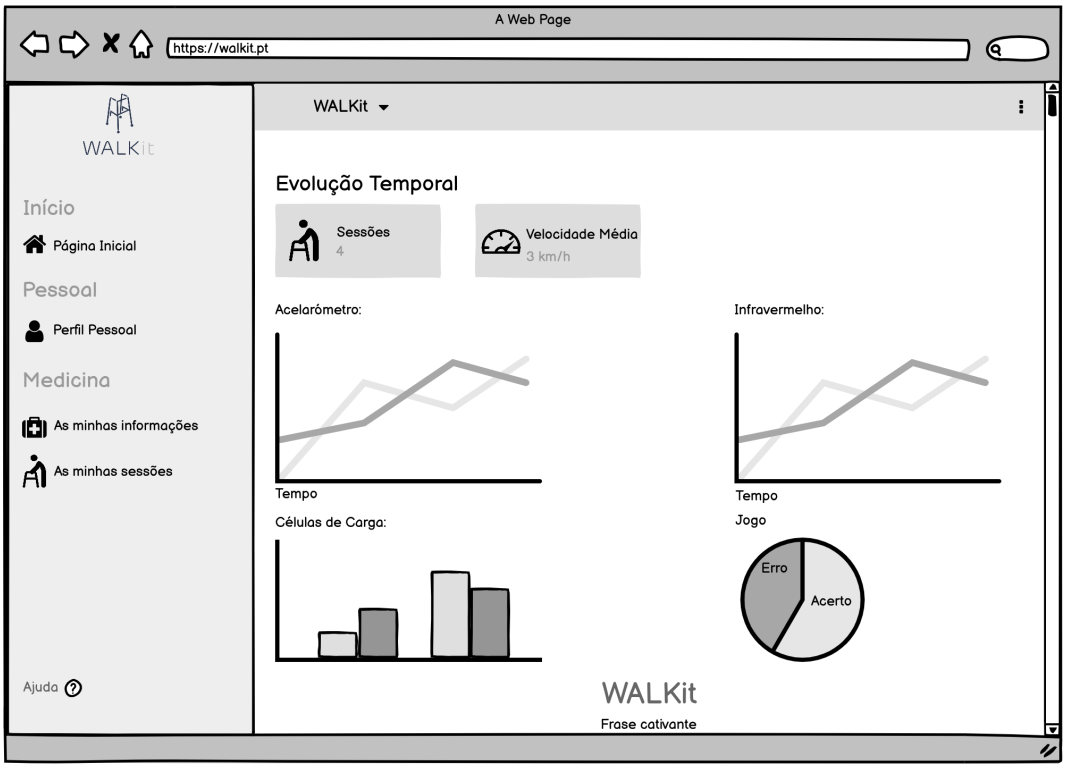

A Web Page
https://walkit.pt
WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
As minhas informações
As minhas sessões
Ajuda
WALKit
Bem vindo!
Sessões
4
Última sessão
22/04/2022
Último Registo
3 quedas
As minhas informações
Imagem
As minhas sessões
Imagem
WALKit
Frase cativante


A Web Page
https://walkit.pt
WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
As minhas informações
As minhas sessões
Ajuda
WALKit
Nome de Utilizador:
username
Email:
user@email.com
Nome:
User
Password
******
Guardar Alterações
WALKit
Frase cativante


A Web Page
https://walkit.pt
WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
As minhas informações
As minhas sessões
Ajuda
WALKit
ID:
HY78P
Nome:
Filipa Morais
Género:
F
Ano de nascimento:
1959
Patologia:
Ataxia hereditária
Admissão a terapia:
29/03/2020
Distância entre ombros:
34
Altura:
172
Comprimento do torso:
67
Altura do tornozelo:
13
Observações:
Algumas observações da paciente.
22/02/2022
2 months ago
Observações da Sessão.
21/03/2022
1 month ago
Observações da Sessão.
01/04/2022
23 days ago
Observações da Sessão.
24/04/2022
Today
Observações da Sessão.
Evolução das Avaliações clínicas:
Registar Queda
Registo de Quedas Diárias:
WALKit
Frase cativante


A Web Page
https://walkit.pt
WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
As minhas informações
As minhas sessões
Ajuda
WALKit
Sessões
4
Procurar
Ver Evolução Temporal
Data
Observações
Ver detalhes
Eliminar
22-02-2022
Algumas observações da sessão.
21-03-2022
Algumas observações da sessão.
01-04-2022
Algumas observações da sessão.
24-04-2022
Algumas observações da sessão.
WALKit
Frase cativante

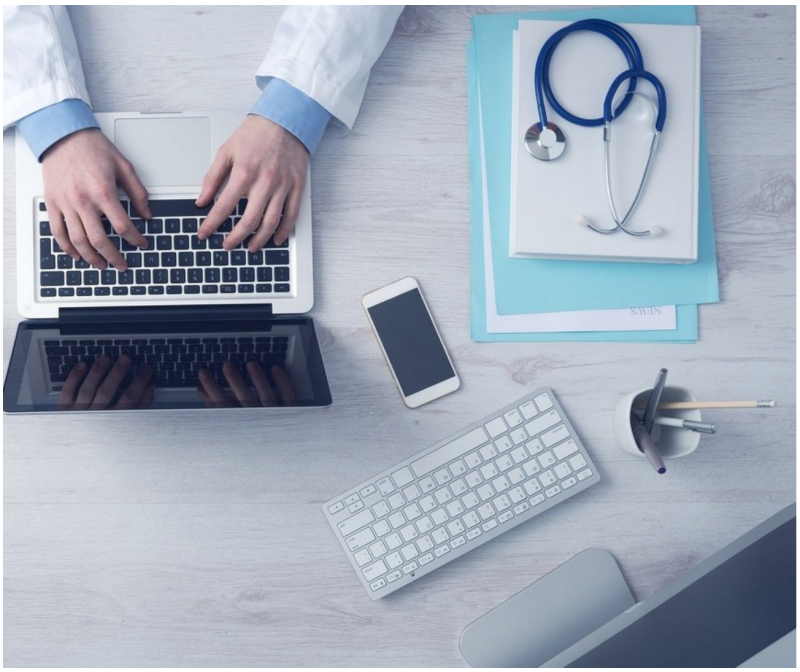

A Web Page
https://walkit.pt
WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
As minhas informações
As minhas sessões
Ajuda
WALKit
22-02-2022
Algoritmos:
Equilíbrio, Postura
Marcha, Sensores
Velocidade Máx.:
3 km/h
Modo Segurança:
Ativo
Equilíbrio:
Parado
Biofeedback:
Ativo
Equilíbrio:
Parado
Biofeedback:
Ativo
Marcha:
Redes Neuronais
Biofeedback:
Ativo
Sensores:
Infravermelhos,
Sistema Inercial,
Sonar, Jogo
Acelarómetro:
Tempo
Jogos:
Errou
Acertou
Infravermelho:
Tempo
WALKit
Frase cativante


A Web Page
https://walkit.pt
WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
As minhas informações
As minhas sessões
Ajuda
WALKit
Evolução Temporal
Sessões
4
Velocidade Média
3 km/h
Acelarômetro:
Tempo
Infravermelho:
Tempo
Células de Carga:
Jogo
Erro
Acerto
WALKit
Frase cativante

