DEPARTMENT OF ELECTRICAL
AND COMPUTER ENGINEERING

ÁLISON VÁLTER DIAS
Bachelor Degree in ELECTRICAL AND COMPUTER ENGINEERING

PLATFORM FOR AI-DRIVEN MEDICAL DATA ANALYSIS TO SUPPORT CLINICAL DECISION.

MASTER IN ELECTRICAL AND COMPUTER ENGINEERING

NOVA University Lisbon
November, 2021

DEPARTMENT OF ELECTRICAL
AND COMPUTER ENGINEERING

PLATFORM FOR AI-DRIVEN MEDICAL DATA ANALYSIS TO SUPPORT CLINICAL DECISION.

ÁLISON VÁLTER DIAS
Bachelor Degree in ELECTRICAL AND COMPUTER ENGINEERING

Adviser: Fernando Luís Ferreira
Invited Assistant Professor, NOVA University Lisbon

Co-adviser: Ricardo Luis Rosa Jardim Gonçalves
Full Professor, NOVA University Lisbon

Examination Committee:

Chair: Ana Inês da Silva Oliveira
Assistant Professor, NOVA University Lisbon

Rapporteur: João Filipe dos Santos Sarraipa
Invited Assistant Professor, NOVA University Lisbon

MASTER IN ELECTRICAL AND COMPUTER ENGINEERING

NOVA University Lisbon
November, 2021

Platform for AI-driven medical data analysis to support clinical decision.

To my parents.

Acknowledgements

This dissertation represents the conclusion of a great phase of my life. Therefore, I would like to express my appreciation for all opportunities that NOVA School of Science and Technology gave me and also my adviser, Professor Fernando Luís Ferreira.

I must thank my colleagues and friends André Barradas, Alexandru Tabarcea, Diogo Simões, Miguel Azevedo and Ricardo Cruz with whom I spent most of my academic time, learning and overcoming difficulties. Without them this stage would not be the same and I will be eternally grateful for that.

To my parents whom I owe everything, from the education to the reference that they represent on my life. Without their support I could not have accomplished this stage.

"Tudo vale a pena quando a alma não é pequena." (Fernando Pessoa)

Abstract

Cancer is one of the leading causes of death on the world and surviving its treatment does not mean that the process is over. Several patients that have undergone cancer treatment, feel insecure in relation to their health, due to the stress and anxiety of cancer reappearance and post-treatment symptoms such as: sleeping disorders, fatigue and memory problems, pain, anxiety, and stress.

Patients that undergone cancer treatment are followed periodically by a clinician, that evaluates its clinical situation, but also, his Quality of Life. This information is vital to understand the patient well-being, since cancer as a huge impact on all aspects of the patient's life. Nevertheless, clinicians lack on tools capable of measuring objectively the patient's Quality of Life, nor tools that enable more data visualization that could improve the clinician's decision-making.

So, the purposed aim of this dissertation is to provide a Clinical Decision Support System Platform with visualization tools capable of giving information from patients, gathered from a wearable device and a smart scale, and using Fuzzy Logic, an Artificial Intelligence subset, to give new insights about patient well-being.

The designed CDSS Platform was able to integrate commercially used smart device, with minimal human intervention required. Also, the data gathered from those devices was used to create a continuous monitoring system, associated with visualization tools that enhanced the clinician knowledge of the patient. Furthermore, an indicator denominated as Patient Progression Indicator was developed with the use of the Fuzzy Logic algorithm, that provides an indirect but objective measurement of the patient well-being.

Although the results seem promising, more in-depth research is required such as a trial study capable of validating the results obtained.

Keywords: Quality of Life, Clinical Decision Support System, Smart Devices, Wearable Devices, Fuzzy Logic, Artificial Intelligence

Resumo

O cancro é umas das maiores causas de morte no mundo e sobreviver ao seu tratamento não significa que o processo tenha terminado. Vários pacientes que ultrapassaram o processo de tratamento permanecem inseguros em relação à sua saúde, devido ao stress e ansiedade causados pelo medo de reaparecimento do cancro e pelos efeitos do tratamento tais como: problemas de sono, cansaço e problemas de memória, dor, ansiedade e stress.

Os pacientes que terminam o tratamento são seguidos periodicamente por clínicos, que avaliam a sua Qualidade de Vida. Esta informação é essencial para compreender o seu estado de saúde, dado que o cancro tem um impacto enorme em todos os aspetos da vida do paciente. No entanto, os clínicos têm à sua disposição poucas ferramentas capazes de mensurar objetivamente a Qualidade de Vida, ou de ferramentas que possibilitem uma maior visualização de dados que proporcione uma melhor tomada de decisão.

Portanto, a solução proposta nesta dissertação é a de desenvolver um Sistema de Apoio à Decisão Clínica com ferramentas de visualização capazes de disponibilizar mais informação do paciente, obtidas com o uso de uma pulseira inteligente e uma balança inteligente. Também com o uso de Lógica Difusa, um subconjunto da Inteligência Artificial, proporcionar uma nova informação sobre o estado de saúde do paciente.

A plataforma projetada foi capaz de integrar dispositivos inteligentes de uso comercial, de forma a necessitar o mínimo de interação humana. Além disso, os dados adquiridos pelos dispositivos foram usados para criar um sistema de monitorização contínuo, associado a ferramentas de visualização de dados que proporcionam mais informação em relação ao paciente. Mais ainda, foi desenvolvido um indicador designado por Indicador de Progresso do Paciente com a utilização do algoritmo de Lógica Difusa, que providência uma forma indireta, mas objetiva de mensurar o estado de saúde do paciente.

Apesar dos resultados parecerem promissores, um estudo mais aprofundado é necessário, tal como um ensaio clínico capaz de validar os resultados obtidos.

Palavras-chave: Qualidade de Vida, Sistema de Apoio à Decisão Médica, Dispositivos Inteligentes, Pulseira Inteligente, Lógica Difusa, Inteligência Artificial

Contents

List of Figures

List of Tables

Acronyms

UI User Interfaces 18, 19, 21
URL Uniform Resource Locators 42

1

Introduction

Cancer patients and their relatives fight daily during the whole treatment process and need to deal with different new struggles due to this type of diseases. The process and the way it is managed can weaken different aspects of the patient life, derived mainly from the treatment side-effects and the diseases symptoms by itself, which will directly affect treatment quality [1]. Thus, it is important to monitor the patient well-being as accurately as possible, so clinicians can react and adjust quickly in an adequate manner, as well as to evaluate the patient health evolution during and after treatment.

The post-treatment follow-up is important, since patients feel insecure in relation to a cancer reappearance and the post-treatment symptoms related to sleep disorders, fatigue, memory problems, pain, anxiety and stress [2, 3]. So, medical efforts should be taken not only to the cancer treatment survival rate increase, but also the process that takes place after the treatment process is accomplished [4–6].

The presented dissertation is framed in a clinical environment and related to the evaluation of patient's Quality of Life (QoL) after cancer treatment, as this diseases impact several aspects of the patient's life [7]. The problem faced by clinicians on those follow-ups are mainly related to the difficulty to measure the QoL and the lack of continuous patient monitoring, that is done with periodic appointments, which gives only a small amount of information from the patient's day-to-day life. The purposed solution involves designing and implementing a prototype Clinical Decision Support System (CDSS) Platform to enhance patient monitoring, in order to make more information available for clinicians, with visualizations tools and the use of Artificial Intelligence (AI) to improve decision-making through new insights from data processing.

1.1 Motivation

CDSS can be defined as a technological tool capable of providing clinical information in medical environments, with the aim of improving decision-making and the overall health care provided by health professionals [8].

Research have proven that patient healthcare is not always optimal, since recommendations given are not complete and there are often occurrences of medical mistakes that could be prevented, sometimes due to the lack of information. Unfortunately, some of those errors may lead to a decrease of treatment quality or even to the death of patients which increases the need to find solutions to those problems. Thus, with the objective to support healthcare and improve results in clinical environment there was an increasing need to develop CDSS, capable of providing suggestions and preventive warnings [9], ultimatly improving the quality of care.

This idea increased the need to study and evaluate CDSS performance and advantages, and in fact many of those studies showed that improvements were achievable. So, to better understand which factors may enhance healthcare quality through CDSS, further research will be made in the scope of the present document.

One tool that might improve patient healthcare quality would be a system capable of monitoring patient's daily life and make the gathered information available to clinicians. This feedback could then be correlated with the clinical treatment information to generate new information to help decision-making.

A fast growing technology that could improve the referred patient monitorization are the smart wearable devices such as: smart watches and smart bands. Those wearable devices are widely commercialized, lowering costs and thus increasing availability. In 2020 the global shipments for wearable devices surpassed 400 million units and there was a clear indication that it was a growing market. Moreover, according to a Pew Research Center survey conducted on June of 2019, approximately one in five adults in the United States wear and use in a daily basis a wearable device [10, 11].

1.2 Objective

The increasing availability of wearable devices could be an opportunity to direct this technology to a clinical environment. Clinicians are daily faced with decision-making that may drastically affect their patient's life, thus performing a vital role in our society. The addition of technological tools in medical diagnostic aims to exploit device's capabilities that can help decision making, in complex scenarios.

The main objective of this dissertation is to design and implement a CDSS, designated as CDSS Platform, composed by a web application integrated with devices that together will be able to create a set of visualizations tools fed with data from a smartband and a smart scale. The conceptualized CDSS Platform will be used to enhance patient monitorization and with the use of a Fuzzy Logic algorithm give new insights about the patient's health indicators. The core focus is that those tools empower the clinician decision-making due to a more informed treatment follow-up.

The conducted research also aims to better understand some key aspects that should be taken in account to better evaluate the resulting solution, such as:

- capability to improve data visualization, so clinicians can perform more informed decisions;
- possibility to integrate commercially used wearable devices for clinical use;
- platform capacity of continues/real-time remote monitoring;
- opportunity to re-utilize past gathered data to inform about the past habits and use them as guidelines.

The purposed platform is targeted for clinical follow-up of patients that have conducted cancer treatment within the scope of the European Research Project FAITH: a Federated Artificial Intelligence solution for monitoring mental Health status after cancer treatment. The aim is to provide with new insights that will aid in better evaluation of the patient's QoL. The platform is not going to measure the patient's QoL, but rather make more information available to the clinicians so he can better understand the patient well-being using a new health indicator obtained with the fuzzy logic algorithm.

The fuzzy logic algorithm is a subset of AI capable of modulating complex problems, as the ones found in diagnosis systems due to the complexity of human biology. The use of this algorithm is also a well-known solution in clinical applications because it does not suffer from the black box problem. The black box is a problem of most AI algorithms because it is hard or almost impossible to understand the mechanism behind its resulting output [12–14].

The advantage of fuzzy logic lies on the combination of resolving complex systems by means that are easy to understand and to explain to non-technical person, such as clinicians. Another key factor that improves this algorithm performance is the possibility to be designed in cooperation with specialized teams that already know and deeply understand the problem that is wanted to be solved. So, this solution might be a powerful tool in the clinical scenario faced on this dissertation, to help on the QoL evaluation.

QoL can be defined in a simplistic way as a person's own perception of well-being from domains such as physical, psychological, and social, as described on table 1.1 [1, 15–17]. Nevertheless, measuring it might seem to be a complex process, due to its inherent subjectiveness. Different researchers tried to understand how some factors could influence the quality of life QoL from cancer patients and several self-report or physician-report forms were created to evaluate and measure different domains or the overall QoL from patients.

Table 1.1: Examples of situation that lead to a decrease on QoL due to cancer treatment. [1, 15–17]

Domain	**Areas that might be affected**
Physical	Pain due to symptoms and reduction of daily activity and function.
Psychological	Anxiety, depression, and sleep disorder.
Social	Isolation and distance from family and friends.

The problem of using those self-reported forms is that they require an exhaustive patient interaction subjective to the resembling capability of the patient, being those subject to inaccuracies. Furthermore, those forms are not used weekly or systematical enough, reducing the amount of patient's information available for the clinicians. But on the other hand, technological solution might be able to monitor data from patients continuously.

A technology that has huge potential in this field include the already widely used smartphones, wearable smart bands, and similar gadgets. The major benefit comes from, as referred, the daily and constant capability to monitor, collect, and analyse data in an easy and non-intrusive way. Those features could then be exploited to empower clinicians at providing more informed decisions [18–20] and although most conducted studies suggest bigger samples to provide stronger results, they are promising.

The approach in this study is focused on methods and tools development, with the use of a smartphone, a wearable device and a smart scale to track and monitor patient's data, to help in the clinical evaluation of QoL from patients, that might be done by giving new insights from data that was not available before. The benefits from this approach will enhance monitoring, since real-time data will be collected and more objective information will be available for medical decisions, without the need of constant medical appointments.

In order to fulfil the proposed objective, further research will be taken to better understand the state-of-the-art design principles, implementation concepts, useful tools and methods that need to be considered for the prototype platform or even future works. The research will cover topics such as:

- Clinical Decision Support Systems. Quality improvement, design methods, must have features, studied real-life impact, acceptance problems, inherent risks, obstacles, and limitations;

- Artificial Intelligence. AI algorithms used on medical applications;

- Software Design Methods. Understanding which the best software development methodology, given the needs for this research;

- Quality of Life. Factors that can help in the quality-of-life evaluation from patients.

The development and investigation of the purposed solution requires a good framework to magnify the results obtained. Therefore, to better understand which development methodology is best suited to this particular situation, the following section will start by analyzing the main possibilities available.

2

State of the Art

This chapter will overview key concepts required to design and implement the purposed CDSS Platform. The concepts further researched will enable a better understanding of what was already done in these field so that this and future works can develop a solution on top of the knowledge already available.

The concepts covered are very broad and some of them will not be applied in this dissertation, however it is important to mention them in case any future work is developed in this scope or the purposed platform is further developed. For those cases the key concepts here covered will be useful in the way they might be already summarized, thus easier to understand and be used as guidelines. Hence, working as an implementation accelerator, since future works might focus on the development and not so much on the research, being that step already covered in this chapter.

2.1 Software Development Methodologies

In the design and development of a software application it is important to follow a methodology that provides an organized planned procedure, so that all requirements are taken in account and are guaranteed to be met for users and developers. This process follows a path that ensures the quality of the resulting product.

To pursue that goal, there are several stages in the process of software development as table 2.1 summarizes, to manage and get the most out of each stage in an effective way considering the specific case of a given project. It is important to choose the right software development methodology, suited to our needs and restrictions and based on the amount of information that is available.

The software development methodology is focused at providing support to the project development leading its management, from requirements analysis to code development. There are several important points that need to be taken in consideration to develop software, some of those essential points are summarised in the following table 2.2.

Table 2.1: Main stages of software development [21].

Research	**Information gathering to formulate requirements and goals.**
Planning	Consists in organizing the development flow as well as technologies that satisfy the project needs.
Design	Project the graphical environment or interface.
Development	Code and testing environment implementation.
Testing	Error diagnostic and fixing.
Setup	Deployment of the software in real environment.
Maintenance	Monitoring software and functionality adding.

Table 2.2: Points to be taken in consideration during software development [22].

Software specifications	**All functionalities, requirements, and restriction assessment.**
Project and software implementation	Modulation of the project and further implementation.
Software Validation	Verification and control of implementation to check if the software specification where met.
Software Evolution	Making sure that the software keeps evolving to keep its usefulness.

As usual there are different types of methodologies, each suited to one's respective needs and can be separated in two different perspectives [22, 23]:

- Heavyweight focuses on high planning, detailed specifications, and long-term design. Used in development where the requirements are static and predictable.
- Lightweight, alike heavyweight methodologies it accepts change and can easily adapt to them due to its flexibility. Focuses on getting executable code and getting feedback to adapt the software to its needs.

The lack of methodology on the development of a project can lead to poor product quality which creates a massive effect in the final cost due to extension of the deadline or on corrections efforts. To eliminate this problem the next sections will briefly summarize two distinct methodologies and understand which fits best in this development.

2.1.1 Methodology Types

2.1.1.1 Traditional methodology

Known as well as heavyweight methodology, since it is planning and documentation oriented. The idea behind this methodology is based on the creation of a sequence of steps that need to be strictly followed to progress through the software development [22, 24].

The biggest disadvantage in these methodologies is due to the core idea of following each step rigidly to achieve the result. The change of a requirement in the middle of one step will, most likely, influence each other previous steps, making change a high-cost problem and making a revision on all previous steps necessary.

That is why planning, and risk management is so important on traditional methodologies, since they are the main tool to prevent future unexpected changes. These types of methodologies are as adequate as the level of rigidness, clarity, and comprehension of the requirements.

Another disadvantage of these methodologies is that the final product is only delivered after all steps are complete, therefore the client has no knowledge on the progress of development, and so there is no feedback and no idea about how the software is being made.

2.1.1.2 Agile methodology

In these methodologies the focus is oriented to the people and the interaction between them, with the goal to make successive interaction that will give incremented value to the development with the objective to produce a high-quality product, meeting the initial requirements and needs from the client [22, 24].

The main difference between these methodologies and the traditional ones is an acceptance that requirements are dynamic and therefore there is constant evolution to be included in the project. This high-level interaction gives opportunity for the clients to have continuous information about the project progress. To enforce this knowledge, Agile methodologies aims to have constant working products that although are not final, feedback is provided regularly with no extra effort from developers. There is no waste in time on big, detailed planning since agile methodology divides the development in smaller and tangible tasks thus alleviating the planning needs.

The main disadvantage of this methodology is that since there is almost no effort in the predication and analysis of long-term development there is a lack of risk management.

2.2 Clinical Decision Support Systems

The Clinical Decision Support System (CDSS) concept is the core of this dissertation, so understanding how to correctly design one is of most importance. The following sub-sections focuses on what makes a good quality CDSS, which features were considerate important on already developed systems, their resulting impact, inherent risks and obstacles found on studied systems.

2.2.1 CDSS Quality

The CDSS objective is to give recommendations in clinical environments to help for more informed decisions, based on scientific data that was previously gathered and analysed by software.

It seems obvious that CDSS would increase clinical quality, but nevertheless that increase is highly dependent on factors intrinsic to the data within the knowledge base as already covered.

To maintain a good level of recommendation it is required that the knowledge presented on the software remains always the most up to date, based on research literature and practice-based sources. The knowledge base from the software is gathered from research, scientific literature and evidence-based, which refers to the best practices based on available knowledge [25] and has the goal to give a more uniformed practical methodology [26]. The data is then matched with a given clinical situation through the software which presents a clinical recommendation.

The quality of these systems will only be as strong as the evidence-bases of the software, so the better and more robust the research available, the higher the quality of the recommendations will be.

There are several important points to give better clinical decision support and reduce mistakes which may become obstacles in the development of such software, for example [25]:

- Specify the context of the given recommendation since it can be applied only to certain special situations;
- Keep the knowledge based the most updated possible, to give recommendation that are truly corrected based on the current evidence and research;
- Give recommendation supported by different studies and perspectives.

A study based in Swedish emergency departments, tried to explore which factors affected the data quality and in what ways there could be an improvement to reduce data quality degradation. So, to better understand those real-life factors, the study will be shortly addressed but only on the most relevant topics [27].

2.2.1.1 Swedish Case Study on the emergency departments

The study approaches the use of CDSS in the medical emergency department where patients need to be submitted to a triage based on vital sign measurements so that a prioritization system can be implemented.

In this case the quality of the vital sign data will directly affect the outcome of the CDSS, since the lack of quality will lower fairness and efficient of the prioritization system. In order to enhance triage scores, the study suggests that data needed to be correct, complete, and timely available [25], as referred on table 2.3.

The three relevant results obtained from the analysis of the research can be summarized in topics as shown on table 2.4.

Table 2.3: Meaning of the three main quality factors of a vital sign data.

Correctness	How true the documented vital signs are.
Completeness	Whether all expected vital signs are registered.
Currency	Delay between the measurement and the registration of the data.

Table 2.4: Swedish relevant research results summarized.

Equipment	Equipment outputted results should never be blindly trusted. It is always vital to understand and critically evaluate the output given but a certain medical equipment, if we blindly trust the device a malfunction can create data with no use and quality, compromising the whole CDSS.
Workflow Support	The benefits from the introduction of data in the CDSS need to be noticeable and useful, to increase the willingness to use it. If there are no major advantages secured by the CDSS, then the process will not be supportive enough to be perceived as needed.
Interoperability	The CDSS software should not create problems of interoperability, since incompatibility within services may reduce the amount of valuable data that is gathered duo to difficulties in the share process. It is referred as well that there should be used the same terminology, to create cohesion between services systems.

Although the article concludes that the CDSS is still not that easy to implement effectively and efficiently having the knowledge to consider those problems will grant a better understanding on the data quality level.

2.2.2 Critical CDSS Features

The development of software for the CDSS required further understanding from the features that should be implemented to improve clinical outcome.

A study tried to evaluate those features that should be taken in consideration in the CDSS software development so that it was possible to enhance results [9]. The study included fifteen features that were then statistically analysed to check if there was any correlation between their use and better success outcome. The results showed that the overall performance of the CDSS were significantly better in 68% the cases studied with the fifteen features that were being evaluated.

Five of the fifteen features showed better performance increase in comparison to the other ten, those features should be the top requirements to be taken in consideration during software development. The study identifies as well the automated decision support as the biggest independent predictor of a higher success rate. Some features showed almost no improvement in their implementations or worst then that, showed a better performance in the absence of that feature. Does features should be taken in consideration since it suggests that they can bring more disadvantages than advantages in the clinical environment. The following tables summarize the most relevant features for the discussion of this topic.

Table 2.5: Result from the top six best features [9]. adapted.

Feature	Success rate with (%)	Success rate without (%)	Independent Predicter
Automatic clinical support	75	36	Yes
Integrated care reminders on patients charting and order entry system	73	36	No
Computer generated decision support	76	50	Yes
Reason recording for not following the CDSS	100	59	No
Provision of action recommendation	76	41	Yes
Support at time and location of the decision making	73	25	Yes

Table 2.5 shows the results from the top six best features and the success rate from the CDSS with or without the respective feature.

Table 2.6: Result from the top six less significant features [9]. Adapted.

Feature	Relevance (%)
Promotion of action rather than inaction	1
Justification via provision of reasoning	12
Recommendations executed by noting agreement	12
CDSS accompanied by periodic performance feedback	-1
CDSS accompanied by conventional education	-19
Local user involvement in development process	-30

Table 2.6 summarizes those features that showed low relevance, being the relevance calculated by the difference between success rate with the feature minus the success rate without the feature, thus showing the percentage of improvement in the implementation of the feature.

2.2.3 CDSS Alerts Acceptance

Although CDSS are great tools with high expectations of improving quality healthcare, there may exist problems related to the acceptance of the recommendations generated by the software. Since, healthcare personal can ignore the warnings and recommendations from the software, trusting more in their own experience, there should be taken some care in how the information is given [8, 28].

One research evaluated some factors, resumed on table 2.7, that could potentially affect the level of acceptance from CDSS recommendations and warnings, being them clinical or human, to better understand how to improve future CDSS designs which could, for example directly affect error prevention.

Table 2.7: Factors studied in the research.

Impact of knowledge quality	Patient age
Alert display	Dose-dependent toxicity
Textual information	Alert frequency
Prioritization	Alert level
Setting	Required acknowledgment

The research relevance is due to the high prevalence of overrides from CDSS, which main objective is to warn mistakes, since there is no sense in having a support system that is ignored by its users. The research suggests that the relevance of the warning and the way it is presented could be one major reason.

The results obtain by the study strongly suggest several points to be taken in account to better design the systems interaction with the user, being them [8, 29]:

- The alert presentation to the user was found to be a determinant factor, to have higher user alert acceptance. Thus, it is important to better understand the design aesthetics and human-computer interactions in the software development.;

- Recommendations and warning should be intrusive in the user work-flow since non-intrusive warnings were most entirely overridden. So, warnings should interrupt and require user acknowledgment via interaction with the system to greater the chances of alert acceptance;

- The text information quality from alerts is taken in account on how the user react to the information displayed. Giving low quality information would lower the chances of user response to alert, since overridden was the most likely reaction;

- Textual information containing detailed information on what measurements should be taken to respond to the alert had lower likelihood of overridden.

The others referred factors presented on the above table 2.5 were not considered relevant or there could not be found significant correlation to be considered relevant for alert acceptance.

It is still important to remember that the objective is not to influence or manipulate the user to do something that the CDSS suggests, especially in a blindly way. The main objective is to take in consideration how to make sure that the user receives the information in the best way and with the best quality possible, to truly pay attention to the suggestions and to make better decision without saturating or blindly trust the software.

This reinforces the importance of data quality to produce better quality recommendations which finally enhances the healthcare support system.

Once again, it is important to understand that overridden is not always a negative attitude, since there is always a chance of false positives, being them wrong suggestions that should not indeed be followed. However, as many studies suggested when the data is analysed there was reported a higher number of true positives, being them correct or good suggestions which should not be ignored [29].

2.2.4 CDSS Risks

To better understand the risk of a negative impact in clinical decision-making when using CDSS in patient diagnostics a study was carried by the University of Alabama [30].

This study tried to evaluate how CDSS suggestions would influence the clinician in to changing his decision from a correct to an incorrect and vice-versa, and if the position of the correct suggestion in a list of suggestions would have any correlation in the influence.

To make this study, clinicians would make their decisions based on their knowledge and then make use of the CDSS to check what suggestions they would receive based on the diagnostic. Clinicians could then keep their decision or change based on the CDSS outputted suggestions.

The results obtained from the study showed that, if the clinicians made the correct decisions from the start the software would output the correct suggestions on the top 10 list in 25% of the cases. In the opposite side, if the clinician decided wrongly at the first place, CDSS would give the correct suggestion only 7% of the cases on the top 10 suggestions.

This underlines the importance of CDSS that require human data input and the way it influences the response and its overall performance, because human perspective in this case can trick the software in going at the wrong direction.

The study suggested as well that, if the CDSS suggestion was supporting the clinician decision, which meant there was a congruence, the clinician decision was less likely to change is decision, since there was a positive feedback from the software. Nevertheless, if the clinician decision was not supported by the software, there was a higher chance of change even though the diagnose was correct.

The observations showed that when the first clinician diagnose was made incorrectly, if the software showed the correct diagnostic in the top 10 list, then most of the time the diagnosis would be corrected. It was also observed that in the previous situation, if the correct suggestions were not in the top 10 list, then no change would be made by the clinician and the decision would remain incorrect.

From this research it is important to highlight that CDSS should as much as possible extract information automatically and rely on as less as human input as possible, since its judgment could affect performance.

It is of utmost importance to increase the rate of receptiveness and make the software as much reliable as possible. At the same time, it is important that clinicians make a critical evaluation of the suggestion received from the software [30]. Some reports also showed that clinicians are less willing to change their decisions after the decisions is made, so there is better acceptance from suggestions that make little corrections, as opposed to a change of the whole decision.

2.2.5 CDSS Obstacles

Having a good prediction system is not enough to guarantee real-life application of the CDSS, because statistical results from a testing set scenario does not mean good results in the medical environment. Thus, for the acceptance of a new CDSS there should be real and reliable test that grant credibility. One common procedure is the double-blind study, and only after the evaluation of the results from this controlled study there will be a wider acceptance and believe from the system [28].

The next section will summarize some of the technologies that might be used in the medical sphere, with the objective of aiding and improving the healthcare quality. These technologies will then be analysed to better understand which case scenario they are more appropriate and useful.

2.3 Artificial Intelligence in Clinical Environment

A key aspect when developing a software or algorithm to solve a problem is the implementation of an Artificial Intelligence (AI). A simplistic definition of AI would be an algorithm capable of making a machine solve problems in a similar way as human intelligence does. So, it basically tries to replicate the human thinking process and behave, to enable problem-solving algorithms [31].

AI algorithms can be used in conjunction with large amounts of data, in order to teach the algorithm to predict results. This process is used to improve the algorithm until its predictability performance is maxed or satisfactory.

Artificial Neural Networks are a very popular subset of AI and is used in the previously presented context.

