Modular Framework for Data Acquisition and Annotation to Support Interaction Scenarios

University of Aveiro(*UA*)

Guilherme Mendonça Claro,
Eduardo Lopes Fernandes,
João Afonso Pereira Ferreira,
Pedro Durval Cruzeiro,
Tiago Figueira Mostardinha

VERSAO 1

Acronyms

HCI Human Computer Interaction

ML Machine Learning

HD High Definition

DDS Data Driven SCRUM

IoT Internet of things

AI Artificial Intelligence

MFDAA Modular Framework for Data Acquisition and Annotation to Support Interaction Scenarios

Project Report

Computer and Informatics Engineering's Project
2022/2023

Supervised by: Samuel Silva & Fábio Barros

DETI - Departamento de Electrónica, Telecomunicações
e Informática

Mestrado Integrado em Engenharia de Computadores e
Telemática
&
Licenciatura em Engenharia de Computadores e
Informática

November 15, 2022

Abstract

Interaction scenarios have seen an increased relevance with the growth of digital environments, however despite the outside need for more engaging technology, it is still a very difficult area to be approached by companies that don't possess immense investment power.

The growing world of technology has brought people from all sorts of backgrounds into a common space. However, despite its benefits, it also means that not everyone is suited to keep up with these new developments. Adding new ways of interacting with these novel systems is, therefore a priority for developers, with one of the main focus being non-verbal interaction.

For systems to recognize these cues they need to have a fitted Machine Learning algorithm, capable of accurately recognizing the intended gestures, expressions, sounds, etc. Be that as it may, modelling these algorithms is a repetitive, time and resource consuming process often causing struggles to smaller teams and independent developers. The platform we developed facilitates data acquisition, annotation and allows for visualization of valuable insights while providing collaboration by exchanging the collected data and its information between users. In addition, this framework promotes data re-usability, offering users the possibility to build upon existing modules that already fit their needs. Overall, Modular Framework for Data Acquisition and Annotation to Support Interaction Scenarios (MFDAA) presents a comprehensive system that aims to face the struggles of Non-Verbal Cue recognition by providing a versatile and automated framework for data acquisition and annotation, empowering users with visualization tools and pushing towards a collaborative environment.

Index

List of Figures

Chapter 1

Introduction

The growth of technology has had a significant impact on our daily lives. In recent years, we have seen rapid advances in fields such as artificial intelligence, robotics and machine learning [1]. These technologies have made our lives more convenient and efficient in many ways, making interaction with these technologies a more intuitive process. However it still exists a very big divergence between the way we interact with technology and the way we interact with each other, this lack of naturality when interacting with a machine comes mainly from the unavailability of recognizing non-verbal cues.

Machine Learning (ML), the process used to develop Artificial Intelligence (AI) works by analysing a relevant amount of data specific to the scenario its being trained to and comparing its judgement to the expected answer and adjusting itself as it progresses to the data. Teams looking to develop these kinds of technologies to recognize non-verbal cues accurately face a relevant issue early into their work, this issue is the process of gathering enough information that fits the scenario they're trying to model their AI after. As of now this acquisition and annotation process is a very repetitive and time consuming process with lack of standardization and scalability.

Our framework, therefore, aims to provide a standard for data acquisition and annotation, tailored towards non-verbal interaction scenarios thus striving to make this a more efficient, cooperative and scalable process creating a more inviting environment for new stakeholders trying to get involved in this novel area.

1.1 Motivation

Finding a dataset that fits a developer's needs is a struggle found by several teams around the world that constantly need to face the grueling reality of initiating the creation of a new plathera of information that can be used to model their technology. More often than not, the dataset developed wont be re-used by teams outside this group's small circle and even if it was it would probably be a hard to truly adapt it to this third-party work due to the lack of standardization.

A way of enabling the use of certain levels of abstraction of already existing datasets would greatly diminish the difficulty of this early stage tasks, providing

users with the power to extract the information that fits their needs and allowing for personal expansion upon it while also motivating their collaboration towards the public scientific community.

Besides this re-usability we also strove to implement a easy and efficient way of acquiring and annotating data making it as accessible as possible to people from the most varied backgrounds in an effort to attract not only developers but also researchers from various areas looking to study any angle of non-verbal interaction. With our work these secondary stakeholders can also easily create from scratch small datasets to use with test subjects in their experiences. 000

1.2 Goals

To initiate this project and gain a better understanding of the significance of the problem at hand, the team embarked on delineating the goals of the envisioned platform. The main goals is:

- Propose a framework to allow a systematic creation and annotation of *datasets* to support research on Non-Verbal Interaction. This should encompass:
  1. Understanding the overall user needs and usecases to establish the requirements
  2. Propose a conceptual user aligned with the requirements
  3. Develop and validate a first prototype of the proposed framework

Aditionally, this project is performed on the scope of PECI, so, in this context, the goals also include:

- Gain hands-on experience in executing a project, to actively participate in all stages of a project, from conceptualization to completion, to gain practical experience in project management, problem-solving, and technical implementation.
- Practical Application allowing the team to apply the knowledge and skills acquired during our studies.
- Project Management Skills, to develop abilities in planning, organizing, and executing projects effectively within specified constraints such as time, resources, tasks and scope.
- Agile Project Management aims to familiarize students with iterative and incremental project development approaches, emphasizing adaptability, collaboration, and continuous improvement
- Problem-Solving, enhance team members problem-solving and critical thinking skills when confronted with a new challenge.

1.3 Document Structure

Clarifying the **structure** of this report, firstly, we are going to address Background, Related Work at the chap 2 which is where is present the literature

research made by the team along side with a description of how it was adapted to our own needs. Secondly, will be displayed the Personas, Scenarios, Use Cases and Requirements chapter (chap. 3) where all the requirements, the creation fictional characters to understand the needs for this project, the technologies that were chosen for the execution of this project will be present by making a succinct description of their purpose and the main reasons leading to their choice. Additionally, in the Conceptualization chapter (chap. 4) where the concept of this framework will be elucidated as well as the initial steps undertaken to create and validate the preliminary UI low-fidelity interface. On the Development of MFDAA chapter 5, a comprehensive and detailed description of the project will be provided, the methodology and steps taken addressed to this project, including a description of the architecture, the technologies and reasons of choice. In this same chapter, will also be exhibited an overview of the system, explaining every module and how they interact with each other and finally an user validation/evaluation of the final product. Subsequently, in Chapter 6, the goals will be discussed in the conclusions to assess whether the initial objectives were successfully achieved. Furthermore, the discussion will explore the potential for further improvement and future work for our framework as well as the team distribution and contributions.

Chapter 2

Background and Related work

> "*Data acquisition is the process of taking measurements of real-world physical occurrences using signals and digitizing them so that a computer and software may alter them.*"
>
> *in* `https://shorturl.at/yVZ07`

A modular framework for data acquisition and annotation refers to a flexible [1] and scalable [2] approach for collecting and labeling data in a structured manner. It involves breaking down the process into modular components that can be easily customized, extended, and interconnected to suit specific data collection and annotation requirements. Thus, this kind of framework typically consists of two main components:

1. **Data acquisition** [2] involves gathering raw data from various sources. This can include scraping web pages, accessing APIs, collecting sensor data, or any other means of obtaining the required data. In a modular framework, the data acquisition component is designed to be flexible, allowing different data sources to be easily integrated. Each data source can be represented as a separate module with defined interfaces, making it easier to add, remove, or modify data sources as needed.

2. **Data annotation** [3] is the process of labeling the acquired data with relevant information or metadata. This can involve human annotation, where human annotators manually label the data, or automated annotation techniques, such as using pre-trained machine learning models for labeling [4] [5]. In a modular framework, the data annotation component provides a flexible infrastructure to handle different annotation methods. It allows the integration of multiple annotation tools or algorithms as separate modules, allowing users to choose the most suitable approach for their specific data annotation needs.

