Universidade de Aveiro
2021

Vasco Rodrigues Cardoso

Plataforma Computacional para Análise de Dados Afetivos Multimodais.

Computational Platform for Multimodal Affective Data Analysis.

Universidade de Aveiro
2021

Vasco Rodrigues Cardoso

Plataforma Computacional para Análise de Dados Afetivos Multimodais.

Computational Platform for Multimodal Affective Data Analysis.

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia de Computadores e Telemática, realizada sob a orientação científica do Doutor Ilídio Castro Oliveira, Professor Auxiliar do Departamento de Electrónica, Telecomunicações e informática da Universidade de Aveiro, e da Doutora Susana Manuela Martinho dos Santos Baía Brás, Investigadora do Instituto de Engenharia Eletrónica e Informática de Aveiro, da Universidade de Aveiro.

Dedico este trabalho aos meus pais.

o júri / the jury

presidente / president

Professor Doutor António Joaquim da Silva Teixeira
Professor Associado C/ Agregação do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro

vogais / examiners committee

Doutor Sérgio Miguel Fernandes dos Santos
Engenheiro de Algoritmos Sénior, Talkdesk

Professor Doutor Ilídio Fernando de Castro Oliveira
Professor Auxiliar do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro

agradecimentos / acknowledgements

Em primeiro lugar tenho de agradecer aos meus pais. Tenho de lhes agradecer por tudo o que fizeram e sacrificaram para que nunca nada me faltasse. Sempre estiveram presentes em todos os desafios que fui encontrando ao longo do meu percurso até aqui. Depositaram sempre em mim toda a sua confiança e carinho, o que sempre me fez sentir amparado e pronto para enfrentar o que quer que fosse. Lembro-me perfeitamente de tempos de maior dificuldade em que, para que que eu pudesse ter todas as oportunidades e experiências possíveis, muitas vezes se sacrificavam para que isso pudesse acontecer. Sei perfeitamente que lhes devo todos os agradecimentos do mundo por tudo isto e por me terem tornado a pessoa que sou hoje. Se hoje posso ter a oportunidade de estar nesta posição, é a eles a quem eu tenho de agradecer. Tudo que eu fiz até hoje e tudo o que eu irei fazer daqui para a frente foi e será graças a vocês, por isso, o meu muito obrigado, aos dois!

Em seguida gostava de agradecer à minha namorada, Inês, por me apoiar e dar a mão, sempre que eu necessitei. Por ser aquela pessoa que tinha sempre uma palavra de conforto e carinho quando mais precisava. Por ser a pessoa incrível que é, a pessoa que mais me animava e motivava quando ia abaixo. Por sempre ter acreditado em mim, desde o primeiro dia, sempre. Por todas as vezes em que, por mais que me fosse difícil, me fazia ver o lado positivo das coisas. Por me ter dado conforto e por ter ouvido os meus problemas e frustrações quando eu mais precisava. Por ser a pessoa que, independentemente da situação que eu esteja a passar, eu sei que irá estar lá para mim. Sem ti não teria conseguido, obrigado!

Quero também deixar uma palavra de apreço a todos os meus amigos mais próximos. Sem eles teria sido tudo muito mais difícil de lidar e suportar. Tenho a sorte de poder contar com várias pessoas em quem genuinamente confio e sei que me iram sempre apoiar. Aqui, quero deixar um agradecimento especial às três pessoas mais importantes que esta experiência universitária me deu: Afonso, Rui e Tiago. Obrigado por me aturarem durante cinco anos, o que eu sei que não é fácil. Por todos os trabalhos de grupo, noites sem dormir, ajuda e apoio sempre que qualquer um de nós necessitava. Foi um grupo de pessoas formidáveis que tive a sorte de conhecer e com quem eu tive o prazer de privar durante esta jornada. Obrigado por todos os momentos inesquecíveis que tivemos juntos. Momentos, que vou para sempre levar comigo e recordar com o maior dos carinhos possíveis. Foram o que de melhor me podia ter acontecido durante a vida universitária, obrigado!

Em seguida queria agradecer às duas pessoas que me acompanharam durante esta dissertação. Ao meu orientador, Doutor Ilídio Oliveira, e à minha coorientadora, Doutora Susana Brás. O meu muito obrigado pela incrível disponibilidade que demonstraram para comigo desde o primeiro dia em que começamos este projeto. Pela forma como me trataram e ajudaram durante todo este percurso, com um método de trabalho e comunicação incríveis, que permitiu que aprendesse bastante ao longo deste ano. Por sempre me terem ajudado quando os procurava, sempre com boa disposição e atenção. Espero ter correspondido às vossas expectativas e que tenham gostado de acompanhar o processo desta dissertação tanto quanto eu gostei de vos ter como orientadores. Aos dois, o meu muito obrigado!

Para finalizar, gostava de agradecer a todas as pessoas que se disponibilizaram para me ajudar durante a realização deste projeto. À Gisela Pinto, pela ajuda e disponibilidade demonstrada durante a fase inicial desta dissertação, pondo-me a par e ajudando-me a entender conceitos relativos ao material que me forneceu. Também quero deixar o meu agradecimento aos alunos do curso de engenharia biomédica, que fizeram parte do grupo de testes de validação da plataforma, pela sua disponibilidade e atenção.

O meu muito obrigado a todos, por me acompanharem durante esta jornada.

Palavras-chave

Computação Afetiva, Biosinais, Multimodal, Classificação Emocional, Plataforma Web, API, Filas de Processamento, Extração de Informação, Análise de Dados Afetivos.

Resumo

Os biossinais podem ser usados para quantificar vários tipos de informações sobre uma pessoa. Estes sinais, por si só, podem revelar, direta ou indiretamente, aspectos interessantes sobre sua fonte, tendo com isto uma vasta gama de aplicações. Uma aplicação emergente é o uso desses sinais como entrada de sistemas tecnológicos que fazem a detecção de emoções, enquadrado na área da designada Computação Afetiva.

Os sistemas de identificação emocional vêm ganhando cada vez mais atenção: projeta-se que, no futuro, muitos produtos sejam desenvolvidos com base na análise emocional dos seus utilizadores, a fim de proporcionar experiências mais personalizadas e adequadas, obtendo assim os melhores resultados possíveis. Para isso, são precisas ferramentas computacionais que permitam analisar e estudar os biosinais de forma fácil, abstraindo os processos computacionais subjacentes.

Este trabalho propõe e demonstra uma plataforma computacional que permite aos seus utilizadores analisar dados multimodais para deteção de emoções, gerando um relatórios de sumário em que se identifica três estados base: felicidade, medo e neutralidade. A plataforma pode usar dados de eletrocardiograma (ECG), atividade eletrodérmica (EDA) e eletromiograma (EMG). A plataforma possibilita ao utilizador realizar a extração e seleção de caraterísticas, e a classificação emocional, nesses tipos de biosinais.

A plataforma está disponível em ambiente Web e orientada para o utilizador final, que não precisa de ser um especialista, quer em biossionais, quer nos métodos de processamento. Como algumas análises podem ser exigentes e, por isso, não imediatas, a plataforma recorre à abstração de filas de trabalho para a submissão e acompanhamento das tarefas.

Uma versão inicial da plataforma encontra-se desenvolvida e foi explorada com problemas concretos de alunos que realizam a análise de emoções nos seus projetos académicos.

Keywords Affective Computing, Biosignals, Multimodal, Emotional Classification, Web Platform, API, Work Queues, Information Extraction, Affective Data Analysis.

Abstract Biosignals can be used to quantify several types of information about a person. These signals, by themselves, can reveal, directly or indirectly, interesting aspects about their source, thus having a wide range of applications. An emerging application is the use of these signals as input for technological systems that detect emotions, in the area of Affective Computing. Emotion identification systems are gaining more and more attention: it is expected that, in the future, many products will be developed based on the emotional analysis of their users, in order to provide more personalized and appropriate experiences, thus obtaining the best possible results. To this end, computational tools are needed that allow to analyze and study biosignals in an easy way, abstracting the underlying computational processes.

This work proposes and demonstrates a computational platform that allows its users to analyze multimodal data for emotions detection, generating a summary report in which three base states are identified: happiness, fear and neutrality. The platform can use electrocardiogram (ECG), electrodermal activity (EDA), and electromyogram (EMG) data. The platform enables the user to perform feature extraction and selection, and emotional classification, on these types of biosignals.

The platform is available in a Web environment and is oriented towards the end user, who does not need to be an expert either in biosignals or in processing methods. As some analyses may be demanding and therefore not immediate, the platform makes use of the abstraction of work queues for task submission and tracking. An initial version of the platform has been developed and explored with concrete problems of students performing emotion analysis in their academic projects.

Contents

List of Figures

List of Tables

List of Acronyms

ADHD Attention-Deficit / Hyperactivity Disorder.

AI Artificial Intelligence.

AMQP Advanced Message Queuing Protocol.

APAC Asia Pacific Region.

API Application Programming Interface.

ASCII American Standard Code For Information Interchange.

AX Advanced Vector Extensions.

CSV Comma-separated Values.

DGS Direção Geral de Saúde.

ECG Electrocardiogram.

EDA Electrodermal Activity.

EEG Electroencephalogram.

EGG Electrogastrogram.

EMG Electromyogram.

EMG-MF Electromyogram Frontalis Muscle.

EMG-Z Electromyogram Zygomatic Major Muscle.

EOG Electrooculogram.

EPG Electronic Program Guides.

ERG Electroretinogram.

GPS Global Positioning System.

GSR Galvanic Skin Response.

HDRS Hamilton Depression Rating Score.

HR Heart Rate.

HS256 Hash-based Message Authentication Code with SHA-256.

HTML Hypertext Markup Language.

HTTP Hypertext Transfer Protocol.

IEEE Institute of Electrical and Electronics Engineers.

JSON JavaScript Object Notation.

LLD Late-life Depression.

MARP Multimodal Analysis Reports Platform.

MIT Massachusetts Institute of Technology.

NN Neural Networks.

OCD Obsessive Compulsive Disorder.

PHP Hypertext Preprocessor.

PKL Pickle.

PPG Personal Program Guide.

PST Problem Solving Therapy.

PTSD Post-traumatic Stress Disorder.

RDBMS Relational Database Management System.

REST Representational State Transfer.

RF Random Forest.

SHA-256 Secure Hash Algorithm 256-bit.

SUS System Usability Scale.

TXT Text files.

UI User Interface.

UML Unified Modeling Language.

VR Virtual Reality.

WSGI Web Server Gateway Interface.

XLT Microsoft Excel Template Files.

YAML Yet Another Markup Language.

Chapter 1

Introduction

The world has become a place where, nowadays, technology is seen as an extension of the human being. In order to continue to evolve in many different fields and areas, both financially and socially speaking, modern civilizations are now reliant on technology. This dependence is more of a consequence of all the benefits brought to us by technology. There are currently many branches in the technological world, almost everything that today produces something to the society maintenance and evolution has a technological touch.

Scientific fields that study human responses to a great variety of impulses or situations can make use of biosignals, which are electrical signals produced by our body in reaction to these impulses. These fields of science are getting more and more relevant nowadays, due to an increased interest in the study of the human responses and how these can help to figure out or resolve some problems in peoples lives, as well as how it could be cooperatively implemented alongside technology.

Despite all this, everyone knows that there exist some properties that only humans can perceive, like feelings and emotions. But what would happen if the human being made it possible for a machine to be able to interpret these same feelings and emotions? What would happen if a machine could reason and act upon human physiological and psychological changes? There is still great debate around this matter, and the consequences of such actions are yet unclear. Understanding a person's preferences may be helpful in the design of efficient and more appealing systems. Preferences are associated with cognition, learning, and therefore to basic emotions. These emotions may be understood by the physiological or behavioral responses of a person.

Nevertheless, something that all scientific community agrees is that there is a need for more studies, in order to evaluate this question and to try to predict the impact of this subject, today, commonly known as affective computing. The impact of these types of technologies can be reflected on society, on people's lives, and even on solving other technological problems. The emerge of affective computing made clear that there is also a lack of platforms and resources for researchers and experts to have a simplified way of analyzing biosignals in general, due to the current existent system's complexity or even due to the lack of available features on this systems to make them useful.

All these variables make clear that the existence of a system able to eliminate such barriers that researchers have when analyzing biosignals in order to apply them to an end as affective computing does, is something of high importance and that if made available would help the scientific and technological community to evolve.

1.1 Motivation

As previously suggested, a technological platform that would be able to allow it's users of uploading biosignals of any kind and then perform some designated action upon these same signals, serving as a comprehensible interface that would eliminate some existing restraints these types of users encounter when they try to process these signals, would be a very helpful tool for scientific advances regarding the treatment and processing of biosignals.

Several fields of science that base their ideologies on the analysis of biosignals in order to conduct their studies could fit in this description and benefit of such a platform, like many medical-related fields, educational fields where this could be applied into the educational programs to modernize and update teaching techniques, among other related areas. Another one of these scientific fields would be, the already mentioned, affective computing which for reasons of interest, content, material available, and relevance in the future of technology, will be the one in which this work will be focused the most despite it being thought and developed in order to be a generic platform that could easily scale to any other application of biosignals.

Affective computing is an on-growing field of the technology industry that aims in serving humans with technological services that can adapt to the user, by being able to interpret their physical/psychological state through the analysis of available data regarding this user. With this type of technology, the modern world can suffer a major change, with affective computing bringing many benefits to already identified problems in current developed technologies. For example, a virtual house assistant interpreting it's user's physical expressions, in order to identify their mood and change the way it operates according to this information, or for video games that use this same physiological data to better the game's performance and adjust it to the user, or to help people with mental disorders as PTSD, ADHD, anxiety, among other issues, that a system with the capability of gathering, processing and adjusting it's behavior according to the emotions the user shows or reports, can become a major help to battle these type of problems.

In a world where constant advance and innovation is seen as a requirement and something that is constantly encouraged to the new generations of workers, the stress and anxiety that this lifestyle brings can be something challenging to deal with[3]. If technology is idealized to serve us, human beings, why not use it as a tool that helps to heal or repair something caused by this constant need for innovation that technology brings? Here, systems prepared to interpret, understand and act upon human bio-related data can be of great importance. Some tests on cases that try to use for example mobile apps [4] [5] [6] for helping people with this diagnosis were already conducted and can be further analyzed later on this document.

Enabling, for example, systems like home assistants or smart televisions [7] to understand the user's physiological signals or facial expressions, could make these systems evolve according to the user's state of mind without them having to report it or adjust the system according to this. These types of features that can be achieved with affective computing would increase exponentially the interest and satisfaction levels that people have with those systems.

Making use of gamers' facial expressions and some physiological signals like heart rate would also help game engines to deliver a better experience to the user, adapting the game to each player and each individual experience. Studies regarding human interactions with a system in order to bring more evidence on this related matter have already been conducted and brought some interesting results [8] as it can be consulted later on this document. This type of technology promises to revolutionize the gaming industry by delivering games that become unique for each individual.

The options possible in these and other cases are so many that is still difficult to predict the impact or even differentiate the hype from true potential around this matter. Since it is a so recent and promising area, researchers and experts are still trying to measure affective computing impact on the current technological world. The one thing that all scientific community is sure of is the necessity of more studies and researches regarding this matter, so that technologies like the ones idealized above, and others, can come to life.

In order for technologies like the ones that affective computing projects to become a reality in the near future, several research projects need to be made to better understand how this area will interact with society and how will people respond to such a change in the way they see technology and ultimately the way technology perceives them. These studies are being developed and conducted but some constraints slow down the work needed. In the case of using artificial intelligence models to analyze and try to predict something out of biosignals, several constraints are identified as being a repressor of this advance. Such constraints may vary from the different formats in which the data from the software that extracts this data recurring to sensors, comes in, the need for systems that allow researchers of making a more visual representation of the data provided by the application of the AI models to certain biosignals datasets, the lack of a platform where the analysis made can be stored and organized and later observed by the researchers, among other gaps that make researches, slower and more difficult to the professionals executing them. In this case, the barriers or gaps identified relate to affective computing, but could also be applied to many other fields.

It was from these identified problems that came the idea to develop a web platform capable of answering to the issues reported above and others, that act as a barrier that prevents advance in several research fields. From this, the idea of Multimodal Analysis Reports Platform (MARP) was born, with it being a platform where users could register themselves in order to submit and store their files and AI models, perform analysis by combining these files containing data from biosignals and the submitted models, store this analysis, check it's informations, check it's results, to see the results as a visual representation by means of a graph, among other useful features for researchers or even someone with access to these type of files and that wishes to conduct a study in an easy, direct and simple way.

1.2 Objectives

The main objective of this work is to produce an initial platform that can be used as a tool to assist people like the ones described in the sections before and which are further explored in section 3.1.1. This goal comes in aid of some identified problems in the research field, described on those personas. Since there are many representations possible for the data extracted on the experiments, since sometimes, the usage of technologies and/or programming languages in order to use some libraries that help acquiring data is not trivial, or that there is not a visual platform where the processes behind the execution of all these tasks can be hidden from the user so that they can focus entirely on the conclusions to take from the results obtained, this is an issue that is not something to be overlooked.

With this, the main objectives taken as a starting point for a project like MARP are:

- Create a web platform capable of executing analysis to biosignal files using artificial intelligence models in a straightforward way;

- Enable the platform of concurrency regarding the processing of each request of analysis, so that many users with many requests can use the platform simultaneously and independently;
- Make the platform able to interpret the results of the analysis and produce graphs and plots so that the users can visually comprehend the results;
- Allow the platform to securely store the information and files of different users independently.

Chapter 2

Background & State-of-the-art

As mentioned in the section before, this work, despite being a platform focused on covering all the areas that use biosignals to perform their experiences and take their results, it will be majorly focused on the affective computing field of study. For this, it is important to know what affective computing is, what does it study/develop, what are the already developed works that can be a mean of comparison to this one, along with some other questions that will be answered on the development of this section.

There is an on-growing interest in human emotional recognition technologies nowadays since this is seen as an important next step in human-computer interaction. If there is a way to make systems responsive to human emotional states, making them adapt according to the user's emotions and state of mind, it will be a game-changing accomplishment in terms of user experience and feedback, when using these systems. Since some studies reinforce this idea[9], many technology fields are focusing on developing technologies that try to give an effective and natural interaction between humans and computers. One of these areas, and maybe the most popular, is affective computing.

2.1 Biosignals: Current State & Technologies

A biosignal is any signal acquired on living beings that can be continuously measured and monitored. Biosignals may be divided into different categories, depending on the physical principle behind their measurement. One of the most used are the bioelectrical signals like it is the case of the ones used in this work.

Electrical biosignals or bioelectrical signals are produced by the electrical activity that arises from the biological activity that takes place within different tissues, organs or cell systems of the human body [10]. Thus, examples of bioelectrical signals are seen in table 2.1.

In the scope of this work, EMG, ECG and EDA were the chosen signals for emotional evaluation, due to their relation and description on literature as relevant and indicative of emotional responses[11]. ECG and EMG are measured using a differential amplifier that registers the difference of electrical potential between two electrodes attached to the skin of the subject. EDA measures electrical conductivity produced by the sweating of the skin of the participant, between two electrodes.

As it can be obtained by the information from table 2.1, ECG gives information about the cardiac electrical activity produced by the subject. With EDA it is possible to quantify the subject's skin conductivity. About the EMG, it evaluates and records the electrical activity

Signal	Description
Electroencephalogram (EEG)	Records electrical activity on the scalp of the subject, that has been shown to represent the macroscopic activity of the surface layer of the brain.
Electrocardiogram (ECG)	Gives information about the cardiac electrical activity produced by the subject.
Electromyogram (EMG)	Evaluates and records the electrical activity produced by skeletal muscles, more often known as muscle fibers.
Electrooculogram (EOG)	Measures the corneo-retinal standing potential that exists between the front and the back of the human eye.
Electroretinogram (ERG)	Measures the electrical activity of the retina in response to light stimulus.
Electrogastrogram (EGG)	Records the electrical signals that travel through the stomach muscles of the subject, and control the muscles' contractions.
Electrodermal Activity (EDA) or Galvanic Skin Response (GSR)	Measures the electrical conductivity produced by the sweating of the skin of the participant.

Table 2.1: Most relevant biosignals and their respective descriptions.

produced by skeletal muscles, more often known as muscle fibers. Specifically the Electromyogram Frontalis Muscle (EMG-MF) and Electromyogram Zygomatic Major Muscle (EMG-Z) that are the EMG type biosignals more used in this work, they are measured detecting this muscular electrical activity or tension on the frontalis muscle and on the zygomaticus major muscle, respectively. The frontalis muscle covers parts of the forehead and is a muscle that only serves for facial expressions, in humans, so it is relevant for this work. Similarly, the zygomaticus major muscle is also associated with facial expressions, majorly to the smile, since it is located on the human face, extending from each zygomatic arch (cheekbone) to the corners of the mouth.

Many of the technological systems that work with this kind of systems focus on their features. Biosignals do not provide a straightforward interpretation. However, they present relevant information in the physiological description of the person. So, features that characterize the signal are extracted. For example, from the ECG signal it is possible to extract a feature called Heart Rate (HR) and with this, it is possible to deduct for example if the subject indicates physical effort. These features are very important specifically on technological systems that use biosignals as a source of information because with these it is possible to better categorize a signal and ultimately the participant behind that signal[12].

The advance of technology has brought benefits to the biosignals field as well, with the development of methods for the measurement of electric signals using new remote sensor technology. Studies show that electric biosignals such as EEG[13] and ECG[14] can be measured without electric contact with the skin, showing this same benefits brought by technological advances on the biosignals field. This can be applied, for example, for remote monitoring of brain waves or to detect the heartbeat of patients who must not be touched, as it is the case of patients with serious burns. Computational systems that can detect the user's heartbeat without having to connect a device to them, consist of a less intrusive gathering of biosignals when conducting some studies, and many other applications.

Even artistic areas can benefit from the biosignals field advance, for example, Andrew Brouse et al. [15] documents how students of the University of Toronto and professor Steve Mann were developing an advanced electronic and computer technology for a "one-of-a-kind" sonic and visual performance, an improvised cooperative musical piece made interactively from audience members' brainwaves.

It is vast the application of biosignals potential to many other areas, with some having more interest in this matter due to it's direct participation in these fields of study. Hagen Malberg and Werner Wolf et al. [16] give a perspective of the advances of biosignals and predictions about it's effects on the clinical area. Despite mentioning that the full potential of the benefits that biosignals could bring to the area are yet to be accomplished. They also mention that these advanced techniques are candidates for new concepts in clinical biosignal processing since it is to note that even the most perfect processing of biosignals can not compensate for deficits during recording, therefore, progress with new sensors and sensor technologies will also bring innovations in biosignal processing. The authors proceed by mentioning that these techniques of biosignal gathering allow dynamic adaptation to the organism observed and facilitate the understanding of the underlying physiological process. On the ending part of their work, they mention that these new perspectives on biosignal processing can result in new achievements in clinical diagnosis, physiological monitoring, and treatment control, as well as the evaluation of genetic and pharmacological effects for medical application. Furthermore, the replacement of sometimes unpleasant, intrusive treatments with noninvasive technologies that provide similar information, will result in revolutionary diagnostic and therapeutic concepts [16]. With all this data and predictions it is possible to expect a bright future for biosignal related areas, that can achieve great potential, but that in order to become true in a near future, more studies and investments in the area need to be made.

