INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO
MESTRADO EM ENGENHARIA INFORMÁTICA




Brain-Computer Interface and Silent Speech Recognition on Decentralized Messaging Applications

FÁBIO ANDRÉ VILAR LOURENÇO
Outubro de 2020

POLITÉCNICO
DO PORTO

Instituto Superior de Engenharia do Porto

Brain-Computer Interface and Silent Speech Recognition on Decentralized Messaging Applications

Fábio Lourenço

A dissertation submitted to obtain the degree of Master of Science in Engineering, Specialization Area of Graphics Systems and Multimedia

Supervisor: Dr. Paula Escudeiro
Co-Supervisor: Dr. Carlos Ferreira

Porto, October 15, 2020

Dedicatory

Aim for the stars. If you fail, you may reach the Moon.

The others will still be on the ground.

Abstract

Online communications have been increasingly gaining prevalence in people's daily lives, with its widespread adoption being catalyzed by technological advances, especially in instant messaging platforms. Although there have been strides for the inclusion of disabled individuals to ease communication between peers, people who suffer hand/arm impairments have little to no support in regular mainstream applications to efficiently communicate with other individuals. Moreover, a problem with the current solutions that fall back on speech-to-text techniques is the lack of privacy when the usage of these alternatives is conducted in public. Additionally, as centralized systems have come into scrutiny regarding privacy and security, the development of alternative decentralized solutions has increased by the use of blockchain technology and its variants.

Within the inclusivity paradigm, this project showcases an alternative on human-computer interaction with support for the aforementioned disabled people, through the use of a brain-computer interface allied to a silent speech recognition system, for application navigation and text input purposes, respectively. A brain-computer interface allows a user to interact with the platform just by though, while the silent speech recognition system enables the input of text by reading activity from articulatory muscles without the need of actually speaking audibly. Therefore, the combination of both techniques creates a full hands-free interaction with the platform, empowering hand/arm disabled users in daily life communications.

Furthermore, the users of the application will be inserted in a decentralized system that is designed for secure communication and exchange of data between peers, enforcing the privacy concern that is a cornerstone of the platform.

Keywords: brain-computer interface, silent speech recognition, peer-to-peer communication, hands-free interaction, privacy, decentralization

Resumo

Comunicações *online* têm cada vez mais ganhado prevalência na vida contemporânea de pessoas, tendo a sua adoção sido catalisada pelos avanços tecnológicos, especialmente em plataformas de mensagens instantâneas. Embora tenham havido desenvolvimentos relativamente à inclusão de indivíduos com deficiência para facilitar a comunicação entre pessoas, as que sofrem de incapacidades motoras nas mãos/braços têm um suporte escasso em aplicações convencionais para comunicar de forma eficiente com outros sujeitos. Além disso, um problema com as soluções atuais que recorrem a téqunicas de voz-para-texto é a falta de privacidade nas comunicações quando usadas em público. Adicionalmente, há medida que sistemas centralizados têm atraído ceticismo relativamente à privacidade e segurança, o desenvolvimento de soluções descentralizadas e alternativas têm aumentado pelo uso de tecnologias de *blockchain* e as suas variantes.

Dentro do paradigma de inclusão, este projeto demonstras uma alternativa na interação humano-computador com suporte para os indivíduos referidos anteriormente, através do uso de uma interface cérebro-computador aliada a um sistema de reconhecimento de fala silenciosa, para navegação na aplicação e introdução de texto, respetivamente. Uma interface cérebro-computador permite o utilizador interagir com a platforma apenas através do pensamento, enquanto que um sistema de reconhecimento de fala silenciosa possibilita a introdução de texto pela leitura da atividade dos músculos articulatórios, sem a necessidade de falar em voz alta. Assim, a combinação de ambas as técnicas cria uma interação totalmente de mãos-livres com a plataforma, melhorando as comunicações do dia-a-dia de pessoas com incapacidades nas mãos/braços.

Além disso, os utilizadores serão inseridos num sistema descentralizado, desenhado para comunicaçōes e trocas de dados seguras entre pares, reforçando, assim, a preocupação com a privacidade, que é um conceito base da plataforma.

Acknowledgement

Firstly, I want to express my acknowledges to my great colleague Luís Arteiro, who I had the pleasure to work with again for this project, as we have been working together for the past years. His perseverance and dedication have been essential and have kept me motivated.

Moreover, I would like to thank our supervisors, Dr. Paula Escudeiro and Dr. Carlos Ferreira, who did not limit us on this ambitious project and successfully help us in the pursuit of international academic recognition through participation in academic conferences.

I also want to acknowledge professors and colleagues with who I have worked during my journey at ISEP.

Lastly, I want to extend my appreciation to my family and friends that help me throughout these couple of years, supporting and motivating me.

Contents

List of Figures

List of Tables

List of Source Code

List of Acronyms

AHP	Analytic Hierarchy Process
API	application programming interface
ASR	Automatic Speech Recognition
BCI	Brain-Computer Interface
CER	Character Error Rate
CRISP-DM	CRoss Industry Standard Process for Data Mining
CTC	Connectionist Temporal Classification
DFT	Discrete Fourier Transform
DHT	Distributed Hash Table
ECoG	electrocorticography
EEG	electroencephalography
EMG	electromyography
EU	European Union
FE	Front End
FS	Forward Secrecy
GDPR	General Data Protection Regulation
GILT	Games Interaction and Learning Technologies Research Center
HCI	Human-Computer Interface
HCII	Human-Computer Interaction International Conference
HMI	Human-Machine Interaction
HMM	Hidden Markov Models
IHCI	Interfaces and Human-Computer Interaction
ISEP	Instituto Superior de Engenharia do Porto
KDF	Key Derivation Function
LSTM	Long Short Term Memory
NCD	New Concept Development
NPD	New Product Development

PD	Parkinson's Disease
RNN	Recurrent Neural Network
SAM	Served Available Market
SSI	Silent Speech Interface
SSR	Silent Speech Recognition
STFT	Short-time Fourier Transform
STT	speech-to-text
SUS	System Usability Scale
TAM	Total Addressable Market
TLS	Transport Layer Security
TMDEI	Tese/Dissertação/Estágio
UI	User Interface
VA	Value Analysis
VR	Virtual Reality
WER	Word Error Rate

Chapter 1

Introduction

The project that is inherent to this document was developed as part of the Tese/Dissertação/Estágio (TMDEI) course. This chapter's main purpose is to provide an overview of the project and what is to be expected throughout this document. This chapter will begin with a overview of the document structure, proceeding with a background context of this very own report, as well as the context and respective problem in which the application is currently being developed. Afterwards, the approach is briefly synthesised, alongside the inherent main goals driving the development of the project and its restrictions. The Introduction chapter ends with a description of the overall planning of the project.

1.1 Document Structure

This document is divided into seven chapters referring to the content of this dissertation, with extra chapters related to appendices and bibliographic references. The order in which these chapters are laid out is meant to be logical, chronological, and represent a sequence of thought.

The first chapter is meant to introduce the reader to the main problem being tackled and the project in which the dissertation focuses and ultimately aims to solve the problem. Furthermore, the project's goals are outlined, as well as providing a brief context in which the document was written. Additionally, the planning strategies and an overview of the chosen approach are described.

Secondly, the next chapter outlines the context of the in which the project related to this document inserts in and the adjacent theoretical concepts related to the development of the project. These will delve into some key concepts, such as hand/arm disabilities.

The third chapter will consist of an analysis of state-of-the-art, the current viable approaches to help improve disabled subjects' interaction with devices, and similar systems that have undertaken a similar route to this project.

In the fourth chapter, a value analysis regarding the application will be made, identifying the parties involved as well as crucial business processes inherent. Moreover, the criteria for the target-audience will also be explained.

The fifth chapter begins by outlining the functional and non-functional requirements of the system. Then, explains the chosen approach to solving the problem and the criteria that drove such a decision. Furthermore, within the fifth chapter, the diagrams and overall architecture of the application, including the conceptual diagram, will be demonstrated. The main goal of this section is to showcase the logical sequence of the application and outline the interaction between the components that make it up. Essentially, it will delve into concept designs of the platform and provide an overlook of the approaches implemented.

The sixth chapter focus on the implementation and development of the project components that this report focuses on. Moreover, this chapter showcases development decisions and further describes how different components were connected to build the platform.

The seventh chapter of this report will explain the assessment criteria for the evaluation part of the application and outline the evaluation methodology. This chapter further contains details about the experiences performed and a review of the results. It additionally reveals how the project was promoted, applying to several academic conferences, for scientific approval of this project.

Lastly, the last chapter sums up the overall project and contributions of this project, also discussing some alternative approaches and experiences that might be implemented in future work.

1.2 Document Background

This report constitutes the cornerstone of the TMDEI course, as part of the Master's in Informatics Engineering of Graphics Systems and Multimedia in Instituto Superior de Engenharia do Porto (ISEP). The report aims to describe the project that is being developed, through a complete analysis of the surrounding context, state-of-the-art, and to justify the approach that ultimately was chosen. To this, a value analysis is inherent which further gives more context to the problem at hand and better grasp the advantages of this solution. Additionally, each approach discussed will be compared with each other and a design for the implementation of the project will also be outlined. Finally, the development of the project with be described and explained, as well as the methodology used for tests and experiments. The team enrolled in the development of this project is composed by the author of this document, Fábio Lourenço, responsible for the hands-free interfaces that will be detailed further (which are the main focus of this report), and his colleague, Luís Arteiro, responsible for the decentralization counterpart of the system. The team is supervised by both Dr. Paula Escudeiro, as a supervisor, and Dr. Carlos Ferreira, as a co-supervisor. Games Interaction and Learning Technologies Research Center (GILT), which has partnered the project, also helped in the development.

This project can be summarized as a new approach to human-machine interaction solely by brain activity and silent speech, data which is extracted through an electroencephalography (EEG) headset and electromyography (EMG) electrodes, respectively. This interaction occurs in an unorthodox messaging/social application that is decentralized and distributed, attempting a new approach on privacy and communication between peers in a trustless context and with anonymity, security, and privacy as crucial cornerstones.

Even though this document refers to the messaging platform as a whole and to the decentralized backend counterpart in several moments, it is important to mention that this report focuses on the study and development of the Human-Machine Interaction (HMI) components of the project.

1.3 Context and Problem

The overall research project at hand ties directly with two state-of-the-art and rising concepts of different fields: hands-free and silent interaction with applications and development of scalable and decentralized applications with security, anonymity in mind, within a trustless, peer-to-peer communication ecosystem. In layman's terms, it aims for a private and hands-free platform where users can participate securely.

With the advent of messaging applications, there is an apparent lack of support for hand/arm impaired individuals to communicate on these platforms and use such applications to interact efficiently with other people. There has been significant progress addressing such issues, mainly through Brain-Computer Interface (BCI) technologies capable of recording brain activity that, powered by machine learning, can be translated into actions (Vanacker et al. 2007). This can prove to be helpful for people with specific motor disabilities to also make use of such applications. Additionally, creating a Silent Speech Recognition (SSR) system that can measure activity from muscles responsible for speech, also using machine learning techniques, could produce an output that can be used and applied to a myriad of fields, including the messaging and communication one.

In addition to this, security/privacy of data in messaging platforms is a recurrent topic of debate that has been catalyzed by the recent Cambridge Analytica scandal from Facebook (Hoyle, Das, Kapadia, A. Lee, et al. 2017). Current mainstream messaging applications neglect these aspects as these are usually centralized solutions with dedicated servers. Such applications are, therefore, inherently vulnerable to central points of failure and specific attack and data breaches, occurrences that have happened in the past (Cadwalladr and Graham-Harrison 2018). Contrarily, decentralized systems are usually associated with another heightened level of security and anonymity, which would improve the approaches to tackle these privacy problems.

All in all, the main problem this project refers to is the inclusivity of individuals with hand/arm disabilities and their possibility of communication privately.

1.4 Objectives

The main objective of this project is to develop a secure and private messaging application for communication between peers that both supports and is inclusive for people with hand/arm movement impairments. It also aims for these type of individuals to be able to efficiently communicate with others, regardless of disability status.

1.4.1 Overall objectives

Considering this main objective, some goals can be outlined that branch from the former and that directly affect the development of the project and how it intrinsically works and is designed.

- Promote a new approach on hands-free human-machine interaction and navigation through no movement whatsoever.
- Showcase a different proposal on the text input paradigm through silent means, using muscle activity measurement to perform SSR.

- Test a new approach to messaging peer-to-peer that is both resilient to attacks and maintains a high level of privacy whilst doing so, all within a democratic ecosystem without centralized power.

These objectives translate, generally, the components that constitute the overall project. That is, the communication between the user and the application through an BCI and Silent Speech Interface (SSI) approaches, and the decentralized secure backend, the two being considered the cornerstone of the application.

1.4.2 HMI objectives

After enumerating the overall objectives for the project, it is important to outline in more detail the goals for the counterpart related to this document. Therefore, being this report related to the HMI of the platform, the objectives focus on the two main components of this interaction (BCI and SSI):

- Showcase an approach to BCI in apps navigation, mitigating the need for any movements, while still maintaining a wide range of possibilities for interaction modes.
- Showcase a proposal on the use of a SSI for text input in message exchange, enabling this feature without typing or speaking audibly.
- Promote the use of these new interfaces to augment human interaction with computers for general-purpose HMI.
- Apply these approaches on a messaging platform inclusive for people with hand/arm impairments, by eliminating the need for movements or sounds.

These objectives resume in the mitigation of the difficulties of hand/arm impaired people in their online communications for providing a new perspective on HMI.

On the grand scale, the expected objective for the application is to function properly regardless of the subject that is utilizing a BCI with at least rank B in System Usability Scale (SUS) score (SUS scores and ranks are further explained in Section 7.3.1), and communicate efficiently with a decent level of accuracy, aiming for 85%, regarding SSI, making, therefore, the algorithm as generalized as possible.

1.5 Hypothesis

This project showcases the application of BCI and SSI technologies to messaging platforms allowing hands-free and private communications, aiming to a more comfortable and agile alternative for usage by, primarily, hand/arm disabled subjects. This project also proposes alternatives to different features improving usability by disabled users. The hypothesis aimed to prove is:

Hypothesis $\mathcal{H}$ - The use of both BCI and SSI approaches together, when applied to messaging platforms, is a reliable method to achieve hands-free and private communications for hand/arm disabled subjects.

1.6 Approach

As it was previously mentioned, this project deals with two rather distinctive fields that are: hands-free and silent interfaces, and a decentralized secure messaging system.

Regarding the former, which is the topic of focus of this document, the project uses BCI offering a medium in which a disabled user effectively interacts with the application. Recurring to an EEG headset and associated software, the platform will translate a user's brain activity into actions and will, consecutively, make it able for them to navigate through the application without any need for arm or hand movement. Additionally, considering the private communication objective of this project, a SSI is employed and results in text input without any movement or need for speaking aloud in public. Accurate SSR and translation into text are possible through machine learning algorithms, although the accuracy levels might dwindle from subject to subject and session to session.

On the other hand, the backend counterpart will engage in a decentralized approach in which the system is agent-centric, considering the great benefits and potential of blockchain-related technologies on applications and the heightened levels of security and privacy inherent to these. The main characteristic that makes this choice enticing is its decentralization that, from the get-go, mitigates any tampering attack and any possible central points of failure, as the data is spread through the network. This reliability on the state of the system is adequate for communication between users who are sure the information circulating within the system is reliable and checked throughout. The security of data is made through validation and cryptographic methods.

The outlined project can be thought not only as a different approach to the messaging and hands-free interfaces panorama but also as a new application of distributed ledger technology where privacy is at the forefront of the user's priority. Furthermore, it makes it so people with specific motor disabilities can interact with others, thus breaking down barriers of communication regardless of one's hand/arm disability status.

1.7 Brief Value Analysis

The value analysis of this project is more deeply explained in Chapter 4. The value proposition of the project relates directly with the support for disabled individuals for private communication, targeting specific disabilities whose support is extremely scarce, more specifically upper-body limb loss or impairment of conventional hand use. Additionally, the heightened level of personal data security is an added value to the system, as well as its wide range of applicability, not only for the messaging panorama, for example.

Regarding the product development paradigm, the Front End Innovation refers to the first stage of product development and, using the New Product Development mode, there are activities and processes within the project that correlates with these and are included in the development cycle.

The opportunity identification part of the project stemmed from the increasing scepticism regarding privacy in message exchange and support for impaired individuals to do so. To further outline the previous aspect, opportunity analysis was made on the market and technology trends. The idea generation and enrichment aspect of the project stemmed from the idea of empowering the user on a more democratic and safe ecosystem, which grew and culminated in the development of the idea of a decentralized system. The idea selection for the approach for the latter was chosen according to criteria based on performance and scalability. Finally, the concept definition winded up from an evolutionary and incremental idea selection processes, that passed through centralized, EEG-based communication, culminating in a blockchain variant-based messaging application with BCI and SSI.

The Canvas Business Model of this application deals primarily with key partners, those being GILT and hospitals for the assessment from subjects of the performance and accuracy of the application. Key resources include everything related with the SSI and BCI, more specifically EEG and EMG sensors, partaking as well within the cost structure of the project. The customer segments refer to the aforementioned messaging app users that want more privacy and comfort of usage, and hand/arm disabled people.

The value network of the application used to analyze the relationship with the project and respective entities comprise mainly of working with public institutes to gather subjects for evaluation and to obtain feedback for study data and improvement on the feature set of the application.

Finally, the Analytic Hierarchy Process (AHP) was applied to choose the best approach for the interfaces counterpart based on interaction speed, comfort, and privacy.

1.8 Planning

The project was organized in order to fulfill many needs, to be applicable to scientific conferences and to uphold the outlined objectives inherent to the application.

1.8.1 Tasks Planning

The planning for the project is showcased in the diagram in Figure 1.1, which was firstly created for the development of the project and to tend to the objectives that had be set.

The development cycle that refers to the design, implementation and testing of the application as a whole follows a Scrum methodology, with weekly *sprints* between the development members, with sporadic meetings with the supervisors to track the development of the application. The week objectives, tasks and priorities are discussed in these weekly meetings.

In addition to this, the group utilized a Scrum board to withhold the tasks and assign these accordingly, having a back log for future features. To aid this management, Trello was used to assign tasks to assignees and track the progress of each one, doing so on a weekly basis.

As it is possible seeing in the diagram, the project planning encompasses the proposal, submission and acceptance stage, the research, both technological and scientific regarding the technologies and approaches for the project and the conferences work-in-progress report's writing. The development of the application had an initial design phase for both counterparts, having the remaining months to dedicate exclusively for the development of the project, also including the evaluation and assessment phase with individuals from the target audience. Nonetheless, the report development is made throughout the year and tweaked in accordance to the changes in the project.

It is also outlined in Figure 1.1 that there was an experimental phase that would be dealt with target audience. Although initially planned, this was later not possible because of the current COVID-19 crisis, making it impossible to schedule meetings in hospitals. This information is further explained in Chapter 7.

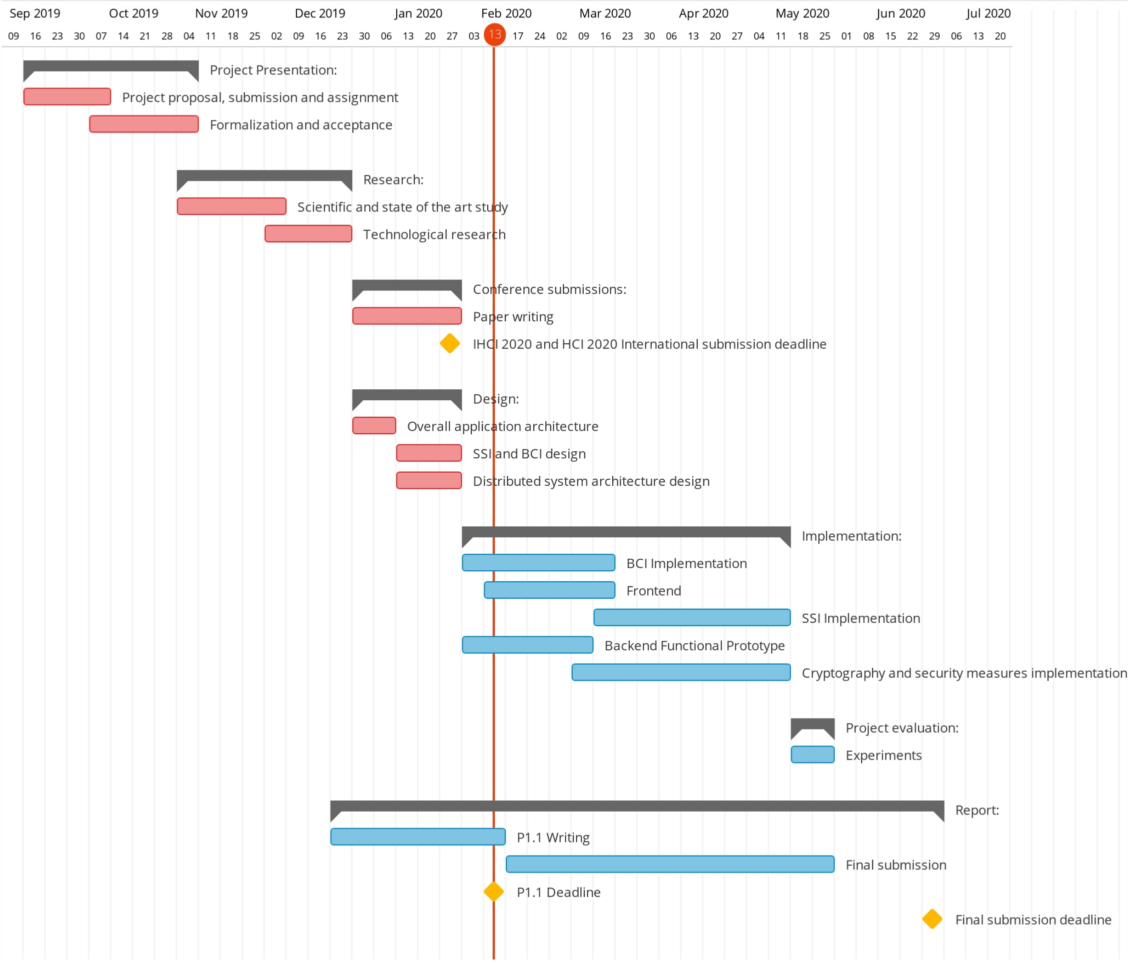


Figure 1.1: Project's Gantt diagram.

1.8.2 Meetings

As this project looks for non-conventional ways to interact with computers, using out-of-the-box approaches, the team felt a need to look for guidance in the different fields that include the main features of our project. Moreover, the team elected a supervisor, Dr. Paula Escudeiro, and a co-supervisor, Dr. Carlos Ferreira. Throughout the lifetime of this project, several meetings were arranged to discuss objectives and approaches with the supervisors, as well as with other experts in different fields that can guide ahead in some areas that our supervisors' mastery does not encompass. Appendix A displays a brief of each meeting, along with the participants and its date. Besides in-person meetings, there are also common contacts via email to receive feedback about the academic conferences submissions (detailed in Section 7.4), and informal and unplanned face-to-face consultations to answer simpler questions.

Chapter 2

Context

Throughout the development of the application there were a few concepts inherent to the problem that affected the development of the project. In fact, the application was tailored to some of these. These concepts were key to outline the most important features and, therefore, how to address the latter.

2.1 Domain Concepts

As it was mentioned previously, this chapter will introduce and briefly describe the important concepts that are implicit within the application being developed, in which its proper definition is crucial to better understand the problem and adapt to the audience to, at later stages, obtain a scientifically valid solution.

2.1.1 Hand/arm disabilities

As it was previously described, this project focus is on assisting subjects with upper-body limb disabilities in private online communications. Therefore, this concept must be clarified by enumerating some of the impairments of which we refer to. The next sections will explain and contextualize these.

Parkinson's disease

Parkinson's Disease (PD) affects 1% of the population over the age of 60, going up to 5% after the age of 85, being the second most neurodegenerative disease after Alzheimer's disease. It is caused by neuronal loss, mainly in the *substantia nigra* brain region, which is the main cause of the motor symptoms associated (Reeve, Simcox, and Turnbull 2014).

The main symptoms related to hand/arm impairments of PD are akenisa and bradykinesia, which translate into motor complications in terms of speed and amplitude, as well as difficulty in hand movements and grip (e.g. handwriting or holding a cup). These disabilities make the use of common solutions for online messaging challenging, as it becomes harder everyday for these people to handle their devices properly and accurately.

Amputees

Another condition that affects the use of devices for messaging (and not solely) is the amputation of an upper-body limb. People can, for many reasons, lose a hand or even an arm, which makes the use of these platforms unpractical. However, phone usage for these individuals is possible as mobile touch keyboards nowadays support and ease the handling of the keyboard with only one hand by providing native features prepared for this type of usage (Welch 2016). However, there are people that do not possess any arms/hands for whom it is impossible to interact physically with a device for messaging. Hence, there is a lack of solutions for online communications for these subjects, even more so on a secure environment.

Paralysis

Similarly, in terms of difficulty to the previously described condition, paralysis is an impairment caused by many sources (e.g. spinal cord injury, traumatic brain injury, stroke, complications from surgery, amyotrophic lateral sclerosis, multiple sclerosis) (Armour et al. 2016). Although some of these causes do not provoke permanent paralysis, as some are able to recover after rehab and therapy (Dobkin 2005), this disorder has a huge impact on the everyday life of these people while they do not totally recover the functionality of the paralyzed limbs. Therefore, like the condition described in the previous section, these subjects suffer from the same problem while trying to communicate online.

2.1.2 Synthetic telepathy

Synthetic telepathy can be described as a process of interacting with a computer through thought, without the usage of movements or speech. This is usually nominated as "brain-computer interaction" (Millán et al. 2010). In our approach, we go further on this description and also include the interaction through silent speech, as subvocal muscle activity is still detectable while the subject is thinking and reading. Thus, synthetic telepathy can be achieved by measuring both brain activity and speech-related muscle activity while a user is thinking, and translate this data to actions and inputs to a computer. Further in this document, the methods for this data gathering will be analyzed as well as different technologies used, regarding the invasive degree on the user body.

2.1.3 Privacy

The concept of privacy within the messaging panorama, although broad, can be narrowed down to the attempt of not leaking personal user information to third parties or to malicious entities and communication between peers without the chance of its data content being exposed. The security and exchange of data do not entirely boil down to this aspect, though. Many messaging applications claim to have implementations of secure protocols for message exchange, even though there are situations where that is not the case whatsoever (Hoyle, Das, Kapadia, A. J. Lee, et al. 2017).

Concerns for the privacy of data have also simultaneously been increasingly put into the spotlight due to recent events regarding social media, more recently with the Facebook and Cambridge Analytics data scandal, where personal data of millions of people's Facebook profiles were retrieved without consent and used for political campaigning purposes (Lapaire 2018).

Tighter regulation of protection of data has seen a major increase of attention also because of the implementation of the General Data Protection Regulation (GDPR), an European Union (EU) regulation law which aims to provide control to users over their personal data, granting these a more fine-tuned control on what applications/companies can host of their user's data.

These events affected the way applications are designed in accordance to comply with such regulations and to provide a bottom-line of privacy regarding user content. For messaging applications, several aspects concern the user's privacy of data. One of these is data collection for advertising or selling other companies data for other purposes. Retrieving data for more than is required for the functioning of any messaging app is a current concern that deals with security of data in not only messaging applications.

Decentralization is also a catalyst for the privacy of data. Regular mainstream applications are centralized and their user's data are stored in servers, which make up a unique target for malicious attacks and data breaches. Decentralization addresses such problem by spreading all data distributively, making it impossible for data to be tampered with or compromised.

Moreover, the presence or input of sensitive data (i.e. personal e-mail, full name, telephone number) in messaging applications is a subject of discussion for the privacy of data. The option to use the app anonymously matters because having to provide sensitive data to create a unique ID within the system can be used for malicious users to track information and specific users. This can be, however, remedied by hashing the data, which makes it unreadable to companies. This can be used to protect contact lists.

Additionally, when discussing the level of confidentiality of message exchange, the subject of cryptography is pertinent to be addressed. The spectrum in which messaging applications use end-to-end encryption in communication is wide and not always guaranteed, which is the case of Facebook's Messenger, for example (Jaeger 2014). Additionally, using dedicated frameworks to encrypt network traffic, such as Transport Layer Security (TLS) or noise, provide an important improvement of security on communication over the internet, since all of the data within is encrypted. Moreover, the existence of cryptographic primitives, such as specific key derivation [1], encryption or hashing algorithms, that are considered and peer-reviewed by cryptographers and having been target of research and attack resilience are crucial if an application is to boast about higher levels of privacy, especially when their cryptographic methods employ Forward Secrecy (FS) [2].

Therefore, the privacy, security and data integrity in peer-to-peer communication deals with cryptographic methods that are employed at surface-level and also at internet-protocol level, as is the case with TLS and with removing means to the user of supplying sensitive data that can eventually be exploited.

2.2 Stakeholders and Processes

The project inherent to this dissertation counted with the participation of stakeholders and actors who have contributed to the development of the final solution. In this chapter, both the participants of the project will be outlined, as well as describing the actors that make use of the application.

2.2.1 Project Stakeholders and Processes

The development team that is charged with creating, designing and implementing the solution, is made up of two developers, coupled with the main supervisor and co-supervisor. The

[1] A Key Derivation Function (KDF) derives secret keys from a confidential asset (e.g. password) and can be used to stretch or shorten keys to output in a desired format (Kaliski 2000).

[2] Also known as Perfect Forward Secrecy, FS is a feature in key agreement-based protocols that ensures session keys are not compromised even if the server's private key is compromised (Ristic 2013)

common goal spanning all of the aforementioned elements is to create the application that showcases an alternative solution on user-machine interaction and decentralized systems and make it functional for regular use. The supervisors also provide the necessary equipment for the proper usage and development of the project, as well as contributing by applying the dissertation to adequate scientific conferences, in order to attain further technical feedback and scientific validation.

Furthermore, individuals that suffer harm or hand impairments will be stakeholders of the application, namely to provide feedback and evaluation on its accuracy and performance. Although the institution or entity is not yet defined regarding this group of people, it is planned for a questionnaire session with these.

2.2.2 System Stakeholders and Processes

Being a messaging platform, the system's stakeholders are simply and solely the end user, that is, the one that is making use of the application. There is no other actor in the system other than them, as the platform is broad and does not have different roles embedded, therefore not having any extra features for other stakeholders. With that said, the user has access to all messaging features which are, indeed, simple, as the base of the platform consists only of message exchange. Figure 2.1 showcases the actor of the platform and its use cases.

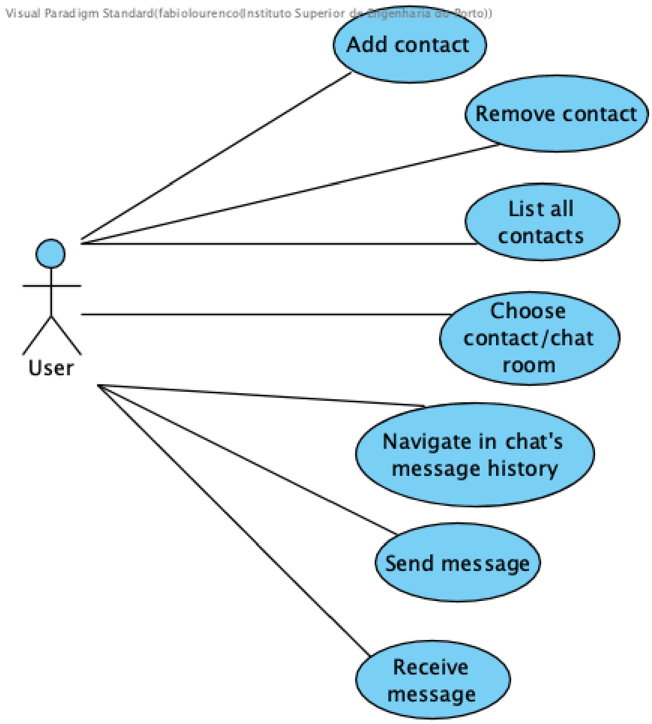


Figure 2.1: Use Cases diagram of the project

Thus, the definition of processes and stakeholders on a system perspective is not complex. Instead, the complexity of this project will focus on the innovation in both interaction interfaces and the technical aspect of privacy management, while maintaining the user experience and its actions and features as simple and with an accessible learning curve as possible.