The modular framework enables the separation of concerns and promotes reusability of components. It allows data scientists or developers to focus on specific

[1] Users can adapt the framework to their specific data requirements, experiment with different approaches, and efficiently process and label data in a customizable manner.

[2] Users can handle growing data volumes, adapt to changing requirements, and effectively process and label data while maintaining performance and efficiency.

tasks within the data acquisition and annotation pipeline, while also facilitating collaboration and experimentation. With modular components, it becomes easier to iterate, improve, and scale the data acquisition and annotation processes as requirements evolve. Moreover, a modular framework provides opportunities for standardization and interoperability [6] [7]. By defining well-defined interfaces and protocols between modules, it becomes possible to integrate different modules from various sources, making the framework more adaptable and versatile.

So, a modular framework for data acquisition and annotation provides a structured and customizable approach to efficiently gather and label data, making it a valuable tool for various applications, such as machine learning [8], natural language processing [9], computer vision [10], and data analysis [11].

2.1 Related work

In the early stages of the literature search related to our project's subject, it is possible to notice the relevance of the problem we are trying to solve, such as clues to take the first steps and meeting the most valuable tools to develop this modular framework. Therefore, we found some related work, for instance, *HaGRID* [12] that is one of the largest data sets for HGR (Hand Gesture Recognition) systems, this dataset contains 552,992 full High Definition (HD) RGB images (Fig. 2.1), along with *Google ML Kit Pose Detection API*[3] is a lightweight versatile solution for app developers to detect the pose of a subject's body in real time from a continuous video or static image (Movement Recognition using Machine Learning).

Besides these two, there is this *article* about *Hand Gesture Recognition Based on Computer Vision* [13] that provides us a better idea about the concept of Human Computer Interaction (HCI) on the implementation of interactive computational systems. Furthermore, this article gives us some HCI applications, such as "wearable glove-based sensors", a glove that captures hand motion and position using finger and wrist sensor grooves, strain sensors and control board, "camera vision based sensor", cameras that provide contact less communication between person and computer and it has the ability to identify gestures, and there is also this application called "Color-Based Recognition Using Glove Marker", a colorful glove that by following the colors of it enables the camera sensor to detect the exact location of the hand, whether it's the palm or the fingers.

In the same way that all these previous mentioned projects use as support HCI and Machine Learning (ML), it's intended for this modular framework (Fig. 2.2) to use these two technologies once obtaining data from HCI enables us to build and learn more effectively smarter and successful systems and using ML allows the user to feed a computer algorithm an immense amount of data and have the computer analyze and make data-driven recommendations and decisions based on only the input data [14] and for organization and methodology was used FAIR and Data Driven SCRUM (DDS) focused on the improvement of our data science team's communication.

FAIR data [15] refers to content that follows the principles Findable, Accessible, Interoperable and Reusable that pushes towards the goal of standardizing

[3] `https://developers.google.com/ml-kit/vision/pose-detection`

Figure 2.1: HaGRID

data sharing and management. Overall the adoption of these principles improves usability, quality and value of the data.

Quic/sense is a github repository that houses a library of pre-trained models on a variety of datasets which can be configured or used as a starting point for development The repository also includes tools for training, testing, visualizing and analyzing interaction scenario related data. To understand the use of these tools there is a set of demonstrative tutorials.

However, despite all of these features, quic/sense lacks in flexibility, as it restricts users to the set of defined classes it can work with.

Michel Kramer and Ivo Senner proposed a system architecture that relates to ours in the report " A modular software architecture for processing of big geospatial data in the cloud" [16]. In this work the investigators mention the "Processing Services", which is a component of their architecture that is capable of chaining algorithms together as needed thus countering heterogeneity of data (which in their case was geospatial). This module receives a interpreted version of a workflow defined by the user as an input to select which algorithms to chain together and in what order, given this, the component would than accesses a distributed file system to gather the data to be processed. Finally it would store it again in the file system for later access.

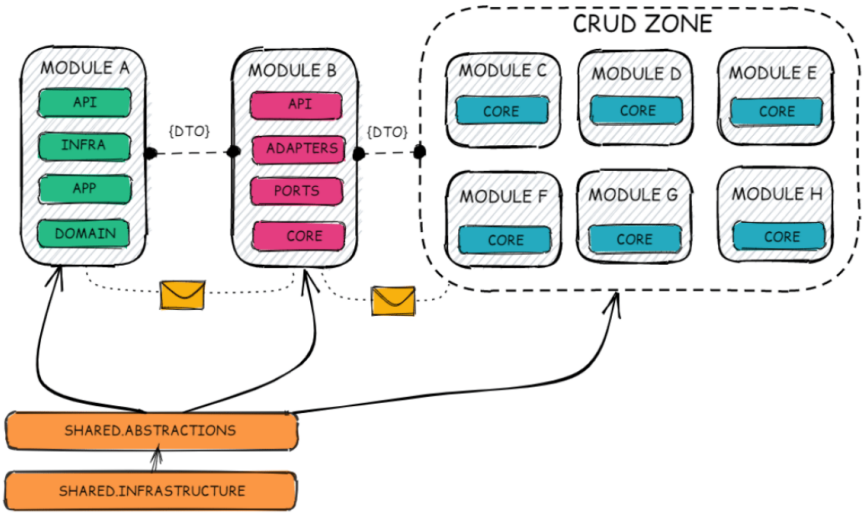

Figure 2.2: Modular Framework representation

Chapter 3

Personas, Scenarios, Use Cases and Requirements

For this chapter, relying on the information we gathered from the previous chapter, the next sections will expose the main reasons leading to the choice of the fictional personas, scenarios and use cases, as well as the *requirements* needed for the development of this project.

3.1 Personas

In the early stages of this project we found it hard to understand the needs, goals, and behavior of the users of a product or service. Therefore, resorted to the creation of three different *Personas* and *use cases*. Personas are fictional characters that represent the different types of users that might use a product or service, and they are typically based on user research and real data about the target audience. Use cases, on the other hand, describe specific scenarios in which a user might interact with a product or service, and they outline the steps that the user would take and the goals that they would be trying to achieve.

By creating these personas and use cases, our team was able to ensure that we are designing and developing products and services that are tailored to the needs and goals of their target users. This can increase user satisfaction, help improve the user experience, and ultimately make the product or service more successful. Additionally, personas and use cases were used to prioritize features and requirements, plan user testing, and communicate design decisions within the team and with stakeholders. Thus, in the next chapter will be presented the *Personas* along side with their use cases that led to the definition of this framework's goals and the requirements and architecture of this framework.

With our project fully developed, we will provide to our users a powerful and dynamic tool, which can assist them in a wide range of roles and scenarios. However, the principle focus is the creation of a new dataset from scratch, completion of an existence dataset and its manipulation providing the users the opportunity to filter the desired data of each dataset.

In the next section, we will describe the motivation of each persona and how our platform will be beneficial to them, as well as its user stories and user cases.

3.1.1 DoTaSet

Biography

DoTaSet is a six year old startup company from Florida that has already developed a few data analysis projects. In their most recent work, they are developing an emotion detection platform through voice and image analysis, for this, they need a dataset that fits their purpose. Knowing how broad this project of theirs is, DoTaSet knows it's impossible to find a public dataset that fits their needs, however developing one from scratch would be consuming too many resources that this startup doesn't possess.

Motivation

DoTaSet is actively seeking a platform that offers enhanced capabilities for acquiring and creating *DataSets*. By leveraging a more comprehensive DataSet, this company aims to handle complex projects and obtaining richer data sources to drive better outcomes and improve their performance in the global market.