In terms of technological systems to study and/or apply certain tools to biosignals, currently, there are limited options in terms of available systems for storing and/or processing biosignals like it is the goal of MARP. This lack of options is another factor that motivated the development of the platform presented in this work, alongside with the sometimes excessive specificity that some of these tools are characterized for, the difficulty of use some of them have, among other problems.

One of the most recognized tools available is entitled PhysioNet[1]. This is a website that offers free access to large collections of physiological and clinical data and related open-source software. Alongside with this, PhysioNet also hosts an annual series of challenges, focusing research on unsolved problems in clinical and basic science. The PhysioNet resource has three closely interdependent components [17]:

1. A repository of well-characterized digital recordings of physiologic signals, time series, and related data, nicknamed "PhysioBank," is available for use by the biomedical research community. Data from a wide range of studies, as data created and contributed by members of the scientific community, is included in these collections. Clinical and imaging data relevant to critical care are also included in PhysioNet.

2. "PhysioToolkit" is the nickname of a wide library of software for physiologic signal processing and analysis, as well as detection of physiologically significant events and a variety of additional signal processing and analysis methodologies and techniques.

3. A collection of popular tutorials and educational materials, that offer expert guidance in

approaches for exploring and analysing health data and physiologic signals. The unifying theme for these resources is the focus on the extraction of "hidden" information from biomedical data, providing information that may have diagnostic or prognostic value in medicine, or explanatory or predictive power in basic research.

This tool is an example of a platform that works with biosignals that serve as a data bank and also provides software for users to explore. Despite it making available all this data and tools it does not provide a way for users to make use of these tools and datasets on the platform. The users must always download what they are interested in and then use it on their own. This can bring some technical issues to the user, like not being able to use what they intend due to technological limitations, not so explicit documentation that could help to overcome difficulties, among other factors. These issues would not exist or at least, would be mitigated if the platform handled some of this technical work for the user.

It is here that MARP emerges as a potentially helpful tool, it wouldn't serve as a library for the exchange of data but it would enable the users of uploading their data, possibly even data available on platforms like PhysioNet, and then execute the actions the users pretend, eliminating the technical difficulties that some of these users might feel in terms of applying some transformations, the application of the tool itself, among other difficulties. With this they would only have to focus on analyzing the results that the platform would generate according to the specifications that were defined by them, making this a faster and easier work process.

2.2 Affective Computing

Emotions are a challenging topic of research since there is an ongoing divergence of opinion on the definition and concept of emotion. Emotion modulates almost all modes of human communication, these are word choice, tone of voice, facial expression, gestural behaviors, posture, skin temperature, respiration, muscle tension, and more. Emotions can significantly change the message as sometimes it is not what was said that was most important, but how it was said.

The concepts of emotion and feeling are frequently used interchangeably, according to António Damasio, but he believes they should not be. From a research perspective, it is advantageous to use separate terms to designate separable components of this enchained process. Damasio et al. [18] states that the term emotion should be used to identify a number of responses produced by neuronal and humoral connections from portions of the brain to the body and from sections of the brain to other different parts of the brain. The culmination of these responses is an emotional state, which is defined by changes in the physical body. He also claims that the term *feeling* should be used to describe the mental state that arises as a result of an emotional condition. With the mental state including the representation of changes in the body-proper that have recently occurred and are being signaled to body-representing structures in the central nervous system and also including a variety of changes in cognitive processing, produced by signals secondary to brain-to-brain reactions. Damasio also emphasizes the relevance of emotion in the survival and social relationships of living beings, as well as being a fundamental factor in describing organisms. He cites four reasons why neglecting emotion in the concept of an organism is illogical, emphasizing the importance of emotion in the survival and evolution of living organisms [18]:

[1] Accessible at: `https://physionet.org/`

1. In complex organisms, emotion and the experience of emotion are the highest-order direct expressions of bioregulation. Without emotion, there is no way to fully comprehend bioregulation, especially when it comes to the relationship between an organism and the most complex parts of it's environment: society and culture.

2. Emotion is essential for survival in complex organisms that are capable of processing it. Would not it be interesting to learn how emotion works, whether in animals or people, since emotional problems can be fatal?

3. Emotion plays a role in memory, and one of the fundamental goals of neuroscience and cognitive science is to better understand memory. Memory is a characteristic of living systems that are focused on survival, and these two functions, emotion and memory, are so tightly linked in complex creatures that one cannot fully comprehend the latter without first understanding the former.

4. From the fundamental decisions that animals make to avoid danger or affirm a beneficial encounter to the more complicated decisions that we humans may evaluate, emotion plays a role in reasoning and decision-making. Without an understanding of emotion, it is impossible to make sense of the mechanisms that underpin the greatest human achievements: rationality, ethics, legislation, artistic, scientific, and technological creativity.

Michel Cabanac et al. [19] mentions that there is no universal agreement on the definition of emotion in the literature. The concept is taken for granted, and emotion is frequently described in terms of a list: joy, sadness, anger, disgust, fear, and surprise. These are considered the basic emotions produced by human beings. He suggests that emotion is any mental experience with high intensity and high hedonic content (pleasure/displeasure) as a follow-up to his work on a four-dimensional definition of consciousness (quality, intensity, hedonicity, and duration).

Emotions and the physiological changes that these same emotions cause in humans can be used as input to fuel specific technologies and/or studies, which is something that is of significant interest and relevance. As much as can be discerned at this moment, despite the fact that they are still surrounded by a layer of uncertainty, emotions indicate that if used correctly and further research to remove this uncertainty is performed, are expected to have a significant impact on a variety of domains. Incorporating emotions as a research object in order to develop new technologies and better understand their significance and power is what affective computing does.

Affective computing is an area that describes computational approaches to the detection and deliberate induction of affect[20]. It is seen as a multidisciplinary field that studies how technology can comprehend and inform human affect, how affect can change the interactions between humans and technology, what can change in systems design so that these can use affect to enhance capabilities, and how sensing allied with affective strategies can shape new human-computer interactions[21]. Affective computing is a relatively new but very promising field of study, with the capability of changing the current panorama of human-computer interaction by making systems capable of, intelligently, adapting and responding to it's user's state, being much more responsive and becoming user-oriented, which will lead to a better user overall experience.

When this field of study first came to the public many questioned it's place on the technology field since emotions were viewed as comprehended and reproducible only by humans, but as Picard and Klein suggest et al. [22], it is not required for a computer to have the emotional qualities of a human being for it to address their emotional needs. For example, by using physiological measures, as face recognition, heart-rate variability, and skin conductance, a computer can recognize key emotional states and related changes on the user. The communication of such information to the user can help increase emotional skill needs by contributing to self-awareness and experiential needs by helping the user to feel as if their strong affective state has been communicated in an effective way[22] [20].

There are also numerous areas where affective computing can be used as help to fill in some gaps. For example, the gaming industry can monitor users while they test their products prior to their launch, in order to keep track of their satisfaction levels so that the final product has the best possible impact on the final consumers. Another more ambitious application is to monitor users' facial expressions in order to make the game adapt to the user's experience. Affective computing has a lot of potential in the educational industry as well, it may be used to track student's happiness and/or dissatisfaction levels throughout a class and assist teachers to adjust their lectures to benefit the students, it can also be used in schools with autistic children to assist them in recognizing emotions in others or the inverse. There are numerous applications of this field of technology in customer service, such as systems capable of detecting angry customers through voice recognition and redirecting them to more experienced employees. The human resources field, for example, can improve upon systems capable of interacting with training employees, by giving intelligent and evolving interactions with the employees based on their reactions and voice patterns, it can use systems to keep track of their employees' mental health by assessing their stress and anxiety levels to evaluate if the jobs they're doing are being well received or if modifications need to be made. Insurance firms, for example, can benefit from emotional computing systems by deploying systems that can detect fraud by monitoring, for example, a person's voice changes and act accordingly[23].

There are numerous more scenarios in which emotional computing can be used to meet human demands, which can range from economic to health-related concerns. These systems have far too many applications to be overlooked or disregarded.

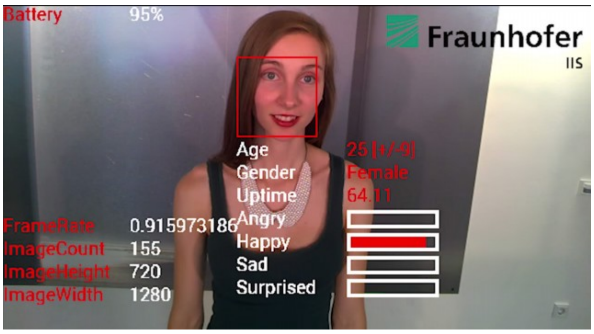

Figure 2.1: World's first real-time emotion detection app for Google Glass, from Fraunhofer Institute in Germany.[2]

The majority of this field's study is focused on identifying emotions through speech or facial expressions. Humans have the ability to identify another people emotional state based on some characteristics present on their facial expression or even some change in their tone

[2]https://www.extremetech.com/extreme/189259-real-time-emotion-detection-with-google-glass-an-awesome-creepy-taste-of-the-future-of-wearable-computers

of voice, so facial expressions and speech are two of the main ways one might demonstrate a certain emotion, and they are the most noticeable ones for humans to comprehend. Some computer software can now analyze these facial expressions or vocal changes during different situations, incorporating video images or photographs along with microphones to record the person's voice during a conversation. As a consequence, an artificial intelligence developed to detect emotions based on this type of information can detect, as well as recognize patterns in the data and correlate those to it's predetermined data, describing each emotion, attempting to determine what the user is feeling. There are also studies that indicate that combining audio and video signals for emotion recognition can even give improved results [24].

Studies like the one conducted by Brás et al. [4], in which it was shown that spider phobics differed from control participants in face motion captured by the smartphone camera. BeMonitored was the smartphone app used in this study, that allows external resources connected via Bluetooth or a smartphone's own resources, to deliver customizable, context-dependent audiovisual stimuli while attempting to capture the participant's behavior, physiology, and environment. The results of this study revealed that phobic participants' heart rates differed when they were exposed to spider and neutral stimuli. The findings of this study highlight the utility of smartphones for phobia monitoring because, considering their inherent characteristics, smartphones may represent a natural evolution from the laboratory to more realistic contexts, which might help to improve a great number of studies in terms of efficiency and cost. Smartphones and Virtual Reality (VR) can be used to improve studies of emotional experience and presence, according to Diemer et al. [25], resulting in new findings, better data extracted from users due to more immersive experiments, and other potential benefits that this collaboration with technology can bring.

Unfortunately, this type of information may not always be obtainable. Furthermore, in some instances, biosignal data may be required to fully comprehend a user's emotional state [26]. What biosignals can better describe and provide information to a system about a user's mental state, is still under investigation. Several research works have been conducted, and according to Gisela Antunes et al. [27], the optimum combination of signals for identifying emotion is the combination of ECG, EDA, EMG-MF, and EMG-Z signals with their respective features. According to the same study, the ECG signal appears to be the most informative in emotion stratification, and that the use of facial EMG in emotion is reliant on monitoring two (or more) muscles, allowing for facial expression changes to be identified by corresponding muscular contractions.

According to the founder and director of the Affective Computing Research Group at the Massachusetts Institute of Technology (MIT) Media Lab and founding faculty chair of MIT's Mind+Hand+Heart Initiative, and who is credited with starting the branch of computer science known as affective computing, Rosalind W. Picard[3]et al. [28], collecting data relating to emotional arousal is not something new. Skin conductance, for example, has been researched for over a century. However, what's different is how technology can now measure, communicate, respond to, and transform emotion. She also claims that the convenience of a new affective computing technology can lead to greater self-awareness, improved interpersonal communication, and much more, including new technologies that alleviate rather than increase stress (if researches make it possible). She continues by stating that there is a commune proverb among researchers that goes as "if you can't measure it you can't manage it."[28].

[3]https://web.media.mit.edu/~picard/

Measuring, for example, the frustration caused by technology, can enable engineers to pinpoint what causes it and work to prevent or reduce it. Technology can also be improved if it has the intelligence to respond to emotion, and also by applying emotion principles acquired from biological systems. However, in order to bring about new benefits, several challenging difficulties must be overcome [28].

In the same paper, Picard et al. [28], talks about how difficult it was, in the beginning, to introduce affective computing as a legitimate technological field. After this troubled start, she decided to give this new field of affective computing a chance, publishing her tech note envisioning affective computing as a broad area that Picard believed engineers, computer scientists, and many others should consider working on and submitting it as a manifesto to a non-IEEE journal that had a history of publishing bold new ideas. It was rejected and one of the reviews indicated that "the content was better suited to an in-flight magazine". This provided her with even more motivation to pursue her goal of making affective computing a reality. So, she began to devote more of her time to all of it. As a result, her work started to acquire traction, and during a meeting with the renowned scientist Peter Hart, he told her that he thought affective computing was going to become very relevant and encouraged her to abandon the research for which she had just raised over a million dollars in funding (content-based retrieval) and devote herself completely to affective computing. This and a couple of other occurrences refereed on her paper made Picard pursue the field of affective computing and later on submitting to MIT, her fresh copy of *Affective Computing*. After some rejections from IEEE and ACM, for articles she submitted because they "didn't fit" the supposed ideals they valued, the inaugural issue of the *IEEE Transactions on Affective Computing* was published, establishing the subject as legitimate and respectable [28].

After such a rough start for the field and with all the data on how affective computing is a branch of computer science that, among other benefits, can assist in overcoming many problems that arise in other domains of technology as well as improving significantly user experience and gratification, it is undeniable the possibilities this area brings. It has the ability to revolutionize and transform a wide range of fields, not just computational ones, and to provide several benefits in order to keep society and technology moving forward.

A recent study by Markets and Markets published on the 20^{th} of June of 2020 projects that after a global pandemic like it was with COVID-19, the affective computing market size is still projected to grow and in a large way, passing from roughly 28.6 billion USD in 2020 to 140.0 billion USD by 2025 [1].

Affective computing is expected to have a considerable impact on any company's future, which would include ergonomics, human factors, project management, and organizational changes. This factor has encouraged the global adoption of emotion AI/affective computing solutions in a variety of industries. The cost of developing affective computing systems is now higher than desired, and the tangible return on investment is limited. This is a significant impediment to the affective computing market's growth, as most of the enabling technologies, such as wearable computing and gesture detection, have high development costs. As a result, businesses with financial constraints do not choose the platform, despite wanting to use affective computing to improve productivity and boost their businesses. This is a roadblock that, once overcome, will increase the size of the affective computing market, even more, offering up a whole slew of new opportunities for the field.

The biggest current problem in the AI emotion/affective computing industry, according to the Markets and Markets study, is that, as a consequence of being a relatively new area, it is not yet fully understood how affective computing will impact business applications and processes. Affective computing is cutting-edge technology and has far more advantages than various existing technologies, including AI-based computing solutions and sentiment computing solutions. The enormous excitement surrounding technology is also addressed, yet, top executives have yet to discern or separate this enthusiasm from the true potential. Furthermore, the technology's societal and economic implications have yet to be investigated. There is still a lot of uncertainty about what affective computing is capable of and what it is just not.

During the forecast period, North America is estimated to account for the majority of the affective computing market share. As shown in the plot from figure 2.2. Despite this, major economies in Asia Pacific Region (APAC), such as China, India, Japan, and Australia, are projected to have fast growth in the affective computing market [1][29][30].

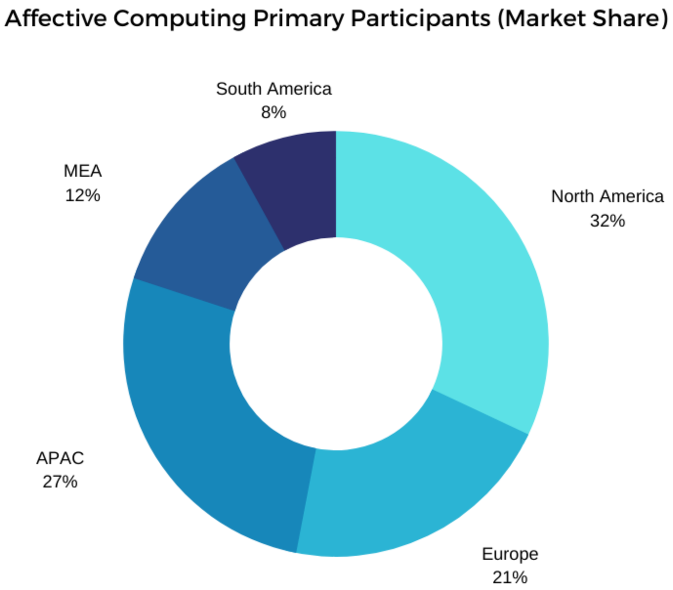

Figure 2.2: Affective computing market share participants by region.(Base on: Markets and Markets report[1])

With all of this knowledge, the affective computing business appears to have a promising future. It is also evident that it is an area with certain limitations and skepticism, implying that further investigation is necessary to fully comprehend how this exciting concept of emotional computing might benefit humans and revolutionize technology and the way the current world perceives it. Affective computing is bringing new methods of interacting with technology and integrating technology into people's lives, and thus the world is waiting for it to be established and incorporated into society with great expectations and enthusiasm.

2.3 Case Study on Multimodal Emotional Analysis

Having as background the work developed by Pinto et al. [11], where it was evaluated how well different signals described physiological response towards emotional stimuli, as well as what signal/combination of signals may better describe this response. To achieve the conclusions, fifty-five participants were exposed to three different stimuli on three different days. To induce the stimuli movies were used to induce the user with three different emotions, fear, happiness, and neutral emotion, utilizing horror, comedy movies, and documentaries, respectively.

In the study was observed that the best combination of signals for emotion description corresponds to using all the signals available to the study, that were ECG, EDA, EMG-MF and EMG-Z, with all the selected features refereed on the study. When considering isolated signals on classification, the study showed that two signals have obtained good results: ECG and EDA. It is referenced as well that when analyzing the sensitivity and specificity results, the ECG had more correct classifications than EDA. The EMG also achieved good results when both collected signals, EMG-Z and EMG-MF, are combined. This is said to occur because the two separated signals will not provide the complete description of the emotional response, but together they complement each other's information and the emotional description is more accurate. It is also mentioned that EMG-Z captures more information at a level of happiness, and the EMG-MF captures more information at the level of fear.

In terms of machine learning algorithms, there were discrepancies between random forest and neural networks. The results indicated that the neural network describes better the emotional context when a subject or emotion independent test was performed, while the random forest described better the subject and emotion dependent context.

With these conclusions gathered from the previous work, it was provided to this study the artificial intelligence models, trained with the data gathered on the previous work. These models are referent to each signal individually (ECG, EDA, EMG-MF, EMG-Z), the combination of all the signals and with both EMG signals (EMG-MF and EMG-Z), all trained with random forest and with neural networks, resulting in twelve models available to use in this project. All these models are the models that achieved better results in each "category". Along with the artificial intelligence models it was also provided the source code of some functions that allow to run analysis on datasets with data from groups of participants. These functions helped to understand better how the models are applied to the information gathered from the dataset, as well as the results it produces and how these results could be used in order to achieve the objectives of this work.

Although these functions were helpful it was necessary to change almost everything regarding their content or structure, in order to integrate the models and the analysis on this work's pipeline and to redirect the results as intended during the development of the platform.

2.4 Selected Applications on Affective Computing

The subject of affective computing is steadily expanding, with numerous new initiatives aimed at creating machines capable of modeling, recognizing, expressing, communicating, and responding to emotions. With this, there are many interesting works related to affective computing with potential, and that brings to the surface the importance and strength of this subject.

One project that shows how an industry can benefit and be completely reinvented by affective computing is the experiment made by Regan Lee Mandryk et al. [8] where, in this study, Regan found evidence that the human brain generates different physiological responses when a person is playing a game against the game's AI and when played against a friend. In the research, users' physiological, verbal, and facial reactions to game technology were recorded, and post-processing techniques were used to correlate an individual's physiological data with their subjective reported experience and events in the game. The ultimate goal was to create a methodology for the objective evaluation of collaborative play technology. GSR is a highly responsive body signal, it provides a fast-response time-series, reactive to events in the game, so it was used to measure users' responsiveness upon scoring a goal against the computer and against a friend, in addition to EMG.

This study can show how human reactions change based on who or what the user is interacting with, highlighting that if it becomes feasible to generate systems capable of reading and acting upon human physiological responses, it is possible to become the system reactive to the user and with that make it adjust to the users bettering their experience.

Another interesting study involving affective computing was developed by Joaquin Anguera et al. [2], who created a proof-of-concept trial in which twenty-two subjects diagnosed with Late-life Depression (LLD) were randomized into two different therapies in order to try to help these subjects reduce depressive behaviors. Problem Solving Therapy (PST) was used, which is an eight-week strategy that includes three treatment phases:

- Phase one lasts three weeks and is designed to help participants understand problem solving steps and use the problem solving action planner to work on psychosocial issues;
- Phase two consisting of independent practice of PST skills,
- Phase three that consists of two relapse prevention sessions that use the problem solving model to develop plans to keep depression and function at a minimum.

The other therapy used was a cognitive intervention that in this case was a mobile, iPad, an intervention called "Project: EVO™" based on the video game called NeuroRacer.

The objective of this game is to navigate a character through an immersive environment while responding to certain targets. The integrated adaptive algorithms iterate the user's experience to guarantee that the game is hard but nevertheless entertaining, with room for improvement over time [2].

Care managers introduced participants to the intervention, explained the therapeutic justification for EVO, and the justification for the weekly check-in visits during the first session. With the care manager present, participants then conducted a practice run with the iPad and EVO. For four weeks, participants were required to play EVO five days a week for roughly twenty minutes.

The results of this experiment were that EVO participants demonstrated similar improvements in mood and self-reported function after four weeks of treatment to PST participants, as we can see from the results shown on the graph from figure 2.3, these improvements were concluded after comparing the results of the measured Hamilton Depression Rating Score (HDRS) that is a multiple-item questionnaire used to provide an indication of depression by the form of a score for each method. Participants in the EVO condition also demonstrated generalization to untrained measures of working memory and attention, as well as negativity bias, which was not seen in the PST condition. It is to notice that individuals assigned to EVO also demonstrated 100% adherence [2].