2.3.1 Artificial Neural Networks

Artificial neural networks (ANNs) are algorithms that are enabled to learn and adapt to a problem solving strategy by using the adequate sources of information. The usage of those algorithms opens the possibility promote the development of a system that can aid and support medical diagnoses and the overall medical environment.

There are already several studies that show good results from the usage of neural networks in the medical field, and many had results equal or even higher when compared with human results [28].

Thus, to achieve this kind of systems, it is required large amounts of data to build a quality knowledge base, so that the artificial neural network have examples to learn from. Nowadays, the amount of information that is currently available is increasing, since the digitalization on devices and clinical analysis used in the medical sphere is growing, making information available and usable in software.

Surpassed the lack of data available, ANNs are capable of behaviour and outcome predictions, features that could give a good use in medical environment, thus being able to produce medical suggestions based on the given inputs and previous learning process.

Investigations on the implementation and development of tools using ANNs for diseases diagnoses have been studied for long, and their findings show high rates of accuracy achieved in several fields as: cardiovascular diseases, cancer, and diabetes. One study tried to measure the quality of life from diabetes patients such as levels of satisfaction, social interaction, and depressions. The results obtained from the ANNs model were similar to the traditional statistical methods [32].

In conclusion, studies have shown the importance and effectiveness of implementing ANNs in medical environments, mainly by its capability to learn from large amounts of data in reduced amount of time and by giving extra medical feedback in diagnoses, which can be a key factor in the reduction of mistakes and misleads.

2.3.1.1 ANNs Basic Principles

ANNs principle is to replicate human brain neuron activity and its learning process in a mathematical way, making it programmable and easier to understand. The mathematical representation recreates a neuron system that links neurons separated in layers that are then linked to the next layer and previous layer, similar to a biological neuron works as compared in figure 2.1. Since each link as a weight the network will try to adjust them to adapt the network so that it outputs the target output, during the training process, in other words weights represent the learning mechanism.

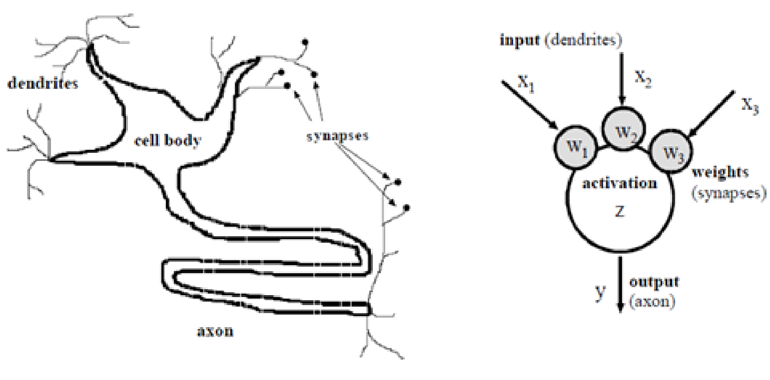


Figure 2.1: Comparison from biological and artificial neuron [28]

The learning process acquired from ANNs is in most cases not understandable, is hard to understand or does not seem to have a tangible meaning. So, those networks can be viewed as a “black box” that receive an input and gives an output based on examples that were used for the leaning process.

2.3.1.2 Major Concerns of ANNs Implementation

Implementation on medical applications have seen good results, nevertheless there are some concerns that should be kept in mind during the development steps, such as: the process of making the dataset and the performance [28].

The network learning process might create a false sense of accuracy. In some cases, the network memorizes the training set instead of learning and generalizing from the examples shown, which creates over fitting. When this problem occurs, the network will have good accuracy at predicting already seen cases, but its performance will drastically fall when confronted with new and unknown cases. This situation makes the network incapable of dealing with real-life scenarios, since in most cases there might not be same cases as from the training set. This is the main reason why testing sets should be different from the training set, so that the accuracy evaluation is as least biased as possible.

In complex environments, which need to be taken in consideration several health conditions, some results may be mistaken due to not expected presence of more than one pathology. Nevertheless, those mistakes can be avoided by being taken in consideration from the beginning of the network modelling process and by warning medical personal for these occurrences [32].

2.3.1.3 Obstacles for ANNs applications

As referred when explained the mechanisms in which ANNs learns, it was introduced the concept of "black box", which is the lack of understanding on how the network learned to generalize the examples given and adapt its weights in order to make predictions with good accuracy. Although this can be seen as not an obstacle, being able to present results and to explain how those results were acquired is extremely important and should not be despised. If the actual user does not believe or sees the system as not reliable due to his lack of understanding, they may not be willing to accept the software, since on the other hand clinicians are highly experienced professionals.

This results ultimately on vain developing work, given that there is no use of ANNs implementations, if there is no real-life use.

2.3.1.4 Variations of the ANNs

Although artificial neural networks have wide medical applications, there are other techniques used that are basically variations from the ANNs. Does techniques will be referred, as they have potential on the medical sphere.

2.3.1.5 Neural-Fuzzy ANNs

A well-known concept used to break the barrier between the human and machine understanding is the fuzzy logic, that uses human concepts and languages to define categories.

With this concept in mind, it is possible to integrate the fuzzy logic and exploit its advantage in an interface that functions as a middleman between the neural network and the user, so that data can be inputted and outputted in a way that clinicians can work and understand, as illustrated on figure 2.2.

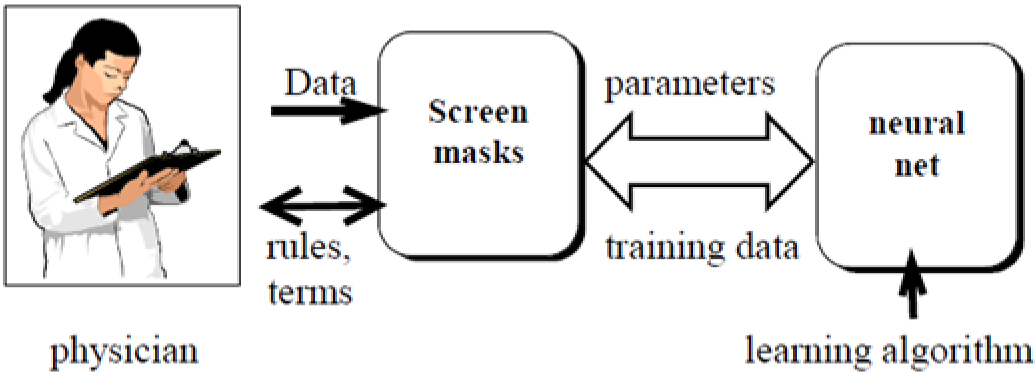


Figure 2.2: Neuro-Fuzzy interface example [28].

The addition of this concept to the medical field brings a better understanding of the prediction mechanism that is hidden in the neural network "black box", since it can provide some explanation for the network output. This insight is valuable in the real-life use and at the same time is a way of explaining how ANNs work so that acceptance is made easier, by giving some transparency and therefore enhancing levels of trust and reliability.

A study showed satisfying results of the neuro-fuzzy implementation, where results seem similar to the normal neural network but with the advantage of having some transparency and knowledge on how the network worked to make the outputted prediction [28].

Although prediction networks are not always able to give the correct diagnoses, the application of neuro-fuzzy gives better insight on how the given output was chosen. Thus, giving some tangible warning signs to clinicians even though they do not agree to the suggested output, if they see some value that is too high or too low, for example. Therefore, it is important to have an interface with visible and understandable data available for the clinicians use, without this transparency when the network gives a mistaken output there is no additional warning that the clinician can rely on.

2.3.2 Fuzzy Logic

The Neural-Fuzzy algorithm is a powerful tool, by adding fuzzy logic to the known neural network algorithm. Nevertheless, in scenarios where a large dataset is not available to teach the neural network, this solution cannot be applied.

Fuzzy logic is a form of artificial intelligence used to imitate the human reasoning process, by providing flexible solutions, in contrast with the boolean logic in which an output is strictly 1 or 0. Hence, this algorithm is considered an AI subset [33–35].

This algorithm is very well suited in problems that are hard to modulate and is applied in several systems such as in medical diagnosis systems and decision systems that have higher levels of uncertainty. The application on medical environments is supported due to the complexity of human biology, making this problem hard to module, thus the fuzzy logic is useful to extract meaningful data through imprecise data [36].

In contrast with several other AI algorithms, such as neural networks, fuzzy logic is easy and understandable even for non technical people. Furthermore, the fuzzy logic models does not require a dataset, since it can be design and implemented in cooperation with an specialist in the field. For instances, if the algorithm were to be used in a medical diagnosis system, medical personal could use their knowledge to help in the development of the algorithm [36]. The algorithm can also be modeled with a dataset, in cases which no knowledge about the process is known. The data extraction is made by clustering related information, and then fine tuning the algorithm.

2.3.2.1 Clustering Method

Clustering is a known subset of AI, called Machine Learning (ML), capable of aggregating similar/related data [37]. This feature creates a opportunity to model and visualise patterns in classification algorithms.

The application of this method in fuzzy logic resides on finding relations and patterns of input and output, to create the knowledge base required for the algorithm [38].

2.4 Visualization of Medical Data

To ease and optimize the clinician's understanding of the visualization tool, the following section will make a simple summary of some of the basic graphical representation and visualization tools that must be taken in account to better transmit the outcome from the processed data.

2.4.1 Graphical Representation

Transmitting information correctly and in a time-effective way has a key role at improving clinician's comprehension and quicker risk visualization. The lack of data representation quality my lead to misunderstood and mistakes, due to the false perception of information.

Studies showed that the way data is presented in clinical environments affect the diagnostic quality, which highlights the importance of visual representations to improve communication of medical data [39, 40].

Several methods have been applied to better communicate medical information digitally such as graphs, tables, and variants summarized on table 2.8, since those represent numerical data in an accessible and efficient manner. To visualize textual information in a less time-consuming way, especially long texts, information can be presented with icons or images. This is enforced by the human natural capability to resemble and memorize information in the visual form, although in real-life situation this type of visual representation may require a learning process, which may become an obstacle to its applications. An easy way of smoothing the learning phase is to create an interface that is capable to giving the learning experience during the process of usage, for example, an icon could have a pop-up alternative legend that is displayed when required further detail.

Medical information can be represented in different types of data and depending on the data type that is wanted to present, one graphical representation will be more suited. Table 9 will organize the information, summarizing the different ways of visual representation that could be used and when to used them effectively [41].

The knowledge of the referred methods of visual presentations may enhances the quality from the initial design draft, nevertheless the rate of success is strongly dependent to the final user perception and comprehension. Thus, the most important step during the design of the UI is to receive feedback from as much users as possible, so that the design can be improved and perfected.

2.4.1.1 Graphical Representation Basic Design Principles

The quality of the visual information methods used in the UI from a CDSS directly influences the software quality, since data observation and analysis are directly dependent to the way data is presented. A good graphical representation could improve time efficiency at observing important values or even reduce mistakes from misinterpretations. So, a good visual representation of data should be simple to interpret and understand [42, 43]. Those design principals are summarized on table 2.9, where it is referred five basic principals to have in mind followed by examples.

Overplotting. Multiple line plot and scatterplot are graphs severely affected due to large amounts of data presented at the same plot, which leads to overplotting, meaning that data will be densely presented, and overlaps will occur, making comprehension hard or not feasible. Table 2.10 summarizes 2 alternatives to overcome overplotting problems.

2.4.2 Data Visualization Tools

Designing data representation is much easier with nowadays visualization tools, which may increase data quality and visual analysis, at the same time featuring interactivity such as zooming and change of visual angles which could increase detail comprehension.

The table 2.11 summarizes some of those tools from a study that tried to evaluate and compare the main advantages, and drawbacks.

Table 2.8: Types of visual data representation in the medical environment.

Type	Description
Simple Lists	Capable of presenting information in a simple way. The main difficulty is to present the needed information in the simplest and shortest form, so that it does not become time-consuming. These lists are used mainly on menus that might have clickable functions on UI.
Nested Lists	More complex, where each topic from the list as a sub list. The main use of nested lists in UI is to hide elements from a determined topic, so it reduces visual noise but still is accessible when needed.
Table	Commonly used to present numerical values from vital signs, for example, although they are not effective at showing increase and decrease patterns. Nevertheless, they can be multidimensional and still be easily readable. On UI, tables can be enhanced by adding additional features such as clickable values to check for additional information and pop-ups or have colour schemes, red and green for example, that easily show abnormal values or those that become normal.
Graphs	Useful for pattern visualization.
Simple Charts	Used to aggregate data visually with points or lines through their axes of n-dimensions. They are very useful at showing variations on the observed values. The major drawback is the visual confusion created by several series in the same chart.
Configural Charts	Charts can be created with a specific shape or format, to help data analysis, such as polar-polygon plot, which try to highlight one specific feature from the rest, so it becomes easier to spot values out of normal boundaries.
Graph Notation	Features the possibility to show relationships from the medical information that is presented and in an organized form. It can be useful to justify decisions from algorithm results.
Icons	This visual representation has the advantage to be easily remember and resembled.
Atomic Icons	Each atomic icon has its own meaning and represent the most basic form of icons. Those are very useful to illustrate UI buttons, clickable functionalities, and to draw attention to some information. Icons can also be used with other types of visual representations to enhance a more intuitive comprehension.
Iconic Languages	Those icons are utilized as a substitute to textual information as a "visual language", the same way a danger sign would be automatically recognized as warning for a danger situation or as traffic signs. The "visual language" as advantage as it is easy to learn, remember and at the same time its information is quickly observed. Nevertheless, there is always the possibility of misleading due to icon ambiguity, which must be taken in consideration.

Table 2.9: Design basic principles [43].

Principal	Example
Graphical design should be focused on the principal feature that is wanted to highlight and at the same time show as much detail as possible.	Box plots can be substituted by violin plots to add density information, without any drawback. Although it's important to remember that detail presentation that is not useful is also not efficient.
Represent data with visual efficiency.	Dot plots is a good alternative for bar plots since the valuable information is not in the height of the bar but rather in the position of the top end. Furthermore, it makes easier to visualise confidence intervals.
Graphics that are usually compared should be as close as possible if they cannot be placed in the same graph.	Time-to-event visualization is a good approach to make comparison easier, since it shows how values changed in a chronological way. Time perception should be easily visible, so that is observable the time lapse between each measured value with the correct proportions.
Use plots as simpler as possible, so it makes the least "visual pollution".	A simpler representation of bar plots is the dot plot which makes less visual impact. The colour scheme should be taken in account as well.
Avoid visual effects that may trick the user to misinterpretation.	Pie charts are a good example that shape, and colour schemes may mislead visual interpretation, since some times it is hard to see which is the bigger or smallest area. Presenting a result/total-value bar, on a bar plot with more bars, in a more privileged separate positions removes the possible misunderstood that it may represent the same as other bars.

Table 2.10: Overplotting alternatives.

Type	Alternative
Multiple line plot	Lasagna plot are easier to read and analyse. The major disadvantages are information loss and colour schema may not be the best way to represent numerical values.
Scatterplot	Sunflower density plot or the addition of heatmaps. Both alternatives are used to show which places show more density, thus recovering some of the information lost due to overlap.

Table 2.11: Visualization tools summary [44].

Tool	Advantages	Disadvantages
Tableau	Interactive. Custom. Several visual options. Fast and flexible. Several data formats and server support. Intuitive UI. R models import capability.	Might require good programming skills based on the intended out-come. Not free.
Microsoft Power BU	Interactive. Flexible. +60 types of source integration. Easy to start. Queries are made in a natural language. No programming skills required. R usage is available.	Slow compared to Tableau. Limited free features.
Plotly (uses python and Django framework.)	Free features, although limited. Variety of charts available. WPD feature which automatically reads data from images. Privacy features. Python, R, Matlab and Julia APIs are available.	Limited free features. Programming skills might be required.
Gephi	Open-Source. Capable of handling large and complex data. Dynamic data exploration. No programming skills are required. Performance enhances.	Graphs understanding is required. Only for graph visualization is available, which is a major limitation.
Excel	Query capability to connect to most services. Semi-Structured data.	API only for Office 365. Not free.

Comparing the advantages and disadvantages made in table 2.11, showed that Tableau, Power BI and Excel were more flexible, feature wise. Nevertheless, further research must be made to better understand which is the most suited tool for the use on the CDSS specific scenario.

Having the referred concept covered, the next chapter will summarize reviews from some relevant studies that were taken to develop CDSS with the help of wearable devices to monitor and predict medical outcomes. Those reviews are important not only to acknowledge some of the challenge that those researches faced, but also to better understand which technical designs were already though. Although those solutions are not directly related to the purposed on this dissertation they create good foundations as they use similar concepts.

2.5 Review of Related Studies

There is a vast number of studies that explored CDSS based on daily monitoring of patient's activities and biological signs. This review will understand some of the case scenarios studied on medical research from different non-cancer related diseases and what methods were used, afterwards cancer related studies will be analysed. But firstly, some research will be made to understand key aspects that should be considered, regarding the factors that affect the Quality of Life from cancer patients and survivors.

2.5.1 Factors that affect Quality of Life

A wide variety of studies tried to understand which factors were directly correlated with the increase or decrease of Quality of Life (QoL). Those research are mainly focused on cancer patients undergoing treatment, nevertheless a wide range of research is starting to emerge on cancer survivors as well.

A research published on Plos One Journal [1], conducted on 140 breast cancer survivors concluded that physical and psychological symptoms had significant effect on the Quality of Life, being the later the most significant. It was also found that, anxiety and depression are significantly correlated with poorer Quality of Life. Those symptoms occur mainly due to the fear of the cancer reappearance. In contrast, physical symptoms such as fatigue and sleep disorder may cause negative effects on the patient's psychology [45].

2.5.1.1 Heart Rate

A research article from the European Journal of the Heart Failure [2], verified if there was a correlation between the resting heart rate and the survivability of patients in cancer treatments. The study concluded that patients with a higher resting heart rate had indeed higher levels of mortality, with a special incidence in lung and gastrointestinal cancers. It is also stated that several other studies, have observed cardiovascular and non-cardiovascular mortality rates in patients that showed higher levels of resting heart rate. Also, a study that analyzed and followed 300 patient that undergone cancer treatment, for colorectal adenoma, observed that patients with higher resting heart rate had a higher recurrence of cancer then those with lower rates [46].

Another study published on the European Journal of Heart Failure, enrolled with 145 cancer patients and 59 control patients, showed results that the mean resting heart rate from the healthy control group was 70 Beats per Minute (BPM) and 79 BPM for the cancer patient group. The article also correlated a resting heart rate lower then 75 BPM with a higher survival of cancer patients [47].

[1] https://journals.plos.org/plosone/

[2] https://www.escardio.org/Journals/ESC-Journal-Family/European-Journal-of-Heart-Failure

2.5.1.2 Weight Variation

The researches, suggest that weight loss is correlated with worst outcomes on cancer that undergone treatment, nevertheless is warns that more evidence is required about intentional weight loss [48, 49]. Also, another study published on Cancer Epidemiol Biomarkers Journal, verified similarly on breast cancer patients, that weight maintaining was correlated with a lower death risk. In contrast, weight loss was associated with an increase of death rate and weight gain was associated with no increase [50].

2.5.1.3 Step Count

Research published on Health and Quality of Life Outcomes Journal [3], verified among cancer survivors, if physical activity (including walking) would affect their Quality ff Life. The study group was composed by 13 adult cancer survivors. The results observed support the importance of physical activity in the improvement of Quality of Life, since patients felt improvements on physical and mental health [51].

Several studies suggest that step counts are correlated with cancer patient's Quality of Life. A research article published on Integrative Cancer Therapies [4], observed 39 patients with an advanced stage of lunge cancer an concluded that a higher average daily step count was associated with a higher patient Quality of Life [52]. Another article published on Digital Medicine Journal [5], observed on 37 subjects with different cancer types, that an increase of 1000 steps/day was correlated with the reduction of negative events such as: hospitalization or death. It also analyzed that patients with a daily step count higher then 2000 steps shown much higher survivability rates [53].

2.5.1.4 Sleep Quality

Research published on Cancer Management and Research Journal [6], with a control group total of 124 cancer survivors and 48 cancer-free individuals, tried to verify the relation of each group's sleep quality. Cancer survivors were significantly less active and had worse sleep quality in comparison with the control individuals. Quality of Life was also found to be lower in cancer survivors. In terms of physical activity, it was found a significant positive correlation with QoL and mental health. Also, quality of sleep had a strong correlation with physical and mental health [54].

Another study published in the Lung Cancer Journal [7], composed by 76 lung cancer survivors and 78 cancer-free control group, showed that cancer survivors had lower sleep quality, in contrast with the control group. Also, 49,2% of cancer survivors who did not have sleep disorders, developed them after diagnosis. Cancer survivors also showed a

[3] https://hqlo.biomedcentral.com/
[4] https://journals.sagepub.com/home/ict
[5] https://www.nature.com/npjdigitalmed/
[6] https://www.dovepress.com/cancer-management-and-research-journal
[7] https://www.lungcancerjournal.info/

lower sleep efficiency in relation to the control group, due to increased awakening during the night. The results also verified a significant relation between sleep quality and Quality of Life [55]. In terms of sleep duration a study published in Sleep Journal [8], composed by a total of 104.010, correlated sleep duration with mortality risk, of all kinds. The results obtain shown that, subjects who slept less than 4 hours had an elevated relative risk in relation to those who slept 7 hours. The analysis confirmed a significant correlation between a shorter sleep duration and risk of mortality. In relation to longer sleep duration, there was also an increasing risk of mortality with the increase of slept hours, for more than 8 hours. As a comparison, 10 hours of sleep had a higher relative risk value then sleeping less then 4 hours. So, longer sleep duration was also found to be a risk factor to mortality. Nevertheless, it is referred that more study is required to understand the relation of longer sleep duration and all-cause mortality. Finally, subjects that slept 7 hours shown less relative risk of mortality [56]. Another study published on the Sleep Journal, shown that a decrease and increase of sleep duration increased mortality ratio. Nevertheless, the decrease was related to cardiovascular mortality and the increase with non-cardiovascular mortality [57].

2.5.2 Clinical Use of Smart Devices

2.5.2.1 Tremor Evaluation

The study [58] objective was to evaluate the involuntary movement from an arm or body part in a oscillatory form, designated as a tremor. Tremors can be essential, which manifest as non-diseases related, or due to Parkinson, multiple sclerosis, or cerebellar damage.

The need for this study came by the limitation from available equipment and techniques in the clinical environment due to the lack of flexibility, accuracy, and cost.

The alternative studied was the use of a smartphone with a seismograph application. The application used the built-In accelerometer to evaluate the movements on the 3-dimensional axes, which was displayed graphically. In addition to the use of the software a control reading with Electromyogram (EMG) was recorded.

The results obtain with the smartphone software matched with the EMG recordings, which means that both had similar tremor readings with a marginal peak difference between 0 to 0,2 Hz.

2.5.2.2 Rehabilitation Remote Monitoring

In this study [59] the objective was to evaluate the feasibility of a cardiac rehabilitation made remotely and with proper monitoring. The need came due to high costs of cardiac rehabilitation, accessibility limitation, and low participation ratings that put in danger cardiac related prevention.

[8] https://academic.oup.com/sleep

For this study six patients completed an average of three walking sessions per week through 6 weeks. During each session a smartphone was required and additional GPS receiver was used to record travelled distance, speed, and a small ECG (Electrocardiogram) device to monitor heart rate. Results obtained showed improvements in distance travelled, cardiac depression and overall physical component.

A similar study [60] focused on pulmonary rehabilitation and was made using a mobile phone, software ,and a pulse oximeter. Heart rate and blood oxygen saturation was continuously recorded during rehabilitation exercises and warning signs would be showed on display and sound would be emitted if the patients hear rate exceeded normal values.

2.5.3 Cancer Related Studies

A ranging variety of studies show QoL evaluation from cancer patients in undergoing treatment through patients self-reported questionnaires, that although may show significant accuracy is not an optimal method, since it requires the burden of daily/weekly manual report. The optimal method would be an automated and independent methods of data acquisition that would measure QoL from patients with the use of wearable and/or smartphone devices.

The next reviews will refer some studies that reflect those efforts on measuring QoL with the use of different methods and approaches.

2.5.3.1 Overall Smartphone and Sensors use in Cancer Research Study

This study [61] although not a particular case scenario, reviewed information about various studies and the use of smartphone or wearable devices in cancer research.

In this study it is referred the effectiveness and feasibility from the usage of wearable and smartphone devices on daily physical activity as a powerful and with significant precision on predicting treatment outcomes. At the same time, those measurements are objective assessment of patients QoL.

Several reported studies in the article showed significant correlations between step counts and self-reported symptoms, physical health, functioning, QoL, depression, hospitalization risk and survivability. Another two studies showed a positive improvement on daily activity, symptoms and QoL due to the usage of wearable devices that had the objective to promote activity. Different pilot studies using different sensors such as smartphone screen time, location, social activities, time away from home, resting heart rate, decreased heart rate variability, increased step speed and self-reporting pain or distress were able to correlate with factors like symptoms detection, mood and anxiety improvements, depression and rate of emergency visits.

A wide range of challenges were also addressed related to huge amounts of data storage, formatting and processing from the smartphone and wearable devices. Device selection was another problem since different hardware specification and cost may influence the software and data quality. The choice of methods to handle with missing data or data loss is another important scenario to have in account, since data flow may encounter technical problems. Finally, the importance of data presentation is also referred as an important tool to display complex information in an accessible an understandable way.

Overall, this article as very good insights from different studies that were able to obtain very significant results on monitoring and prediction of medical outcomes.

2.5.3.2 QoL Assessment with Sensors Monitoring

The following studies were able to measure and evaluate data in order to predict clinical outcomes and measure QoL from cancer patients and show a good insight on how they performed those researches.

The study [62] focus was on the reduction of continuous burden caused on patients that would use monitoring methods that required self-reporting assessments. The objective was to design a monitoring method that would require the least patient's intervention, with the use of their own smartphone that would gather information of movement, location, calls, conversation, and amount of data used. These data would be then correlated with QoL aspects such as physical, psychological, and social.

The results obtain on this study show a good potential on these methods of monitoring, although the study was very limited and assessed by a total of five university volunteers with no cancer related problems, so a clinical study is required on cancer treatment patients to understand the effectiveness and significance of the usage of this monitoring method.

The next study [53] similarity to the previous, focuses on the problem raised from the time spent from patients away from the clinical environment. The objective was to analyse the correlation between data gathered from a wearable device and performance status. The need comes from under treatment resulted from the underestimation of patient's physical capabilities, which is directly related to low medical results and QoL. Additionally, there was an effort to associate the data gathered from the device and survival rates, adverse events, reported pain, fatigue or physical functioning. This study showed a more robust sample group constituted by thirty-seven patients with advanced cancer.

Results showed significant statistical correlations with the measured values (average travelled distance, stairs climbed, resting heart rate values) and the patients reported performance status, with the biggest correlation found with average daily steps count. There was no significant correlation with values measuring sleep duration. Also, an increase of steps per day by 1000 was correlated with a significant lower chance of hospitalization, reduced high grade adverse occurrences and an increase on survivability rates.

Those promising results highlight patients physical functioning correlated with average steps, fatigue correlated with steps count, travelled distance and stairs climbed, and patients-reported depression correlation with stairs climbed. Once again, patients step count was observed as a major reliable metric for patients QoL assessment and performance status.

The article also refers other valuable studies about the feasibility and QoL improvement that showed significant improvements on patients' risk of hospitalization associated with average daily step count.

2.6 Summary

The overall research presented in this section showed the relevance and importance of this thesis, supported by several different articles that studied the feasibility and effectiveness of smartphone and wearable devices use on CDSS.

The knowledge gathered about state-of-the-art adapted techniques and methods in combination with the current evidence from field studies, show a huge opportunity in the developing and design of a CDSS capable of providing good results.

The findings observed on this research have great implication on the direction that should be taken and insights of the design, implementation, and overall challenges that might be encountered.

Although much more in depth research and article assessment is obviously required, the document show a good starting point for this and future developments in this specific medical field.