2.2.3 Limitations

A limitation imposed in this project regards the SSI counterpart, where accuracy is not guaranteed to be with no errors and with complete accuracy. Being still an area that is still under research, utilizing machine learning methods and algorithms to obtain proper SSR results through EMG surface electrodes are usually more accurate when they are user and session dependent, not yet being applicable on a commercial scale that can be used by general audience and still produce good accuracy output. Nevertheless, this project is aiming towards a more session independent system, thus reaching more subjects. Additionally, a similar narrative can be referred to the BCI component. Although accuracy can be higher, it is never fully accurate across any subject that utilizes it and can fluctuate from person to person.

Chapter 3

State of the art

Since this document is focused on the HMI development for this project, in this chapter will be analyzed solutions related to this kind of interface and relevant technologies used to accomplish this fully hands-free and silent interface objective.

3.1 Platform visual navigation

Regarding the navigation in our platform through a visual interface, we envision an hands-free interface, allowing subjects with arm/hand impairments to interact with an application easily. There are a myriad of approaches possible to accomplish this goal, ranging from interfaces enabled using tongue and mouth, eye movement, head movement to voice controls, or even with brain activity, as will be further described.

3.1.1 Physical movement-based approaches

Presenting some of the examples of the above-mentioned interfaces, for mouth controlled there are devices such as TetraMouse, resembling a common mouse using two joysticks instead of buttons (*TetraMouse* 2018), QuadLife (*QuadLife* 2020), and Jouse (*Jouse* 2020) that are more similar to a conventional joystick. These devices have great versatility, and some, such as QuadStick, can be even used in gaming consoles (*QuadStick* 2020). A limitation is that these require a mounting arm, which can become unpractical to carry if the user wants to use its personal device elsewhere. Regarding head movement tracking, there are available options like SmartNav (NaturalPoint 2020) and TrackerPro 2 (AbleNet 2020), however, with these devices, the left and right clicks need an external or additional switch.

3.1.2 BCI-based approaches

Brain activity-based approaches were alternatives that seemed more appealing from a research standpoint and will be the focus on the next sections. This method is based on the recording of brain activity that is later translated into actions on a computer. BCI researches have significant progress addressing issues related to the communication of people with hand/arm impairments and paralysis (Niels Birbaumer 2006), which is the main focus of this project. In this section, the two types of BCI (invasive and non-invasive) will be presented and compared.

The main characteristic that differentiates these two techniques is that the invasive one required surgery on the user and the non-invasive does not. The non-invasive one is based on EEG, that is, the recording of the brain's electrical activity by electrodes placed on the subject scalp. This method is simpler and cheaper, although having some drawbacks, as the reduced quality of the recordings since the electrodes are far from the source, and interference in the signal caused by blinks and eye movement (Ball et al. 2009). On the other hand, the invasive method uses electrocorticography (ECoG), which is similar to EEG, except it records the brain activity using cortically implanted microelectrode arrays, being less vulnerable to signal noise in the recording, but it is, without a doubt, far more difficult to implement, as it requires a surgery and skull opening (del R. Milan and Carmena 2010).

Comparing with both techniques, ECoG reveals to be more promising as electrodes are closer to the signal source and it is not limited to a bandwidth of about 50Hz as EEG does, which is better for bands correlated to movement direction where is crucial information is in 80Hz or more. Furthermore, ECoG as a more spatial resolution as it has a millimeter range, allowing a more accurate signal source localization (Wilson et al. 2006).

Further, we will analyze approaches using both invasive and non-invasive techniques to BCI.

3.1.3 Invasive BCI examples

One well-known approach to this technique is the one developed by Neuralink, having the most advanced electrode threads and even have built a neurosurgical robot to insert these threads into the subject's brain with high precision, targeting specific brain regions (Musk and Neuralink 2019).

Moreover, a study on BCI using ECoG experiments its application for a one-dimensional cursor movement comparing this approach with a non-invasive one, obtaining an accuracy of 75% (Wilson et al. 2006).

Concerning this invasive approach, although showing better results and accuracy in the recorded brain activity, as it requires surgery and is considerably harder to implement, there is not a wide range of projects related to app navigation or command classification already published.

3.1.4 Non-invasive BCI examples

Concerning non-invasive brain-computer interfaces, the most commonly used method for this purpose is electroencephalography. A myriad of works regarding BCI have emerged, allowing users to control virtual keyboards, which will be further analyzed regarding text input features), but also mouses and even moving wheelchair just by thought (Vanacker et al. 2007), empowering individuals with severe motor disabilities to have an improvement in their mobility and interaction with computers.

Regarding apps navigation using this approach, there are experiments on cursor movement. Fabiani et al. 2004 have created a BCI system that allows a user to move a cursor, testings different approach within movements for one or two dimensions. The tests on this project showed an accuracy of around 70% in different models tried (Fabiani et al. 2004). A similar project uses a hybrid near-infrared spectroscopy-electroencephalography technique trying to extract four different types of brain signals, decoding them to four direction symbols, namely, “forward,” “backward,” “left,” and “right". The classification accuracy for "left" and "right" commands was 94.7%, 80.2% for "forward", and 83.6% for "backward" (Khan, M. J. Hong, and K.-S. Hong 2014). Later in this document, it will be shown that our approach uses a similar approach for the set of commands used for navigation in the platform.

3.1.5 Relevant BCI technologies

Most EEG equipment is not quite user-friendly as a majority of these products are still in laboratory experimentation that has never required much thought on portability, comfort, and aesthetics. As most of the uses of this technology are for inclusion and overcoming difficulties due to motor disabilities, users do not want to look odd and want to be socially accepted. Devices developed by Emotiv, such as displayed in figure 3.1, are prepared for EEG interfaces out-of-the-box with tools for mental command training and allow the use of these technologies in new products and also help to research possible applications for it. These products enable the creation of user-independent applications with EEG-based interfaces, enabling more people with particular constraints to interact with it. These devices also have special attention to the concerns related to aesthetics and portability beforementioned, as some are already based on wireless helmets and dry electrodes.

Figure 3.1: Emotiv EPOC 14-Channel Wireless EEG Headset (*EMOTIV EPOC 14-Channel Wireless EEG Headset* 2020)

3.2 Text Input and Speech Interfaces

Begin this project, at its core, a messaging platform, the text input is a key feature and it should likewise be inclusive for people with disabilities. Therefore, the approach for this feature must be accomplished without using movements. Likewise, since one of the key values of this project is user privacy, this feature should not require publicly audible inputs, as it would compromise our concept.

3.2.1 BCI for text input

A method to input text could be using the described above technologies of EEG. Several experiments have been made trying to use BCI to input text using virtual keyboards, however, none are practical or fast enough for daily usage as it can be exhaustive and frustrating.Using this technique, it would allow a user to input text using solely its thoughts.

In 1999, a group created a Thought-Translation-Device which split a virtual keyboard with the alphabet in half and allowed the user to make binary decisions, selecting one of the sides using a BCI, iteratively splitting each part in half until a letter was selected. This is a tedious task and the spelling rate was about 0.5 char/min (Birbaumer et al. 1999). Another interesting approach is an application called Hex-O-Spell, which instead of binary choices, the user can select one amongst six hexagons, each containing a set of letters (shown in figure 3.2), enabling the input of about 7 chars/min (Williamson et al. 2009).

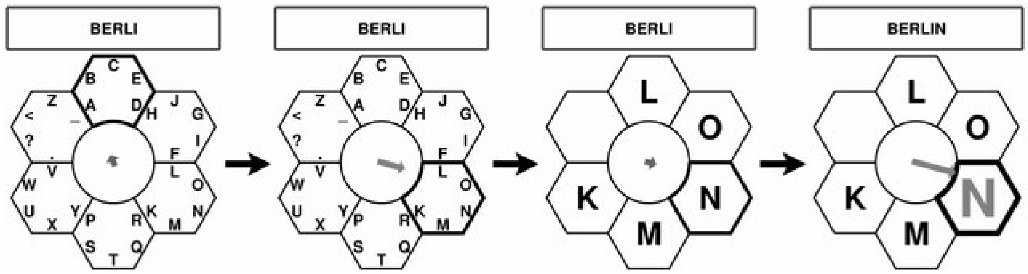


Figure 3.2: Showcase of a sequence of keyboards showed to the user typing the letter 'N' using the Hex-O-Spell (Blankertz et al. 2006)

Yet another approach consists of displaying a matrix of 6 by 6 independent cells, each one containing a letter of the alphabet, and the user focus on the cell that it wants to select, reaching 7.8 chars/min with 80% accuracy (Donchin, Spencer, and Wijesinghe 2000). Although the input speed approach is not the primary focus of these research projects, since the goal for our solution is to integrate with a full-fledged messaging app, it must be somewhat fast and user-friendly for instant message exchange to be conceivable.

3.2.2 Silent Speech Recognition

Another studied approach to allow text input was through a SSI. This interaction is accomplished by using a device that can record the activity of muscles used in speech, using EMG, and these signals can be translated into text. This method also allows people with speech impairments (e.g. laryngectomy) but maintains the levels of privacy expected for this project.

An interesting project, for which it was created a large dataset called *EMG-UKA* corpus, tries to reach the same objectives as our project, using the EMG data of articulatory muscle activity used in speech. This dataset includes 63 recorded sessions from 8 speakers, with a total of 7:32 hours of data. However, the free trial version only has 1:52 hours, still being a great starting point for testing new classification methods. This is the largest dataset found with EMG recordings for audible, whispered and silent speeches, with different subjects and sessions, providing a huge variety of data, enough for trying different approaches for SSR. Using this approach it would be faster to input text, but recent researches on this field still have a high Word Error Rate (WER) when the set of words is large, as Wand, Janke, and Schultz (2014) achieved around 48% WER for their silent speech mode for a vocabulary of only 108 words, using a EMG-based speech recognizer based on tristate Hidden Markov Models (HMM) (Wand, Janke, and Schultz 2014). Rosello, Toman, and Agarwala (2017) proposed a different solution, using a three-layer Recurrent Neural Network (RNN) with Long Short Term Memory (LSTM) cells as their baseline architecture, also using the *EMG-UKA* corpus. However, this new approach only achieved around 30% char accuracy for silent speech data (Rosello, Toman, and Agarwala 2017).

A study on SSR performed on subjects with laryngectomy recorded EMG activity with 8 sensors placed on speech responsible muscles, on 8 individuals at least 6 months after total laryngectomy. The recorded activity was conducted while the subjects read phrases from a 2,500 English words vocabulary. The model was created for word recognition based on phoneme identification for the 39 commonly used phonemes in English. This model obtained 10.3% WER in tests with all of the 8 sensors, while it went up only to 13.6% when half of the sensors were removed, using, as seen in other studies, a HMM-based system (Meltzner et al. 2017).

On the other hand, some approaches use optical images and ultrasounds to perform SSR, instead of using EMG recordings. While also using HMM-based systems, this approach combines ultrasound images of the tongue and video images from the lips to associate those to perform phonetic recognition (Hueber et al. 2010). This approach is less portable and mobile than the EMG-based ones since it requires a camera always facing the subject for lip movements recording.

A relevant product developed in this field is AlterEgo, a wearable device that accomplished 92% accuracy uses the EMG approach, although these results were for a small vocabulary of a set of commands. This creates a silent speech interface to communicate with devices and AI assistants, allowing the user to receive aural output from them with its bone conduction headphones embedded in the device. This project also preserves our core values of privacy, as is allows people to communicate privately and seamlessly with other devices. Furthermore, although it focuses on silent interaction with other devices (e.g IoT devices) inputting predefined commands, an approach to the messaging paradigm can also be applied. For now, this product only allows an elected set of messages to reply, being these "hello how are you", "call you later", "what's up", "yes", and "no", though this vocabulary can certainly be expanded (A. Kapur, S. Kapur, and Maes 2018).

Studies for this approach do not show yet great results for a wide-range vocabulary, yet, it appears to be a more promising method than a virtual keyboard controlled by EEG-based interfaces.

3.2.3 Automatic Speech Recognition

The most used method for input text hands-free is the speech-to-text (STT) feature, using Automatic Speech Recognition (ASR). Nowadays this feature can be found on several apps on our smart devices, assisting a user to make calls during a drive or controlling an automated task in smart homes. The accuracy of implementations using this approach is higher than the other studied ones, as its WER can go down to around 5%, using a deep learning approach (Xiong et al. 2017). Another advantage of this approach is, because of its maturity, the myriad of implementations for different languages working with higher accuracy than state-of-the-art SSR systems.

Regarding the use of ASR on our project, the major drawback of this approach is the lack of privacy in communications. Since a user has to speak out loud the messages he wants to send, it compromises the secrecy in public. On the other hand, ASR can be less effective also in public usage as environment noises can affect the effectiveness of this feature. The previous two approaches overcome these concerns, although they are not as mature methods.

3.2.4 Relevant Algorithms

Further explaining the previously referenced machine learning models, namely HMM and LSTM, this section will dive into their description in a more grained way. Both of these algorithms are used in projects related to SSR and ASR, being the most commonly compared for these types of problems. Therefore, a study on these two algorithms was made to better understand their differences.

Briefly, HMM is a statistical Markov model in which the system being modeled is assumed to be a Markov chain and it is used to compute a probability for a sequence of these both hidden or observed states (Jurafsky and Martin 2014). Explaining further what is a Markov chain, it is a model that calculates probabilities of sequences of random states, which values are defined in some set. It assumes the next state of the sequence based on the current one. A Markov model can be represented in a graph, each state being a node, and the transition between states is an edge, weighing the probability of that transition happening in a sequence. Since this transition values are probabilities, the sum of all the edges leaving a node must be 1. An example of this is displayed in Figure 3.3.

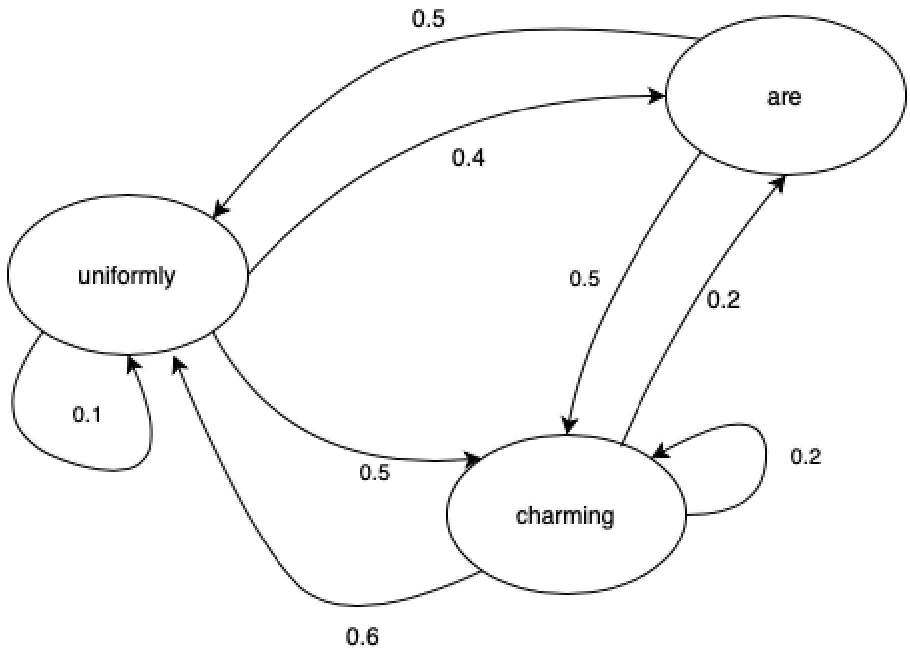


Figure 3.3: A Markov chain for words, showcasing states and its transitions with the probabilities of each.

More formally, this model is defined in three components:

- a set of N states
- a transition probability matrix A, where a_{ij} represents the probability of transitioning from state i to state j
- an initial probability distribution over states, where π_i is the probability that the Markov chain will start in state i

On the other hand, as aforementioned, the HMM is used in cases where the states are both observable or not be observable (hidden), being therefore composed by the following components:

- a set of N states
- a transition probability matrix A, where a_{ij} represents the probability of transitioning from state i to state j
- a sequence O of T observations, each being known in a *vocabulary* V

- a sequence of emission probabilities, each being the probability of an observation o_t being generated from state i
- an initial probability distribution over states, where π_i is the probability that the Markov chain will start in state i

Regarding LSTM, it is a RNN specially designed to handle long time-dependencies. As speech is a time-varying signal with correlations between different ranges of timescales, RNN containing cyclic connections that can handle these sequences, as the output feeds back in the network (see Figure 3.4).

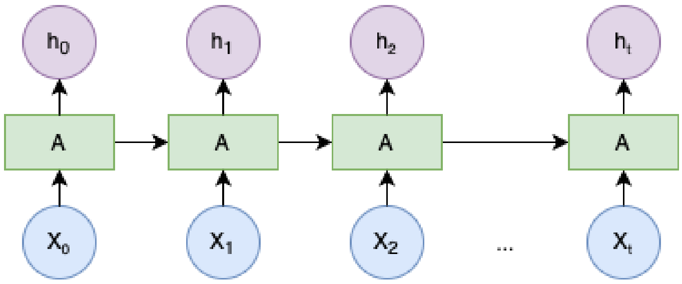


Figure 3.4: A diagram showcasing a recurrent neural network, illustrating the cyclic connections, where the output feeds the network again, creating a chained sequence.

Storing these in internal states, it can hold long-term contextual information. LSTM, in particular, fix some problems of typical RNN, by having in its architecture special units called *memory blocks*. These memory blocks are composed of a memory cell, that stores the temporal state of the network, and by gates which control the flow of information. These gates are the input gate (which controls the flow of input activations into the memory cell), the output gate (which controls the flow of output activations to the rest of the network), and forget gate (which adaptively forgets or resets the cell's memory) (Sak, Senior, and Beaufays 2014). Moreover, this new architecture overcomes the problem of vanishing gradient of typical RNN (Hochreiter and Schmidhuber 1997).

A comparison between the architecture of a simple RNN in Figure 3.5a and a LSTM in Figure 3.5b is showcased, demonstrating the recurrent aspect of these models feeding itself in a cycle, but contrasting in its complexity, LSTM having more layers for the memory cell aforementioned.

Furthermore, building a LSTM network requires a few more steps of research to choose and experiment with different hyperparameters, testing which best fits the problem at hand. Examples of hyperparameters are:

- Optimizer - responsible for the minimization of the objective function of the neural network, being stochastic gradient descent the most common;
- Number of LSTM layers - the number of LSTM layers that are within the architecture;
- Number of Recurrent Units - the number of recurrent units in each LSTM layer.

These and several other hyperparameters have to be chosen and experimented with to find the combination that produces the best results in each problem (Reimers and Gurevych 2017).

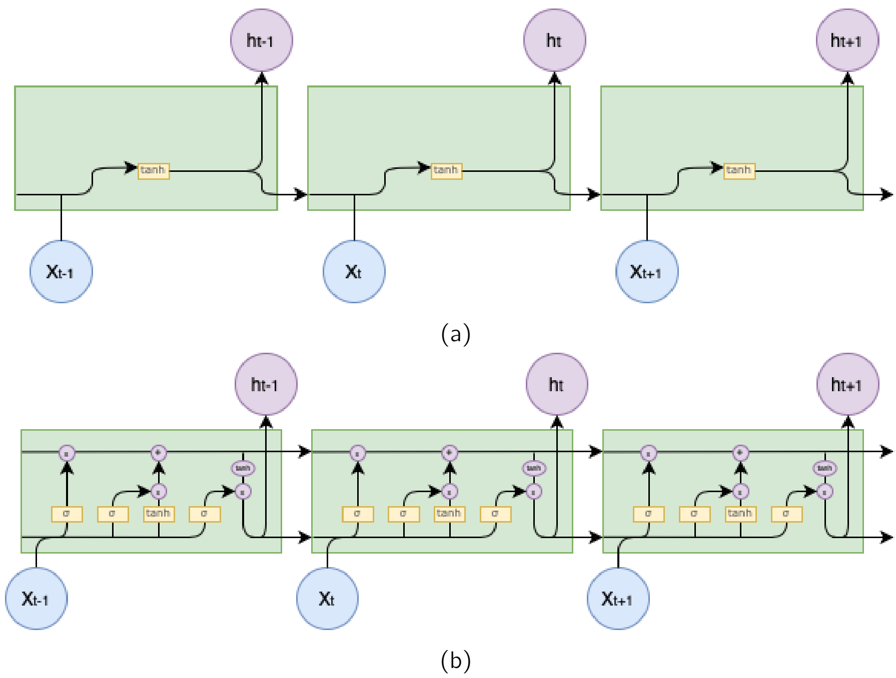


Figure 3.5: (a) RNN architecture. (b) LSTM architecture.

Chapter 4

Value Analysis

The Value Analysis (VA) of a project/product is made in order to boost its value with the least cost possible, all the while maintaining quality. It allows for the cycle of development to stay on track and consistent throughout and can lead to design choices that, ultimately, effect the product itself. VA is a systematic and formal process of evaluation that comes from management activity, that analyzes the function of a product and how does it serve its purpose. VA is expected to be an incremental process of improvement and the goal is to maximize the following function:

$$Value = \frac{(Performance + Capability)}{Cost} = \frac{Function}{Cost}$$

VA defines a basic function for the product which cannot change since it must be the feature that is fundamental and makes the product sell. In addition to this, there are supporting functions which are features that only support the basic function and help in selling, and can change over time and even be eliminated to reduce the cost of the product, increasing its value without removing worth. Value is not about minimizing the cost, but can rather about increasing the function more than the cost.

4.1 Value and perceived value

When attempting to improve the value of a product, there are three elements that ought to be considered: the use value, which refers to how functional the product is (i.e. performance; capability); esteem value, related to aesthetic value and subjective value (i.e. emotional appeal; style); and market value, referring to how much the market is willing to pay for the product. These are subjective, since each customer perceives the value of a product differently (Ulaga and Eggert 2006). The aspects that come into play for a customer to value a product are shown in Figure 4.1.

4.2 Value proposition

"A product's value proposition is a statement of the functional, emotional and self-expressive benefits delivered by the brand that provide value to the target customer" (Aaker 2002). The value proposition of a product comes out of the necessity in identifying the product, the target segment of the market (to whom it is intended to create value), which specific value the application offers and to explain why the client should purchase the product being presented in lieu of those that are already in the market (Prince Sales et al. 2017). Therefore, this definition can be broken into key components, including what the product offers to the

Scope / Domain	PRODUCT	SERVICE	RELATIONSHIP
BENEFIT	Alternative solutions Product quality Product customization	Responsiveness Flexibility Reliability Technical competence	Image Trust Solidarity
SACRIFICE	Price		Time/effort/energy Conflict

Figure 4.1: Table defining value-based drivers (source: (Lapierre 2000)).

customers, what type of value is associated with such offering and to whom the value is being offered. The value proposition should, therefore, in a business perspective, persuade the potential client by inserting the product within the market and showcase how there is value in acquiring the product and why it does better than the rest of competition.

4.2.1 Market analysis

Analysing and identifying target market segments and specific clients and individuals for whom the application has the most potential to deliver value is a crucial step in determining the value proposition (Lanning 1998). As it was aforementioned, the application being developed inserts within the messaging and communication market. Therefore, its serviceable market comprises of anyone that communicates through digital devices. There are two, however, segments that are especially targeted and which this project aims to target, one being disabled individuals who have arm, hand or vocal impairments and want to communicate with other peers like regular people, and the second being users that value the privacy of their data and the exchange of messages within a secure and private environment. These two segments can overlap.

The most popular global mobile and desktop messenger applications have a staggering number of monthly users which perfectly displays the immense potential of usage and revenue of this market. For example, WhatsApp has 500 million daily active users (Facebook 2019), WeChat attains 113.7 million monthly active users (Tencent 2019) and Facebook's Messenger registers traffic of 1300 million monthly active users (Johnson 2017). The aforementioned comprise of three of the most popular messaging applications (Figure 4.2) that can be found on both mobile and desktop applications.

All the applications shown in Figure 4.2 are centralized. That is, there are central servers in which data passes through between the sender and the recipient. Telegram is highly regarded as the most secure mainstream messaging application, boasting 200 million monthly active users (Durov 2019). All of the aforementioned applications do not have support for disabled individuals, which are one of the target audience, as explained in Section 4.2.2. Incidentally, in the United States of America alone, 2.1 million people live with limb loss in 2014 (Amputee Coalition 2014). This exhibits an opportunity to integrate these individuals and incorporate within a single application that is both inclusive for disabled individuals and can also be used by regular users.

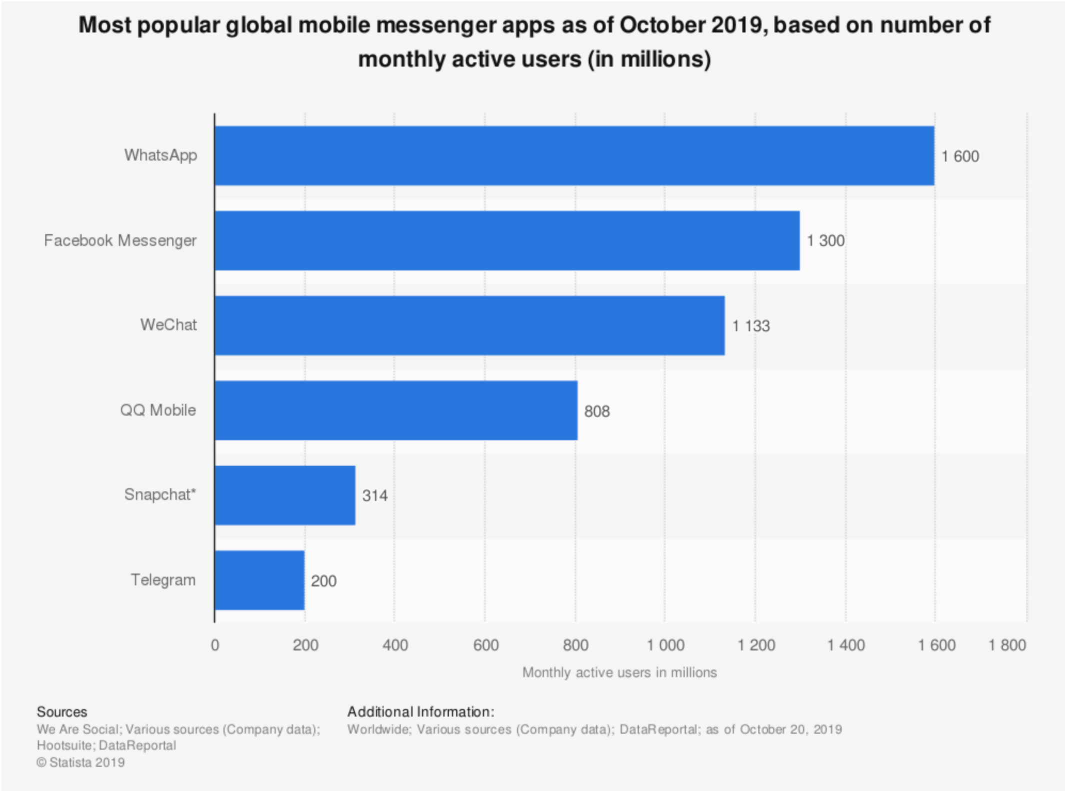


Figure 4.2: Histogram showing the most popular messaging apps with the number of monthly active users, as of October 2009 (source: (Clement 2019)).

For a business, it is important to define three main segments in the market it aims to address: Total Addressable Market (TAM), Served Available Market (SAM), and Target Market. After the analysis above, it can be stated that the total of monthly messaging apps active users is approximately 6 billion, being that the total market reachable. That is, in other words, our TAM. The concern in privacy narrows down the project's focus to a SAM that encompasses users who deeply care about these concepts and aspire to take action switching to new solutions where they find these ideas being applied. That might be the example of the users of Telegram and other similar apps, which subsequently reducing the target audience to 200 million monthly active users. The Target Market of this application encloses people with arm/hand disabilities, as we provide the solution at hand provides solutions for these individuals to communicate with peers using fairly new set of technologies. These disabilities stem from a myriad of different sources such as Parkinson's, amputations, or paralysis. Regarding upper-limb amputees, there are 3 million people with these conditions, of which 30% are in developed countries (LeBlanc 2008). Within the United States of America, it is revealed that nearly 1 in 50 people have paralysis from different causes, a total of 50 million (Armour et al. 2016). Considering the same ratio in worldwide internet users (estimated to be 4 billion people (International Telecommunication Union 2019)), people who are reasonable to use our platform, it translates in a total of 80 million people with paralysis. For this matter, individuals affected by Parkinson's we will not be counted, considering this illness mostly appears after the age of 60 (Reeve, Simcox, and Turnbull 2014), and the number of internet users in this age is not relevant for these values. The aforementioned niche market has little to no competitors for the type of platform and service provided and after the previous analysis, it can be concluded that the Target Market for the current application revolves between 80 and 85 million people, which are directly designated as target audience.

4.2.2 Value proposition of our application

The main value that stems from this application entails support for disabled individuals and heightened levels of privacy of data. In a longitudinal perspective of the value that is being offered, it is important to state that the value of this product will always be present since it is inherent to its implementation and design choices. The proposed solution encompasses both SSI and BCI for the interaction counterpart of the application. This is one of the relevant aspects which differentiates the solution with the rest. It essentially provides value by including and giving support for disabled individuals, more specifically people with arm/hand and speech impairments, and presenting these with a way to efficiently communicate with other peers, regardless of disability status.

On another hand, every user partakes in a decentralized ecosystem that is user-centric that aims to have a high throughput of message exchange and maintaining a high bottom-line of privacy of data. This not only showcases a new approach to decentralized messaging applications whilst also maintaining a high level of security through cryptography that is peer-reviewed. Therefore, users may find value in secure communication between peers, something that is above the level of the mainstream solutions that are presented currently.

Another aspect that further boosts the value of this application is its wide range of applicability. In other words, this messaging application can be used in different scenarios (e.g., education). By including support for these type of disabled individuals, many barriers are broken down and allow for a more efficient peer-to-peer contact.

On a business perspective, the users could have access to the application for free, albeit some premium features would need to be purchased for the user to have access to these. This would be the only expense from the user, which would have benefits even if the free version was used. On a long-term perspective, the product would not lose value for its clients, since the former will be maintained to incorporate new aspects regarding security and peer-reviewed cryptographic methods. The same occurs on the SSI and BCI counterpart, in which we would strive for a better accuracy in the interaction between the user and the interface.

Inserted in this dissertation's scope, this project's aim is to provide a new insight on decentralized application with support for disabled individuals, therefore pushing further the state-of-the-art when it comes to accuracy in SSI and BCI and privacy of data in messaging platforms that are decentralized and based on distributed ledger technologies that is viable on a production level.

4.3 Front End Innovation

On the product development paradigm, the Front End (FE) is considered as the first stage of new product development, which concerns the period from the idea generation to its approval for development, or its termination for specific reasons. Essentially it is the starting point where the main opportunities are outlined and concepts are developed before entering the formal product development process (Peter A. Koen, Bertels, and Kleinschmidt 2014), partaking the first part in Figure 4.3, where the New Product Development (NPD) is where the products are indeed developed, proceeding with its launching and commercialization activities.

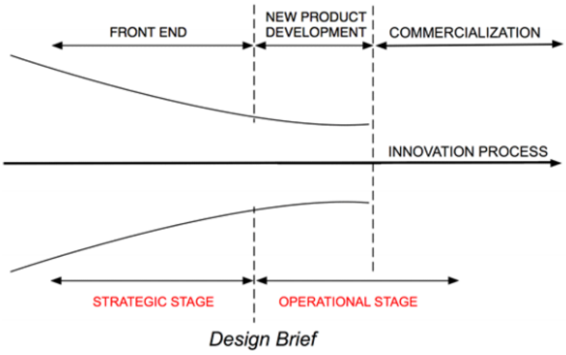


Figure 4.3: The overall product innovation process (source (P. Koen et al. 2001)).

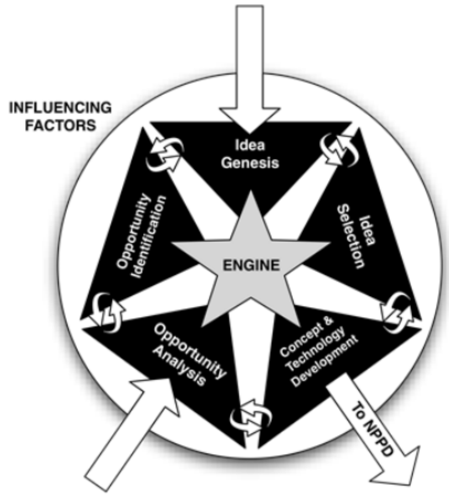


Figure 4.4: Diagram showcasing the New Concept Development model, according to Koen, reproduced by the latter (source (P. Koen et al. 2001)).