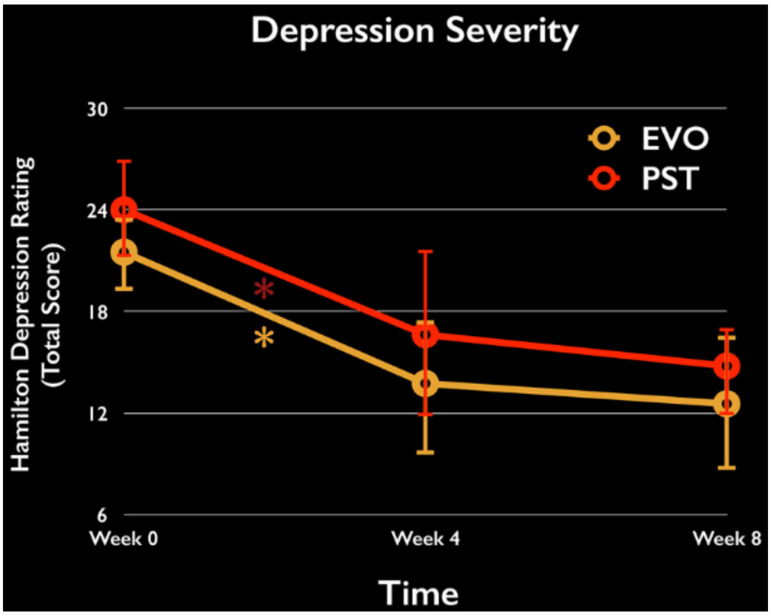

Figure 2.3: Change in depressive symptoms over time for PST and EVO therapies. (Source: Joaquin Anguera et al. [2])

Although Joaquin mentions that further studies are required in order to conclude the real impact of therapeutic games in treating late-life depression, also mentions that the study indicates preliminary findings that this therapeutic video game targeting cognitive control deficits may be an effective intervention to fight this psychological disease [2].

A paper of a study conducted by L. Ardissono et al. [7], was presented on personalized recommendation on TV programs is also very interesting to relate to this affective computing area. In this paper, the Personal Program Guide (PPG) is a prototype system that generates personalized Electronic Program Guides (EPG) which were developed in a collaborative project between Telecom Italia Lab and the University of Torino. The PPG incorporates a variety of user modeling approaches to recognize TV viewers' interests and generate individualized recommendation lists as a result. User modeling is a sub-field of human-computer interaction that describes the process of constructing and updating a user's conceptual understanding. User modeling's major purpose is to customize and adapt systems to the user's individual needs. From this research, three key conclusions may be drawn. First, managing several perspectives on user preference recognition, as well as cooperation/competition between different user modeling methods, has proven to be very beneficial in strengthening the system's recommendation capabilities. It was also possible to deduct that, as expected, personalization based solely on preconceptions is problematic because people do not always match stereotypes completely. Simultaneously, recommendations based on explicit user data are prone to failure since users frequently refuse to express their true preferences or provide the system with incomplete information. Finally, recommendations based on user behavior observation suffer from the so-called "cold start problem," in which it mirrors the user's usual selections and do not support variety in the system's recommendations, so combining three (or more) user modeling techniques improves the system's predictability and fruitfulness [7]. This research also suggests that affective computing could be used in people's daily lives by supporting the user models that describe the user preferences based on their demonstrated emotions when watching certain TV programs or channels. The same study also demonstrates the need for greater research to improve the results of similar technology.

Another interesting work making use of affective computing is the application of this field to help to treat and prevent social and emotional problems, like depression, anxiety, OCD, PTSD, among other disorders that nowadays have become more and more usual due to the increase of pressure on people's lives, making it more stressful, what leads to people developing more mental related illnesses. For example, in the United States of America, mental illness was becoming more common among adults even before COVID-19. In 2017/2018, 19% of adults reported having a mental illness, up 1.5 million persons over the previous year's figures.[3] The pandemic scenario arrived only to boost even more these numbers, primarily regarding anxiety and depressive mental issues as it can be visualized on the Statista graphs present on figure 2.4, according to data compiled by the U.S. Census Bureau and the National Center for Health Statistics.

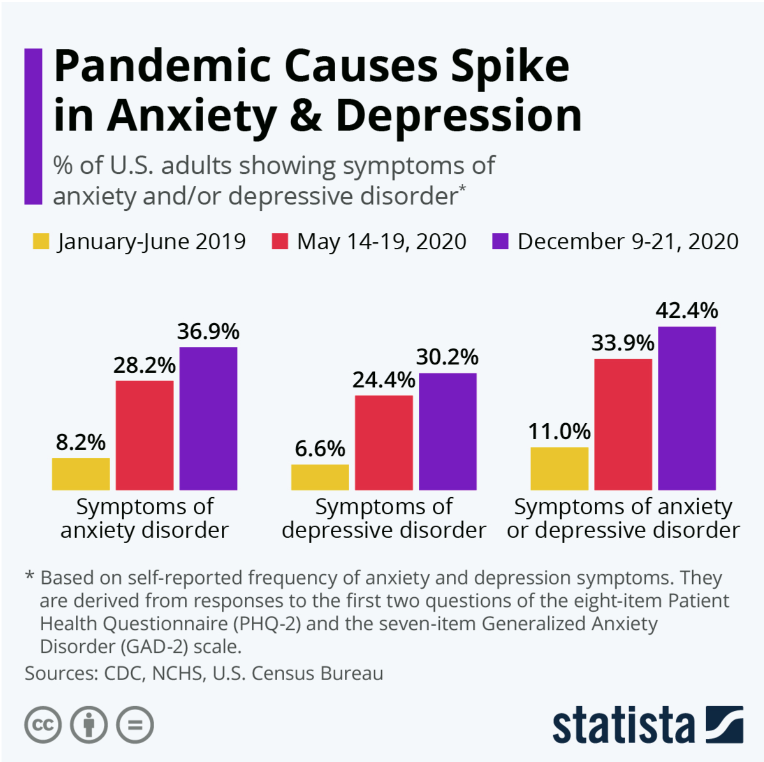

Figure 2.4: Impact of COVID-19 Pandemic on Mental Health.[4]

This reaffirms what was previously addressed, namely, that mental diseases are a threat to modern society, and that action is required to address this issue. This is a case that affective computing is capable of helping to solve. Systems designed to interact with it's users and adapt to what the users report or what the system detects as the user's mental state in order to help them overcome their problems, or even to intervene in much more serious cases. These systems can and must be developed and made available to common users so that it can begin to fulfill it's mission of assisting those in need. There are already some interesting studies that approach this matter like, for example, the ones by Madalina Sucala et al. [5] and by Michael

[4] `https://www.statista.com/chart/21878/impact-of-coronavirus-pandemic-on-mental-health/`

Van Ameringe et al. [6]. In these publications, it is indicated how these apps can assist their users, such as assessment apps, which can be developed with preset algorithms in which participants react to a series of questions based on their subjectively reported symptoms in order to acquire a possible diagnosis.

These apps can also provide therapists with suggestions for possible pharmaceutical therapy alternatives. Tracking apps that allow clinicians and patients to monitor symptoms from a distance using active and passive data collection information. Active data collection requires the user to physically enter the data into the system, which is typically done through surveys, diaries, and subjective mood and anxiety evaluations. Passive data collection, on the other hand, is information gathered automatically by sensors present, for example, on a user's smartphone, such as GPS to track location, accelerometers to track movement and physical activity, and phone calls, text messages activity and microphones to identify social engagement. Treatment apps are a form of mHealth (Mental Health) software that aims to solve the problem of restricted access to specialized mental health care, which is frequently mentioned as a major barrier to treatment. These apps could perhaps theoretically be used as a standalone treatment or as a supplement to more traditional treatments. Several meta-analyses have found that computer-based therapies are useful in the treatment of anxiety and depressive disorders. Finally, there are multipurpose applications, which are said to be the most popular among all age groups and often target various symptoms of anxiety or mood disorders, with their functioning tailored to the user's specific needs and symptoms.

All of these apps offer a variety of benefits and options for people suffering from mental illnesses. For example, tracking apps can assist clinicians by providing detailed information about a patient's symptoms in between appointments. Multipurpose or treatment apps can also reduce the need for in-person appointments with clinicians, which can help to reduce geographical barriers, time, and cost issues and also reduce workload to the healthcare professionals [6]. The app *AnxietyCoach*, for example, has been found to boost treatment engagement when therapist contact is disrupted since it has a built-in exposure recommendation and provides an alternative line for patient-therapist conversation when therapist contact is disrupted.

Apps have shown to be capable of overcoming common barriers to treatment as well as providing care to those who do not have access to traditional treatment methods, making them a good option for all those who, for one reason or another, cannot afford these treatments, such as those with low socioeconomic status, who rely on smartphones more commonly than their counterparts. Because these apps are considerably less expensive than traditional therapy methods, they could be a way of attempting to address the problem of mental illness among low-income people, as it is widely acknowledged that this sector of modern society is the most impacted by those same issues.

Despite the excitement surrounding this topic, both studies emphasize the potential of mHealth apps for both users who are looking for self-help apps and clinicians seeking for apps for their patients, but they also point out that the majority of apps have not been critically examined, are not scientifically supported and that the few existing studies evaluate apps that are rarely available for download to mobile devices. For example, in one of the studies, the authors state that the majority (67.3%) of the currently available anxiety mobile applications were found to lack the involvement of health care professionals during their development, and very few (3.8%) of them have been rigorously tested. While anxiety apps have the potential to improve access to mental healthcare, there is a significant disparity between the abundance of commercially accessible applications and the lack of data regarding their efficacy and effectiveness. Even though applications have the potential to increase access to mental health care, the area is still evolving, and the full potential of apps for treating anxiety has yet to be fulfilled [5].

Chapter 3

Multimodal Analysis Reports Platform

3.1 Personas & Scenarios

Having in mind the idea of describing and characterizing the concept of MARP there were developed some personas and scenarios that correspond to the type of people and kind of problems that this type of platform aims at assisting and solving, respectively. Personas are fictitious characters created to represent and expose the needs of the intended users for a product and the scenarios complement the personas describing how the needs described in each persona can be accomplished.

3.1.1 Personas

MARP is a platform thought to serve a more specific audience as it is the group of people described by the persona Carolina Pinto on table 3.1, however, other possible applications of the platform by different types of individuals, as stated by the other two personas, are depicted here. As a result, these were the personas thought and formed:

- Carolina Pinto is de primary persona of this work, as it is the one that better portrays the platform's ultimate user. She is described in table 3.1 and represents a student of a scientific area that deals with biometric data to perform some analysis regarding human emotion. She wishes to use a platform that allows her to simply input the files she extracts from her experiments and analyze the results in an easy and fast way. Carolina heard of affective computing and since then thought of using a system capable of identifying emotions in her study, in order to correlate this information with the feedback given by the subjects of the study. In order to do as she projected, Carolina intends to equip each participant with biosignals sensors like ECG, EDA, EMG, during the experiment, recording their physiological responses to the stimulus the study presupposes. After capturing all the data, she desires to make a computational system evaluate each one of the participants' biosignals for the two cases of the study. This means that she wants each individual dataset to have the biosignals from the session of the video with neutral sounds and with laughter sounds and then analyze and identify which emotion had the most representation in what the participants were feeling during each of the sessions of the study. Carolina's final desire is to compare these results with the notes that she, as her own, gathered during and after the study with each participant, due to user feedback and also to be able to compare different sessions for different participants, as well as to go a little further and try to identify which features in the agglomerate of the signals used, can be more conclusive about the results she will get.

- André Lopes is described in table 3.2 and represents a researcher who wishes to study companies' employees' productivity, analyzing their physiological reactions to some tasks performed, to best suit each task to the best employee. With this in mind, André is willing to conduct an experiment with the staff of a local company, in which they will be exposed to real-world circumstances while carrying biosignal receivers. They will be proposed to do their daily work as they usually do, but this time there will be a computational system recording their biosignals, in order to further analyze these recordings. After gathering this data, André wants to evaluate it, using a computational system, so that the company, on receiving this information, can make some adjustments in certain situations in order to make sure that their employees have their respective satisfaction levels high. He also wishes to study how changes made according to data revealed by the study can, not only improve individual productivity, but also the general productivity of the group. This general productivity improvement could be a sign that applying the analysis of biosignals to companies' human resources could be something with a positive impact and with potential beneficial applications . André intends to do his research by submitting these recordings with biosignals' data to Multimodal Analysis Reports Platform (MARP) alongside with an Artificial Intelligence (AI) model he thinks has the best identification results for the work he intends to do. This model identifies, by analyzing the biosignals provided, three emotions that can be represented on the subject, happiness, neutrality, and fear. With this analysis, André pretends to compare the results of each analysis with the work that was being performed by the subject to whom the signals belong. With this, potentially interested enterprises can manage the tasks they give to their employees, the work hours each employee must do on a specific task, among other production boosting techniques, all through the analysis of biosignals affective data.

- Ana Marques is described in table 3.3 and is a child psychiatrist that works with individuals that, sometimes, have difficulty in explaining and expressing their emotions. With this, she wishes to use a platform that allows her to study each of the children individually, in order to improve her work with each one of them. Some children are not always "easy to read", depending on the diagnosis they have. This is a barrier that technology can help to overcome since computational systems do not "see" emotions as humans. Human beings make use of a sensory part of the brain to try to detect and perceive emotions, a machine, on the counterpart, does not have this capability. For example, in this case, some of Ana's patients tend to not show many emotional clues of what they are feeling, at least until a point where this begins to be frustrating and then, there is an "explosion" of emotions that sometimes can be difficult to deal with, especially for the children. Here, for example, computational systems that make use of biosignals data to identify emotions have the advantage of analyzing something that humans can not quite see on human-human interactions but that do not lie when the matter is to identify changes in the subjects physiological changes. With this in mind, Ana can equip her patients with some sensors and conduct an appointment as she usually does, but taking notes of the time where she observed some changes. After the session, she can fuel MARP with these signals, as well as a model for emotional identification. Then, according to the results and making a relation between those results and her notes, she can make a more detailed and individualized diagnosis of each of the children she is responsible for. In Ana's case, technology can be of much aid to find the best methodology and/or therapy to address and solve her problems more easily and effectively, also improving her patients' experience and satisfaction.

Carolina Pinto is a single, twenty-two years old student from Porto, living in Aveiro and she is doing her masters dissertation on Universidade de Aveiro's biomedical engineering course. Her dissertation consists in getting physiological data from a group of participants in order to compare people's reactions when seeing the same video, with or without laughing sounds. For example, videos of people falling with and without this audio stimulus. The idea behind this study is to verify if the same visual stimuli when presented to the subject with other stimuli makes the subjects have different reactions. With this study having the potential to then be under more attention in other further complementary studies, to try to give an application to this knowledge, if proven. For Carolina, it is an interesting experience to study the different reactions that subjects can manifest when presented to the same video but with and without another source of stimulus, in her case being an audio stimulus. Being a modern woman with much curiosity for the technological world, she also wants to incorporate technology in this study to help her take faster conclusions of the results she will gather. Carolina has some technical difficulties that sometimes prove to be frustrating and that slow her work rate. This is the problem that she intends to solve with the help of a framework, that can make this specific technical part of her work abstract and invisible from her perspective.

Figure 3.1: Image from `https://pxhere.com/en/photo/108386`

Motivation: Try to discover if sound stimuli are relevant to the perception one has when exposed to a certain situation, by making use of computational systems in her aid in order to boost the study precision and to make the process faster and easier.

Table 3.1: Persona Carolina Pinto, Master's degree student at Universidade de Aveiro.

André Lopes is a forty-four-year-old researcher working at Universidade do Minho. He lives in Braga, is married, and has two daughters. André always thought that technology could be an option for companies not only to perform some tasks but also to improve the quality of work and satisfaction levels of the human workers. He believes that technology can have a major impact on reducing stress and anxiety at work, through some techniques and projects oriented to these issues. André is a researcher that is connected to the affective computing field of study and intends to develop a project where workers of a company have their biosignals captured during the performance of their daily work tasks. To conduct the research, he collaborated with a medium-sized company that produces traditional Portuguese tiles, also based in Braga. This company's workers would serve as the study's participants. Using the biosignals recordings, André's ultimate goal is to study the emotions identified for each employee and associate them with the task they were performing. This association and study could help companies' human resources to better manage their workers, boosting productivity while elevating the employees' satisfaction levels to better standards. What made André develop this idea was that he believes that, if the people that are doing the work are happy and relaxed while doing it, the work will also be improved and that better work will lead to better opportunities for everyone.

Figure 3.2: Image from `https://pxhere.com/en/photo/718635`

Motivation: Study how analyzing biosignals from a company's employees and making adjustments according to the results can have a positive impact on the company's productivity and the employee's satisfaction levels.

Table 3.2: Persona André Lopes, affective computing researcher at Universidade do Minho.

Ana Marques is a sixty years old single woman that lives in Porto and is a psychiatrist specialized in working with children. She has her own private office located on Rua do Bonjardim with currently over fifteen children over her supervision. Ana thinks of herself as a person always open to technological innovations, she tries to keep track of technology advances, especially with innovations among the affective computing branch. Ana thinks that the world is progressing into a more technological and connected globe, and instead of thinking of technology as a treat, she prefers to view it as a resource, something that she and others can and must use, in order to better their work and lives. She defends that technology can help professionals of her field of study, to carry out more targeted analysis and with the greatest number of data and certainties about the emotional state of their patients since in her specific case it is not always easy to identify her patients' emotions.	 Figure 3.3: Image from `https://pxhere.com/pt/photo/627407`
Motivation: Help her patients with the aid of technology, allowing her to achieve the goal of making her work better, easier in some specific cases, and making it's process more certain and efficient.	

Table 3.3: Persona Ana Marques, children psychiatrist.

3.1.2 Scenarios

Using the technology to make academic studies

Carolina is expecting to find some interesting results in her experience. If the results of her study show that, for example, the sound of laughter can affect a person's interpretation over some kind of stimulus, her main idea is that in the future by applying this idea she can discover a method to treat certain traumas/fears by exposing people to images of their traumas/fears with joyful sounds and slowly change their perspective on that matter. With this goal in mind, she gathers a group of twenty-five people and records their biometric data through a session where they observe videos of people falling with and without laugh sounds. After this she wants to know what kind of emotions her subjects had when seeing the videos, so she inserts the data of the subjects on the platform as well as an AI model trained for detecting emotions. After submitting all the requests, Carolina starts to inspect every participant's report results, with graphs representing these results of the evaluation in different ways so that the interpretation of the data can become easier to perceive, and also analyzing the features of each signal used on the evaluation. This analysis allows her to deeply analyze each session of her experiment. With this technology, in addition to the time saved in all the process of analyzing results, Carolina also made usage of the capability that the system has for processing the biosignals files in many formats, saving her also the time and work of having to treat the files in order to use them. Alongside this, she also makes use of the comparing feature of the system to compare results among some subjects, by selecting two of the requests already made. With this platform, Carolina was able to boost her productivity and eliminate some technical barriers that could emerge, when analyzing the results of her experiment.

Using technologies in order to manage and monitor human resources

André is a person that thinks that the state of mind of a company's employees is a major case that leaders need to have in mind. This, in order to boost production and keep the satisfaction levels high, among all the working staff. With this technology he can, after recording the biosignals of the employees that belong on his study's group, during certain activities, analyze the final report containing the emotional identification results. This, in order to try to find anomalies in the results so that there can be made changes or adaptations to counter these problems. First, as said before, there must be recorded the subject's biosignals while they are executing the tasks that are necessary to evaluate. So, André equips the employees, subjects on the study, with EMG, EDA, and ECG sensors and start recording the signals on his favorite software. After gathering the data, the program extracts it into a file. André logs in into the platform and goes to the "submit files" page to upload this file with the information about the subjects to the system along with the subject's metadata information as a JSON manifest file, and the artificial intelligence model which will evaluate the signals. After uploading all the data, an analysis must be requested to the system, so all the characteristics of the study are filled and the analysis requested. When this analysis finishes the results can be accessed and analyzed in order to then take conclusions about the experiment. With this type of information, the company's responsible for the employees can be informed of anomalies detected and make adjustments, in order to, ultimately, boost their business' productivity.

Using technology to obtain and analyse the report generated

Ana, like many professionals from the field of psychiatry, sometimes finds it difficult to "read" or interpret some of her patients, since they are children, some have special difficulties when manifesting what they feel so, sometimes, the task becomes even harder. With this technology, it is possible for her to receive a report that is based on the analysis of the biometric data of her patients generating their possible moods/emotions. To do so, after her patient is in the room, Ana equips them with the measuring equipment necessary and starts the recording while she conducts her appointment. When it finishes, Ana stops the recording, extracts the data, and fuels MARP with the files generated with the information about the patient biosignals. She also submits an artificial intelligence model that is supposed to have better results for her study. After this, she goes to the "New Request" page, fills out all the necessary information, selects the files she just submitted, and clicks on "request". She goes back to the home page and sees that her request is being processed. After a while, she sees the state of the request she has made is "SUCCESS" and starts to analyze the results of that request. She now clicks on that task and decides to analyze it's results in the form of a chart that informs about the most relevant emotion through the experience, from the three base emotions: happiness, fear, and neutrality. This can help Ana to carry out her work in a more targeted way and to eliminate some errors during the patient's diagnosis, which could lead to therapies that do not suit the needs of the patient if their emotions were misinterpreted.

3.2 Requirements

Taking into account the three scenarios as the perspectives and ideologies that gave birth to the whole concept of MARP there were idealised some main requirements for the initial version of this system.

The requirements considered have different priorities according to their importance to the platform as a product and/or as a service. With all this in mind, there were developed the following requirements listed on table 3.4.

Priority	Requirements
1.	1. Run multimodal analysis 2. Upload files for analysis 3. Allow the analysis with AI models trained using Random Forests and Neural Networks 4. Draw graphs with the results from the analysis 5. Manage users' sessions and workspaces
2.	1. Run feature extraction followed by analysis to the same given files 2. Allow the submission of the files containing the signals data in more than one format 3. Compare results between two different analysis 4. Plot each signal's selected features values
3.	1. Allow analysis to files with a variable number of sessions 2. Draw graphs with the results of the analysis in more than one plot type 3. Allow users to provide JSON manifest files to describe each file's subject 4. Give feedback to the user about each request's state and other relevant information

Table 3.4: MARP's system requirements.

With this list of requirements, one can have a more targeted idea of the platform's ideas and objectives. These requirements were implemented according to it's priority to the system, taking into account how fast they could be implemented, taking into account the state the platform was at the moment, and considering the resources available. This means that even if a requirement is listed as one of top priority, this does not mean it was developed and/or implemented into the platform first then one of a lower grade priority. The table 3.4 only lists requirements in their order of importance for the concept of the platform.

Chapter 4

System Design

The platform presented in this work aims to make up for the lack of interfaces that provide an intuitive analysis of biomedical information as a means of evaluating a person's condition. Given the vast applications that this platform can have, it is important to design a system capable of effectively communicating between all components. To make all this happen, the communication among these several components of the system must be specified so that it becomes simpler to understand how the system platform works and operates.

4.1 Platform Architecture

MARP can be divided into two main components, the front-end, and the backend. Each one of these components is responsible for specific and important tasks. They both help to build the system as a whole and work in constant communication in order to make the bridge between the users and the background processing of the information.