As a starter the use of AI would be able to produce good results, but requires huge amounts of data already stored and classified, so those could be processed and used to teach an Artificial Neural Network. Since, that data is not available another algorithm of data processing must be used. The Fuzzy Logic proved to be a good alternative since it is capable to process data from other sources besides from a dataset.

The research also proved that the use of smart devices were a good option in what regards to continues monitoring and improvement of remote treatment quality. Also, that this kind of methodology was already tested in real use-cases and resulting in improvements.

The chapter also analysed that CDSS are a well-known solution to several clinical problems and various articles cover important points that should be taken in consideration to design those systems with higher quality. Those articles support the purposed solution, as it goes towards some aspects covered of clinical use.

Finally, it is reasonable to conclude that there is a good amount of information in what regards to the clinical correlation of health indicators and quality of life indicators which are able of supporting the purposed Fuzzy Logic algorithm.

To sum up, this section showed from different technical field perspectives, the challenge that this thesis might encounter. Additionally, presents big efforts from other authors that are trying to innovate in CDSS field, which gives a good reassurance that much more information is available to support and lead the development and success of this dissertation.

3

Architecture

This chapter will cover the overall though process made while designing the purposed Clinical Decision Support System (CDSS), having in consideration its requirements, since it was though to be used mainly as a remote patient monitorization and during appointment, that requires minimal patient intervention.

3.1 Methodology Analysis

First and foremost, a methodology must be chosen so an analysis will be made having in consideration the data acquired on section 2.1.

Traditional methodologies are mostly planning orientated and should be applied in projects with more stable and static requirements where future updates to the project can be easily predicted and executed.

Since the nature of this project's requirements are mostly dynamic, constant changes will be part of the development due to the unknown elements inherent to the lack of knowledge on the matter. So, it is of maximum importance to choose one methodology in which change, and not static requirements, are acceptable and expected.

Agile methodologies are the alternative to the traditional ones. It takes planning and documentation as secondary, focusing on the individual, the interaction, and collaboration as well as on the code development for executable and tangible software, making the acceptance of the constant changing as part of the development process.

To sum up, traditional methodologies create a more rigid development process making change an unhappy and costly effect in the development. Since this project is as previously referred inherently dynamic and mutable, it is wise to choose the agile methodology, since it is more responsive to change, and it benefits from high levels of collaboration during the development. At the same time, it provides executable and visible software available, so it is easier to get feedback which in turns enhance the probability of making a good final product that takes in account the final user expectations [21, 22, 24].

The definition of a software development methodology enables the next phase, in which architecture design will be conducted to understand the approach that better fits the problem faced. But before the design is tackled, it is important to conceptualize the problem to perceive which is the best path to take.

3.2 Conceptualization

The purposed platform designated as CDSS Platform, aims to be used as a tool during follow-up appointments between the clinician and its patients that undergone cancer treatment, so it can aid with better data visualization and ultimately improve the clinicians decision-making.

This case scenario would be the development of a web application, so clinicians could access during appointments or remote follow-up routine. On the other side, at the patient spectrum a wearable device and a smart device will be used to gather data during his day-to-day life. This device should be borrowed during the extend of the treatment in the case that the patient does not have one of its own already. The biggest advantage of the situation where the patient already owns a compatible wearable device is due to historical data already gathered prior to treatment, since that information could be used as control data for comparison.

Data gathering from wearable devices is done by a mobile application running on the patient's smartphone, exactly the same way a normal wearable device user would use. To guaranty more compatibility between wearable devices, Google Fit mobile application from Google will be used. This will eliminate the burden of developing a mobile application from scratch, which is not the focus of this dissertation, and at the same time give good reliable service and compatibility between wearable devices models and brands. This third-party mobile application will allow to focus all effort on the CDSS web application development. The CDSS Platform basic concepts are represented on figure 3.1 which shows in a simplistic way the whole platform functionality:

1. The patient has the wearable device which is continuously gathering data, and may use the smart scale.
2. Both devices communicate with the patient's smartphone, via the wearable device own mobile application.
3. Google Fit application receives the data from the wearable device application and sends it to Google Server for storage.
4. The web application will then request the patient's data to Google Server.
5. The data is then processed and displayed for clinician use on the web application.

So in summary, the CDSS Platform is composed by a web application and a third-party application. This integration is easily accomplished with a Representational State Transfer (REST) Application Programming Interface (API), that is made available by Google. REST API is an application programming interface that allows communication

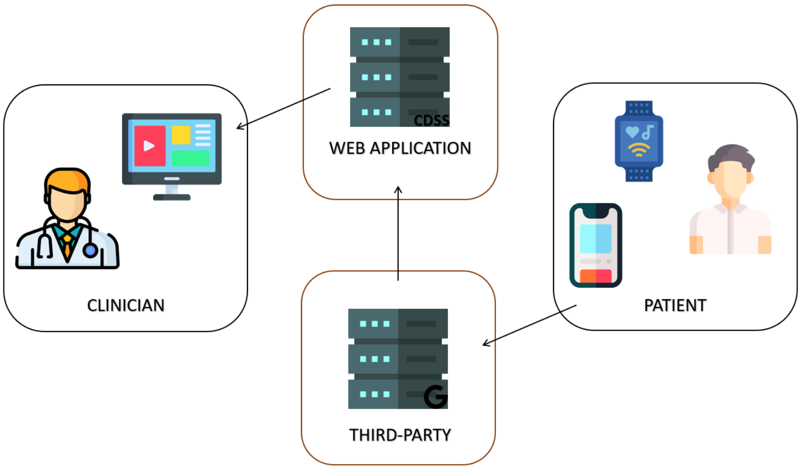


Figure 3.1: CDSS Platform Concept (adapted) [63].

with web services. This interface will be the bridge between the purposed web application and Google Fit.

3.3 Platform Features and Requirements

Finally, to conclude the conceptualization process some basic development requirements should be guarantied. This will ensure that the design is made in a manner that is inline with the state-of-the-art CDSS Quality guidelines and also to guaranty Critical CDSS Features, as presented on sections 2.2.1 and 2.2.2, respectively. Those key features and requirements are:

- Workflow Support. The platform must bring visible and valuable advantages to the clinician;
- Interoperability. It must be easy to integrate the required devices and applications;
- Automatic Clinical Support. The support should be autonomous after setup procedures;
- Integrated care reminders on tools, such as e-mail notification in case of emergency;
- The developed tools should be ready to support at time and location of the decision making;
- Quality data should be presented in a easy and understandable way;

Having the key design concepts been planned and revised, the following section will explain each architecture component of this platform.

3.4 Smart Devices

Data gathering and monitorization is now more accessible then ever, due to emerging technologies known as smart devices capable to communicate and share data continuously [64].

The smart devices integrated on the purposed platform are the Xiaomi Miband 5, a wearable device capable of monitoring a wide range of inputs trough its sensors and the Xiaomi Mi Body Composition Scale 2, a smart scale that can measure the body weight and several other body composition statistics. Both use Mi Fit mobile application to share and display data to the end user [65, 66].

The Xiaomi wearable device as an heart rate sensor monitoring data 24 hours a day, is able to monitor sleep data and naps, counting steps throughout the day and can also record activity statistical data such as duration, intensity and calories burned.

Data from those devices is send via Bluetooth to the smartphone using the Mi Fit app, that is responsible for receiving the data and then send it for storage to the Xiaomi server.

3.5 Google Fit

The Xiaomi Mi Fit application does not provide an API, disabling an access to the information stored in the Xiaomi server related to the wearable device. To surpass this problem, the Google Fit mobile application can be used, in order, to integrate the data from the Xiaomi devices. So, the Google Fit application works as a bridge between both Xiaomi smart devices and at the same time grant compatibility with others smart devices.

The Google application is very similar to the Xiaomi application as both show graphically the data monitored through the devices. Also the Google's application does step count and distance monitoring on its own, not requiring a wearable device, possible by using the built-in sensors of the smartphone. When using a wearable device, Google's own algorithm merges the smartphone data with the wearable device's data to get a more precise and complete output [67, 68].

The API support, enables the required data exchange between the platform and the smart devices. The only requirements imposed by this service is that naturally, a google account is need for development and also to the patient who is using the platform. Those requirements are due to the use of Google API Console to implement the API and because the Google Fit application requires a google account. A rather straight forward guide is available on the google website `https://developers.google.com/fit`.

These referred third-party components are essential, since their integration will allow data flow to the platform. The next section will address the web application architecture required to leverage the platform performance and user-experience.

3.6 Web Application

The core of the CDSS Platform will be the web application, and is basically divided in front-end, back-end, and database, as shown on figure 3.2. Each of those modules have their own responsibility:

- The front-end guaranties data visualization to the clinician and user-experience;
- The back-end side guaranties integration with the Google Fit Services, data exchange inside the web application and platform integrity;
- The database guaranties data secure storage to all CDSS Platform data and information availability;

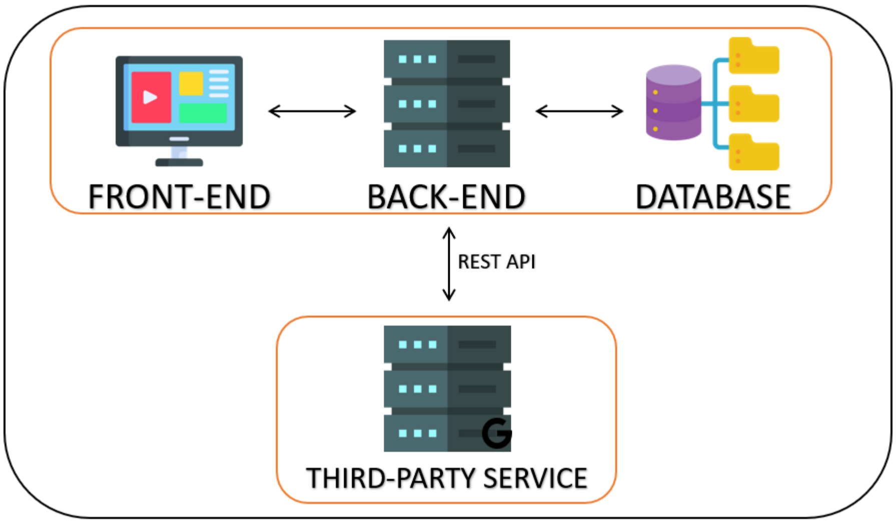


Figure 3.2: CDSS Platform Overview (adapted). [63]

Nowadays, several open-sourced technologies/tools are available to be used on the refereed modules, and each of them have its pros and cons. The following sub-topics will summarize the though process used to choose each module for this purposed architecture.

3.6.1 Front-End

The language chosen for the front-end implementation was HTML, due to its simplicity and ease to use. React could be a good modern alternative but more complex and with a higher learning curve, and since the front-end design and implementation was simple and without several components, HTML was the preferred.

To enhance the HTML styling and responsiveness without investing too much time, Bootstrap was introduced. Bootstrap is an open source development framework that works similarly to a library, so it reduces burden of customizing web styling and components usage. Templates and layouts were also used with Embedded JavaScript Templating

(EJS) and Express EJS Layout packages which saves a single or multiple template layouts and eliminates the need of rewriting code that stays static from page to page.

To better implement charts and graphical tools a javascript library chart.js is available, since it is open-source and it is very simple and flexible to use. An alternative to this library could be Tableau, since it could easily integrate with HTML, nevertheless this tool is costly hence not the best fit in this implementation.

3.6.2 Back-End

The back-end guarantees integration, functionalities and all the requirements needed to run the server-side and make the platform usable. Node.js was the chosen technology to run the web application server-side.

Node.js is an open source popular JavaScript environment used for API services such as in this situation, where the client-side will make requests in real-time to the server-side in order to get data access that is, for example, patient related.

This technology brings scalability, performance to the server-side and JavaScript development, which is very useful when developing a web application alone or without separated developing teams for front-end and back-end. Node.js has also a variety of packages available and accessible through its package manager NPM, which simplifies many of the tasks required. The following table 3.1 briefly describe the most important packages used and their functionality.

Table 3.1: Node.js main packages used brief description. [69]

@hapi/joi	Data validation with simple schema definition, useful to verify user input [70].
Mongoose	Used to manipulate MongoDB objects in Node.js [71].
Bcrypt	Hashing library used to encrypt data and user's password on the database [72].
Jsonwebtoken	JSON Web Tokens library, used for platform integrity [73].
Extract-zip	Extract zip file from the Google Takeout [74].
Multer	File upload management, for patient's manual data upload [75].
Axios	Allow HTTP requests, promises with API and parses JSON data [76].
Cookie-Parser	Enables cookie header usage [77].
Dotenv	Enables environments variables to store secret keys and information [78].
Googleapis	Enables Google APIs support, to request data from Google Fit [79].
JSON-file-loader	Reads JSON files and stores it in a JSON object, used to manage patient's data stored[80].
Node-cron	Schedule process to be executed from time to time, such as continuous patient's data update [81].
Mkdirp	Enables directory management, for data storage [82].
Nodemailer	Provides easy e-mail sending, used to invite new users to the platform and warning the clinician about patients in risk [83].

There are several other popular technologies available for backend development such as: laravel and django [84–86]. Laravel is a PHP framework very used and capable, that offers security and flexibility, since it is a fullstack open-sourced framework. Nevertheless, PHP does not have a library manager similar to npm from Node.js which is handy, as shown on table 3.1.

Django is based on pyhton and it is a good option when developing complex projects, since it has a structured approach. Yet, this framework is not compatible with real-time web applications and requires prior experience, since it has a higher learning curve in comparison with Node.js that uses javascript. Node also allows an easier full-stack development as already mentioned, and Django would require to develop the application in two different languages, which is not so convenient.

3.6.3 Database

The CDSS Platform data is mainly related to users basic information such as: e-mail, password, name, etc... and in the case of that user being a patient it also has the data related to the smart devices. So, with such a simple group of data and the unknown possibilities of storage structure for future developments, the most suited database would be MongoDB.

MongoDB is a non-relational database, so it does not use the tabular schema as mySQL does, in the other hand it uses a JSON-like format designated as documents that are stored inside collections that conceptually are similar to a table in mySQL, as depicted on figure 3.3, which works perfectly with the web application as JSON is commonly used. MongoDB is particularly suited for agile methodology and situations where the development might require continuous modifications to the database schema, and scenarios of uncertainty. In contrast mySQL is a relational database that requires a well sorted and though structure, since changes might require an whole structure and query rearrange. Therefore MongoDB is the most flexible and scalable option for this type of platform, since the general data structure might vary throughout the research and development process.

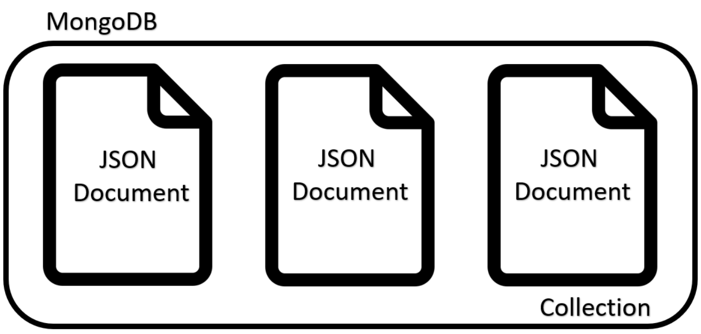


Figure 3.3: MongoDB model example.

To store data related to both smart devices a different storage method should be selected, one that takes in consideration performance and load balance. Since this data might take big proportions and is required to be loaded quickly, a JSON file storage will be used. Although a more powerful solution could be used such as MinIO, which is a cloud storage solution capable of scaling the platform much easier, for this platform a simpler local file storage will be the best solution in terms of cost/benefit and performance.

To sum up, Figure 3.4 summarizes the CDSS Platform technology choices made for the development of the purposed CDSS.

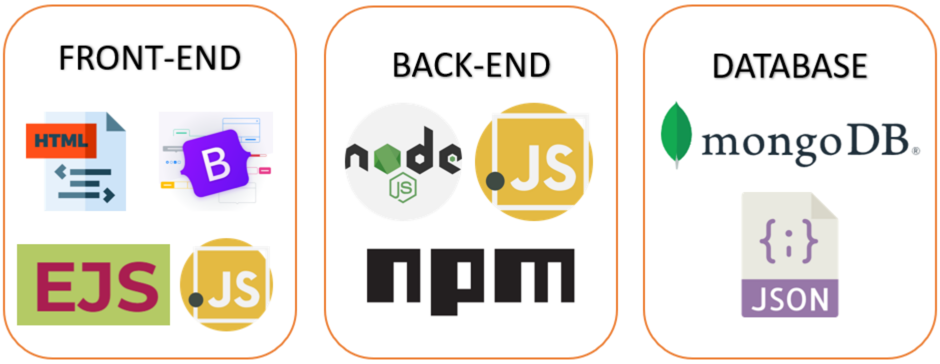


Figure 3.4: CDSS Platform Technologies (adapted). [63, 69, 87–89]

3.7 Platform Flow

The flow of the CDSS Platform will vary depending on the type of user that is using it. In the clinician's scenario the user interaction would be similar to the exemplified on figure 3.5.

The clinician would sign in on the web application and view his patients listed by name, by choosing one, he would access the patient's dashboard. On the dashboard the clinician would have the possibility to visualize data from last 2 months, data aggregated by months from the beginning of the treatment, or daily data from the beginning of the treatment.

The clinician could also add a new patient by sending an e-mail invitation to join the CDSS Platform. The patient would then need to check his e-mail and click on the link received, and by doing so the patient would be redirected to the web application register page.

This steps ensures that the user-patient has his own profile, so after signing in, the patient would be able to visualize his profile and if he wishes, authorize the share of data from his Google Fit Account and the CDSS Platform. The server would then trigger a data request, via REST API to the Google Fit web service and then store the data on a JSON

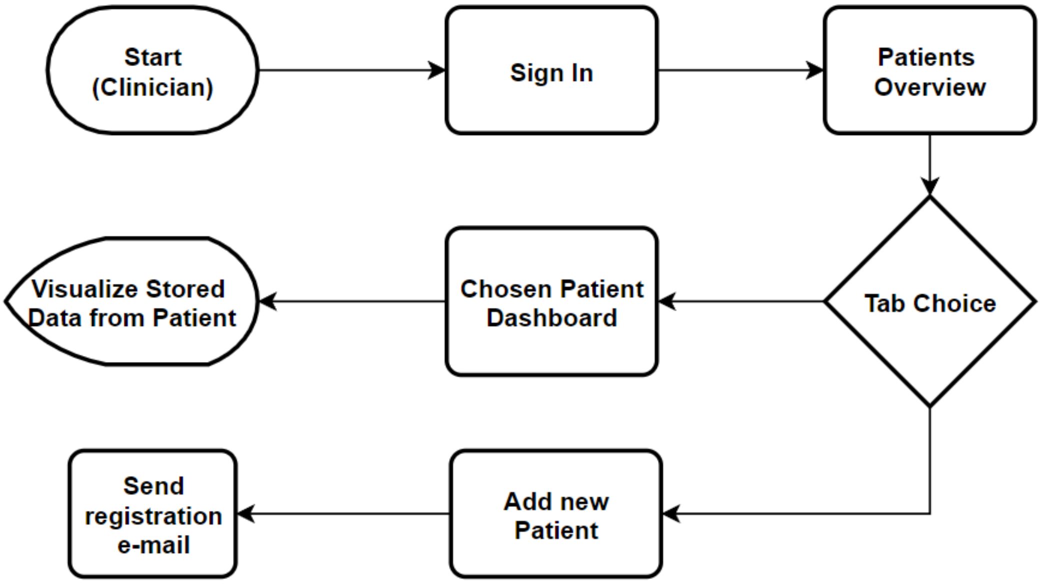


Figure 3.5: Clinician flowchart.

file internally. Figure 3.6 exemplifies the interaction between the patient and the CDSS Platform.

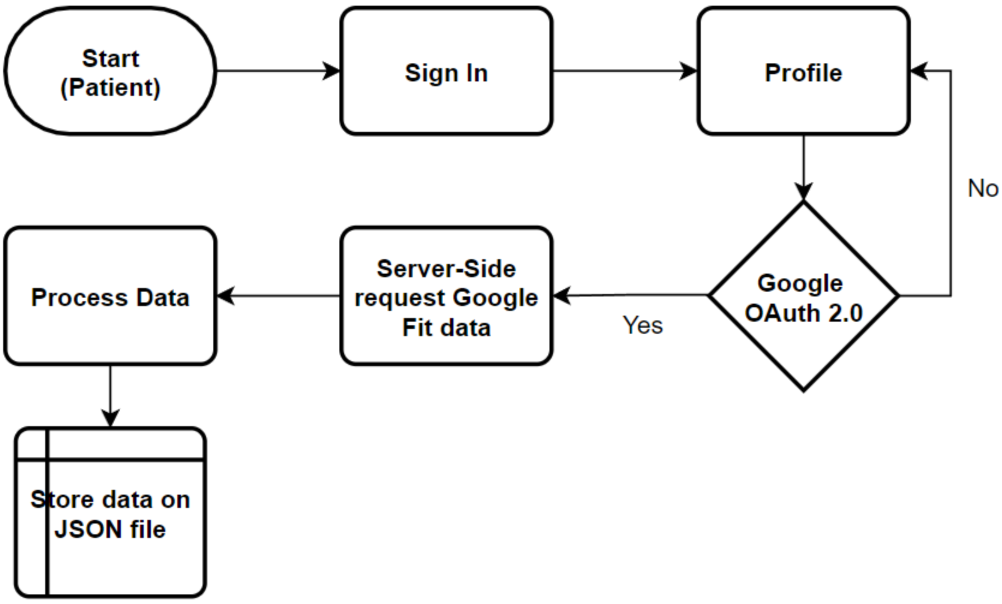


Figure 3.6: Patient flowchart.

The server requests data from up to 1 year ago at the moment of authorization, which would work as a good baseline to analyze and make comparisons.

The requested data must then be stored on the server-side for later use, in order to increase server performance and load balancing data processing. This is vital, since it would not be practical to request data and process it every time the clinician want to view the data, as it takes time and resources. So, by storing data on JSON files the information is always ready to use. This could rise a storage problem in a long-term scenario, so during processing all irrelevant data sent by Google REST API is ignored, insuring that only relevant data is stored. This process is illustrated on figure 3.7.

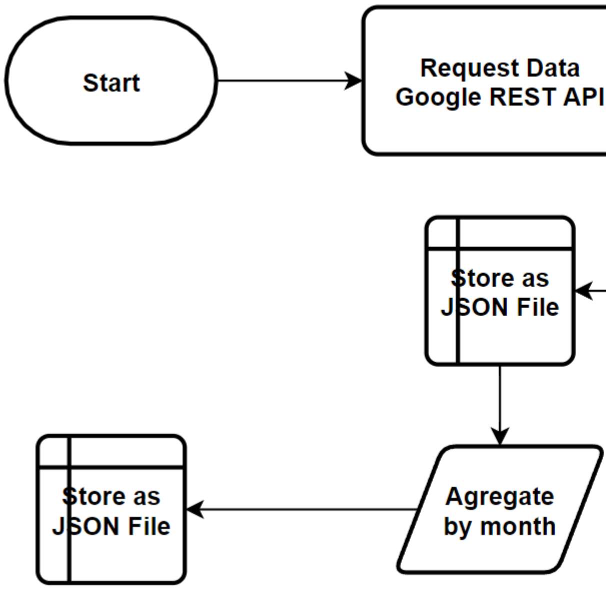


Figure 3.7: Data processing flowchart.

3.8 Fuzzy Logic

The Artificial Intelligence (AI) algorithm chosen for the platform is the fuzzy logic, since no dataset is available to train a model using an algorithm such as Artificial neural networks (ANNs). Although the later or even the neuro-fuzzy algorithm seem to be a good fit in terms of results from other studies, as already discusses on the state-of-the-art in the section 2.3, the lack of data compromises its application in this platform.

Those referred algorithms have another disadvantage in relation to the fuzzy logic, since both are more difficult to understand in terms of complexity and the black-box problem might be discouraged in the medical environments. Thus, the fuzzy logic seem to perfectly fit the designed principles for the CDSS Platform, as it is transparent, easier to understand and be improved.

The algorithm will be designed as proof of concept, since no expert knowledge is available. Nevertheless, as a first approach the algorithm will be defined conceptually with simple adaptations from the research made in the section 2.5.1. In the case of a second phase, such as a trial stage, clinical cooperation would be required to design a more reliable model for testing and finally later model tuning.

An interesting feature about the fuzzy logic, is the control in real-time that the clinician can have in the algorithm as he can fine tune it, without much effort. Neural networks do not provide the control on the algorithm, as its mechanisms are unknown or hard to understand and tweaking would require a lot of effort and time.

3.8.1 Fuzzy Logic Design

The platform patient's data that are stored as JSON files can be visualized with graphics, but this method is not analysing or neither processing the data, its basically showing the data that is requested by the Google Fit services.

Fuzzy Logic introduction, adds the capability of processing the data by analysing all inputs and giving an output that is related to a degree of truth, from a given membership function that is defined. It also work's quite similarly as the way human thinking is processed and decision-making is made so it is easier to understand by clinicians.

On this case the Fuzzy Logic algorithm will be responsible for defining how well the patient is progressing in the treatment, through an indicator designated as Patient Progress Indicator (PPI). This indicator is fundamental, since it will allow a more objective measurement of the patient's indicators gathered using both smart devices. The following illustration, fig 3.8, shows the fuzzy logic architecture.

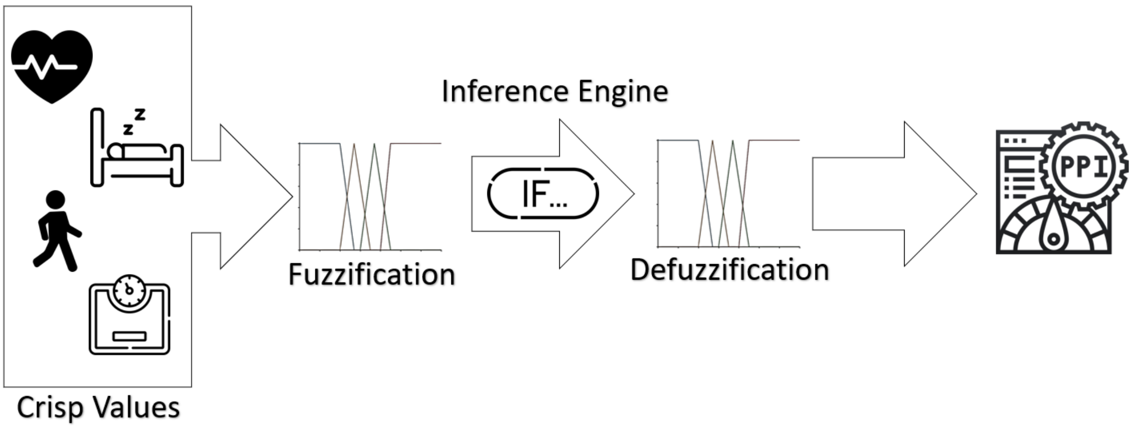


Figure 3.8: Fuzzy Logic Basic Architecture.

The illustration's middle sections is the core of the architecture and is divided in [90–92]:

- Fuzzification Module. This module is responsible for fuzzifying the input values, crisp values, into fuzzy sets.
- Knowledge/Rule Base. Responsible for storing the rules used for the fuzzy logic.
- Inference Engine. This engine processes the fuzzy sets inputs using the knowledge base.
- Defuzzification Module. This module is responsible for defuzzifying the outputted fuzzy set value into a crisp value.

3.8.1.1 Fuzzy Logic Alternatives Discussion

There is an alternative that could complement or substitute the implementation used in this architecture, in which a dataset could be used to develop the knowledge base, as discussed on section 2.3.2.

The dataset mining, using an algorithm such as clustering could work as a starting point of the fuzzy logic modulation. For instance, 75% of the dataset could be randomly selected as a training set for the clustering, resulting in several groups known as clusters. Each cluster center would then be transformed into membership functions, by analysing the clusters spread. Figure 3.9 illustrate an example result from the clustering algorithm.