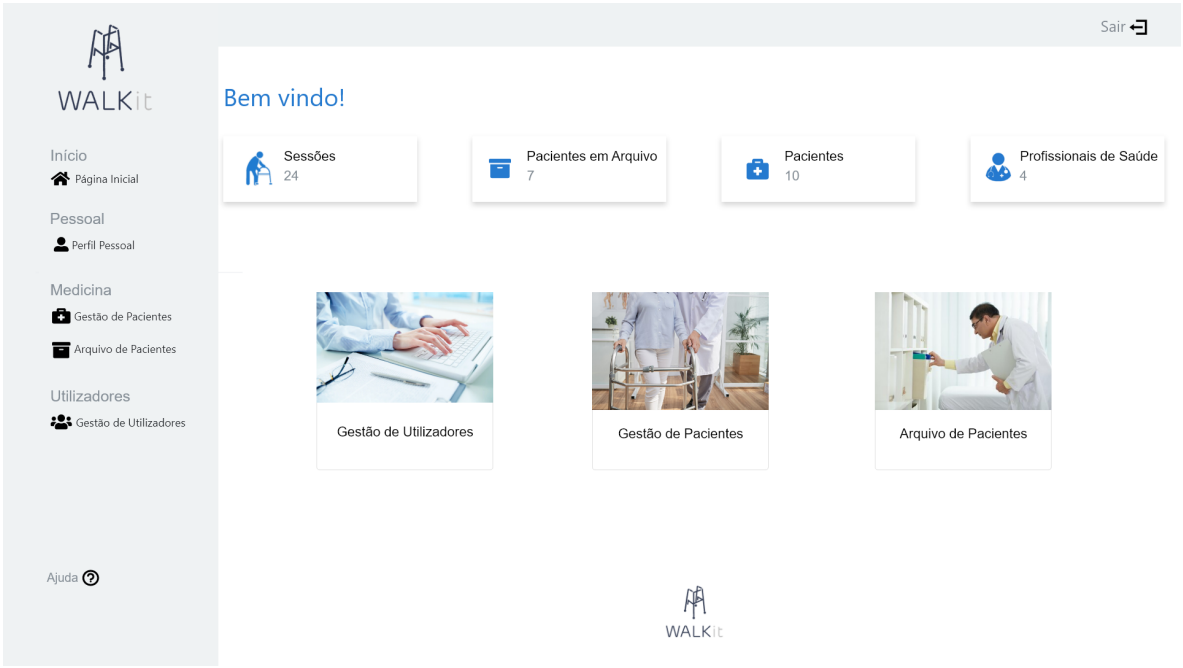
C

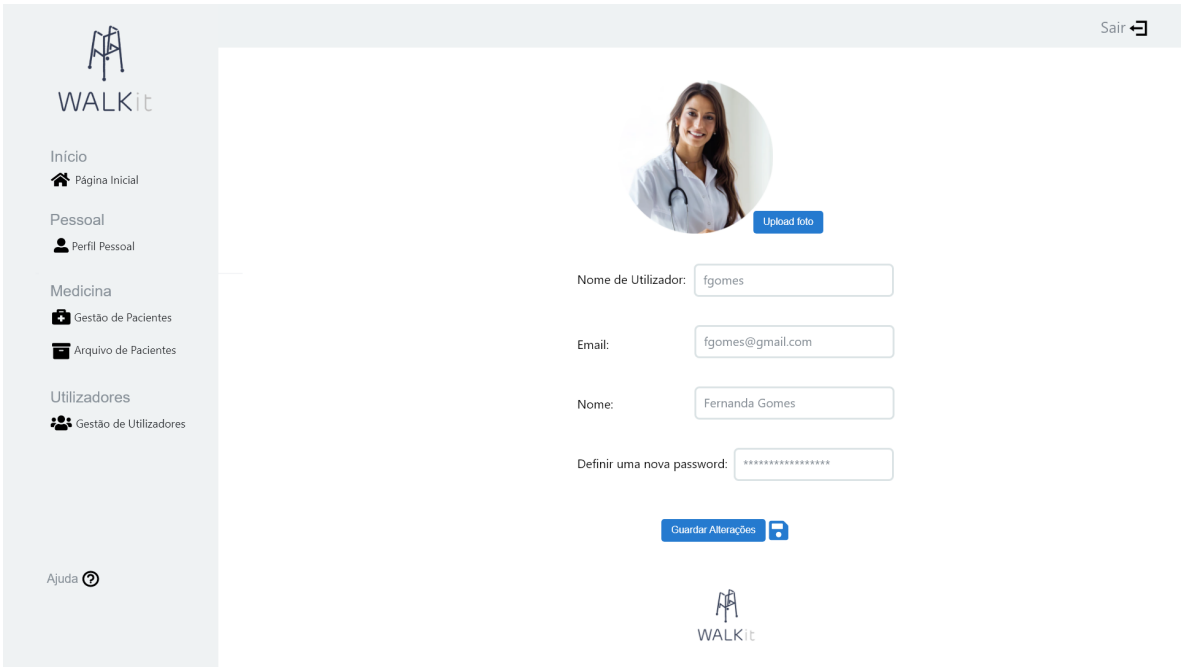
WEB APPLICATION PROTOTYPES

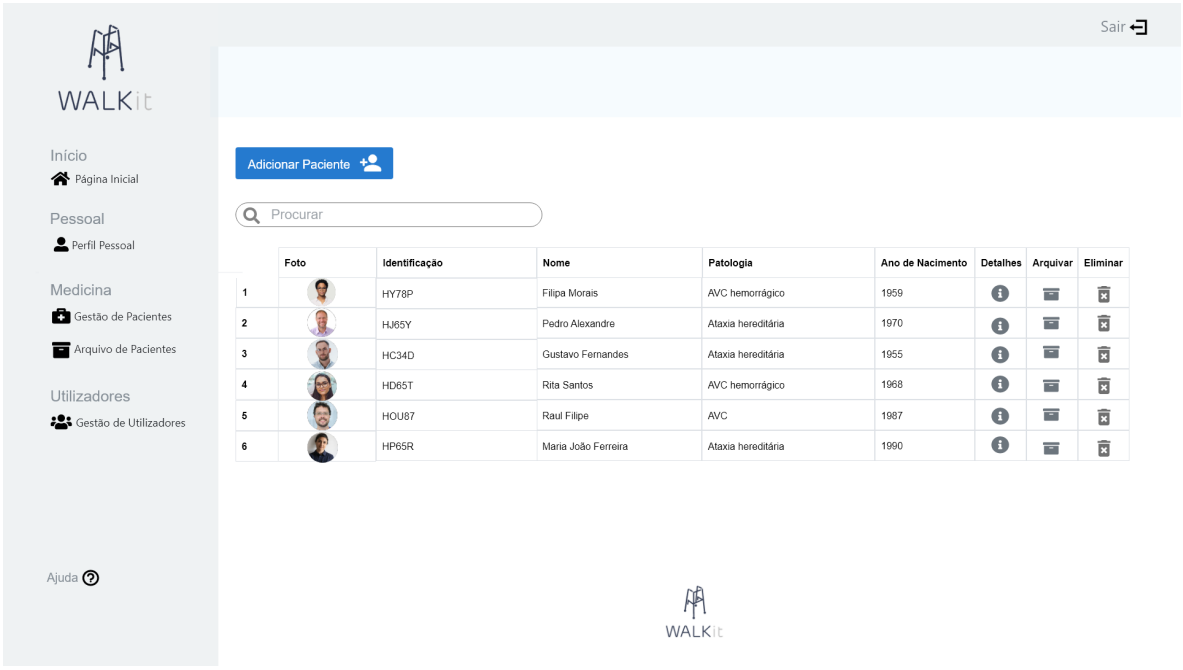
C.1 ADMINISTRATOR AND HEALTH PROFESSIONAL PROTOTYPES

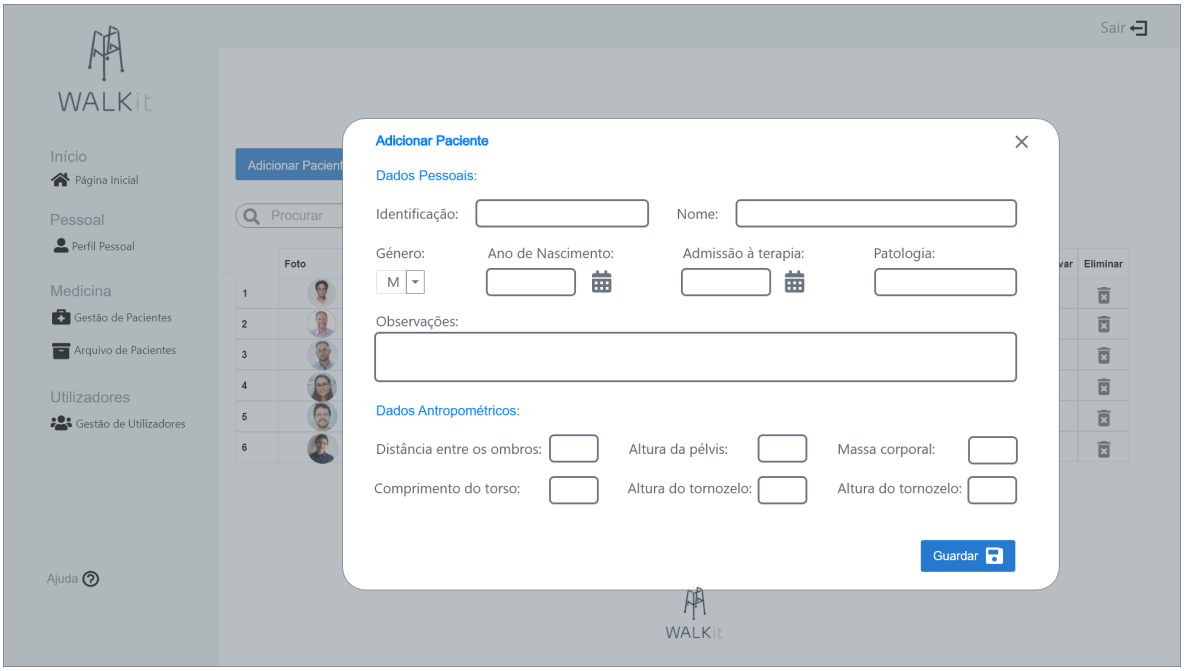


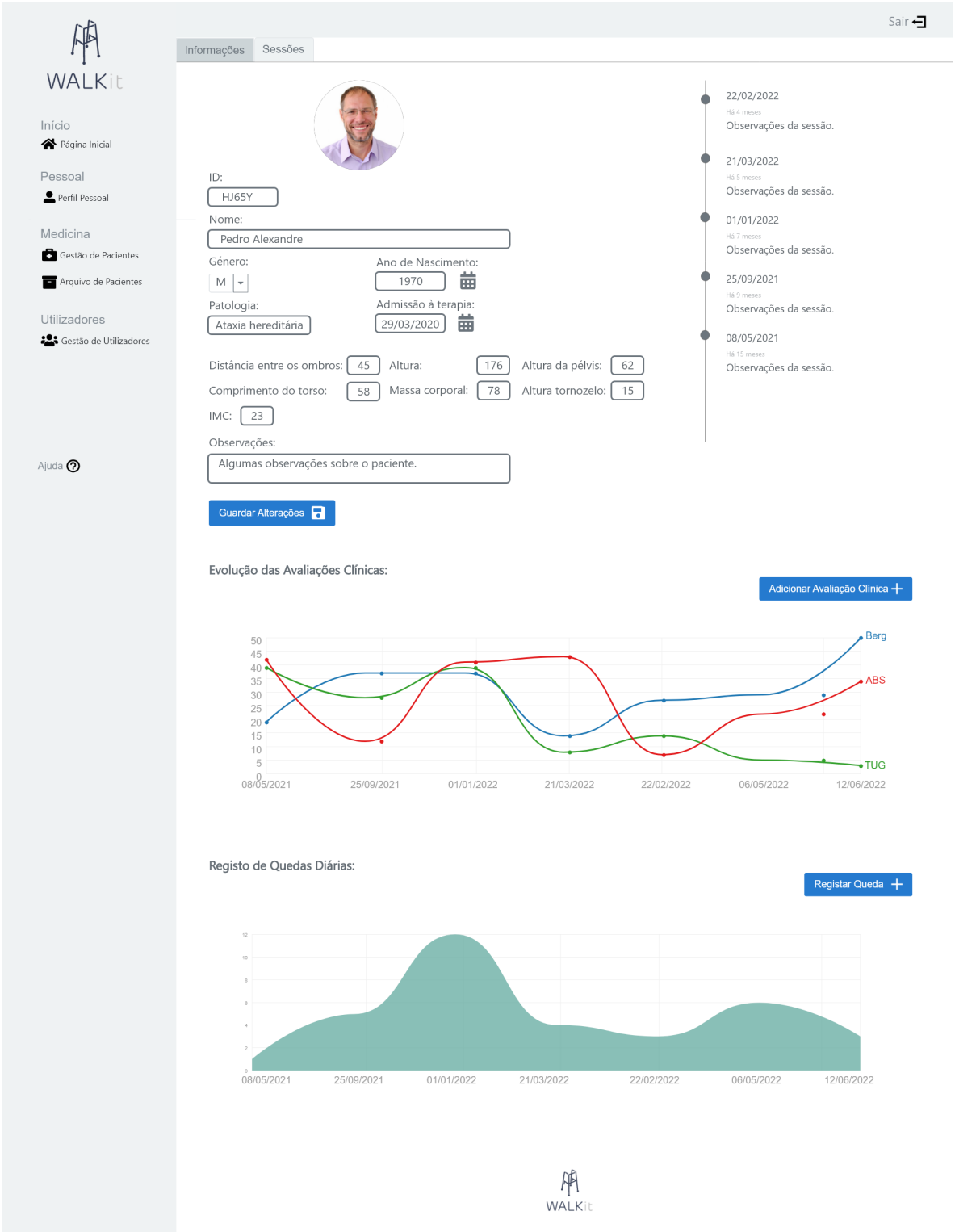


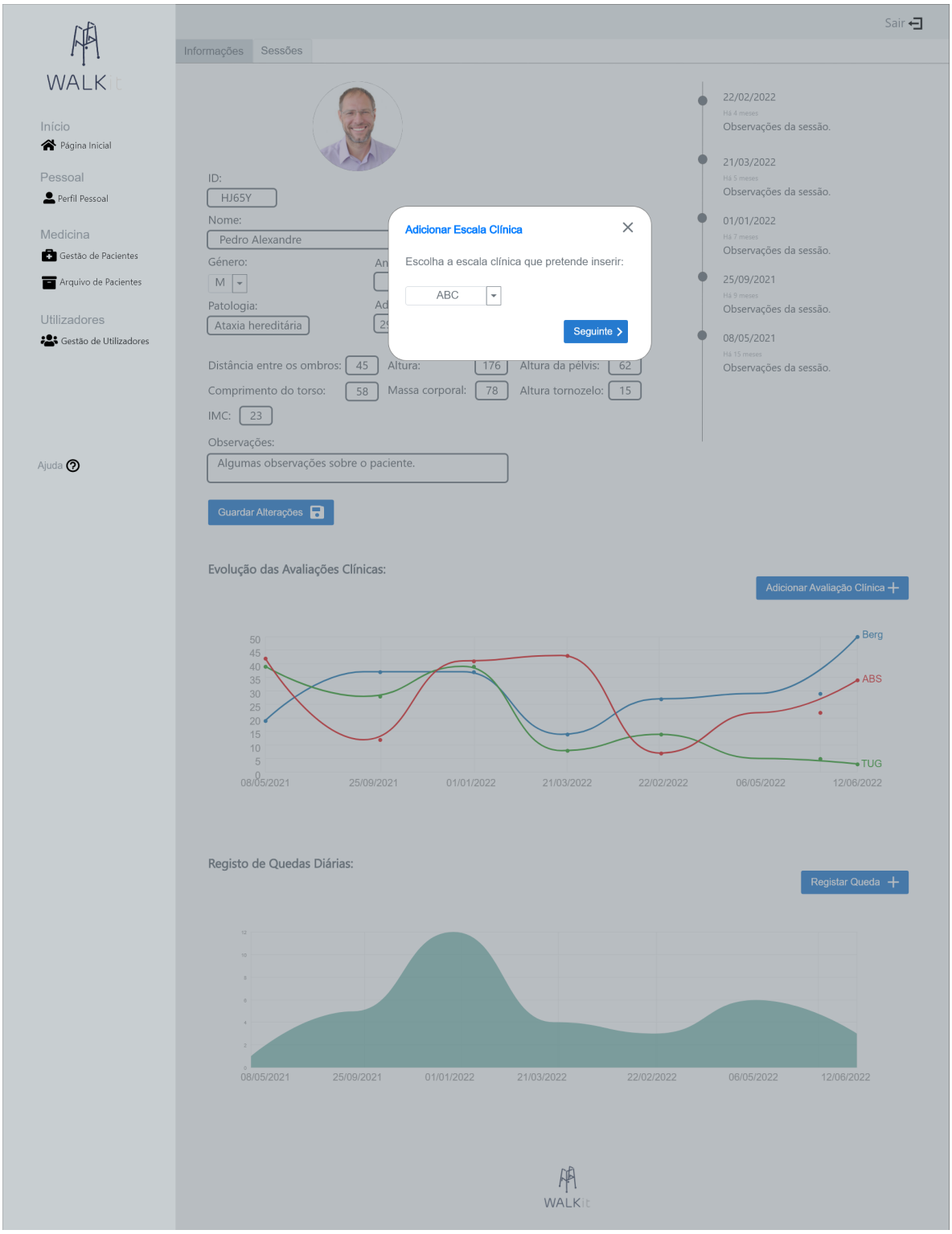

WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Gestão de Pacientes
Arquivo de Pacientes
Utilizadores
Gestão de Utilizadores
Ajuda
Sair
Bem vindo!
Sessões
24
Pacientes em Arquivo
7
Pacientes
10
Profissionais de Saúde
4
Gestão de Utilizadores
Gestão de Pacientes
Arquivo de Pacientes
WALKit


WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Gestão de Pacientes
Arquivo de Pacientes
Utilizadores
Gestão de Utilizadores
Ajuda
Sair
Upload foto
Nome de Utilizador:
fgomes
Email:
fgomes@gmail.com
Nome:
Fernanda Gomes
Definir uma nova password:
Guardar Alterações
WALKit