On image 4.1 it can be observed a visual representation of MARP's system architecture. All the components that will be described can be seen making part of this image, as well as some other components that were mentioned when describing the workflow of the platform. It is possible to see how every component of the system is connected with the others and with this better understand how the platform works and performs it's operations.

The front-end is formed by the web interface of the system. This web client sends requests to the backend of the system and receives it's answers, to which it will apply some transformations in order to provide these answers to the user.

The second main component of the platform is the backend. This is formed by the agglomerate of other, more specific components, that execute all the processing necessary for the platform. The backend is formed by four main parts that have an important role in executing and registering all of the platform's actions, these components are the API component, the processing component, the message broker component, and the database.

The Application Programming Interface (API) component is the main component of the backend, it is seen as a command center of operations between all these other components of the backend, and it is this component that communicates with the front-end web application during the exchange of answers and requests. It is on this component that are implemented and described all the platform's features. It is responsible for receiving requests, starting their processing asynchronously, through the communication with the processing component, and then returning the answers to these requests. In the case of some small actions that do not require to be passed to the processing component due to it's minor processing requirements, it is also this component that executes those tasks.

The main processing component is the one that, as said above, makes use of an asynchronous task queue and it's tools to process all requests made to the system. This component is very important because it is used alongside the API component to process the requests and enable the system with asynchronous processing of those requests, allowing the server not to be fully dedicated to only one request at a time. With this component, the server is able to receive and process several requests simultaneously, giving the platform some abstraction in terms of how it manages and processes the requests but at the same time giving a good amount of information about each request made, so that the system can always be monitoring and have the information about what is happening with each request.

MARP also makes use of a message broker component for communication between the task manager and the control unit of the platform. The usage of this component requires for it to be a lightweight message broker that works alongside the processing component, for messaging exchange between the processing part of the system and the API. This makes that the communication between these two components when exchanging the analysis requests, states of every request, or even the responses to certain requests are made correctly and rapidly.

The fourth main component (tat can be divided into two separate blocks) present on the backend, is the database component and the storing unit. This is database is the data bank that contains all the relevant non-volatile information of the platform. In this component, there are database tables with the information of the platform's users, the requests to the system, and other two tables that are important to the processing component, since some technologies that are implemented on this component use them for internal organization and orchestration of tasks. The storing unit is related to data management, for bigger blocks of information as AI models, datasets, among others. This type of data is not stored on the database but on this storing unit.

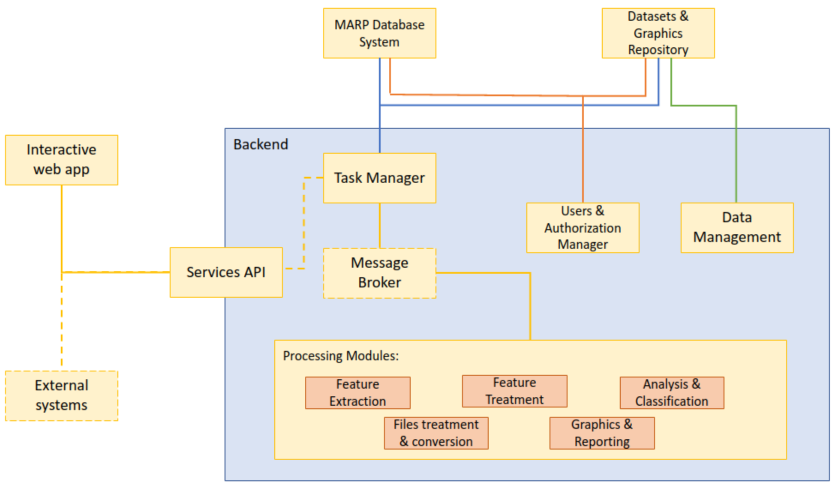

Figure 4.1: MARP's architecture visual representation.

4.2 Components Interactions

All these system's components make use of specific technologies and have specific interactions among themselves, that is what makes possible the whole concept of the platform. The technologies used on these components are explained later in section 5.1. There are several communications on this system that are essential for the entire platform to function properly and that require supplementary explanation to completely comprehend the system's behavior. The image present in figure 4.2 represents these components with their associated most relevant technologies and the connections that exist among them.

First, since it is part of the interaction that makes possible the communication between several other components, there is the interaction between the API and the platform's message broker. As it will be further explained in the next paragraph, this connection is what makes possible the communication between the processing component with the other relevant components, via API information forwarding. It is through this message broker that the components have a fast and lightweight message channel for them to communicate and exchange information. So, this connection between API and broker is very requested during the execution of the platform's tasks. This message broker implements RabbitMq which uses as a messaging protocol the Advanced Message Queuing Protocol (AMQP), version 0-9-1. AMQP 0-9-1 is a binary protocol that defines strong messaging semantics. For clients, it's a reasonably easy protocol to implement, and as such, there are a large number of client libraries available for many different programming languages and environments[31].

One of the most important interactions between components of this system is the interaction between the API and the processing unit services (through the message broker). This connection and interaction is what makes possible the communication between the processing component with the system's database and API data. For example, in the case where the processing component has the necessity of accessing data on the database, the API accesses this information directly and then uses the broker to send it to the processing component. As it was possible to observe from image 4.1 the processing unit is the one that contains the processing modules where, after receiving the information about a request, this will be executed. Using this connection, the API, upon receiving a new request of analysis by the user, sends it to the processing component where it will be handled and executed in a concurrent context among other requests. To each of the requests made to the platform, the processing component provides a good amount of valuable information about each of these requests, to the whole system so that, once it ends, the API component can acknowledge that and act upon such event.

Another group of interactions that are of major importance for this platform are the ones between the platform's API and processing component, with the system's database and files repository. These interactions are vital for a correct execution of the platform since it is on this database and repository that is stored all the vital information about the system's users, requests, datasets, etc. These connections make it possible, for example, the existence of multiple authenticated users on the platform, the possibility of keeping track of each one of the multiple requests made to the platform by each one of these users, among other features of the platform.

The connection between MARP's API and the web application is also an important part of the system since these connections can be seen as the bridge between the backend processing and the front-end pages shown to the user. Using the provided web platform the user can make requests to the API where the actual processing of each request and action takes place.

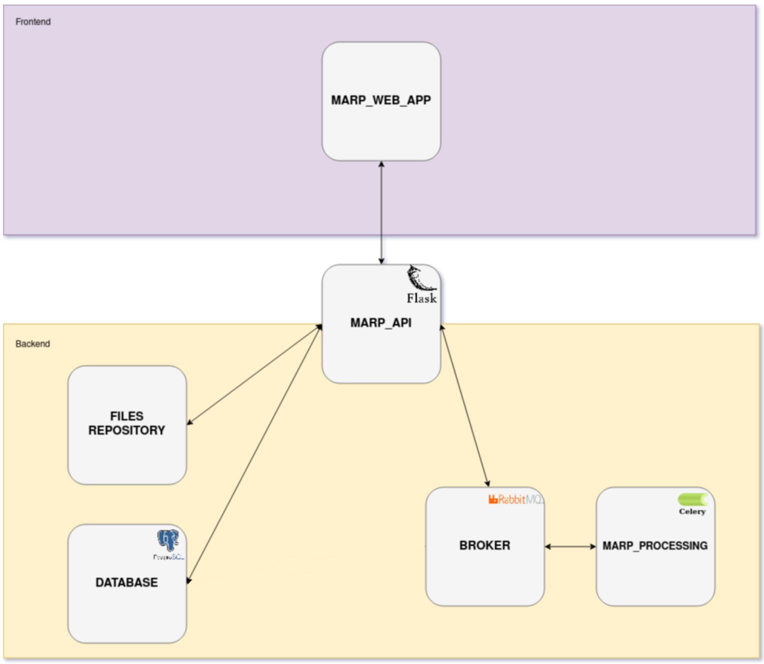

Figure 4.2: MARP system components interactions.

The connection between these two components is what makes this possible and is a constant connection of exchange of information.

All these interactions have critical importance on the outcome of the system and are what enable it to work as projected. It is thanks to these interactions that the information flows through all the platform and is always available to each component when it is needed. As previously mentioned a visual, simple representation of these interactions can be observed from figure 4.2 where all the previously described interactions are represented with connections between the components.

4.3 Domain Model

On this platform, there are some concepts that must be explained. The Multimodal Analysis Reports Platform was designed to serve people like the personas mentioned in section 3.1.1, especially Carolina, the one identified as the primary persona for this work. This means that the system is prepared to overcome some failures but it is designed for a type of user that knows how it should work and that is informed on the requisites for the platform to work as intended.

Users, on this platform, are the actors that register into the platform and that will perform actions on it. They have full access to all the functionalities of the platform by interacting with the MARP's web application. This interaction between MARP and users is mainly made by the request of biosignals analysis by the users, the execution and response to these requests by the platform, and the visualization of the results of these analyses made by the users.

On this platform, the requests of analysis made to the system, are mainly described by having obligatorily a dataset containing the biosignals values, a AI model which will evaluate the signals and a manifest file in JSON format with the information about the subject being analyzed. These requests are part of the fundamental element of the platform. They are the specification of an analysis that is requested to the system.

The analyses that the system performs, are the other part of the fundamental element of MARP. These analyses consist of the results obtained from the application of the AI model to the biosignals file(s) specified and described on the request.

The datasets with the above-mentioned signals are the source of data for and analysis. It is on these datasets that the biosignals data is represented. These files are somewhat flexible. The signals that the system can analyze and interpret include EMG-MF, ECG, EMG-Z, EDA, and EMG. Each dataset can have from one to three sessions of data corresponding to each signal.

The platform also allows for the submission and analysis of these dataset files in various formats, such as:

- TXT format files;
- CSV format files;
- Pickle (PKL) files.

Every analysis on MARP is described by being analysis that require feature extraction or not.

Files that require feature extraction

The following file extensions correspond to data files submitted where the user wishes to perform a feature extraction before submitting the result to evaluation by the model provided. All these files are feature extracted and then, the result of this operation is stored in a PKL file (that will be explained next) with the name provided by the user upon the description of the request.

TXT files are acceptable with a text organization similar to the one produced by the Opensignals software. Typically, these files contain all of the signals values from a single session in one file, so one to three of these files can be submitted, this representing the one to three possible sessions of data per analysis.

It is also possible to provide the dataset as CSV format files, with each signal separated into it's own file. Also following the one to three sessions of data rule, on these files, the data must be organized horizontally with the first element of each row being the ID associated with the subject whose data is on the file, an underscore and a letter from "H", "F" and "N" that name the sessions (for example: ID2_H). Each of these files must have up to three lines, meaning that each line refers to a session of data and that each file has on it the information about every session for that signal. This type of file can have, in addition to the ones describing the biosignal's data, a file containing the information about the request's triggers per session. Triggers are the values that indicate the start and end of the data on the other files. This file must have the name "Triggers.csv" and it's content must be similar to the other CSV files, with the first element of each row being the subject identifier as described above, and then only two values among two different columns that indicate the start and end of the data to analyze from the other CSV files, per row (and by this, per session).

Files that do not require feature extraction

If there is no need to extract features, this means that the user already has a file with this information, maybe from a previous utilization of MARP's services. The system in this cases can analyze Pickle (PKL) files. The internal organization of each of these files is a data structure formed recurring to Pandas'[1]data frame structure with an organization identical to the one that was used by the datasets provided to this study, developed and described by Gisela Antunes et al. [11].

Artificial Intelligence (AI) models

The models mentioned across this work are AI models which were trained using one of two machine learning algorithms that are described later in this section, and that are used to identify certain characteristics on their input in order to produce a result. On MARP's case, the result produced by these AI models is the identification and classification of emotional presence using as input biosignals' features. These models that the user submits to the system, must always be compatible with the dataset. This means that trying to predict results from a dataset with only EDA related data using a model that was trained using only ECG signal, will result in an error. This means that it is the user's responsibility to determine which model may be applied to each dataset. As an example, if the correlation is applied incorrectly, the system displays an error notification suggesting that this may be the case, but it is unable (and should not be) of resolving the issue. Two machine learning techniques have been used to support analysis on the platform: neural networks and random forest.

Neural Networks (NN) are a subset of machine learning and are at the heart of deep learning algorithms. Their name and structure are derived from the human brain, and they mimic the way biological neurons communicate with one another. NN are comprised of node layers that contain an input layer, one or more hidden layers, and an output layer. Each node, or artificial neuron, is connected to the others and has a weight and threshold linked to it. If a node's output exceeds a certain threshold value, the node is activated, and data is sent to the next tier of the network. Otherwise, no data is sent on to the network's next tier. Training data is used by neural networks to learn and increase their accuracy over time. However, once these learning algorithms have been fine-tuned for accuracy, they become formidable tools in computer science and artificial intelligence, allowing us to effectively and rapidly classify and cluster data [32].

A Random Forest (RF) is a machine learning approach for solving regression and classification issues. It makes use of ensemble learning, which is a technique for solving complicated problems by combining several classifiers. There are several decision trees in a random forest algorithm. The random forest algorithm determines the outcome based on decision tree predictions. It predicts by averaging the output from various trees. As the number of trees in a forest grows larger, the generalization error converges to a limit. [33][34].

As previously said, on it's current version, the platform supports these two machine learning algorithms.

[1]`https://pandas.pydata.org/`

JSON manifest file

JavaScript Object Notation (JSON) is a data-exchange format that is straightforward. It is easy for humans to read and write and it is easy for machines to parse and generate data in this format. JSON is a text format that is entirely language independent but that follows conventions recognizable to C-family programmers, including C, C++, C#, Java, JavaScript, Python, and others[35].

Due to these characteristics, JSON was defined as the format of the manifest file to be used on MARP. A manifest file is a file used in systems that describes the information about it's content, on this platform's case it is used to describe each individual to whom the dataset with the signals data is associated. This manifest can be supplied to the system via a JSON file with a specific organization and fields or via a JSON array created by the system with direct information acquired by the platform's web application, which the user needs to fill out in order to submit an analysis. The structure of this JSON manifest file is described on the platform's help page and, on this document, at appendix C where it is present an example that represents this same structure.

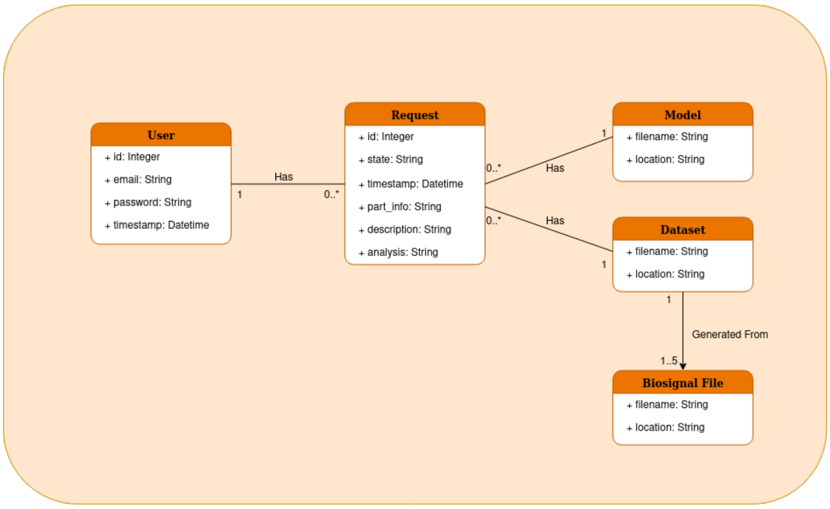

Figure 4.3: MARP User-Request interactions and associations.

To better understand how the majority of these elements interact with each other and the type of association that exists between them, figure 4.3 is present, representing this. The figure makes it possible to better comprehend that on MARP, each user can have associated with them any number of requests and that each request has, obligatorily, associated with it an artificial intelligence model and a dataset. It is also possible to deduct that both an AI model and a dataset can be associated with any number of requests, meaning that they can be used on multiple different requests. The last aspect to extract from this figure is that a dataset is generated from a variable number of biosignals files, as it was described earlier in this section. This is due to the possibility of being necessary to make feature extraction from the files submitted or not. For the case where there is this need, if the biosignals files come in TXT format, the number of files is one. For the case where there is also a need for feature extraction, but the biosignals files come into CSV format, the number of files can be from one up to five. In the case where there is not necessary to perform feature extraction, this means that the file with the data comes in PKL format, so the number of files is also one.

Chapter 5

Implementation

5.1 Implementation Technologies

MARP is a system that was developed with the usage of some well established and useful technologies that have support libraries for python. This technologies are used among the different components of the platform, in order to perform the tasks these are intended to. These technologies are:

- Flask;
- Celery;
- PostgreSQL;
- RabbitMQ.

Flask

Flask is a python-based lightweight WSGI web application framework. It is referred to as a microframework because it does not necessitate the usage of any specific tools or libraries. A Web Server Gateway Interface (WSGI) is a specification, it describes how a web server interacts with web applications and how many web applications may be linked to process a single request. Flask is designed to assist users to get started rapidly while also allowing them to scale up to more complex applications. The developer may use whichever tools and libraries they want with this framework, which was incredibly useful on this project since there were so many python libraries to evaluate. Many of these extensions are used on this platform in order to achieve the desired goal of the web application. Flask is well-known for the large number of libraries it supports, which allow the user to tailor how certain features of their system are handled [36].

Celery

Celery is a simple, flexible, and trustworthy distributed system for processing large volumes of messages that may run on a single machine, numerous machines, or across data centers while giving tools that operations need to keep the system running. It is a task queue that prioritizes real-time processing while simultaneously allowing for task scheduling. Task queues are used to divide work across threads. A task, which is Celery's work unit, is the input to a task queue. Dedicated worker processes continually check task queues for new work to finish. Celery uses messages to communicate with clients and workers, with a broker acting as a middleman, on MARP's case RabbitMQ, which is explained below. To start a task, the client adds a message to the queue, which is then delivered to a worker by the broker. Multiple workers and brokers can be used in a Celery system, allowing for high availability and horizontal scaling [37].

The figure 5.1 visually represents the job workflow performed by a system configured with celery, portraying the communication between producer and consumer through a broker to whom the consumer must subscribe. Communication is made through the exchange of JSON format messages, which aids human comprehension of the messages that are being exchanged by both parts.

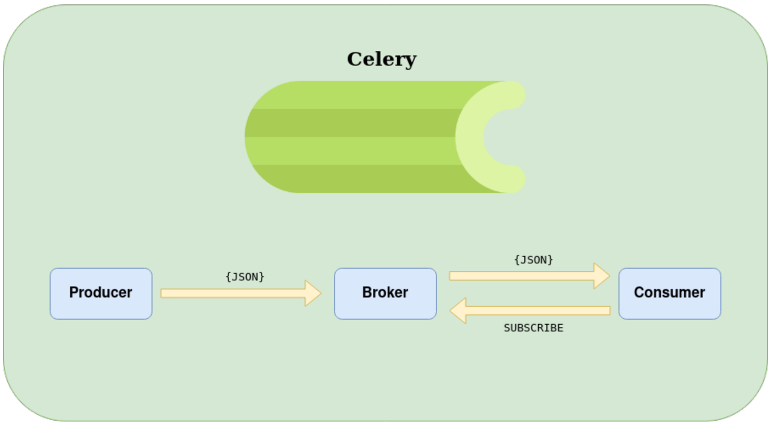

Figure 5.1: Celery job workflow visual representation.[1]

RabbitMQ™

One of the most widely used open-source message brokers is RabbitMQ, which supports numerous messaging protocols. It is lightweight and easy to deploy. Put simply, it is software that defines queues to which programs subscribe in order to send a message or a series of messages. Any type of information can be included in a message. It can, as it is on MARP's case, contain information about a process that should begin on another application, or it could simply be a text message, there is a great variety of possibilities for this messages' content. The messages are stored in the queue-manager software until a receiving application connects and retrieves a message from the buffer. The message is subsequently processed by the receiving application. Message queueing allows web servers to respond to requests quickly instead of performing resource-intensive procedures on the run, which can create delays. Messages are not directly published to a queue, instead, the producer sends them to an exchange. With the help of bindings and routing keys, an exchange is responsible for routing messages to distinct queues [38]. Figure 5.2 shows an image where it is represented the exchange of messages between a consumer and a producer through a RabbitMQ broker.

[1] Based on: https://www.toptal.com/python/orchestrating-celery-python-background-jobs

[2] Based on: https://www.cloudamqp.com/blog/part4-rabbitmq-for-beginners-exchanges-routing-keys-bindings.html

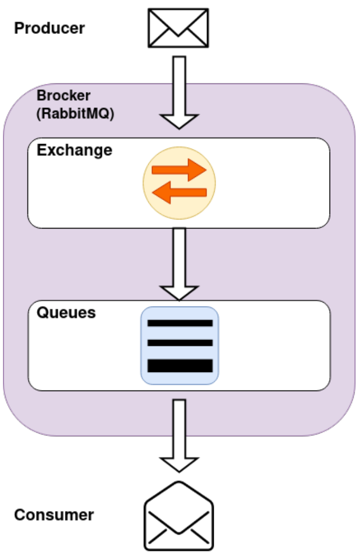

Figure 5.2: RabbitMq exchange of messages visual representation.[2]

PostgreSQL

PostgreSQL, commonly known as Postgres, is a Relational Database Management System (RDBMS) that focuses on extensibility and SQL conformance. This is the RDBMS that MARP uses to assign all of the essential databases for the system's full functionality. SQLAlchemy, a Python SQL toolkit and Object Relational Mapper, is used on the Flask component to connect with the rest of the system. It is defined as the Python SQL toolkit and object relational mapper that provides application developers the full control, power, and flexibility of SQL.

These were the technologies that served as pillars for the different components of MARP to work, communicate and produce results all together as a system instead of isolated modules.

5.2 First Prototype Deployment

To make it easier to deploy the platform, Docker was used, more specifically Docker-compose. Docker makes that repetitive configurations can become automated, this is a feature that is very useful in systems like MARP, that have multiple components that interact among themselves which require initial configurations before starting the whole system.

Docker-compose is a Docker tool that allows to create and run multi-container Docker applications, making use of a YAML file where these containers are specified and can be configured. A Docker container is a standard unit of software that packages up code and all it's dependencies so that the application can run quickly and reliably from one computing environment to another.

Each one of the MARP's components mentioned on section 4 dockerized. The image on figure 5.3 represents how MARP was organized in order for it to be dockerized.

In this image, the rectangles with other two little rectangles on the left side represent the services declared on the docker-compose file. These are the components of the platform

[3]Generated with: `https://github.com/pmsipilot/docker-compose-viz`.