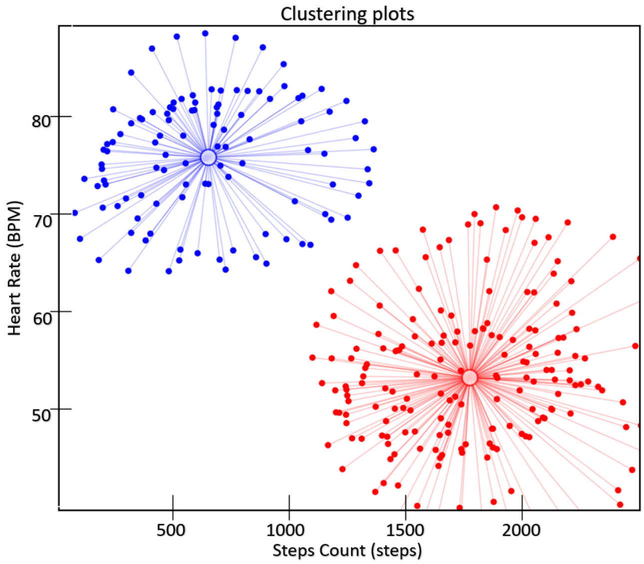
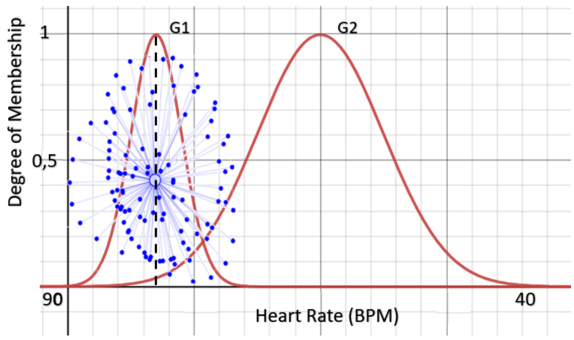


Figure 3.9: Example of a clustering plot result.

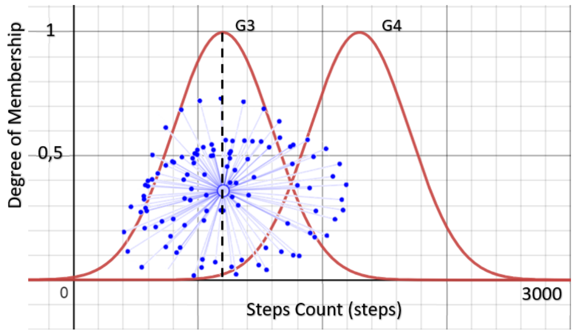
Observing the clustering plot, a membership function can be extracted by finding each cluster center of mass and the dispersion interval. So, the heart rate could be represented by two Gaussian membership function, one ranging approximately from 65 and 90 BPM and the other one from 40 to 70 BPM, the same process can be applied to the step count membership function. Figure 3.10 show a purely representative example of those membership functions, in which the blue cluster is overlapped with his membership functions for better visual comprehension.

Finally, the rule base can also be extracted by verifying each cluster group of data, for example on figure 3.9 considering the blue cluster representative of patients with lower Quality of Life (QoL) and the red cluster patients with higher QoL, it could be extracted the rule base depicted on figure 3.11. The rule base is generated by combining the membership functions that are related to each cluster and the resulting output is the classification given to each cluster.

The examples here illustrated are only a simplistic representation of how the fuzzy logic could be implemented, if a dataset was available and how it could contribute on the algorithm development.



(a) Heart rate



(b) Step count

Figure 3.10: Membership function example.

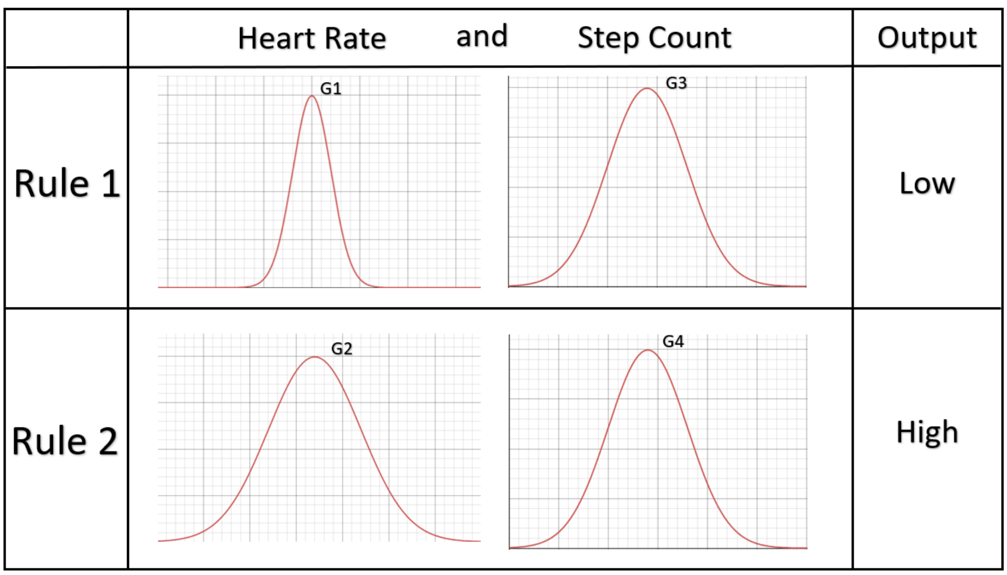


Figure 3.11: Rule base example (adapted) [38]

The modulation that the fuzzy logic algorithm has in its architecture eases the understatement that each module has in the resulting algorithm. Furthermore on section 4.6, a deeper analysis will be made on each module and what relevance they have in the resulting algorithm.

3.9 Platform Security and Privacy

Finally, basic security measurements must be taken in consideration although it is not the main focus of this platform, nevertheless every platform needs to guaranty basic integrity so that sensitive data is not wrongly showed to an unauthorized user. Security guaranties such as:

1. E-mail and password characters validation.
2. User/password verification and validation.
3. Encrypted password database storage.
4. JSON Web Token (JWT) for secure credential exchange and authentication
5. Hyper Text Transfer Protocol Secure (HTTPS).
6. Internally stored JSON files encryption.
7. Patient's related data stored by a secret identifier.
8. Uniform Resource Locators (URL) encryption.
9. User session timeout.

The above measurements 1. and 2. will ensure user credential validation and security, points 3, 4 and 5 guaranty credentials and data secure exchange and storage, the following points 6 and 7 prevents data storage spying and the last two measurements prevent brute force exploitation and user identity theft, respectively.

Those basic protection should be later improved and new measurements should be taken in consideration, since personal information privacy invasion is a serious thread, for a long-term much scaled and real use. The web application must also be compliant with the General Data Protection Regulation (GDPR) principles, in order to guarantee accordance with the data privacy and security law.

Concluded the data security design and future notes, the next chapter will discuss the platform implementation fundamentals that should be taken in consideration in similar approaches.

4 Implementation

This chapter will show with higher in-depth how the CDSS Platform could be implemented with a prototype example, and which technologies were used on major features.

4.1 Database Structure

The MongoDB database was designed with a simple structure composed by two collections: users and invites. The users collection as information related to the patient, clinician and administrator. Figure 4.1 presents the users collection with each user's document fields, as observable the "user type" field stores information that differentiates each user. Also, the patient is the only that has tokens in its document, those tokens are very important for data request and will be explained further on this chapter. On the right-side of the figure it is presented the invites collection vital for the management of new users, it has the e-mail from who invited the new user, the code to generate the unique link and the date to verify if the invitation has already expired.

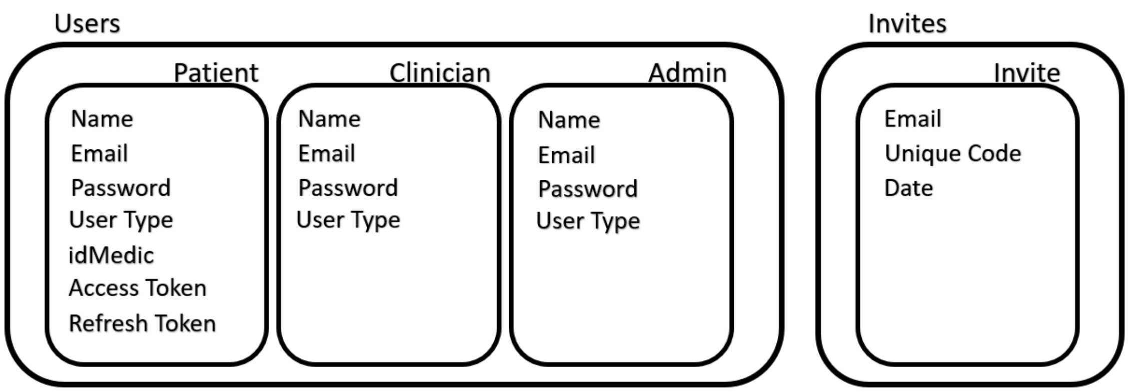


Figure 4.1: MongoDB dataset composition.

This database composition will be used to explain some of the platform's functionalities that will be presented in the following section.

4.2 Features Implementation

The platform has important features that are important to highlight as they contribute strongly to the implementation integrity and performance.

The user by accessing the web application will be faced with the home page, as illustrated on figure 4.2 and after pressing the "Sign In" button, will be asked to insert his credentials.

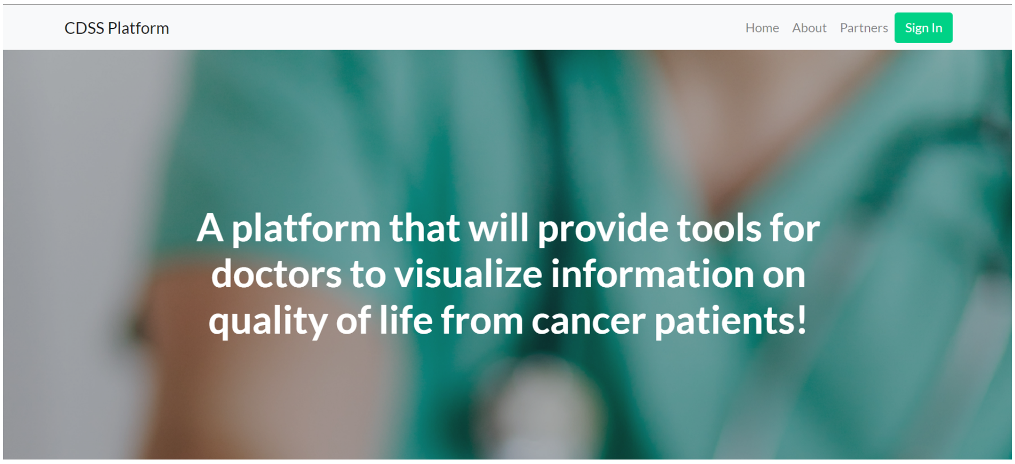


Figure 4.2: Platform home page.

4.2.1 Sign In

The sign in is handled by the front-end, as illustrated on figure 4.3, using a simple form that submits an Hyper Text Transfer Protocol (HTTP) POST method to the back-end server. At the server-side, security verification are made such as:

1. Input validation using @hapi/joi package, which is a data schema character validator.

2. User credentials validation, by searching on MongoDB, through mongoose user schema and bcrypt password comparison.

The user's password is encrypted on the database to ensure security using the bcrypt package, thus for user validation the password provided in the form needs to be encrypted as well and compared with the stored encrypted password.

Lastly, a JSON Web Token (JWT) is created using the jsonwebtoken package, which is then saved for future user authentication and send via secure Hyper Text Transfer Protocol Secure (HTTPS) protocol. This process will ensure integrity on the whole platform, since token authentication is required for every data exchange and access between the client-side and server-side.

For extra security the token as an expiration time of 1 hour, after this period the user is logged out and will need to sign in again in order to receive a new token.

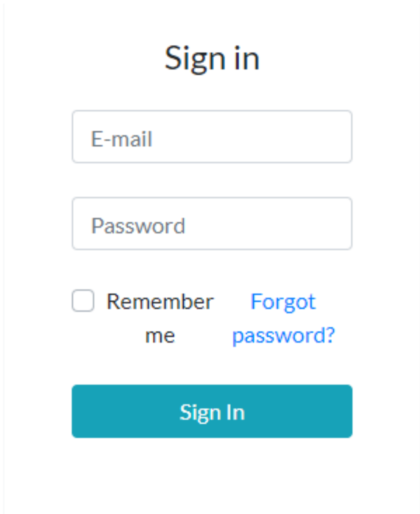


Figure 4.3: Sign in form.

4.2.2 User Registration System

The user registration should not be open to everyone for security reasons, thus the clinician and patient invitation should be implemented in a way that only allowed users can register in the platform.

So, the clinician's registration is performed by an administrator e-mail invitation, as depicted on figure 4.4, being the administrator an entity responsible for the platform management or someone in charged of this role.

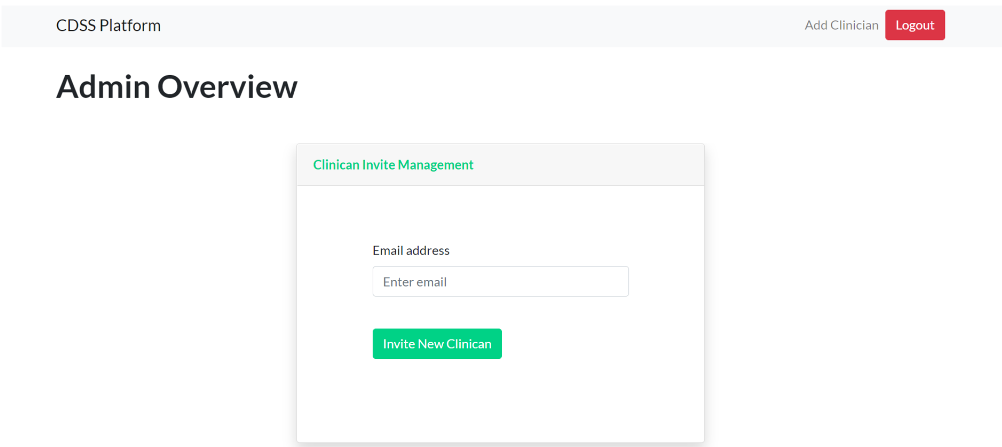


Figure 4.4: Administrator user management page.

The e-mail is sent using nodemailer package after verifying if the destination e-mail is not already registered in the platform, and as an unique registration link with information about the user type, that on this case is clinician. The unique link is created with the jsonwebtoken for security and integrity reasons and as an expiration time, as presented on figure 4.5. These information is stored on the MongoDB invites collection after the e-mail is sent, and will be used to compare the user's link to ensure its valid.

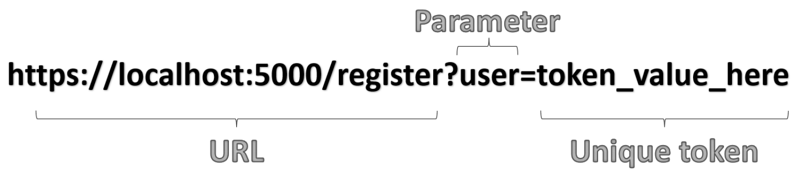


Figure 4.5: Registration e-mail example.

The clinician will be sent to the registration page, after pressing the link and will be asked to insert his full name and password to complete the process. The next step would be the clinician's patient invitation by navigating to the "Add new patient" tab, as depicted on figure 4.6, the process is similar to the previous invitation, the only difference is that the token has data related to the clinician that sent the e-mail. This additional information is crucial so that the patient is associated with the clinician and his data is visible to him.

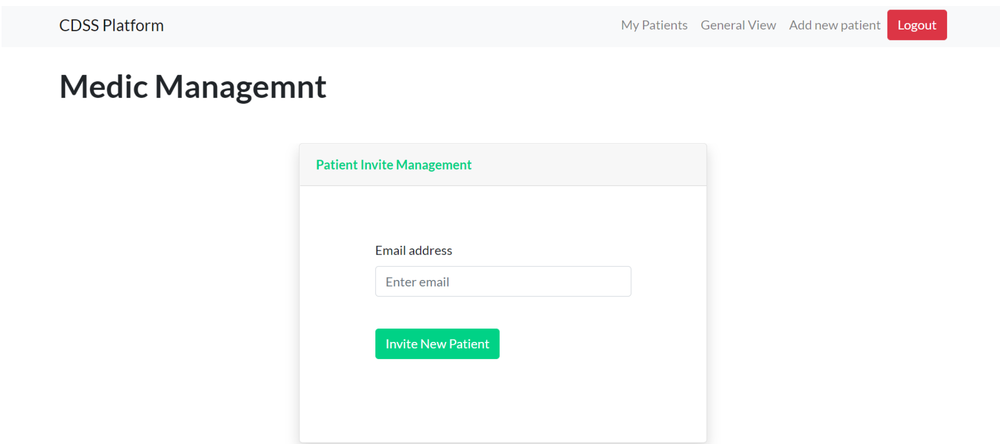


Figure 4.6: Clinician patient invitation page.

4.2.3 Patient Authorization

Having the registration process been completed, the patient will need to access his account and select one of two methods to make data available to the clinician, as depicted on figure 4.7, being those:

- Data share method: Authorize data share between Google Fit and the CDSS Platform;
- Takeout method: Use Google Takeout to extract data related to Google Fit and upload it directly on the platform;

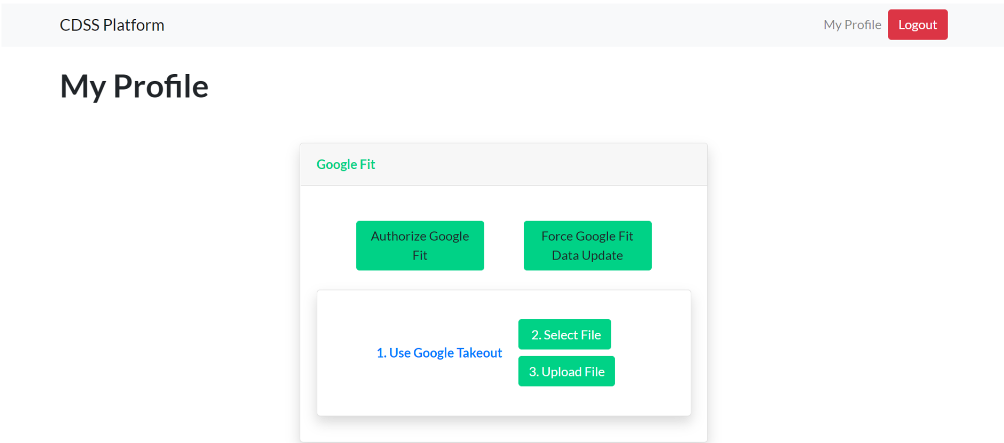


Figure 4.7: Patient Google Fit.

The optimal method is the Google Fit data share authorization, since the process becomes autonomous from that point on and no more patient interaction is required. Nevertheless, for the sake of the patient's free choice the second method is available. The takeout method requires the patient to visit the Google Takeout page by pressing on "1. Use Google Takeout" label and request Google Fit data ,as illustrated on figure 4.8a, to be sent via e-mail in zip file. It is also possible to schedule more exports so ease the process as it is possible to observe on figure 4.8b.

Finally, the patient has to download the zip file received via e-mail and select it on the platform by pressing the "2. Select File" button, visible on figure 4.7 and then pressing the "3. Upload File" button. This triggers the zip submission to the server-side via POST method, that is managed using the multer package.

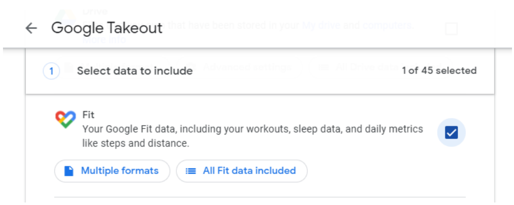
The server-side will create a folder, using the mkdirp package, unique and secret to that patient on the server storage and using the extract-zip package, extract the file before storing it. The folder would then be ready to be used when requested, since inside that zip file is a folder with several JavaScript Object Notation (JSON) files with all data related to the wearable device and the smart scale.

This approach is a complementary method and sub optimal, due to the interaction required by the patient, each time an update is required.

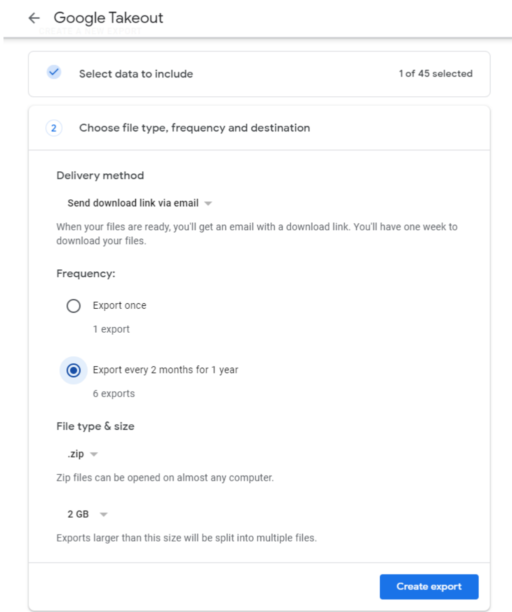
The data share authorization method is required because the platform depends on a third-party service, Google Fit, responsible for storing data related to the smart devices. So, the CDSS Platform needs to ask for permission in order to access the patient Google Fit stored data. Google uses OAuth 2.0 protocol for authentication and authorization, making this process easy and straight forward.

Googleapis package enables support for Google Application Programming Interface (API), so with a few lines of code the CDSS Web Application will redirect the user patient to the Google OAuth 2.0 page as illustrated on figure 4.9.

The OAuth page will then ask the user for data sharing permission and return a code



(a) Selection of data to extract



(b) Selection of extraction settings

Figure 4.8: Google takeout method example.

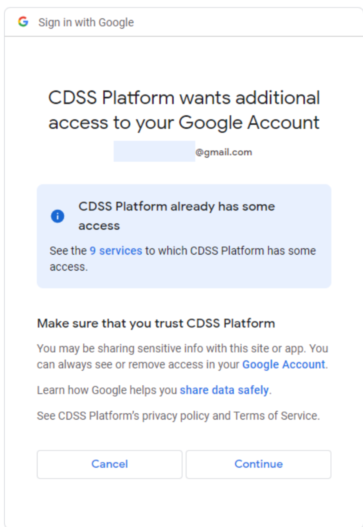


Figure 4.9: Google authorization form.

via Query String on the CDSS Web Application. This code is very important because it had the key to get an access token and refresh token, responsible respectively for granting permission on Google Representational State Transfer (REST) API requests and to get a new tokens when the actual access token expires. Those tokens are saved on the MongoDB for later use as it is required for all future data requests.

4.3 Data Request and Store

Surpassed the authorization problem, the server-side is then provided with the key to proceed to an actual data request. The back-end will automatically try to retrieve all data from the user-patient and store all available and relevant data. This process is repeated with each data type used, which are: step count, sleep duration, sleep interruptions, heart rate in Beats per Minute (BPM) and weight data.

Before any data is saved internally on the server-side, the server will create a folder relative to the patient unique to him using one unique identifier, using Node.js mkdirp package. The following explanation will be regarded to the sleep duration data but it is mostly similar to all data types.

On the first step, all available data from 1 year ago is requested to the Google Fit Service and saved as a JSON file designated as raw data, this is the same JSON file obtained when using the Takeout method so the following steps work as well. The raw designation is due the sleep dataset from each night is received as multiples chunks of information related to the duration and respective sleep stage. Since every sleep stage occurs multiples times during the sleep session as exemplified on fig 4.10, data should be aggregated/processed before it is actually useful, hence being called raw.

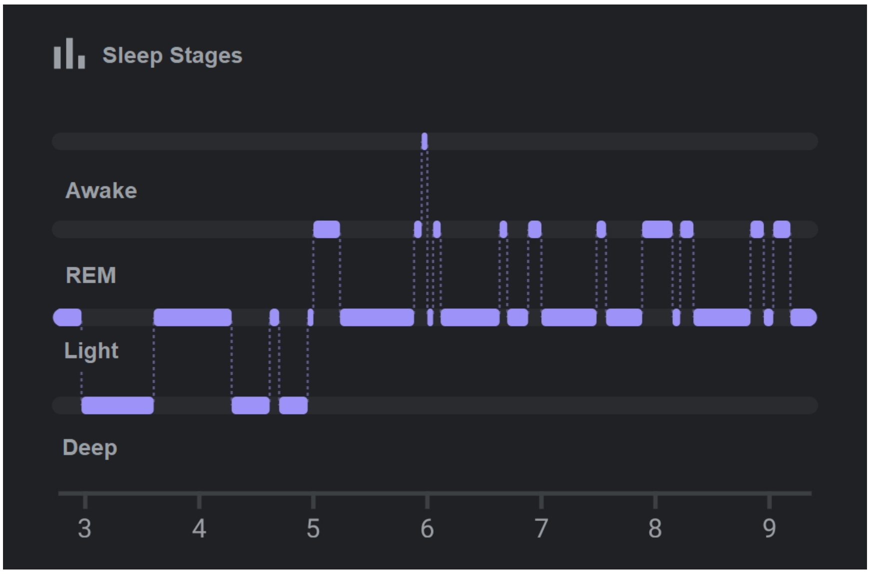


Figure 4.10: Google Fit sleep stages. (adapted)

The data it then aggregated by date so that all sleep stages duration's are summed on their respective stage, then the new processed information is stored in a separate JSON file, designated daily. This file is ready-to-use and will be capable of showing daily and monthly sleep data, nevertheless performance wise processing monthly data in real-time would require time and processing resources, so to improve performance another processing step is executed.

The last step will aggregate daily data into monthly data similarly as the last step, but now sleep stage duration will be summed by month and then divided by its respective number of days, resulting on average sleep duration per month. Finally, the aggregated data is stored as a JSON file and will then be ready-to-use as well. This process makes data more accessible, since just reading data from the internal storage is much faster then reading data and process it in real-time.

4.4 Data Continuous Update

The referred process ensures historical data request up to 1 year, if data is available since the patient might already use the Google Fit application. Nevertheless, to maintain continuous data monitoring there should be a way to update new data systematically after the initial data request, explained previously. Three different methods were implemented to secure redundancy: patient forced update, clinician forced update and platform autonomous update.

The patient as the functionality to force the data update, as depicted on figure 4.7, by pressing the "Force Google Fit Data Update". This feature is intended to be used only as a backup solution, in the case of the clinician or the platform for some reason not being able to do it on its own. The clinician also has the capability to force a data update, by searching the patient on his "My Patients" tab as presented on figure 4.11. However, this interaction is once again not optimal and should be only used in case that the platform fails, for some reason to make the update automatically. Both presented features were implemented as a redundancy backup feature, the following method is the main data updating process executed in the server-side as an automated and continuous process that does not require any human interaction and enables continuous data monitoring.

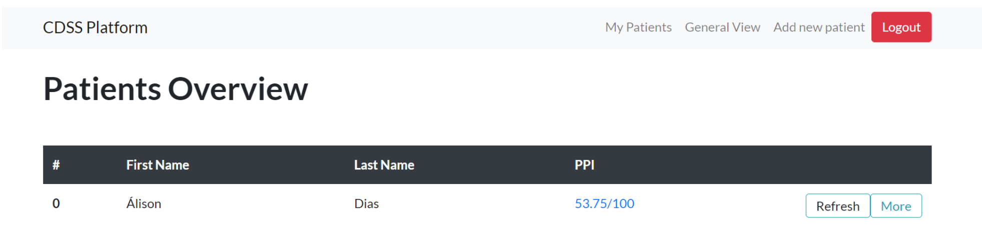


Figure 4.11: Clinician patients overview page.