The New Concept Development (NCD) diagram showcased in Figure 4.4 provides an overview of the main FE activities that transpire prior to the product development stage and commercialization as is split into three counterparts: the surrounding influencing factors, the engines that drives the activities of the FE and five activity elements, of which this document will frame in relation to the application being developed.

4.3.1 Opportunity identification

The opportunity identification part relates to the technological changes and opportunities that are identified and are meant to be pursued (Peter A Koen et al. 2002). Concerning the project, the opportunity that was identified was the apparent lack of support for individuals with arm/hand impairments to communicate with other people on a common messaging platform. To add to this, there are not available solutions that incorporate BCI and SSI on a decentralized ecosystem with the privacy of communication in mind, with the potential of growing into more than a conventional peer-to-peer or chatroom messaging platform and be applied to a wider range of scenarios.

Moreover, the presented application showcases a new product that encompasses two growing fields: machine learning and blockchain and its variants. Decentralization of data allows for a higher level of security in communication between peers and a new approach to messaging.

Market research and technology trend analysis are tools that can aid and reinforce this concept. Referring to this project, the increase of scepticism of privacy on messaging applications creates a space in which this application can strive (Schrittwieser et al. 2012). Furthermore, it was recorded that approximately 2.52 people worldwide made use of messaging applications at least once time per month (Enberg 2019). The messaging market is extensive and panders to the human's most natural trait, which is communication. With this in mind and with growing concerns regarding the security of data in communication, messaging platforms like Telegram have been thriving mainly due to the end-to-end encryption features it provides. However, these methods have not been peer-reviewed in an open-source environment and have been found to be vulnerable to certain attacks (J. Lee et al. 2017). This poses as an opportunity to bring a peer-reviewed, decentralized platform that utilizes encryption methods that are secure to at least quantum levels, all the while providing support for disabled individuals at arm and vocal level, something that can not be found currently.

4.3.2 Opportunity analysis

Opportunity analysis pertains to if the idea/opportunity being assessed is worth pursuing. This encompasses market and technological assessments. For the latter, refers to the level of maturity of the current technologies that can aid development and if it can be reliable on a production level (Peter A Koen et al. 2002). Framing this activity within the project being developed, this application exhibits an opportunity to showcase a new prototype that concerns subvocalization and decentralization of network for communication purposes. The value inherent is that it is inclusive to arm and vocal impairment, thus allowing people with this kind of disabilities to communicate with virtually anyone on a platform that is similar to others within the market, doing so with security and communication that is not centralized and compromisable.

The methods to validate this activity are similar to those used in Opportunity Identification (Section 4.3.1), which can include market research and technology trend analysis.

4.3.3 Idea generation and enrichment

The idea generation and enrichment activity concerns the genesis, development and maturation of a specific idea and is meant to be evolutionary (Peter A Koen et al. 2002). About the project, the idea generation stemmed from the will to encompass two different fields that were booming and gaining traction, those being machine learning and blockchain/distributed ledger technologies. Besides, a distinctive lack of BCI solutions coupled with SSI within the messaging paradigm led to the genesis of this project. The current state-of-the-art of SSR does not showcase high accuracy, therefore providing an opportunity to further push the limits of the current state of the field.

When it comes to decentralized systems, the notion of using a decentralized network came with the idea of empowering the user on a more democratic ecosystem. Additionally, the advantages of using decentralized systems over centralized ones on communication platforms are crucial when it comes to privacy and integrity of data that is being exchanged.

As a method to verify this activity within this project, a former role was assigned for supervisors and co-supervisors to coordinate ideas from the generation of the project, throughout its development and assess it iteratively to maintain the development cycle on track.

4.3.4 Idea selection

Frequently, the main conundrum is not creating new ideas and innovations but rather choosing the best one that would achieve a higher value. This process is not straightforward but rather iterative that should culminate in a final outlining of the product idea (Peter A Koen et al. 2002). Concerning the project presented in this paper, the selection of the idea stemmed from an iterative process of what would be liable, market-wise and value-wise. The main focus of the application is the push the state-of-the-art on both synthetic telepathy and decentralized systems on messaging platforms with private and secure data exchange. The conjunction of these target markets where the support for disabled individuals are scarce, all the while addressing common messaging security issues.

In all, techniques that aid verifying the Idea Selection encompass the success probability on technical and commercial, leveraging with a selection process overseen by supervisors which prompt feedback on different methodologies throughout. The latter is an iterative process (displayed in Figure 4.5) and has been present in choosing which technologies would be adequate in addressing the problem the application is addressing. An example of this was straying away from a pure blockchain solution as it was initially thought out, opting for variants that allow for high throughput within the network.

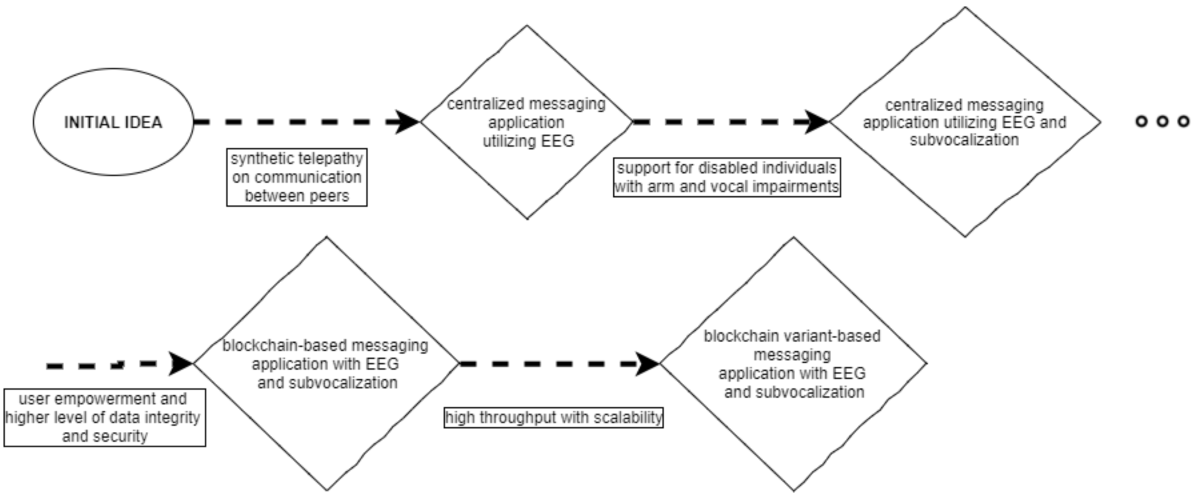


Figure 4.5: Idea selection process for the application at hand, in which culminates in the last stage that will eventually lead to NPD.

4.3.5 Concept definition

The Concept Definition pertains the final element and stage of the NCD. As shown in Figure 4.4, it is the only exit to the NPD. To reach the latter, it must make a compelling case for investment in the proposition. Methods to validate this activity include setting criteria that describes how the application fairs in term of the market, how it addresses it and how it adds values in comparison to others (its objectives).

For this project, the main objectives, as it was aforementioned, are to provide a new solution to the messaging communication panorama that offers state-of-the-art privacy and security, all the while making it inclusive for individuals with arm and vocal impairments and not only. The serviceable market for this application is widely broad, since it can be used by both regular or disabled individuals. However, it is tailored to the latter and are targeted for individuals that value the privacy of their communication and data when messaging other peers.

4.4 Canvas Business Model

A business model describes the rationale of how a company generates value by analyzing a problem, creating a solution, and using the correct means to supply its customers. A business model is the logical process to define who the customer is, what the problem is, how to solve it and how to benefit from it, so a product or service is not a business model by itself (Osterwalder and Pigneur 2013).

The Canvas model is a simple and visual model for business definition which briefly introduces the key aspects of a company. It analyses nine viewpoints on which the company and its activities are based on, which are the following:

- **Key partners** - Who are the key partners/suppliers? Which resources do we acquire from them?
- **Key activities** - Which activities are fundamental for our value proposition?
- **Key resources** - What resources do our value proposition require?
- **Value propositions** - What do we offer to the customer?
- **Customer relationships** - What relationships do we establish with our customers?
- **Customer segments** - For whom are we creating a product?
- **Channels** - How do we reach customers?
- **Cost structure** - What are the most important costs for creating our product?
- **Revenue streams** - For what are our customers paying? How do we profit?

The Canvas model attempts to answer these questions above in a simple format. For our project, the Canvas model is shown below in Figure 4.6.

As shown in the Canvas Model, the key activity is indeed the development of the messaging platform. This presupposes the integration between the sensors and our platform. The target audience of the platform are people who desire to message securely and privately, although keeping in mind people with some specific disabilities and pursue their inclusion within the community. The way the majority audience is reached will be exclusively online. However, during the development phase, some tests might be arranged in person, as it is planned to establish contact with hospitals and related entities to aid testing some features that aim to help impaired subjects.

4.5 Value chain

The set of activities that a company performs to create value can be defined in a value chain. As Porter (2008) proposed, a general value chain (shown in Figure 4.7) allows companies to examine all the activities and how they are connected. The way these operations are executed defines costs and profits, and the analysis with this tool eases the understanding of the value.

This model divides the activities in two groups: primary and support activities, both of which will be detailed in subsequent chapters.

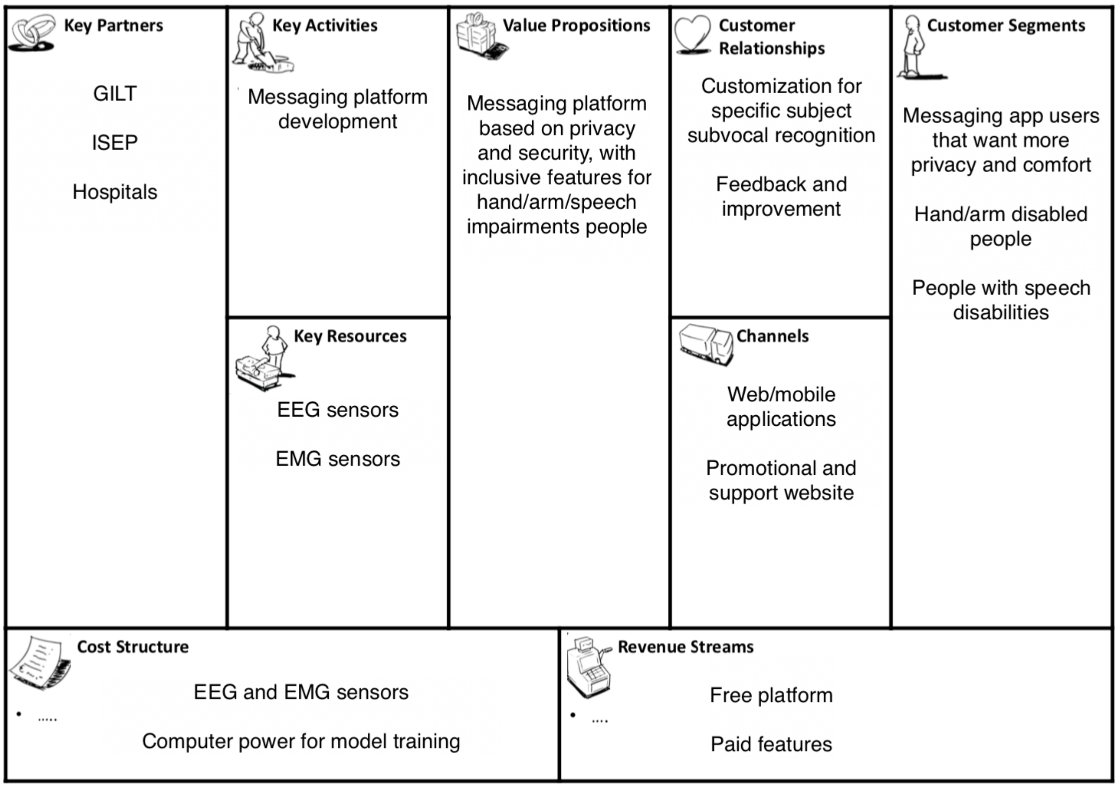


Figure 4.6: The Canvas Model for the project, showcasing each aspect that were explained prior.

4.5.1 Primary activities

This group refers to the activities related directly to the product (i.e. creation, sales, maintenance) which can be classified in five different sections:

- Inbound logistics - related to receiving and storing inputs.
- Operations - related to the transformation from input into outputs (creation of value).
- Outbound logistics - related to the delivery of the product to the customers.
- Marketing and sales - related to customers gathering and communication with them, persuading them to buy a product from us instead of our competitors (value for the customer).
- Service - related to post-purchase services, maintaining value for the customer.

These mainly flow from left to right as shown in Figure 4.7, from the reception of the supplies to the transformation of materials, to store, and then finally to sales and services for after-purchase support.

4.5.2 Support activities

The support activities groups every activity that does not create value. However, these can assist each one of the primary activities, regardless of what section it is, supporting throughout the entire process of the product. These can also be divided into four types:

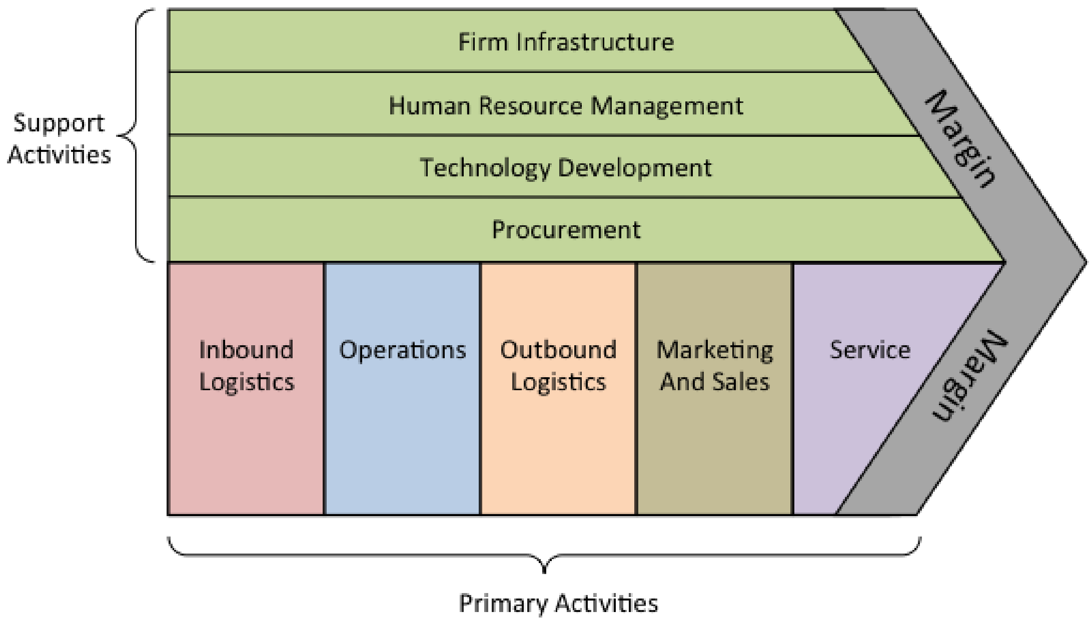


Figure 4.7: The Porter's value chain, as proposed by Michael Porter (source: (Porter 2008)).

- Firm Infrastructure - related to the company's support systems such as accounting, legal, administrative, and general management.
- Human Resource Management - related to recruitment, motivation, and retention of employees.
- Technology Development - related to the information and knowledge of the company.
- Procurement - related to finding suppliers to get resources needed to operate.

Although this model might fit for most companies, the new era of technological companies do not. This paper's project does not have inputs by suppliers since the product is, essentially, knowledge created by the community and the developers themselves. Thus, a new perspective must be taken to analyze how value is created within this application.

4.6 Value network

Proposed by Verna Allee, the value network is a new model to analyze a company based on its relationship network with other entities. Although knowledge and intangible value exchanges are the base of most emerging companies, most business models do not consider these. This model takes into account two key types of value exchange, those being tangible and intangible. The first is related to revenue and goods that two entities transact with each other. The latter refers to benefits and knowledge trade, which is not quite considered in other models, although it is a key element in some businesses like ours. In a value network model, the focus is on the stakeholders of a project and their exchange of tangible and intangible benefits, usually all providing and receiving value, this way contributing to the success of each and the success of the network (Verna 2008).

As this model appears more promising to fit the needs of the project and its field of work, when compared to a value chain, the value network refering to this current project is shown in Figure 4.8.

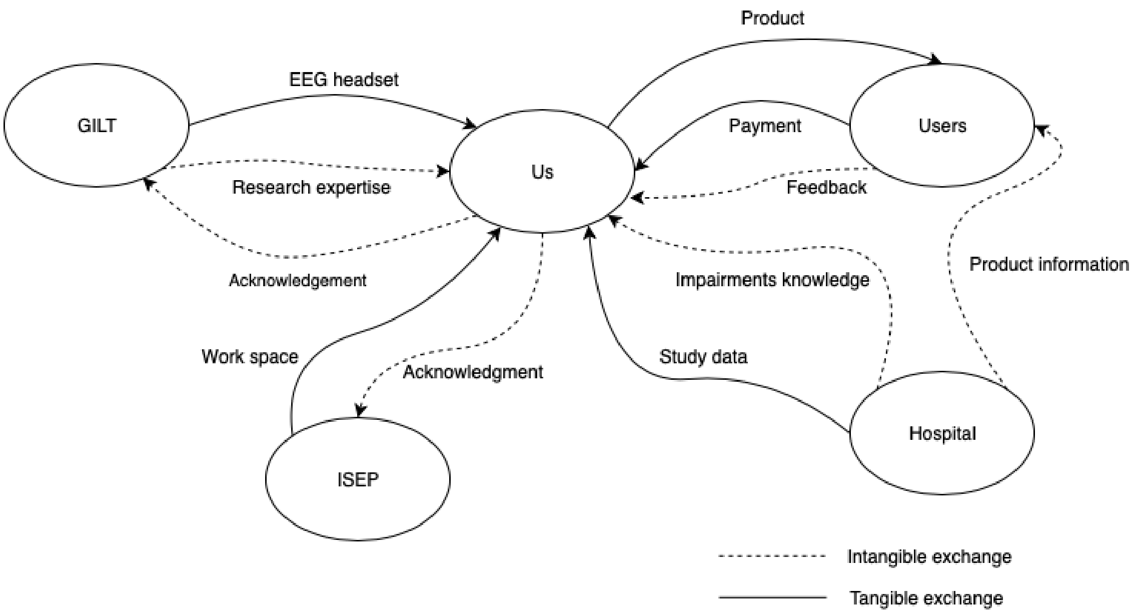


Figure 4.8: Proposed value network for the project related with this paper, following Verna's model.

As Figure 4.8 depicts, since we are working with public institutes that are essentially helping us with intangible goods, the value exchange occurs mainly towards us, as we do not have much to offer besides the research and build of the product to the costumers, and recognition to those who support our project. These exchanges are a key factor for the research and development, as otherwise the development group wouldn't have access to an EEG device that GILT offers us to work with, nor direct contact with impaired people who we are offering value, that can be established through the hospitals.

4.7 Analytic Hierarchy Process

AHP is a process that helps in decision making providing a structured representation by analyzing the criteria elements and comparing them pairwise, creating a priority scale (Saaty 2008). AHP, in a simple explanation, decomposes decision making in four steps:

- Problem definition
- Criteria definition
- Pairwise comparison to weight priorities
- Weight priorities and alternatives

Since this document focus on the development of the Human-Computer Interface (HCI) component of this project, the criteria represented below only refers to decision making related to it. Trying to choose a new approach for these components, answering "What approach/technology suits best with our project requirements?", an AHP with be structured for future decision making. Thus, the analyzed criteria are:

- Interaction speed: how fast can a user give inputs and receive feedback
- Privacy: how much privacy can be guaranteed
- Comfort: how comfortable and convenient it is for the user

This process will be used to analyze in BCI and SSR components since these three factors can be applied to both. After defining the problem and the criteria, a pairwise comparison must be done, on as scale of 1 to 9 (Saaty 1988). Table 4.2 displays the first matrix for the AHP analyzes, were it compares all factors with each other.

Table 4.1: Pairwise comparison matrix.

	Interaction speed	Comfort	Privacy
Interaction speed	1.0	8.0	4.0
Comfort	1/8	1.0	1/4
Privacy	1/4	4.0	1.0

Translating the table above to a matrix notation:

$$M = \begin{bmatrix} 1.0 & 8.0 & 4.0 \\ 1/8 & 1.0 & 1/4 \\ 1/4 & 4.0 & 1.0 \end{bmatrix} \tag{4.1}$$

With this matrix, it can be calculated the Priority Vector. This vector is the average of each row of M normalized. With M normalized as

$$\begin{bmatrix} 8/11 & 8/13 & 16/21 \\ 1/11 & 1/13 & 1/21 \\ 2/11 & 4/13 & 4/21 \end{bmatrix} \tag{4.2}$$

, the priority vector is:

$$v = \begin{bmatrix} 0.701 \\ 0.072 \\ 0.227 \end{bmatrix} \tag{4.3}$$

The next step it to calculate the Consistency Ratio (CR), which certificates the consistency of the judgments. If this value is higher than 0.1, then the judgments are considered not trustworthy.

Before that, we need to calculate

$$M * v = \lambda_{max} * v \iff \begin{bmatrix} 2.18271 \\ 0.21617 \\ 0.68931 \end{bmatrix} = \lambda_{max} * \begin{bmatrix} 0.701 \\ 0.072 \\ 0.227 \end{bmatrix} \tag{4.4}$$

in order to get λ_{max}, which is needed for the CI calculation. Having

$$\lambda_{max} = average\left\{\frac{2.18271}{0.701}, \frac{0.21617}{0.072}, \frac{0.68931}{0.227}\right\} = 3.0542 \quad (4.5)$$

This value is calculated with this following expression:

$$CR = \frac{CI}{index} = \frac{0.0271}{0.58} = 0.046724 < 0.1 \quad (4.6)$$

, where Consistency Index (CI) is:

$$CI = \frac{(\lambda_{max} - n)}{(n-1)} = \frac{(3.0542 - 3)}{(3-1)} = 0.0271 \quad (4.7)$$

, where n is the initial matrix dimension.

Since CR is less than 0.1, it can be concluded that the results are consistent and reliable. By the results of the AHP, it can be concluded that the most important factor is the Interaction speed, followed by the privacy and then the comfort. Thus, the decisions on which and how to implement the components of the HCI will consider this results.

4.7.1 AHP application example

In this section, it will be applied the AHP previously defined to a real case for this problem. For this, it is constructed one matrix comparing alternatives pair-wise for each criterion, calculating the priority vector that compares the importance of each alternative regarding the respective criterion. The three compared approaches are ASR, BCI, and SSR. Each of these approaches presented as alternatives are described further in Section 3.

Table 4.3: Comparison matrix for the Interaction Speed criterion.

Interaction speed	ASR	BCI	SSR	Priority Vector
ASR	1.0	9.0	1.0	0.474
BCI	1/9	1.0	1/9	0.052
SSR	1.0	9.0	1.0	0.474

Table 4.5: Comparison matrix for the Comfort criterion.

Comfort	ASR	BCI	SSR	Priority Vector
ASR	1.0	9.0	4.0	0.664
BCI	1/9	1.0	1/9	0.052
SSR	1/4	9.0	1.0	0.284

The results of the priority vectors in Tables 4.4, 4.6 and 4.8 showcase which approach is relatively better at each criterion. In order to find the most suitable solution for the problem the following equation shows an overall result.

Table 4.7: Comparison matrix for the Privacy criterion.

Privacy	ASR	BCI	SSR	Priority Vector
ASR	1.0	1/7	1/9	0.057
BCI	7.0	1.0	1/3	0.295
SSR	9.0	3.0	1.0	0.648

$$\begin{bmatrix} 0.474 & 0.664 & 0.057 \\ 0.052 & 0.052 & 0.295 \\ 0.474 & 0.284 & 0.648 \end{bmatrix} * \begin{bmatrix} 0.701000 \\ 0.072000 \\ 0.227000 \end{bmatrix} = \begin{bmatrix} 0.393 \\ 0.107 \\ 0.500 \end{bmatrix} \begin{matrix} ASR \\ BCI \\ SSR \end{matrix} \tag{4.8}$$

After combining all the priority vector in a matrix and multiplying by the priority vector previously calculated, which weights each criterion priority related to the problem, it is shown in the Equation (4.8) which alternative suits best our project.

As showcased, the best approach for our problem is SSR, followed by ASR and then BCI.

Chapter 5

Design

This chapter describes the planning of the design used in the development phase of the project, by outlining its main components and its relationships, and showcasing an overall concept of the architecture of the platform created.

5.1 Functional and non-functional requirements

As with every project, the design process must be preceded by a thorough requirement analysis. In this chapter, the functional and non-functional requirements of this project will be explained according to the FURPS+, which is a model for requirement specification in projects. It represents an acronym, meaning Functionality, Usability, Reliability, Performance and Supportability, thus dividing the model in five domains. The "+" symbol represents other attributes as Implementation, Interface, Operations, Packaging, and Legal (Larman 2005).

5.1.1 Functionality

The Functionality counterpart directly represents the functional requirements of the project which, in turn, are related to the user stories and use cases. As it was mentioned in Section 2.2.2, the current project does not have many use cases, since the focus will be on providing an inclusive and safe experience. Table 5.2 depicts the user stories of the system, which are common in any messaging platform.

Table 5.1: Table showcasing the user stories of the system and the correspondent actor.

Actor	Use Case
As a **common user**, I want to...	Create contact
	Remove contact
	List all contacts
	Choose contact/chatroom
	Navigate in a chat's message history
	Send message
	Receive message

In addition to the use cases, the functionality aspect of the application also correlates with Security. For this project, the Security of the application has as requirements the existence of a public/private key pair to prove the authorship of data to specific users. Another requirement is it is mandatory for all users to have the same version of the backend/frontend bundle and, therefore, system configuration. Communication between users also has to be end-to-end encrypted.

5.1.2 Usability

This topic covers the requirements regarding interaction with users through user interfaces. For this project, usability is key given the target audience and their barriers in commonly used interfaces, which we are trying to overcome. One of the requirements for the User Interface (UI) is to be intuitive and compatible for usage through an EEG headset enabling a BCI. Therefore, the frontend has to have a simple and intuitive design for users from different backgrounds and age ranges. Moreover, the frontend must also support text input through silent speech using SSR techniques. Finally, the frontend must have a consistent design spanning its screens and be responsive.

5.1.3 Reliability

This section concerns requirements related to service availability and failure recovery. For this project, the project at hand must be able to handle increased network traffic. Additionally, no central points of failure that propagate and affect several users may occur. In addition to this, redundancy of data is a requirement for users to maintain data even if the original author is offline.

5.1.4 Performance

Performance covers requirements associated with the response time and recovery time of the system. Specifically to this project, the system must have a reduced response time and prompt feedback. Moreover, the time of each transaction must be low and consistent as the system scales with more users joining the network.

5.1.5 Supportability

For this section, supportability is concerned with characteristics such as testability, adaptability, maintainability, and scalability. The latter aspect is a requirement, it being scalable without compromising performance to a limitless amount of users. Additionally, the platform has to be available in all operating systems and web browsers. The system has to be inclusive for users with upper-body limb impairment and be able to pander to both regular and disabled users and maintain usability. Lastly, the system must allow text input from the silent speech feature for, at least, the English language.

5.1.6 + (Others)

This final section covers physical, design and implementation constraints. For this project, one non-functional requirement for this domain regards the need for the system to be GDPR compliant. Another constraint is that the system will use the Emotiv EPOC+ headset, as it is the only EEG device the team has access to at this time.

5.2 Approaches assessment

Following the previous analysis of approaches and technologies, this section evaluates these, tackling the specific problem of this project, and aims to a better selection of which ones must be used. This section will display a comparison between the previously mentioned approaches and other solutions, and the decision making behind process the choices made.

5.2.1 Criteria and Approach Comparison

Regarding the topic of this document, it concerns the development of three components of the project: BCI, SSI, and a Frontend app. The first two will serve as interfaces between the user and the latter.

As previously analyzed, BCI can be adopted for the user to navigate within the Frontend app, allowing hands-free control over it. In the Section 3.1.2 were presented two approaches to this interface, invasive and non-invasive. Since the invasive approach requires surgery in each subject that uses the platform, it is not by any means what we can or even want to implement. Therefore, the chosen approach for BCI is the non-invasive one. Between technologies that can be used for the BCI, it is beforehand selected the usage of the Emotiv's EPOC+, since we are working in a partnership with GILT and they already own this model prepared for EEG. As beforementioned, this product is a powerful tool for EEG related projects, since it has pre-programmed code for detection of some mental commands, facial expressions, and emotions. Most, if not all, of the used mental commands for navigation, are some of the already built-in ones, improving the development speed of this component. With this, the only task needed for the usage of this product with our project, after integration between components, is that the user using this device pre-trains this device for his brain activity, allowing the Emotiv's software to detect which command is being produced. For integration with this product, Emotiv also provides developers an application programming interface (API), called Cortex API, which allows any authorized software to receive EEG data and results of the classification of mental commands. This relationship between pieces of software will be further analyzed in Section 5.5.1.

Concerning the text input, and following the AHP from the Section 4.7, it is concluded that the best approach for this component is using an SSR, as it maintains privacy while using in public, as well as a theoretical more accurate method when more mature, comparing to STT, since it not affected by background and environment noises. Within the SSR component, a myriad of approaches must be taken into account, since there are multiple ways to implement it. As aforementioned, there are some studies that try a session-independent and/or subject-independent models, which would be the best implementation for our project as it is assumed the usage from different people and between different sessions. In order to implement SSR, the chosen dataset is the *EMG-UKA* corpus as it was the only publicly available dataset for recorded EMG data for this purpose, and has enough data for testing different machine learning algorithms, aiming to build a SSI with the lower WER as possible. The first chosen approach for the SSR algorithm is using a LSTM-based architecture, exploring a deep learning alternative. If the results turn out to be unsatisfactory and if possible during the period of this project, another approach using HMM might be tested, as it is the model used by Wand, Janke, and Schultz (2014), creators of the *EMG-UKA* corpus. This approach was the one that has shown the best results among the studied ones.

The focus of this project is on the presentation of methods of interacting with a computer for hands-free message exchange. Nevertheless, a central application has to be built for the user to interact with, and for enabling the message exchange. That said, the emphasis will not be on the selection of approaches for the Frontend. Notwithstanding, on the Frontend app the first decision to be made was between mobile or desktop approaches. Since the interface devices will not be completely portable, and we are aiming for an audience not quite capable of managing a smartphone, it was decided for this first implementation to build a desktop-driven platform. Considering the team has vast experience with web technologies, it was easily decided to build a web app that could run on any device with a browser, expanding further our audience. Taking into account the popularity of frameworks for building web applications (StackOverflow 2019), the chosen one was React, as the team also has extensive expertise with this technology. Likewise, for the SSR system, there was also a need to choose which from different possibilities of programming languages and frameworks to work with. Python is a very popular programming language and is the fastest-growing major programming language (StackOverflow 2019). Amidst all the frameworks that might be used in a machine learning project, PyTorch stood out as the go-to and most growing machine learning framework (He 2019).

Nevertheless, other frameworks and dependencies might be used within the platform's software, although the mentioned above are the most important as cornerstones of the development.

5.3 Conceptual Design of the Application

The proposed solution encompasses SSI and BCI for the interaction of an application that partakes in a decentralized ecosystem that is user-centric, with high throughput and maintaining a bottom-line of privacy of data and message exchange. It envisions a way for people with arm/hand and speech impairments to be able to communicate with anyone.

It is, therefore, divided into two distinctive counterparts: human-machine interaction through methods of machine learning and decentralized communication between peers. This is further outlined in Figure 5.1.

From a conceptual point of view, for the BCI component of the project, an EEG headset (more specifically, the Emotive EPOC+) will be used alongside the Emotiv BCI software for the training and classification of the data gathered from the user. Some mental commands are already predefined, which speeds up the project development considerably. These commands are applied to the UI navigation. The latter ought to be accessible and intuitive for users with or without impairments, with a specific design that allows navigation through an EEG headset or conventional means (mouse/keyboard inputs). On the other hand, text input will be dealt with by the SSR counterpart, using software developed by the team, with a corpus containing data gathered from surface electrodes performing EMG.