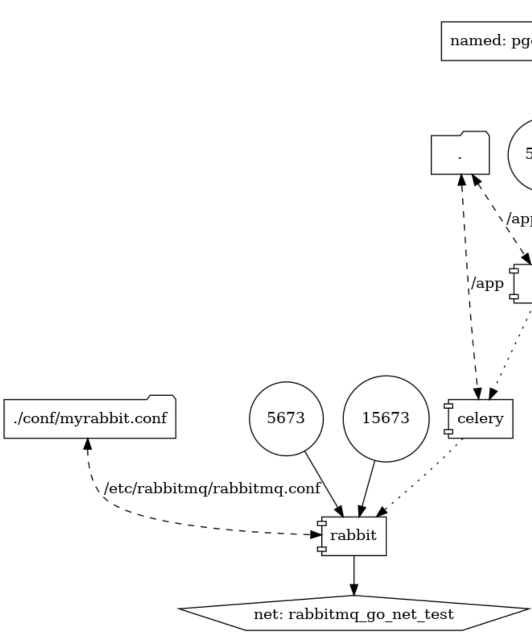

Figure 5.3: MARP's docker-compose architecture visual representation.[3]

already discussed in section 4, where "web" is the web application component, "api" is the API component, "celery" the processing component, "rabbit" the messaging broker, and "db" the database.

The circles, that some of these have associated with them, represent the ports that these services are exposing so that they can communicate among themselves and to the exterior. Some of these services have dashed arrows pointing to other services. This means that one service depends on the one it is pointing at, meaning that it will only be initialized after the one that it points to is already up and running. This dependence among services serves as an assurance that all components are started in the right order so that the system is deployed correctly.

The folder-like figures on the image that some services are connected to by double-sided dashed lines, represent Docker volumes. These volumes are the preferred mechanism for persisting data generated by and used by Docker containers. In the case of "api" and "celery" services these are pointing to a folder that is tagged with a "." caption, this is because these two containers store persistent data generated by them on the current folder where the docker-compose file is stored, that consists of the MARP's project repository. The captions that are adjacent to these double-sided dashed arrows connecting the containers and the folder figures is the name of the path where these volumes are mapped inside the container.

The "rabbit" container also has a straight line arrow connecting it to a figure with the tag "net:rabbitmq_go_net_test" this element of the figure represents a docker-compose network. These networks act as an abstraction component that, depending on the configuration, manage the access to the containers inside this network. All other containers do not have a specified network and can be accessed using the ports they have attributed to them but in order to make MARP work on a server where another RabbitMQ container is already running and using this technology's default ports. In order to MARP's message broker component to be able to start, operate, and communicate with other containers when deployed on an environment like this, the mentioned readjustment was mandatory.

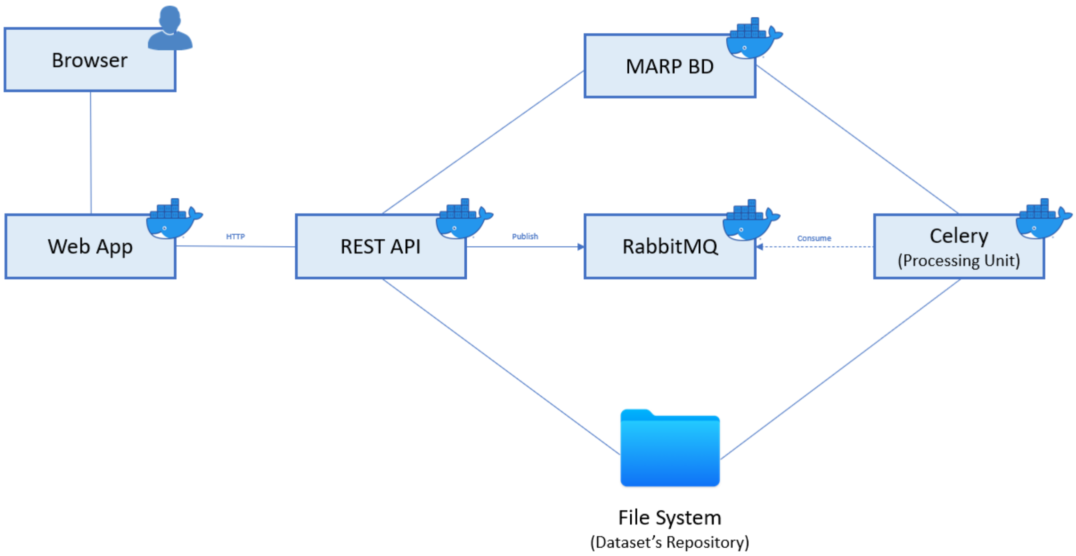

Figure 5.4: MARP's docker containers interactions and structure.

With the image present in figure 5.4 it is possible to analyze a more simplified vision of the final state of the platform's docker containers and their interactions. This image represents the whole MARP's container system, which works with cooperative containers that perform specific tasks in order to make the system operate as a whole, as idealized and designed.

With this configuration, the deployment of MARP in new environments becomes an easy process, different from what it would be if it was done by a manual set up of all system's components and intervening parties. With this "dockerization" of the platform, all it takes is to have the project's repository on the machine it is pretended to run MARP's server and run the "docker-compose up" command. After this, all the problematic questions that could appear on this stage, like version errors, due to the vast tools used on the project, are surpassed and it is possible for the system to be easily used on different systems and machines.

5.3 Backend Implementation

On MARP's backend, many important components will communicate among themselves to perform the numerous tasks that this platform offers. In order to better understand how these operations are done, it is important to know and understand each one of these components. After knowing the backend components, there is necessary to learn about the system's endpoints, to understand how the requests are made and what action each of these endpoints executes.

5.3.1 Backend Main Components

The Multimodal Analysis Reports Platform is made out of a backend module that has a REST API as it's core, which communicates with a front-end module with a web application implemented, allowing the user to conduct a range of options mentioned in the section 1.2 as being the main objectives of the platform. This API is made up of a number of components that work together to create a system that can perform the tasks it was designed to do. Although an overview of these components and interactions was provided in section 4.1, they will be further investigated and discussed in this section.

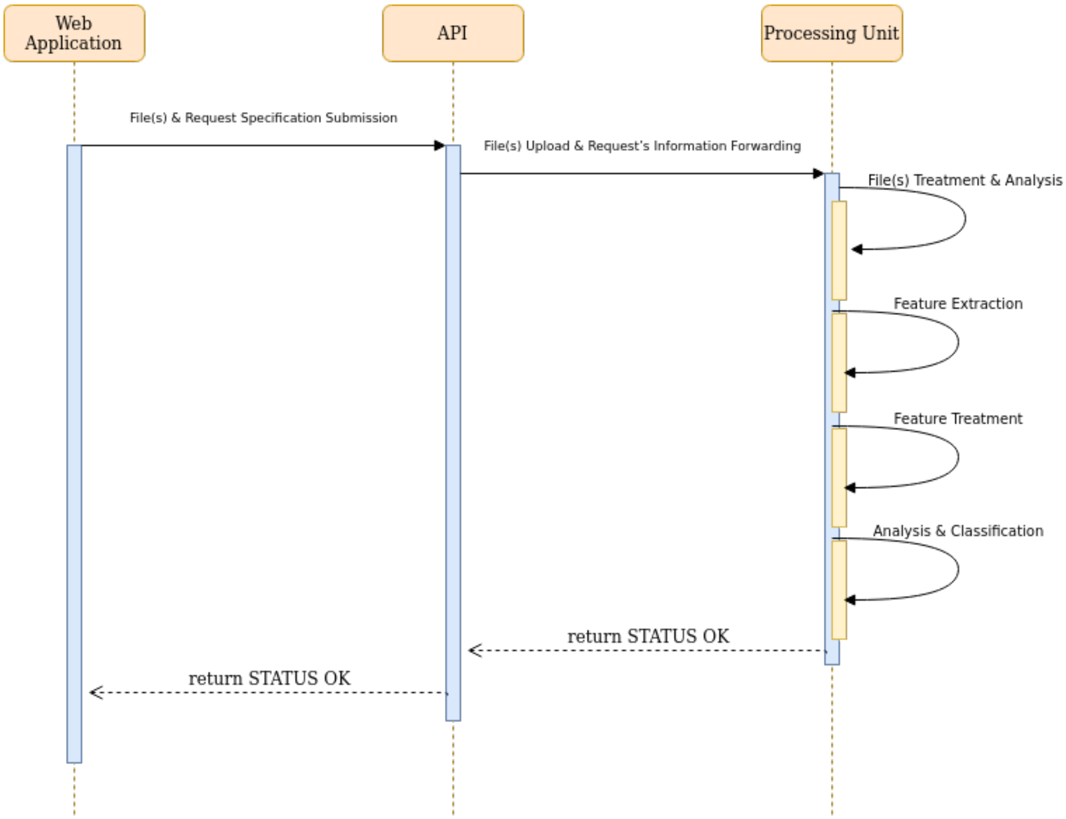

Figure 5.5: Sequence diagram representing the request of an analysis on MARP's system.

For example, on image 5.5 there is a sequence UML graph where it is possible to verify a simplified interaction between three of the main components MARP has and that are described below, in order to perform a simple request of analysis to the system. This allows to understand the actions taken by each part as well as the interactions each component takes with the other and the consequences of those interactions. It also permits to see clearly that the system works as a pipeline system where actions take place in a sequential order among several parties.

API Component

There is a particular component on the platform that serves as a center of operations, where all of the data and actions handled and executed by the system are processed and then presented to the user as a response. The REST API is this component. API's can be thought of as a middleman between users and the resources or online services they want to work with. To look at it in another way, if there is a need to interact with a system, to retrieve information, or execute a function, an API can assist that need by describing the needs to that system so it can comprehend and fulfill them. It's also a mechanism that organizations use to share information and resources while maintaining security, control, and authentication, determining who gets access to what. REST is not a protocol or a standard, but a collection of architectural constraints.

REST can be implemented in a variety of different ways by API developers. A RESTful API transmits a representation of the resource's state to the requester or endpoint when a client request is made. JSON, HTML, XLT, Python, PHP, or plain text are some of the forms in which this data can be given via HTTP. JSON has been the most widely used programming language because, as it was described in section 4, it is language-agnostic and understandable both by humans and machines [39].

The API component is at the backbone of MARP, it is this component that manages and connects all other components, allowing them to work together and synchronizing them in order to fulfill the user's requests. Almost everything is handled by this component, from authentication and user sessions, processing requests (together with Celery), having descriptions of database tables to producing results and communicating them to the web application. This component can be thought of as the system's "heart", where everything is choreographed to achieve a single purpose. All of the platform's activities or features are documented and implemented on this component. This component makes use of Python's Flask functions and tools to manage a REST API. With this, it was possible to enable this component of connecting with the other elements of the platform making it manage and serve every request and task the system has. In order to make these connections between components, there are some environment variables that Flask needs that must be specified at the beginning of the code so that the paths associated with these environment variables can be used by Flask library to mount it's structure. Since this component makes use of Flask and it does not handle the performance of concurrent tasks by default, it is for this reason that the following component was implemented.

Processing component

This component instantiates Celery that adds to the platform the ability to perform concurrent request processing. This characteristic allows MARP to have numerous users using and making requests to the platform at the very same time, with the server processing these multiple requests simultaneously. The platform gains desirable scaling and abstraction as a consequence of this. Celery has tasks to execute, and in the MARP case, the Flask API component commands Celery to perform these tasks. For this component to be operational it must have present two others that sustain it's operations: a message broker and a database. These two must be part of the whole system for the component that uses Celery to work, having with this a relation of dependence relative to those two components.

It is here that the system's next components, first the RabbitMQ message broker, which facilitates all message exchanges between the API and Celery, and secondly the MARP database where the information is stored, are introduced and will be addressed on the next sections.

Message broker component

RabbitMQ is the technology implemented by this platform's message broker. It handles message/request exchange between Flask and Celery and (indirectly) between Celery and the database on MARP's system. This simplifies the process of dividing work, allowing for the parallel processing of multiple requests while also making communication of the same work much easier, faster, simpler, and perceptible for the developer.

As it was mentioned before, on RabbitMQ an exchange is responsible for routing the messages to different queues with the help of bindings and routing keys. The binding necessary for the exchange of messages on MARP is defined at the beginning of the code where RabbitMQ configuration is described. Since MARP only has the necessity to use one queue (that will connect the API with the Celery component, handling the requests processing), this binding makes the exchange of messages possible for this system.

Database

All of the databases required for the platform optimal operation, such as the users table, requests table, and celery management tables, are hosted in databases managed by PostgreSQL. These tables contain all of MARP's non-volatile information, which is regularly created, analyzed, and updated. Messages to create new users, check logins, check if a user is legitimate, verify user's requests, request's status, and other information verifications are sent directly from the API to the database. Despite keeping all of this critical information, the files submitted by users to the platform are not stored on this database, instead, they are stored separately for each user on the platform's repository file system. The images created by the platform for visualization of specific requests are not stored in the database either, they're saved alongside uploaded files, but they're deleted as soon as a new request for graphic information is needed, both to save server space and because the generation of these graphs is a low-cost computational process.

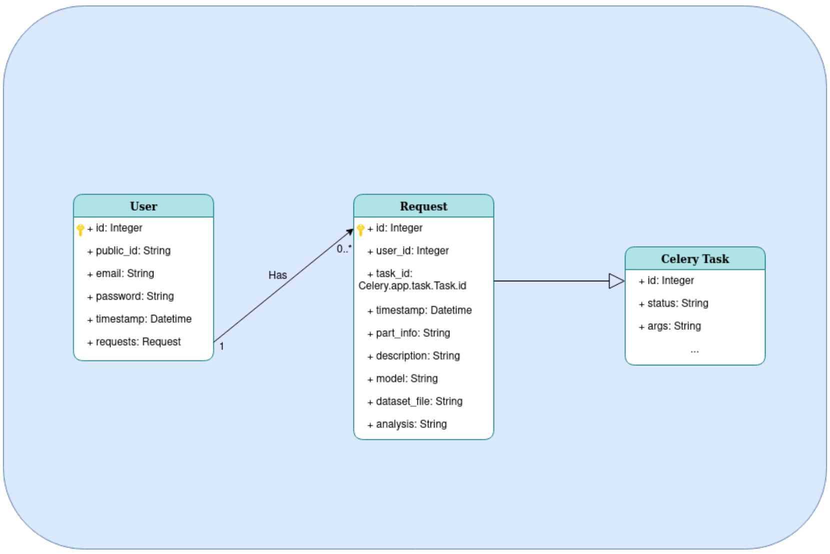

Figure 5.6: MARP's databases.

The database tables are related to each other, having relations between them. As it can be observed analyzing the image from figure 5.6 there is a connection in order to associate users with requests, since the platform needs to keep track of all the requests related to each user of the platform. Also, there is an inheritance of Celery task's variables by the MARP's requests in order to give the later specific information that these tasks have, and that are helpful for the platform.

5.3.2 MARP API Endpoints

MARP's API has it's endpoints exposed so that potential users can access it's services. It is through these endpoints that MARP's web application interacts with the platform so, these endpoints must be specified and described for external systems to be able to understand what they do and use them.

To comprehend MARP's endpoints, table 5.1 describes the goal of each endpoint and lists the name of the endpoints, the respective method used, and the parameters taken by that endpoint. This table also contains a column with the description "Secure?" which informs if that endpoint requires a session token validation in order to be executed. In other words, informs the endpoints that can only be used by users that are already logged into the platform. On table 5.2 it is described the output produced by those same endpoints.

Path	Method	Description	Path Parameters & Request Body	Secure?
/login	POST	Logs in the user into the system.	- Request Body: • email:string - User's email. • password:string - User's password.	-
/register	POST	Registers user into the platform.	- Request Body: • email:string - user's email; • password:string - User's password; • password2:string - User's password.	-
/requests	GET	Fetches the user's requests.	-	Yes
/requests/req_desc	POST	Submits a new request.	• req_desc:string - String(JSON) describing the req.	Yes
/plotdata/task_id	POST	Plots the data of a request.	• task_id:string - String with the request id. Request Body: • session-check:string - Sessions to plot. • chart-check:string - Type of plot desired.	Yes
/compare/task_id1/task_id2	POST	Creates graphs of two requests.	• task_id1:string - String with the 1st request id; • task_id2:string - String with the 2nd request id. Request Body: • session-check0:string - 1st req sessions to plot; • session-check1:string - 2nd req sessions to plot; • chart-check:string - Type of plot desired.	Yes
/onlyuploadfiles	POST	Uploads user's files.	- Request Body: • files:object - JSON format containing files; • typeofmodel:string- Type of model uploaded.	Yes
/featureview/task_id	POST	Plots the features of a request.	• task_id:string - String with the request id. Request Body: • feature-check:string- Specifies the feature to plot.	Yes
/getmetadata	GET	Gets Manifest information.	-	Yes
/getfiles	GET	Gets all the user's files.	-	Yes
/getmodels	GET	Gets all the user's models.	- Request Body: • typeofmodel:string- Model category to list.	Yes
/getreqinfo/task_id	GET	Gets information about a request.	• task_id:string - String with the request id.	Yes
/deletefile/f_name/extension	POST	Deletes a file.	• fname:string - Name of the file; • extension:string - Extension of the file.	Yes
/deletemodel/path/f_name/extension	POST	Deletes a model.	• path:string - Describes the model category; • fname:string - Name of the file; • extension:string - Extension of the file.	Yes
/getstatus/t_id	GET	Gets status of a request.	• t_id:string - String with the request id.	Yes
/getsessions/t_id	GET	Gets a request's valid sessions.	• t_id:string - String with the request id.	Yes

Table 5.1: MARP's endpoints description and parameters.

Path	Method	Output
/login	POST	{"Login":String, "Token":String}
/register	POST	{"Register": String}
/requests	GET	{"Requests": Object}
/requests/req_desc	POST	{"Submit_Request": String}
/plotdata/task_id	POST	{"Participant_ID":String, "Metrics_Dict":Object, "Graphs_Imgs": Object}
/compare/task_id1/task_id2	POST	{"Part_ID_1": String, "Metrics_Dict1": Object, "Graphs_Imgs1": Object, "Part_ID_2": String, "Metrics_Dict2": Object, "Graphs_Imgs2": Object}
/onlyuploadfiles	POST	{"Upload_Files": String}
/featureview/task_id	POST	{"ParticipantID": String, "FeaturesImgs": Object}
/getmetadata	GET	{"Get_Metadata" : String}
/getfiles	GET	{"Files_List" Array}
/getmodels	GET	{"Models_List": Array}
/getreqinfo/task_id	GET	{"Task_ID": String, "Time_Created": Object, "Subject_Info": String, "Description": String,"Dataset_File": String, "Analysis": String, "Model": String}
/deletefile/f_name/extension	POST	{"DELETE_FILE": String}
/deletemodel/path/f_name/extension	POST	{"DELETE_FILE": String}
/getstatus/t_id	GET	{"GET_STATUS": String}
/getsessions/t_id	GET	{"VALID_SESSIONS": String}

Table 5.2: MARP's endpoints return objects.

On table 5.1 the parameters can be of two distinct "categories" that represent the way they are passed to the API. Path parameters are the ones that are passed within the path of the endpoint, before the query string (signalized with a "?"). Request bodies are closely similar to parameters but are not technically a parameter, despite at least, in this case, serving as one, these are passed to the API, within the request's body[40].

Table 5.2 output's are all in JSON format and it's elements can come in one of these data types:

- String - A simple alphanumeric sequence of letters and/or numbers;
- Object - JSON format data with respective content as key-value pairs;
- Array - List of elements that can be of type String or Integer.

5.4 Web Application

The web application provided by MARP to the user is a collection of HTML pages that allow communication between the user and the platform's processing level. The user has on this web client presented all of the options and functions that MARP can perform. Through these representations of actions, the user communicates with the back-end processing part of the platform on which, each request is processed, treated, and returned to the user in the form of understandable data so that they can analyze and perform which actions they intend.

The MARP web client was designed having in mind a simple and minimalist design without too many distractions, since MARP is to be seen as a tool to help it's users, so it's web application must be simple and easy to interpret for a fast and easy experience.

This layer of the whole system communicates with the back-end processing layer by making calls to the MARP's API over HTTP requests. After making a request, the API processes it and responds to it, this answer being received and treated by the web client. For example, the home page of the MARP system shows to the user all their requests made to the system. Every time this page is loaded, a request is sent to API for it to gather all the requests present on the database that are associated with this user. After querying the database, the

API responds to the web browser, with a JSON format message, containing the information corresponding to the request that has been made by this user. Then, the web client page, through JavaScript goes through this list and performs the actions so that the page can inform the user about their requests in the way that figure 5.7 shows. This type of interaction happens in almost every page. These HTTP requests are made by the web page using a JavaScript function called "fetch" that uses the web browser to make a XMLHttpRequest to the API or, in other cases, where the response may change, this meaning that different responses can be obtained of that interactions, (in the majority of the cases a positive response meaning that all worked as intended on the server side, and a negative one meaning the complete opposite) a JavaScript XMLHttpRequest object is instantiated directly and there are added event listeners to act upon the responses in order to make the system act in a certain way, according to the response obtained.

MY REQUESTS

	Description	Status	Type of Analysis	Time Created
☐	hgnnhgnhg	SUCCESS	All_Random_Forest	2021-07-19 20:49:00
☐	pickle_eda_nnng	FAILURE	All_Random_Forest	2021-07-19 20:47:45
☐	vhccc	SUCCESS	All_Neural_Networks	2021-07-19 20:43:58
☐	hnnh	FAILURE	Eda_Random_Forest	2021-07-19 20:22:46
☐	iiiiiiiiiii	SUCCESS	Eda_Random_Forest	2021-07-19 20:22:01
☐	fdsdfsdfjjh	SUCCESS	Ecg_Random_Forest	2021-07-19 20:20:31
☐	test csvs	SUCCESS	Ecg_Neural_Networks	2021-07-19 20:08:36
☐	test opensignals id30	SUCCESS	Ecg_Neural_Networks	2021-07-19 20:07:10
☐	pickle_eda_nn	SUCCESS	Eda_Neural_Networks	2021-07-19 15:13:07
☐	pickle_ecg	SUCCESS	Ecg_Neural_Networks	2021-07-19 15:10:53

« First | 1 | 2 | 3 | 4 | Last »

Figure 5.7: Table with a user's analysis requests and respective navigation bar.

An interesting solution made on the platform is the way that the requests are shown to the user, as it is presented on image 5.7. In the early stages of the development of the web application, this task was made by simply sending a request to the API, as explained above, asking for all the requests of the user. The API on it's part, after verifying the user's authenticity will answer with a list of requests. Upon receiving this response, these requests were then simply presented to the user in the form of a table, already with all the actions that were desired for the platform, but there was an issue. This approach was simple to implement and to understand but when the user had too many requests done, it was a very long list of requests, which led to interface issues. With this came the idea of paginating the requests, this was made still by requesting for the same information to the API but, before filling the table with this information, the list of requests is trimmed into groups of ten requests, making the list of requests become a list of lists of requests. With this it was also possible to deduct the number of pages there would be for that user. With all this, the number of buttons for navigating along the requests is created, with the existence of "Last" and "First" buttons that appear in situations that could be helpful to the user to jump right to the last or first pages with requests, respectively. With this new organization of requests also came the necessity of storing the ids of the requests that the user selects for it to be possible, for example, to select a request from page number one and then navigate to page number five, select another request and compare those two. This was a feature that did not took much time to implement but that introduced a very helpful and good feature into the system.