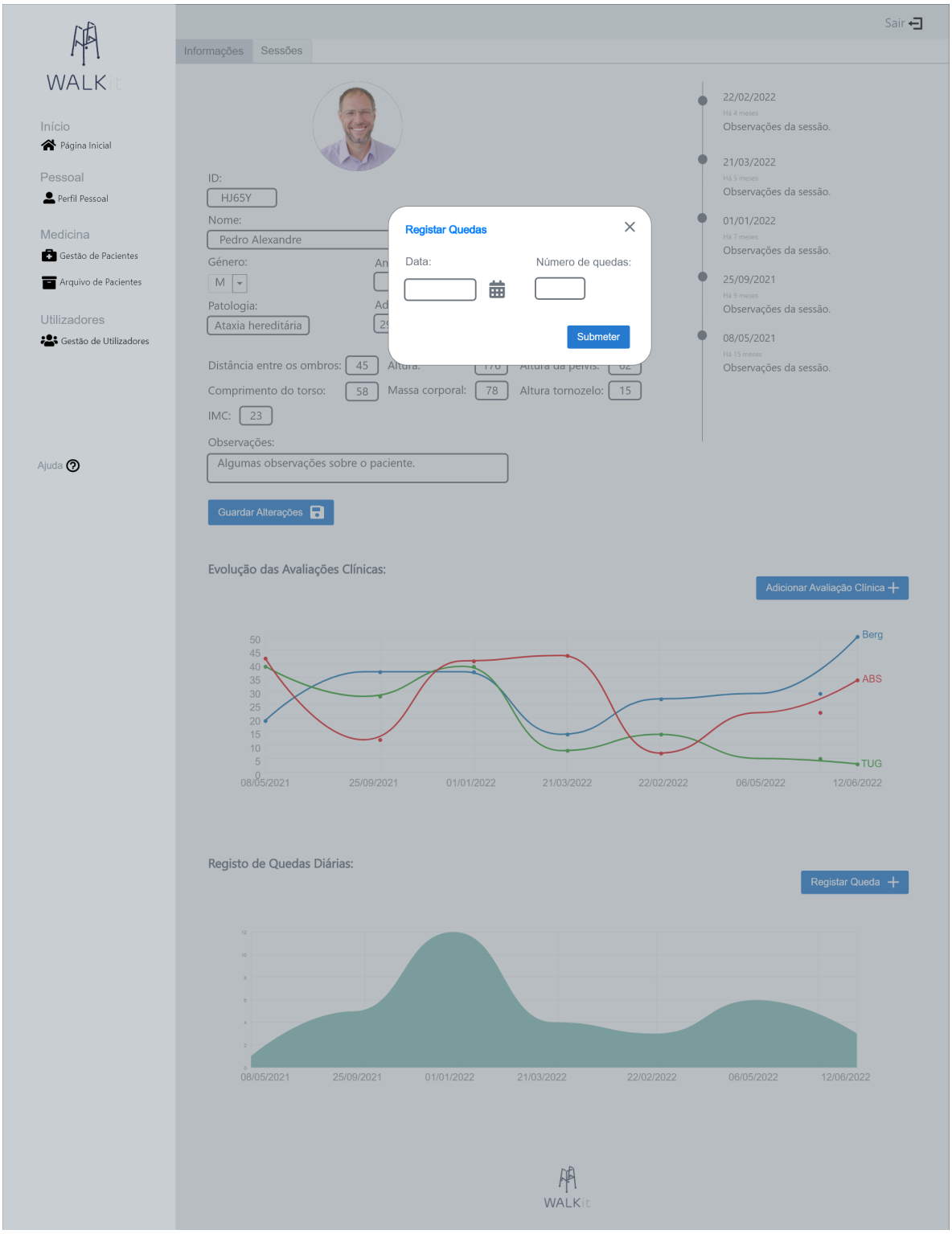

Sair
WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Gestão de Pacientes
Arquivo de Pacientes
Utilizadores
Gestão de Utilizadores
Ajuda
Adicionar Paciente
Procurar
Foto
Identificação
Nome
Patologia
Ano de Nacimento
Detalhes
Arquivar
Eliminar
1 HY78P Filipa Morais AVC hemorrágico 1959
2 HJ65Y Pedro Alexandre Ataxia hereditária 1970
3 HC34D Gustavo Fernandes Ataxia hereditária 1955
4 HD65T Rita Santos AVC hemorrágico 1968
5 HOU87 Raul Filipe AVC 1987
6 HP65R Maria João Ferreira Ataxia hereditária 1990
WALKit

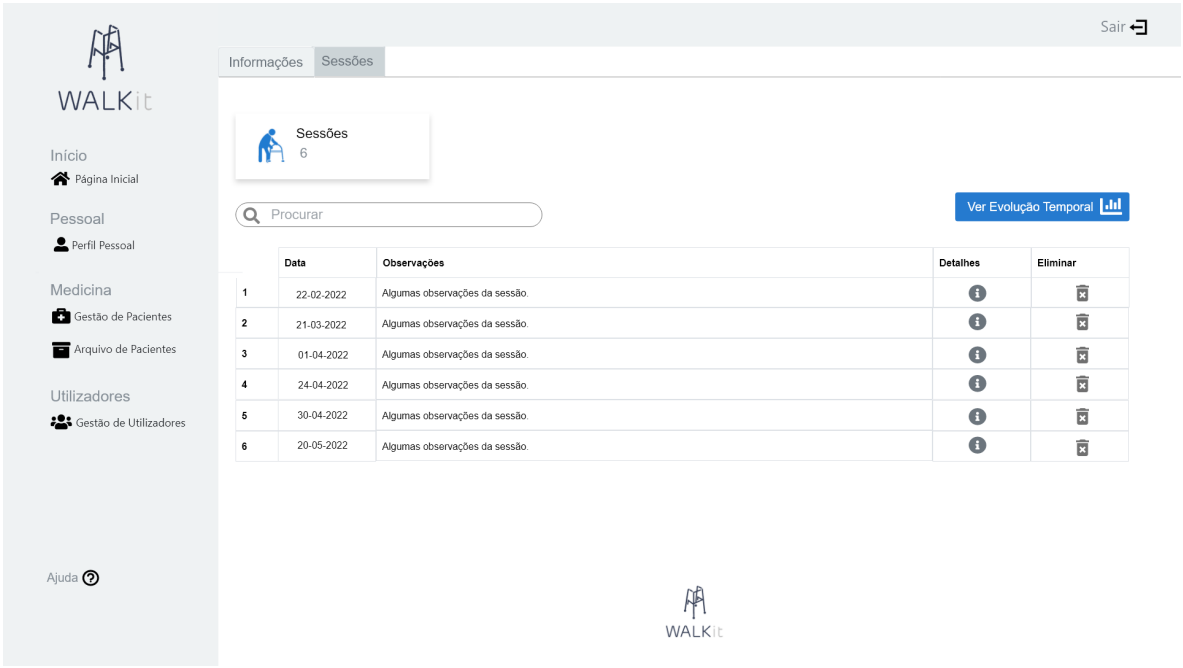

Sair
WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Gestão de Pacientes
Arquivo de Pacientes
Utilizadores
Gestão de Utilizadores
Ajuda
Adicionar Paciente
Dados Pessoais:
Identificação:
Nome:
Género:
M
Ano de Nascimento:
Admissão à terapia:
Patologia:
Observações:
Dados Antropométricos:
Distância entre os ombros:
Altura da pélvis:
Massa corporal:
Comprimento do torso:
Altura do tornozelo:
Altura do tornozelo:
Guardar
WALKit

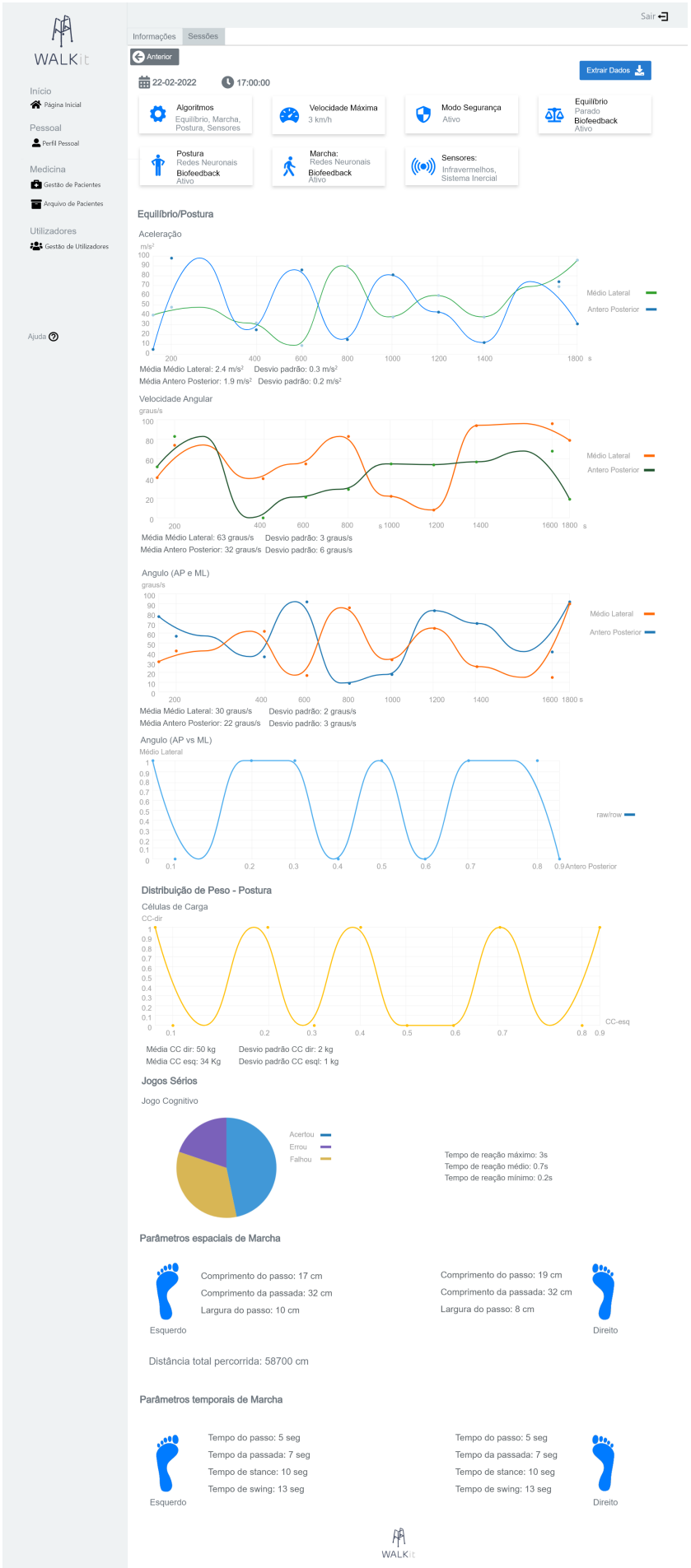

WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Gestão de Pacientes
Arquivo de Pacientes
Utilizadores
Gestão de Utilizadores
Ajuda
Sair
Informações
Sessões
ID:
HJ65Y
Nome:
Pedro Alexandre
Género:
M
Ano de Nascimento:
1970
Patologia:
Ataxia hereditária
Admissão à terapia:
29/03/2020
Distância entre os ombros: 45
Altura: 176
Altura da pélvis: 62
Comprimento do torso: 58
Massa corporal: 78
Altura tornozelo: 15
IMC: 23
Observações:
Algumas observações sobre o paciente.
Guardar Alterações
22/02/2022
Há 4 meses
Observações da sessão.
21/03/2022
Há 5 meses
Observações da sessão.
01/01/2022
Há 7 meses
Observações da sessão.
25/09/2021
Há 9 meses
Observações da sessão.
08/05/2021
Há 15 meses
Observações da sessão.
Evolução das Avaliações Clínicas:
Adicionar Avaliação Clínica
Berg
ABS
TUG
08/05/2021
25/09/2021
01/01/2022
21/03/2022
22/02/2022
06/05/2022
12/06/2022
Registo de Quedas Diárias:
Registar Queda
WALKit