The data automated update is executed every 2 hours using the node-cron package, which allows a constant monitoring of each patient. The process starts by verifying each patient in the platform's MongoDB database and requesting the last stored refresh token. This step is important because the access token has an expiration time of 1 hour, thus a new access token is required in every update. The patient old refresh token is used to get a renewed access token and refresh token using the googleapis package. Both tokens are stored in the patient's document on MongoDB for later use.

The process is then made similarly as the first data request explained on the previous section 4.3, but to improve performance the data requested is done from the last day on storage. So, in the scenario that the platform is up-to-day until 10 of October the data request is from that day inclusive, until the present time. Including the last day is important to maintain coherency and performance, because in some cases data close to the present moment might come incorrect or incomplete, and this process ensures data recheck. With the request process completed the data is added to the server storage and can be visualized by the clinician with up-to-date information.

4.5 Data Visualization.

The main objective of the purposed CDSS Platform is giving clinicians a visualization tool capable of quickly and effortlessly show patient's new insights that were not available before. So, with this goal in mind, several charts were implemented using chart.js a javascript charting library. Each chart can filter data related to the last 2 months, per month and per daily data from the beginning of the treatment as illustrated in figure 4.12, for a treatment that began in April.

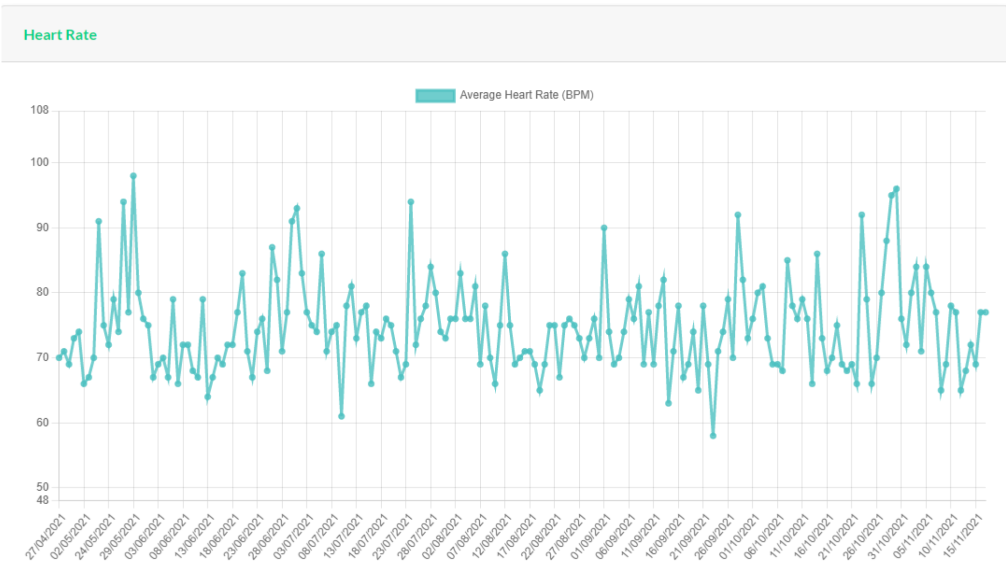


Figure 4.12: Daily data chart concept for heart rate.

The charts related to heart rate and weight data are visualized has line charts for better understanding of value fluctuations without to noisy visual information. Sleep data is visualized either with bar and line chart simultaneously, on the average data from each month and last 2 months data, and only with a line chart for the daily data, since in this case too dense granular information would visually pollute and difficult understanding. With this mixed chart, sleep data can inform clinicians about the patient sleep stage quality (bar charts) and total hours slept (line chart), in each night.

Finally, data related to number of steps is visualized with bar charts, since it gives quantified information by bar height and density information created by adjacent bars, where the clinician can easily check in which period of time the patient walked more regularly. For example, by checking bars local density, has illustrated on figure 4.13 an observer can easily spot on the left side a low density and height area, and on the right side spot an higher density and height area.

This chart selection will ease visual information processing without the need to pollute each chart with to much unnecessary information, staying in line with the state-of-the-art subsection 2.4, where was stated to keep visual data as simple and cleaner as possible.

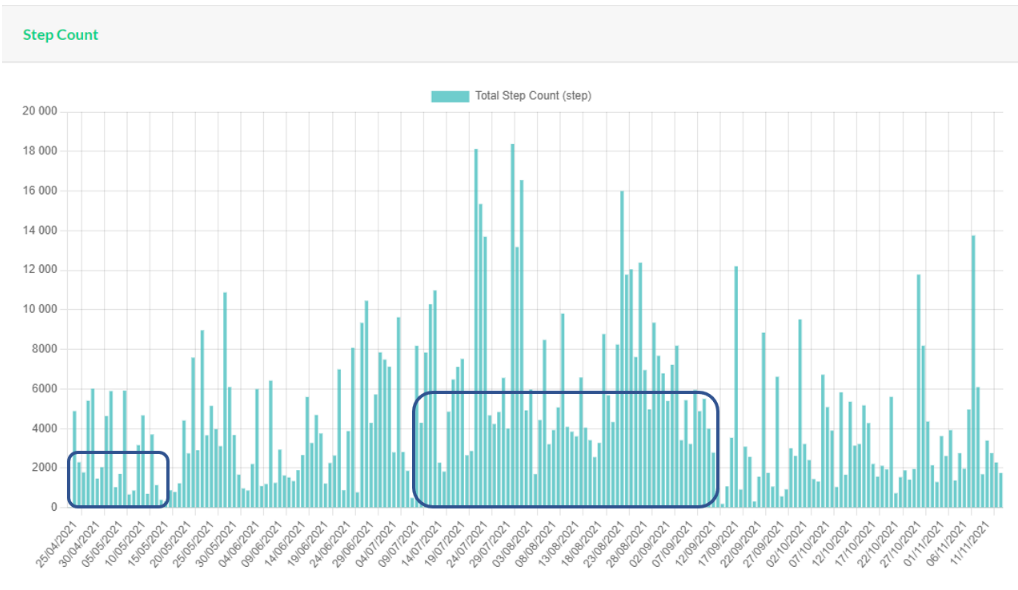


Figure 4.13: Steps density visualization example.

The color schema used on the data visualization tools are also of extreme importance and a special attention was taken, since polluting the visual experience with too vibrant and distinguishing colours might distract and confuse the observation. Thus a neutral color schema was chosen and taken through the whole user interface. Furthermore, choosing universal colors to distinguish positive data from negative data is very important to quickly pinpoint where the clinicians should focus his attention.

The usage of graphical representation to ease clinician's visualization of patient's progress might be a powerful tool as well. This is useful specially in the cases where the patient is, for example, reducing his activity. The clinician could challenge the patient to walk more daily, as it normally would do, and at the same time register this challenge on the platform. By doing so, on the next appointment or in between appointments the clinician could evaluate how well the patient is progressing by checking the progress bar.

The data visualization tools will be complemented with the implementation of the fuzzy logic algorithm, the following section will explain in detail its implementation.

4.6 Fuzzy Logic Implementation

Several important implementation aspects directly affect the fuzzy logic performance. Furthermore, the fuzzy logic output should give a relevant, valuable and usable information, in order to help in the clinical decision-making and function as an objective patient progress indicator.

The fuzzy logic was implemented with python using some already existing libraries such as: numpy, matplotlib.pyplot and skfuzzy. The python code was then integrated with the Node.js using the python-shell library. Those libraries made the implementation quite straight forward and enabled to focus all attention to what was really important, the problem modulation. The following table 4.1 explains briefly each library.

Table 4.1: Python packages brief description.

numpy	Library used to work with arrays, algebra and others mathematical methods [93].
matplotlib.pyplot	Library sub-module used to plot data in visual diagrams [94].
skfuzzy	Fuzzy logic algorithms library [95].

4.6.1 Fuzzification Module

The first step and probably the most important is implementing the fuzzification module, since it is responsible to transform input values in data that can be used to infer and process into a given output.

The fuzzification is possible by defining a fuzzy set to each step of each input type, which is easily defined using the skfuzzy library. The implementation of each data step was made by defining a respective linguistic name and respective interval of values, as shown on figures 4.14, 4.15, 4.16, 4.17. A step can be defined as a linguistic variable that will be parameterized with a specific range of values.

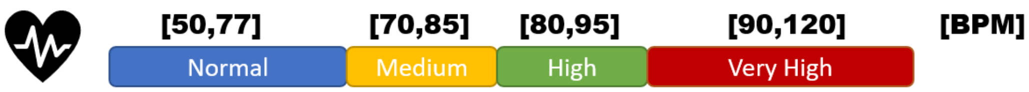


Figure 4.14: Fuzzy set steps for heart rate input.

For each input type step, intervals are parameterized in accordance to values from articles that support those intervals with some adaptations, as such, for this CDSS Platform those values are considered as trustworthy for an initial prototype stage.

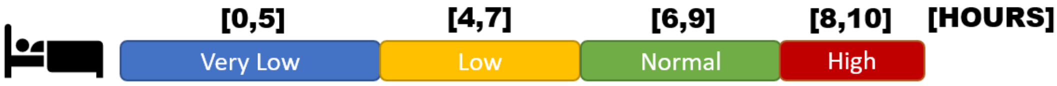


Figure 4.15: Fuzzy set steps for sleep input.

Figure 4.16: Fuzzy set steps for weight variation input.

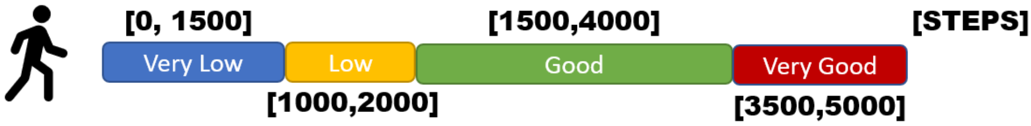


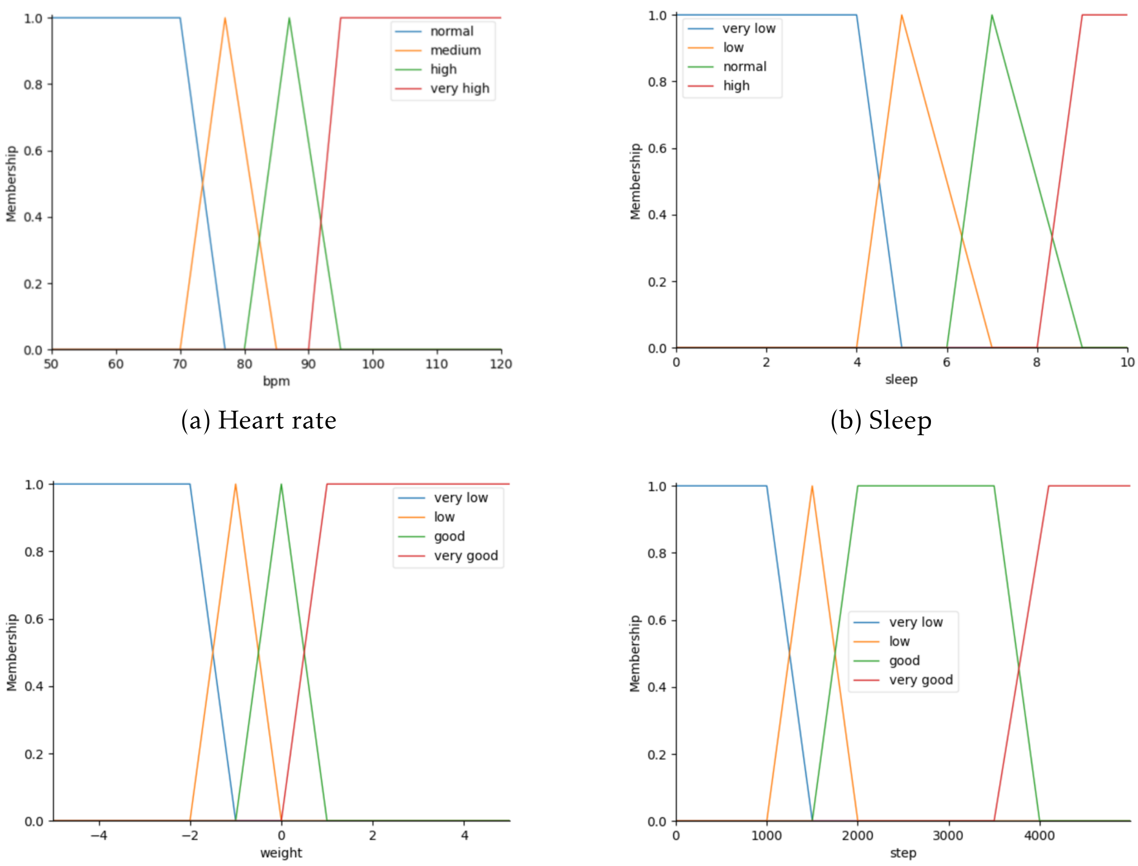
Figure 4.17: Fuzzy set steps for steps input.

Those fuzzy set steps will be used to create each membership function. In this implementation the type of fuzzifier chosen was the trapezoidal fuzzifier, since it simplifies computation, development and eases understanding for future development with clinicians.

The membership function was implemented having in consideration the referred parameterization for each linguistic variable. The only precaution to have in mind is, in which value the linguistic step should hit pick value 1 or lowest value 0.

The following figure 4.18a shows the heart rate membership function in BPM. As it is observable the heart rate is divided in different steps each with its own meaning, for example, normal heart rate (in blue line) is defined from 50 BPM to 70 BPM with a degree of membership of 1 and from there to 77 BPM it starts declining until it hits a degree of membership of 0. This means that the normal heart rate goes from 50 BPM to 77 BPM but its degree of membership is not always 1. So, following this line of thought if a input value for heart rate is 75 BPM, then the degree of membership would be 0,281 for normal and 0,704 for medium. The conclusion is that a patient with a average heart rate value of 75 BPM has a heart rate that is not normal or medium, but a mixture of both in which is more medium then normal. This shows clearly the advantage of fuzzy logic, since it does not classify an input as good or bad but rather in a similar way like the human reasoning would.

Figure 4.18b and 4.18c show the membership function for the sleep and weight variation input in hours and kilograms respectively, both similar to the heart rate and the step count membership function. The step count membership function is slightly different from the rest, as observable on figure 4.18d. The variation on step membership function is due to the range where the step count is defined as good, in other words from 2000 steps to 3500 steps the degree of membership is 1. So, if the input for step count is 3000 the step count is definitely good, but in the other hand if the input value is 1750 then the step count is neither good nor low, but once a again a mixture of both and in that



(a) Heart rate

(b) Sleep

(c) Weight variation

(d) Step

Figure 4.18: Membership Functions.

case 0,5 for low and 0,5 for good. This output might seem erratic, but when considering the human reasoning process it is plausible that sometimes defining a linguist term to a specific value might be hard and subjective, hence the subjectiveness of defining 1750 steps as 50% low and 50% as normal.

Those membership functions will be responsible for defining the defuzzified output value based on the fuzzy output set, but before that a rule base and an inference engine is required.

4.6.2 Knowledge/Rule Base

The knowledge base used on this fuzzy logic implementation was mainly built with common sense and based on some articles. So for the resulting implementation, 4 rules where defined in order to satisfy each output step from the membership function.

Table 4.2 summarizes the knowledge base used to create the if-then rules. So by observing the first line it is possible to determine that in order to obtain an output fuzzy set of very good the input values should be normal, high, very good and good for heart rate, sleep, weight variation and step counter respectively. The performed operation is a OR, which represents a MAX, in other words the maximum input value will define the final output value.

The knowledge base is one of the core points of the fuzzy logic algorithm and its implementation affects directly the algorithm output. Although basing the knowledge base from articles might be a good starting point for this prototype, a more advanced stage would require a dataset used to understand how to better implement the knowledge base rules by verifying data consistency for a given output, as explained on section 3.8.1.1 or the use of the know-how from a specialized medical team with experience that would help in the creation of the knowledge base. Both methods are not viable, since a dataset would require a prior trial with real patients wearing the devices during the clinical process and then data classification would be required by a specialised medical team or a specialised medical team would be required to cooperate with the development of this CDSS Platform and afterward tested in real patients.

Given the constraints to develop a better knowledge base for the platform this prototype stage will take in consideration the example knowledge based shown on table 4.2 as a proof of concept.

Table 4.2: Fuzzy Logic Knowledge/Rule Base.

Heart Rate	**Sleep**	**Δ Weight**	**Step Count**	**Output**
Normal	High	Very Good	Very Good	Very Good
Medium	Normal	Good	Good	Good
High	Low	Low	Low	Low
Very High	Very Low	Very Low	Very Low	Very Low

4.6.3 Fuzzy Logic Inference Engine

The inference engine is responsible for processing the inputs already fuzzified and by using the if-then statements output a fuzzified result. In this case the inference engine is already implemented in the used python library skfuzzy, hence the success of the fuzzy logic algorithm will not be determined by the inference engine and rather by the remaining modules. Thus, no further information will be revised about this component.

4.6.4 Fuzzy Logic Defuzzification Module

The last step of the fuzzy logic algorithm is to defuzzify the output resulted from the inference engine. This happens because the inference engine will evaluate each rule, thus combining each membership function and output a fuzzy set.

The process of transforming the fuzzy set into a crisp value is the defuzzification and it requires a membership function similar to those used on the fuzzification module, as figure 4.19 shows.

The output resulting from the defuzzification must have a simple and understandable meaning to the clinician, since this output should support and aid the clinician decision-making. Thus, the purposed output will range from 0 to 100, working as a percentage rating that will define an indicator designated as Patient Progress Indicator (PPI).

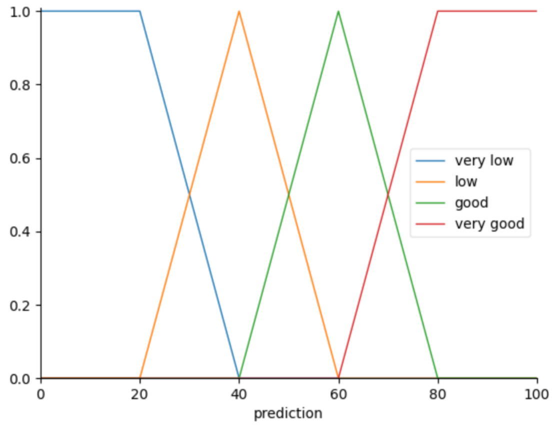


Figure 4.19: Output Membership Functions.

As already referred, each input has 4 steps with their respective linguistic term, so the output membership function was also designed with 4 steps and to create some equality between each of them they were evenly divided. Hence, each linguistic term has a total of 40% of the total range which is possible due to overlap between membership functions.

With the implementation review, the following chapter will center its attention on the results obtain for the this prototype design and implementation.

5

Results and Discussion

This chapter will be focused on the understanding of all results acquired by the design and development of the purposed CDSS Platform. Furthermore, the results obtained in terms of integration of the whole architecture, the visualization tool and the algorithm responsible for the Patient Progress Indicator (PPI) which will be responsible to aid clinical decision-making will be analyzed and discussed.

5.1 Architecture Integration

The key point of the purposed CDSS Platform was to enhance clinicians with a new insight on patients health without the need of scheduling more appointment, this would be a determinant factor on the improvement of the patient well-being, since more monitoring would be provided without the need of requiring more time from both patient and clinician. Another important key aspect is the possibility to notify the clinician in case of patient health sudden or severe decrease during the period in between appointments, so for example if suddenly the patient has an huge increase in heart rate and a significant decrease of hours slept in the last week it might be a predictive sign that the patient's well-being is deteriorating and per consequent its Quality of Life (QoL). On those scenarios it might be urgent to contact the patient and insure his well-being and in last instance schedule an urgent appointment in order to plan new recovery/alternative strategy. This aspect is easily managed using the node-cron and nodemailer libraries, as already referred, since both allow constant patient data monitoring and email sending feature.

The wearable device gives the opportunity to gather data continuously throughout the whole day, monitoring the patient. The wearable device used to test and develop this CDSS Platform was a smartband and integration was successfully managed by the usage of the third-party Google Fit Representational State Transfer (REST) Application Programming Interface (API). Since the smartband used was a Xiaomi Mi Band 5, the patient would need to install the Mi Fit application so that the smartband could send its data to the Xiaomi application. Nevertheless, the patient would need to install Google

Fit application, and associate the Google Account on the Xiaomi application as shown on figure 5.1.

Figure 5.1: Mi Fit data share with Google authorization. (adapted)

The data sharing step can be replicated with any other smartband brand and model, as long as the smartband application is compatible with Google Fit application.

The resulting integration from the smartband, the third-party application and the CDSS Platform worked seamlessly and data flowed with no extra complications.

5.2 Data Visualization Tools

Completed the integration phase of the development of the CDSS Platform, the next objective was to empower clinicians with a visualization tool capable of giving new insights about the patient daily life and health measurements.

The data that can be used by clinicians to be visualized is the available in the Mi Band 5 device which are summarized on table 5.1. So using Google Fit, data from heart, sleep and step was gathered and stored in the platform. In addition to those, a Xiaomi Mi Body Composition Scale 2 was used to monitor weight from time to time. The integration of this device was effortless, since the Xiaomi Mi Fit application connects directly with the weight scale, saving this data in real-time. Since the Mi Fit application has authorization to share data with Google Fit application, data can be stored by Google an accessible at anytime by Google Fit REST API. This device could be used only during each clinical appointment to track weight variation or even used daily, if the patient had the possibility of having one at home.

The main advantage of those devices is their mass commercialization, making them significantly affordable in comparison to medical specialized equipment, although precision wise those commercial devices are way behind.

Table 5.1: Summary of data monitored on the Mi Band 5.

Monitored Input	**Description**
Heart	Heart rate data automatically measured periodically.
Sleep	Sleep duration and sleep stage tracking.
Step	Step count through the whole day.
Activity	Activity duration and calories burned monitoring.

The development of the visualization tools required design principles that were in line with the principles addressed on the state-of-the-art chapter 2.4 and feedback from researchers. The following sections will explain how each of those aspects influenced those tools implementations, keeping visual pollution at its lowest and highlighting what is important.

5.2.1 Visual Tools Design

The graphical representation chosen for the implementation were simple charts, more specifically line, bar or mixed charts. The reason for these choices were due to a simpler data aggregation when used with fewer series of data are used. The focus was to keep visual information as simple as possible without losing information and effectiveness.

The data related to similar aspects were also placed next or close to ease comparison and data interpretation. The color schema was also taken in consideration, in order to simply the analysis and observation without losing the focus on whats important. Also, colour was specifically used to ease a quicker view of where the clinician should focus his attention.

The validation was also very important to gather feedback from different sources and understand what and how data should be shown to the clinician, so data could be presented on its most effective way.

5.2.2 Tools Validation

The visual tools developed suffered several changes during implementations due to feedback received from Champalimaud Foundation researchers, partners from the project FAITH, whom advised and cooperated to create a baseline to guide the clinicians through the visual process. According to them, the clinician would be much interested on comparing the patient's data with his own older data or even with overall population average data, this baseline would then be used to understand how the patient progressed in relation to that baseline. The conclusion that resulted from the discussion of the best method to create the baseline, was that the patient comparison with others patients would be imprecise due to the subjectiveness of the baseline. For example, one patient could be fine by sleeping six hours a day, since normally that amount of sleep was enough to give a good overall QoL, nevertheless for another patient six hours was not enough since normally eight hours was the amount of sleep required to make him feel good. This resulted on the assumption that a baseline related to the patient own historical data would be the best fit for a baseline.

The persistence of negative results was another important aspect discussed, since it would be important to quickly inform the persistence of bad sleeping experiences and for how many days or weeks the patient did not sleep well in relation to the baseline, for example. This was an important factor to have in consideration since a decrease of QoL is not related to one or two days without a good night of sleep, since it is normal to happen sometimes, but rather the persistence of bad nights of sleep. This is equivalent to the rest of inputs analysed on this CDSS Plaltform, since one day or two with a higher heart rate might not be a problem but rather when it is persistent for several days or even weeks. Data consistency in relation to days of the week was also a good comparison tool, since in the weekend people might tend to walk more or be less stressed/anxious in comparison to the rest of the week.

The color schema was also a very important point to consider, according to the researchers and they agreed that the universal knowledge for the red and green colors as negative and positive values should be used to quickly demonstrated where attention should be given. Furthermore, it was unanimous that the visual tool should focus mostly on the negative data, since they pinpointed where the clinicians should take action.

The feedback received from the Champalimaud Foundation researchers shaped the visual tools implementation and their features, in order to better fulfill the clinicians needs.

The researchers also stated that it was important to observe the patient progression after a negative event, such as a sleep disorder event. Since it will not be a singular event that is going to indicate any QoL increase or decrease, but rather successive negative events and its progression throughout a long period of time. So, it is vital that the tool enables the clinician to observe successive negative events easily.

Filtering features were added to the visualization tools, to enhance the clinician experience and to better observe the data. Three different filters can be used: last months, average per month and daily. The next sections will focus in each one of them and explain how they can help the clinicians decision-making.

5.2.3 Visualization by Last Months

Filtering data by last months will display information related to the previous two months, giving to the clinician a much granular view from recent times. This tool will be useful to better understand how the patient progressed and understand with a higher detail how it could improve its routine.

The figure 5.2 shows an example of visual data displayed in the platform filtered by data from the two last months. As it can be observed, a clinician could easily see at first glance major spikes on the patient's heart rate in Beats per Minute (BPM), between 22 and 31 of October and since those values were above the guideline, they are red to draw the clinician's attention. Furthermore, with a deeper analyses the clinician could check that the patient heart rate stayed mostly under the guideline, which could be a good indicator of the patient health or even the patient's levels of anxiety throughout the months.

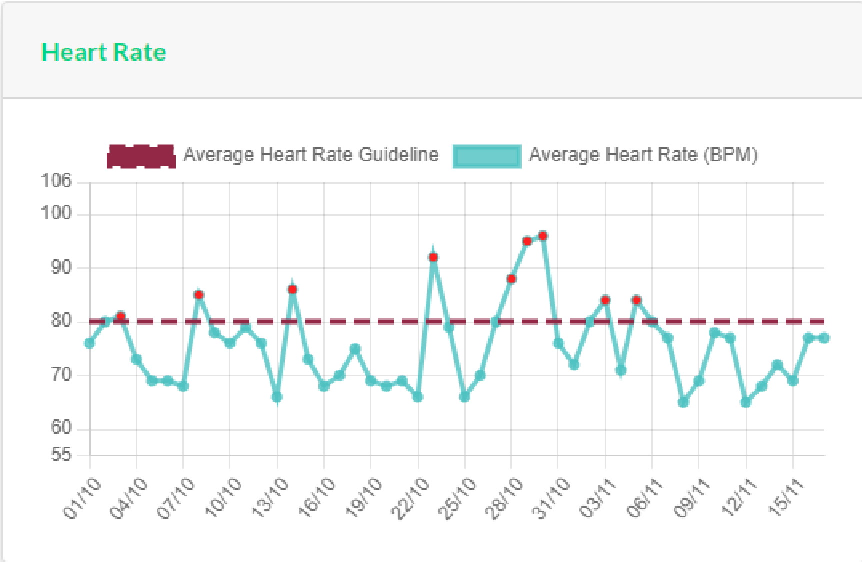


Figure 5.2: Last months heart rate chart example.

Those insights by its own might help the clinician better understand in which periods of time the patient felt more stressed or anxious, as it could be the case from the periods when the major spikes were observed. However, more information is required, since those variations could be related to a more active day, such an hiking activity for example, in contrast to a routine day in which the patient stayed mostly sitting or in a resting state.

This information could be enlightened during an appointment by questioning the patient about his routines on those periods of time, but this would focus the burden on the patient memory and capacity to recall his past weeks, so to improve the information reliability, the clinician could observe the Step Count chart, located next to the previous. So observing figure 5.3 at first glance it is possible to verify that the patient activity was on average higher then the guideline of 2000 steps.