Another interesting feature of this web client is how it handles images, more specifically the images that represent the graphs with the plots required by the users. When a user requests the system to inspect the results of an analysis request, this comes in the form of a graph. The way these images are transited to the web application is made in a way where the image is not explicitly present on the message. To perform this action, the API commands the processing component to start a task and create the graph. The processing component executes this task, creates the graph, and saves it on the user's respective partition of the platform's file system. Then comes the part where the API needs to send this graph back to the web application. So, the system fetches the image, converts it into a byte array, and encodes this array in base64 format. This is then sent to the web application through a JSON object and the web platform upon receiving this data simply applies the inverse process and has the image with the graph so that it can be presented to the user. With this, it becomes possible to exchange images through a text-only JSON response.

5.5 Compatibility Related Adjustments

This platform was designed and developed using some material related to a work previously done in Universidade de Aveiro by Gisela Pinto et al. [11], where a study was conducted to deduct the biosignals that achieve better results in emotion identification processes. From this work emerged artificial intelligence models for this purpose which could be applied to this work. Having this in mind and with this work as an initial introduction to the theme, some adaptations needed to be made so that the platform could be able to implement the results of this previous work and the ideas projected for MARP. In addition to this, due to changes in crucial libraries, primarily for the biosignals analysis and process, it was necessary to make some changes as well.

The reuse of previous processing models required adaptation. These models were used to validate the platform's main objective that is to enable it's users to submit a dataset with biosignals data, a AI model, and analyze it to obtain results. The datasets from the same previous work, to train these models, were also made available for MARP. These datasets had the information of three sessions of features extracted for all the signals, this for all participants that made part of that study.

MARP was not thought to deal with such type of datasets, so there was necessary to make some adjustments in order to use this data that was available and ideal for the purpose of the platform. The Multimodal Analysis Reports Platform was thought to perform individual analysis. So, to be possible to use datasets containing one or more subjects' information on it, the ID parameter was introduced. This means that, when a user wants to perform an analysis, a numerical ID must be indicated on the request and also present with the same value on the dataset submitted (in the same form as described for CSV files on section 4.3), identifying the subject who's data is being analyzed. The way each file must represent it's data was already described in section 4.3. This was the solution found in order to have datasets with information of only one or many subjects at once and at the same time to make it possible for the platform to be capable of dealing with files submitted by users in many different formats and organizations.

Another constraint found is related to the models used to validate the system. These models only worked with datasets containing three sessions of data. This was not desirable by the platform, so to overcome this issue, it is possible to submit datasets with only one session of data and analyze it because, on the background processing, the platform creates the remaining sessions with null data that the system later recognizes as not being valid and does not show it to the user, making this session arrangement imperceptible in the eyes of the user.

The next major change that was necessary to be made was relative to the Neurokit library. Neurokit is a Python toolbox for statistics and neurophysiological Signal Processing (EEG, EDA, ECG, EMG...). This library was the one used in the previous work to make the feature extraction of the signals and then, this data was all wrapped in files that ultimately resulted in the datasets. The problem regarding this library appeared when it was decided to enable the platform with the possibility of performing analysis with prior feature extraction. This became a problem because the library used to generate these datasets has been replaced by it's successor, Neurokit2. This new version of the library generated data in a totally different way, so there was necessary to make substantial alterations to the data coming from these new methods of the updated version Neurokit, for the models to be able to understand it and work as intended.

All this could be done, nevertheless, another issue occurred with this library because the original Neurokit2 source code had a bug that incapacitated the system of working with some files originated by the Opensignals software. This is because the Neurokit2 functions were not prepared to deal with an EMG signal that had the offset on the last element of data, in the dataset, throwing an error to this case. So it was necessary to change the Neurokit2 source code and to use this changed version of the library instead of the original, for the platform to have all the functionalities desired.

Later, this issue was submitted to the Neurokit2 team, it was well received by them mentioning that it was something that their code was not prepared to deal with. Then this report served as an update commit and it is currently being handled by them. On the neurokit2 GitHub web page, under the following link, you can see the problem report and discussion of the same topic: `https://github.com/neuropsychology/NeuroKit/issues/524`

There is also the need to pay attention to the machine which is running MARP, since Tensorflow, another of the libraries used to analyze the biosignals, requires the system's CPU to be able to execute Advanced Vector Extensions (AX) instructions. If the system does not support this type of instructions the library will fail to launch and, by consequence, the whole MARP system will also not start.

5.6 Data Privacy and Security

MARP is a platform that deals with some possible sensitive data. This can range from users' passwords to the files they upload to the system as well as the results from their analysis. In order to enable MARP with security and data privacy, some policies were implemented in order to protect this sensitive data.

Data Privacy

The first and most logical data privacy measure is to hash the user password when this registers on the platform. After the user's register confirmation, in no circumstance, the platform holds the original form of the user's password. In exchange, the password provided on the register phase is hashed using the Secure Hash Algorithm 256-bit (SHA-256). Cryptographic hash algorithms produce irreversible and unique hashes, with this it is possible to store the hashed password on the platform's database and use it to compare with the hashed version of the password introduced by the alleged user on the login phase. If both hashes match, this means that the password is correct, otherwise, the login should fail.

The user public id, which will be mentioned below, is an identifier used internally on the platform to manage access to the user's files and information. Upon registration into MARP, the user's sensitive data is hashed, but it is necessary to have a way to quickly make an establishment between the user and the file repository for file storage and management, without explicitly exposing this data. This is where the user public id enters. The platform contains an internal file system where all user-related files are stored. The association between a user and the file storage partition that belongs to that same user is made using this public id. With this, each user has it's own partition of the file storage associated with their public id which can only be accessed and managed by making use of the session token. With this, there is a non-direct association of file system partitions with users, while also making it possible only for verified users to access their own partition.

Security Measures

One of the most important security measures taken was the generation of session tokens for logged users. These tokens consist of a digital signature assigned to each user when they login into MARP. This signature is obtained using Hash-based Message Authentication Code with SHA-256 (HS256) that will, using the SHA-256 algorithm, hash the user public id, that was explained before on this section, with the platform's secret key that consists of a large random key generated upon the system set up. This token is also provided with an expiration date, this is what enables these tokens of becoming session signatures, because after the time they were projected to cover expires, the token is no longer valid, so it is necessary to re-validate it, by executing the login again. This token is verified every time the user requests some action to the platform and the latter only executes it if this session token is valid, making sure that the person requiring the request is in fact the user that performed the login.

Upon a file upload to the system, the platform also makes a verification regarding the name of the file stored. The passed filename when submitted to upload is analyzed and it is returned a secure version of it. This filename can then safely be stored on the file system. The filename returned is an ASCII only string for maximum portability. This makes a verification stage on the upload so that no files with the same name are stored on the file system, which would lead to further platform problems. On Windows systems, this stage also makes sure that the file is not named after one of the special device files.

5.7 Final Platform

MARP is a system that allows user authentication and that stores each user's files and results. A system that, as intended, allows users to submit their biosignal dataset files as well as artificial intelligence models to run analysis to these datasets and produce perceptible and easy-to-read results to be ultimately presented to the user. It also allows for the users to submit these dataset files in different formats. It is a system capable of executing feature extraction for dataset files with data from biosignals like ECG, EMG, EMG-MF, EMG-Z and EDA. It allows the user not only to produce different types of graphs to analyze the results of one analysis as well as to compare two distinct analysis graphs. It lets the user manage what files and models they want to delete from it's database. It allows users to plot the data from each individual feature from the biosignals used to run the analysis, check every analysis information, etc.

The platform was projected to be a useful tool for researchers to use, in order to bring a major improvement to their work, saving them of not so popular work that could be costly in terms of time and sometimes difficulty of operation, as well as to make them focus on more important tasks and on the results.

This automatization of some tasks is the ideal scenario of future workflow for many areas, where the kind of tasks as the ones handled by MARP are delegated to computational systems, that treat them and inform the human of their availability in order for them to operate over this ready-to-use material. This can bring many advantages to several areas, mainly where some parts of the work can be costly to the workers, not necessarily because of it's complexity but simply because it is something that they are not well prepared to do. So, computational systems can help to reduce this barrier in order to improve productivity and the overall work quality.

As mentioned, the final result of MARP was a platform capable of performing all the actions that were thought to be useful in a platform like this one, namely the ones described in each of the personas and scenarios from section 3.1.1.

Chapter 6

Results and Validation

In an initial phase of the platform, it was necessary to perform validation work upon the artificial intelligence models provided to this work as available material. The number of models available was high and they were not all necessary, only the optimal models for emotion identification using biosignals, isolated or as a group of signals were necessary. So, it was necessary to make a filtering of these models as well as verify it's results and accuracy when used to classify the datasets of biosignals available. Later, with the goal of evaluating the platform developed on a "real scenario" there were projected some tests to apply to the system so that it could be measured how the platform was evolving as well as if it was living up to expectations to which it was projected. To do so, the system was presented to a group of people that fit in the description of potential future users (described in section 3.1.1) in order for them to perform usability tests and evaluate the platform as a potential product.

6.1 AI Models Validation

The artificial intelligence models used as material for this work had their origin in the work developed by Gisela Pinto [11]. From Gisela's work resulted in a large number of models with many applications. The validation work on this case was done by analyzing the objective for which each one of the available models was trained, to find the ones that would give better results when used in a platform like MARP. It was also pretended to find out, to validate, and use the models that would give the best results for this first "version" of the platform.

MARP was developed with the idea of working with these available models, despite the fact that the platform was projected in order to be possible to scale and use other models. Some adaptations to the structure of the platform were made in order to be possible to use these models. For example, these models were all used to evaluate three sessions of data each time they were used, so an arrangement was made in order to always provide the models with three sessions of data per input file, as it was previously explained and described in section 5.5. In the same section it is also explained the changes made in order to direct the system to make analysis on only one subject's data and not on all the subjects present on the dataset.

After these transformations, the testing of the models available for this project has begun. Since in this work, the analyses that were idealized for a user to make are individual and for a single user's biosignals, it is not logical to make an analysis with test and validation groups. Validation group datasets are the data that is used to train the model so that, later, this model could be applied to the test group that consists of data, present or not in the validation group, that the model will use to try to predict results. In this work since there is no training needed for the models, the validation group is empty and the test group consists of the data to analyze, uploaded by the user. This was something that the models provided were not tested to do. First, a group of models was made available to this project, and on the early testings, still with validation and test groups, they produced results similar to the ones reported on the work[27] they were from. But, when the changes regarding the necessity to make the system analyze only one participant's data with no information on the validation group, the results changed. It began to produce incorrect values, making that every analysis reported almost 80%-85% of the same emotion. This was caused by the usage of a group of models that were trained to give the best results for the specific properties of the study they were developed to: a large group of participants on the dataset, with a large validation group, and with a not so big (but always with more than 4 participant's data) test group.

After realizing this was the error condition, models from the same study but that obtained better results, being considered the optimal models in a general case, were made available. After some tests with the same dataset as the previous tests, it was possible to conclude that not only the error where one of the emotions was always the majority of the detected emotions was eliminated, but also that the results, although worse than the ones obtained when the conditions are similar to the ones described on Gisela Pinto's work [27], were correct in terms of the majority of emotion detected for each session. This was possible to verify due to the use of labels on the dataset's sessions, so it was possible to know which emotion was expected to be detected the most per session.

Another restriction found when testing the AI models was that those that were trained with the Random Forest algorithm could not be trained and then used by programs with different versions of a python library called "sklearn" which is a tool for predictive data (machine learning) because it caused exceptions when trying to load the data for the model to analyze.

All these models inputs are not the raw biosignals data but the data related to the features extracted from each one of these same signals. The features considered to be the ones that better describe an emotion for each signal were also the ones reported on Gisela Pinto's study et al. [11]. These features had defined names, and data organization structures that were defined by the python library that was used to extract them, in the case of the study mentioned, Neurokit. As it was explained in section 5.5 this library was substituted by it's successor Neurokit2 that introduced a major change in the way feature extraction functions return these same features. After realizing this, a work of adjustments in order to make a bridge between the organization of these features was needed, since the models available for this study were all trained using data with features extracted with the Neurokit version and would not recognize the Neurokit2 structure. In addition to this, some of the features that once came directly from the feature extraction function of Neurokit needed some additional conversions and processing when using the Neurokit2 version. This was a work that took some time and dedication since it was very specific, so it was necessary further investigation and study in order to be able to adapt the data from the current version of Neurokit2 correctly, for it to work with these models trained using, the now dead library, Neurokit.

6.2 Usability Tests

To evaluate the platform so that it could be improved and better analyzed, it was decided to perform some usability tests. These usability tests are a common and proved good way to evaluate a system's current state and to inform the developers on which points could be improved. These types of tests are made by asking a group of participants that are unfamiliar with the system, to use it, by performing some tasks proposed by the people leading the tests. After performing the usability tests each subject must report the experience so that the team can use this data to modify and improve the platform, if necessary, so it becomes more and more user-friendly and with this, improve the overall experience when using the system. On these types of tests, after the user finishes the tasks suggested, it is a common practice to ask some questions to these same users according to Nielsen's heuristics or System Usability Scale (SUS), about the experience they just have been through.

The main goal of any heuristic evaluation is to identify any usability problems associated with the User Interface (UI) design. Nielsen's heuristics also follow this principle. It specifically involves examiners evaluating the interface and judging it's compliance with recognized usability principles, the so-called heuristics. Rolf Molich and Jakob Nielsen proposed ten heuristics in order to be used as those "principles" to evaluate UI's quality, these heuristics can be analyzed et al. [41].

These heuristics allow developers to build and evaluate a system so that it better suits real users, making the system better and more coherent.

Another way to conduct post usability tests is with SUS questionnaires. It provides a "quick and dirty", reliable tool for measuring the usability of a platform. The System Usability Scale is a ten-item questionnaire with five response options ranging from "Strongly agree" to "Strongly disagree" for responders. It enables developers to test a wide range of products and services, such as hardware, software, mobile devices, websites, and applications. In this case, it was a website/application related situation. SUS has two primary characteristics: it is a simple scale to present to participants and it can be used with small sample sizes and yet produce accurate results. Despite it's obvious benefits, SUS is not diagnostic, which means that it's primary purpose is to classify the ease of use of the website/application, being evaluated rather than to identify specific flaws and propose solutions, at least not directly. When a SUS is used, participants are asked to score the following 10 items with one of five responses with the ranges mentioned above[42]:

1. I think that I would like to use this system frequently.

2. I found the system unnecessarily complex.

3. I thought the system was easy to use.

4. I think that I would need the support of a technical person to be able to use this system.

5. I found the various functions in this system were well integrated.

6. I thought there was too much inconsistency in this system.

7. I would imagine that most people would learn to use this system very quickly.

8. I found the system very cumbersome to use.

9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.

To interpret the results, of each questionnaire, for each of the odd numbered questions, it is required to subtract 1 from the score. For each of the even numbered questions, one must subtract their value from 5. After doing so, the SUS test provider must take these new values, add up the total score and then multiply this by 2.5. Though the scores are from 0 to 100, these are not percentages and should be considered only in terms of their percentile ranking. The average System Usability Scale score is 68, this means that if the system's score is under 68, then there are probably serious problems with the website/app usability which should be addressed as soon as possible. Scores of 80.3 or higher are said to be in the "very good" label, meaning that people love the system and would recommend it. A score of 68 or thereabouts means that the system is doing *OK* but could improve and at least, if the system scores 51 or under means that developers should fix the system fast [42].

6.2.1 Method and Results

The usability tests made for MARP were done taking into account that the users evaluating the system would be people that fit in the description of the primary persona, Carolina Pinto, that is described in section 3.1.1. Taking into account that Carolina was a university student doing her masters degree and that would be open to using computational mechanisms to help her with her study, there were contacted some biomedical engineering students from Universidade de Aveiro. It was possible to gather a group of four of these students, two men and two women, with ages ranging from twenty to twenty-three years old, to conduct the study.

The tests consisted of a group of tasks identified as being of most importance through the usage of the platform, to be done by the subjects. The files, as datasets, models and other files needed in order to interact with the platform were all prepared and given to the users with the respective previous description about each one: what each one contains, it's use, among other relevant facts. This was in order to make the participants comfortable with the files they were going to use. After the briefing and explanation of the platform's concept and vision, they were invited to explore the system as they wish and to perform some tasks that can be further analyzed in appendix B.

After performing those tasks, they were also presented with a questionnaire that was divided into two parts, the first one with questions created by the developer in order to register the subject's opinion about some aspects of the platform and to give them space to identify what they thought that was wrong or that could benefit of changes so that the developer could also get some real feedback from the tests and with this, having usability tests for validation but also with space for improvements, and a second part with the questions from the System Usability Scale questionnaires, so that, in the end, the system could be rated in order to understand the state of the platform at that point in time.

The results of the test sessions were very conclusive and with good overall results. Through the interaction with the users conducting the validation tests, the idea that each one gave at the end of their respective session was that the platform was a good system, perceptible, and most of all useful for someone with those characteristics. To top this positive feedback from the interaction with the users exploring the app, the answers to both parts of the questionnaire confirmed this good performance of the app. In the first part of the questionnaire each user

Figure 6.1: Biomedical engineering student performing usability tests.

	Part. nº 1	Part. nº 2	Part. nº 3	Part. nº 4
SUS Q nº 1	5	5	5	5
SUS Q nº 2	1	2	2	1
SUS Q nº 3	4	4	4	4
SUS Q nº 4	1	3	3	3
SUS Q nº5	5	4	4	5
SUS Q nº6	1	2	1	2
SUS Q nº7	5	5	5	4
SUS Q nº8	2	3	2	1
SUS Q nº9	5	4	4	4
SUS Q nº10	3	2	2	2
Total Score	90	75	80	82.5

Table 6.1: MARP's usability test SUS questionnaire results for each participant[1].

described the app as being useful to someone with those characteristics, they also stated that the platform would help a lot in the process of going over the results of an analysis. Most of the users also stated that the app had a good workflow. This part of the questionnaire also served as an ”inbox” for the users to state what they thought could be upgraded or changed, here some suggestions were made that are deeply analyzed in section 6.2.2. The second part of the questionnaire was, as stated before, the questions related to SUS. On this part of the questionnaire, the platform was also very well classified with a total average score of 81.9 points, meaning that it is a very good platform, on the overall. The results for each SUS question per user present on this testing session can be analyzed on table 6.1 where the score each participant of the questionnaire gave to each SUS question is represented, as well as the final total score for each participant, according to the formula explained above.

Adding these good results, the suggestions made by the participants on the study were very well pointed out and almost all converged to the same points, so, these flaws were identified and work to deal and fix these issues was planned and executed. This suggestions and respective changes are described on the following section.

[1]Score results calculated using the formula described on section 6.2

6.2.2 Improved Version After Usability Tests

The majority of the changes suggested by the users were interface-related.

For example, one of the suggestions was that after selecting the option to plot a feature or an analysis result, the page auto-scrolled to the graphs. This was something that was not implemented during the tests and that one of the users pointed out. Acknowledging that this is an interface feature that is helpful and that improves significantly the user experience since it was observed during the tests that some users seemed confused after giving the system the order to plot some information because since the page did not move, some thought that the process was not ready or that they did not click on the right option. With this change, this issue was eliminated because the auto-scroll once the user decides to plot jumps directly to the plots enhancing the user experience on these cases and eliminating any possible questions the user may have, like the ones mentioned.

Another issue that was pinpointed by the subjects performing the validation tests was that the platform they were using did not have a help page. A space where, in the case the user is struggling with something on the platform, they can go to in order to clear their doubts. This page was created and added to MARP with information vital for someone that wants to work with the platform. With this page, it is eliminated the feeling of lack of support from the platform. Through this page, MARP's users have a "safe space" where they can better understand how the system works and hopefully clear their doubts.

One more question raised upon the realization of the validation tests was that, to a small set of users, the page where the users can upload all types of files supported by the app, was not clear that those files could be independent. To them, the way the web page was organized seemed to induce some type of order upon the submitting process, passing the idea that not only in order to submit for example one dataset, the user had to also submit a model and a JSON manifest file, but also that on this step of the platform, the files would be tight together and associated, which is not the case. This was a simple upload page, that enables the user of uploading all the files they want before they make a request to the system so that when this occurs all these necessary files are already loaded on the platform and ready to be selected to use. Working on this, the "UPLOAD FILES" page was updated, increasing the distance between elements and changing the way the text was written and presented, in order to go in direction of the wanted idea for that page. Also after the click on the upload button, the page auto-scrolls to the progress bar that informs about the upload's percentage, so that it becomes automatically clear to the user that their upload has started and that they can check it's state.

6.3 First Prototype

MARP's visual interface, as mentioned before, was designed to be simple and direct to use. So, when the user executes the login into the system, they are presented with something similar to what is represented in figure 6.2. Here is the main page of the system, where the user may decide which action they intend to do.

A use case example that represents the flow across the platform that could be performed by the primary persona defined on 3.1 in order to exemplify how this type of user would interact with MARP is described on this section.

After entering her email and the password on the login page, Carolina, that is a user already familiarized with MARP, is presented to a web page depicted on image 6.2. Intending

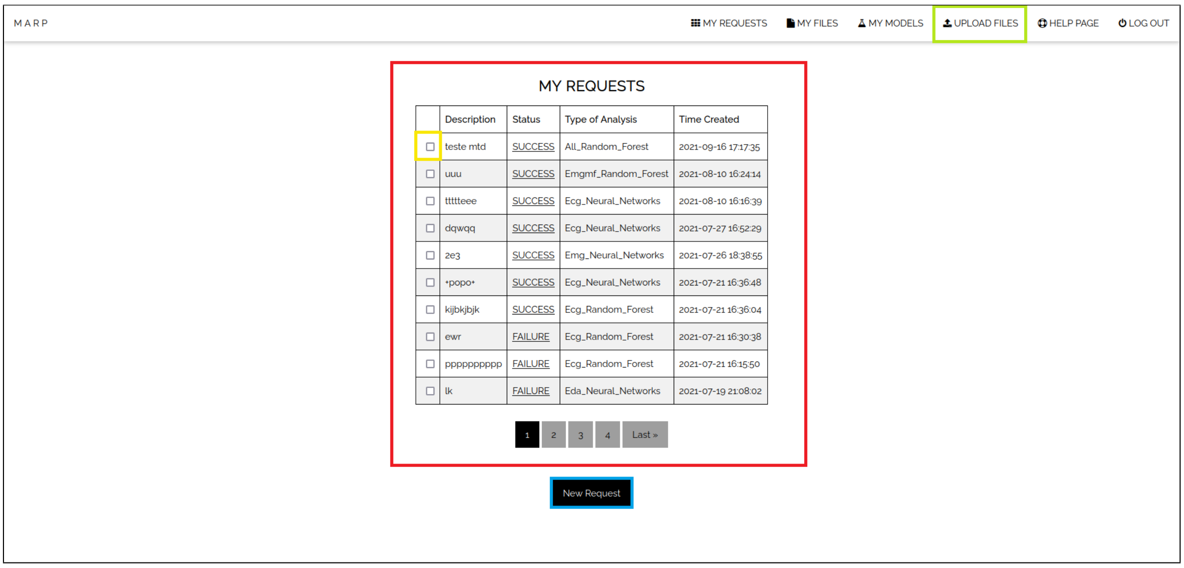

Figure 6.2: MARP's home page.