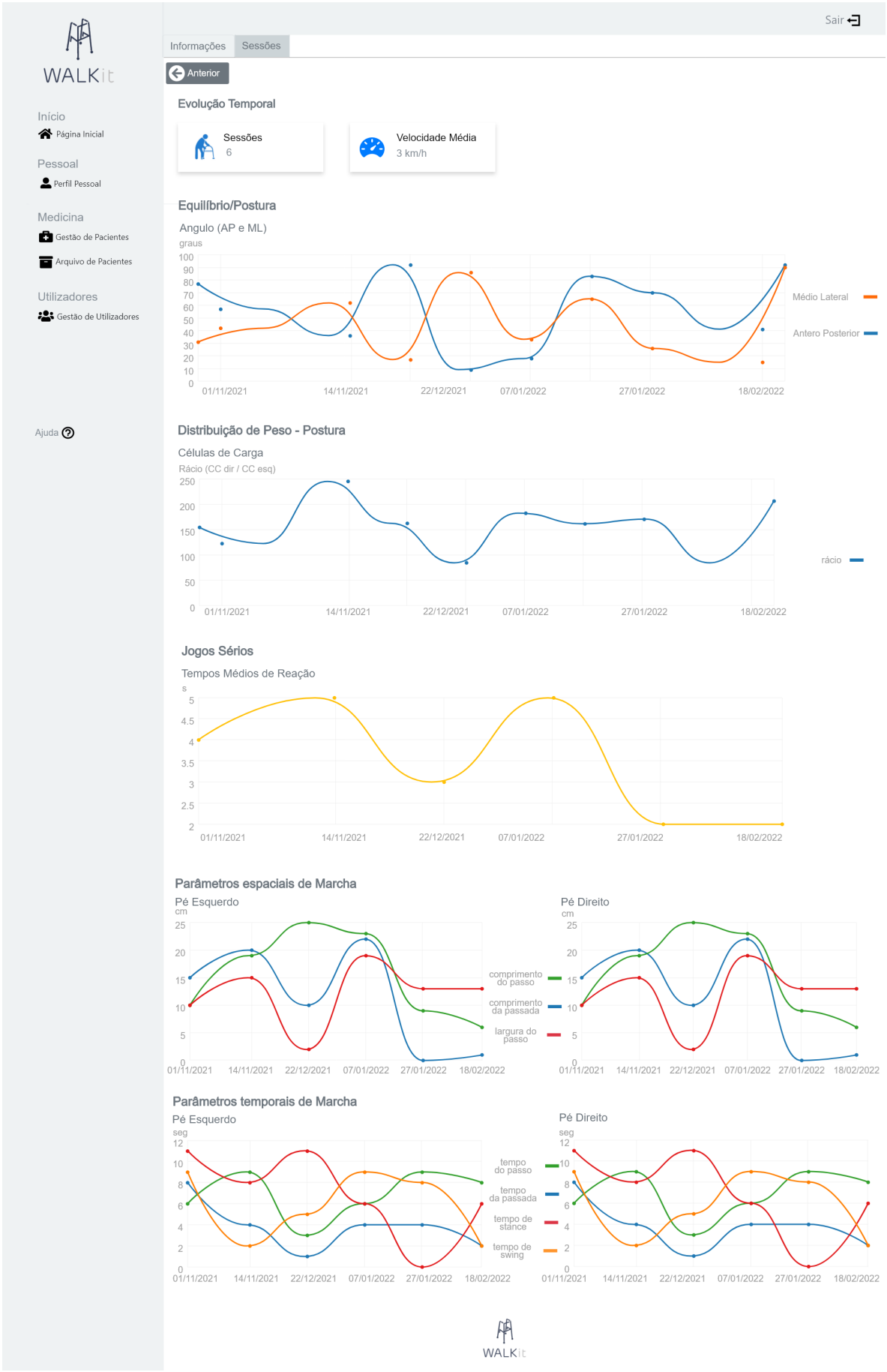

WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Gestão de Pacientes
Arquivo de Pacientes
Utilizadores
Gestão de Utilizadores
Ajuda
Sair
Informações
Sessões
ID:
HJ65Y
Nome:
Pedro Alexandre
Género:
M
Patologia:
Ataxia hereditária
Adicionar Escala Clínica
Escolha a escala clínica que pretende inserir:
ABC
Seguinte
22/02/2022
Observações da sessão.
21/03/2022
Observações da sessão.
01/01/2022
Observações da sessão.
25/09/2021
Observações da sessão.
08/05/2021
Observações da sessão.
Distância entre os ombros: 45
Altura: 176
Altura da pélvis: 62
Comprimento do torso: 58
Massa corporal: 78
Altura tornozelo: 15
IMC: 23
Observações:
Algumas observações sobre o paciente.
Guardar Alterações
Evolução das Avaliações Clínicas:
Adicionar Avaliação Clínica
Berg
ABS
TUG
08/05/2021
25/09/2021
01/01/2022
21/03/2022
22/02/2022
06/05/2022
12/06/2022
Registo de Quedas Diárias:
Registar Queda
WALKit

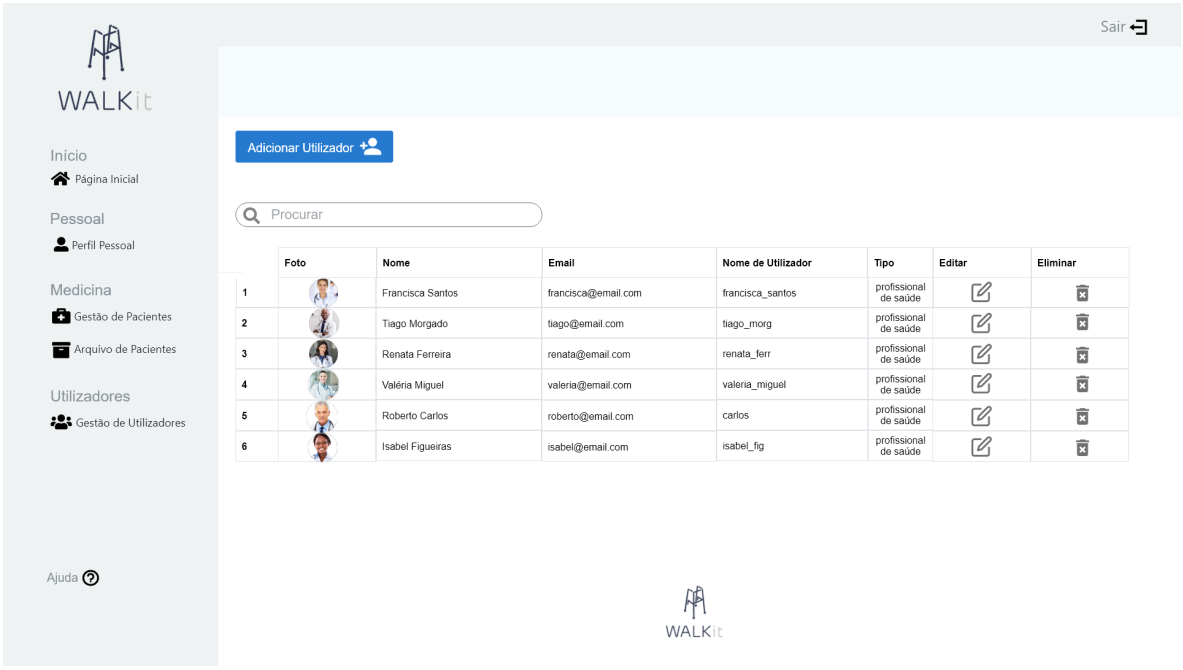

WALKit
Sair
Informações
Sessões
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Gestão de Pacientes
Arquivo de Pacientes
Utilizadores
Gestão de Utilizadores
Ajuda
Pé Esquerdo
ID:
Nome
Género
Patolo
Distân
Comp
IMC:
Obser
Guardar
Escala de Confiança no Equilíbrio Específica para a Atividade
Por favor, indique o seu nível de autoconfiança para realizar cada uma das seguintes atividades, escolhendo o número correspondente na seguinte escala de avaliação:
0% 10 20 30 40 50 60 70 80 90 100%
Sem nenhuma confiança - Confiança completa
Que confiança tem em que não vai perder o equilíbrio ou ficar instável quando…
1. Anda em casa?
2. Sobre ou desce escadas?
3. Se inclina para a frente para apanhar um chinelo do fundo de um armário?
4. Alcança uma lata pequena de uma prateleira ao nível dos olhos?
5. Se põe em bicos de pés para alcançar alguma coisa acima da sua cabeça?
6. Se põe em pé em cima duma cadeira para tentar alcançar alguma coisa?
7. Varre o chão?
8. Sai de um prédio e se dirige a um carro parado em frente à porta?
9. Entra ou sai de um carro?
10. Atravessa um parque de estacionamento até um centro comercial ou supermercado?
11. Sobe ou desce uma rampa?
12. Anda num centro comercial ou supermercado com muita gente onde as pessoas passam rapidamente por si?
13. Leva encontrões de pessoas quando anda num centro comercial ou supermercado?
14. Entra ou sai de uma escada rolante segura(o) ao corrimão?
15. Entra ou sai de uma escada rolante com embrulhos ou sacos na mão, de forma que não se pode segurar ao corrimão?
16. Anda na rua em passeios escorregadios?
Submeter formulário
Evolução das Avaliações Clínicas:
Adicionar Avaliação Clínica
Berg
ABS
TUG
08/05/2021
25/09/2021
01/01/2022
21/03/2022
22/02/2022
06/05/2022
12/06/2022
Registo de Quedas Diárias:
Registar Queda
WALKit