Since the user interacts through unorthodox means, for the UI a new design idea had to emerge to guarantee that it supports these new ways of input and eases the process by a fluid and intuitive feedback. Thus, besides from regular click/tap to interact, the user can also navigate through different chat rooms using four different commands, *pull*, *push*, *left*, and *right*. Simulating a three-dimensional space, the *pull* and *push* commands, pull closer to and push away from the user view, respectively, a whole screen related to a single chat room (Figure 5.2).

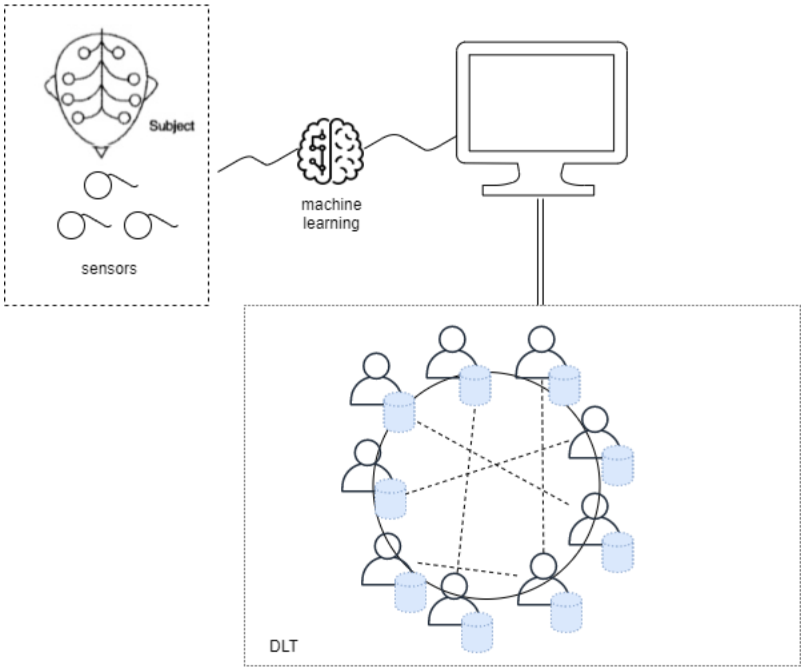


Figure 5.1: Overall conceptual view of the platform, outlining the interaction between the user and interface and also the decentralized system where the user is inserted.

The remaining *left* and *right* commands are used to slide through a carousel-type view with each chat room, always fixing a chat room in the middle, until the user pulls the one selected (see Figure 5.3).

The other counterpart refers to the decentralized backend in which each peer holds part of the Distributed Hash Table (DHT) and information is shared and verified by the nodes that make up the network. Each node has its own identity and private and public shared data. Each data has ownership of a specific user, in which the user has control of. Having this type of system allows for future expandability of features and expand from peer-to-peer messaging to a social network with shared public data. Each user has the same influence over the network, there are no supernodes.

5.3.1 EEG Electrode Positioning

Focusing on the design and definition of the EEG step for the BCI counterpart of this project, the electrode placement must be established. In order to standardize the electrode positioning, the International Federation of Clinical Neurophysiology adopted a model called *10-20 System* (Klem et al. 1999). This protocol defined the placements and names of 21 electrodes on the scalp. Each name is composed by a letter and a number, in which the letter refers to the region of the brain where the electrode is positioned (F: frontal, C: central, T: temporal, P: posterior, and O: occipital) and the number refers to the hemisphere (odd and even numbers, for the left and right hemispheres, respectively). Later, an extension to this system was added and the number of electrode positions standardized increased from 21 to 74 creating the *10-10 System* (Klem et al. 1999; Rojas et al. 2018). These are illustrated in Figure 5.4.

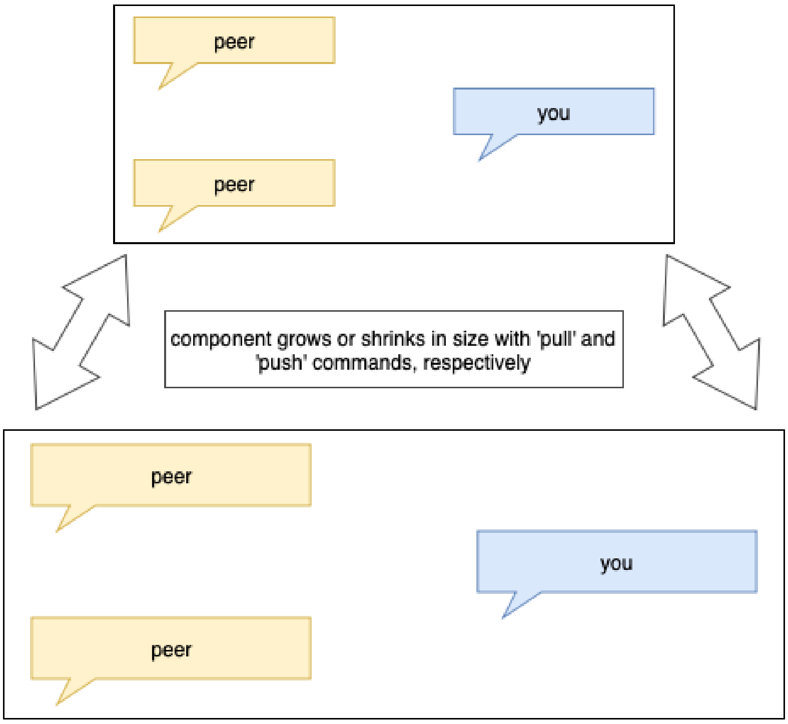


Figure 5.2: Diagram explaining how *pull* and *push* commands effect the UI.

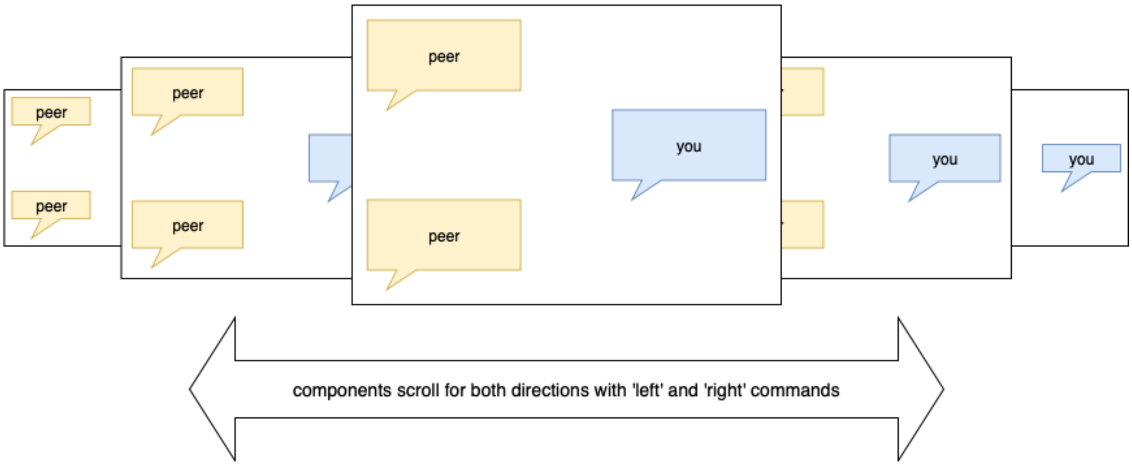


Figure 5.3: Diagram outlining how the *left* and *right* commands are used for navigation.

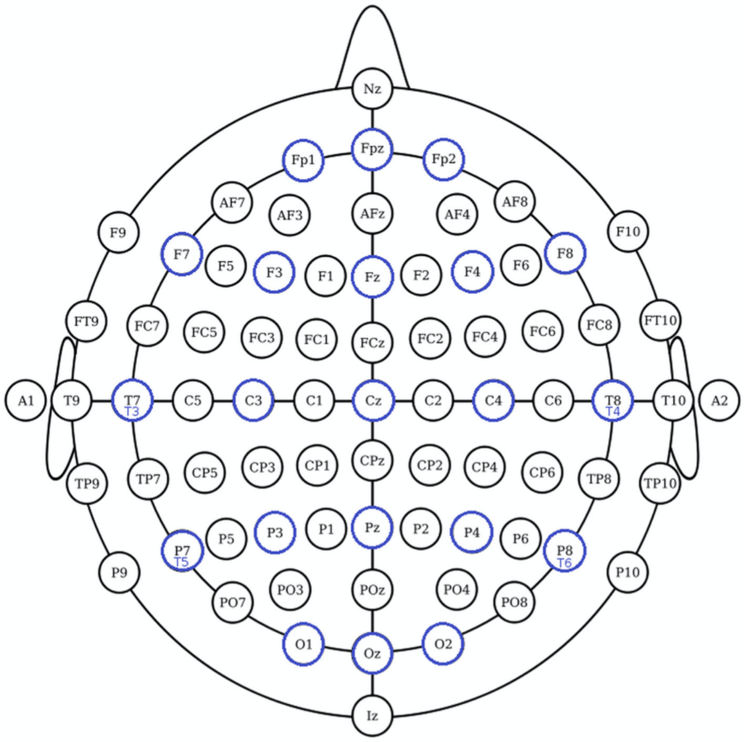


Figure 5.4: EEG electrode placement following the *10-10 System*. Blue circles represent the *10-20 System* electrode positions (Rojas et al. 2018).

As previously stated, the EEG headset used for this project is the Emotiv EPOC+. This headset is designed to cover different regions of the brain, namely temporal, parietal, and occipital, and has a special focus on the frontal and prefrontal lobes. The 14 electrodes positioning is already defined following the aforementioned system, specifically in the positions AF3, AF4, F3, F4, F7, F8, T7, T8, P3, P4, P7, P8, O1, O2, as showcased in Figure 5.5 (Emotiv 2019).

5.4 Architectural Design of the Solution

This section will present an overview of the architecture of the entire solution, showcasing its components and how they interact, from the user input interfaces to the platform network and data management components. This is showcased in Figure 5.6 where it describes in a high granularity each component and its relationships with others.

Regarding the components related to the focus of this work, the two elements that operate as interfaces for the Frontend component are the Emotiv App and our SSR App, the first responsible for the BCI which will lead to navigation actions within the platform and the latter responsible for the SSI enabling the input of text in the chat rooms.

The Emotiv App, with its already pre-set of mental commands that can be quickly trained by the user to fit his brain activity, allow the navigation on the Frontend to be hands-free, as the user can interact with it only by thinking. On the other hand, the SSR App is a self-developed application that uses machine learning for transforming the muscle activity of the user, into text that is suitable for message input in the Frontend. The user will interact

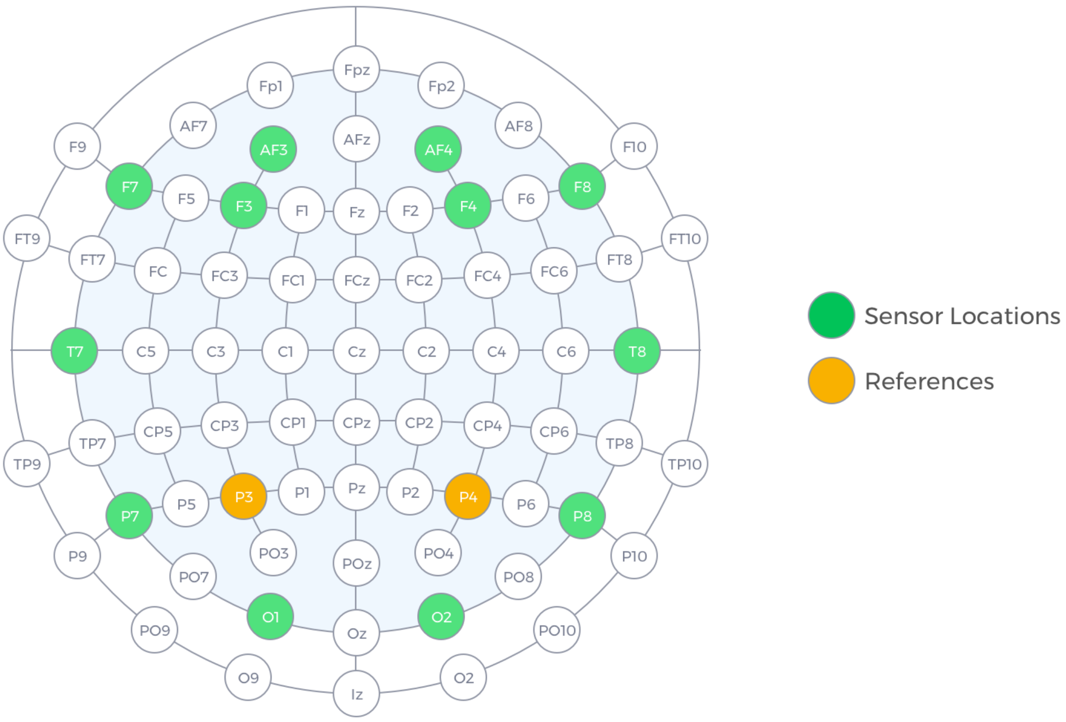


Figure 5.5: EPOC+ electrode locations following the international *10-10 System* (Emotiv 2019).

with these applications trough the Emotiv's EPOC+ and trough electrodes placed above muscles related to speech, respectively.

The Frontend app development is also a focus of this document, as it connects the user inputs into one visual platform. This application not only provides interfaces for user interaction but nevertheless it is also deeply connected to the decentralized ecosystem through its relationship with the backend counterpart. The components relative to the backend component are denominated in Backend and DHT, and are not further analyzed in this document, being related to the work of the other team member.

5.5 Structure of the frontend application and its interfaces

In the previous section, it was showcased the relationships between components. Therefore, this one focuses on the components more present and related to this thesis subsystem and further explains how these are connected. Moreover, these details must be present during the development phase, susceptible to changes. As previously described, the frontend application has two main relationships for it to be usable by hand/arm impaired people, connecting to the Emotiv Cortex and the SSR app.

5.5.1 Emotiv App and Frontend relationship

Starting with the Emotiv App and Frontend relationship, the connection is established using WebSocket Secure [1] protocol, with the first acting as the server and the latter as the client,

[1]WebSocket is a communication protocol allowing full-duplex channels over a single TCP connection.

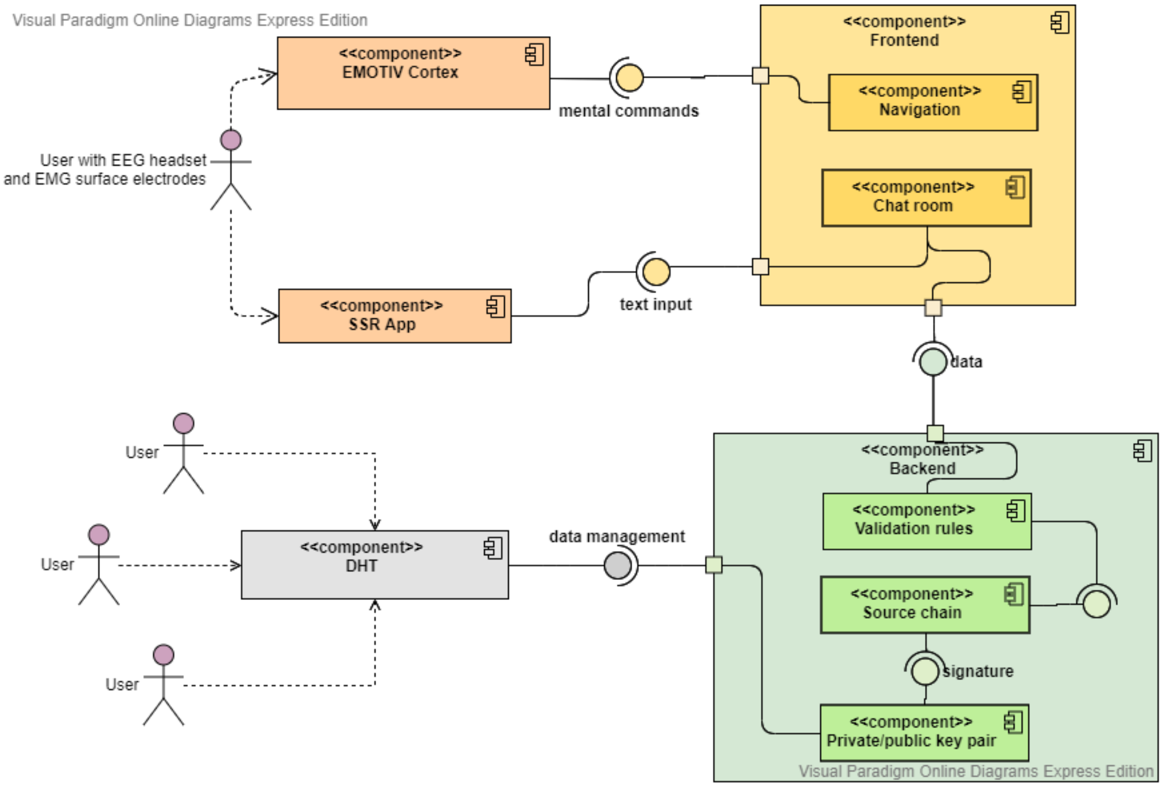


Figure 5.6: High-granularity diagram showcasing how each component/concept interacts with each other across the application. The frontend (top of diagram) and the backend (bottom of diagram) counterparts both reside within the user's device.

using JSON-RPC [2] protocol in the exchanged messages. Emotiv provides an API called Cortex API allowing applications to subscribe to its messages, sharing the data gathered by Emotiv devices with its clients. This way, it is possible to read from the Emotiv App which mental commands it classifies from the data collected with the EEG device. On the other hand, these EEG devices connect to the Emotiv App (running on the user computer) through a Bluetooth connection. The process of connection and subscription to the Cortex API is described in Figure 5.7.

5.5.2 SSR App and Frontend relationship

Analyzing in-depth the relationship between the SSR App and the Frontend, it is expected that the former provides an API which allows the latter to send a recorded EMG signal and it should return the resulted transcript. As the SSR App is self-developed, the API is built as necessary for the text input features. A sequence diagram of the planned interaction is illustrated in Figure 5.8.

This connection is somewhat simple, although being relevant for the integration between the platform components, integrating these different software pieces (related to each interface

[2] JSON-RPC is a stateless, light-weight remote procedure call (RPC) protocol. Primarily this specification defines several data structures and the rules around their processing. It is transport agnostic in that the concepts can be used within the same process, over sockets, over HTTP, or in many various message passing environments. It uses JSON (RFC 4627) as data format.

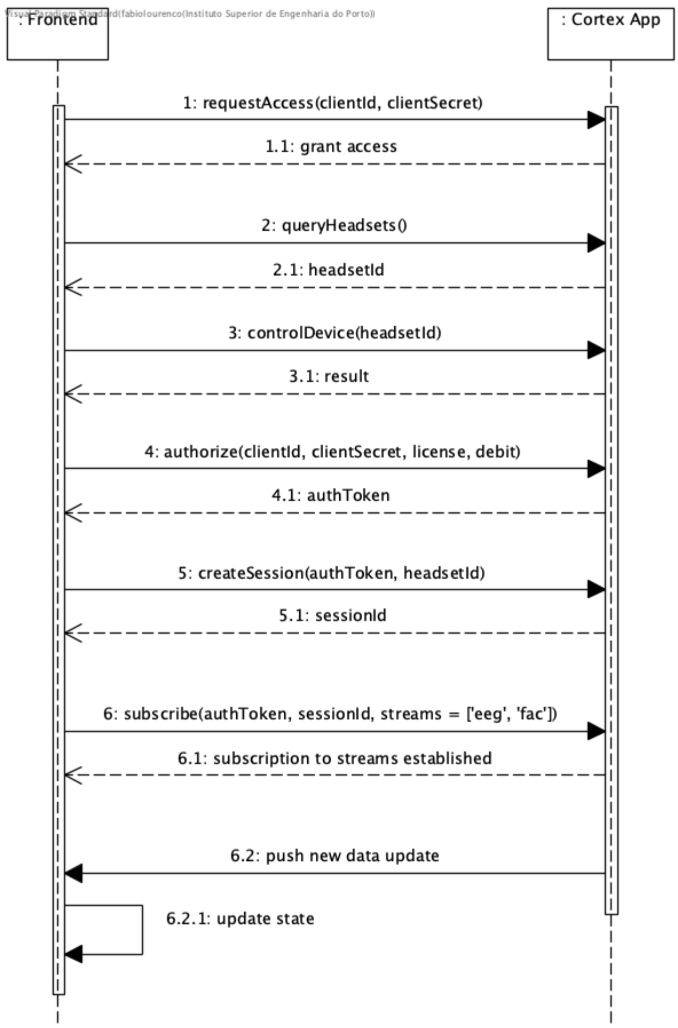


Figure 5.7: Sequence diagram showcasing the main interaction between the platform and the Emotiv Cortex via WebSocket. Multiple steps are required to get access and subscribe to data from the Cortex (EMOTIV 2020). After this initial connection, the Cortex constantly updates the Frontend with current data read from the headset for the streams requests (in this case, *eeg* and *fac* named streams).

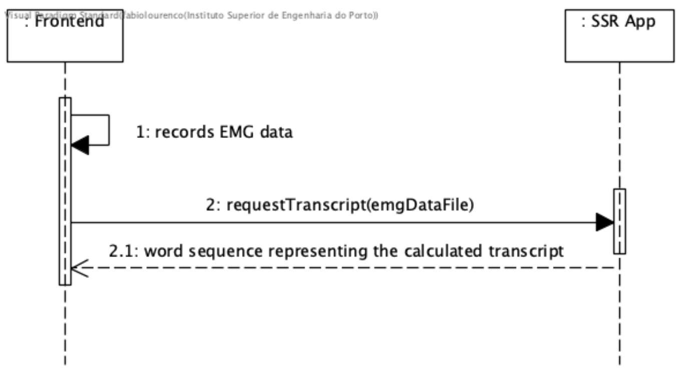


Figure 5.8: Sequence diagram showcasing the planned base interaction between the Frontend and SSR apps.

device) to the main Frontend app that the user interacts with.

5.6 SSR System Development Process

To better develop a concise system, a standardized process will be followed for improved planning and description of the development methods. The studied methodology that will be used is CRoss Industry Standard Process for Data Mining (CRISP-DM), which provides a framework to develop data mining projects, making it more reliable and efficiently repeated for different people (Wirth and Hipp 2000). The CRISP-DM process model breaks down a life cycle of a project into six distinct phases, as illustrated in Figure 5.9. The sequence of phases can vary, depending on the outcome of each phase, but there is depicted the most important and frequent course.

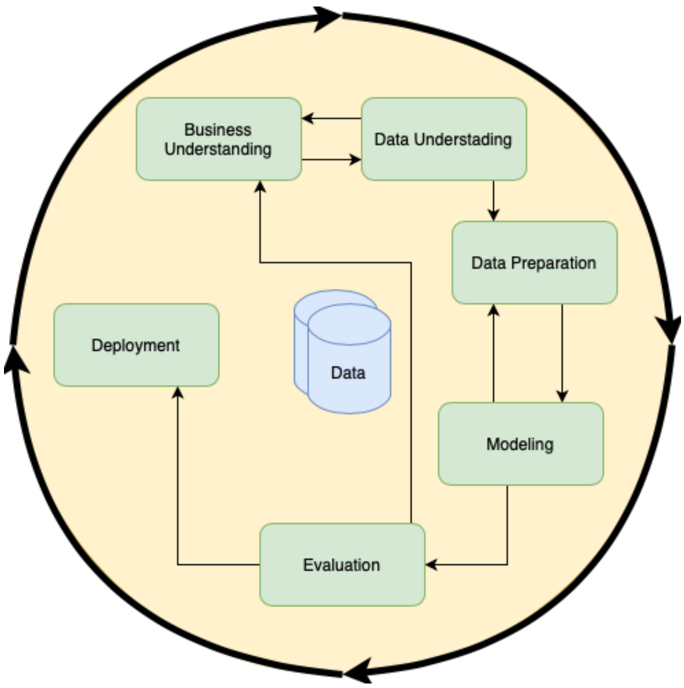


Figure 5.9: Diagram representing the six phases of the CRISP-DM process model and its correlations.

Briefly describing each phase, the CRISP-DM methodology encompasses:

- Business Understanding: This phase focus is to understand the business requirements and converting it to a data mining problem, defining and planning a preliminary solution to accomplish the objectives.
- Data Understanding: In this phase starts the data collection and attempt to discover the first insights of the data or related subsets. This phase is highly close to the previous one since defining the data mining problem and designing the initial plan of the project requires some understanding of the accessible data.
- Data Preparation: In this phase is expected to transform the initial data to construct a concise data set that is likely to feed a modeling tool. This transformation can include operations such as data cleaning, data transformation, or discovery of new attributes.

- Modeling: This phase consists of experimenting with different model techniques and calibrating parameters. This phase has a close link to the previous one since some model techniques required a specific data format, or because while modeling it is found problems or opportunities with the data, which require going back to transform the data.
- Evaluation: This phase is responsible for the evaluation of the models built in the previous one, verifying if the business objectives are sufficiently responded to, before advancing to the last phase.
- Deployment: This phase consists of organizing the knowledge acquired or deploy software that can be used by the customer. Thus, this phase can range from a simple report showcasing how the business goals were met, to a complex implementation of a data mining process across the enterprise.

Following this methodology, the development of the SSR system for this project is more concise and organized, easing the flow of procedures needed and helping with decision making and reporting.

5.6.1 SSR Pipeline Design

Creating a SSI for concise message input required more than just the recognition and classification of single words. A system that also evaluates if the recognized word makes sense in the context or not, may increase the accuracy in some cases and improve the overall performance of the interface, helping to choose and fixing the wrongs that do not fit in the context. Thus, a machine learning pipeline has to be created to chain these processes from data acquisition to word transcription.

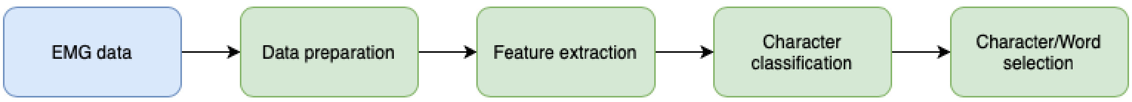


Figure 5.10: Diagram showcasing the designed pipeline for SSR.

As shown in figure 5.10, after the EMG data is acquired, it requires some preprocessing to extract features. After this data is processed, the SSR classifies the data as one or more words. Finally, the last component checks if the chosen words make sense and fix the less correct classifications.

As mentioned in Section 5.2, it will be used the *EMG-UKA* corpus as a base dataset for the implementation and testing of this system. For this, in each run, this dataset will be divided into three different subsets, a training set, a cross-validation set, and a test set.

Possible machine learning alternatives for this implementation are described in Section 3.2.4, being them HMM and LSTM networks, as the state-of-the-art projects for speech recognition in general use mainly these two approaches as a base for their problem resolution.

5.7 Deployment

For a better understanding of where the components will be deployed, Figure 5.11 showcases the planned view for where each should be running and using the aforementioned communication protocols between them.

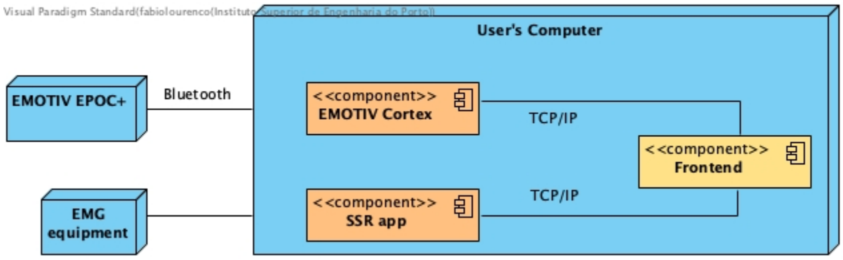


Figure 5.11: Deployment view of a Frontend and its interface apps in a single computer, connecting to the required external devices.

All the software components will run in the user's computer and interact between them as necessary. However, at least three physical devices are needed for the planned hands-free interaction with the system, as it requires electrodes for EEG and EMG and the computer itself.

As an example of an alternative, the SSR app can be running on a remote server and the Frontend would directly receive the EMG data and share it with that server. This solution could have higher delays in response to the user from the system as it would depend on network speed. This alternative would also compromise the decentralized paradigm of the system, as it would have a centralized server receiving all the silent speech data.

Further in the development phase, after the SSR shows promising results with the aforementioned dataset, it is expected that this Frontend app connects to surface electrodes placed on a subject's articulatory muscles, enabling the text input from anyone. Moreover, the SSR must work as user and session independent.

Chapter 6

Development

The development of the platform and interfaces for contactless user interaction is divided into two different interface types, as before in this document. Primarily, this chapter describes the process of development of the Frontend application and integration with components related to the whole platform, such as its Backend that holds and manages the platform decentralized data and the Emotiv App that allows the usage of its EEG headset. Later, it is described the research on the field of SSR and the approaches tested during this project, as well as the experiments executed to test its performance.

6.1 Platform development and Connection with Cortex API

The development of this application did not only had the challenge of programming a messaging platform, but it also encompassed the design of components and visual feedback having in mind it is focused on supporting usage by contactless and movement-free means, intuitively and fluidly.

The key commands and interactions with the UI are described in Section 5.3, although just from a conceptual perspective. During the development, other commands were proven to be needed and added for other features that were not mentioned in the conceptual design, as well as some previously designed ideas were slightly changed to ease the development process but still maintaining its core aspect.

The frontend of the platform is, as stated before, a web application, which allows reaching a wider audience since it runs on any device with a web browser.

Hence, this section will describe the development decisions, highlighting connections between software components, and propose a myriad of new approaches for different UI components to better serve disabled subjects' needs and ease their usage of messaging platforms.

6.1.1 Interaction Modes and Mental Commands

Since Emotiv limits the number of active commands at a time to four (plus the *neutral* command), and to reduce the learning curve of training numerous mental commands, the Frontend application was developed following the notion of interaction modes. Each of these modes interprets the user input differently, depending on the current mode, where the same input can trigger different actions (Page 2019). In other words, the same mental command has different behaviors depending on the current mode, and each mode defines the active commands.

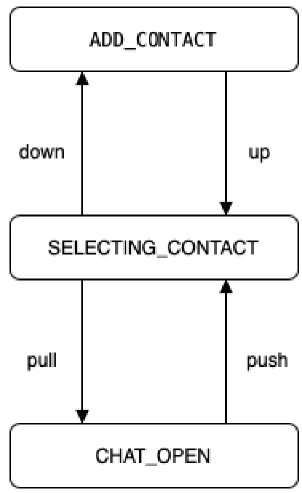


Figure 6.1: Diagram showcasing the available modes and which command triggers a change from one mode to another.

The transition between modes is triggered by the user with a specific mental command, which is different for every mode, as showcased in Figure 6.1. The selection of what command should make a transition in each mode was selected by which command would be the most intuitive, regarding the related UI presented to the user, as will be further discussed.

Selecting Contact Mode

The first mode presented is the *selecting contact* mode, which is related to the first view the user is presented to when it starts using the platform. While on this mode, the user is able to perform a set of features by triggering the respective mental commands, i.e. select a contact (*left*/*right* commands), open a chat (*pull* command), and open the *Add Contact* view (*down* command). As described in Section 5.3, the user is presented with a carousel with all its added contacts allowing it to "scroll" trough it. Once the user founds the desired contact and centers it on the carousel (contact at the center has an outer glow and and an arrow on top of it), the user can *pull* the contact to open a chat window, transitioning to the *chat open* mode. This view is illustrated in Figure 6.2.

Chat Open Mode

The *Chat Open* mode is related to the UI where the user can send a new message to a contact and read all the messages exchanged between them, as shown in Figure 6.3. Similarly to the previously described mode, it is also possible to "scroll", although in this case in a more conventional design, with the purpose of reading older messages. This feature is associated with the once again most intuitive commands related to the design of the UI, i.e. *lift* and *drop*. The focus of this mode is the text input which had to be rethought for hands-free and silent usage. The used command to trigger the functionalities of text input and message forwarding is *right*, encompassing a set of steps that are further detailed in an extensive analysis of the implementation choices for this feature in Section 6.1.5.