---

to perform a new request, she selects the option "UPLOAD FILES", highlighted with a green rectangle on that same image, and is presented to the upload page that is the one represented on image 6.3.

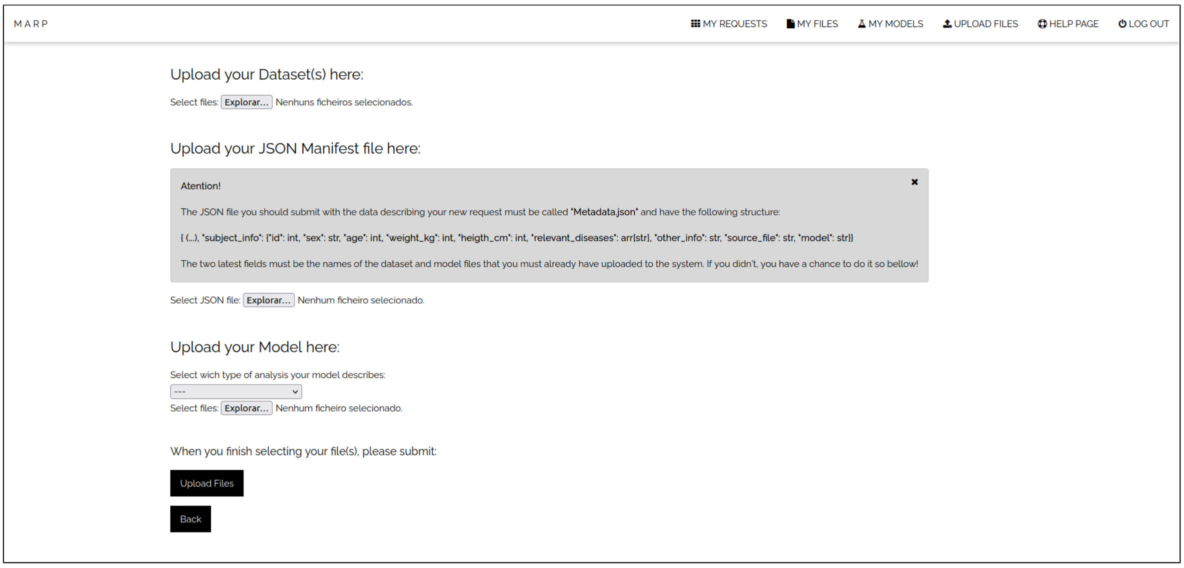

Figure 6.3: MARP's upload files page.

On this page, Carolina starts by uploading her dataset by selecting the "Explorar" button on the "Upload your Dataset(s) here:" section, the JSON manifest file she had already prepared, with the information about the subject whom the data on the dataset belongs to, and finally, uploading also a AI model by using the similar "Explorar" button on the "Upload your Model here:" section and selecting from the drop-down button the category which that model is related to (for example: "Random Forest (ECG)"). After this, she clicks the "Upload Files" button and is presented with a progress bar that represents the upload progress of her files. When the progress hits 100%, the system presents a message indicating that all files were submitted with success and Carolina is returned to the home page.

Again on the home page, now it is time for her to submit a new analysis request to the system, to do so, she clicks on the button highlighted with a blue rectangle on image 6.2 and is presented with a page that looks like the one represented on image 6.4 and 6.5.

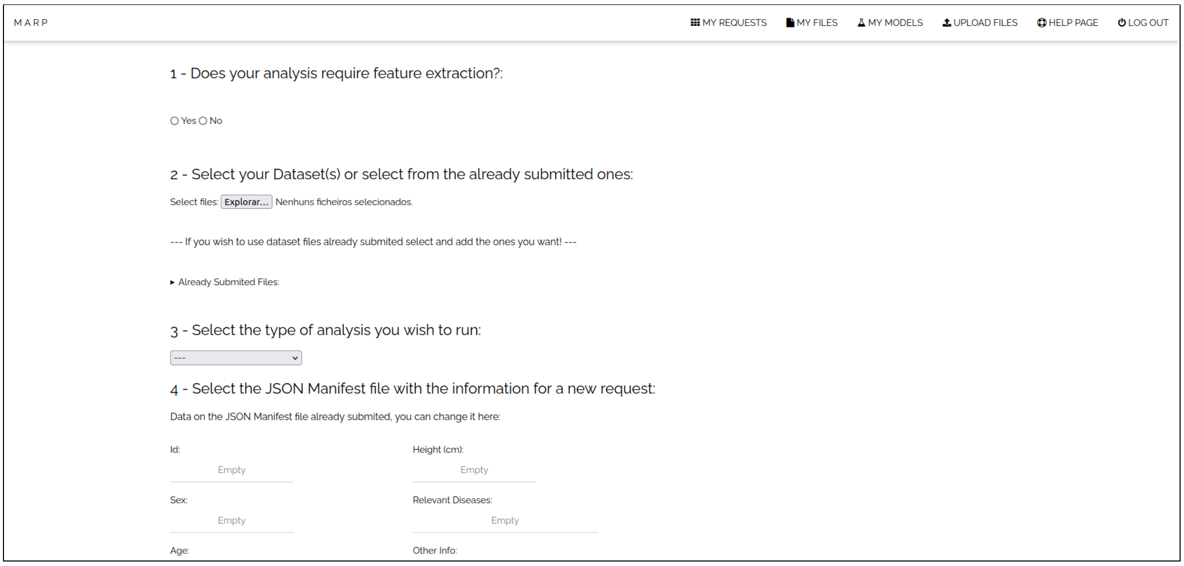

Figure 6.4: MARP's new request page part 1.

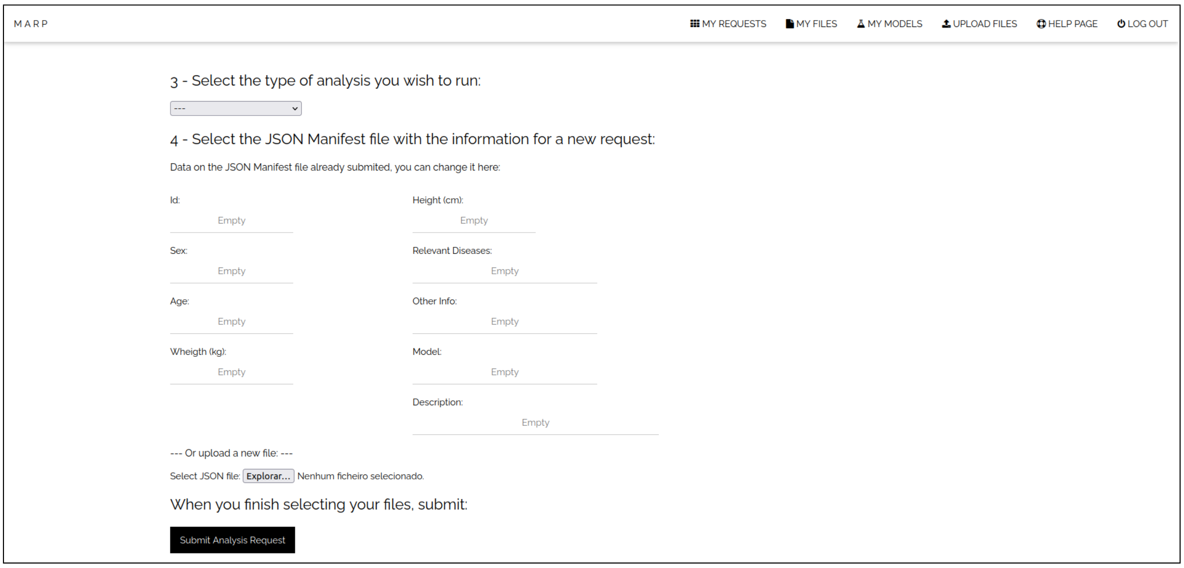

Figure 6.5: MARP's new request page part 2.

On this page, Carolina starts to set up her analysis' request. First, she indicates that the analysis will not require feature extraction since the dataset she uploaded has the data with the features already extracted. Next, she clicks the little arrow with the text "Already

Submitted Files" to open a drop-down list with the files she has in her MARP's repository, selecting the one she just submitted. Next, in step three she selects the type of analysis she pretends to make and which the model she submitted describes. On the next and final step of this process, since Carolina did not have a JSON manifest file describing the subject's information, the name of the model to use, or the description to give to this request, she fills up the empty spaces of the represented fields and hits the "Submit Analysis Request" button on the bottom of the page. After doing so she is, again, returned to the home page.

Now, on this page, Carolina can see her new request listed on the table, highlighted with a red square on figure 6.2. She quickly notices that the column status relative to her new request says her request is being processed. After a little Carolina checks again and sees the state changed to "SUCCESS" indicating that her analysis is done and ready to be visualized. With this, she selects the check-box highlighted with a yellow square on the same image and realizes that a new button with the description "Analyse Request" appeared above the already existing one. She clicks on this new button and is presented with a page like the one represented on image 6.6.

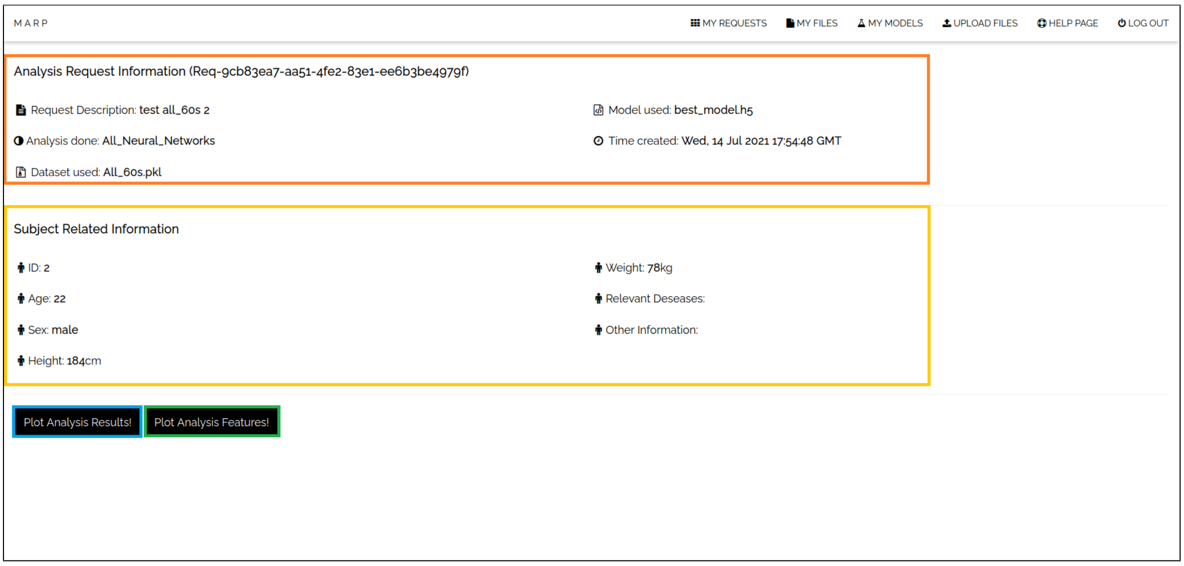

Figure 6.6: MARP's request's information page .

On this page, Carolina can check the information about the specifications of the request she selected and also about the subject that this request describes, highlighted respectively as orange and yellow squares on this same image. First, the user desires to check the analysis results, so, on this information page, she clicks the button highlighted with a blue rectangle. After doing so, a page like the one on figure 6.7 is shown in which Carolina can specify how she wants to view her results. After selecting the number of data sessions to visualize on the same graph and the type of graph desired, she click the "See graph!" button. Almost instantly, the page scrolls to a state like the one represented on image 6.8 where the graph as she described is presented, representing the data from the analysis made.

After taking her notes and conclusions, Carolina decides that she wants to inspect how the heart rate of the user changed through the experience. So, she goes back to the request's information page (image 6.6) and selects the button highlighted as green.

This act makes her be presented to a page that follows the organization represented on

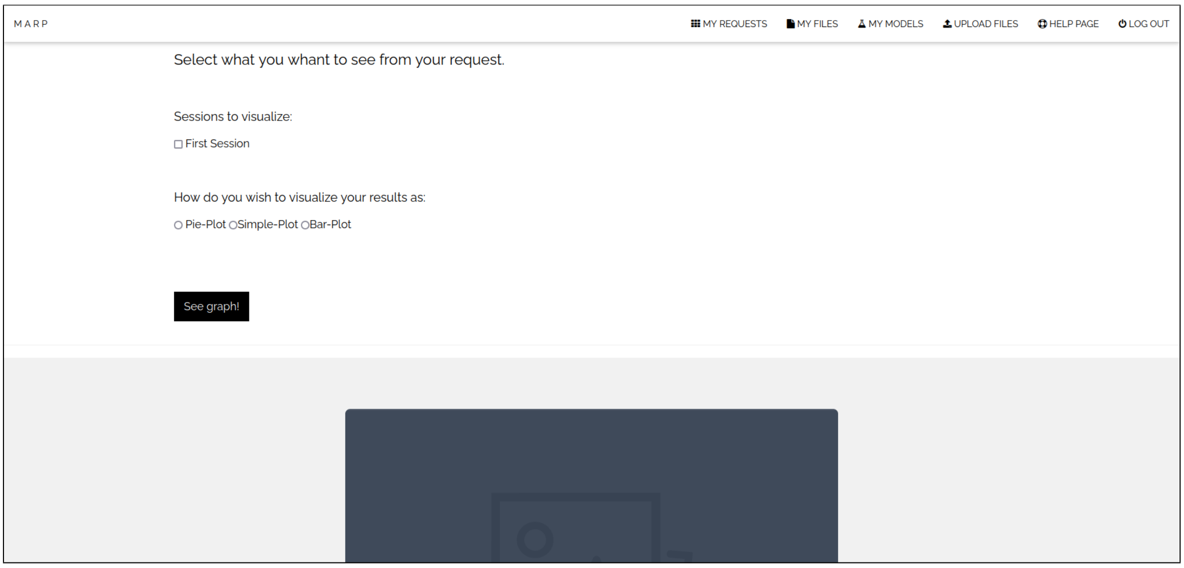

Figure 6.7: MARP's request results page before executing the request.

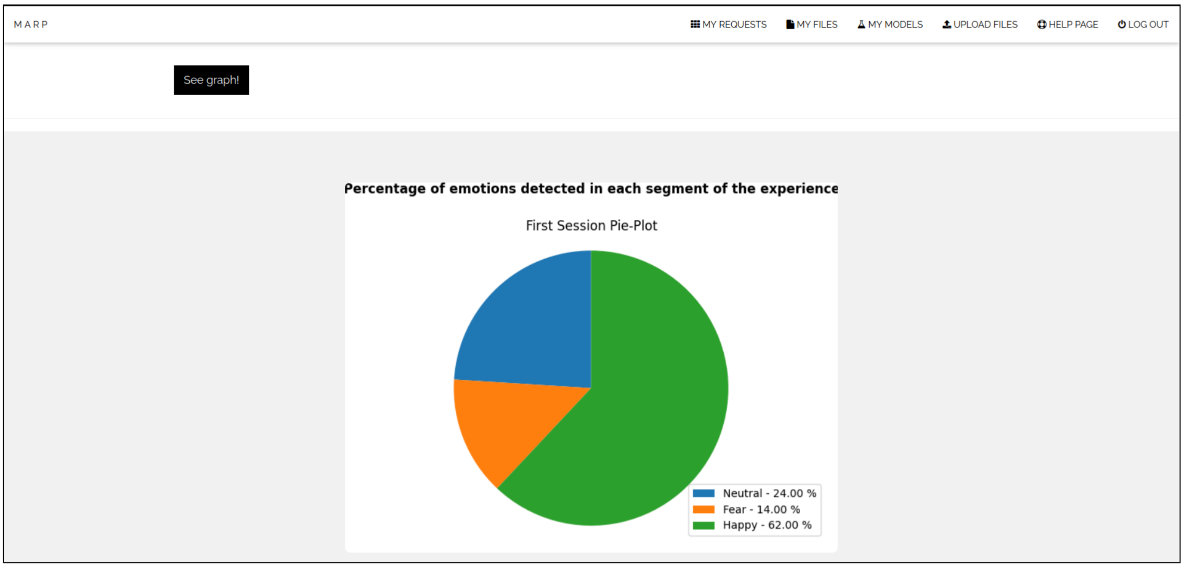

Figure 6.8: MARP's request results page part after executing the request.

image 6.9. Again, after describing how many sessions she wishes to visualize and the feature to plot, she hits the "See graph!" button, and the page auto-scrolls to a state similar to the one on image 6.10. In here she takes her notes and decides to go back to the main page by using the buttons on the top navigation bar.

On MARP's main menu, Carolina decides that it would be interesting to compare the results of this recent analysis with one she made a couple of weeks ago. She selects this recent request, and then uses the navigation buttons on the bottom of the table with the list of requests to scroll through her requests. After finding the second request, she notices that a button with the title "Compare Results" prompts on the bottom of the page, again above the "New Request" button. She proceeds to click on this button and is presented with a page

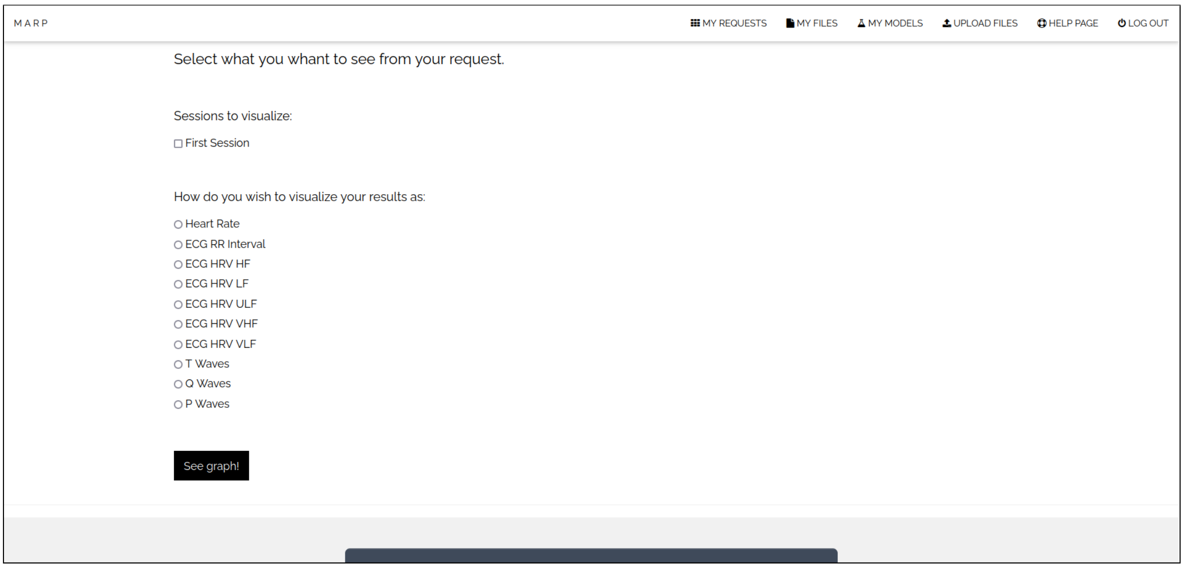

Figure 6.9: MARP's request features page before executing the request.

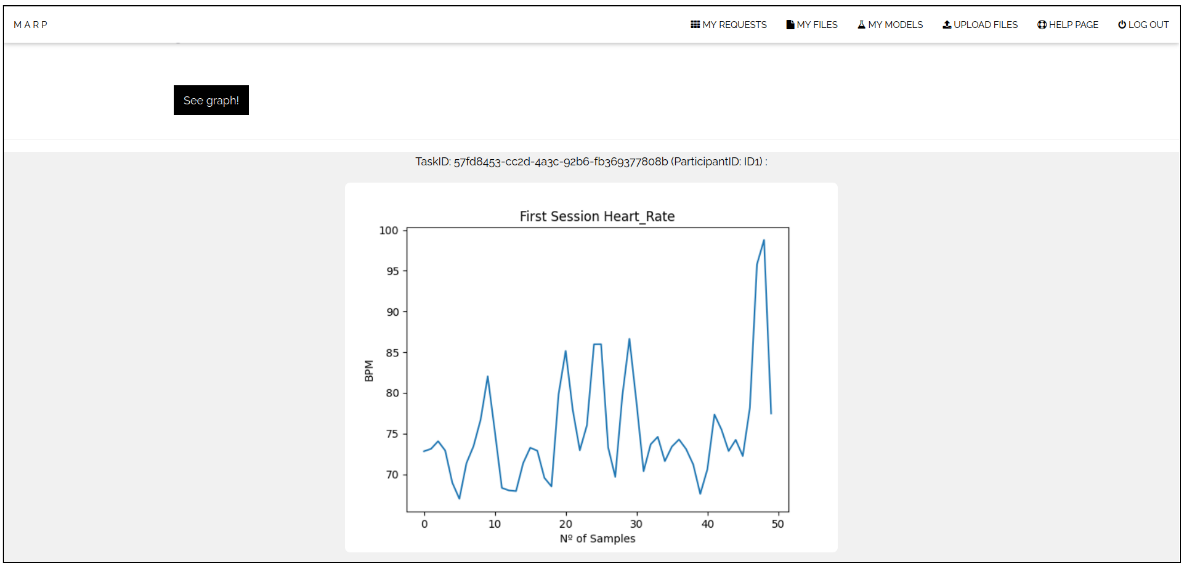

Figure 6.10: MARP's request features page after executing the request.

like the one shown on image 6.11.

On this page she can see the fields in which she can specify each request's graph. She selects the options that better suit the comparison between requests that she wishes to perform and hits the button on the bottom of the page. After this, the page jumps to the graphs location where the data is represented as she described, with two separate graphs, each one for one of the requests she selected on the main page (image 6.12).

When Carolina finishes taking the conclusions she intended, she clicks the "LOGOUT" button on the navigation bar, finishing her session on the platform.