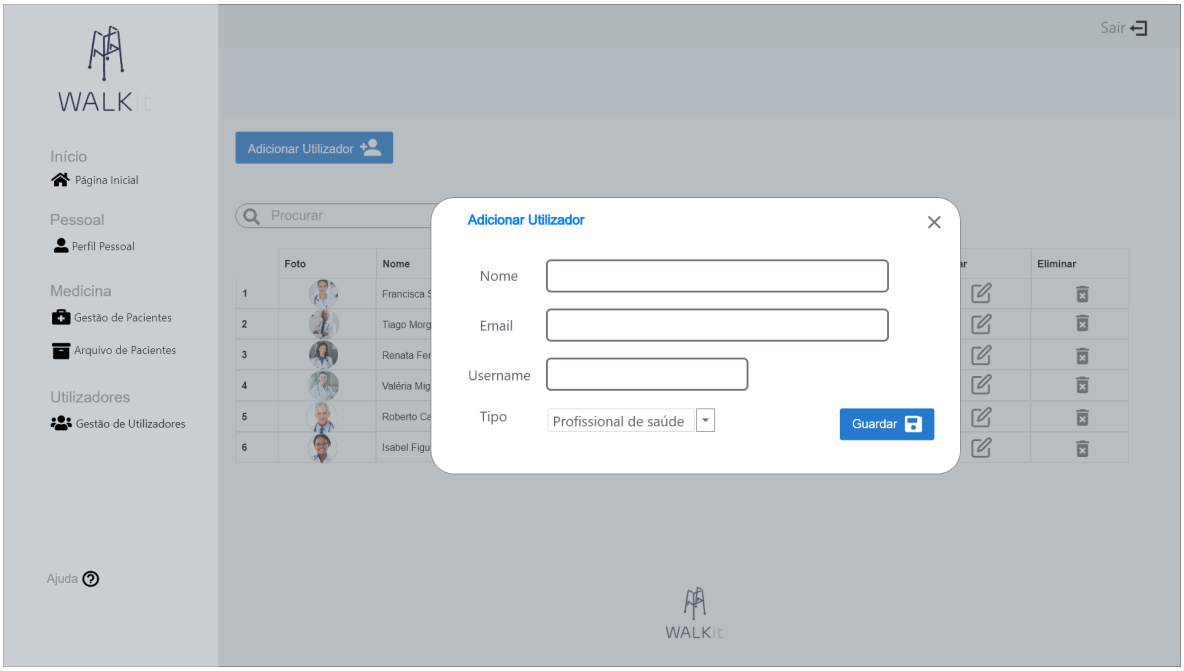

WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Gestão de Pacientes
Arquivo de Pacientes
Utilizadores
Gestão de Utilizadores
Ajuda
Sair
Informações
Sessões
ID:
HJ65Y
Nome:
Pedro Alexandre
Género:
M
Patologia:
Ataxia hereditária
Distância entre os ombros: 45
Comprimento do torso: 58
Massa corporal: 78
Altura tornozelo: 15
IMC: 23
Observações:
Algumas observações sobre o paciente.
Guardar Alterações
22/02/2022
Observações da sessão.
21/03/2022
Observações da sessão.
01/01/2022
Observações da sessão.
25/09/2021
Observações da sessão.
08/05/2021
Observações da sessão.
Registar Quedas
Data:
Número de quedas:
Submeter
Evolução das Avaliações Clínicas:
Adicionar Avaliação Clínica
Berg
ABS
TUG
08/05/2021
25/09/2021
01/01/2022
21/03/2022
22/02/2022
06/05/2022
12/06/2022
Registo de Quedas Diárias:
Registar Queda
WALKit


WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Gestão de Pacientes
Arquivo de Pacientes
Utilizadores
Gestão de Utilizadores
Ajuda
Sair
Informações
Sessões
Sessões
6
Procurar
Ver Evolução Temporal
Data
Observações
Detalhes
Eliminar
1 22-02-2022 Algumas observações da sessão.
2 21-03-2022 Algumas observações da sessão.
3 01-04-2022 Algumas observações da sessão.
4 24-04-2022 Algumas observações da sessão.
5 30-04-2022 Algumas observações da sessão.
6 20-05-2022 Algumas observações da sessão.
WALKit


WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Gestão de Pacientes
Arquivo de Pacientes
Utilizadores
Gestão de Utilizadores
Ajuda
Sair
Informações
Sessões
Anterior
Extrair Dados
22-02-2022
17:00:00
Algoritmos
Equilíbrio, Marcha, Postura, Sensores
Velocidade Máxima
3 km/h
Modo Segurança
Ativo
Equilíbrio
Parado
Biofeedback
Ativo
Postura
Redes Neuronais
Biofeedback
Ativo
Marcha:
Redes Neuronais
Biofeedback
Ativo
Sensores:
Infravermelhos, Sistema Inercial
Equilíbrio/Postura
Aceleração
m/s²
Médio Lateral
Antero Posterior
s
Média Médio Lateral: 2.4 m/s² Desvio padrão: 0.3 m/s²
Média Antero Posterior: 1.9 m/s² Desvio padrão: 0.2 m/s²
Velocidade Angular
graus/s
Médio Lateral
Antero Posterior
Média Médio Lateral: 63 graus/s Desvio padrão: 3 graus/s
Média Antero Posterior: 32 graus/s Desvio padrão: 6 graus/s
Angulo (AP e ML)
graus/s
Médio Lateral
Antero Posterior
Média Médio Lateral: 30 graus/s Desvio padrão: 2 graus/s
Média Antero Posterior: 22 graus/s Desvio padrão: 3 graus/s
Angulo (AP vs ML)
Médio Lateral
raw/row
Antero Posterior
Distribuição de Peso - Postura
Células de Carga
CC-dir
CC-esq
Média CC dir: 50 kg Desvio padrão CC dir: 2 kg
Média CC esq: 34 Kg Desvio padrão CC esql: 1 kg
Jogos Sérios
Jogo Cognitivo
Acertou
Errou
Falhou
Tempo de reação máximo: 3s
Tempo de reação médio: 0.7s
Tempo de reação mínimo: 0.2s
Parâmetros espaciais de Marcha
Comprimento do passo: 17 cm
Comprimento da passada: 32 cm
Largura do passo: 10 cm
Esquerdo
Comprimento do passo: 19 cm
Comprimento da passada: 32 cm
Largura do passo: 8 cm
Direito
Distância total percorrida: 58700 cm
Parâmetros temporais de Marcha
Tempo do passo: 5 seg
Tempo da passada: 7 seg
Tempo de stance: 10 seg
Tempo de swing: 13 seg
Esquerdo
Tempo do passo: 5 seg
Tempo da passada: 7 seg
Tempo de stance: 10 seg
Tempo de swing: 13 seg
Direito
WALKit


WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Gestão de Pacientes
Arquivo de Pacientes
Utilizadores
Gestão de Utilizadores
Ajuda
Sair
Informações
Sessões
Anterior
Evolução Temporal
Sessões
6
Velocidade Média
3 km/h
Equilíbrio/Postura
Angulo (AP e ML)
graus
Médio Lateral
Antero Posterior
Distribuição de Peso - Postura
Células de Carga
Rácio (CC dir / CC esq)
rácio
Jogos Sérios
Tempos Médios de Reação
s
Parâmetros espaciais de Marcha
Pé Esquerdo
Pé Direito
cm
comprimento do passo
comprimento da passada
largura do passo
Parâmetros temporais de Marcha
Pé Esquerdo
Pé Direito
seg
tempo do passo
tempo da passada
tempo de stance
tempo de swing
01/11/2021
14/11/2021
22/12/2021
07/01/2022
27/01/2022
18/02/2022
WALKit