Add Contact Mode

Lastly, the *add contact* mode is active when the user "drags" down from the *selecting contact* view and reveals a UI with an input field to add a new contact (see Figure 6.4). As

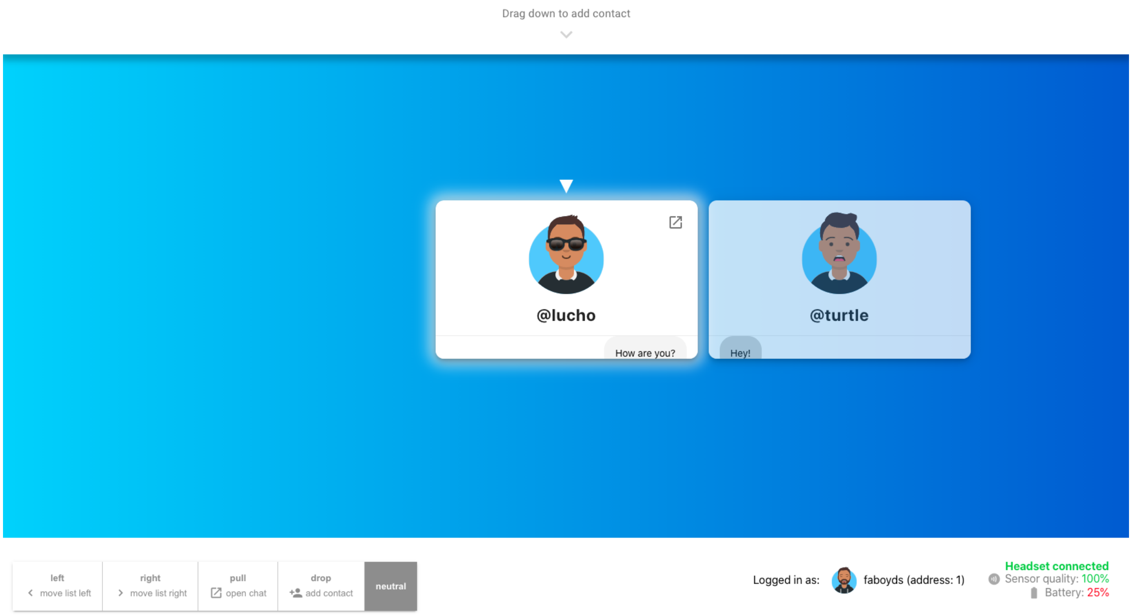


Figure 6.2: Screen shown in *selecting contact* mode. The current selected contact is the one most centered and is highlight with an outer glow and an arrow on top.

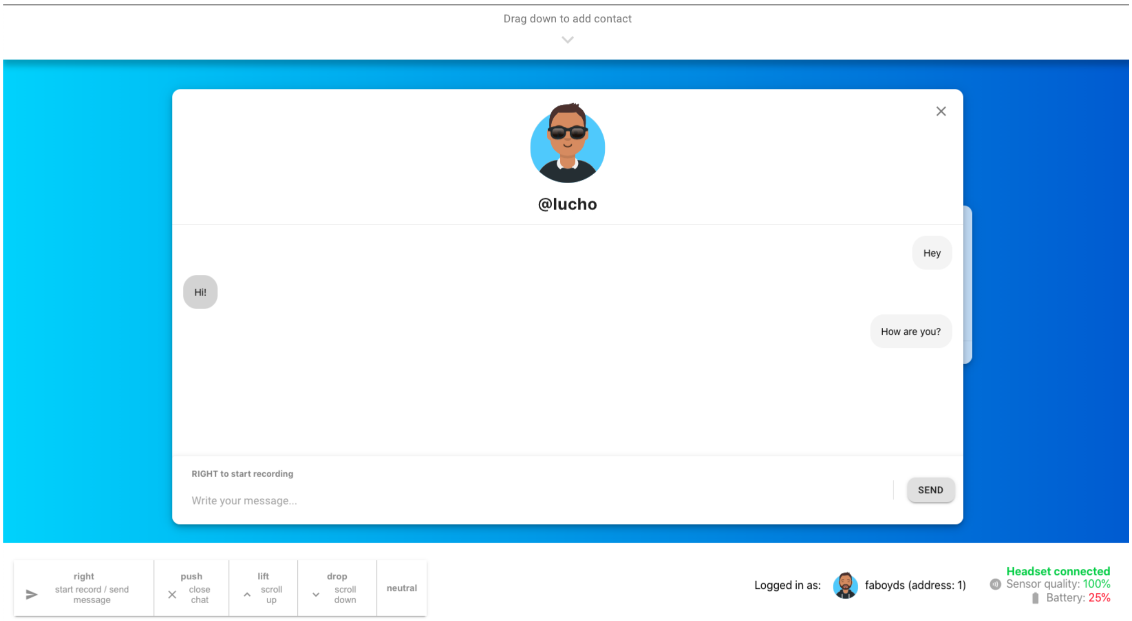


Figure 6.3: Screen shown in *chat open* mode.

in the previous mode, this uses the same command (*right*) to submit the inserted text, in this case, the address of another contact. After this action, the newly added contact will show up on the carousel of *selecting contact* mode UI and the user will be able to exchange messages with them. The only other available command on this mode is *lift* to "drag" this UI back up and change to the *selecting contact* mode.

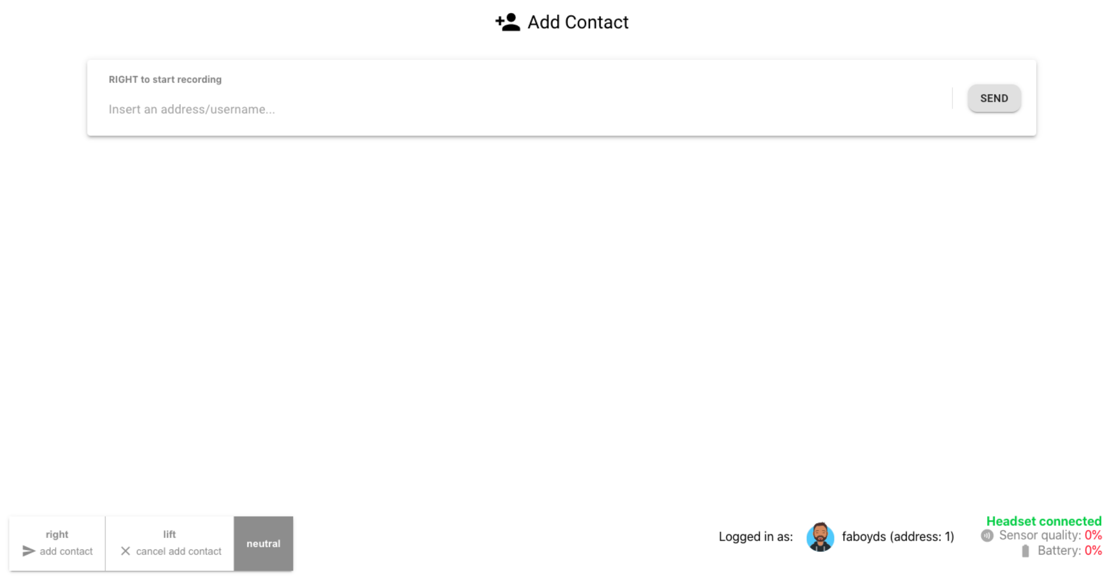


Figure 6.4: Screen shown in *add contact* mode. Although being the more distinct, yet simple, screen, the bottom components keep visible to help the user throughout the whole interaction.

To summarize, Table 6.1 shows the commands available in each mode and possible transitions.

Table 6.1: Table summarizing modes, commands available in each one, and possible mode transitions.

Mode	**Available Commands**	**Transits to**
selecting contact	left, right, pull, down	chat open, add contact
chat open	push, up, down, right	selecting contact
add contact	up, right	selecting contact

6.1.2 Training Area

Before the interaction with our platform, the user must train each mental command required (described in Section 6.1.1), so that the Emotiv software can associate each command to a specific pattern in the EEG signal recorded.

Emotiv has its own software to help with this process, the EmotivBCI app, where the user can create multiple training profiles to test out different approaches when deciding which thought should be related to each mental command. Then, for each profile, the user can activate up to 4 mental commands (plus the *neutral*), and separately train each as many times as desired, until a satisfactory accuracy is achieved. Figure 6.5 showcases the screen that enables the referred actions. When selected a command to train, the user is presented with a training screen (shown in Figure 6.6), where a cube is shown moving/rotating accordingly to the selected command.

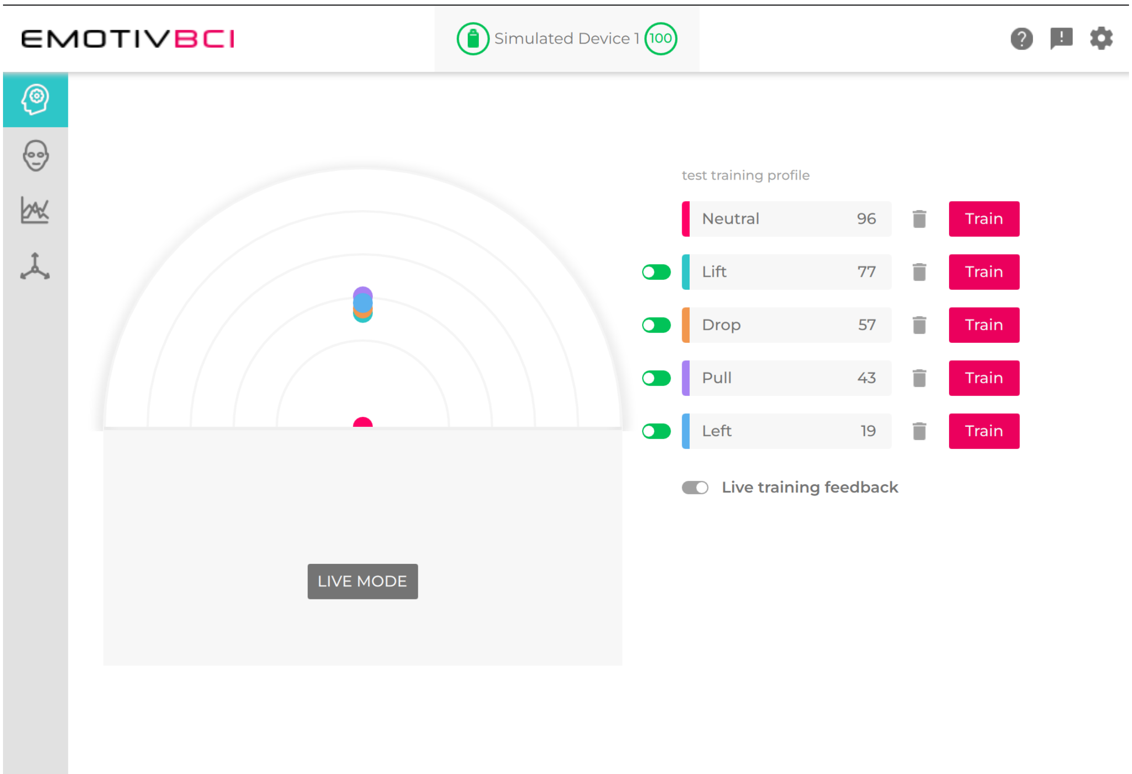


Figure 6.5: Main screen of a profile where the user can selected which commands it wants active, choose which to practice, or enter Live Mode to test the accuracy of the classifications.

Aiming to improve accuracy in interaction with our platform, a training screen was added to the Frontend app, allowing practicing inside the platform. Instead of the default cube displayed in EmotivBCI, a more appropriated component is shown, resembling the contact UI view the user is/will be familiarized with when using the Frontend app. An example of a practice screen is showcased in Figure 6.7. However, as this software is not as mature as the EmotivBCI, this does not yet allow the selection of a mental command to train, forcing the user to practice all the active commands in succession, with intervals of 15 seconds for each 8-second practice.

Even though the user does not have as much control in our solution, when this feature is more developed it is expected that the user obtains better results and performance after utilizing it, since the UI will be similar throughout the whole experience, from training mental command to interaction with the platform.

6.1.3 Buttons for Mental Interaction

The UI components had to be rethought for the interaction through mental commands. Thus, the items that in a conventional app would be expected to be clickable had to change in some way that would feel more natural and with the proper visual feedback. Accordingly, in the proposed solution in this project these components' actions are triggered by a mental command assigned to it (e.g. the component representing a contact the user wants to select

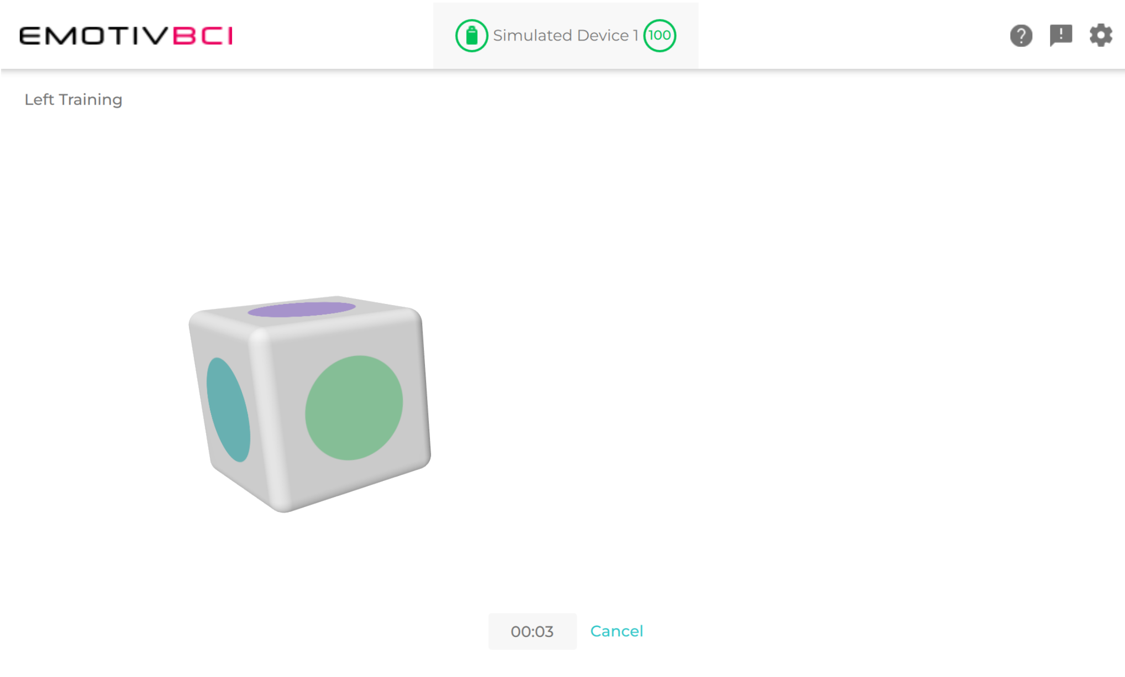


Figure 6.6: Example of a practice exercise (in this case for the *left* command), where the cube moves as expected to help with the training.

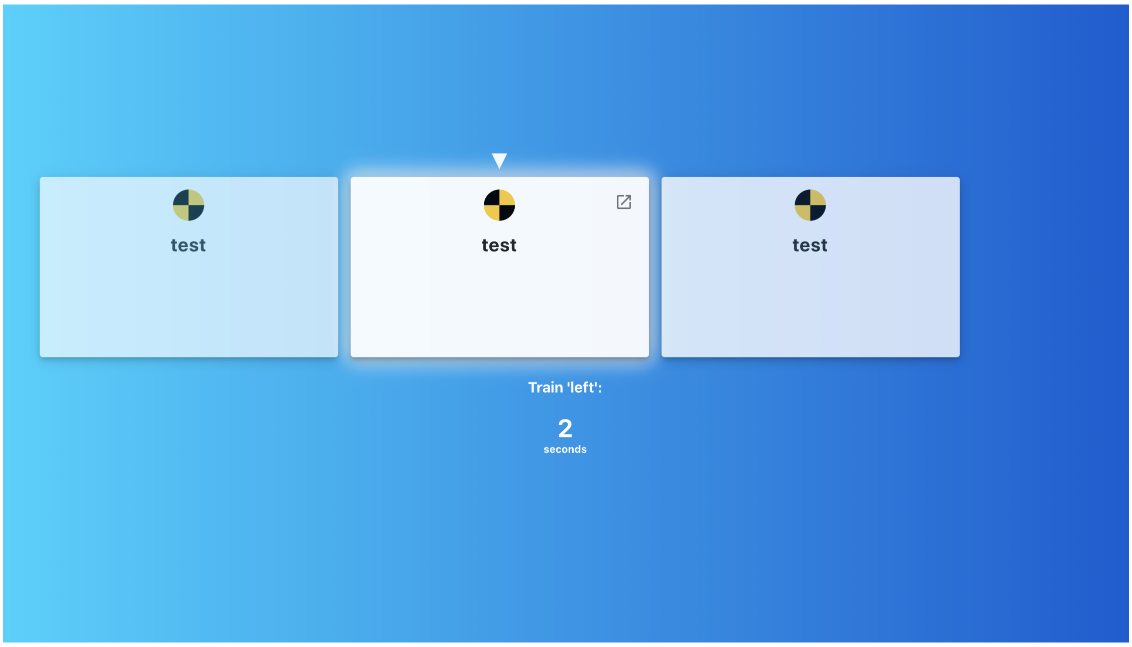


Figure 6.7: Frontend app training screen, currently practicing the *left* command, with a UI similar to the one presented in *select contact* mode, where the *left* command is required.

is triggered by the *pull* command (see Figure 6.8)) and this action is delayed a few sections while the users keep performing the same mental pattern while a progress indicator is shown. Therefore, this avoids triggering actions incorrectly and too quickly, by a miss classification of the data retrieved from the Emotiv headset.

Figure 6.8: Example of a mental button, representing a contact in the carousel. The yellow highlight around this component is the progress indicator. When the progress indicator is all around the view, the action assigned to the button will be triggered. In this example, a chat will open and the application will switch to *chat open* mode.

This type of interaction is more similar to a long press click rather than just a simple click/tap conventionally used on web apps. This technique of long-duration interaction, while receiving progress feedback is also used in buttons of other types of interfaces. Although associated with movement and with haptic feedback, hover buttons have been used in areas such as Kinect-based and Virtual Reality (VR) applications and games, where the user should hold its hand over a button, while an animated progress indicator is completed (Jakob 2010). An example of a hover button is showcased in Figure 6.9, from the Kinect Sports game.

Figure 6.9: Example of an hover button (top right) activated when the hand is over the button until the end of the progress circle (Jakob 2010).

Having this delay gives the user more control by reinforcing that the command received triggered the action the user actually expected, as it must keep being performed during an established period. Besides, this solution also avoids the need for performing a different command to undo the unwanted action. For example, if a user that is in *selecting contact* missuses *pull* command and starts opening a chat, stopping triggering commands and activating the *neutral* command is enough to stop that action, while the mental button is still in progress. Without that feature, after that mistake, the user had to trigger the *push* command to get back to *selecting contact* mode, which would require a heavy mental load.

6.1.4 Command Control Panel

Since the platform uses a myriad of commands, with only some active at a time (depending on the current mode), it may slow down the learning of using these controls. To inform the user which commands are available, a control panel, visible at all times, was added to the bottom left of the screen.

This panel shows all the active commands, and for each one, it has a short description of the action it performs in that case, alongside an illustrative icon to help to perceive the information.

Moreover, this control panel also indicates which mental command the user is performing, highlighting the section related to that command. In Figure 6.10b is shown an example where the active command is *lift*, to scroll up the chat to older messages, while in Figure 6.10a and Figure 6.10c the user is not performing any mental command, therefore the *neutal* option is highlighted.

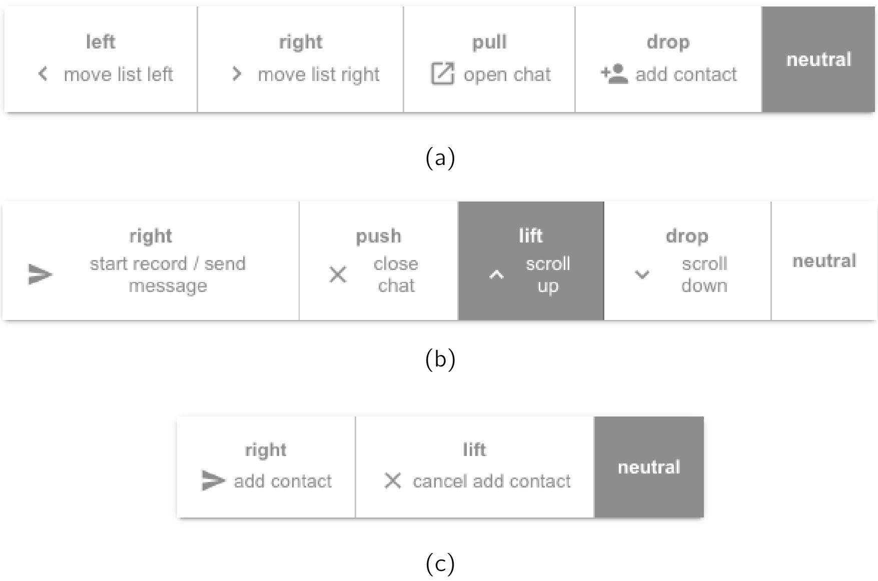


Figure 6.10: (a) Command control panel for the *selecting contact* mode, with the *neutral* command selected. (b) Command control panel for the *chat open* mode, with the *lift* command selected.(c) Command control panel for the *add contact* mode, with the *neutral* command selected.

Finally, each control panel section is also clickable, triggering the selected command. This feature is used to simulate mental commands if the headset is not connected, in order to test how the UI reacts to command selection and quick changes.

6.1.5 Text Input

Regarding text input into the platform, the designed project planned the usage of a SSR system that would convert EMG data from articulatory muscles and convert it into words (see Section5.3). As will be further detailed in Section 7.3.2, the developed system did not meet sufficient accuracy with the corpus used, making it not stable enough for integration with the Frontend application. Therefore, a temporary solution had to be found to allow a user to input text without the need to perform hand/arm movements. Thus, going back to the research described in Section 3.2 about different possible approaches and the conclusions about its comparisons in Section 4.7, a STT system was integrated into the application to accomplish hands-free text input. This approach is not for a permanent alternative solution, as it does not meet the privacy requirements previously established. However, it is a temporary solution to achieve a hands-free interface for this feature.

With either method, the Frontend application is integrated with the SSR application. The latter is prepared for receiving both EMG and audio data, translating these into words as planned using our machine learning models, as will be further discussed. Although in this case, the SSR app will not perform SSR, but instead ASR, using a similar model trained also using the *EMG-UKA* corpus.

Furthermore, abstracting from what type of speech recognition is happening on the background and focusing on UI component and feedback to the user, the input component has some unusual aspects worth mentioning. This component has an add-on to the conventional text input allowing the user to start the speech recognition feature with a mental command, in this case, the *right* command is the chosen one to interact with the text input component. The choice of this command is not as intuitive as the others, because it does not correspond to a visual movement (e.g. *left* and *right* to move the contacts carousel). It was chosen by excluding the other options that did not match in any means the visual feedback, while avoiding forcing the user to train another mental command, keeping in mind the list of possible actions designed by Emotiv (e.g. *push*, *pull*, *lift*, *drop*, *left*, *right*, and rotation related actions). Therefore, excluding the rotation related actions, since *lift*, *drop* and *push* were already being used in this mode, the already trained commands were only *pull*, *left* and *right*. As *pull* was used in the transition to this mode (from *selecting contact*) and could trigger underised actions, only *left* and *right* were remaining as valid options. From those two, *right* was chosen.

Although the interaction with this component through mental commands is always executed with the *right*, the process is divided into small steps. First, the user must trigger the command to make the component start recording speech. Once no speech is detected, the component connects to the SSR App, sending the speech data and retrieving the transcription calculated by the speech recognition system in place. Then, the input component is filled with this transcription and it is given the user a total of 10 seconds to send the message (with *right* command) otherwise the text is cleared. That timeout was defined to allow a user to delete text if the speech recognizer had a poor performance, without the need of using a mental command to do so, simplifying its usage and avoiding even more mental load. This process is illustrated in Figure 6.11 in which is perceivable the states and triggers.

During all these steps, the UI provides the user with the feedback of the process showing a small text on top of the input component. For each step, the label changes and presents information about what is happening in the background, or of what is expected from the user to do. The four different cases are illustrated in Figure 6.12.

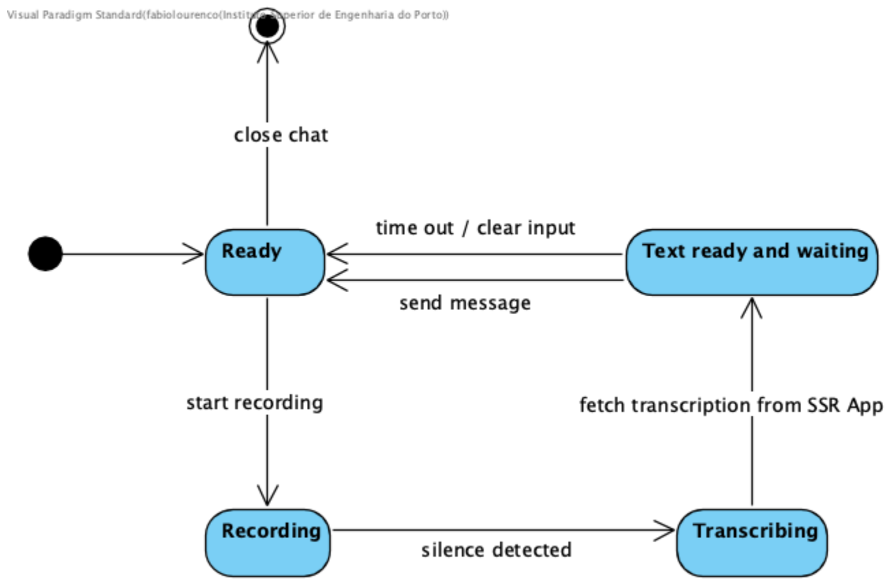
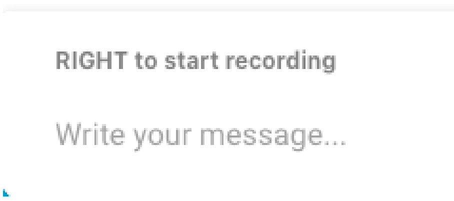
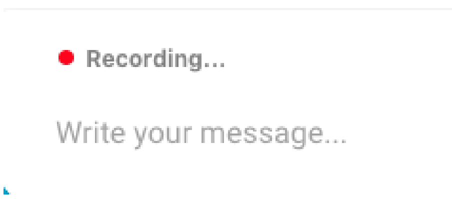


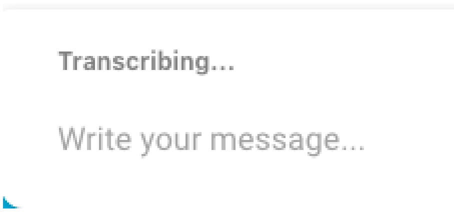
Figure 6.11: State diagram describing the process of interaction with the text input component to perform speech recognition, triggering actions with mental commands.



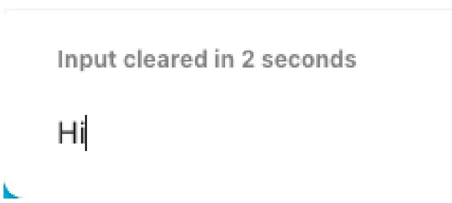
(a) Ready state.



(b) Recording state.



(c) Transcribing state.



(d) Text ready state.

Figure 6.12: Figure illustrating the four different steps of the interaction with the input component to trigger the speech recognition process.

6.1.6 Integration with Cortex API

To trigger the actions and state changes mentioned above, the EEG data has to be retrieved to the application. For this purpose, an EPOC+ was used as the EEG headset during the development phase. Any headset from Emotiv that can classify mental commands through its EEG readings can be used to interact with our platform since the connection is done through Emotiv's Cortex API, which abstracts each headset specific sensor manipulation and allows to fetch only the data streams that are required, as will be further explained. Connecting to the EEG data recorded from the headset requires the use of its manufacturer API, Cortex API, provided by the Emotiv App that must be installed on the device running the platform. This API allows a client to subscribe to multiple data streams for different purposes (e.g. *eeg* for raw EEG data, *com* for results of the mental commands detection, *fac* for results of the detection of the facial expression, *dev* to fetch information about the headset state). For this project, only the *com* and *dev* streams were used.

The data retrieved is then used to change the state of the application and interact with it through the mental commands received, as explained in Section 6.1.1. A snippet of code developed of the higher level of handling data received from the headset is showcased in Listing 6.1.

```
init_cortex.start((e, data) => {
                // command[0] corresponds to the command label
                // command[1] corresponds to the probability
                const command = JSON.parse(data)['com'];
                const deviceInfo = JSON.parse(data)['dev'];

                if (command && command[1] > 0.5) {
                    dispatch(actions.app.changeMentalCommand(command[0])
    );
                } else {
                 dispatch(actions.app.changeMentalCommand(allCommands.
    neutral));
                }

                if (deviceInfo) {
                    setHeadsetBattery(deviceInfo[0]);
                    setHeadsetSensorQuality(deviceInfo[2].pop())
                }
            }
        );
```

Listing 6.1: Code snippet of the handling of data retrieved from the EEG headset changing the active mental command on the application or updating information relative to the headset battery and sensor quality.

Moreover, this connection with the headset allows retrieving data about the hardware, such as battery and EEG sensors positioning quality. To give the user a better experience using the platform and centering information of the whole system, a small UI component (illustrated in Figure 6.13) was added to the bottom of the screen showing some details about this connection. This provides the user with useful information and quick information without the need for using the external Emotiv App to understand which connection problem might be occurring.

Figure 6.13: UI view showing information about the headset hardware.

6.1.7 Integration with Decentralized Backend

Beyond the HCI and the proposal of new approaches to this topic and adaptation of existing concepts to its usage with BCI, this project focus on the private exchange of messages between peers. Furthermore, the platform has to have a backend that handles these transactions securely and privately, extending the same principles used to design some frontend features, such as SSR to keep the leaks of communications to a minimum.

As planned in Chapter 5, this project encompasses a decentralized backend beside the Frontend application and its interfaces, enabling the integration of both to create a system capable of maintaining the assumptions of privacy throughout the whole platform. Although this report does not discuss the development of the backend counterpart, it does address the communication between the two and how this integration is accomplished.

Moreover, the backend application provides an API by using JSON-RPC over a WebSocket connection (similar to what is described in Section 5.5.1 for the communication between the Frontend and the Cortex API). The Frontend establishes this connection as a client to fetch and send information to the backend to handle the data.

From a developing standpoint, the calls to the backend are abstracted by using Apollo GraphQL, in order to promote modularity and decoupling of the software components. To define the data transferred and the functions available, GraphQL uses schemas with three types, i.e. Object, Mutation, and Query. Beginning with the Object types, each type defines a collection of fields, from a data object. Each of these fields has a type of its own, which can be either an Object type or a scalar type (ID, String, Boolean, or Int). Listing 6.2 shows the Objects defined for this project.

```
type Address {
    address: String
}

type Message {
  from: String
  to: String
  createdAt: String
  message: String
}

type Contact {
  address: String
}

type Friend {
  agent: String
  public_key: [Int]
}

type Profile {
```

```
  username: String
  avatar_url: String
}
```

Listing 6.2: Object types created to define the data objects exchanged between the Frontend and backend applications.

Beginning with the Object types, each type defines a collection of fields, from a data object. Each of these fields has a type of its own, which can be either an Object type or a scalar type (ID, String, Boolean, or Int). Listing 6.2 shows the Objects defined for this project. Focusing on functions definition, one can define Queries used to *fetch* data or Mutations used to *modify* data. As shown in Listing 6.3, there was created a myriad of Queries, used to fetch data about users, contacts (referred to as friends), and messages, as well as Mutations, used to add a contact or to send new messages.

```
type Query {
  getOwnAddress: Address
  getMessage(id: String): Message
  listMessages: [Message]
  listMessagesFromContact(address: String!): [Message]
  listContacts(ownAddress: String!): [Contact]
  getOwnProfile: Profile
  getProfile(address: String!): Profile
  listFriends: [Friend]
}

type Mutation {
  sendMessage(messageInput: MessageInput): Message
  registerProfile(username: String, avatar_url: String): Profile
  addFriend(address: String!): Address
}
```

Listing 6.3: Queries and Mutations created to fetch and modify data respectively. These functions receive and/or return data in the form of the Objects specified.