All these interactions described above covers a use case that performs actions representing all the requirements defined at the beginning of this dissertation as being the goal for the

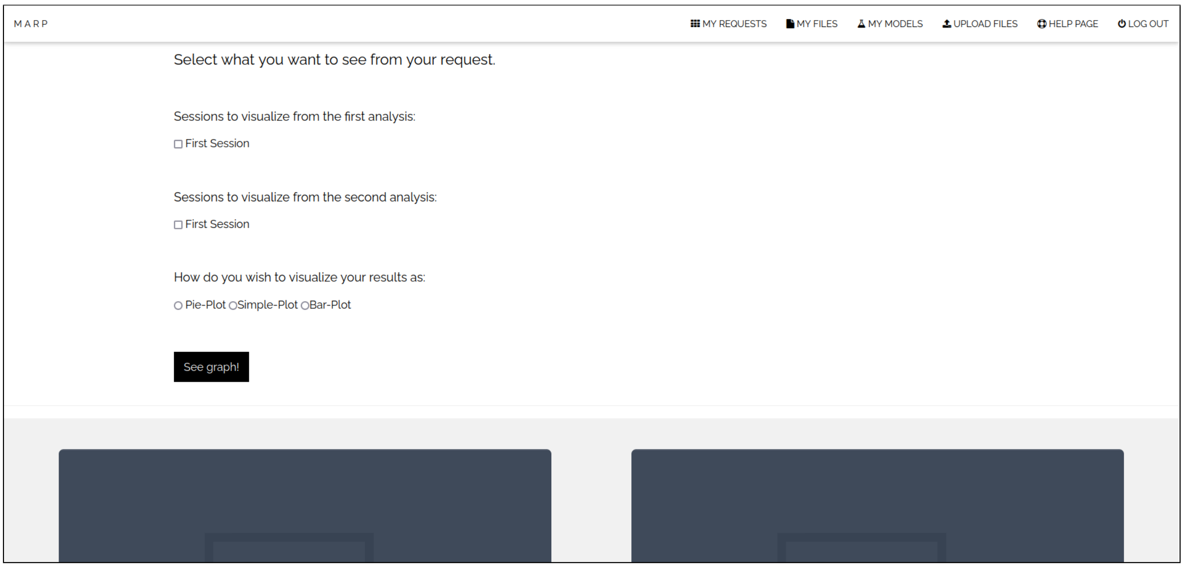

Figure 6.11: MARP's requests comparison page before executing the request.

Figure 6.12: MARP's requests comparison page after executing the request.

---

platform, but, there are some other interactions with the platform, that were not here represented, like listing and deleting the files or models uploaded and some pages like the help page that helps to users better understand the platform in case something isn't clear. Among some other actions possible to experience by exploring and using MARP.

Despite this, in this section, it is described most of the final result of MARP platform applied to the utilization of emotion identification models in order to explore it's usability and value. This final result was as stipulated on the initial idealization of a platform such as this one. In the end, all requirements were filled and implemented on the final result of the platform and since the usability tests performed also gave good indications about the system's usage it is safe to say that the development and implementation of the platform were successful.

6.4 Additional Usage

The Multimodal Analysis Reports Platform was also introduced as a mean of science propagation among young high school students that are considering their options before beginning their journey at the university. Universidade de Aveiro's Summer Academy is a program that allows students from the 10^{th}, 11^{th} and 12^{th} year of high school to experiment some programs that enables them to know a little more about the fields of study offered by the Universidade de Aveiro. A total group of nine high schoolers from the range of the grades mentioned before was divided into small groups through a time duration of three days, participated in the biomedical engineering program where MARP could be introduced to let these students understand how some signals are visualized as well as being introduced to the possibility of technology interacting with diverse fields of study.

To this program, MARP was prepared with a user account with a set of different analyses for the students to explore. The process of uploading files, and requesting analysis was explained to them as part of the platform but was not performed first because of the time it could consume on the process and secondly because it was not the objective of the session of experiments. The main objectives of this interaction with those students were that they could see the final result of a platform like MARP and that they could observe the graphs presented, the plots, analyze the signals as well as see how this can facilitate researchers work serving as a support platform.

The sessions with these students consisted of them gathering ECG, EMG and EDA signals on themselves by making use of the Opensignals software present on one of the computers in the room. While doing this they could experience and observe changes on the captured signals by performing some movements, as well as being informed about what each signal is focused on, how it captures the user's biosignals, and other questions they might have. After this, they would move to another part of the room where a computer with MARP was waiting for them to use the platform freely. They were educated about the objective of the platform, how it processes the signals they just observed on the previous part of their session, and then they were asked to explore the platform and the analysis request already present as they wished, under supervision in order to help them if needed and to answer some questions that could emerge during the session.

The sessions were very positive in terms of interest from the high schoolers and engagement when performing the tasks prepared for them. All mentioned that they could see the benefits of the alliance between technology and several areas, in this case, biomedical researching, and all thought of MARP as an interesting and helpful platform for researchers of this field of study.

All the sessions were made according to the COVID-19 prevention rules stipulated by the university's health department and by the recommendations of the Direção Geral de Saúde (DGS).

Figure 6.13: High school student exploring MARP.

Chapter 7

Conclusion

Biosignals can convert a living being's physiological changes into information that can be explored and further analyzed. The use of this type of information, as input, in order to perform specific tasks, has shown to be of much value in a vast range of different areas, from artistic, to medical, to technological applications.

The platform that was planned during the early stages of this work, was thought to be applied to these biosignals and at the same time improve some identified problems that related systems show. After all the study and development it was possible to produce a product oriented to the identification and report of human emotions.

During the development of the platform, it was possible to notice, that despite all the community hype around the area in which the platform was most focused on, that is affective computing, with several projects that provide support software for new technologies regarding the matter, those same technologies have some limitations. That can have many causes, but the one that appeared to be the most common was the lack of help they receive, mostly due to their limited user base. This lack of user base may be attributed to a variety of issues, ranging from the fact that the subject is still relatively new to the fact that it is not receiving the attention it deserves. Despite all this, almost all these technologies had very well developed documentation and were open to receiving community issues that would help them grow and overcome the inevitable issues that occur during the development. This work also served to improve one of the most essential Python toolkits for biosignal analysis with the execution of an extreme case that one of the datasets used on the testing, described. This served as a warning to the community that in a matter of days started working on the problem, ending with overcoming it, ultimately improving the toolkit.

Having into account the current state of the platform, it is possible to state that it fulfilled the requirements specified at the beginning of this work, as well as to claim that it became a platform directed to the type of user described on the personas, especially the one that represents a university student from the biomedical engineering course, which can be considered to be the user that would get the full experience and benefit from all the features and actions that the platform provides. This can be corroborated by the fact that the test group which conducted the validation tests on the platform, all fitted into the description given by the persona in discussion, and, as it could be verified by the results to those same tests, the platform had a positive impact when presented to this audience. From this testing, it was also possible to identify some aspects on which the platform could evolve and better itself. This feedback was taken into account and used to make some updates to the platform in order to fill in the gaps that were identified and with this improve even more the final result of MARP.

MARP provides a useful, easy-to-use, and easy-to-deploy platform that could be used by a variety of potential users who use biosignals as a resource in their activities. The platform would be especially useful for conducting studies regarding biosignals for recognition and identification of human emotions, but this can be expanded to other related areas, in the future.

This work also contributed to drawing attention to the importance of biosignals in a variety of fields. The research conducted during the development of this work also showed the potential of technology that explores and works with biosignals. Many areas were mentioned as potential benefactors of these types of technologies, and the idea that stays from this research is that these same sectors would be substantially revolutionized by such technologies. Primarily focusing on affective computing, this work also provided information that revealed that this is an area with many perspectives of growth and with ideas and projects that emerge as being of vital importance and interest for the future of technology and the ways we perceive it. As a result, an increase in tools and investment targeting this field have shown to be critical for developing new innovative technologies that will alter the current technological paradigm.

MARP could benefit from some future work aiming to improve the platform so that it could analyze more type of biosignals data (more than ECG, EDA and EMG) and where the current limitations, caused by the datasets and models restrictions could be surpassed. Potential vulnerabilities would be identified and documented as a result of these improvements, allowing for the development of a second version of the platform that is even more flexible to new models and biosignals data. Following these platform upgrades, MARP could be used in a research project, addressing affective computing, as a support tool for evaluating affective data. This study would provide a good real-world example of how the platform could be used to improve researchers' productivity and results.

Appendix A

Validation Tests - Questionnaire

On this appendix it is presented the questionnaire that the participants of the validation tests answered. This questionnaire was used in order to evaluate the system and to leave space for suggestions by the participants about potential flaws the system had. The questionnaire was made using the Google Forms tool. This document was presented to the users in Portuguese for a question of convenience.

Avaliação do sistema MARP

*Obrigatório

1. Nome: *

2. 1 - Sente que a plataforma é útil para a análise multimodal para identificação de emoções? *

   Marcar apenas uma oval.

   - Sim
   - Não

3. 2 - A plataforma constitui uma ferramenta que facilita o processo de análise de resultados associados ao processamento de sinais (neste caso, identificação de emoções)? *

4. 3 - Sente que o uso da plataforma é fluído e explicito para novos utilizadores? *

   Marcar apenas uma oval.

   - Sim
   - Não

5. 4 - Quais foram as suas maiores dificuldades, se encontradas, ao longo do teste da plataforma (em termos de interface, clareza de fluxo da plataforma, etc.)? *

6. 5 - Sentiu que os passos envolvidos no processo de submissão de um novo pedido de análise fazem sentido e são apresentados ao utilizador de forma clara? *

Marcar apenas uma oval.

- Sim
- Não

7. 6 - Sentiu que as páginas de detalhes da análise, plot de resultados, plot de features, comparação de resultados estavam bem organizadas e/ou a sua apresentação era amigável ao utilizador? *

8. 7 - Sentiu que faltava alguma funcionalidade útil ao utilizador, na plataforma? *

Marcar apenas uma oval.

- Sim
- Não

9. 8 - Tem alguma sugestão de mudança no que diz respeito à interface da plataforma de forma, a seu ver, a facilitar a sua utilização? *

10. Utilize este espaço para justificar as suas respostas às escolhas múltiplas anteriores, caso necessário.

System Usability Scale

11. 1. Eu acho que gostaria de usar esse sistema com frequência. *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo completamente						Concordo completamente

12. 2. Eu acho o sistema desnecessariamente complexo. *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo completamente	⬭	⬭	⬭	⬭	⬭	Concordo completamente

13. 3. Eu achei o sistema fácil de usar. *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo completamente	⬭	⬭	⬭	⬭	⬭	Concordo completamente

14. 4. Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema. *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo completamente	⬭	⬭	⬭	⬭	⬭	Concordo completamente

15. 5. Eu acho que as várias funções do sistema estão muito bem integradas. *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo completamente	⬭	⬭	⬭	⬭	⬭	Concordo completamente

16. 6. Eu acho que o sistema apresenta muita inconsistência. *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo completamente	◯	◯	◯	◯	◯	Concordo completamente

17. 7. Eu acho que as pessoas irão aprender rapidamente a utilizar este sistema. *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo completamente	◯	◯	◯	◯	◯	Concordo completamente

18. 8. Eu achei o sistema confuso de utilizar. *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo completamente	◯	◯	◯	◯	◯	Concordo completamente

19. 9. Senti-me confiante ao utilizar o sistema. *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo completamente	◯	◯	◯	◯	◯	Concordo completamente

20. 10. Foi-me necessário aprender várias coisas novas antes de conseguir utilizar o sistema. *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo completamente						Concordo completamente

21. Se tiver algo mais a comentar sobre o sistema, utilize o campo abaixo.

Appendix B

Validation tests - Tasks Script

This appendix has the list of all the tasks that the validation tests users were asked to perform during their testing sessions. This document was presented to the users in Portuguese for a question of convenience.

Tarefas:

1. Faça o registo na plataforma
2. Efetue o login na plataforma com as credenciais disponibilizadas no passo
3. Dê upload do modelo presente na pasta /Desktop/Signals-testes/Models/ECG/Neural_Networks, do ficheiro JSON presente na pasta /Desktop/Signals-testes/json e do dataset presente na pasta /Desktop/Signals-testes/txt;
4. Realize um novo request com feature extration, selecionando os ficheiros submetidos no passo anterior, indicando que pretende uma avaliação que utilize um modelo ECG Neural Networks, ajustando as informações do/a sujeito/a do/a qual o dataset é proveniente e atribuindo um título a este request;
5. Verificar etapa em que se encontra o request que acabou de efetuar
6. Fazer upload de um novo modelo, à sua escolha
7. Fazer um novo request, desta vez selecionando o dataset que pretende na própria página de 'New Request', estes ficheiros serão todos os que quiser, presentes dentro da pasta /Desktop/Signals-testes/csv (tenha em atenção que deve sempre selecionar no mínimo o ficheiro para o qual deu upload do modelo e deseja que seja utilizado na análise, ou seja, se deu upload a um modelo que utiliza o sinal de ECG, deve sempre submeter no mínimo o ficheiro com os dados deste sinal)
8. Fazer um novo request dando upload do ficheiro presente na pasta /Desktop/Signals-testes/pkl, selecionando o tipo de análise que considera todos os sinais e tendo em atenção o ID do/a participante
9. Verificar o porquê de o pedido anterior ter gerado um erro
10. Verificar quais dos seus pedidos já foram processados e se encontram prontos para serem analisados
11. Escolher um dos pedidos em que o estado seja "SUCCESS" e analisar as suas informações
12. Fazer plot de uma feature retirada dos sinais de um dos pedidos que efetuou
13. Fazer plot de um Pie Plot dos resultados da análise de um dos seus pedidos
14. Comparar plots de dois pedidos cujo estado seja "SUCCESS"
15. Eliminar um dos ficheiros submetidos anteriormente

Appendix C

File Structure - JSON Manifest

The structure of the JSON manifest file is described bellow. It is composed by two key-value pairs, with the values being JSON structures containing the data. The first key-value pair shown bellow has as it's key a String with the MAC of the device used to gather the data which the manifest is related to. The second key-value pair, must have the structure as the example shows, with the possibility to change de values of the JSON structure that is value of the "subject_info" key. This has the data that describes the subject's information, from whom the biosignals data is related to.


```
{
    "00:07:80:65:E0:24": {
        "position": 0,
        "device": "biosignalsplux",
        "device name": "00:07:80:65:E0:24",
        "device connection": "BTH00:07:80:65:E0:24",
        "sampling rate": 1000,
        "resolution": [16, 16, 16, 16],
        "firmware version": 775,
        "comments": "",
        "keywords": "",
        "mode": 2,
        "sync interval": 2,
        "date": "2020-10-16",
        "time": "16:14:8.23",
        "channels": [1, 2, 3, 4],
        "sensor": ["EMG", "EMG", "EDA", "ECG"],
        "label": ["CH1", "CH2", "CH3", "CH4"],
        "column": ["nSeq", "DI", "CH1", "CH2",
        "CH3", "CH4"],
        "digital IO": [0, 1],
        "sleeve color": ["dark_blue", "red", "red", "gray"],
        "convertedValues": 0},
    "subject_info": {
        "id": 28,
        "sex": "male",
```

```
        "age": 22,
        "weight_kg": 78,
        "height_cm": 184,
        "relevant_diseases": [],
        "other_info": "",
        "model": "best_model.h5",
        "description": "Example Description",
        "feature_extraction_option": "no"
    }
}
```

References

[1] Markets and Markets. ”Affective computing market by technology (touch-based and touchless), component (software (speech recognition and gesture recognition) and hardware (sensors, cameras, and storage devices and processors)), vertical, and region - global forecast to 2025.”. "https://www.marketsandmarkets.com/Market-Reports/affective-computing-market-130730395.html". (accessed: 30.08.2021).

[2] Joaquin A. Anguera, Faith M. Gunning, and Patricia A. Areán. Improving late life depression and cognitive control through the use of therapeutic video game technology: A proof-of-concept randomized trial. *Depression and Anxiety*, 34, 2017.

[3] Otsuka America Pharmaceutical Companies Alkermes and Takeda Lundbeck Alliance. The state of mental health in america. "https://mhanational.org/issues/state-mental-health-america". (accessed: 24.07.2021).

[4] Susana Brás, Sandra C. Soares, Ricardo Moreira, and José M. Fernandes. Bemonitored: Monitoring psychophysiology and behavior using android in phobias. *Behavior Research Methods*, 48, 2016.

[5] Madalina Sucala, Pim Cuijpers, Frederick Muench, Roxana Cardoș, Radu Soflau, Anca Dobrean, Patriciu Achimas-Cadariu, and Daniel David. Anxiety: There is an app for that. A systematic review of anxiety apps. *Depression and Anxiety*, 34, 2017.

[6] Michael Van Ameringen, Jasmine Turna, Zahra Khalesi, Katrina Pullia, and Beth Patterson. There is an app for that! The current state of mobile applications (apps) for dsm-5 obsessive-compulsive disorder, posttraumatic stress disorder, anxiety and mood disorders. *Depression and Anxiety*, 34, 2017.

[7] L. Ardissono, C. Gena, P. Torasso, F. Bellifemine, A. Chiarotto, A. Difino, and B. Negro. Personalized recommendation of tv programs. volume 2829, 2003.

[8] RL Mandryk. Evaluating affective computing environments using physiological measures. *Proceedings of CHI, Portland, OR*, 2005.

[9] Cenk Akbiyik. Can affective computing lead to more effective use of ict in education? *Revista de Educacion*, 352, 05 2010.

[10] Vasileios C Pezoulas, Themis P Exarchos, and Dimitrios I Fotiadis. *Chapter 2 - Types and sources of medical and other related data.* Academic Press, 2020.

[11] Gisela Antunes Pinto. A multimodal evaluation approach for cost-effective emotion classification. "http://hdl.handle.net/10773/29566", 2020.

[12] Miguel Ortiz. A brief history of biosignal-driven art from biofeedback to biophysical performance. *eContact! Biotechnological Performance Practice / Pratiques de performance biotechnologique*, 2012.

[13] University of Sussex Archive. ”New non-invasive sensor can detect brainwaves remotely.”. "https://archive.sussex.ac.uk/news/press-releases/media/media260.html". (accessed: 25.09.2021).

[14] Uniersity of Sussex Bulletin. ”Remote heartbeat monitor will outperform current technology.”. "https://web.archive.org/web/20181101162810/http://www.sussex.ac.uk/internal/bulletin/archive/08feb02/article7.shtml". (accessed: 25.09.2021).

[15] Andrew Brouse. ”Forty years of audio from the human EEG: A young person's guide to brainwave music.”. "https://econtact.ca/14_2/brouse_brainwavemusic.html". (accessed: 27.09.2021).

[16] Hagen Malberg and Werner Wolf. Biosignal processing: The pacemaker for innovations in biomedical engineering – state, developments, trends. 51:39–40, 2006.

[17] PhysioNet. ”Physionet: Brief introduction.”. "https://physionet.org/about/". (accessed: 23.09.2021).

[18] Antonio R Damasio. Emotion in the perspective of an integrated nervous system. Published on the World Wide Web on 27 January 1998.1. *Brain Research Reviews*, 26(2):83–86, 1998.

[19] Michel Cabanac. What is emotion? *Behavioural Processes*, 60(2):69–83, 2002.

[20] Jan L. Plass and Ulas Kaplan. Chapter 7 - Emotional design in digital media for learning. In Sharon Y. Tettegah and Martin Gartmeier, editors, *Emotions, Technology, Design, and Learning*, Emotions and Technology, pages 131–161. Academic Press, San Diego, 2016.

[21] Shaundra Daily, Melva James, David Cherry, John Porter III, Shelby Darnell, Joseph Isaac, and Tania Roy. *Affective Computing: Historical Foundations, Current Applications, and Future Trends*, pages 213–231. 12 2017.

[22] Jocelyn Scheirer, Raul Fernandez, Jonathan Klein, and Rosalind Picard. Frustrating the user on purpose: A step toward building an affective computer. *Interacting with Computers*, 14:93–118, 02 2002.

[23] Cem Dilmegani. 24 affective computing (emotion ai) applications / use cases. "https://research.aimultiple.com/affective-computing-applications/". (accessed: 20.07.2021).

[24] Liyanage C. De Silva, Tsutomu Miyasato, and Ryohei Nakatsu. Facial emotion recognition using multi-modal information. volume 1, 1997.

[25] Julia Diemer, Georg W Alpers, Henrik M Peperkorn, Youssef Shiban, and Andreas Mühlberger. The impact of perception and presence on emotional reactions: A review of research in virtual reality. *Frontiers in Psychology*, 6:26, 2015.

[26] Hamza Hamdi, Paul Richard, Aymeric Suteau, and Philippe Allain. Emotion assessment for affective computing based on physiological responses. *IEEE International Conference on Fuzzy Systems*, 2012.

[27] Gisela Pinto, João M. Carvalho, Filipa Barros, Sandra C. Soares, Armando J. Pinho, and Susana Brás. Multimodal emotion evaluation: A physiological model for cost-effective emotion classification. *Sensors (Switzerland)*, 20, 2020.

[28] Rosalind W. Picard. Affective computing: From laughter to ieee. *IEEE Transactions on Affective Computing*, 1, 2010.

[29] ReportLinker. ”Affective computing market - growth, trends, and forecasts (2019 - 2024).”. `"https://www.reportlinker.com/p05797261/Affective-Computing-Market-Growth-Trends-and-Forecasts.html?utm_source=PRN#backAction=1"`. (accessed: 30.08.2021).

[30] CISION PR NewsWire. ”Global affective computing market 2020-2025 - growing need for telehealth-related affective computing solutions.”. `"https://www.prnewswire.com/news-releases/global-affective-computing-market-2020-2025---growing-need-for-telehealth-related-affective-computing-solutions-301116277.html"`. (accessed: 30.08.2021).

[31] RabbitMq Support Team. Which protocols does rabbitmq support? `"https://www.rabbitmq.com/protocols.html"`. (accessed: 05.08.2021).

[32] IBM Cloud Education. ”IBM cloud learn hub - What are neural networks?”, 17 august 2020. `"https://www.ibm.com/cloud/learn/neural-networks#toc-what-are-n-2oQ5Vepe"`. (accessed: 06.08.2021).

[33] Leo Breiman. Random forests. *Machine Learning*, 45, 2001.

[34] Onesmus Mbaabu. ”Introduction to random forest in machine learning.”. `"https://www.section.io/engineering-education/introduction-to-random-forest-in-machine-learning/"`. (accessed: 06.08.2021).

[35] JSON team. ”Introducing JSON.”. `"https://www.json.org/json-en.html"`. (accessed: 17.08.2021).

[36] The Pallets team. ”Flask.”. `"https://palletsprojects.com/p/flask/"`. (accessed: 11.08.2021).

[37] Celery team. ”Introduction to celery.”. `"https://docs.celeryproject.org/en/stable/getting-started/introduction.html"`. (accessed: 11.08.2021).

[38] RabbitMQ team. ”Rabbitmq is the most widely deployed open source message broker.”. `"https://www.rabbitmq.com/"`. (accessed: 11.08.2021).

[39] The Red Hat team. ”What is a rest api?”, 8 may 2020. `"https://www.redhat.com/en/topics/api/what-is-a-rest-api"`. (accessed: 11.08.2021).

[40] Tom Johnson. ”Step 3: Parameters (api reference tutorial)”. `"https://idratherbewriting.com/learnapidoc/docapis_doc_parameters.html"`. (accessed: 23.10.2021).

[41] Jakob Nielsen. "10 usability heuristics for user interface design.", 24 april 1994; updated 15 november 2020. "https://www.nngroup.com/articles/ten-usability-heuristics/". (accessed: 09.08.2021).

[42] Usability.gov team. "System usability scale (SUS).". "https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html". (accessed: 09.08.2021).