A more careful look, in order to observe if there were any correlation between the heart rate spikes and the steps count, show that at 23 of October the step count was in its lowest, which was inversely correlated with the heart rate of that day. Observing both values at 28 and 29 of October it is observable that patient had a more active day having respectively a step count of 11.782 and 8.188, which is in line with the higher heart rate experienced on those days. These complementary information could help the clinician

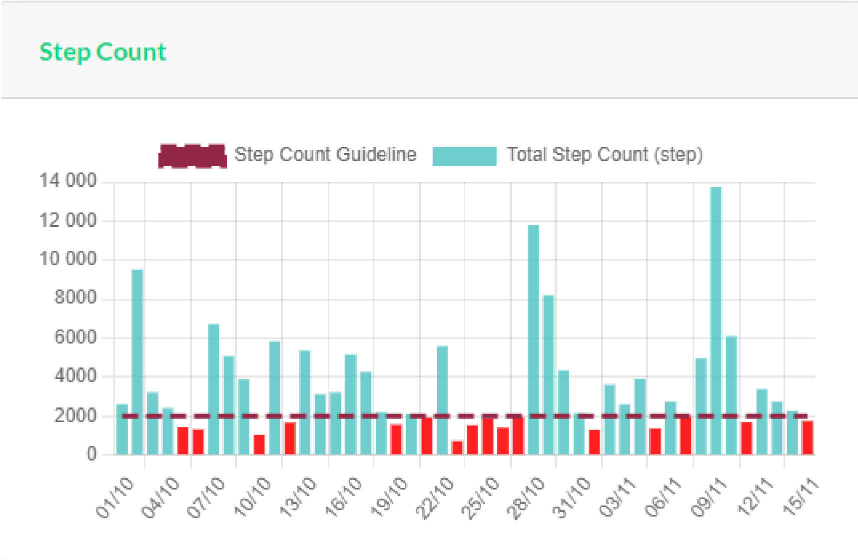


Figure 5.3: Last months step count chart example.

make better conclusions, such as: at 23 of October the patient could have experience high levels of anxiety, taken from the low step count and the high heart rate and at 28 and 29 of October the patient experienced a higher heart rate derived from a more active day, so probably it was not correlated to an anxiety event.

The sleep data is also vital to understand the sleeping routines from the patient, since it might pinpoint moments that the patient suffered stress and/or anxiety which lead to a significant decrease of the sleep quality, since it affects directly the QoL of the patient, hence the importance to extract as much information as possible from this factor.

Observing figure 5.4, at first glance it is easy to spot in which moments the patient had episodes of sleep disorder, by checking the lack of bar chart height.

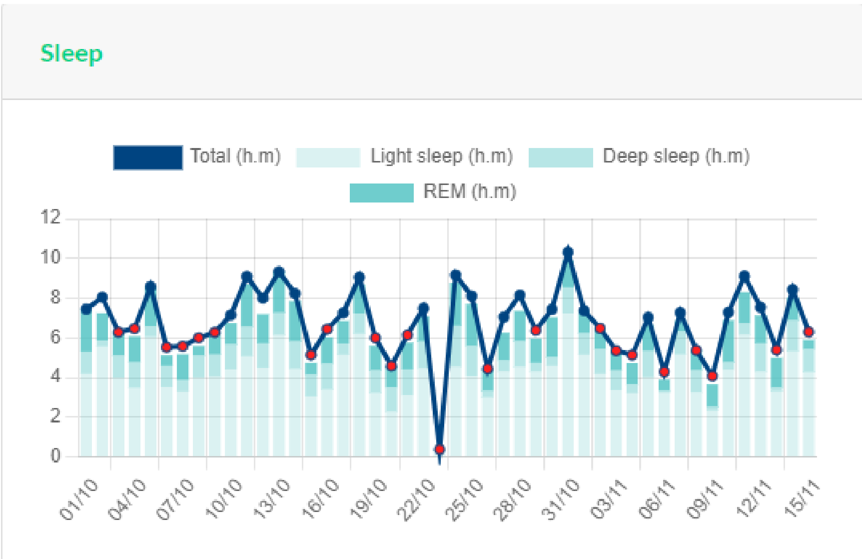


Figure 5.4: Last months sleep duration chart example.

It is also easily observed that on 23 of October there was a huge lack of sleep, which is inline with the already analysed indicators that could indicate an anxiety or stressful event. Once more, the sleep chart reinforces this theory, showing the essential insight and role that this platform has by allowing the clinician to correlate those 3 indicators and be aware that clinical action might need to be taken.

The weight variation chart could be useful to the clinician to easily observe the historical evolution of the patients weight, as it might give insights on the patient health or even work as another anxiety/stress indicator.

A quick observation to the weight variation chart, on figure 5.5, show an initial increase in weight and later a period of minor weight variation that is inline with the defined guideline. These information could work as a good indicator of the patient's diet, since a the more anxious period of life could result an increase of decrease in weight, due to mood variations or health problems related to the overall QoL.

Figure 5.5: Last months weight variation chart example.

The key factor of this tool is the effortless enhance of the clinician knowledge of the patient weight variation. This tool capability is optimized in the case where the patient has a weight scale at home, such as the smart scale used in this dissertation, thus increasing the weight monitorization substantially.

The results observed on this platform should not be blindly treated as true, since as already mentioned, the smart devices commercially available are not medical certified and should only be used as complementary information, as discussed on the state-of-the-art sub-section 2.2.1. For instances, the sleep information from 23 of October could be misleading, in case that the device just ran out of charge during the night or a malfunction occurred in the device or even in the Google Service.

Additionally, due to the importance given to sleep indicators throughout the platform development, a sleep interruption chart and sleep report chart were added. The sleep interruption chart exemplified on figure 5.6, shows information related to the number of interruption that a patient suffered during his sleep. This is an excellent tool to understand the sleep quality of a patient, since sleep disorders prolonged in time have huge impact on the patient QoL. For example, the patient may inform in a clinical appointment that normally he suffers from 1 or 2 sleep interruption specially when trying to fall asleep. The clinician then defines the guideline as 2 interruptions per sleeping session, and then through the platform the clinician can follow the patient progress and observe if the patient is improving his sleep quality. This information is vital, in order to understand the patient sleep quality and then be able to act accordingly.

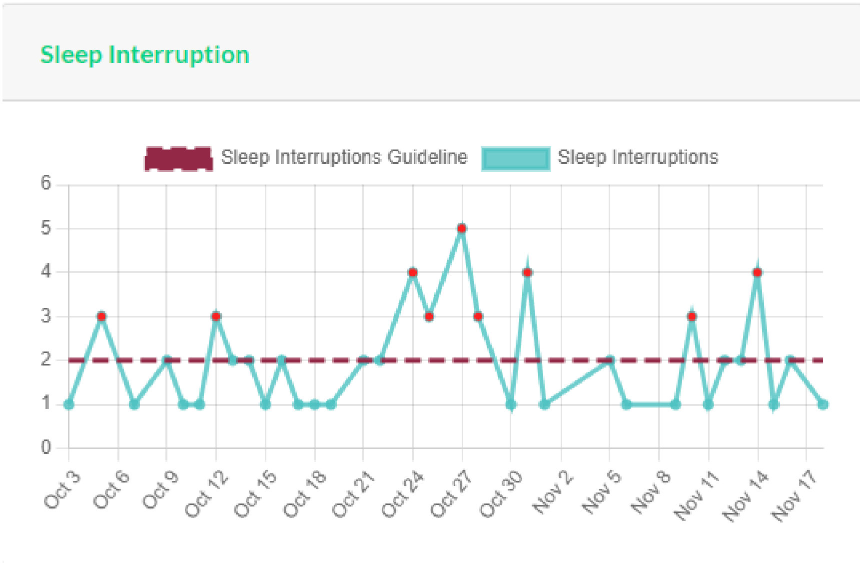
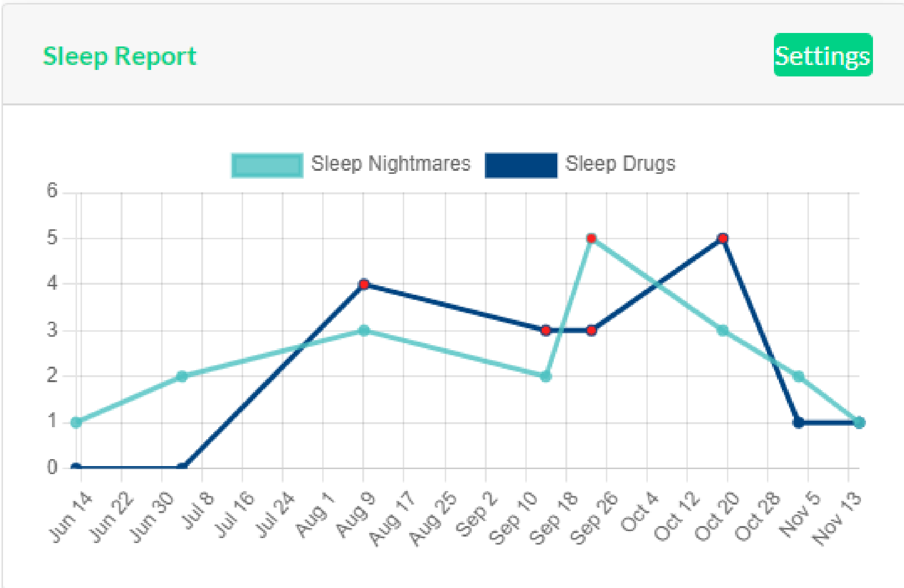


Figure 5.6: Last months sleep interruption chart example.

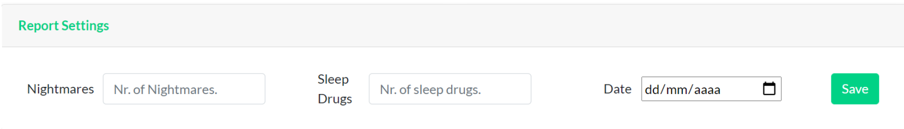
Finally, the sleep report chart, on figure 5.7, is used as a follow-up tool useful during appointments or even remote appointments via phone call to understand more about the patient sleep quality. This chart shows information related to the number of nightmares experienced by the patient and the number of sleep drug used, gathered by questioning the patient directly. Nevertheless, this information could be gathered with the use of a mobile application, that will be discussed in the section 6.2, in order to automate this data flow.

The last tool implemented was a progress bar as shown on figure 5.8, that could be used by the clinician to set goals for the patient and in real-time observe its improvement. The tool potential is that it enables the data monitoring and in case of need the clinician can warn the patient so an alternative could be though with the aim to aid in the goal progression. This feature could be useful to help clinicians acting in a timely manner.

The next topic will discuss the filtering of data by month, with information aggregated by average values from each day of the month.

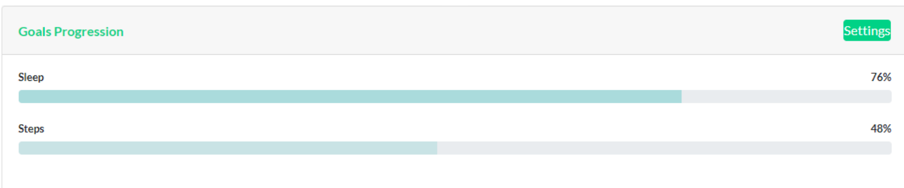


(a) Sleep report chart

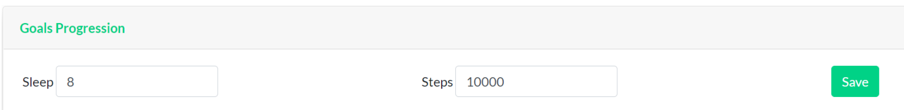


(b) Report settings

Figure 5.7: Last months sleep report chart example.



(a) Progress bar



(b) Goals settings

Figure 5.8: Goals progression example.

5.2.4 Visualization per Month

The data visualization filtered per month is a useful tool to better understand the patient evaluation in each month in a less granular level. This tool works as a historical comparison tool, capable of showing how each monitored input changed in each month.

The heart rate chart, visible on figure 5.9, show very minor oscillations throughout the months, with only the first two showing higher variations but still under the guideline. This information could be useful to the clinician when checking how well on average did the patient maintained his heart rate. A constant increase of the average heart rate on each month, could be a sign of constant increase in the levels of anxiety or stress, for example. Nevertheless, in this example the patient showed good heart rates indicators. This could also be very useful to understand the patient's baseline levels and with that information adjust the guideline value.

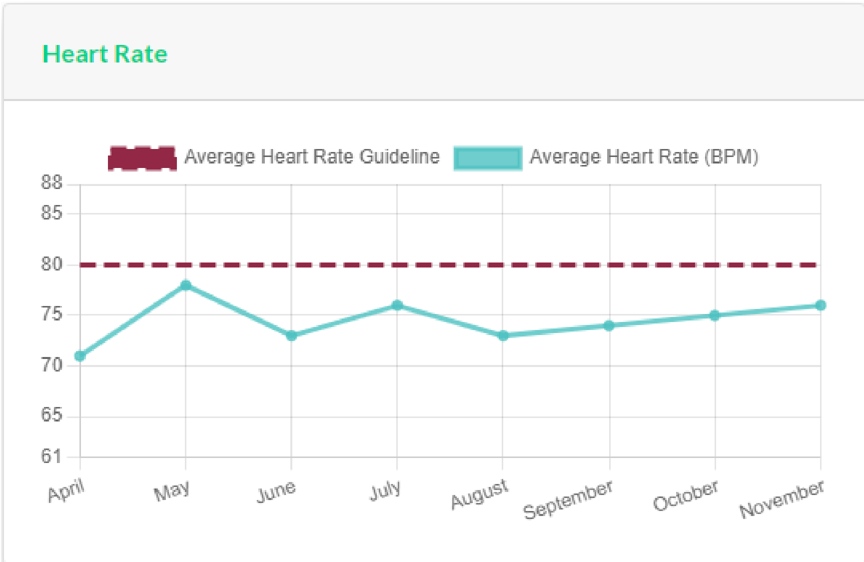


Figure 5.9: Average heart rate per month chart example.

The total amount of activity that the patient had on each month might also be very useful, since a sudden break on those indicators could be a warning that something drastically changed in the patient life and attention should be taken. Analysing figure 5.10 it is observable that the patient improved drastically from the beginning of the follow-up, going above the guideline. This could be viewed by the clinician as a strong indicator of positive patient progression. Nevertheless, July and August coincide with the holiday season so the data could be misleading without more complementary information, this highlights the clinician's precaution when analysing each tool. The noticeable decrease on activity after the month of September could act as a warning sign to the clinician and an e-mail could be sent to warn the clinician if the patient activity keeps declining. The clinician would then timely visualize this information on the platform and measures could be taken in order to improve this indicator.

Similarly the data related to the patient's sleep duration is useful to understand how

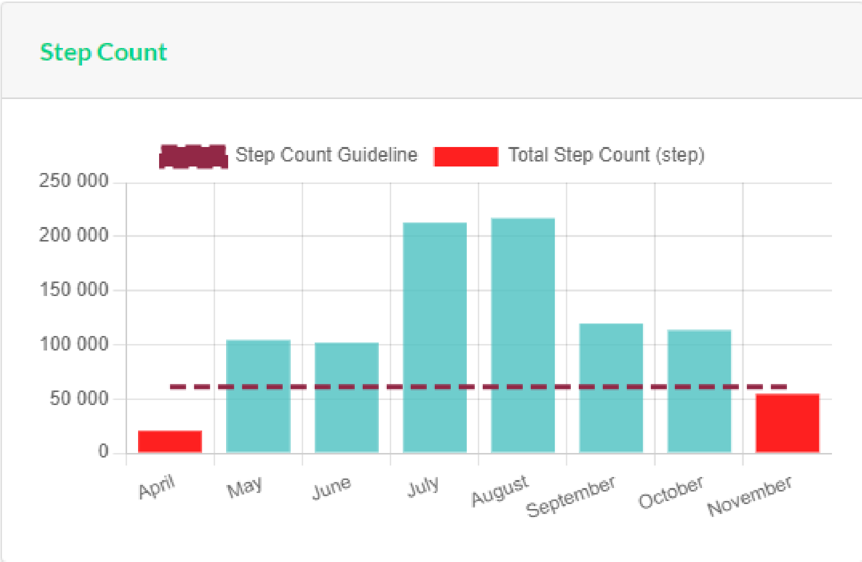


Figure 5.10: Total step count per month chart example.

the sleep quality is oscillating through each month. By observing the figure 5.11, the red point on September draws the attention but at the same it is easily noticeable that the negative situation is isolated and the patient improved significantly its sleep duration, therefore the clinician could conclude that most probably its quality indicator also improved. Once again this tool is useful, since a constant decrease on the patient sleep duration might be a strong indicator of health degradation, as it can be observed on figure 5.12.

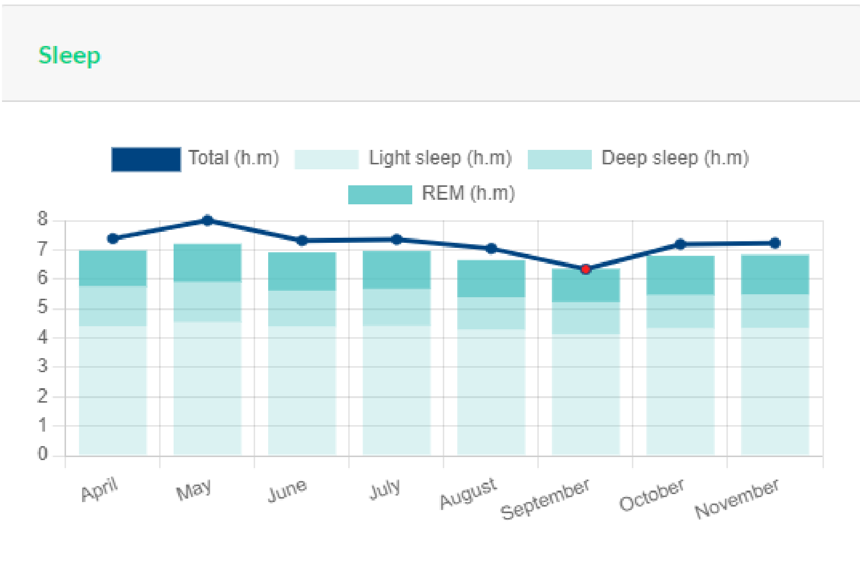


Figure 5.11: Average sleep duration per month chart example.

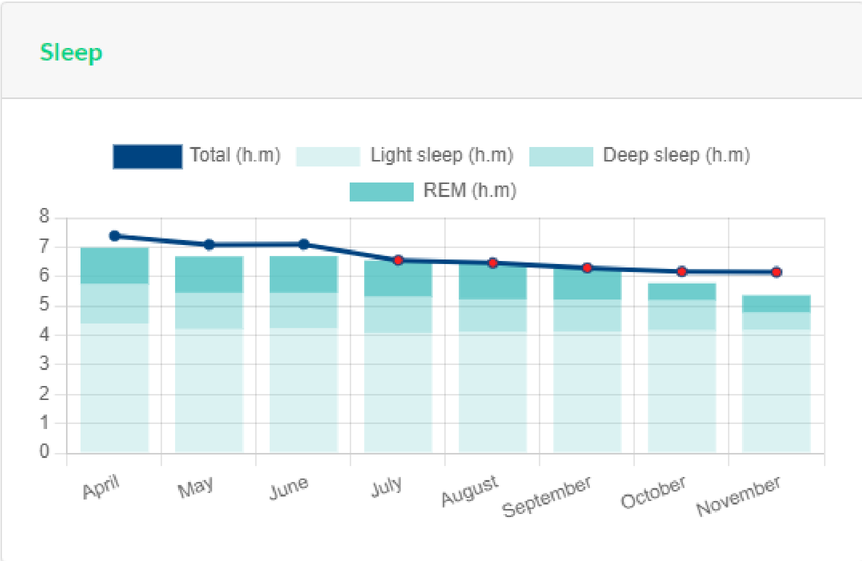


Figure 5.12: Average sleep duration declining per month chart example.

5.2.5 Visualization by Daily History

The daily filter shows all available data from the patient by each day from the beginning of the treatment and it is not expected to be used constantly. Nevertheless, it might be a helpful tool for very specific situation in which the clinicians might want to analyse days from three or more months from the present time.

This feature is mainly a time-lapse of the patients data and it should be taken in consideration in a real-scenario to better understand its utility. The following figures 5.13, 5.15, 5.14, 5.16 illustrated each charts by the daily filter. With the objective to reduce visual pollution and make data more clear to understand, some charts were not implemented with the red pinpoint observed when the value is under or above the guideline.

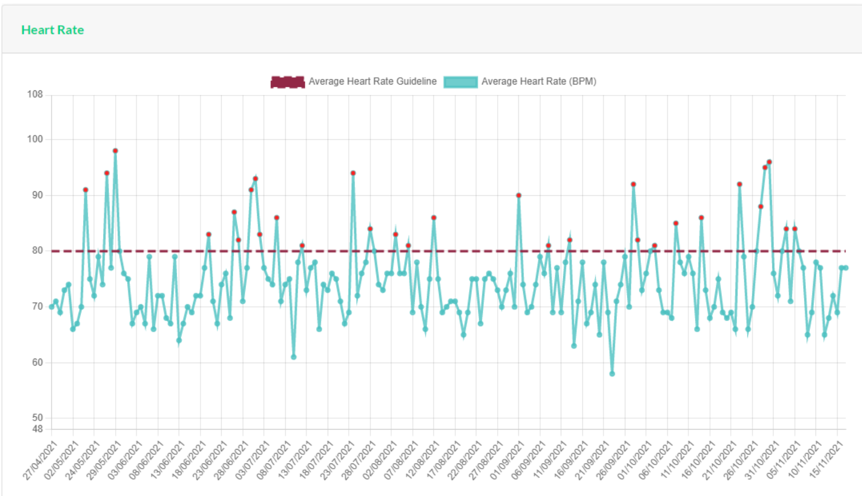


Figure 5.13: Daily history of heart rate data example.

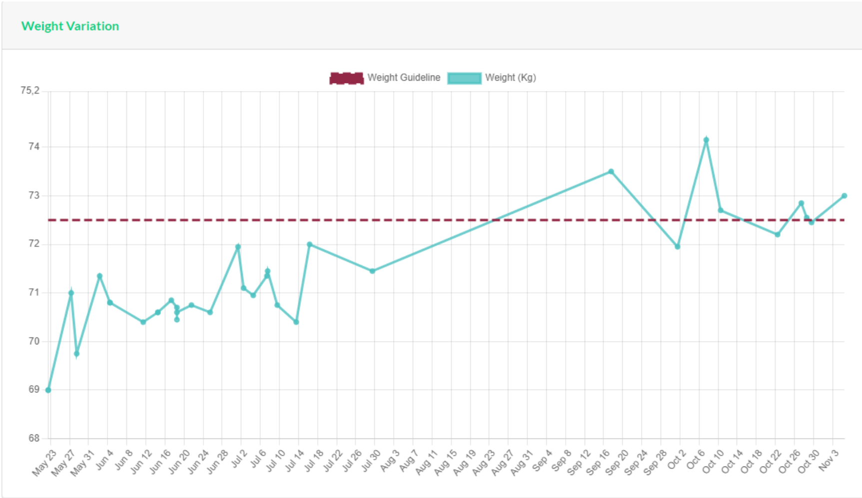


Figure 5.14: Daily history of weight data example.

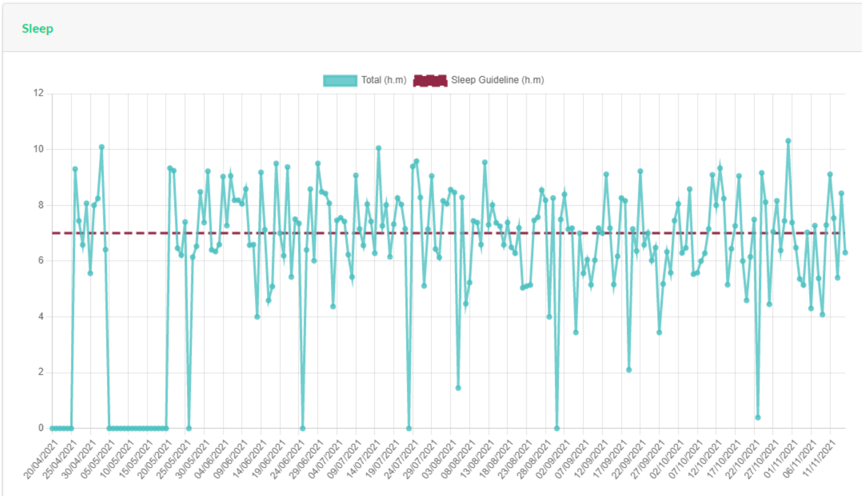


Figure 5.15: Daily history of sleep data example.

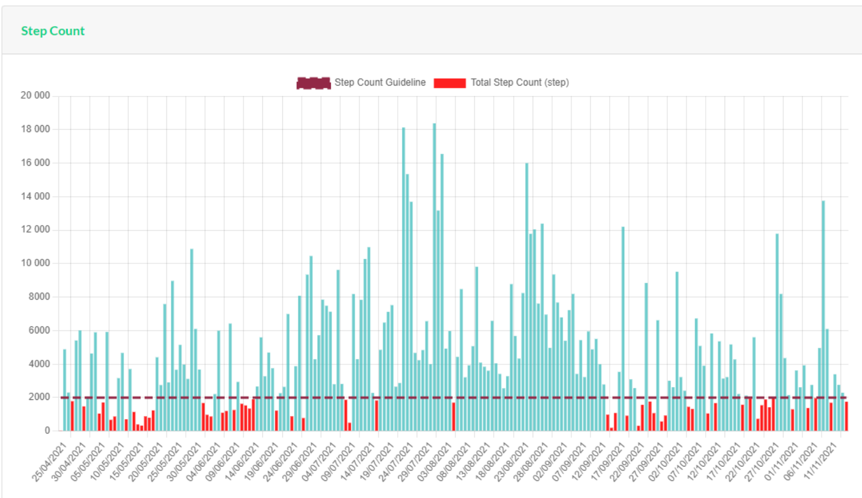


Figure 5.16: Daily history of step data example.

Finally, every chart from the charting library can show additional information when selecting a specific day, as illustrated on figure 5.17, displaying information regarding the day and values. This improves data granular analyse, since the clinician can select a specific day.

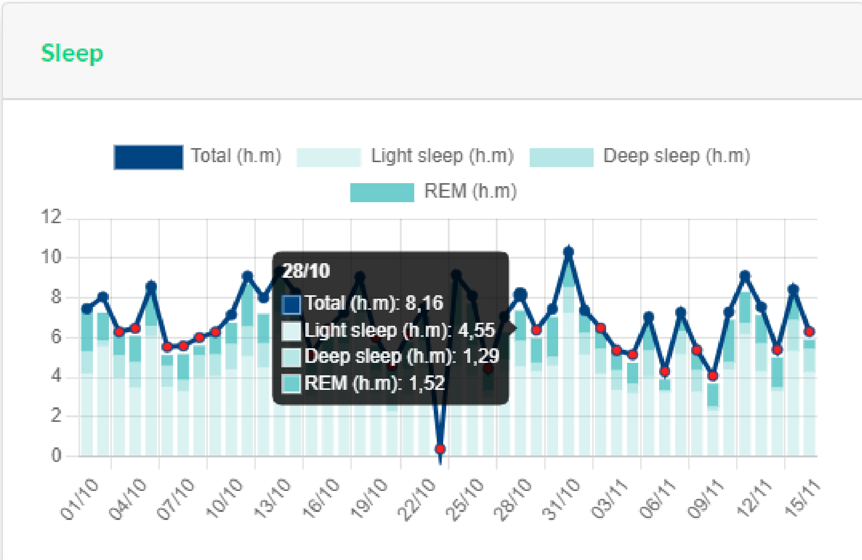


Figure 5.17: Additional sleep information overview.

5.3 Fuzzy Logic

The fuzzy logic outputted value is visualized as the PPI in the patient's dashboard, as depicted on figure 5.18. This indicator is obtained by the data from the last week averaging, so it is related to the patient's weekly progression.

Additionally, since each patient has its own baseline, as previously explained, the fuzzy logic algorithm will adjust its modules to the baseline. This is easily accomplished by feeding the fuzzy logic algorithm with each patient's baseline. This baseline normalization will ensure that the algorithm has in consideration which values are considered positive and negative.