Once GraphQL schema is defined, a collection of resolver functions must be created to handle requests for the Queries and Mutations previously mentioned. This is the code component that does, in fact, perform the calls to the backend services, receiving the parameters and returning data as is established by the created schema. Examples of these resolver functions are showcased in Listing 6.4.

```
export const resolvers = {
  Query: {
    // ...

    listMessagesFromContact: async (_, { address }) =>
      (await createZomeCall('/notes/chat/list_messages_sent_to')({
    address})).map(dnaToUiMessage).sort(function (a, b) {return new Date(
    a.createdAt) - new Date(b.createdAt)}),

    listFriends: async () =>
        (await createZomeCall('/notes/chat/fetch_all_friends')()),

    // ...
  },

```

```

  Mutation: {
    // ...

    sendMessage: async (_, { messageInput }) =>
        dnaToUiMessage(await createZomeCall('/notes/chat/send_message')
    ({message: messageInput.message, to_agent: messageInput.to})),

    addFriend: async (_, { address }) =>
        (await createZomeCall('/notes/chat/add_friend')({address})),

    // ...
  }
};
```

Listing 6.4: Examples of resolver functions used to handle requests to the backend application following the GraphQL schema.

With this implementation, requests to the backend are structured and used throughout the Frontend application when needed, to make each user's client app connected to the whole platform enabling private message exchange.

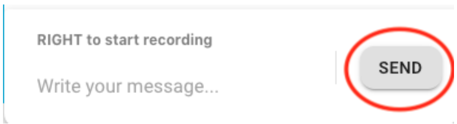
6.1.8 Conventional Interactions

Since this project aims to join people with or without hand/arm disabilities, our platform should likewise be suitable for use with conventional interfaces. Therefore, all the UI components and interactions are also prepared for common mouse clicks and text input with the keyboard, allowing for subjects without the aimed disabilities to interact with the platform using interfaces that they are already familiar with. These alternatives are not only useful for subjects without impairments but can also be useful for impaired subjects that are already familiar with these conventional interfaces or other alternatives suitable for them (some of these alternatives are described in Section 3.1.1).

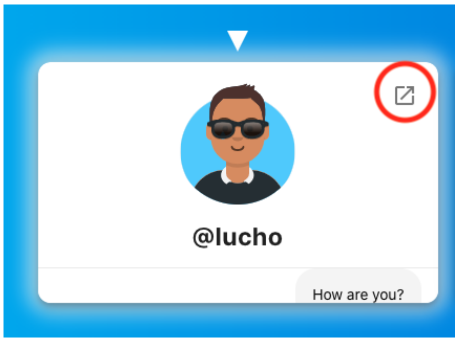
With this redundancy of different possible interactions for the same action, any user can easily interact with the platform while they are still getting used to the new proposed approach.

Furthermore, this alternative was implemented by enabling mouse interaction with UI components. An example of these changes is found in the contacts carousel, in which a "horizontal swipe" was enabled for scrolling through contacts with a "click and drag" using a mouse, instead of the previously mentioned approach of *left* and *right* mental commands. Other examples are some non-intrusive or distracting extra buttons, showcased in Figure 6.14.

In conclusion, besides all the connections between the different components of the system and HMI interfaces provided to the user in order to build a concise messaging system, the Frontend component developed in this project proposes multiple new perspectives for common UI views aiming for a more suitable usage with BCI interfaces. These will help to learn how to handle the application trough these unorthodox methods, being more intuitive, while still having an option to manage it with conventional input devices. These aspects should also contribute to HMI state of the art, seeking to satisfy a more fluid stream of interactions and feedback to the user when utilizing BCI technologies.



(a) Button to send message is both a mental button and a normal clickable button.



(b) Button added on contact view to allow opening a chat by using a mouse instead of using the *pull* mental command.

Figure 6.14: Examples of buttons used for conventional interactions with the platform.

6.2 Silent Speech Recognition Study

The development of a SSR system demanded the investigation of different approaches to creating the best possible model with the data available.

For this project, as previously mentioned, it was used the publicly available trial subset of the *EMG-UKA* corpus, which is a reasonably small dataset, however, it was the only one found with public access that fit the needs of this project.

In this section, the details of the experiments executed during the development of this project will be explained, as well as the different approaches that were attempted.

6.2.1 Data Set

ASR research and products have been emerging in the last years, having numerous different publicly available datasets and many approaches obtaining good results, allowing for public usage of these solutions in our daily lives. However, for the problem of SSR, even though it has a similar objective of recognizing speech trough data waves, the quantity of data and research is quite limited.

This section presents further the publicly available dataset that was used in this project, the *EMG-UKA* corpus, and its characteristics (Wand, Janke, and Schultz 2014). The *EMG-UKA* corpus was developed at Karlsruhe Institute of Technology, comprises both EMG and acoustic recordings with the objects of researching speech recognition based on EMG signals. Despite the full corpus having 7:40 hours of data by 8 speakers, the distribution available for this project was only the *EMG-UKA* Trial corpus, which contains a subset of recording from 4 speakers with a duration of 1:52 hours of data. The recordings were performed in three different mode, audible, whispered, and silent. It is relevant to have that type of variety in the recordings since the intensity of muscle activity differs from these modes of speech.

Data Collection

The *EMG-UKA* corpus EMG data was collected using a six-channel electrode setup, using standard gelled Ag/AgCl surface electrodes with a circular recording area of 4 mm. The electrodes were positioned in a way that could capture EMG signal from six different articulatory muscles: the levator anguli oris (channels 2, 3), the zygomaticus major (channels 2, 3), the platysma (channels 4, 5) the depressor anguli oris (channel 5), the anterior belly of the digastric (channel 1) and the tongue (channels 1, 6), as showcased in Figure 6.15.

At the same time, acoustic data was also collected using a standard close-talking microphone, with a 16 kHz sampling rate. This data is in stereo format, having the first channel for the audio signal, and the second channel is used for containing *markers*, used in synchronization.

For the synchronization of the EMG and acoustic data, a seventh channel was added to the EMG data where the marker signal appears as a binary signal. In the audio data, an analog signal was used in the second channel. This marker is found in the peak of those channels and is used to find the same point in time in the two types of data, allowing synchronization.

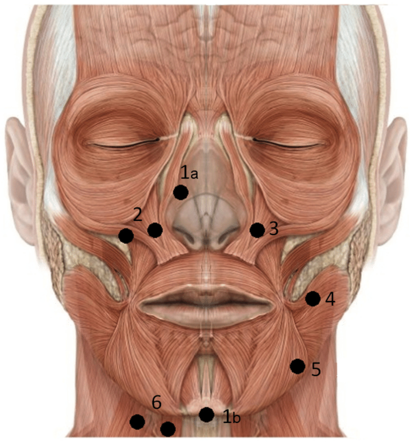


Figure 6.15: Position of the electrodes used for the *EMG-UKA* corpus, numbered by channels (chart adapted from Michael 2005).

Corpus Overview

The used corpus contains recordings from four different speakers, ranged with ages between 24 and 30 years at the time of the recording. The subjects were not native speakers of English, however, there was an effort to make sure the sentences were correctly pronounced. The data was recorded in different sessions with data from all three speaking modes and having, for example, eight sessions all by one speaker, and a large session, allowing multi-session (with the same speaker) and single-session experiments with enough training data. The *EMG-UKA* data was recorded in three speaking modes, audible, whispered, and silent (performing the normal articulatory movements while suppressing the glottal airstream). In the silent mode, although some phones are slightly audible, the utterance is not understandable, which distinguishes the silent and whispered speech. Table 6.2 presents an overview of how the data is distributed in different subsets and the duration of the recordings.

The data comes already subdivided into training and test data by default, since the content of training data varies between session, whereas the test data always consists of the same 10 sentences. The division was designed to allow training the model with enough vocabulary for the recognition.

Table 6.2: Duration of data subsets in the *EMG-UKA* Trial corpus ([h]:mm:ss). For the small subsets, each session includes 50 utterances, while for the large subset contains 520 utterances.

EMG-UKA Trial Corpus				
Subset	# Speakers	# Session	Average Duration	Total Duration
Audible (small)	4	12	3:19	39:47
Whispered (small)	4	6	3:38	21:47
Silent (small)	4	6	3:44	22:21
Audible (large)	1	1	28:29	28:29
All data	1:52:24			

6.2.2 Deep Learning Approach

Currently, several approaches for ASR are being developed recurring to deep learning models (Yu and Deng 2016). Thus, being a similar problem of classification of a continuous signal, SSR can be solved also using deep learning approaches (Rosello, Toman, and Agarwala 2017). In this thesis, as will be detailed further, the solution experimented is based on Deep Speech 2, an end-to-end deep learning system (Amodei et al. 2015). Architectures based on Deep Speech 2 consist of one or more convolutional layers, followed by one or more recurrent layers, followed by one or more fully connected layers, depending on the problem.

Model Architecture

For this project, the architectures of the system tested have two convolutional layers, followed by three or five recurrent neural-network layers with LSTM cells with a size of 512 or 1024 for all three hidden layers, followed by one fully connected layer. The different tested models are outlined in Section 7.3.2. It was used Connectionist Temporal Classification (CTC) loss function, and the optimizer chosen was Adam with a learning rate of 1e-3. A diagram illustrating an architecture of the system is found in Figure 6.16.

From a training set $X = \{(x^{(1)}, y^{(1)}, (x^{(2)}), y^{(2)}), ..., (x^{(\mathrm{m})}, y^{(\mathrm{m})})\}, x^{(\mathrm{i})}$ is a single utterance and $y^{(\mathrm{i})}$ is its transcript. Each utterance, $x^{(\mathrm{i})}$, has 7 time-series of length $T^{(\mathrm{i})}$, one corresponding to each EMG data channel (illustrated in Figure 6.17).

The output of the network is a sequence of characters, each being an alphabet letter, space, or apostrophe. The network objective is to transcribe an input sequence $\mathrm{x}^{(\mathrm{i})}$ into a final $\mathrm{y}^{(\mathrm{i})}$.

Feature Extraction

For each EMG example having 6 different channels of raw data (the seventh channel is only for synchronization, as explained in Section 6.2.1, thus only no real value for model training), features need to be extracted. Figure 6.17 showcases a plot showing the original EMG of an example utterance, with all 7 channels.

Thus, each utterance $x^{(i)}$ being a group of 6 time-series of length $T^{(i)}$, for each of these time-series, $x_j^{(i)}$, is applied a Short-time Fourier Transform (STFT). The STFT represents a signal in the time-frequency domain by computing Discrete Fourier Transform (DFT) over short overlapping windows, defined by

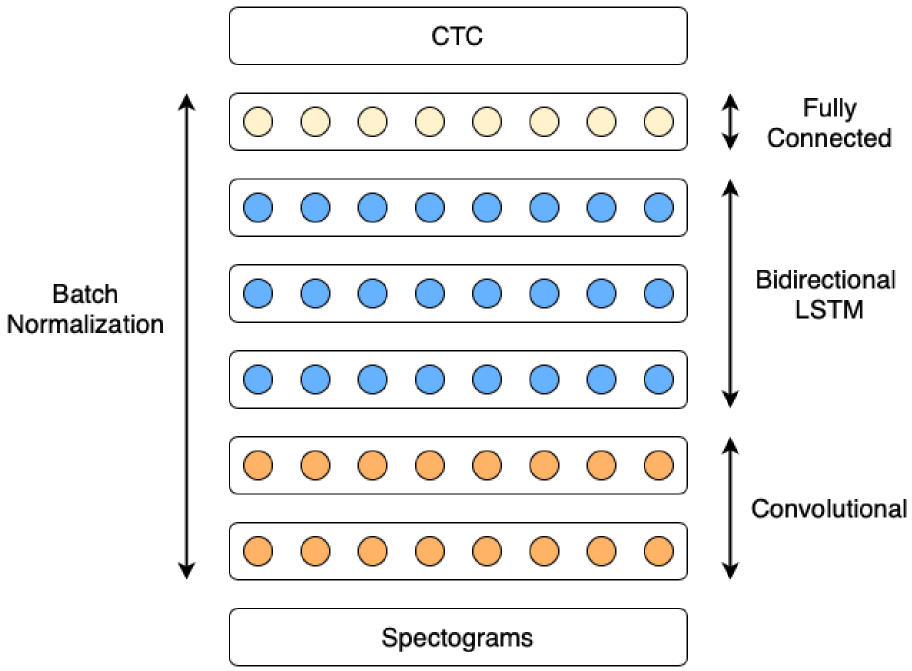


Figure 6.16: Architecture of the system implemented for SSR, which consists of two convolutional layers, followed by three bidirectional recurrent layers using LSTM, followed by one fully connected layer.

$$X_m(\omega) = \sum_{n=-\infty}^{\infty} x_j^{(i)}(n) w(n - mR) e^{-j\omega n} \tag{6.1}$$

where

$$\begin{aligned} x_j^{(i)}(n) &= \text{input signal at time } n\text{, in channel } j \text{ from example } i \\ w(n) &= \text{length } M \text{ window function } (\textit{e.g.}\text{, Hamming}) \\ X_m(\omega) &= \text{DFT of windowed data centered about time } mR \\ R &= \text{hop size, in samples, between successive DFTs} \end{aligned}$$

as explained in (Smith 2011). The time series were split into windows with sizes of 27 ms and with hops of 10 ms, overlapping around 37%.

Then, a spectrogram is used to represent the STFT as features, as illustrated in Figure 6.18.

Feature Scaling

Feature scaling is relevant to change values to a common scale without interfering in different ranges. The method used in this project is standardization, which means that for each example feature vector is converted into a standard score (z-score) used a standardization function,

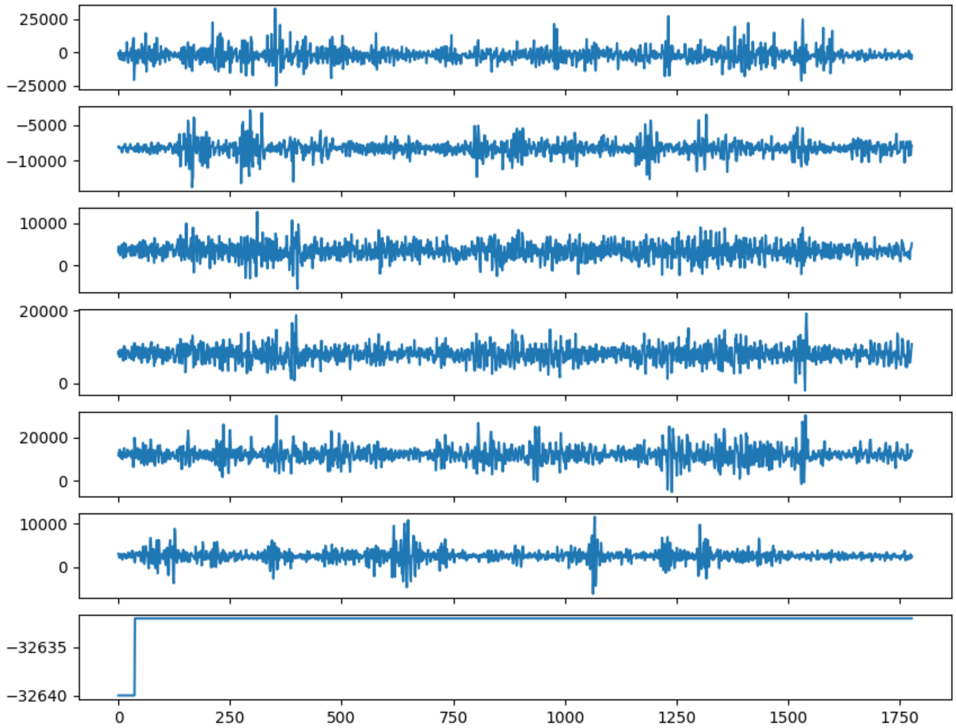


Figure 6.17: Plot showing EMG data from an example utterance. Each subplot represents a channel.

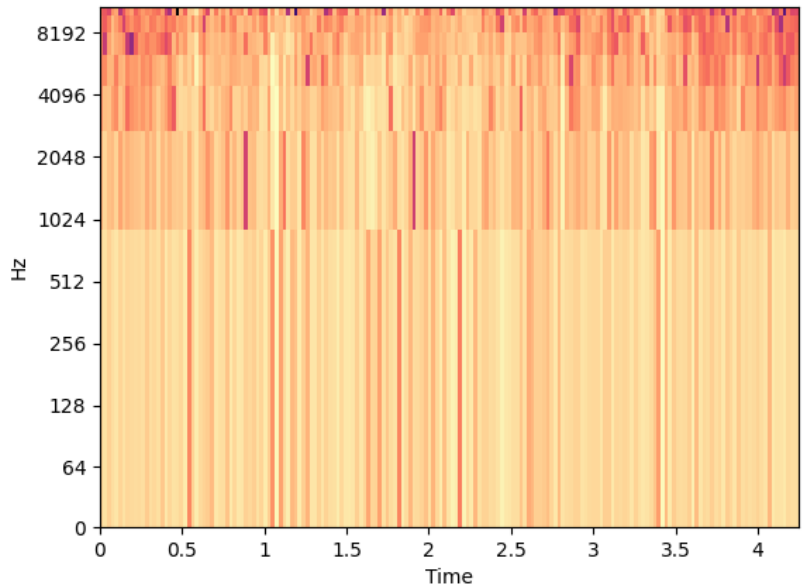


Figure 6.18: Example of a spectrogram that resulted from feature extraction of a channel's time series.

$$z_i = \frac{x_i - \bar{x}}{\sigma} \tag{6.2}$$

where x_i is the feature vector, $\bar{x}$ is the mean of that feature vector, and σ is its standard deviation. This method makes each feature vector have a zero-mean and unit variance.

In the case of this project, this feature scaling is applied to each spectrogram obtained from the process described in Section 6.2.2.

Decoding Step

The decoding step is the part of the system that, from a trained model, transcribes an input sequence of EMG data into a sequence of words. CTC operation is responsible for this step, where two different algorithms were studied for the decoding, Greedy Search and Beam Search.

Greedy Search algorithm simply selects the highest-likelihood output label at each timestep. By selecting the one best option, it can be accurate for this specific timestep, but not the most suitable to construct a full sentence (Zenkel et al. 2017). That is the problem that Beam Search tries to solve.

Beam Search algorithm selects multiple output label options for each timestep based instead of only one. The number of options selected depends on a parameter called *beam width*. For instance, if beam width is set to 10, the algorithm will store the 10 most likely alternatives. After this selection, the system will use a language model that will calculate the conditional probability of each alternative, considering the options selected in the previous timesteps, therefore forming a more concise sentence (Williams et al. 2018).

Language Model

By including a language model in a Beam Search decoding step, it can improve the model performance by correcting spelling errors and fixing word selection. Briefly, a language model calculates the probability of a sequence of words occurring in a sentence. These models calculate the likelihood of a word occurring depending on *n-1* previous words, thus, being called *n-gram* models (Jurafsky and Martin 2009).

In an n-gram model, the probability $P(w_1, \ldots, w_m)$ of observing the sentence $w_1, \ldots, w_m$ is approximated as

$$P(w_1, \ldots, w_m) = \prod_{i=1}^{m} P(w_i \mid w_1, \ldots, w_{i-1}) \approx \prod_{i=1}^{m} P(w_i \mid w_{i-(n-1)}, \ldots, w_{i-1}) \tag{6.3}$$

N-gram models are defined by a Markov chain, explained in 3.2.4. The probability distribution of the vocabulary, given the sequence of *n-1* words, is estimated based on n-gram counts from natural language data collection.

The language model used on this project is a 3-gram ARPA model trained with LibriSpeech corpus (Panayotov et al. 2015).

6.2.3 SSR Web API

The SSR system on its own is not enough for integration with the platform. As planned in Chapter 5, it had to be built the SSR App, which consists of a server app ready to receive data files, transcribe those file's signal into a sequence of words, and return the results to the client. This software component is simple, yet essential for the integration of the SSR model with the Frontend app. A snippet of the code used is showcased in Listing 6.5.

```
@app.route("/", methods = ['GET', 'POST'])
@cross_origin()
def upload_file():
    if request.method == 'GET':
        return '''
          <!doctype html>
          <h3>Upload new File</h6>
          <form method=post enctype=multipart/form-data>
            <input type=file name=file >
            <input type=submit value=Upload>
          </form>
        '''

    if request.method == 'POST':
        # check if the post request has the file part
        if 'file' not in request.files:
            return '<h1>No file part</h1>'
        file = request.files['file']
        # if user does not select file, browser also
        # submit an empty part without filename
        if file.filename == '':
            return '<h1>No selected file</h1>'
        if file and allowed_file(file.filename):
            filename = secure_filename(file.filename)
            file.save(os.path.join(app.config['UPLOAD_FOLDER'], filename))
            print('File saved with success at ' + os.path.join(app.config['UPLOAD_FOLDER'], filename))
            print('Starting transcription...')

            transcription = subprocess.check_output('python transcribe.py --model-path /models/final.pth --audio-path ' + os.path.join(app.config['UPLOAD_FOLDER'], filename) + ' --decoder beam --lm-path /content/3-gram.pruned.3e-7.arpa --beam-width 128 --lm-workers 16', shell=True)

            r = json.loads(transcription)

            result = r['output'][0]['transcription']
            print('Result: ' + result)

            return result
```

Listing 6.5: Code snippet of the software built to provide the Frontend app an API to get results from the SSR model.

All in all, the development of the SSR system required a profound study of the fields of signal processing and machine learning in general. The experiments conducted with the different architectures aforementioned in Section 6.2.2 are further described in Chapter 7.

Chapter 7

Evaluation of the solution

Once the platform is ready for its first tests, there is a need for evaluation methods and criteria to judge if it fulfills the proposed objectives, and which objectives are less accomplished and its solution must be rethought. Therefore, this chapter outlines the testing methods and guidelines to measure the development of the project.

7.1 Criteria

The criteria that are going to be analyzed for the BCI counterpart are intuitiveness and learning curve related to the use of this interface on our platform. Related to the SSI, the aspect evaluated is the accuracy of the developed system. The methods for evaluating these factors are further described in the following sections.

7.2 Hypothesis

Once the platform is mature enough and has good enough results in a fully controlled environment (metrics and results are described in Sections 7.3.1 and 7.3.2), it will be tested with a group of subjects, preferentially with hand/arm impairments, as they are our target audience. These subjects might be easily found through a possible partnership between our team and a hospital.

Testing the platform with our target audience is an effective method to get proper feedback for future improvements on the system. They are more capable of than any other subject to test if this new approach to messaging fits their needs and how it could help to break barriers in remote communications.

Lastly, after these tests, it will be possible to verify the hypothesis suggested by this project, demonstrating that the use of BCI allied to SSI approaches does help subjects with hand/arm impairments to communicate online via messaging platforms.

7.3 Evaluation methodology

As aforementioned, the tests must be as performed by hand/arm disabled people, in order to get their feedback and effectively evaluate the platform. This selection of the subjects with these motor disabilities is enough to test the BCI component. However, if the SSI component achieves enough accuracy with the *EMG-UKA* corpus and is ready for testing with subjects, it would demand another requirement: the subjects must speak the language that the SSR is prepared for (in our case, English). As better the pronunciation of the words, better should be the results of the SSR. Age is not a factor that is decisive for the experiments however it is important to verify if the solution developed is effective at any age range group. That said, two different tests can be conducted, using the same subjects or not.

7.3.1 BCI Evaluation

BCI evaluation process will consist of usability tests to gather feedback on how the platform improves their communication and overall navigation compared to similar apps. First, the subjects have to train the BCI headset to their mental activity. This task might take some time with inexperienced people, but the most adjusted it is to the subject, the better it will perform in classifying the mental command performed. Then, subjects will be asked to perform a set of actions, navigating in the platform and interacting with it. The set of actions that must be performed are:

- Navigate through the contacts
- Choose one specific contact requested
- Send a message to the selected contact
- Scroll trough messages with the selected contact
- Go back to contact list

Lastly, a questionnaire will be conducted in order to evaluate the platform. This survey will follow the SUS, with 10 items answered in a degree of agreement or disagreement with each statement on typical five-level Likert scale. The template statements for the SUS are:

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.

In order to score the SUS, after each statement is measured by the subjects on a 1 to 5 scale, and for items 1,3,5,7, and 9 (positive statement) the score contribution is the scale position minus 1, and for items 2,4,6,8 and 10 (negative statements) the contribution is 5 minus the scale position. Then, to obtain the overall value of the System Usability, it must be multiplied the sum of the scores by 2.5. The SUS scores range from 0 to 100 (Brooke et al. 1996).

A study on different surveys presented a grading system for the SUS scores (Sauro and Lewis 2016). The ranks and percentile ranges formulated in this study are showcased in Figure 7.1. The score obtained in our test will be later placed in one of these ranks for comparisons.

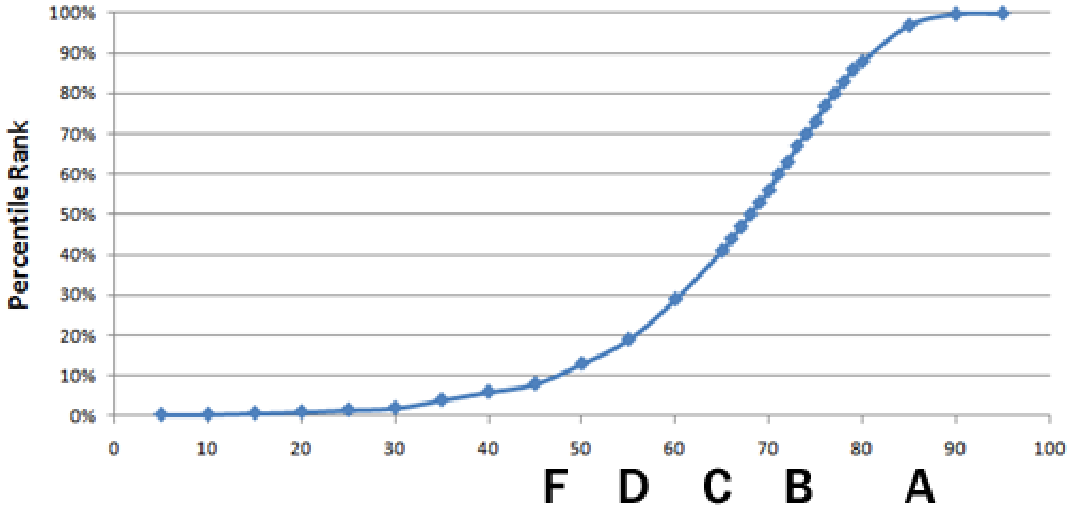


Figure 7.1: Curved grading scaled for interpretation of SUS scores (*Measuring Usability with the System Usability Scale (SUS)* 2011).

The factors that will be analyzed, as stated before, are intuitiveness and learning curve, and each will be evaluated with multiple questions answered also on a typical five-level Likert scale, for each of the actions performed. After the collection of data, exploratory data analysis will be made.

Moreover, if made possible, based on the feedback from this first analysis, it is expected to conduct a second experiment with another sample group, this time with improvements made with the platform. This is to ensure if the applied changes upon the feedback from the first version had a positive impact on user satisfaction or not.

In addition to this, in each test, the Emotiv headset could collect data that would record the emotional state of the user. This is merely an additional step to ensure that the feedback gathered is reliable and not contradictory to what was answered in the questionnaire.

Experiments and Results

The process of evaluation of the BCI counterpart of the platform suffered major impediments and difficulties. Firstly, due to the COVID-19 pandemic and social distancing constraints, contact to test subjects with the aimed disabilities became impracticable, especially when the existing connections were to hospitals and possible associated patients. Secondly, the EEG headset has proven to be hard to train the mental commands for each subject, demanding several hours until a subject dominates only a few commands. This factor further aggravated the first obstacle, since testing with disabled subjects might require numerous training sessions before the user is able to effectively handle mental commands and use the platform.

7.3.2 SSI Evaluation

For SSI, the accuracy will be evaluated in WER. During development, the data sets used for each model created will be divided into smaller sets of training and test sets. These test sets will examine the accuracy of classification by testing different models, trying to find the best one, as other state-of-the-art similar systems already were capable of classifying with an accuracy of around 90% in the best test results (check Section 3.2.2). Later, this feature might also be tested by placing electrodes in test subjects and asking them to speak a set of words, analyzing the WER. Using multiple subjects in multiple sessions would be the best way to test the versatility of the SSR.

WER calculation is based on the *Levenshtein distance* metric, which measures the differences between two *strings*. In this case, the strings are two sequences of words from a transcription.

WER is computed as:

$$WER = \frac{S+D+I}{N} = \frac{S+D+I}{S+D+C} \tag{7.1}$$

where

$$\begin{aligned} S &= \text{number of substitutions,} \\ D &= \text{number of deletions,} \\ I &= \text{number of insertions,} \\ C &= \text{number of correct words,} \\ N &= \text{number of words in the reference (N=S+D+C)} \end{aligned}$$

Explaining further the three concepts referred to in this metric, a substitution occurs when a word is replaced, an insertion happens when a word is added, a deletion is when a word is missing in the transcript.

Therefore, a lower WER indicates better performance and a more accurate model, and a higher WER indicates the opposite. Several models were tested and evaluated using this metric, and its results and comparison are detailed in the next section.

Experiments and Results

Once designed and implemented the SSR system, several experiments must be executed to evaluate the performance of the approach used. The selected set of tests helps to estimate how different speech modes affect the performance of the model, and what problems can be happening with the system. Therefore, these could help improve the recognizer and try to pinpoint possible problems with the model or with a lack of data. Moreover, as the division between the train and test sets used was the same as design by the corpus authors, the results of these tests can be compared with other research projects that followed the same layout. In this section, these experiments and their results will be analyzed and will be discussed the reasons that could affect different values in performance for different sets of data.

As previously mentioned in Chapter 6, several hyperparameters were tested with different subsets of data (detailed in Section 6.2.1), to get a better overview of how each architecture performs with these different sets. These alternatives are outlined in Table 7.1, where each configuration is named in alphabetic order. This naming will be utilized further.

Table 7.1: Table showing the different configurations experimented, varying in number of layers, layer size and decoding algorithm.

Config. Name	**Layers**	**No. Cells**	**CTC Algorithm**
A	3	512	greedy search
B	3	512	beam search
C	3	1024	greedy search
D	3	1024	beam search
E	5	512	greedy search
F	5	512	beam search
G	5	1024	greedy search
H	5	1024	beam search

As shown in Table 7.2 the results are pretty unsatisfactory, with the best models getting results of around 94 WER, which is far from what is required in order to properly use this system. Considering the results obtained had an WER of more than 90, it does not show many differences between the models. Even with beam search, as the transcriptions were too inaccurate, in some cases this algorithm decreases the performance of the model, which is not expected. Table 7.4 showcases examples of results transcriptions, where only a few characters were correctly decoded.

Table 7.2: Table showcasing WERs obtained from each different configuration for 4 subsets of EMG data (recorded in audible, whispered and silent speech modes, and the last with all the data).

Configuration	**Subset**			
	Audible	**Whispered**	**Silent**	**All data**
A	**94.877**	94.733	94.781	96.309
B	95.022	95.238	95.791	94.561
C	97.908	**94.084**	95.455	**94.289**
D	95.238	97.403	95.623	94.872
E	95.238	96.825	**94.444**	95.144
F	95.166	95.527	94.613	96.503
G	99.423	94.661	95.623	93.978
H	95.527	95.094	96.296	94.949

Table 7.4: Sample of generated transcriptions from configurations C and D compared to the expected transcription.

Expected Transcription	THEY JUST WROTE IT OFF
Config. C result (with greedy search)	THE A S
Config. D result (with beam search)	THE EISENS

Following these results, another metric was analyzed that, with these inadequate results, would help better understand which models are more accurate when transcribing the signal.

The mentioned metric is Character Error Rate (CER), and similarly to WER, it is calculated as:

$$CER = \frac{S + D + I}{N} \tag{7.2}$$

where

$$\begin{aligned} S &= \text{number of substitutions,} \\ D &= \text{number of deletions,} \\ I &= \text{number of insertions,} \\ N &= \text{number of characters in the reference} \end{aligned}$$

but instead of substitutions, deletions, insertions being counted for words, it is now applied to characters.

Considering the transcriptions are produced character by character, and as the models are not accurate enough to build suitable words (or even incomplete sequences of characters that would make up a word), it makes sense to try to analyze the rate of correct characters alone. Using this metric, variations of accuracy between models and with different subsets of data are much more noticeable, as outlined in Table 7.5.

Table 7.5: Table showcasing CERs obtained from each different configuration for 4 subsets of EMG data (recorded in audible, whispered and silent speech modes, and the last with all the data).