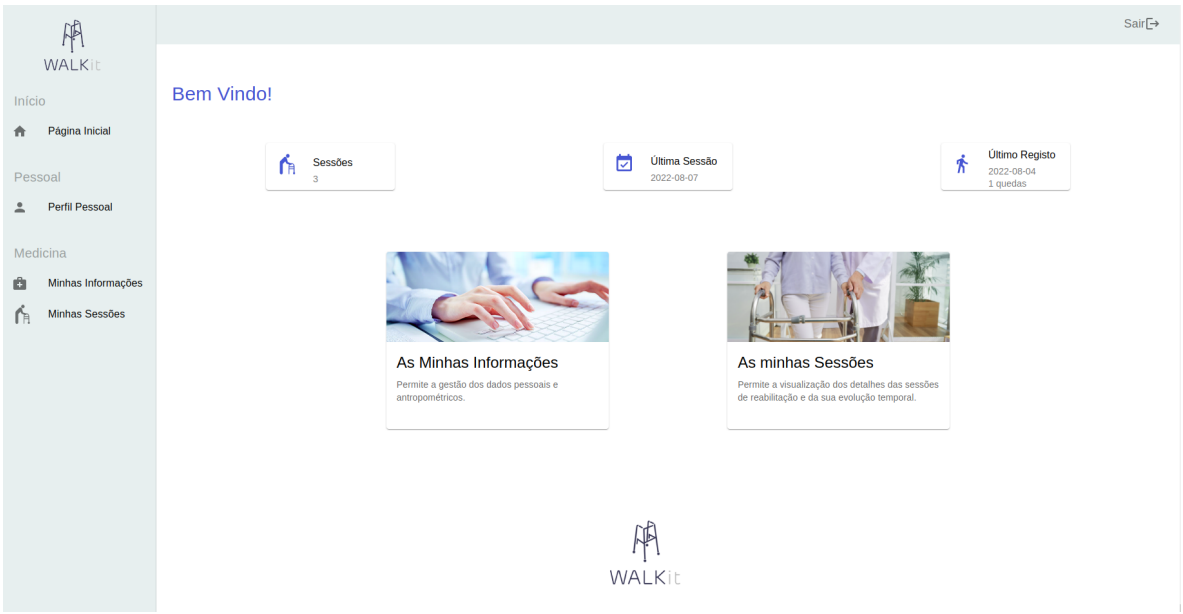

WALKit
Sair
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Gestão de Pacientes
Arquivo de Pacientes
Utilizadores
Gestão de Utilizadores
Ajuda
Adicionar Utilizador
Procurar
Foto
Nome
Email
Nome de Utilizador
Tipo
Editar
Eliminar
1 Francisca Santos francisca@email.com francisca_santos profissional de saúde
2 Tiago Morgado tiago@email.com tiago_morg profissional de saúde
3 Renata Ferreira renata@email.com renata_ferr profissional de saúde
4 Valéria Miguel valeria@email.com valeria_miguel profissional de saúde
5 Roberto Carlos roberto@email.com carlos profissional de saúde
6 Isabel Figueiras isabel@email.com isabel_fig profissional de saúde
WALKit

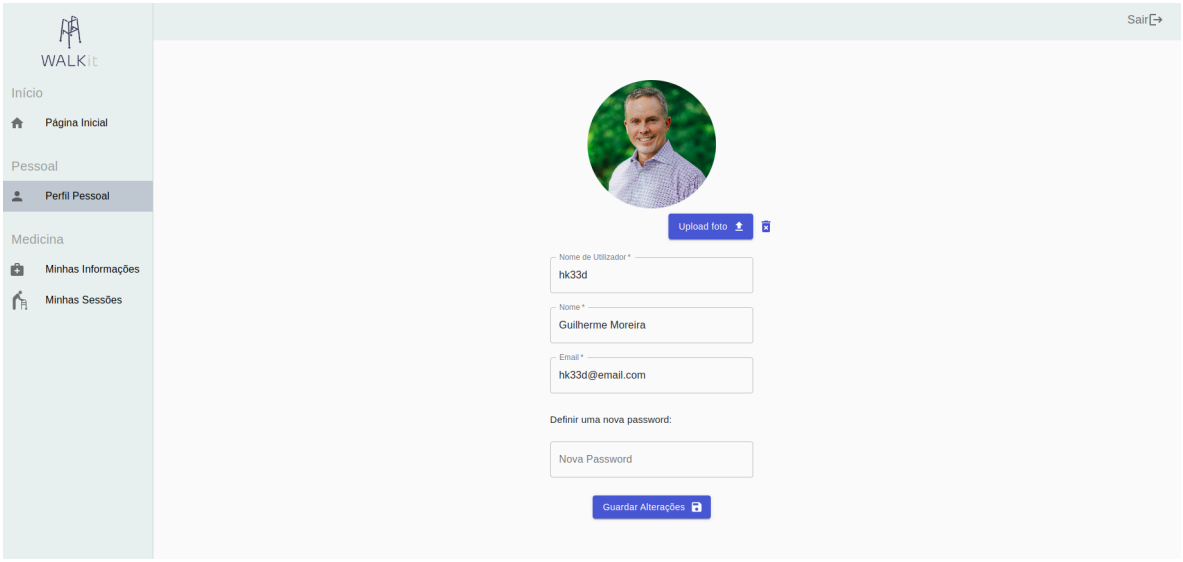

WALKit
Sair
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Gestão de Pacientes
Arquivo de Pacientes
Utilizadores
Gestão de Utilizadores
Ajuda
Adicionar Utilizador
Procurar
Adicionar Utilizador
Nome
Email
Username
Tipo
Profissional de saúde
Guardar
Eliminar
WALKit