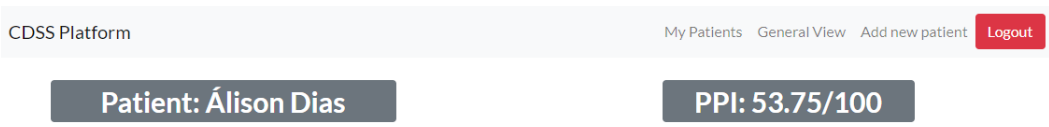


Figure 5.18: PPI value obtained with the Fuzzy Logic.

The following subsections will show fictitious results to a given input for the fuzzy logic algorithm and the resulting outputs. Those simulation will have in consideration a patient, whose baselines values are already normalized, in other words, the baselines coincide with each "normal"/"good"step membership function peak value. Furthermore, those results will be discussed to verify if there is a good performance having in account the prototype implementation and its inherent constrains.

5.3.1 Simulation Scenario 1

The first simulation will use the inputs defined in table 5.2 and the resulting if-then statements are resumed in table 5.3. Those values will simulate a scenario in which the patient has an overall positive measurements but has a lower sleep value.

Table 5.2: Simulation 1 input values.

Heart Rate	**Sleep**	**Δ Weight**	**Step Count**
70	5	0	3000

Table 5.3: Simulation 1 if-then rules for inference engine.

Heart Rate	**Sleep**	**Δ Weight**	**Step Count**	**Output**
1	0	0	0	Very Good(=1)
0	0	1	1	Good(=1)
0	1	0	0	Low(=1)
0	0	0	0	Very Low(=0)

The resulting table 5.3 is generated by analysing each input membership function and finding the degree of membership, figure 5.19 illustrate how the concept of finding in which step the input value belongs to, and then analyse which is the degree of membership by verifying the y-axis.

Analysing the heart rate membership function it is observable that the input is only member of the normal step and the degree is 1. So the degree of membership for medium, high and very high is 0. The same process is applied to the rest of the inputs.

The if-then statements is then applied with the OR operand, resulting in the following statements:

$$\boxed{\text{IF HR=Normal OR Sleep=High OR }\Delta\text{Weight=Very Good OR Step=Good}}$$

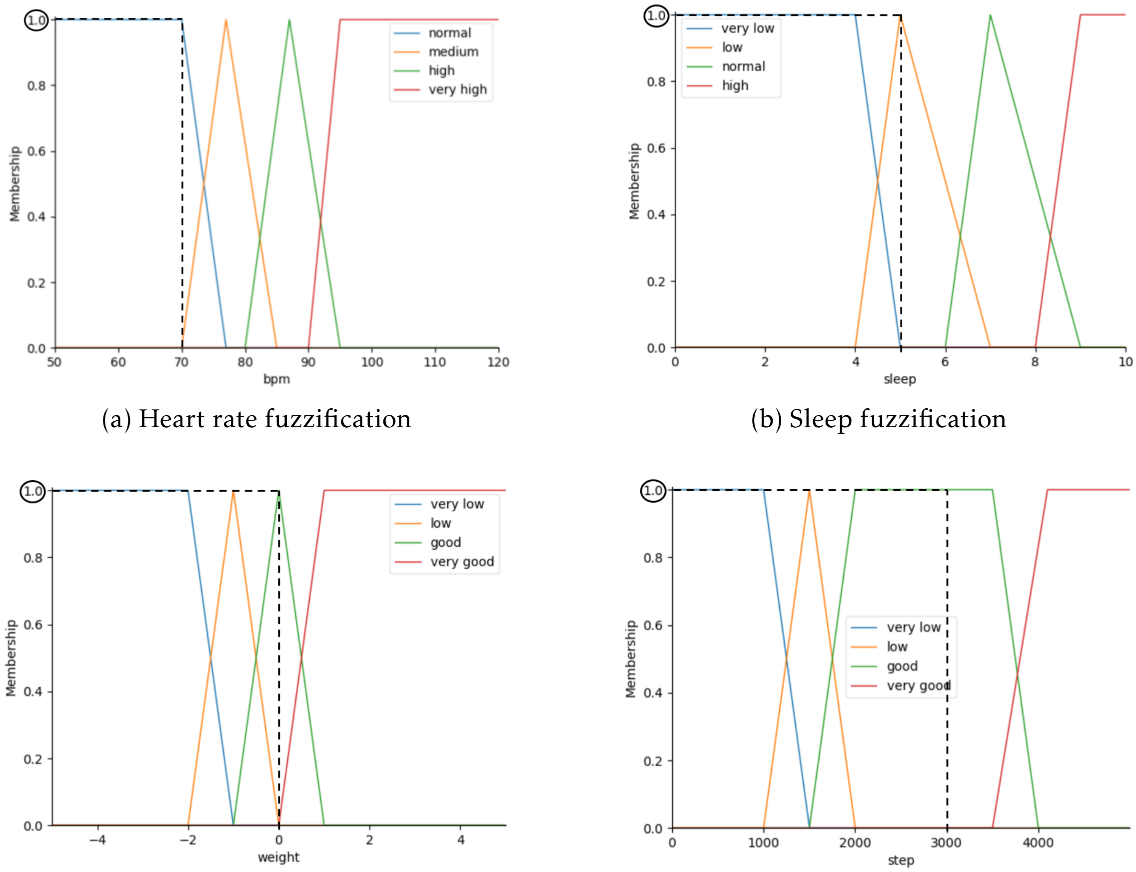
$$\boxed{\text{THEN Output=Very Good}}$$

The step in the statement is then substitute by the degree value of each linguistic term, resulting in:

$$\boxed{\text{IF HR=1 OR Sleep=0 OR }\Delta\text{Weight=0 OR Step=1}}$$

$$\boxed{\text{THEN Output=1}}$$

The resulting output is observed on figure 5.20, where it is observable that the linguistic terms low, good and very good have a degree of membership of 1, as determined in the if-then statements. Finally after the defuzzification process, center of gravity, the resulted output is 65,6 and his displayed as PPI.



(a) Heart rate fuzzification

(b) Sleep fuzzification

(c) Weight variation fuzzification

(d) Step fuzzification

Figure 5.19: Simulation 1 fuzzification.

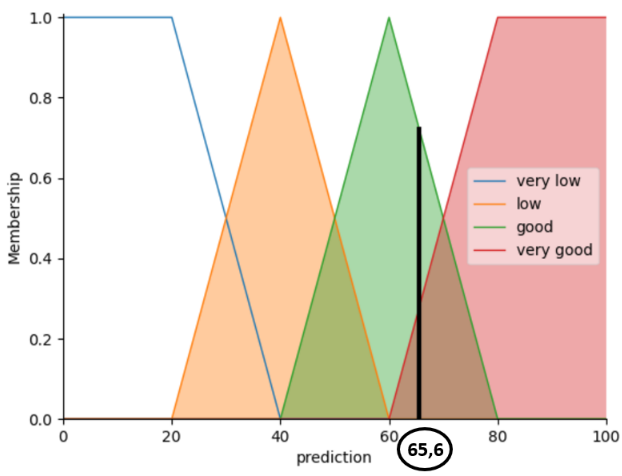


Figure 5.20: Simulation 1 fuzzy logic membership function output.

5.3.2 Simulation Scenario 2

The second simulation will consider a fictitious example of a patient measurements that indicate a negative progression and in comparison to the first simulation has a worst overall progression, since the heart rate in this case is higher and has an even lower sleep input.

The inputs are defined in table 5.4 and the resulting if-then statements are resumed in table 5.5.

Table 5.4: Simulation 2 input values.

Heart Rate	**Sleep**	**Δ Weight**	**Step Count**
82.35	4.5	0	3000

Table 5.5: Simulation 2 if-then rules for inference engine.

Heart Rate	**Sleep**	**Δ Weight**	**Step Count**	**Output**
0	0	0	0	Very Good(=0)
0,33	0	1	1	Good(=1)
0,33	0,5	0	0	Low(=0,5)
0	0,5	0	0	Very Low(=0,5)

The variations on inputs is significant and in figure 5.21 it is observable that both heart rate and sleep have quite different degree of membership from the last simulation, thus the resulting table 5.5 from the if-then statements processed differently and resulted in different outputs.

The if-then statements is then applied with the OR operand, resulting in the following example statement:

$$\boxed{\text{IF HR=High OR Sleep=Low OR }\Delta\text{Weight=Low OR Step=Low}}$$

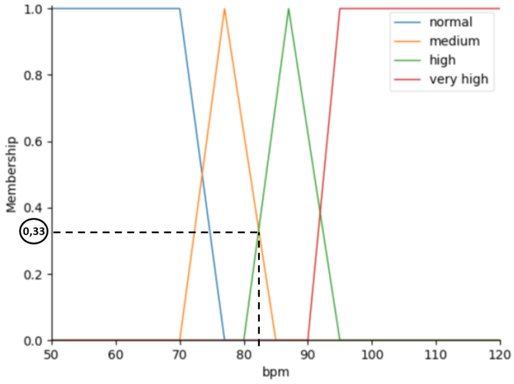
$$\boxed{\text{THEN Output=Low}}$$

The step in the statement is then substitute by the degree value of each linguistic term, resulting in:

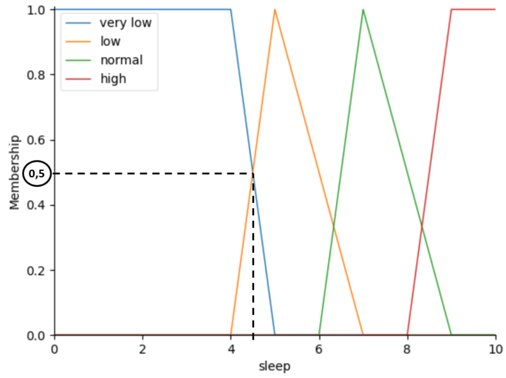
$$\boxed{\text{IF HR=0 OR Sleep=0,5 OR }\Delta\text{Weight=0 OR Step=0}}$$

$$\boxed{\text{THEN Output=0,5}}$$

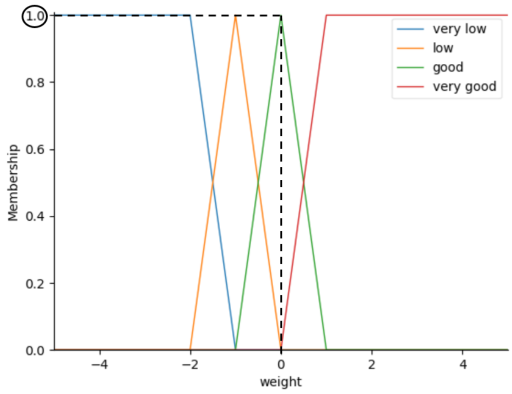
The resulting output is observed on figure 5.22, where it is observable that the linguistic terms good and very low have a degree of membership of 1 and 0,5 respectively. The low term has a degree of membership of 0,5, as determined in the if-then statements. Finally after the defuzzification process, using the center of gravity method, the resulted output is 40 and his displayed as PPI.



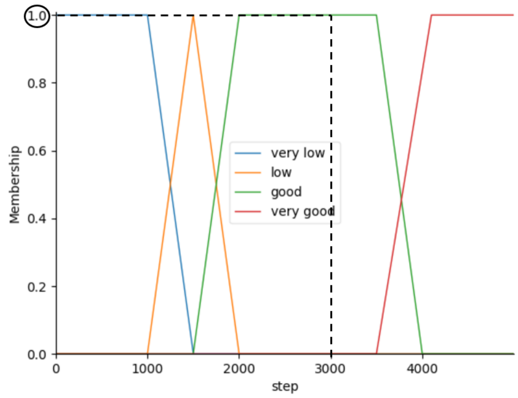
(a) Heart rate fuzzification



(b) Sleep fuzzification



(c) Weight variation fuzzification



(d) Step fuzzification

Figure 5.21: Simulation 2 fuzzification.

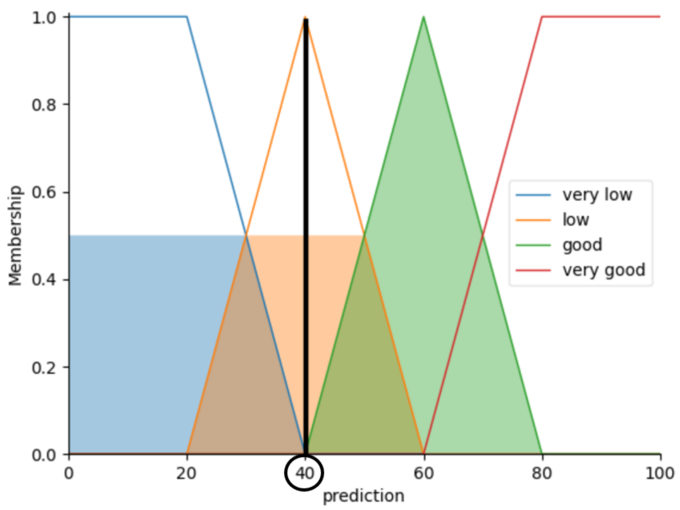


Figure 5.22: Simulation 2 fuzzy logic membership function output.

5.3.3 Simulation Scenario 3

Simulation 3 will consider an even worst case scenario then the previous two, in which the patient measurements indicate a negative progression of hear rate, sleep and step counts. In comparison with simulation 2, here both heart rate and sleep indicators remain the same, but the activity indicator lowers drastically to 1000 steps.

The inputs are defined in table 5.6 and the resulting if-then statements are resumed in table 5.7.

Table 5.6: Simulation 3 input values.

Heart Rate	**Sleep**	**Δ Weight**	**Step Count**
82.35	4.5	0	1000

Table 5.7: Simulation 3 if-then rules for inference engine.

Heart Rate	**Sleep**	**Δ Weight**	**Step Count**	**Output**
0	0	0	0	Very Good(=0)
0,33	0	1	0	Good(=1)
0,33	0,5	0	0	Low(=0,5)
0	0,5	0	1	Very Low(=1)

The variations on step count input is observable in figure 5.23 and the resulting output from the if-then statements can be found on table 5.7.

The if-then statements is then applied with the OR operand, resulting in the following example statement:

IF HR=Very High OR Sleep=Very Low OR ΔWeight=Very Low OR Step=Very Low

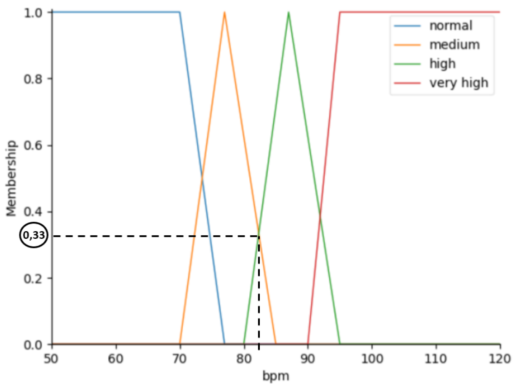
THEN Output=Very Low

The step in the statement is then substitute by the degree value of each linguistic term, resulting in:

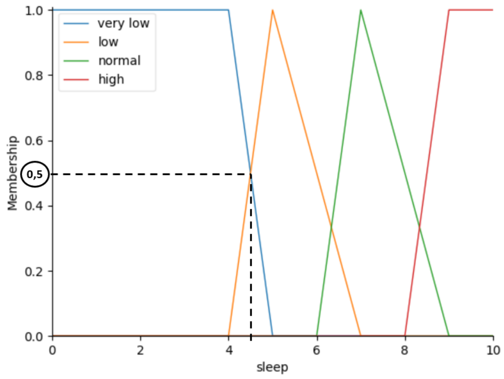
IF HR=0 OR Sleep=0,5 OR ΔWeight=0 OR Step=1

THEN Output=1

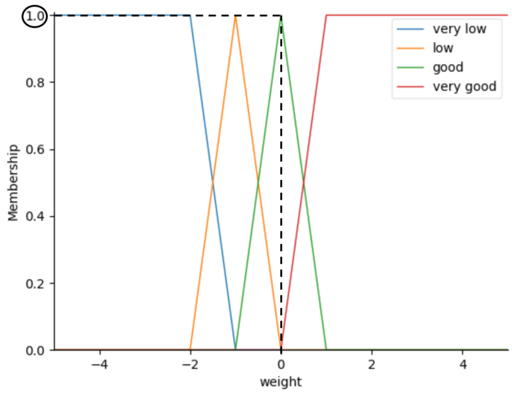
The resulting output is observed on figure 5.24, where it is observable that the linguistic terms low and good have a degree of membership of 0,5 and 1, respectively. The very low term has a degree of membership of 1, as determined in the if-then statement. Finally after the defuzzification process, using the center of gravity method, the resulted output is 33,9 and his displayed as PPI.



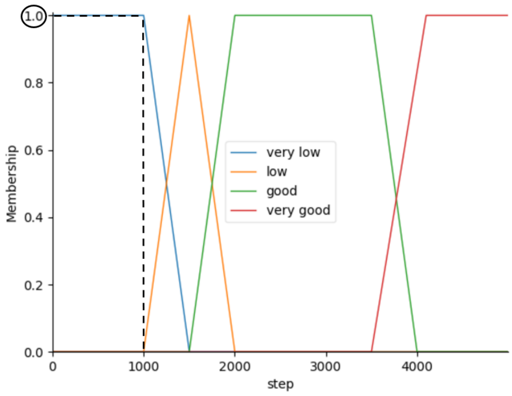
(a) Heart rate fuzzification



(b) Sleep fuzzification



(c) Weight variation fuzzification



(d) Step fuzzification

Figure 5.23: Simulation 3 fuzzification.

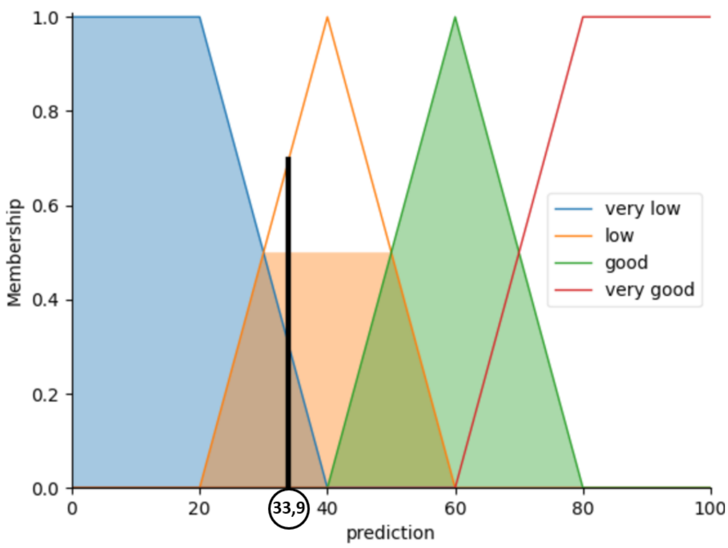


Figure 5.24: Simulation 3 fuzzy logic membership function output.

5.3.4 Simulation Results Discussion

The three simulations presented are able to give a better insight on how the fuzzy logic algorithm would react to those given inputs and although the algorithm was developed to be used as a prototype, for an initial starting stage the results show positive results.

The first scenario, simulation 1, resulted in an output PPI of 65,6 which seems to be a reasonable value, since the patient showed good health indicator's for the heart rate, weight variation and activity measured in steps. Nevertheless the sleep indicator was lower then the expected, thus the final output should not be that high. The result seem to be in line to a common sense evaluation, since it was expected a value higher then 50 indicating that the patient is above the limier of positive progression, but at the same time not too high due to the lack of sleep.

The following simulations 2 and 3 represented negative situations, since two or more patient's inputs indicate a bad progression, thus the expected output was an indicator lower then 50. The PPI value below 50 would show that the patient had a negative progression, since 50 is the limier of a positive progression.

Simulation 2 was expected to have a lower PPI then the previous, nevertheless the indicator should not be that low since the patient showed a good activity indicator. Hence, the resulting output of 40 seems to be a good estimation of patient's progress. The last simulation represented a critical patient progression, since heart rate levels were high, sleep levels were low and the activity indicator was low as well. Thus the expected result should be lower then the previous simulation 2, since the indicators were the same with the addition of the activity levels degradation. Hence, the resulting output of 33,9 seems to be in line with the expectations.

To sum up, the results obtained in each simulation showed the expected degradation as the simulation's bad indicators increased.

6 Conclusion

The results acquired throughout the whole CDSS Platform development and research will be summarized in this chapter. A reflection about its limitations will also be made, so that future works can take them in consideration, in order to improve and optimize this solution. Finally, some notes and requirements for future developments will be noticed so that more value can be extracted from the presented solution and some guidelines are defined.

6.1 General Conclusions

The purposed CDSS Platform had the goal to empower clinicians with technologies that are already commonly used and commercialized by people all around the world, and use it in an innovative way that aims to help in decision-making by enhancing the amount of collected information that will be readily provided to a clinician. This data can be gathered and made available with a small investment, especially if considered to have an added benefit for the user, since it provides clinicians with a better overview about the patient's daily life and physiological parameters.

As a main outcome from the proposed solution, it was proven that it is actually possible to use data that becomes readily available while gathered by a commercial wearable device. Thus, if the patient already has a wearable device compatible with the Google Fit application, data from the past becomes usable to provide insights about that person, as it becomes available for the clinician to observe and analyze. Furthermore, the baseline feature would benefit the most with such historical data, since more information would be available to create the guiding line on the visual tool. In a scenario where wearable devices become part of the life of most people, similarly to smartphones, this solution becomes even more relevant and more powerful. Nevertheless, if this scenario does not become true, the solution proved to be equally elegant and useful. This is due to the effortless integration between the wearable device, the mobile applications and CDSS Platform. The setup is rather straight forward, being only required for the patient to download the wearable device application and authorize data sharing with the Google Fit application. After that process, the clinician would send an invitation to the patient by e-mail so that he could authorize the CDSS Platform to use the monitored data. From that point on, the patient just needs to use the wearable device daily and make his normal routine, without any constrain. During this development of the present solution a wearable from Xiaomi, the Mi Band 5 was used to gather data becoming a source of real results. During the trial no problem was detected, beyond the need to charge it from time to time, since it could always be used, even when contact with water could happen because of its water resistance. This proves the elegance of the solution, since no major constrains were given to the patient and the whole setup process could be done with the clinician's aid. The usage of the smart scale device to monitor the patient's weight was another success, since its integration is effortless. This happens since the smart scale used was from the same brand as the wearable device, hence the mobile application would receive the data similarly to the way it receives data from the wearable device. In conclusion, if possible it is relevant to use devices from the same manufacturer so that no extra implementation becomes necessary. This is important to note, since if these devices were from different brands it would possibly require more integration and technical support, but feasible anyway.

The possibility to provide every patient with a continuous monitorization device, without the need to increase the number of appointments is also an advantage to be noted. The resulting solution provides clinicians with a monitorization tool that might warn, for example, patients with indicators that start to reduce drastically in a short amount of time. This could work as a predictor tool for clinicians, since it might allow a premature observation of a patient's reduction in Quality of Life (QoL). Although the platform does not measure the patient's QoL, it pinpoints indicators that might help in its analyses. It is relevant to notice that even if measurements provide relevant indicators on a patient's QoL, it does not substitute the clinicians experienced analysis, and is not intended to mislead the clinician's decision-making but rather give more data to support the clinician's evaluation process.

In what regards to the effectiveness and value of the visualization tools developed for the CDSS Platform, there is no definitive result, since the platform was not actively tested/used by clinical personal. Nevertheless, feedback was received during the development process to guide the development of the tools, its features and most relevant aspects, so that the resulting solution would become better suited for daily clinical usage. The conclusion that result from the present work ensure that clinical data can benefit from non-clinical data visualization, can surely increase the clinician's insight on the patients QoL, and although there is no actual evaluation of the resulting visualization solution, it seems reasonable to say that the tools give indeed new and useful information about the patient.

The tools where discussed with researchers and a psychologist from Champalimaud Foundation. According to their experience, the main important aspect is that the clinicians will be mostly interested in being able to observe changes in time and compare data from the patient, relate with baselines and visualize its progression, which is indeed possible with the developed visualization tools. Those results give more confidence about the usefulness and potential future use of the purposed Clinical Decision Support System (CDSS) including its application in a real clinical trial in a near future.

The CDSS Quality is guarantied as long as cooperation efforts are periodic and available, as section 2.2.1 refers, that the system quality is highly dependent to most up to date knowledge. Furthermore, each developed tool can be evaluated and tested in order to be certified, hence each tool can be supported by scientific data, and further improved.

Finally, it is important to note that the developed platform was design to be easily updated according to clinical feedback and requirements, as proven during the validation steps, and as already stated through this document the fuzzy logic algorithm has a great potential by allowing clinicians to easily tune its behavior according to requirements and specific needs. The feedback received from professionals was of major value for the relevance of the research and work performed so far with an opening interest for future developments and applications of the platform.

6.2 Future Works

The development of the presented solution faced a major limitation, which should be noted for future works. The lack of an available dataset gathered with the use of a wearable device from several patients, capable of monitoring the inputs used in the development of this dissertation and then its correct classification by an experienced clinician. Those aspects would help in the development of the designed fuzzy logic algorithm or even in the development of an artificial neural network or neural-fuzzy as covered in the state-of-the-art chapter 2.3.1.

The fuzzy logic algorithm would benefit from a classified dataset, since with this knowledge a better rule base could be implemented. The rule based would in that case be implemented with a reliable and real data, analysed and classified by a clinician, resulting in a much better algorithm performance.

A dataset also enables the use of neural networks or even neural-fuzzy, which would be an interesting perspective to research, since Artificial neural networks (ANNs) are powerful tools capable of finding patterns and learning from the dataset. The resulting model could then be evaluated on its capability of correlating different health indicators and QoL indicators. The researchers from Champalimaud Foundation referred that sleep routines could work well at being an indirect indicator of patient QoL.

The availability of the required dataset might not be as easy to obtain as expected, so in that scenario, future work should be done in this scope. The approach that is recommend would be to work in cooperation with one or more hospitals, in order to follow clinical appointments and use the approach designed in this dissertation. So, by providing patients with a smart band and a smart scale, data could be gathered and at the same time the clinician could classify the dataset. This research should be done for a large period of time, providing as much data as possible and ultimately creating a large quality dataset available.

The introduction of the purposed CDSS Platform on the previous example or in a parallel study would be a good complement to the research and an excellent way of understanding and improving the current solution. This is due to another limitation of this dissertation, the lack of feedback from an actual use case, in which the clinician could work with the platform and understand which features should be added, corrected or even eliminated. Thus, a future work that evaluates the performance of the current solution would be vital, in order to produce more knowledge in field and to better understand how similar CDSS should be designed.

Future developments with a similar approach from the described in this dissertation must have in mind the limitations inherent to integrating the platform with a third-party application, that is the Google Fit application and its services. The use of Google services boosted the development process and eased the design of the CDSS Platform, nevertheless a third-party application is always a component that creates uncertainty in term of reliability. For instances, the Google service could be made unavailable or deprecated which would let the whole platform useless, since it is such a key piece in the integration between the web application and the patient's smart devices. To surpass this problem, a mobile and web application could be developed from scratch, in order to replace the Google Fit Application and the Google Fit Representational State Transfer (REST) Application Programming Interface (API).

The development of this service would also enhance some aspects that the Google service is missing, since some data monitored by Xiaomi Mi band 5 is not available to be requested by the Google Fit REST API, such as nap information. Developing this component from scratch would allow the implementation of any needed data request or new feature that a new wearable device was capable to monitor.

Finally, the development of a mobile application focused on the patient experience might be an interesting addition to the purposed CDSS Platform architecture that could be studied. The mobile application could be used to complement information that is not available to be gathered with the use of smart devices, such as the sleep reports implemented and presented on section 5.2. The application could also be used to check the goals progress set by the clinician, or even check how well its overall progression is going and what could be improved. This feature could also be interesting to understand if a more visual awareness could influence the patient to act in order to improve its indicators. By creating a mobile application where the patient is presented with visual information about health indicators and is challenged with weekly goals, the patient might feel motivated to change his routines and observe the results.