Configuration	**Subset**			
	Audible	**Whispered**	**Silent**	**All data**
A	85.265	88.136	89.873	85.912
B	84.095	82.667	84.476	82.703
C	83.605	85.946	89.651	87.128
D	85.401	83.891	**80.794**	85.458
E	84.122	85.252	88.921	92.440
F	**79.156**	**81.143**	86.127	90.652
G	83.156	84.544	91.651	87.714
H	81.456	81.374	84.254	**80.535**

In all configurations (except D), the results for models trained with the silent speech subset show worse results than the other speech modes subsets. This is explained by the fact that the signal is less intense in silent speech than in audible speech, as the articulatory muscles do not have as much activity. Even though configuration F has the best performance for audible and whispered speech modes data, configuration H is the one that generalized better for all modes. Furthermore, has expected, configurations with beam search have shown better results than configurations with greedy search. In conclusion, these results are not yet accurate enough for reliable use in this project's messaging platform.

Nevertheless, as this system is inspired by Deep Speech 2 (as mentioned in Section 6.2.2), which is used for ASR, and having the acoustic data of all the recordings from the *EMG-UKA* corpus, it was possible to adapt our system to transcribe audio data to text, instead of using the EMG data. This adaptation of the system served not only to compare with the results obtained using the EMG data but also to integrate with the Frontend application, considering the transcribes achieved by SSR were not sufficiently accurate to deploy. Therefore, another experiment was performed to test the quality of the *EMG-UKA* dataset. As it is shown in Table 7.6, the results obtained with audio data were much more satisfactory than the results from previous experiments with EMG data.

Table 7.6: Table showcasing WERs and CERs obtained from G and H configurations for the *EMG-UKA* corpus audio data.

Configuration	WER	CER
G	79.504	32.684
H	56.227	27.867

Moreover, as showcased in Table 7.8, the transcripts decoded from these models are accurate enough to understand correctly build words and, in this case, almost the full sentence was correct. The configuration with beam search fixed spelling and overall improved the performance of the model, again as expected.

Table 7.8: Sample of generated transcriptions from configurations C and D compared to the expected transcription.

Expected Transcription	THEY EVEN KNEW THAT THEY WERE RUNNING INTO THE MOUNTAIN
Config. G result	THEY EVEN NEW THAT THEY WERE RONNING ING TO THE MOUNTAIM
Config. H result	THEY EVEN KNEW THAT THEY WERE RUNNING IN THE MOUNTAIN

Lastly, as mentioned in Section 6.1.5, as the SSR models did not achieve accuracy enough, this STT solution was the one integrated into the hands-free message text input feature of the platform, as a temporary solution.

7.4 Project Promotion

Having scientific validation of the project regarding its capabilities and contributions was considered important. To promote the project, two paper proposals have been submitted and accepted by two conferences. However, only one was actually published due to the overlapping of the conferences' dates. A talk proposal has also been submitted and accepted by a third conference. Each of these is described further in the following sections.

7.4.1 14th International Conference on Interfaces and Human Computer Interaction

The Interfaces and Human-Computer Interaction (IHCI) 2020 is a conference which focuses on the main issues concerning interface culture and design, as well as aspects of design, development and implementation of interfaces and their possible applications and implications for human and technology interaction. Therefore, it aims to venture innovative approaches and their applications in interface-related work.

With the help of both supervisors Paula Escudeiro and Carlos Ferreira, a short paper was submitted to this conference showcasing the project context and design, as well as evaluation methods. This paper can be found in Appendix B. The short paper was accepted and was due to be presented between the 23rd and 25th of July. However, due to the ongoing COVID-19 situation, the on-site event was canceled and was replaced with a virtual version. Despite this, the group had to decline the invitation for the presentation virtually because after the cancellation of the original conference and subsequent change to a virtual format, the dates overlapped with the conference that is going to be outlined in the following section.

7.4.2 22nd International Conference on Human-Computer Interaction

The Human-Computer Interaction International Conference (HCII) 2020 is a conference encompassing 21 distinguished international boards from 49 different countries, with two main thematic areas: Human-Computer Interaction and Human Interface and the Management of Information. Regarding the former, it addresses innovative topics regarding Human-Computer interaction theory, methodology and practice, with the development of novel methods and technologies regarding interface and UI.

Similarly to IHCI 2020, an abstract was submitted for this conference detailing its uses and new approach of interaction on the messaging panorama and on decentralized systems to be showcased on poster presentations. Similarly to the previous conference, the paper was submitted and was promptly accepted for poster presentation and publication on the proceedings of the conference. This publication can be consulted in Appendix C. In spite of that, due to the current COVID-19 pandemic situation, the physical venue for the conference was canceled and replaced with a virtual version that was held between the 19th and 24th of July. A poster to present the project was displayed on the virtual conference website during this period, to gain feedback on the project and possible future prospects. The poster exhibited is showcased in Appendix D. Moreover, as it was previously stated, the paper was published in this year's conference proceedings in the HCI International 2020 - Posters book published by Springer (Arteiro et al. 2020).

7.4.3 36th TERENA Networking Conference

TERENA Networking Conference (TNC), is one of the largest and most renown research and education networking conferences conducted in European soil, allowing to present and network with different universities, industry representatives, and organizations. The 2020's edition was to be hosted by the United Kingdom's National Research and Education Network organization and be held in Brighton, United Kingdom. This project was nominated and supported by the Future Talent Programme (FTP20) by Fundação para a Ciência e a Tecnologia (FCT). The application was made through FCT and, after being reviewed against other applicants, it was accepted to be presented in the conference lightning talks under FCT's alias. Lightning talks consist of five-minute presentations to the whole audience of TNC to present and promote the project to a wider audience. For this, three meetings, alongside other presented, were scheduled with Nadia Sluer, a representative of GÉANT, a pan-European data network encompassing research and education networks across Europe, and Michael Koenka, founder of MDK//Amsterdam. These meetings had the goal to further perfect the delivery and fine-tune the content of the presentations that were to be given during the conference and do so effectively. Even so, because of the rampant COVID-19 dire situation, the conference had to be canceled and could not be held virtually. Despite of this conundrum, the lightning talks were recorded and sent to the organization to be compiled. The presentation video which focused on this project has been delivered and is published on the GÉANT Learning and Development (GLAD) website, from GÉANT, the organizer of TNC (Sluer 2020).

Chapter 8

Conclusion

This chapter summarizes the final thoughts about the results obtained in this project and wraps up some contributions of this report to future works in the fields of BCI and SSI, as well as in HMI in general.

8.1 Results Discussion

Unfortunately, it was not possible to evaluate the BCI counterpart and its interaction with the Frontend App, due to the social distancing order and other restrictions imposed due to the current COVID-19 pandemic. Furthermore, the experiments with the team to use this interface revealed to be harder than expected, demanding several hours of training with the appropriate Emotiv software to master each command. This would difficult the test in inexperienced test subjects, expectedly taking more than one session of training before the actual test of the platform. Another project has also noticed that the mental command's training for Emotiv devices demands extreme focus for a long period, which is mentally challenging. This project test subjects complained about how hard it is to do this training and how painful it is to wear the EEG headset for that time (Aoun and Berg 2018).

Regarding the SSR system implemented, as previously described, the results were not satisfactory enough to be integrated into the overall platform and used alongside the BCI in the Frontend application. Due to limitations in time, it was not possible to investigate further into more subvocalization aspects and speech recognition techniques and models. Therefore, although the system developed for this project applied approaches from other research projects trying to accomplish better results using slight differences, the results lacked enough performance for the requirements of the project as an integrated platform. Another factor that reduces the performance of SSR in general when compared to ASR projects is the lack of public data and lack of standardization of the data collection is aspects such as electrode placement.

Regarding another phase mentioned as a possible next step, the test with sensors placed on subjects for data acquired by our team, also did not happen. This step would only happen if the performance of the SSR system was close to the goals established as a test to multisession capabilities. This data would vary from the *EMG-UKA* corpus since new subjects would be recorded, enhancing our system towards multiuser and multisession usage.

8.2 Work Contributions

The application provides a new approach and support for communication for both disabled people and those who do not suffer impairments, through BCI and SSI methods. Effectively, it can be considered as an attempt to be a breakthrough in the communication of impaired people, breaking barriers in interaction with their daily basis devices. Moreover, the project as a whole contributes to the state-of-the-art of two different fields, HMI and decentralized applications, promoting new interfaces using BCI and SSI cutting edge methods, and discussing privacy concerns, tackling these issues with both fields working together in a coherent platform.

Regarding the interaction between user and machine, the unorthodox way that is inherent to this application showcases the use of BCI on a real-life practical application. Furthermore, related to the silent speech paradigm, this project provides more experiments and approaches on the topic.

8.3 Future Work

The ongoing research of BCI devices and techniques is rapidly evolving and improving in performance, as invasive approaches begin to emerge, as mentioned in Chapter 3. All the advances in the field will enhance and push forward BCI applications in different areas, adapting these methods to be more mainstream and easy to use by anyone. These performance improvements would facilitate our efforts for training mental commands, lowering the learning curve, and easing the experiments and tests with our platform.

On the other hand, to improve the SSR system, different machine learning methods and architectures should be explored, such as the previously mentioned HMM approach. It is also plausible to explore an approach based on phoneme classification, and with a more rigorous analysis of the challenges of each language, as analyzed in (Freitas, Teixeira, and Dias 2012). Other aspects might be experiments as removing specific channels from the EMG samples from the *EMG-UKA* corpus in the data preparation step, as channel 5 is artifact-prone and can decrease the quality of the data (Jou et al. 2006).

Moreover, the combination of a BCI and a SSI can be applied to a myriad of other fields and projects other than just a messaging platform. This approach could help hand/arm disabled subjects in many other daily use cases, or be even applied to people without these disabilities to interact with their everyday devices in a private, silent, and hands-free method, especially in the Internet of Things era.

8.4 Final Appreciation

Although the project did not accomplish a messaging platform ready for production use by hand/arm disabled people, it does propose multiple new approaches towards more intuitive and hands-free applications, controlled with any type of BCI. The redesigns of conventional visual components from the Frontend App developed, propose new ways for interaction with applications promoting a more interactive and hand-free-focused design, building with these features in mind from scratch.

The same thing happens with the SSI approach that, even though it did not yield outstanding results, does sparkle discussion on the use of this approach for private communications and pushes forward an alternative to hands-free text input features, that as for now are dominated by STT systems.

Bibliography

Aaker, D.A. (2002). *Building Strong Brands*. Free. isbn: 9780743232135. url: https://books.google.pt/books?id=Dtq0QgAACAAJ.

AbleNet (2020). url: https://www.ablenetinc.com/trackerpro-2.

Amodei, Dario et al. (2015). *Deep Speech 2: End-to-End Speech Recognition in English and Mandarin*. arXiv: 1512.02595 [cs.CL].

Amputee Coalition (2014). *Statistics on hand and arm loss*. url: https://www.ishn.com/articles/97844-statistics-on-hand-and-arm-loss.

Aoun, Peter and Nils Berg (June 2018). PhD thesis. url: http://bth.diva-portal.org/smash/record.jsf?pid=diva2:1213634&dswid=1002.

Armour, Brian S. et al. (2016). "Prevalence and Causes of Paralysis—United States, 2013". In: *American Journal of Public Health* 106.10, pp. 1855–1857. doi: 10.2105/ajph.2016.303270.

Arteiro, Luís et al. (2020). "Brain-Computer Interaction and Silent Speech Recognition on Decentralized Messaging Applications". In: *HCI International 2020 - Posters*. Ed. by Constantine Stephanidis and Margherita Antona. Cham: Springer International Publishing, pp. 3–11. isbn: 978-3-030-50732-9.

Ball, Tonio et al. (2009). "Signal quality of simultaneously recorded invasive and non-invasive EEG". In: *NeuroImage* 46.3, pp. 708–716. doi: 10.1016/j.neuroimage.2009.02.028.

Birbaumer, N et al. (Apr. 1999). "A spelling device for the paralysed [4]". In: *Nature* 398, pp. 297–8. doi: 10.1038/18581.

Birbaumer, Niels (2006). "Breaking the silence: Brain–computer interfaces (BCI) for communication and motor control". In: *Psychophysiology* 43.6, pp. 517–532. doi: 10.1111/j.1469-8986.2006.00456.x. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1469-8986.2006.00456.x. url: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1469-8986.2006.00456.x.

Blankertz, B. et al. (Jan. 2006). "The Berlin brain-computer interface presents the novel mental typewriter Hex-O-Spell". In: *Clinical Neurophysiology* 113.

Brooke, John et al. (1996). "SUS-A quick and dirty usability scale". In: *Usability evaluation in industry* 189.194, pp. 4–7.

Cadwalladr, Carole and Emma Graham-Harrison (2018). "Revealed: 50 million Facebook profiles harvested for Cambridge Analytica in major data breach". In: *The guardian* 17, p. 22.

Clement, J. (2019). *Most popular global mobile messaging apps 2019*. url: https://www.statista.com/statistics/258749/most-popular-global-mobile-messenger-apps/.

del R. Milan, J. and J. M. Carmena (Jan. 2010). "Invasive or Noninvasive: Understanding Brain-Machine Interface Technology [Conversations in BME]". In: *IEEE Engineering in Medicine and Biology Magazine* 29.1, pp. 16–22. issn: 1937-4186. doi: 10.1109/MEMB.2009.935475.

Dobkin, Bruce H. (Apr. 2005). "Clinical practice. Rehabilitation after stroke". In: *The New England journal of medicine* 352.16. 352/16/1677[PII], pp. 1677–1684. issn: 1533-4406.

doi: 10.1056/NEJMcp043511. url: https://www.ncbi.nlm.nih.gov/pubmed/15843670.

Donchin, E., K. M. Spencer, and R. Wijesinghe (June 2000). "The mental prosthesis: assessing the speed of a P300-based brain-computer interface". In: *IEEE Transactions on Rehabilitation Engineering* 8.2, pp. 174–179. issn: 1558-0024. doi: 10.1109/86.847808.

Durov, Pavel (2019). *200,000,000 Monthly Active Users*. url: https://telegram.org/blog/200-million.

EMOTIV (2020). *Overview of API flow*. url: https://emotiv.gitbook.io/cortex-api/overview-of-api-flow.

Emotiv (2019). *EPOC Headset Details*. url: https://emotiv.gitbook.io/epoc-user-manual/using-headset/epoc%20_headset_details.

EMOTIV EPOC 14-Channel Wireless EEG Headset (2020). url: https://www.emotiv.com/epoc/.

Enberg, Jasmine (2019). "Global Messaging Apps 2019". In: eMarketer. url: https://www.emarketer.com/content/global-messaging-apps-2019#page-report.

Fabiani, G. E. et al. (Sept. 2004). "Conversion of EEG activity into cursor movement by a brain-computer interface (BCI)". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 12.3, pp. 331–338. issn: 1558-0210. doi: 10.1109/TNSRE.2004.834627.

Facebook (2019). *First Quarter 2019 Results Conference Call*. url: https://s21.q4cdn.com/399680738/files/doc_financials/2019/Q1/Q1-'19-earnings-call-transcript-(1).pdf.

Freitas, João, António Teixeira, and Miguel Dias (Jan. 2012). "Towards a silent speech interface for portuguese surface electromyography and the nasality challenge". In:

He, Horace (2019). "The State of Machine Learning Frameworks in 2019". In: *The Gradient*.

Hochreiter, Sepp and Jürgen Schmidhuber (1997). "Long short-term memory". In: *Neural computation* 9.8, pp. 1735–1780.

Hoyle, Roberto, Srijita Das, Apu Kapadia, Adam Lee, et al. (May 2017). "Was my message read?: Privacy and Signaling on Facebook Messenger". In: pp. 3838–3842. doi: 10.1145/3025453.3025925.

Hoyle, Roberto, Srijita Das, Apu Kapadia, Adam J Lee, et al. (2017). "Was my message read? Privacy and signaling on Facebook messenger". In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pp. 3838–3842.

Hueber, Thomas et al. (Apr. 2010). "Development of a Silent Speech Interface Driven by Ultrasound and Optical Images of the Tongue and Lips". In: *Speech Communication* 52, pp. 288–300. doi: 10.1016/j.specom.2009.11.004.

International Telecommunication Union (2019). *Measuring digital development: Facts and figures 2019*. ITU Publications.

Jaeger, Erica (2014). "Facebook Messenger: Eroding User Privacy in Order to Collect, Analyze, and Sell Your Personal Information, 31 J. Marshall J. Info Tech. & Privacy L. 393 (2014)". In: *J. Marshall J. Info. Tech. & Privacy L.* 31, p. i.

Jakob (Dec. 2010). url: https://www.nngroup.com/articles/kinect-gestural-ui-first-impressions/.

Johnson, Khari (2017). *Facebook Messenger passes 1.3 billion monthly active users*. url: https://venturebeat.com/2017/09/14/facebook-messenger-passes-1-3-billion-monthly-active-users/.

Jou, Stan et al. (Jan. 2006). "Towards continuous speech recognition using surface electromyography." In:

Jouse (2020). url: https://www.compusult.com/web/guest/assistive-technology/our-at-products/jouse3.

Jurafsky, Dan and James H. Martin (2009). *Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition*. Pearson Prentice Hall.

– (2014). "Hidden Markov Models". In: *Speech and language processing*. Prentice Hall, Pearson Education International.

Kaliski, Burt (2000). *PKCS# 5: Password-based cryptography specification version 2.0*. Tech. rep. RFC 2898, september.

Kapur, Arnav, Shreyas Kapur, and Pattie Maes (Mar. 2018). "AlterEgo: A Personalized Wearable Silent Speech Interface". In: *Proceedings of the 2018 Conference on Human Information Interaction and Retrieval - IUI 18*. doi: 10.1145/3172944.3172977.

Khan, M. Jawad, Melissa Jiyoun Hong, and Keum-Shik Hong (2014). "Decoding of four movement directions using hybrid NIRS-EEG brain-computer interface". In: *Frontiers in Human Neuroscience* 8, p. 244. issn: 1662-5161. doi: 10.3389/fnhum.2014.00244. url: https://www.frontiersin.org/article/10.3389/fnhum.2014.00244.

Klem, GH et al. (1999). "The ten-twenty electrode system of the International Federation. The International Federation of Clinical Neurophysiology". In: *Electroencephalography and clinical neurophysiology. Supplement* 52, pp. 3–6. issn: 0424-8155. url: http://europepmc.org/abstract/MED/10590970.

Koen, Peter et al. (2001). "Providing Clarity and A Common Language to the "Fuzzy Front End"". In: *Research-Technology Management* 44.2, pp. 46–55. doi: 10.1080/08956308.2001.11671418. eprint: https://doi.org/10.1080/08956308.2001.11671418. url: https://doi.org/10.1080/08956308.2001.11671418.

Koen, Peter A et al. (2002). "Fuzzy front end: effective methods, tools, and techniques". In: *The PDMA toolbook 1 for new product development*.

Koen, Peter A., Heidi M. J. Bertels, and Elko Kleinschmidt (2014). "Managing the Front End of Innovation—Part I: Results From a Three-Year Study". In: *Research-Technology Management* 57.2, pp. 34–43. doi: 10.5437/08956308X5702145. eprint: https://www.tandfonline.com/doi/pdf/10.5437/08956308X5702145. url: https://www.tandfonline.com/doi/abs/10.5437/08956308X5702145.

Lanning, Michael J (1998). *Delivering profitable value*. Perseus Books Group Cambridge.

Lapaire, Jean-Rémi (2018). "Why content matters. Zuckerberg, Vox Media and the Cambridge Analytica data leak". In: *ANTARES: Letras e Humanidades* 10.20, pp. 88–110.

Lapierre, Jozee (2000). "Customer-perceived value in industrial contexts". In: *Journal of business & industrial marketing*.

Larman, Craig (2005). *Applying UML and patterns: an introduction to object-oriented analysis and design and iterative development*. 3rd Edition. Prentice Hall PTR.

LeBlanc, Maurice (Nov. 2008). *"Give Hope - Give a Hand" - The LN-4 Prosthetic Hand*.

Lee, Jeeun et al. (2017). "Security analysis of end-to-end encryption in Telegram". In: *Proc. 34th Symposium on Cryptography and Information Security (SCIS 2017), Naha, Japan*.

Measuring Usability with the System Usability Scale (SUS) (Feb. 2011). url: https://measuringu.com/sus/.

Meltzner, Geoffrey S. et al. (Dec. 2017). "Silent Speech Recognition as an Alternative Communication Device for Persons with Laryngectomy". In: *IEEE/ACM transactions on audio, speech, and language processing* 25.12. PMC5851476[pmcid], pp. 2386–2398. issn: 2329-9290. doi: 10.1109/TASLP.2017.2740000. url: https://pubmed.ncbi.nlm.nih.gov/29552581.

Michael, Schünke (2005). *Prometheus: LernAtlas der Anatomie*. Georg Thieme Verlag.

Millán, J. D. R. et al. (Sept. 2010). "Combining Brain-Computer Interfaces and Assistive Technologies: State-of-the-Art and Challenges". In: *Frontiers in neuroscience* 4. 161[PII],

p. 161. issn: 1662-453X. doi: 10.3389/fnins.2010.00161. url: https://www.ncbi.nlm.nih.gov/pubmed/20877434.

Musk, Elon and Neuralink (2019). "An integrated brain-machine interface platform with thousands of channels". In: *bioRxiv*. doi: 10.1101/703801. url: https://www.biorxiv.org/content/early/2019/08/02/703801.

NaturalPoint (2020). url: https://www.naturalpoint.com/smartnav/products/4-at/.

Osterwalder, Alexander and Yves Pigneur (2013). *Business model generation a handbook for visionaries, game changers, and challengers*. Wiley & Sons.

Page (Apr. 2019). *Modes in User Interfaces: When They Help and When They Hurt Users*. url: https://www.nngroup.com/articles/modes/.

Panayotov, V. et al. (2015). "Librispeech: An ASR corpus based on public domain audio books". In: *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5206–5210.

Porter, Michael E. (2008). *Competitive Advantage: Creating and Sustaining Superior Performance*. Free Press.

Prince Sales, Tiago et al. (Oct. 2017). "An Ontological Analysis of Value Propositions". In: doi: 10.1109/EDOC.2017.32.

QuadLife (2020). url: https://quad.life/product.

QuadStick (2020). url: https://www.quadstick.com/.

Reeve, Amy, Eve Simcox, and Doug Turnbull (2014). "Ageing and Parkinsons disease: Why is advancing age the biggest risk factor?" In: *Ageing Research Reviews* 14, pp. 19–30. doi: 10.1016/j.arr.2014.01.004.

Reimers, Nils and Iryna Gurevych (2017). "Optimal Hyperparameters for Deep LSTM-Networks for Sequence Labeling Tasks". In: *CoRR* abs/1707.06799. arXiv: 1707.06799. url: http://arxiv.org/abs/1707.06799.

Ristic, Ivan (2013). *Deploying Forward Secrecy*. url: https://blog.qualys.com/ssllabs/2013/06/25/ssl-labs-deploying-forward-secrecy.

Rojas, Gonzalo et al. (Mar. 2018). "Study of Resting-State Functional Connectivity Networks Using EEG Electrodes Position As Seed". In: *Frontiers in Neuroscience* 12. doi: 10.3389/fnins.2018.00235.

Rosello, Pol, Pamela Toman, and Nipun Agarwala (2017). "End-to-end neural networks for subvocal speech recognition". In:

Saaty, Thomas (1988). "What is the analytic hierarchy process?" In: *Mathematical models for decision support*. Springer, pp. 109–121.

– (Jan. 2008). "Decision making with the Analytic Hierarchy Process". In: *Int. J. Services Sciences Int. J. Services Sciences* 1, pp. 83–98. doi: 10.1504/IJSSCI.2008.017590.

Sak, Haşim, Andrew Senior, and Françoise Beaufays (2014). *Long Short-Term Memory Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition*. arXiv: 1402.1128 [cs.NE].

Sauro, Jeff and James R. Lewis (2016). *Quantifying the user experience: practical statistics for user research*. Morgan Kaufmann.

Schrittwieser, Sebastian et al. (2012). "Evaluating the Security of Smartphone Messaging Applications." In: *NDSS*. Citeseer.

Sluer, Nadia (July 2020). *Luis Sampaio*. url: https://www.learning.geant.org/luis-sampaio/.

Smith, Julius O. (2011). *Spectral Audio Signal Processing*. online book, 2011 edition. http://ccrma.stanford.edu/~jos/sasp/.

StackOverflow (Feb. 2019). *Developer Survey Results 2019*. url: https://insights.stackoverflow.com/survey/2019#technology-_-most-loved-dreaded-and-wanted-web-frameworks.

Tencent (2019). *Tencent Q3 2019 Results, page 4*. url: https://cdc-tencent-com-1258344706.image.myqcloud.com/uploads/2019/11/13/8b98062831f2f28d9cb4616222a4d3c3.pdf.

TetraMouse (2018). url: https://tetramouse.com/tmxa.html.

Ulaga, Wolfgang and Andreas Eggert (Mar. 2006). "Relationship Value and Relationship Quality: Broadening the Nomological Network of Business-to-Business Relationships". In: *European Journal of Marketing* 40, pp. 311–327. doi: 10.1108/03090560610648075.

Vanacker, Gerolf et al. (Feb. 2007). "Context-Based Filtering for Assisted Brain-Actuated Wheelchair Driving". In: *Computational intelligence and neuroscience* 2007, p. 25130. doi: 10.1155/2007/25130.

Verna, Allee (Jan. 2008). "Value network analysis and value conversion of tangible and intangible assets". In: 9.1, pp. 5–24. issn: 1469-1930. doi: 10.1108/14691930810845777. url: https://doi.org/10.1108/14691930810845777.

Wand, Michael, Matthias Janke, and Tanja Schultz (2014). "The EMG-UKA Corpus for Electromyographic Speech Processing". In: *The 15th Annual Conference of the International Speech Communication Association, Singapore*. Interspeech 2014. url: https://www.csl.uni-bremen.de/cms/images/documents/publications/WandJankeSchultz_IS14_EMG-UKA-Corpus.pdf.

Welch, Chris (May 2016). *Google Keyboard for Android now has a one-handed mode*. url: https://www.theverge.com/2016/5/2/11568836/google-keyboard-android-app-one-handed-mode-new-features.

Williams, Ian et al. (Sept. 2018). "Contextual Speech Recognition in End-to-end Neural Network Systems Using Beam Search". In: pp. 2227–2231. doi: 10.21437/Interspeech.2018-2416.

Williamson, J. et al. (Oct. 2009). "Designing for uncertain, asymmetric control: Interaction design for brain–computer interfaces". In: *International Journal of Human-Computer Studies* 67, pp. 827–841. doi: 10.1016/j.ijhcs.2009.05.009.

Wilson, J.A. et al. (July 2006). "ECoG factors underlying multimodal control of a brain-computer interface;ECoG factors underlying multimodal control of a brain-computer interface". In: *Neural Systems and Rehabilitation Engineering, IEEE Transactions on;Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 14, pp. 246, 246–250, 250.

Wirth, Rüdiger and Jochen Hipp (2000). "CRISP-DM: Towards a standard process model for data mining". In: *Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining*. Springer-Verlag London, UK, pp. 29–39.

Xiong, W. et al. (2017). "The Microsoft 2017 Conversational Speech Recognition System". In: *CoRR* abs/1708.06073. arXiv: 1708.06073. url: http://arxiv.org/abs/1708.06073.

Yu, Dong and Li Deng (2016). *Automatic speech recognition: a deep learning approach*. Springer.

Zenkel, Thomas et al. (2017). *Comparison of Decoding Strategies for CTC Acoustic Models*. arXiv: 1708.04469 [cs.CL].

Appendix A

Meetings

Table A.1: Table showcasing a briefing of each meeting with the supervisors and experts in different fields.

Data	Brief	Participants
31/10/2019	Overall objectives of the project Guidance in project formalization	Fábio Lourenço Luís Arteiro Dr. Carlos Ferreira Dr. Paula Escudeiro
09/12/2019	Discussion of decentralized approaches	Fábio Lourenço Luís Arteiro Dr. Carlos Ferreira Dr. Miguel Areias (FCUP/INESC TEC)
10/02/2020	Report submission P1 Project progress update	Fábio Lourenço Luís ArteiroDr. Carlos Ferreira
23/03/2020	Feedback on P1 submission and presentation Improvements planning for final submission Project progress update	Fábio Lourenço Luís Arteiro Dr. Carlos Ferreira Dr. Paula Escudeiro
29/06/2020	Project progress update	Fábio Lourenço Dr. Carlos Ferreira

Appendix B

IHCI Paper Submission

SYNTHETIC TELEPATHY ON DECENTRALIZED MESSAGING APPLICATIONS

ABSTRACT

Peer-to-peer communication support for individuals with hand/arm impediments is currently scarce and poorly developed, with these users having heightened difficulty to utilize current mainstream applications. Furthermore, concerns for privacy and authorship of data has gained relevancy throughout the years. This paper outlines a new approach on human-machine interaction that is inclusive to people with these disabilities, with additional support for the vocally impaired by using an electroencephalography headset and electromyography surface electrodes respectively, all on a decentralized application that is developed to guarantee privacy on communication between users.

KEYWORDS

Brain-computer interface, silent-speech recognition, peer-to-peer communication, decentralization, distributed hash table

1. INTRODUCTION

With the advent of messaging applications, there is an apparent lack of support for disabled individuals to communicate on these platforms and to interact efficiently with other people. There has been significant progress addressing such issues mainly through electroencephalography (EEG), capable of recording brain activity that can be translated into actions (Vanacker *et al.*, 2007). Similarly, electromyography (EMG) can measure activity from muscles responsible for speech, enabling silent speech recognition (SSR) (Rosello, Toman and Agarwala, 2017).

Moreover, as centralized systems have come into scrutiny regarding privacy and security (Bouhnik, Deshen and Gan, 2014), development of alternative decentralized solutions have increased, a movement pioneered by Bitcoin (Nakamoto, 2008) that culminated in the blockchain technology. The latter has shown tremendous potential and applicability on a wide range of scenarios, boosted by the introduction of smart contracts by Ethereum (Wood and others, 2014), going beyond currency exchange. The surge of this technology paved the way for variants that maintain blockchain's main advantages but applied to different scenarios, having set a precedent and strides for decentralization across many fields (Wan *et al.*, 2017).

This solution’s main purpose is to provide a new approach for communication with support for users with hand/arm movements impairments, supported by contactless interaction with the device, where SSR is used in lieu of conventional speech-to-text features, consequently allowing oneself with speech disorders to also input text without clearly speaking or manually inserting it. It aims for the exchange of messages within a community where data privacy and integrity are crucial and each data is cryptographically stored with partial consensus, making it resilient against central points of failure with no scaling limits.

1.2 Related work

Since this project is a messaging platform, the text input is a key feature and it should be inclusive for people with disabilities. Experiments have been made using EEG to input text using virtual keyboards, like allowing a user to make binary decisions, iteratively splitting a keyboard in half until one letter remains, having a spelling rate of about 0.5 char/min (Birbaumer *et al.*, 1999). Another approach, which instead of binary choices, allows the selection of one between six hexagons, enabling the input of about 7 chars/min (Williamson *et al.*, 2009). Besides this, another research was conducted using a matrix of 6 by 6 independent cells, reaching 7.8 chars/min (Donchin, Spencer and Wijesinghe, 2000). Although input speed is not the primary focus of these projects, it must be user-friendly for instant messaging to be conceivable. Another studied approach to allow text input was through a silent speech interface (SSI), which also enables input by people with speech impairments (e.g. laryngectomy). Researches on this field still have a high word error rate (WER), as some achieved around 50% WER for their SSR attempt, for a vocabulary of 108 words (Wand, Janke and Schultz, 2014). Another group proposed working in phoneme recognition, using the same dataset. Despite the versatility, it only attained around 30% char accuracy (Rosello, Toman and Agarwala, 2017). Studies for this approach do not yield great results for a wide-range vocabulary, although appearing to be more promising when compared to EEG-based interfaces.

Regarding messaging platforms and their dedicated backend, there have been attempts to develop decentralized messaging applications that aim to address many issues present in centralized applications that are usually perceived as secure (e.g. Telegram) and allow decentralized authoritative messaging (Leavy and Ryan, 2018). Applications that follow the classic blockchain approach towards the messaging panorama include *Adamant*, which showcases an example of a fully anonymous, private and secure messaging between peers. Relying on its own token that is used for each transaction, such application relies on having no personal or sensitive data being extracted from the user (Evgenov *et al.*, 2017). A decentralized solution which follows a distributed hash table (DHT) approach is Jami, where the DHT is used to locate peer addresses and establish peer-to-peer connection without any personal sensitive data associated, with undelivered messages being stored locally.