D

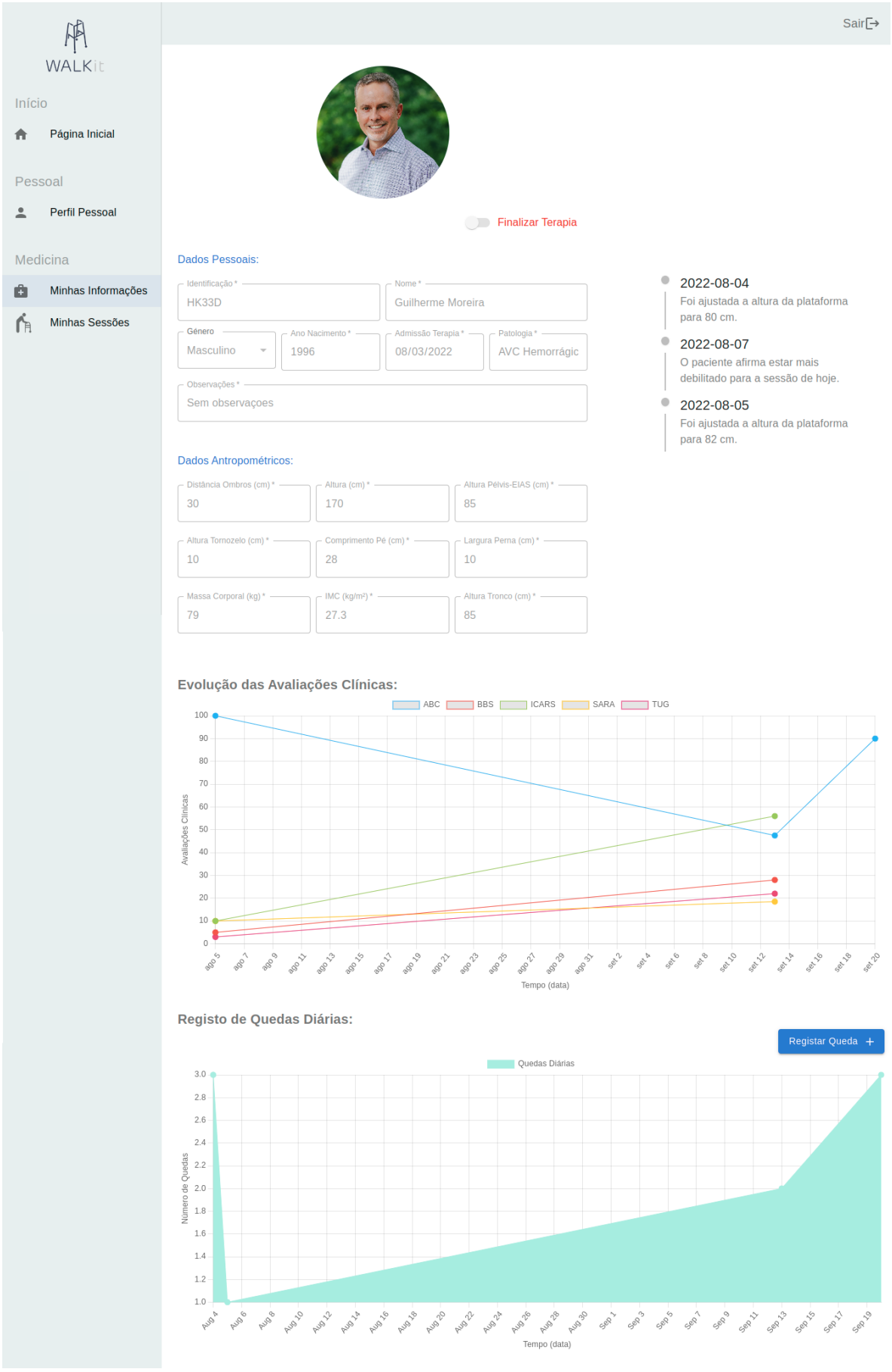
WEB APPLICATION RESULTS

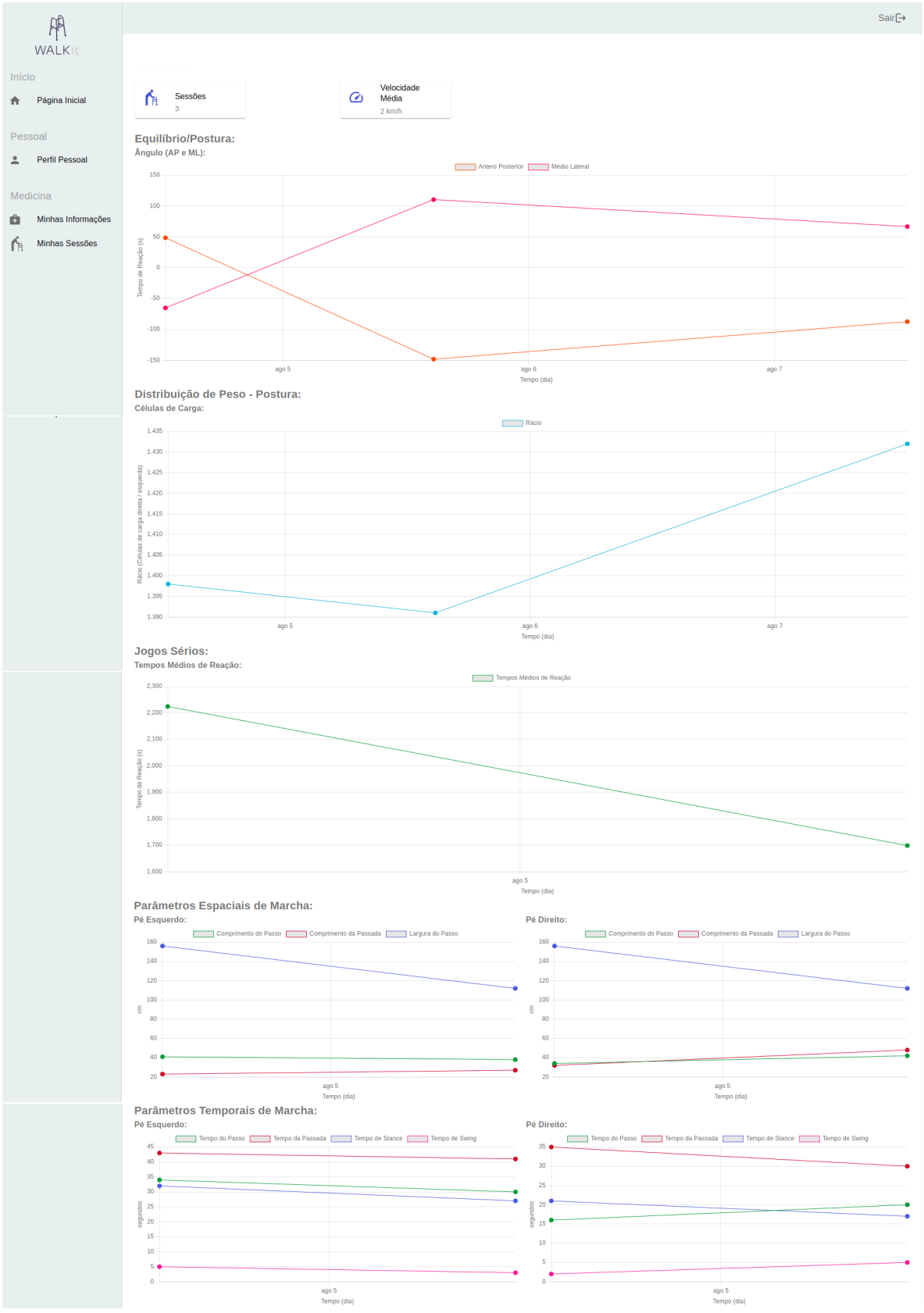
D.1 PATIENT INTERFACES



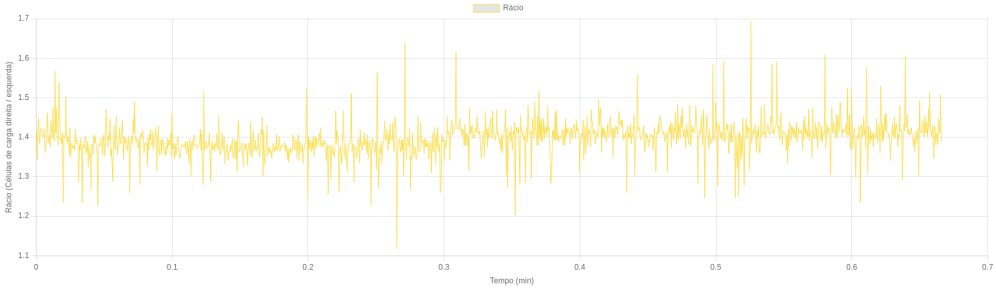

WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Minhas Informações
Minhas Sessões
Sair
Upload foto
Nome de Utilizador *
hk33d
Nome *
Guilherme Moreira
Email *
hk33d@email.com
Definir uma nova password:
Nova Password
Guardar Alterações


WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Minhas Informações
Minhas Sessões
Sair
Ver Evolução Temporal
Procurar
Dia
Hora
Observações
Detalhes
2022-08-04
10:31:07
Foi ajustada a altura da plataforma para 80 cm.
2022-08-07
10:58:19
O paciente afirma estar mais debilitado para a sessão de hoje.
2022-08-05
12:42:42
Foi ajustada a altura da plataforma para 82 cm.
Rows per page: 10
1–3 of 3


WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Minhas Informações
Minhas Sessões
Sair
Finalizar Terapia
Dados Pessoais:
Identificação *
HK33D
Nome *
Guilherme Moreira
Género
Masculino
Ano Nacimento *
1996
Admissão Terapia *
08/03/2022
Patologia *
AVC Hemorrágic
Observações *
Sem observaçoes
2022-08-04
Foi ajustada a altura da plataforma para 80 cm.
2022-08-07
O paciente afirma estar mais debilitado para a sessão de hoje.
2022-08-05
Foi ajustada a altura da plataforma para 82 cm.
Dados Antropométricos:
Distância Ombros (cm) *
30
Altura (cm) *
170
Altura Pélvis-EIAS (cm) *
85
Altura Tornozelo (cm) *
10
Comprimento Pé (cm) *
28
Largura Perna (cm) *
10
Massa Corporal (kg) *
79
IMC (kg/m²) *
27.3
Altura Tronco (cm) *
85
Evolução das Avaliações Clínicas:
ABC
BBS
ICARS
SARA
TUG
Avaliações Clínicas
Tempo (data)
Registo de Quedas Diárias:
Registar Queda
Quedas Diárias
Número de Quedas
Tempo (data)


WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Minhas Informações
Minhas Sessões
Sair
Sessões
3
Velocidade Média
2 km/h
Equilíbrio/Postura:
Ângulo (AP e ML):
Antero Posterior
Médio Lateral
Tempo de Reação (s)
Tempo (dia)
ago 5
ago 6
ago 7
Distribuição de Peso - Postura:
Células de Carga:
Rácio
Rácio (Células de carga direita / esquerda)
Jogos Sérios:
Tempos Médios de Reação:
Tempos Médios de Reação
Parâmetros Espaciais de Marcha:
Pé Esquerdo:
Pé Direito:
Comprimento do Passo
Comprimento da Passada
Largura do Passo
cm
Parâmetros Temporais de Marcha:
Tempo do Passo
Tempo da Passada
Tempo de Stance
Tempo de Swing
segundos


WALKit
Início
Página Inicial
Pessoal
Perfil Pessoal
Medicina
Minhas Informações
Minhas Sessões
Sair
Extrair Dados
2022-08-04
10:31:07
Algoritmos
Manual-Sensores-Equilíbrio-Marcha-Postura-Cognitivo
Velocidade Máxima (Km/h)
1
Modo Segurança
Desativado
Equilíbrio
Estático, sem biofeedback
Postura
Visão por Computador, biofeedback ativo
Marcha
Visão por Computador, sem biofeedback
Sensores
Infravermelhos Sonar Células de Carga Velocidade Sistema Inercial
Equilíbrio/Postura:
Aceleração:
Médio Lateral
Antero Posterior
Aceleração (m/s²)
Tempo (min)
Média Médio Lateral: -0.45 ± 0.00 m/s²
Média Antero Posterior: -0.94 ± 0.00 m/s²
Velocidade Angular:
Velocidade Angular (graus/s)
Média Médio Lateral: -1.08 ± 0.11 graus/s
Média Antero Posterior: 0.33 ± 0.09 graus/s
Ângulo (AP e ML):
Ângulo (graus)
Média Médio Lateral: 48.46 ± 27.83 graus
Média Antero Posterior: -65.20 ± 37.44 graus
Ângulo (AP vs ML):
Médio Lateral (graus)
Antero Posterior (graus)
Distribuição de Peso - Postura:
Células de Carga:

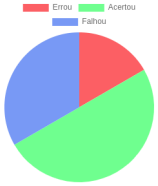


Média C.C. Esquerda: 10.45 ± 0.26 kg

Média C.C. Direita: 14.59 ± 0.33 kg

Jogos Sérios:

Jogo Cognitivo:



Tempo de Reação Máxima: 8896.96 ms

Tempo de Reação Médio: 2223.99 ± 3338.29 ms

Tempo de Reação Mínimo: 417.03 ms

Parâmetros Espaciais de Marcha:



Esquerdo

Comprimento do passo: 58 ± 8.8 cm

Comprimento da passada: 113 ± 15 cm

Largura do passo: 5.6 ± 0.1 cm

Comprimento do passo: 56 ± 7.6 cm

Comprimento da passada: 109 ± 17 cm

Largura do passo: 6.8 ± 0.9 cm



Direito

Distância Total Percorrida: 15000 cm

Parâmetros Temporais de Marcha:



Esquerdo

Tempo do passo: 0.5 ± 0.1 s

Tempo da passada: 1.1 ± 0.3 s

Tempo de stance: 0.6 ± 0.1 s

Tempo de swing: 0.4 ± 0.02 s

Tempo do passo: 0.6 ± 0.2 s

Tempo da passada: 1.2 ± 0.1 s

Tempo de stance: 0.7 ± 0.2 s

Tempo de swing: 0.4 ± 0.05 s



Direito