Use Case - Complete DataSet

DoTaSet, despite being a startup, the owner has already started developing a few data analisys projects. In his most recent work, he is developing an emotion detection platform through voice and image analisys, for this, they need a dataset that fits their purpose. Knowing how broad this project is DoTaSeT knows its impossible to find a public dataset that fits their needs, however developing one from scratch would be consuming too many resources that this startup doesn't possess. So, after some research, the company comes across our crowd-sourced platform. In this platform he finds a database with some emotion recognition features, that even though it doesn't fully fulfill their needs, it can still upgrade complement it to check the rest of the needed features they were looking for, thus saving him an immense amount of work. After fully complemented, he publishes their dataset to enable future use by third-parties that could find themselves in the same situation. Other users that eventually find their work would also be able to view the author, thus granting more visibility to the startup. These third-parties would then have the ability to choose from whichever iteration of the database development fits them the most, that is, if they conclude DoTaSeT's version has too much data that they wouldn't find useful, they could simply go for the previous iteration and adapt the rest themselves.

3.1.2 Rui Veloso

Biography

As a computer engineering student at University of Porto, Rui Veloso, twenty two years old originally from OPorto, a person who loves music and cinema industry as well as read and write books and music, always ready to help the others with his willing of volunteering. Rui is always trying to improve himself and always striving to learn new things. Therefore, he needs to use a specific dataset of hand movement, in this case, the waving action, so that he can apply

it as a feature on his newest new project he is developing for the subject Human-Computer Artificial Intelligence.

Motivation

Rui wants to implement a waving dataset for his latest project, so he needs a platform where he can acquire his own data, export hand features from it and export the valuable insights into his own project and purposes.

Use Case - Use of dataset

After a long day of brainstorming for his start-up, Rui Veloso decided to pursuit an area that was relatively unfamiliar to him. Intrigued by the subject of Non-Verbal Cues, he recognized the potential value in exploring this field for his project.. After this research for platforms that could satisfy his needs he finally found *MFDAA*. Once he opens up the platform he finds that it offers plenty of public projects that includes various and different *DataSets*. After finding the "ThumbsUp" Project, he decides to explore what it provides. He comes across with a complete DataSet related to "ThumbsUp" and decides to export it and download it to use the features calculated from the *DataSet* into his start-up.

3.2 Requirements

This module will present the requirements considered in the first phase of the project and their succinct description.

3.2.1 Functional Requirements

- Modular Framework must be able to capture relevant data for data acquisition, annotation and ML;
- System should support multiple data acquisition modules, each of which specialized for a specific type of data source (sensors, cameras, microphones...);
- Framework must support the importation of raw data previously gathered;
- The system should provide tools for analyzing and visualizing the collected data;
- Framework should be able to store data and manage it in a scalable and efficient manner;
- Project data can be exported, in a standardized format;
- The system should be capable of extracting features of collected data.

3.2.2 Non-Functional Requirements

- The processing of each request should be done within 5 seconds

- The Web platform should load in 3 seconds when the number of users is less than 1000 and 5 seconds (maximum) when the number is higher than 1000
- The framework should provide a user-friendly dashboard to ease the process of interacting with the data and the system
- The video recording should load within 4 seconds or less
- System should be scalable and able to handle a large volume of data from multiple sources
- System must be multi platform supporting *Windows*, *Linux* and *macOS*;
- Framework should be easy to maintain and extend, with an architecture that allows new data insert and analysis components to be added or removed easily
- Low latency and high performance so data can be collected and processed in real-time
- System should be intuitive and easy to understand with extended effort in documentation
- Easy to integrate and configure for different scenarios
- Have reliable ways to provide accurate algorithms to process data
- Support various data analysis techniques (regression, classification, clustering...)
- System must be secure with strict controls over the access to the data and the system itself

Chapter 4

Conceptualization

This chapter starts by presenting the concept of Modular Framework for Data Acquisition and Annotation design in the two first sections. The chapter ends by explaining the steps taken in order to evaluate the prototype.

4.1 Concept

The concept of Modular Framework for Data Acquisition and Annotation is designed to be versatile and extensible, allowing the users to acquire new data, to annotate and label collected data, assemble components according to their specific needs and store these components, extract valuable insights from users data and also export processed data to effortlessly integrate in personal or other purposes (Fig. 4.1) . The framework offers a flexible and customizable environment that supports a plenty of tools that support a wide range of scenarios of interaction, and it accepts a variety of different data types. This framework promotes data reusability, allowing other users to build upon existing modules and components. This eases efficient development by reducing redundant data and promoting best practices. Besides that, it also enhances maintainability, where the users can update or replace data without impacting the entire framework, where it simplifies troubleshooting and debugging. This Modular Framework was also designed to scalable, where it allows user to add or remove components as their data acquisition and annotation need to take a step-up. New Modules can be seamlessly integrated into the existing framework, expanding *datasets* or changing users requirements.

4.2 UI Design

4.2.1 Conceptual Design

Following and recognizing, as a team, the goals and requirements of this framework, with the help of a paper, a pencil, the following prototype for each page of our application was initially created. As we will see in the next chapter, there were still some design changes and adjustments made to the positioning of certain elements in order to make it more accessible and visually appealing to users.