2. OUR SOLUTION

The proposed solution encompasses SSI and BCI for the interaction of an application that partakes in a decentralized ecosystem that is user-centric, with high throughput and maintaining a bottom-line of privacy of data and message exchange. It envisions a way for people with arm/hand and speech impairments to be able to communicate with anyone. It is, therefore, divided into two distinctive counterparts: human-machine interaction and decentralized communication between peers.

2.1 Human-Computer Interaction

Aiming for a hands-free-controlled application with text input by silent speech, this project purposes a synthetic telepathy-based solution. Not only does it allow individuals with speech problems and arm motor disabilities to interact with the platform, but also creates a multitude of use cases for people without these limitations (e.g. privacy and mitigation of background noises when using hands-free mode in public since it uses silent speech as an alternative to the common text-to-speech and speech-to-text solutions; multitasking in everyday tasks while navigating with mental commands).

For the BCI component in this project, the Emotiv EPOC+ is being used for EEG and the Emotiv BCI software for the training and classification of the data gathered from a user. This device and related software are of great advantage since it speeds-up the project development cycle, considering it already has predefined mental commands for a subject to train, which can be applied in the user interface (UI) navigation. Considering the UI must be accessible and intuitive for both users with or without impairments using BCI and SSI or usual interfaces, a new design idea had to emerge to accomplish these goals.

Thus, besides from regular click/tap to interact, the user can also navigate through different chat rooms using four different commands: pull, push, left and right. Simulating a three-dimensional space, the pull and push commands pull closer to and push away from the user view, respectively, a whole screen related to a single chat room (**Figure 1**). The remaining left and right commands are used to slide through a carousel-type view with each chat room, always fixing a chat room in the middle, until the user pulls the one selected (Figure 1).

Regarding SSI, the first samples are being gathered for the prototype of this application. For data acquisition, a pair of EMG surface electrodes are being used, placed on the throat of the user, measuring myoelectric activity using bipolar configuration. For this first implementation, only data from a set of predefined words will be included, for validation and testing of different approaches. After feature extraction, several machine learning classification algorithms will be tested and compared for efficiency.

Other approaches more diverse than classification for only a small set of words are being studied for future work. For this feature, it is considered and tailored to the English language in order to reach to a broader audience.

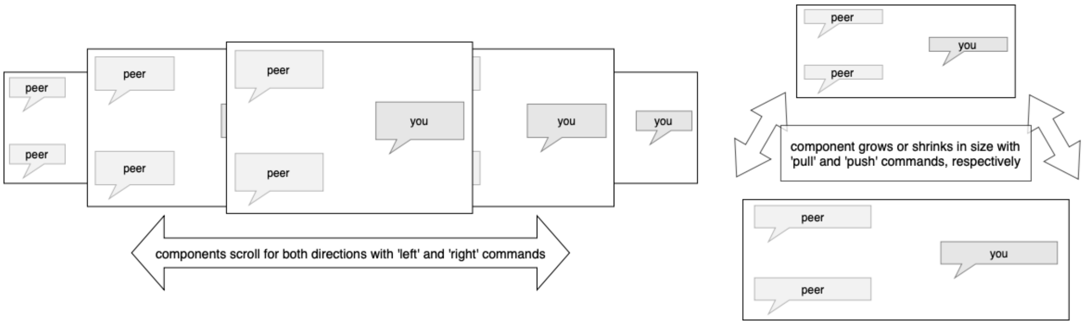


Figure 1. Navigation screens compatible with mental commands from the EEG headset.

2.2 Decentralized backend

The decentralized system is user-centric, that is, each user runs a bundle of the backend and frontend, has their own identity and their private and public shared data. Being on the same encrypted peer-to-peer network means each user can communicate with each other. Each node within the network runs the same application and, subsequently, the same logic and rules in which data exchanged must abide by. Each peer will validate their own data but also other's, making the system tamper-resistant and have data integrity.

Data resilience is also important in any messaging platform. That is, data must not get lost when users go offline. To address this issue, each piece of public data is witnessed, validated according to the system's logic that is present in every user and redundantly stored by a random selection of peers.

Users communicate with each other about data through gossiping. The latter is crucial for the well-being of the system because peers will gossip with each other to detect invalid data, such evidence of its author and take action to deal with malice users. This is a direct application of peer data replication and validation.

Inclusively, the system shares similarities with Jami in a way that it is based on a distributed hash table (DHT). This table, contrary to blockchain-based applications, is not replicated in its entirety each node. This decision of using a DHT is made in order to attain a higher TPS (transaction per second), throughput and lookup speed, in which each peer stores a segment/shard of the table.

2.2.1 User record

Every user has their own chain signed with their rightful private key. This chain can be thought of as a record of all the messages/data the user has produced and exchanged within the app and is stored on their own device. Each user has their own digital identifier using the public/private key pair cryptographic method. The combination of these two keys is imperative for the user online identity and for communication with other peers. The public key is shared with other participants who use it validate data and know who created it. Each user proves authorship of their data through digital signatures stemming from their private key. Any tampering is promptly detected by simply using the correspondent public key to verify the signature. If such validation fails, the data is considered invalid and this finding is gossiped and broadcasted to the rest of the network.

The DHT is where all public data resides in. The word public is used very loosely. Every entry is hashed so as to make it untraceable, so it is not necessarily public as in it is available for everyone to see. Each and every entry that resides within the DHT are essentially user chain entries that are merely marked as public, hashed with the peer's address and signed with its private key.

2.2.2 Validation and direct peer-to-peer communication

The security, state and integrity of the whole system is maintained by the logic that is replicated amongst every node. Data integrity is ruled by these rules which, in turn, uphold the security of the system. Besides normal messaging, users that want to share secret or sensitive data directly with another user that can later be destroyed can use direct node-to-node messaging through an end-to-end encrypted channel by resolving the specific agent's IP address.

The application is built on top of the Holochain framework, who handles a deal of network concerns and cryptography. Additionally, Holochain provides a way for the system to be interoperable with other networks build on the same Holochain network, making the application have more expandable prospects.

2.3 Evaluation

BCI evaluation process will consist of usability tests to gather feedback on how the platform improves their communication and overall navigation compared to similar apps. The factors that will be analysed are comfort, interaction speed, accuracy, and learning curve, and each will be evaluated on a typical five-level Likert scale. For SSI, the data sets used for each model created will be divided into smaller sets of training, test and cross-validation sets. These test sets will examine the accuracy of classification, improving our models to obtain the lowest error rate possible.

The Holochain framework provides support for both unit and functional tests. These will be useful to provide a measurement of code coverage and reliability of the written code at this level. End-to-end tests will ultimately leverage a way to measure performance as the solution scales. The throughput is tested through stress tests and assess if the former is consistent. The resilience of the system is tested through man-in-the-middle attacks, at cryptography level, and denial-of-service attempts.

3. CONCLUSION

The showcased solution mitigates issues from centralized communication, regardless of impairment status of individuals, applying methods of machine learning coupled with SSI and BCI. Moreover, it depicts a new proposal on the decentralized messaging platform, either from human-machine interaction standpoint or the distributed ecosystem counterpart. Taking a more user-centric approach towards the problem, utilizing a distributed hash table with partial consensus instead of data replication in its entirety with global consensus is more adequate to the messaging panorama. These design choices have a direct effect on performance and throughput of the system; more transactions per second can occur whilst the exchange of data is still secure and private and resilient against tampering attacks, while removing scalability limitations.

The limitations of this application comprise of the number of options for navigation and control over the application and the limited vocabulary and language in which the user can utilize subvocalization. Furthermore, the application is yet to support file exchange, although this issue could be addressed through a separate DHT or parallel channels.

REFERENCES

Birbaumer, N. *et al.* (1999) 'A spelling device for the paralysed', *Nature*, 398, pp. 297–298. doi: 10.1038/18581.

Bouhnik, D., Deshen, M. and Gan, R. (2014) 'WhatsApp goes to school: Mobile instant messaging between teachers and students', *Journal of Information Technology Education: Research*, 13(1), pp. 217–231.

Donchin, E., Spencer, K. M. and Wijesinghe, R. (2000) 'The mental prosthesis: assessing the speed of a P300-based brain-computer interface', *IEEE Transactions on Rehabilitation Engineering*, 8(2), pp. 174–179. doi: 10.1109/86.847808.

Evgenov, P. *et al.* (2017) *White paper: Adamant messaging application*. Available at: https://adamant.im/whitepaper/adamant-whitepaper-en.pdf.

Leavy, T. M. and Ryan, G. (2018) 'Decentralized authoritative messaging'. Google Patents.

Nakamoto, S. (2008) 'Bitcoin: A peer-to-peer electronic cash system'.

Rosello, P., Toman, P. and Agarwala, N. (2017) 'End-to-end neural networks for subvocal speech recognition'.

Vanacker, G. *et al.* (2007) 'Context-Based Filtering for Assisted Brain-Actuated Wheelchair Driving', *Computational intelligence and neuroscience*, 2007, p. 25130. doi: 10.1155/2007/25130.

Wan, Z. *et al.* (2017) 'BKI: Towards accountable and decentralized public-key infrastructure with blockchain', in *International Conference on Security and Privacy in Communication Systems*, pp. 644–658.

Wand, M., Janke, M. and Schultz, T. (2014) 'The EMG-UKA Corpus for Electromyographic Speech Processing', in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*.

Williamson, J. *et al.* (2009) 'Designing for uncertain, asymmetric control: Interaction design for brain–computer interfaces', *International Journal of Human-Computer Studies*, 67, pp. 827–841. doi: 10.1016/j.ijhcs.2009.05.009.

Wood, G. and others (2014) 'Ethereum: A secure decentralised generalised transaction ledger', *Ethereum project yellow paper*, 151(2014), pp. 1–32.

Appendix C

HCII Paper In Conference Proceedings

Brain-Computer Interaction and Silent Speech Recognition on Decentralized Messaging Applications

Luís Arteiro(✉), Fábio Lourenço, Paula Escudeiro, and Carlos Ferreira

Polytechnic of Porto - School of Engineering (ISEP), 4200-072 Porto, Portugal
{1150625, 1150434, pmo, cgf}@isep.ipp.com

Abstract. Peer-to-peer communication has increasingly gained prevalence in people's daily lives, with its widespread adoption being catalysed by technological advances. Although there have been strides for the inclusion of disabled individuals to ease communication between peers, people who suffer hand/arm impairments have scarce support in regular mainstream applications to efficiently communicate privately with other individuals. Additionally, as centralized systems have come into scrutiny regarding privacy and security, development of alternative, decentralized solutions has increased, a movement pioneered by Bitcoin that culminated on the blockchain technology and its variants.
Within the inclusivity paradigm, this paper aims to showcase an alternative on human-computer interaction with support for the aforementioned individuals, through the use of an electroencephalography headset and electromyography surface electrodes, for application navigation and text input purposes respectively. Users of the application are inserted in a decentralized system that is designed for secure communication and exchange of data between peers that are both resilient to tampering attacks and central points of failure, with no long-term restrictions regarding scalability prospects. Therefore, being composed of a silent speech and brain-computer interface, users can communicate with other peers, regardless of disability status, with no physical contact with the device. Users utilize a specific user interface design that supports such interaction, doing so securely on a decentralized network that is based on a distributed hash table for better lookup, insert and deletion of data performance. This project is still in early stages of development, having successfully been developed a functional prototype on a closed, testing environment.

Keywords: Brain-computer interface · Silent speech recognition · Peer-to-peer communication · Decentralization · Distributed hash table.

1 Introduction

With the advent of messaging applications, there is an apparent lack of support for disabled individuals to communicate on these platforms and use such applications to interact efficiently with other people. There has been significant progress addressing such issues mainly through electroencephalography (EEG) sensors capable of recording brain activity that, powered by machine learning, can translate it into actions [6]. Similarly, electromyography (EMG) electrodes can measure activity from muscle responsible for speech, enabling silent speech recognition (SSR) [11], that can be used for private and silent text input into the application.

Additionally, privacy and security of data within centralized systems have been under the spotlight, especially on mobile applications [2], contributing to the rise of the development of alternative, decentralized solutions have increased, a movement first created by Bitcoin [8] that effectively stapled blockchain as a growing technology. The latter has shown tremendous potential and applicability on a wide range of scenarios, boosted by the introduction of smart contracts by Ethereum [13], going beyond currency exchange. These assets allow trustless, secure, peer-to-peer value transfer that can easily be applied to the messaging paradigm, thus allowing immutable, secure messaging between nodes. The surge of this technology paved the way for variants that maintain blockchain's main advantages but change their scope. These have set a precedent and strides for decentralization across many fields [10].

2 Related Work

Regarding brain-computer interfaces (BCI) using non-invasive EEG, research projects have emerged allowing users to control virtual keyboards and mouses, empowering individuals with motor disabilities in their interaction with computers. Regarding app's navigation using this approach, there are experiments on cursor movement that allow a user to move it by a BCI. Tests with different approaches within movements for one or two dimensions showed an accuracy of approximately 70% [5]. A similar project utilizes a hybrid near-infrared spectroscopy-electroencephalography technique trying to extract four different types of brain signals, decoding them into direction symbols, namely, "forward", "backward", "left", and "right". The classification accuracy for "left" and "right" commands was 94.7%, for "forward" was 80.2% and 83.6% for "backward" [6]. Later in this document, it will be shown that our system uses a similar approach for the set of commands used for navigation in the platform.

Since this project concerns a messaging platform, text input is a key feature that must offer support for the target audience - disabled subjects. Experiments have been made using EEG to input text using virtual keyboards, allowing a user to make binary decisions and iteratively splitting a keyboard in half until one letter remains, having a spelling rate of about 0.5 char/min [1]. Another approach, which instead of doing binary choices, allows the selection of one between six hexagons, enabling the input of about 7 chars/min [12]. Although input speed is not the central focus of these projects, it must be user-friendly for instant messaging to be conceivable. Yet another studied approach to allow text input was through a silent speech interface (SSI), which also enables input by people with speech impairments (e.g. laryngectomy). Researches on this field still have a high word error rate (WER), as some have achieved around 50% WER for their SSR attempt, using a vocabulary of 108 words through the EMG-UKA corpus [11]. Another group proposed another solution, using the same corpus, attaining around 30% char accuracy [9]. Studies for this approach do not yield satisfactory results for a wide-range vocabulary, although appearing to be more promising when compared to EEG-based interfaces.

Regarding messaging platforms and their dedicated backend, there have been attempts to develop decentralized messaging applications that aim to address many issues present in centralized applications that are usually perceived as secure (e.g. Telegram) and allow decentralized authoritative messaging [7]. Applications that follow the classic blockchain approach towards the messaging panorama include Adamant, which showcases an example of a fully anonymous, private and secure messaging between peers. Relying on its own token that is used for each transaction, such application relies on having no personal or sensitive data being extracted from the user [4]. A decentralized solution which follows a distributed hash table (DHT) approach is Jami, where the DHT is used to locate peer addresses and establish peer-to-peer connection without any personal sensitive data associated, with undelivered messages being stored locally.

3 Our Solution

The proposed solution encompasses a new approach to the interaction of an application that partakes in a decentralized ecosystem that is user-centric, with high throughput and maintaining a bottom-line of privacy of data and message exchange. It envisions a way for people with arm/hand to be able to communicate with anyone. It is, therefore, divided into two distinctive counterparts: human-machine interaction, through BCI and SSI, and decentralized communication between peers. This is further outlined in Figure 1.

3.1 Human-Machine Interaction

Aiming towards a hands-free controlled application with text input through silent speech, this project proposes a synthetic telepathy-based solution to messaging. It allows individuals with arm motor disabilities to interact with the platform and also creates a multitude of use cases for people without these limitations (e.g. privacy and mitigation of background noises when using speech-to-text features in public; multitasking while navigating with mental commands).

Brain-Computer Interface For the BCI component in this project, Emotiv EPOC+ is being used for EEG recordings and the Emotiv BCI software for training and classification of the data gathered from the user. This device and related software are of great advantage since it accelerates the overall project development considering it already has predefined mental commands for a subject to train, which can be applied for the user interface (UI) navigation. Considering the UI must be accessible and intuitive for both users with or without

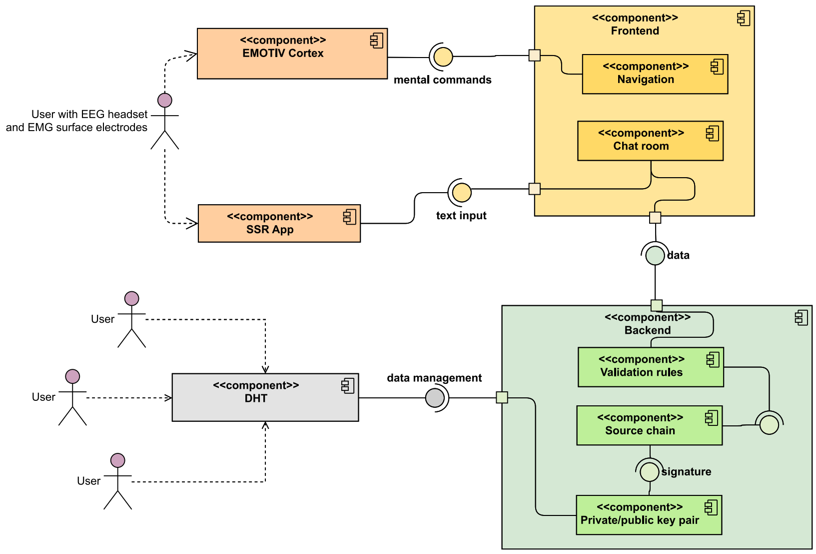


Fig. 1. Overall conceptual view of the platform, outlining the interaction between the user and interface, as well as showcasing the decentralized system where the user is inserted into (DLT - Distributed Ledger Technology), outlining the different components that make up the backend.

impairments, a new design idea had to emerge to accomplish these goals. Thus, besides from regular click/tap to interact, the user can also navigate through different chat rooms using four different commands, "pull", "push", "left", and "right". Simulating a two-dimensional space, the "pull" and "push" commands pull closer to and push away from the user view, respectively (see Figure 2), and the remaining "left" and "right" commands are used to slide through a carousel-type view with each chat room, always fixing a chat room in the middle, until the user pulls the one selected (see Figure 3). Furthermore, more commands may be added for others minor tasks, or these already implemented commands can correspond to multiple actions depending on the state of the application or feature that is being used.

Silent Speech Interface Regarding SSI, it is being used the aforementioned EMG-UKA corpus for implementation and testing of the system in its early stages. Approaches for this system are still being studied, and the first experiments are being conducted on a session-dependent model attempting to classify phones (speech sounds) from EMG data. Further developments will allow users to input full words into the system. In order to reach a broader audience and using the exiting corpus, the system is tailored for the English language.

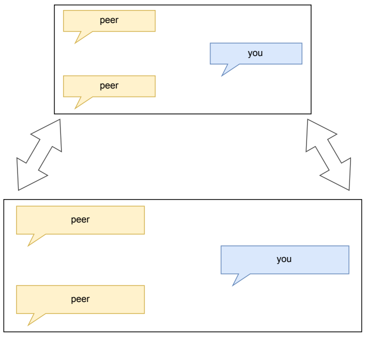


Fig. 2. "Pull" and "push" commands effect the UI, allowing the user to enter and leave chatrooms.

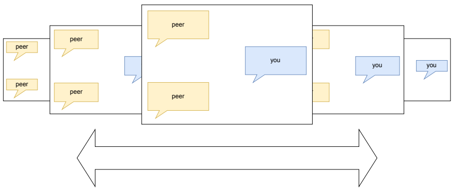


Fig. 3. The user can slide right or left and navigate through his/her chatrooms through "right" and "left" mental commands.

3.2 Decentralized Backend

As it was aforementioned, individuals who make use of the platform, whether these suffer from any impairment or not, will partake in a decentralized ecosystem that is intended to provide secure peer-to-peer communication with data integrity. The system is agent-centric, meaning each user runs a copy of the backend, have their own identity and their private and public shared data. Being on the same encrypted peer-to-peer network entails each user can communicate with each other directly to maintain its integrity. Having each peer hold the same application bundle with the respective logic, it is possible to verify other peer's transactions and data created. Each data has proof of authorship (i.e. a signature). Since every data is recorded and validated by the peers within the system, it is tamper-resistant - this showcases data integrity.

Data resilience is also important in any messaging platform. That is, data must not get lost when users go offline. To address this issue, each piece of public data is witnessed, validated according to the system's logic that is present in every user and stored by a random selection of peers. This makes it that the community and cooperating parties detect invalid data, gossip such evidence of harmful agents and take action to deal with malice data/users. This is a synonym to peer data replication and data validation.

The system shares similarities with Jami in a way that it is based on a distributed hash table (DHT). This table, contrary to blockchain-based applications, is not replicated in each node. To attain a higher transactions per second (TPS), throughput and lookup, each peer stores a segment of the table. This DHT is where the data resides and, in cases where users go offline, is stored to later be retrieved by the recipient. The implementation of this table is based on top of the Holochain framework, allowing for easier data replication and validation between peers. As more users join in the network, more computational power is contributed to the environmnent and data replication is more redundant, allowing the system to scale as more users partake within the system.

User Record Each user has their own chain signed with their rightful private key. This chain can be thought of as a record of all the messages/data the user has produced and exchanged within the app and is stored on their own device. Each individual has their own digital identifier using the public/private key pair cryptographic method. The combination of these two keys is imperative for the user online identity and for communication with other peers. The public key is shared with other participants. Each user proves authorship of their data through digital signatures stemming from their private key. Any tampering is promptly detected by simply using the correspondent public key to verify the signature. If such validation fails, the data is considered invalid and this finding is gossiped and broadcasted to the rest of the network. The distributed hash table is where all public data resides in. The word "public" is used very loosely. Every entry is hashed so as to make it untraceable so it is not necessarily public as in it is available for everyone to see. Each and every entry that resides within the DHT are essentially user chain entries that are merely marked as public.

Validation and Direct Peer-To-Peer Communication The security, state and integrity of the whole system is maintained by logic that is hard-coded and bundled in every node. Data integrity is ruled by these rules which, in turn, uphold the security of the system. All data, whether it is in the user private chain or in the public DHT, is validated according to these rules. Normal messaging occurs when two users are online and is made through direct peer-to-peer connection in an end-to-end encrypted channel by resolving each agent's IP address. This method only works when both peers are online. To circumvent this, private messages are encrypted using the recipient's public key and published to the DHT, to later be retrieved by the recipient when they are back online.

Most of these networking protocols and distributed computing scenarios are handled by the Holochain framework, from which the architecture is based from. Not only it resolves many concerns regarding encryption but also allows for interoperatibility with other systems that take part within the Holochain network.

4 Preliminary Evaluation

As this is a work-in-progress and with no tangible results yet yielded, it is possible to define an assessment strategy for it. BCI evaluation process will consist of usability tests performed by hand/arm impaired individuals to gather feedback on how the platform improves the user's communication and overall navigation compared with similar apps. Prior to testing, the subjects ought to train the BCI headset and tailor it to their mental activity for the model to recognize the commands from a specific user. Afterwards, subjects will be asked to perform a planned set of actions that entails navigating in the platform and interact with it. Lastly, a survey will be conducted following the System Usability Scale [3], with 10 items answered in a degree of agreement or disagreement with each statement on typical five-level Likert scale. For SSI, the datasets used for each model created will be divided into smaller training and testing sets, allowing to evaluate their accuracy, aiming to obtain the lowest WER possible.

The Holochain framework provides support for both unit and functional tests. These are useful to provide a measurement of code coverage and respective reliability. End-to-end tests will ultimately leverage a way to measure performance as the solution scales. The throughput is tracked as the number of users grows and its consistency is assessed. Early results have showcased successful data creation on the public DHT on a small number of users (five) with a step duration of 1000 miliseconds, where the period is halved at every stage. In this, the stress test is conducted indefinitely and increases pace at every stage. Additionally, a small increase of throughput was noticed when more users joined the network, from two peers to five. These experiments have not shown any sensible traces of data corruption nor delay, although more tests on a higher scale are be needed to form more conclusive results.

5 Conclusion

This project mitigates communication barriers between subjects with or without hand/arm impairments and allows the inclusion of everyone into a single messaging system, using a new strategy to this type of platform, applying a synthetic telepathy approach for the interaction with it. Additionally, if the project produces positive results, the range of applicability for both BCI and SSI approaches can extend beyond the messaging paradigm.

The system that was outlined showcases a new proposal on the decentralized messaging platform, either from human-machine interaction standpoint or the distributed ecosystem counterpart. Following an user-centric approach towards the problem, utilizing a DHT with ad hoc peer consensus instead of data replication in its entirety with global consensus is more adequate to the messaging panorama. These design choices have a direct effect on performance and throughput of the system: more TPS can occur whilst the exchange of data is still secure, private and resilient against tampering attacks. This essentially means that scalability does not pose as a problem on a long-term perspective, unlike many solutions that are based on the classic blockchain concept.

6 Future Work

With the drafted architecture and design choices, there is still room for improvement for future project prospects so as to be providable to a broader audience and maintain a performance bottom line. On the human-machine interface counterpart, for the BCI component, other commands can be mapped and used by the platform, giving the user more options for navigation and control over the application, for a wider set of features, but keeping the interaction as intuitive as possible. SSI-wise, aiming for a more natural and reliable text input system for any user, efforts will be made for a session and user-independent system, enabling the its usage by multiple users and sessions without much accuracy fluctuations.

One of the objectives regarding the decentralized backend would be to go beyond message exchange and also extend to file sharing. Approaches could be developed similar to InterPlanetary File System (IPFS). However, this topic is sensitive since file storage takes up more space than common messages. Replication of data between peers would need to be addressed differently, perhaps through parallel channel or DHT altogether. Furthermore, voice calling could be another feature present within the application that could co-exist and be made through node-to-node channels.

References

1. Birbaumer, N., Ghanayim, N., Hinterberger, T., Iversen, I., Kotchoubey, B., Kübler, A., Perelmouter, J., Taub, E., Flor, H.: A spelling device for the paralysed. Nature **398**, 297–8 (04 1999). https://doi.org/10.1038/18581
2. Bouhnik, D., Deshen, M., Gan, R.: Whatsapp goes to school: Mobile instant messaging between teachers and students. Journal of Information Technology Education: Research **13**(1), 217–231 (2014)
3. Brooke, J., et al.: Sus-a quick and dirty usability scale. Usability evaluation in industry **189**(194), 4–7 (1996)
4. Evgenov, P., Lebedev, A., Soloduhin, D., Pikhtovnikov, M., Vorobev, A., Lebedev, S., Anokhov, P.: White paper: Adamant messaging application. Tech. rep. (2017), https://adamant.im/whitepaper/adamant-whitepaper-en.pdf
5. Fabiani, G.E., McFarland, D.J., Wolpaw, J.R., Pfurtscheller, G.: Conversion of eeg activity into cursor movement by a brain-computer interface (bci). IEEE Transactions on Neural Systems and Rehabilitation Engineering **12**(3), 331–338 (Sep 2004). https://doi.org/10.1109/TNSRE.2004.834627
6. Khan, M.J., Hong, M.J., Hong, K.S.: Decoding of four movement directions using hybrid nirs-eeg brain-computer interface. Frontiers in Human Neuroscience **8**, 244 (2014). https://doi.org/10.3389/fnhum.2014.00244, https://www.frontiersin.org/article/10.3389/fnhum.2014.00244
7. Leavy, T.M., Ryan, G.: Decentralized authoritative messaging (Nov 27 2018), uS Patent 10,142,300
8. Nakamoto, S., et al.: Bitcoin: A peer-to-peer electronic cash system (2008)
9. Rosello, P., Toman, P., Agarwala, N.: End-to-end neural networks for subvocal speech recognition (2017)
10. Wan, Z., Guan, Z., Zhuo, F., Xian, H.: Bki: Towards accountable and decentralized public-key infrastructure with blockchain. In: International Conference on Security and Privacy in Communication Systems. pp. 644–658. Springer (2017)
11. Wand, M., Janke, M., Schultz, T.: The emg-uka corpus for electromyographic speech processing (08 2014)
12. Williamson, J., Murray-Smith, R., Blankertz, B., Krauledat, M., Müller, K.R.: Designing for uncertain, asymmetric control: Interaction design for brain–computer interfaces. International Journal of Human-Computer Studies **67**, 827–841 (10 2009). https://doi.org/10.1016/j.ijhcs.2009.05.009
13. Wood, G., et al.: Ethereum: A secure decentralised generalised transaction ledger. Ethereum project yellow paper **151**(2014), 1–32 (2014)

Appendix D

HCII Poster

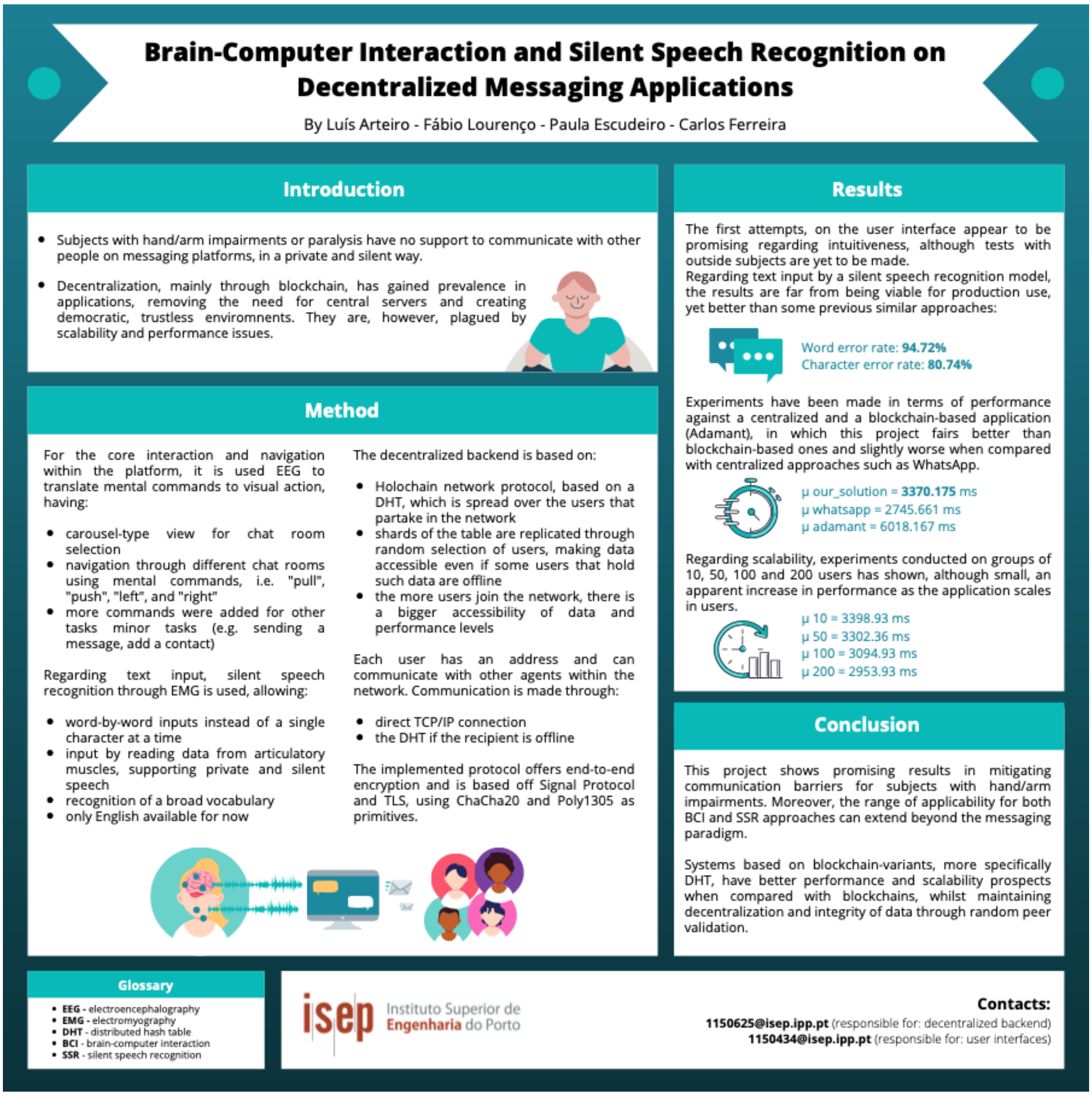


Figure D.1: Poster submitted to HCII 2020 to promote the project.