Bibliography

[1] J. C. de Haes and F. C. van Knippenberg. "The quality of life of cancer patients: A review of the literature". In: *Social Science and Medicine* 20.8 (1985), pp. 809–817. issn: 02779536. doi: 10.1016/0277-9536(85)90335-1. url: https://www.sciencedirect.com/science/article/abs/pii/0277953685903351 (cit. on pp. 1, 3).

[2] D. Schapira. *What Comes After Finishing Treatment: An Expert Q&A | Cancer.Net*. 2020. url: https://www.cancer.net/survivorship/life-after-cancer/what-comes-after-finishing-treatment-expert-qa (visited on 11/24/2021) (cit. on p. 1).

[3] G. Shaw. *Breast Cancer Survivors: Life After the Treatments End*. url: https://www.webmd.com/breast-cancer/features/life-after-breast-cancer-treatment (visited on 11/24/2021) (cit. on p. 1).

[4] Who. *Cancer*. 2021. url: https://www.who.int/news-room/fact-sheets/detail/cancer (visited on 11/24/2021) (cit. on p. 1).

[5] T. Gaffney. *Report: Death rates are declining for many common cancers in U.S.* 2021. url: https://www.statnews.com/2021/07/08/cancer-death-rates-2021/ (visited on 11/24/2021) (cit. on p. 1).

[6] R. L. Siegel et al. "Cancer Statistics, 2021". In: *CA: A Cancer Journal for Clinicians* 71.1 (Jan. 2021), pp. 7–33. issn: 0007-9235. doi: 10.3322/CAAC.21654. url: https://www.cancer.org/latest-news/facts-and-figures-2021.html (cit. on p. 1).

[7] *. Heydarnejad, H. Dehkordi, and S. Dehkordi. "Factors affecting quality of life in cancer patients undergoing chemotherapy". In: *African Health Sciences* 11.2 (2011), pp. 266–270 (cit. on p. 1).

[8] H. M. Seidling et al. "Factors influencing alert acceptance: A novel approach for predicting the success of clinical decision support". In: *Journal of the American Medical Informatics Association* 18.4 (2011), pp. 479–484. issn: 10675027. doi:

10.1136/amiajnl-2010-000039. URL: https://pubmed.ncbi.nlm.nih.gov/21571746/ (cit. on pp. 1, 10, 11).

[9] K. Kawamoto et al. "Improving clinical practice using clinical decision support systems: A systematic review of trials to identify features critical to success". In: *British Medical Journal* 330.7494 (2005), pp. 765–768. ISSN: 09598146. DOI: 10.1136/bmj.38398.500764.8f. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC555881/ (cit. on pp. 2, 9, 10).

[10] E. A. VOGELS. *21% of Americans use a smart watch or fitness tracker | Pew Research Center*. 2020. URL: https://www.pewresearch.org/fact-tank/2020/01/09/about-one-in-five-americans-use-a-smart-watch-or-fitness-tracker/ (visited on 11/12/2021) (cit. on p. 2).

[11] L. S. Vailshery. *Fitness trackers - Statistics & Facts | Statista*. 2021. URL: https://www.statista.com/topics/4393/fitness-and-activity-tracker/%7B%5C#%7DdossierKeyfigures (visited on 11/12/2021) (cit. on p. 2).

[12] A. Bleicher. *Demystifying the Black Box That Is AI - Scientific American*. 2017. URL: https://www.scientificamerican.com/article/demystifying-the-black-box-that-is-ai/ (visited on 11/23/2021) (cit. on p. 3).

[13] B. Cloud. *The Difference Between White Box and Black Box AI - Big Cloud*. URL: https://bigcloud.global/the-difference-between-white-box-and-black-box-ai/ (visited on 11/23/2021) (cit. on p. 3).

[14] T. Automation. *The AI black box problem - ThinkAutomation*. URL: https://www.thinkautomation.com/bots-and-ai/the-ai-black-box-problem/ (visited on 11/23/2021) (cit. on p. 3).

[15] K. Donovan, R. W. Sanson-Fisher, and S. Redman. "Measuring quality of life in rectal cancer patients". In: *Journal of Clinical Oncology* 7.7 (1989), pp. 959–968. ISSN: 14737167. DOI: 10.1586/14737167.3.1.67. URL: https://ascopubs.org/doi/10.1200/JCO.1989.7.7.959?url%7B%5C_%7Dver=Z39.88-2003%7B%5C&%7Drfr%7B%5C_%7Did=ori%7B%5C%%7D3Arid%7B%5C%%7D3Acrossref.org%7B%5C&%7Drfr%7B%5C_%7Ddat=cr%7B%5C_%7Dpub++0pubmed%7B%5C&%7D (cit. on p. 3).

[16] K. C. Calman. "Quality of life in cancer patients - an hypothesis". In: *Journal of medical ethics* 10 (1984), pp. 124–127. ISSN: 14623935. URL: https://jme.bmj.com/content/10/3/124.short (cit. on p. 3).

[17] A. Bottomley. "The Cancer Patient and Quality of Life". In: *The Oncologist* 7.2 (2002), pp. 120–125. ISSN: 1083-7159. DOI: 10.1634/theoncologist.7-2-120. URL: https://theoncologist.onlinelibrary.wiley.com/doi/full/10.1634/theoncologist.7-2-120?sid=nlm%7B%5C%%7D3Apubmed (cit. on p. 3).

[18] E. Rincon et al. “Mobile phone apps for quality of life and well-being assessment in breast and prostate cancer patients: Systematic review”. In: *JMIR mHealth and uHealth* 5.12 (2017), pp. 1–13. ISSN: 22915222. DOI: 10.2196/mhealth.8741. URL: https://pubmed.ncbi.nlm.nih.gov/29203459/ (cit. on p. 4).

[19] C. Jongerius et al. “Research-Tested Mobile Apps for Breast Cancer Care: Systematic Review”. In: *JMIR mHealth and uHealth* 7.2 (2019), pp. 1–15. ISSN: 22915222. DOI: 10.2196/10930. URL: https://pubmed.ncbi.nlm.nih.gov/30741644/ (cit. on p. 4).

[20] C. Cannon. “Telehealth, Mobile Applications, and Wearable Devices are Expanding Cancer Care Beyond Walls”. In: *Seminars in Oncology Nursing* 34.2 (2018), pp. 118–125. ISSN: 07492081. DOI: 10.1016/j.soncn.2018.03.002. URL: https://pubmed.ncbi.nlm.nih.gov/29627143/ (cit. on p. 4).

[21] M. Liviu. “Comparative study on software development methodologies”. In: *Database Systems Journal* 4.3 (2014), p. 20. URL: https://dbjournal.ro/archive/17/17%7B%5C_%7D4.pdf (cit. on pp. 6, 29).

[22] M. D. S. Soares. “Comparação entre Metodologias Ágeis e Tradicionais para o Desenvolvimento de Software”. In: *INFOCOMP Journal of Computer Science* 27.2 (2003), p. 6. ISSN: 0718-3429. URL: https://www.researchgate.net/publication/228931892%7B%5C_%7DComparacao%7B%5C_%7Dentre%7B%5C_%7Dmetodologias%7B%5C_%7DAgeis%7B%5C_%7De%7B%5C_%7Dtradicionais%7B%5C_%7Dpara%7B%5C_%7Do%7B%5C_%7Ddesenvolvimento%7B%5C_%7Dde%7B%5C_%7Dsoftware (cit. on pp. 6, 7, 29).

[23] G. Papadopoulos. “Moving from Traditional to Agile Software Development Methodologies Also on Large, Distributed Projects.” In: *Procedia - Social and Behavioral Sciences* 175 (2015), pp. 455–463. ISSN: 18770428. DOI: 10.1016/j.sbspro.2015.01.1223. URL: http://dx.doi.org/10.1016/j.sbspro.2015.01.1223 (cit. on p. 6).

[24] A. B. M. Moniruzzaman and D. S. A. Hossain. “Comparative Study on Agile software development methodologies”. In: V.3 (2013), pp. 37–56. ISSN: 2069-3230. arXiv: 1307.3356. URL: http://arxiv.org/abs/1307.3356 (cit. on pp. 6, 7, 29).

[25] I. Sim et al. “Clinical decision support systems for the practice of evidence-based medicine”. In: *Journal of the American Medical Informatics Association* 8.6 (2001), pp. 527–534. ISSN: 10675027. DOI: 10.1136/jamia.2001.0080527. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC130063/ (cit. on p. 8).

[26] M. W. Jaspers et al. “Effects of clinical decision-support systems on practitioner performance and patient outcomes: A synthesis of high-quality systematic review findings”. In: *Journal of the American Medical Informatics Association* 18.3 (2011), pp. 327–334. ISSN: 10675027. DOI: 10.1136/amiajnl-2011-000094. URL: https://pubmed.ncbi.nlm.nih.gov/21422100/ (cit. on p. 8).

[27] N. Skyttberg et al. "How to improve vital sign data quality for use in clinical decision support systems? A qualitative study in nine Swedish emergency departments". In: *BMC Medical Informatics and Decision Making* 16.1 (2016), pp. 1–12. ISSN: 14726947. DOI: 10.1186/s12911-016-0305-4. URL: http://dx.doi.org/10.1186/s12911-016-0305-4 (cit. on p. 8).

[28] R. W. Brause. "Medical analysis and diagnosis by neural networks". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 2199 (2001), pp. 1–13. ISSN: 16113349. DOI: 10.1007/3-540-45497-7_1. URL: https://link.springer.com/chapter/10.1007/3-540-45497-7%7B%5C_%7D1 (cit. on pp. 10, 13–16).

[29] D. F. Sittig et al. "A survey of factors affecting clinician acceptance of clinical decision support". In: *BMC Medical Informatics and Decision Making* 6 (2006), pp. 1–7. ISSN: 14726947. DOI: 10.1186/1472-6947-6-6. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1403751/ (cit. on pp. 11, 12).

[30] E. S. Berner et al. "Clinician performance and prominence of diagnoses displayed by a clinical diagnostic decision support system." In: *AMIA ... Annual Symposium proceedings / AMIA Symposium. AMIA Symposium* (2003), pp. 76–80. ISSN: 15594076. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1480080/ (cit. on p. 12).

[31] I. C. Education. *What is Artificial Intelligence (AI)? | IBM*. 2020. URL: https://www.ibm.com/cloud/learn/what-is-artificial-intelligence (visited on 11/19/2021) (cit. on p. 13).

[32] F. Amato et al. "Artificial neural networks in medical diagnosis". In: *Journal of Applied Biomedicine* 11.2 (2013), pp. 47–58. ISSN: 12140287. DOI: 10.2478/v10136-012-0031-x. URL: https://www.sciencedirect.com/science/article/abs/pii/S1214021X14600570 (cit. on pp. 14, 15).

[33] N. DINGLE. *Control Engineering | Artificial Intelligence: Fuzzy Logic Explained*. 2011. URL: https://www.controleng.com/articles/artificial-intelligence-fuzzy-logic-explained/ (visited on 11/19/2021) (cit. on p. 17).

[34] W. Chai. *What is fuzzy logic? - Definition from WhatIs.com*. 2021. URL: https://searchenterpriseai.techtarget.com/definition/fuzzy-logic (visited on 11/19/2021) (cit. on p. 17).

[35] GeeksforGeeks. *Fuzzy Logic | Introduction - GeeksforGeeks*. 2021. URL: https://www.geeksforgeeks.org/fuzzy-logic-introduction/ (visited on 11/19/2021) (cit. on p. 17).

[36] M. Mahfouf, M. F. Abbod, and D. A. Linkens. "A survey of fuzzy logic monitoring and control utilisation in medicine". In: () (cit. on p. 17).

[37] D. Trehan. *Clustering : What it is? When to use it? – Towards AI — The World's Leading AI and Technology Publication.* 2020. URL: https://towardsai.net/p/machine-learning/clustering-what-it-is-when-to-use-it (visited on 11/20/2021) (cit. on p. 17).

[38] S. L. Chiu. "Extracting fuzzy rules from data for function approximation and pattern classification". In: *Fuzzy Information Engineering – A guided tour of applications* February (1997), p. 10 (cit. on pp. 17, 41).

[39] R. Garcia-Retamero and U. Hoffrage. "Visual representation of statistical information improves diagnostic inferences in doctors and their patients". In: *Social Science and Medicine* 83 (2013), pp. 27–33. ISSN: 02779536. DOI: 10.1016/j.socscimed.2013.01.034. URL: http://dx.doi.org/10.1016/j.socscimed.2013.01.034 (cit. on p. 17).

[40] J. B. Lamy et al. "An iconic language for the graphical representation of medical concepts". In: *BMC Medical Informatics and Decision Making* 8 (2008), pp. 1–12. ISSN: 14726947. DOI: 10.1186/1472-6947-8-16. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2413217/ (cit. on p. 17).

[41] J. STARREN and S. B. JOHNSON. "An Object-oriented Taxonomy of Medical Data Presentations". In: *Journal of the American Medical Informatics Association* 7.1 (2000), pp. 1–20. DOI: 10.1136/jamia.2000.0070001. URL: https://pubmed.ncbi.nlm.nih.gov/10641959/ (cit. on p. 18).

[42] F. Andry et al. "Data visualization in a personal health record using rich internet application graphic components". In: *HEALTHINF 2009 - Proceedings of the 2nd International Conference on Health Informatics* (2009), pp. 111–116. URL: https://www.scitepress.org/papers/2009/13788/13788.pdf (cit. on p. 18).

[43] S. P. Duke et al. "Seeing is believing: Good graphic design principles for medical research". In: *Statistics in Medicine* 34.22 (2015), pp. 3040–3059. ISSN: 10970258. DOI: 10.1002/sim.6549. URL: https://www.researchgate.net/publication/279302841%7B%5C_%7DSeeing%7B%5C_%7Dis%7B%5C_%7Dbelieving%7B%5C_%7DGood%7B%5C_%7Dgraphic%7B%5C_%7Ddesign%7B%5C_%7Dprinciples%7B%5C_%7Dfor%7B%5C_%7Dmedical%7B%5C_%7Dresearch (cit. on pp. 18, 20).

[44] S. M. Ali et al. "Big data visualization: Tools and challenges". In: *Proceedings of the 2016 2nd International Conference on Contemporary Computing and Informatics, IC3I 2016* (2016), pp. 656–660. DOI: 10.1109/IC3I.2016.7918044. URL: https://ieeexplore.ieee.org/document/7918044 (cit. on p. 21).

[45] J. H. Park et al. "Determinants of quality of life in women immediately following the completion of primary treatment of breast cancer: A cross-sectional study". In: *PLoS ONE* 16.10 October (2021), pp. 1–13. ISSN: 19326203. DOI: 10.1371/journal.pone.0258447. URL: http://dx.doi.org/10.1371/journal.pone.0258447 (cit. on p. 22).

[46] M. S. Anker et al. "Increased resting heart rate and prognosis in treatment-naïve unselected cancer patients: results from a prospective observational study". In: *European Journal of Heart Failure* 22.7 (July 2020), pp. 1230–1238. ISSN: 18790844. DOI: 10.1002/EJHF.1782/FORMAT/PDF (cit. on p. 22).

[47] M. S. Anker et al. "Resting heart rate is an independent predictor of death in patients with colorectal, pancreatic, and non-small cell lung cancer: results of a prospective cardiovascular long-term study". In: *European Journal of Heart Failure* 18.12 (Dec. 2016), pp. 1524–1534. ISSN: 1879-0844. DOI: 10.1002/EJHF.670. URL: https://onlinelibrary.wiley.com/doi/full/10.1002/ejhf.670%20https://onlinelibrary.wiley.com/doi/abs/10.1002/ejhf.670%20https://onlinelibrary.wiley.com/doi/10.1002/ejhf.670 (cit. on p. 22).

[48] S. E. Jackson et al. "The impact of a cancer diagnosis on weight change: findings from prospective, population-based cohorts in the UK and the US". In: (2014). DOI: 10.1186/1471-2407-14-926. URL: https://ssl.isr.umich.edu/hrs/ (cit. on p. 23).

[49] B. J. Caan et al. "Cancer Causes Control". In: 19.10 (2008), pp. 1319–1328. DOI: 10.1007/s10552-008-9203-0 (cit. on p. 23).

[50] B. J. Caan et al. "Weight Change and Survival after Breast Cancer in the After Breast Cancer Pooling Project". In: (). DOI: 10.1158/1055-9965.EPI-12-0306 (cit. on p. 23).

[51] T. M. Liska and A. M. Kolen. "The role of physical activity in cancer survivors' quality of life". In: 18 (2020), p. 197. DOI: 10.1186/s12955-020-01448-3. URL: http://creativecommons.org/licenses/by/4.0/.TheCreativeCommonsPublicDomainDedicationwaiv 20http://creativecommons.org/publicdomain/zero/1.0/ (cit. on p. 23).

[52] B. C. Bade et al. "Assessing the Correlation Between Physical Activity and Quality of Life in Advanced Lung Cancer". In: *Integrative Cancer Therapies* 17.1 (Mar. 2018), p. 73. ISSN: 1552695X. DOI: 10.1177/1534735416684016. URL: /pmc/articles/PMC5647199/%20/pmc/articles/PMC5647199/?report=abstract%20https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5647199/ (cit. on p. 23).

[53] G. Gresham et al. "Wearable activity monitors to assess performance status and predict clinical outcomes in advanced cancer patients". In: *npj Digital Medicine* 1.1 (2018), pp. 1–8. ISSN: 2398-6352. DOI: 10.1038/s41746-018-0032-6. URL: http://dx.doi.org/10.1038/s41746-018-0032-6 (cit. on pp. 23, 26).

[54] C. Rafie et al. "Impact of physical activity and sleep quality on quality of life of rural residents with and without a history of cancer: findings of the Day and night study". In: *Cancer Management and Research* (2018), pp. 10–5525. DOI: 10.2147/CMAR.S160481. URL: http://dx.doi.org/10.2147/CMAR.S160481 (cit. on p. 23).

[55] N. S. Gooneratne et al. "Sleep and quality of life in long-term lung cancer survivors". In: () (cit. on p. 24).

[56] A. Tamakoshi and Y. Ohno. "Self-reported sleep duration as a predictor of all-cause mortality: Results from the JACC Study, Japan". In: *Sleep* 27.1 (2004), pp. 51–54. ISSN: 01618105. DOI: 10.1093/sleep/27.1.51 (cit. on p. 24).

[57] J. E. Ferrie et al. "A Prospective Study of Change in Sleep Duration: Associations with Mortality in the Whitehall II Cohort". In: *Sleep* 30.12 (Dec. 2007), p. 1659. ISSN: 01618105. DOI: 10.1093/SLEEP/30.12.1659. URL: /pmc/articles/PMC2276139/%20/pmc/articles/PMC2276139/?report=abstract%20https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2276139/ (cit. on p. 24).

[58] R. A. Joundi et al. "Rapid tremor frequency assessment with the iPhone accelerometer". In: *Parkinsonism and Related Disorders* 17.4 (2011), pp. 288–290. ISSN: 13538020. DOI: 10.1016/j.parkreldis.2011.01.001. URL: https://pubmed.ncbi.nlm.nih.gov/21300563/ (cit. on p. 24).

[59] C. Worringham, A. Rojek, and I. Stewart. "Development and feasibility of a smartphone, ECG and GPS based system for remotely monitoring exercise in cardiac rehabilitation". In: *PLoS ONE* 6.2 (2011). ISSN: 19326203. DOI: 10.1371/journal.pone.0014669. URL: https://pubmed.ncbi.nlm.nih.gov/21347403/ (cit. on p. 24).

[60] A. Marshall, O. Medvedev, and A. Antonov. "Use of a smartphone for improved self-management of pulmonary rehabilitation". In: *International Journal of Telemedicine and Applications* 2008 (2008), pp. 8–12. ISSN: 16876415. DOI: 10.1155/2008/753064 (cit. on p. 25).

[61] C. A. Low. "Harnessing consumer smartphone and wearable sensors for clinical cancer research". In: *npj Digital Medicine* 3.1 (2020). ISSN: 23986352. DOI: 10.1038/s41746-020-00351-x (cit. on p. 25).

[62] S. Asensio-Cuesta et al. "Smartphone sensors for monitoring cancer-related quality of life: App design, EORTC QLQ-C30 mapping and feasibility study in healthy subjects". In: *International Journal of Environmental Research and Public Health* 16.3 (2019). ISSN: 16604601. DOI: 10.3390/ijerph16030461 (cit. on p. 26).

[63] Flaticon. *Free Vector Icons and Stickers - PNG, SVG, EPS, PSD and CSS*. 2021. URL: https://www.flaticon.com/ (visited on 10/09/2021) (cit. on pp. 31, 33, 36).

[64] M. Rimol. *Gartner Forecasts Global Spending on Wearable Devices to Total $81.5 Billion in 2021*. 2021. URL: https://www.gartner.com/en/newsroom/press-releases/2021-01-11-gartner-forecasts-global-spending-on-wearable-devices-to-total-81-5-billion-in-2021 (visited on 11/20/2021) (cit. on p. 32).

[65] Xiaomi. *Mi Global Home*. URL: https://www.mi.com/global/mi-smart-band-5/ (visited on 11/21/2021) (cit. on p. 32).

[66] Xiaomi. *Mi Global Home*. URL: https://www.mi.com/global/mi-body-composition-scale/ (visited on 11/21/2021) (cit. on p. 32).

[67] Google. *Frequently Asked Questions (FAQ) | Google Fit | Google Developers*. URL: https://developers.google.com/fit/faq (visited on 11/21/2021) (cit. on p. 32).

[68] W. Guides. *How Does Google Fit Track Steps? | Wearable Guides*. 2021. URL: https://wearableguides.com/how-does-google-fit-track-steps/ (visited on 11/21/2021) (cit. on p. 32).

[69] Npm. *npm*. 2021. URL: https://www.npmjs.com/ (visited on 10/09/2021) (cit. on pp. 34, 36).

[70] Npm. *@hapi/joi - npm*. URL: https://www.npmjs.com/package/@hapi/joi (visited on 11/22/2021) (cit. on p. 34).

[71] Npm. *mongoose - npm*. URL: https://www.npmjs.com/package/mongoose (visited on 11/22/2021) (cit. on p. 34).

[72] Npm. *bcrypt - npm*. URL: https://www.npmjs.com/package/bcrypt (visited on 11/22/2021) (cit. on p. 34).

[73] Npm. *jsonwebtoken - npm*. URL: https://www.npmjs.com/package/jsonwebtoken (visited on 11/22/2021) (cit. on p. 34).

[74] Npm. *extract-zip - npm*. URL: https://www.npmjs.com/package/extract-zip (visited on 11/22/2021) (cit. on p. 34).

[75] Npm. *multer - npm*. URL: https://www.npmjs.com/package/multer (visited on 11/22/2021) (cit. on p. 34).

[76] Npm. *axios - npm*. URL: https://www.npmjs.com/package/axios (visited on 11/22/2021) (cit. on p. 34).

[77] Npm. *cookie-parser - npm*. URL: https://www.npmjs.com/package/cookie-parser (visited on 11/22/2021) (cit. on p. 34).

[78] Npm. *dotenv - npm*. URL: https://www.npmjs.com/package/dotenv (visited on 11/22/2021) (cit. on p. 34).

[79] Npm. *googleapis - npm*. URL: https://www.npmjs.com/package/googleapis (visited on 11/22/2021) (cit. on p. 34).

[80] Npm. *json-file-loader - npm*. URL: https://www.npmjs.com/package/json-file-loader (visited on 11/22/2021) (cit. on p. 34).

[81] Npm. *node-cron - npm*. URL: https://www.npmjs.com/package/node-cron (visited on 11/22/2021) (cit. on p. 34).

[82] Npm. *mkdirp - npm*. URL: https://www.npmjs.com/package/mkdirp (visited on 11/22/2021) (cit. on p. 34).

[83] Npm. *nodemailer - npm*. URL: https://www.npmjs.com/package/nodemailer (visited on 11/22/2021) (cit. on p. 34).

[84] U. Pisuwala. *Node.js vs Django vs Laravel: Qual é a melhor estrutura back-end da web em 2021?* 2021. URL: https://www.peerbits.com/blog/nodejs-vs-django-vs-laravel-back-end-web-framework.html (visited on 11/19/2021) (cit. on p. 35).

[85] C. Gor. *Laravel vs Django vs Node.JS : Which Is The Best Back-End Technology?* 2021. URL: https://www.esparkinfo.com/laravel-vs-django-vs-nodejs.html (visited on 11/19/2021) (cit. on p. 35).

[86] Dhruvpatel. *Which One is Most Demanding Back-End Web Framework between Laravel , Node.js and Django ? - GeeksforGeeks*. 2021. URL: https://www.geeksforgeeks.org/which-one-is-most-demanding-back-end-web-framework-between-laravel-node-js-and-django/ (visited on 11/19/2021) (cit. on p. 35).

[87] MongoDB. *MongoDB Atlas: Cloud Document Database | MongoDB*. 2021. URL: https://www.mongodb.com/cloud/atlas/lp/try2?utm%7B%5C_%7Dcontent=controlhterms%7B%5C&%7Dutm%7B%5C_%7Dsource=google%7B%5C&%7Dutm%7B%5C_%7Dcampaign=gs%7B%5C_%7Demea%7B%5C_%7Dportugal%7B%5C_%7Dsearch%7B%5C_%7Dcore%7B%5C_%7Dbrand%7B%5C_%7Datlas%7B%5C_%7Ddesktop%7B%5C&%7Dutm%7B%5C_%7Dterm=mongodb%7B%5C&%7Dutm%7B%5C_%7Dmedium=cpc%7B%5C_%7Dpaid%7B%5C_%7Dsearch%7B%5C&%7Dutm%7B%5C_%7Dad=e%7B%5C&%7Dutm%7B%5C_%7Dad%7B%5C_%7Dcampaign%7B%5C_%7Did=12212624551%7B%5C&%7Dgclid=CjwKCAjw2P-KBhB (visited on 10/09/2021) (cit. on p. 36).

[88] EJS. *EJS – Embedded JavaScript templates*. 2021. URL: https://ejs.co/ (visited on 10/09/2021) (cit. on p. 36).

[89] Bootstrap. *Bootstrap · The most popular HTML, CSS, and JS library in the world.* 2021. URL: https://getbootstrap.com/ (visited on 10/09/2021) (cit. on p. 36).

[90] D. Ibrahim. "An Overview of Soft Computing". In: *Procedia Computer Science* 102 (2016), pp. 34–38. DOI: 10.1016/J.PROCS.2016.09.366 (cit. on p. 39).

[91] Onesmus Mbaabu. *An Overview of Fuzzy Logic System | Engineering Education (EngEd) Program | Section*. URL: https://www.section.io/engineering-education/an-overview-of-fuzzy-logic-system/ (visited on 10/16/2021) (cit. on p. 39).

[92] Tutorialspoint. *Artificial Intelligence - Fuzzy Logic Systems*. URL: https://www.tutorialspoint.com/artificial%7B%5C_%7Dintelligence/artificial%7B%5C_%7Dintelligence%7B%5C_%7Dfuzzy%7B%5C_%7Dlogic%7B%5C_%7Dsystems.htm (visited on 10/16/2021) (cit. on p. 39).

[93] W3schools. *Introduction to NumPy*. URL: https://www.w3schools.com/python/numpy/numpy%7B%5C_%7Dintro.asp (visited on 10/19/2021) (cit. on p. 53).

[94] W3schools. *Matplotlib Pyplot*. URL: https://www.w3schools.com/python/matplotlib%7B%5C_%7Dpyplot.asp (visited on 10/19/2021) (cit. on p. 53).

[95] S. fuzzy. *SciKit-Fuzzy — skfuzzy v0.2 docs*. URL: https://pythonhosted.org/scikit-fuzzy/overview.html (visited on 10/19/2021) (cit. on p. 53).