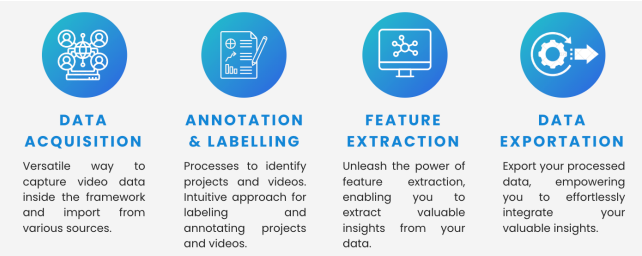

Figure 4.1: MFDAA - Modules

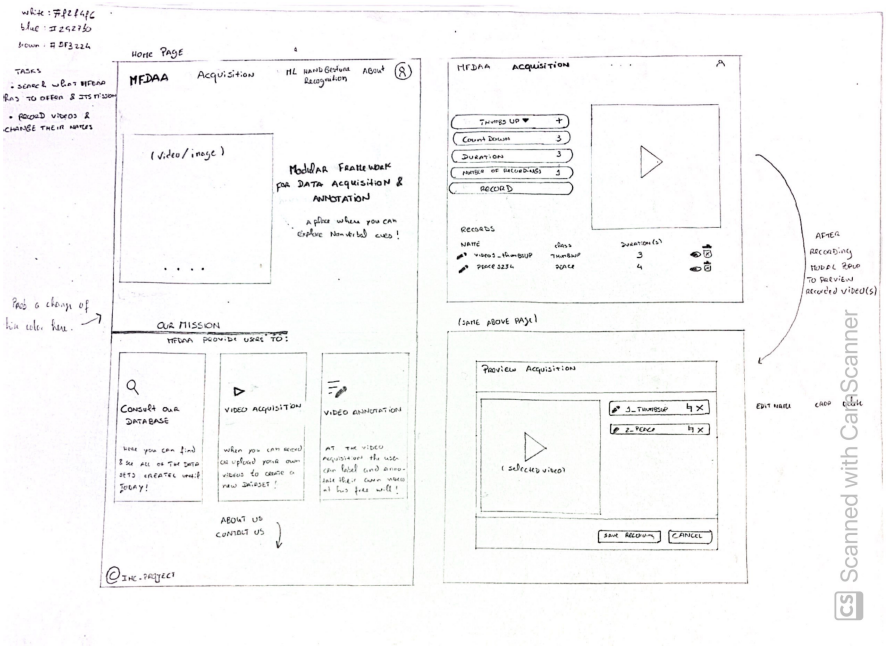

Figure 4.2: *MFDAA* - Home & Acquisition Pages

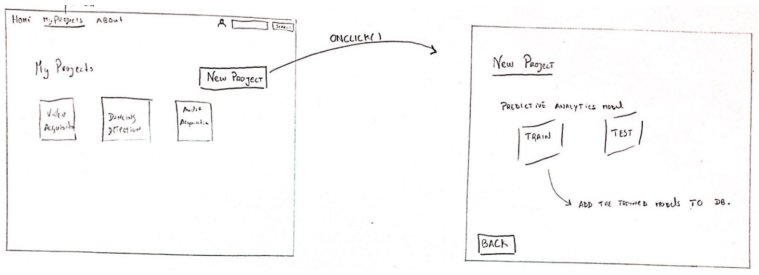

Figure 4.3: *MFDAA* - My Projects Page - Creation of the project

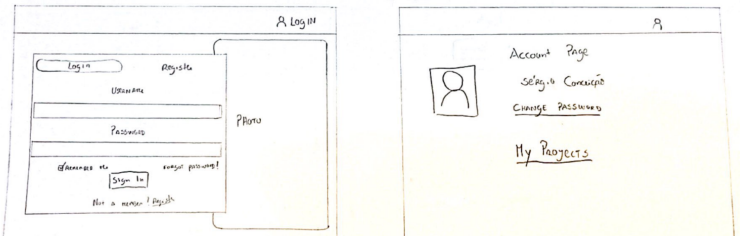

Figure 4.4: *MFDAA* - Profile Page

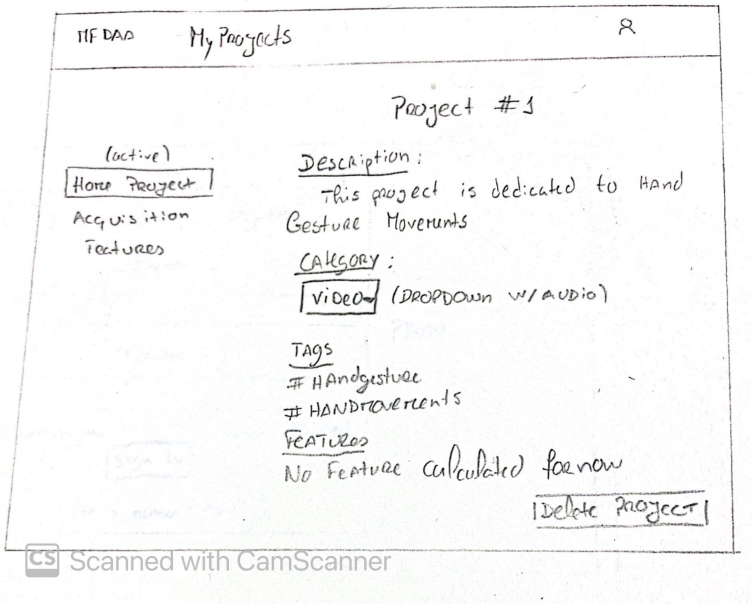

Figure 4.5: *MFDAA* - Project Home Page

These previous prototypes were the base and, most definitely, were the best choice to begin coding the FrontEnd of the WebSite and also to auto-evaluate the Workflow of the WebSite. Afterwards this evaluation, as team discussion, it was noticeable some problems related to the workflow such as the creation of the new Project redirecting directly to the acquisition. With the help of this project's supervisors, we found that the most reliable way to fix this problem was to add a second *navbar* alongside with a project home page, where the user can see all the information associated to the project, description, tags, delete the project. To turn this possible, the first step was to draw the possible prototype for this Workflow as we can see in the Fig. 4.5

To finalize, one of the goals of this project is to extract features from the recorded videos. It was deemed necessary to undertake the task of drawing the Feature Page interface in order to achieve the desired outcome (Fig. 4.6).

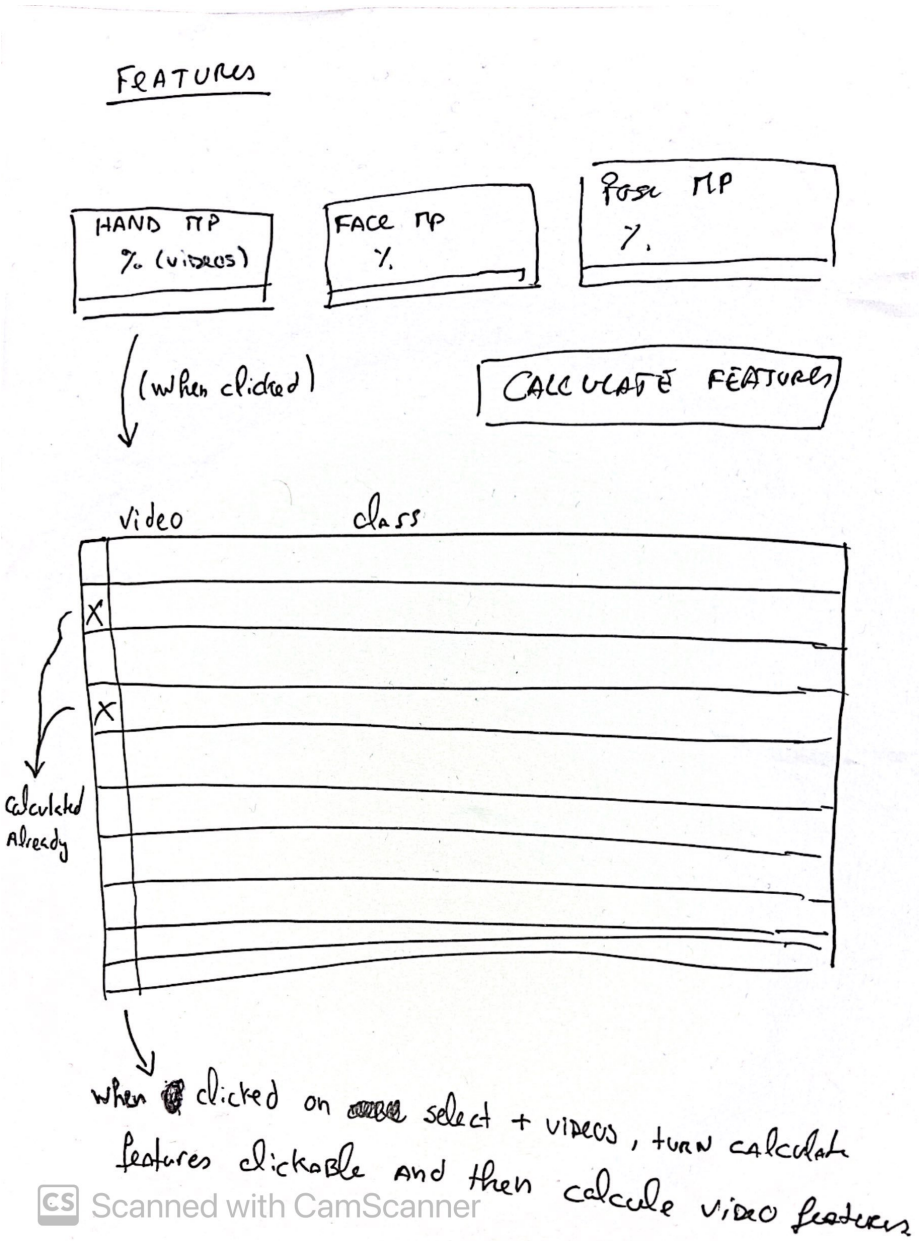

Figure 4.6: *MFDAA* - Features Page

4.3 Validation

In order to evaluate the *MMFDAAFDAA* prototype, the following steps were taken:

1. The search for users to test our Low-Fidelity Mock-Up.
2. A brief explanation was given to the user about what our framework consists of.
3. The user was given a consent form to be signed after careful reading.
4. The tasks that the user must perform were announced in order to evaluate their difficulty of execution.
5. It was necessary to use a timer so that the app's founder could determine the execution time of each task.
6. Another team member counted the clicks necessary to complete each task and evaluated the ease of each task on a scale of 0-5.
7. Finally, each user was presented with the System Usability Scale (SUS) questionnaire.

After defining the topics for *MFDAA* prototype evaluation. Overall, the prototype is being assessed to make sure that it is user-friendly, effective, and fits the needs of its intended users. We may get insightful input from users and decide on the design and functioning of the application by developing a consent form and setting up user tasks.

4.4 Conclusions

After having designed and evaluated the low-fidelity paper prototype , it becomes evident that this framework is ready to turn it into a functional platform, with only minor adjustments highlighted by the evaluation participants.

Chapter 5

Development of MFDAA

This Modular Framework development followed an **iterative** and **user-centered** methodology, focused on user input at every stage, throughout the process. Several iterations, known as "weekly sprints", were conducted to gather feedback from users during platform testing, as well as to receive evaluations from supervisors. These sprints with the supervisors played a crucial role in defining and comprehending the goals and tasks for each week, while also providing a validation of progress made during that time. For efficient team management, we adopted the SCRUM methodology ([17]) , which enabled us to effectively plan, organize, and track project progress. It was used *JIRA Software* to annotate notes and tasks from the weekly meetings. This allowed us to maintain clear communication, track project milestones, and ensure that everyone was aligned with project goals.

For *MFDAA*'s interface validation, an evaluation with both external and internal participant, the details of this evaluation will be described later in this report. These practices not only facilitated effective collaboration within the team but also ensured that user perspectives were considered in the design and improvements of this MFDAA platform. This methodology is described in the Fig. 5.1.

5.1 Architecture

Projects of this caliber face various issues when attempting to make their platform something that can be expanded with the addition of new forms of acquisition and annotation methods. Therefore, our team started early designs of the system's architecture with this problem in mind, and after some attempts we arrived at the following solution.

Our project implements a scalable modular architecture, characterized by the integration of different and independent modules, which guarantees its modularity. This architectural approach worked to significantly enhance the development process and improved the overall performance of our system.

Thus, this architecture is divided into various independent modules, each with its specific functionality. This implementation aims to provide better flexibility and versatility within our framework, making it more open to the addition of new features.

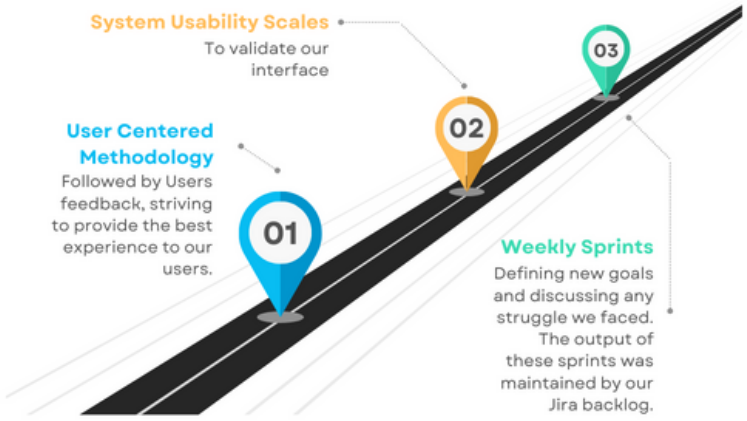

Figure 5.1: Methodology throughout the development of this project

With this design we aim to have a better division of responsibilities through the system as well as easing future implementations of new data types, acquisition sources, annotation processes and visualisation modules, thus embracing scalability. Regarding the former, work assignment between members was also standardized, charging each person with a task correspondent to the module they were specializing in.

- **The Dashboard** is for user interaction only, with the remaining components of the system being fully capable of functioning without it. This module aims to ease the comprehension of the system and allow clients to do tasks such as creating new projects, expanding datasets, visualizing and analyzing data, recording and editing content among other features such as browsing.

- **The API** is the middleware of the system and controls the communication between the main components (database, data treatment modules and dashboard). It controls how data is stored and managed by the database while also executing all the alterations users may do. This API also features the capability of uploading and downloading projects as well as exporting videos from your laptop to the platform.

  Finally it also loads all the informations required by each page and responds to every user action.

- **The Database** is a non-relational storage unity to provide a secure and efficient flow of data between passive and active services. It uses mongoDB's GridFS to store videos while also returning a reference to retrieve them that is stored among the remaining information of each project.

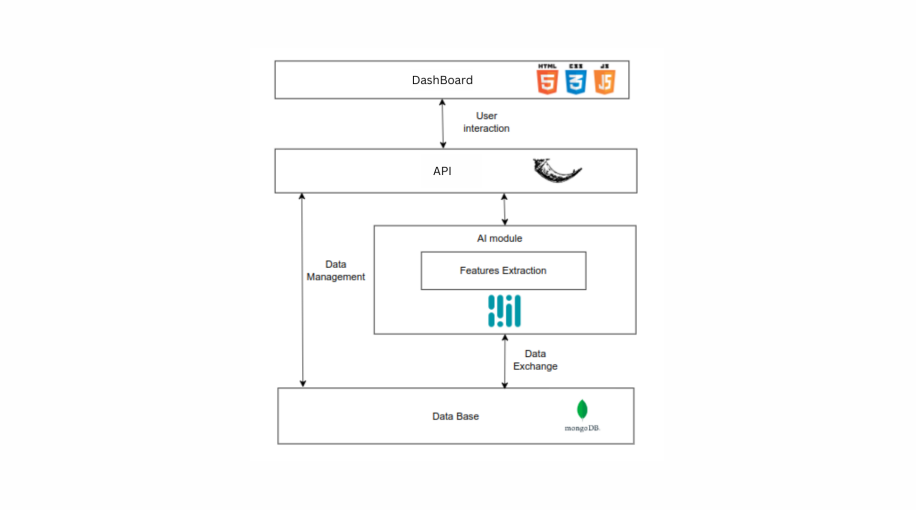

Figure 5.2: Framework Architecture

- **The ML Pipeline** is the main processing feature of the architecture that makes use of Mediapipe's open source libraries to analyze the content it's fed, allowing our framework to display visualization capacities for user data such as the joints in their hands and relevant points in their face.

After having the modular framework architecture for *MFDAA* defined, it was possible to start developing to match the desired goals, according to *MFDAA* requirements previously formulated.

5.2 Technologies

This section will present the technologies chosen for the execution of this project, by making a succinct description of their purpose and the main reasons leading to their choice, likewise their main advantages and disadvantages.

5.2.1 FrontEnd

Was decided to develop a **Web Platform** since there are several reasons why it may be beneficial to develop a web platform for this project. It can provide access to a wide audience, allowing users to access the platform from any device with an internet connection and it can also be easier to maintain and update than traditional software applications, since the updates can be made on the server and do not require users to install anything on their own devices. Additionally, web platforms can be more flexible and customizable than other types of platforms, allowing developers to create unique experiences for users For the *FrontEnd* was chosen **HTML/CSS**. But the question is,

Why choose HTML/CSS when you have PHP, React, Flutter, or others?

That's because HTML/CSS make it possible to create a clean and better architecture that is friendly to testing and another advantage is that this tech-

nology allows data changes to be made without reloading the page and another important fact is that is a Cross-platform which refers to the ability of a piece of software or an application to run on multiple platforms, or operating systems.

5.2.2 BackEnd

For the *FrontEnd* was chosen **Python**[1] , an high-level programming language. Its design philosophy emphasizes code readability with the use of significant indentation, since Python allows developing applications that are clear and simple as well as it is easy to learn and read. At the same time, it is flexible and easy to scale, which means it has various purposes.

why choose Python when you have Java, JavaScript, Ruby, C#, and PHP? Any of these *backend* programming languages are suitable for this development, however, Python is the most favoured *backend* web development technology as it is appropriate for establish the connection with the latest technologies like **Internet of things (IoT)** and **ML** in web applications, and in addition, it provides a more suitable connection with the *frontend* and *API's*.

For the database we opted to go with *MongoDB* [18] as it fits the non-relational requirement of the project easing the work with unstructured data. Besides, it also is a very flexible and scalable tool that fits well with various frameworks, including ours, and data types while also providing various operators to sort and filter content.

In the API we are opting to use *flask* as it matches the remaining backend language and is a flexible and simple tool with a big community to offer support. Other considered options were *Django*, *NodeJS* and *ExpressJS*, these have not yet been excluded and may be chosen in a more advanced phase of development.

5.3 Overview of the System

Our project implements a scalable modular architecture, characterized by the integration of different and independent modules, which guarantees its modularity. This section provides a high-level overview of the system's architecture and its major components, outlining how they interact with each other.

The system contains several key modules that work balanced to achieve the desired goals. This modular platform design grant easy integration and expansion of individual modules as needed. Each module within the system plays a crucial role in fulfilling specific tasks to the overall platform functionality. By breaking down the system into distinct modules, we enable efficient development, maintenance, and scalability.

5.3.1 Data Acquisition

This platform provides users a versatile, flexible and extensible data acquisition module that supports multiple video recordings or importation from diverse sources (Fig. 5.3). This acquired data can be leveraged in future stages, particularly feature extraction, to support various analytical processes and derive valuable insights (subsec. 5.3.3). Notably, the processing of this data occurs

[1] `https://www.python.org/?!`

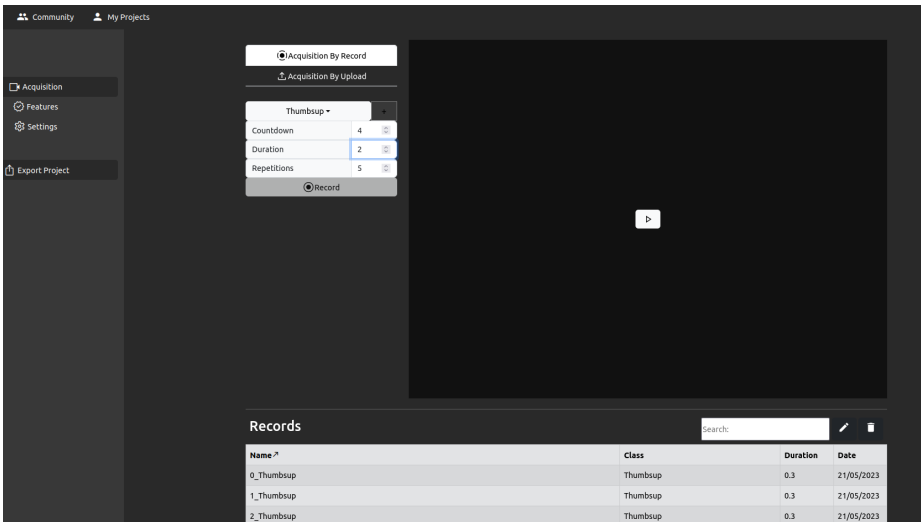

Figure 5.3: Data Acquisition for Feature Extraction Preparation

seamlessly behind the scenes, headlining the user friendly approach to acquire data.

5.3.2 Annotation and labeling

Within our platform, users have the capability to perform comprehensive data annotation and labeling from diverse methods. Including, annotating and labeling videos with classes such as "ThumbsUp", "ThumbsDown," and "Peace", or assigning unique identifiers to ease data organization and management. There's also the possibility to add tags to the video (Ex: #hand; #thumbsUp) to better identify each project (Fig. 5.4).

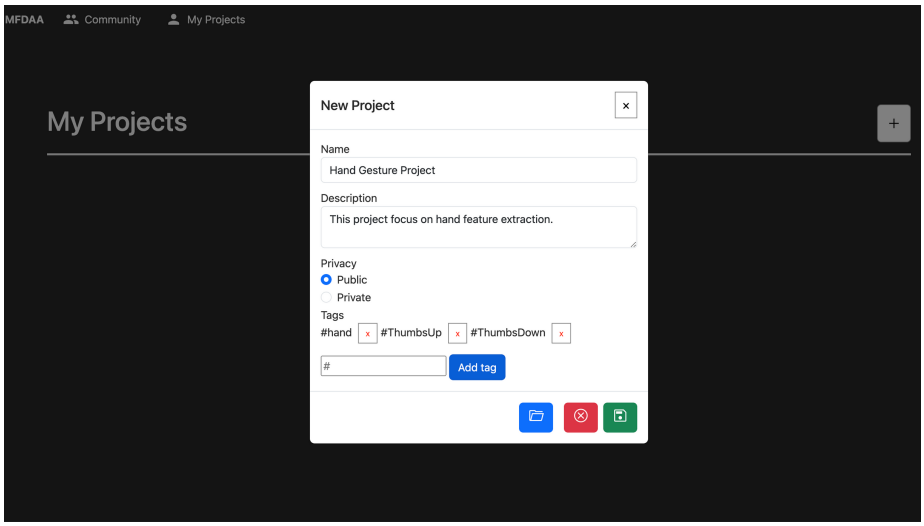

Figure 5.4: Annotation and Labelling processes to identify projects and videos

5.3.3 Feature Extraction

Our modular platform incorporates a powerful feature extraction module that can automatically extract valuable insights. With the help of *MediaPipe* [19], "a Framework for building machine learning pipelines for processing time-series data like video, audio, that provides a suite of libraries and tools for you to quickly apply Artificial Intelligence (AI) and ML techniques in applications" - [20] it was possible, due to its powerful solution for hand gesture, face, pose and holistic recognition, to extract the needed features from each selected video from Data Acquisition (Fig. 5.5).

5.3.4 Data Importation and Exportation

In this module, the system offers users the opportunity to handily export their processed data, empowering them to effortlessly export data, along side with associated features, to either integrate valuable insights into other purposes or to create a new project on our environment platform (Fig. 5.6). Furthermore, this platform, also provides users to import data from various sources for data acquisition module (5.3.1).

5.4 User Evaluation

To properly evaluate *MFDAA*'s user interface, just like it was done to evaluate the low-fidelity Mock-Up prototype in paper, we followed the same steps previously referred on sec. 4.3. At the user testing, each user faced a series of tasks that were crucial so that *MFDAA*'s members could provide their final and overall opinion of our system outside team's eyes. In this way, the tasks served to assess user difficulties, both in terms of usability and functionality of the app, which led us to use the System Usability Scale (SUS) questionnaire.

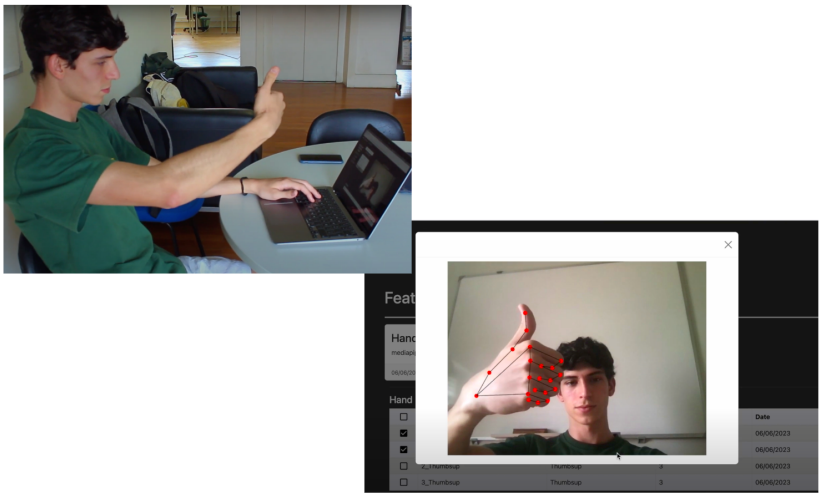

Figure 5.5: Feature Extraction: Insights from Acquired Data using MediaPipe (from: `https://www.youtube.com/watch?v=Qkeur-eGGHo`).

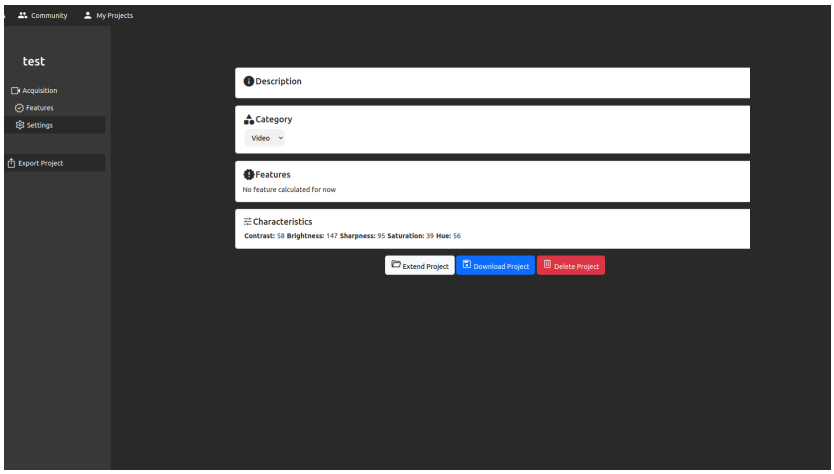

Figure 5.6: Data Exportation: Integration and Export of Collected Data

Once users become familiar with the features and components of our platform, it is introduced to the evaluator the following **tasks:**

1. The user must explore in the home page of the WebSite what MFDAA has to offer;
2. User must create a new project named "Hand Gesture Project";
3. Currently, user it is in the Project Home Page, must go to Acquisition Page and predict where will the future recorded videos will be displayed after being sent to the DataBase;
4. After predicting, record three videos with a countdown and video duration of two seconds and choose "thumbs-Up" class;
5. After recording the videos, change video names to "ThumbsUpVideo#", where the "#" corresponds to the number of the video;
6. Following the videos user just recorded, user should be able to calculate Hand Gesture Recognition. features of the first video;
7. Same task as before but now for Face Mesh Feature and only select two of the two videos;
8. Export your project and data;
9. If user no longer wishes to continue or to cancel its changes, he should be able to delete current project.

System Usability Scale (SUS) This is a standard questionnaire that measures the overall usability of a system. Please select the answer that best expresses how you feel about each statement after using the website today.

5.4.1 Final Platform Evaluation

Having completed the development of this modular framework on 10/05/2023, our assessment was conducted the the following day, where 8 candidates were contacted to evaluate our system. On 20/05/2023, another evaluation was performed (corrected according to the first 8 candidates feedback and considerations).

Participants Profile

To perform a consistent evaluation, the same eight participants were considered in both evaluations. Among the candidates, five of them are currently DETI students, ranging in age from eighteen to twenty three years old, two of them are researchers between forty five years old to sixty and finally, the rest of the participants are students outside IT area ranging in age from nineteen to twenty five years old.

Evaluation Results

In order to establish a difference from the feedback of the first and the second evaluation, the team performed two graphics with the same tasks, obtaining the results presented on Fig. 5.7 and Fig. 5.8, representing respectively, first and second evaluation.

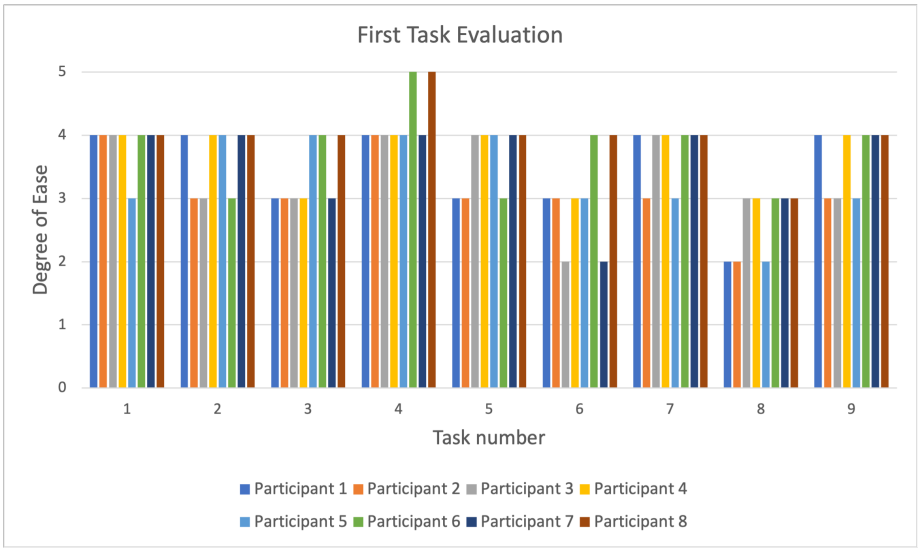

Figure 5.7: First User Evaluation

5.4.2 Evaluation Discussion

Based on the results presented in subsection 5.4.1 and from the feedback received from users on the first evaluation it is evident that the team needed to make a few significant changes on our platform. This change mainly focused on frontend modifications aimed at simplifying and enhancing the user experience, making the framework more user-friendly and intuitive.

The user feedback revealed areas where consumers experienced issues or had suggestions for improvement. Taking this input into account, the team realized the significance of optimizing the frontend components to better correspond with user expectations and preferences. We hoped that by implementing these adjustments, we might provide a more smooth and engaging experience for our users.

Furthermore, *MFDAA* team prioritized the implementation of a more user-friendly platform, which led to this second evaluation results (Fig. 5.8. In light of these results, a noticeable and substantial improvement can be observed when comparing to the first evaluation (Fig. 5.7). These results provide valuable insights, revealing that it was not as intuitive and enjoyable for users as initially anticipated. Nevertheless, thanks to this valuable user evaluation, the team could perform a better user experience to those who intend to use *MFDAA*.

This website[2] was consulted regarding the System Usability Score (SUS) to measure, according to the evaluation results, if our app is suitable. From the results previously provided on the subssec. 5.4.1 it is possible to understand that the SUS score for *MFDAA* is higher than 68% indicated, therefore it can be concluded that our app is advisable and usable for any user.

"Scores below 68 point to issues with the design that need to be

[2] `https://xd.adobe.com/ideas/process/user-testing/sus-system-usability-scale-ux/`

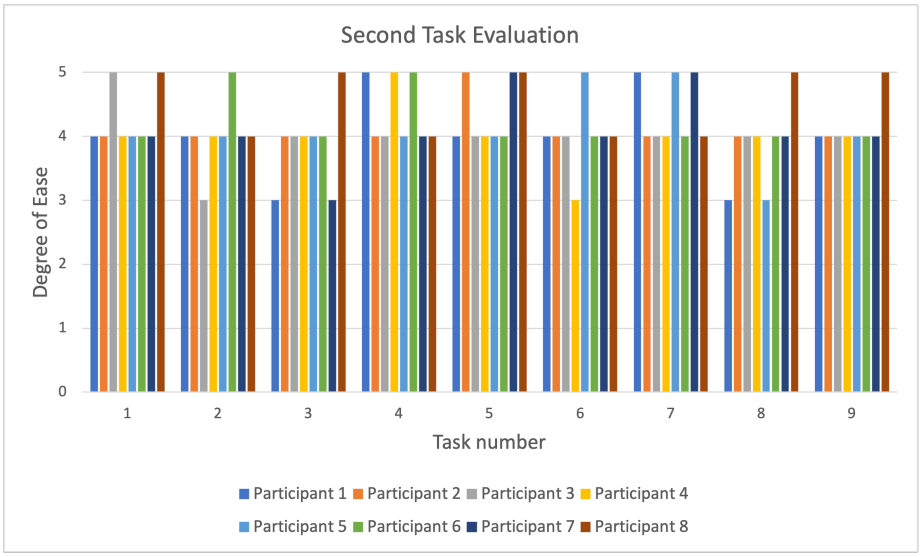

Figure 5.8: Second User Evaluation

researched and resolved, while scores higher than 68 indicate the need for minor improvements to the design."

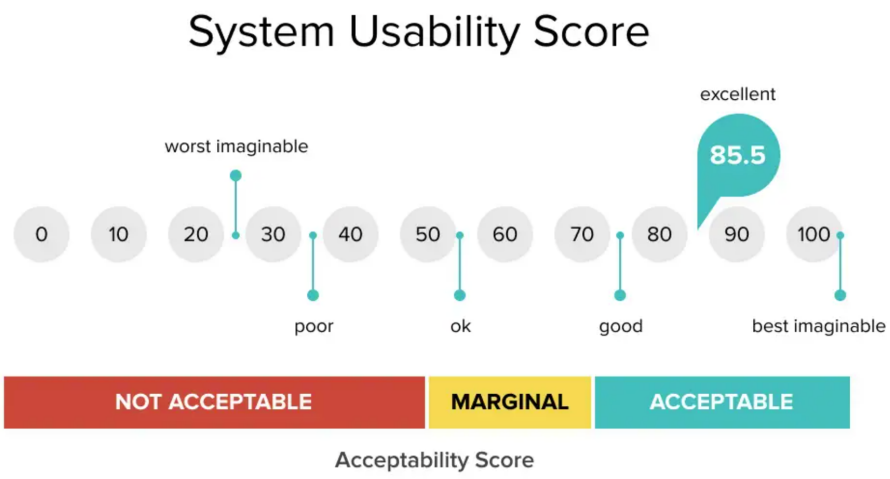

Figure 5.9: System Usability Score (SUS)

Chapter 6

Conclusion and Future Work

This last chapter presents the conclusions drawn from the study, outlines future research opportunities, provides insights into the team work distribution , and highlights the individual contributions of team members.

6.1 Conclusions

In conclusion, the Modular Framework for Data Acquisition and Annotation to Support Interaction Scenarios will be a versatile and powerful tool to collect and analyze data in a wide range of interaction scenarios, thus building *datasets* from it that can be used for various purposes in investigation and development. Its modular design combined with a user-friendly, well documented dashboard with allows clients to easily customize the framework to their requirements and gain insights to improve decision-making based on accurate up-to-date content.

This framework differs from related works, as it is intended to be able to support analysis and processing capabilities for various types of interaction scenario acquired data by customizing the various modules.

Ultimately, we have successfully implemented a functional platform that fulfills the initial proposed requirements[1]. Nonetheless, this modular framework lays a solid foundation and demonstrates substantial potential for future advancements and continuous improvements .

6.2 Future Work

The current framework has laid a solid foundation for data acquisition and annotation. However, there are several areas where further development and improvements can be made to enhance its maintainability and extend its functionality. With the existing implementations, the platform can be extended to support the processing of other types of data, such as audio and radar data, missing only an interface that could interact and process each type of data, where right now it can only process and interact with video data type. When it is implemented, it will enable users to extract valuable insights from diverse sources of information.

[1] `https://www.youtube.com/watch?v=Qkeur-eGGHo`

MFDAA framework, even though it has components to extract features with high accuracy, the efficiency it is not the greatest, where sometimes when extracting more complex features from a big amount of data, it takes a long time to extract the feature. This could be improved by implementing state-of-the-art algorithms for anomaly detection and pattern recognition, or even by expanding the range of available analysis tools.

Deploying the framework on a server infrastructure and allowing users to access it from a distance via web interfaces is a significant step toward improving the framework's scalability and accessibility, after being implemented. This would make it simpler for other users to collaborate and share data and analysis findings.

Finally, the framework's user experience and data management capabilities can be improved by giving users a designated personal zone. Where each user could have his private dataset, without sharing it to others user. By focusing on these potential future work areas, the framework can develop into a more complete and adaptable platform that can support diverse data types, enhance data analytic capabilities, and offer a seamless user experience across various devices and deployment circumstances.

6.2.1 Team Distribution(%)

- Eduardo Fernandes: 27%
- Guilherme Claro: 8%
- João Afonso Ferreira: 27%
- Pedro Durval: 27%
- Tiago Mostardinha: 11%

6.2.2 Contributions

- Eduardo Fernandes: Planning and Strategy, Product design creation and user testing of that design. Implementation of the frontend and its interaction with the backend. Mediapipe features extraction and preview.
- Guilherme Claro: Some Frontend design, development of frontend features, User evaluation and proper changes to the platform, some communication between backend and frontend. Other varied contributions
- João Afonso Ferreira: Planning and strategy, team and project management, Ddocumentation and report elaboration, Frontend design and all design-related aspects, and facilitating communication between the backend and frontend.
- Pedro Durval: Planning and Strategy, development of the main backend features, all system interactions with the database along with its processing methods and methods of communication between the various modules.
- Tiago Mostardinha: Processing and characteristics Extraction of video-related data, including frontend and backend methods, methods for Filtering and Searching datasets, techniques for adding new data to the platform.

Bibliography

[1] J. Fuhrer and J. Little, "Technology and growth: An overview", *Conference Series ; [Proceedings]*, pp. 1–32, Feb. 1996.

[2] Simplilearn, *Data acquisition: Definition, process, and methods*, https://www.simplilearn.com/data-acquisition-article, Accessed: [05/2023].

[3] D. AI, *Machine learning essentials: What is data annotation?*, https://www.defined.ai/blog/machine-learning-essentials-what-is-data-annotation/, Accessed: [05/2023].

[4] IBM, *IBM Data Labeling*, https://www.ibm.com/topics/data-labeling, Accessed: [2023].

[5] Cameralyze, *What is data annotation? the basics*, https://www.cameralyze.co/blog/what-is-data-annotation-the-basics, Accessed: [05/2023].

[6] M. L. Zeng and L. M. Chan, "Metadata interoperability and standardization-a study of methodology, part ii", *D-Lib Magazine*, vol. 12, no. 6, pp. 1082–9873, 2006.

[7] J. A. Mykkänen and M. P. Tuomainen, "An evaluation and selection framework for interoperability standards", *Information and Software Technology*, vol. 50, no. 3, pp. 176–197, 2008.

[8] P. A. Flach, "On the state of the art in machine learning: A personal review", *Artificial Intelligence*, vol. 131, no. 1, pp. 199–222, 2001, ISSN: 0004-3702. DOI: https://doi.org/10.1016/S0004-3702(01)00125-4. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0004370201001254.

[9] D. Khurana, A. Koli, K. Khatter, and et al., "Natural language processing: State of the art, current trends and challenges", *Multimedia Tools and Applications*, vol. 82, pp. 3713–3744, 2023. DOI: 10.1007/s11042-022-13428-4. [Online]. Available: https://doi.org/10.1007/s11042-022-13428-4.

[10] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN", in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

[11] D. Patel, R. Patel, and S. Patel, "A comprehensive study on data annotation for machine learning: Challenges, techniques, and applications", *Artificial Intelligence Review*, vol. 53, no. 8, pp. 4757–4788, 2019. DOI: 10.1007/s10462-019-09685-9. [Online]. Available: https://link.springer.com/article/10.1007/s10462-019-09685-9.

[12] K. K. Alexander Kapitanov Andrew Makhlyarchuk, "Hagrid - hand gesture recognition image dataset", *Computer Science, Computer Vision and Pattern Recognition*, p. 11, Feb. 2022.

[13] M. Oudah, A. Al-Naji, and J. Chahl, "Hand gesture recognition based on computer vision: A review of techniques", *journal of Imaging*, vol. 6, no. 8, p. 73, 2020.

[14] NetApp, "Artificial intelligence, machine learning", 2022.

[15] OpenAIRE, *How to make your data fair*, `https://www.openaire.eu/how-to-make-your-data-fair`, Accessed: [4/2023].

[16] M. Krämer and I. Senner, "A modular software architecture for processing of big geospatial data in the cloud", *Computers & Graphics*, vol. 49, pp. 69–81, 2015.

[17] TechTarget, *Scrum*, `https://www.techtarget.com/searchsoftwarequality/definition/Scrum`, Accessed: [04/2023], 2021.

[18] MongoDB, *Mongodb*, `https://www.mongodb.com/`, Accessed: [04/2023].

[19] Google Developers, *Mediapipe*, `https://developers.google.com/mediapipe/`, Accessed: [04/2023], 2022.

[20] Learn OpenCV, *Introduction to mediapipe*, `https://learnopencv.com/introduction-to-mediapipe/`, Accessed: [04/2023].