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

Abstract

This report presents a modular data acquisition framework to support interaction scenarios that aims to provide a flexible, customized and unified platform for collecting and cataloguing data from various input sources. 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. So, with this proposal we attempt to create a platform that saves clients from the hard-work that is creating a new dataset from scratch by providing a common ground where they can browse and expand the effort that has been submitted previously, spreading a "work smarter not harder" mindset. In the real world the improvement of interaction methods with modern technology have the goal of making life more comfortable, accessible and overall intuitive to the most amount of people possible by implementing accurate and precise identification of user command inputs that still lack in quality such as gesture recognition, movement recognition, voice analysis. The framework consists of several modules whose objective is to reduce content heterogeneity with their integration and configuration capabilities in order to find the best fitting solutions to support the specific needs of different use cases. At the heart of the system is the data insert module that takes a scalable design and is responsible for all treatment of data. It begins with acquisition from various sources such as sensors, camera and microphones, following this the content goes through an annotation process where it is catalogued and info is abstracted from it, finally data can also be filtered according to the specific requirements of interaction scenarios. A data storage unit is responsible for organizing and storing the collected content in a non-relational database. This module is only intended to communicate with the data insert module and in a secure and efficient manner. In addition to the previous components, there is also a Machine Learning (ML) pipeline in this architecture. This module's function is to optimally create adequate algorithms that can process received data and output models to be integrated in other systems. Connecting the pipeline, the data management and the user dashboard is an API that works as Middleware and communication ground for the system. Overall, the proposed modular framework is projected to be a comprehensive and flexible platform for collecting, storing, analyzing and processing data from various sources. It is designed to support a wide range of interaction scenarios and is easily customizable and integrated with other systems.

Acronyms

HCI Human Computer Interaction

ML Machine Learning

HD High Definition

DDS Data Driven SCRUM

UI User Interface

IoT Internet of things

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

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, machine learning and more. These technologies have made our lives more convenient and efficient in many ways, but they have also raised important questions about privacy, security, and the potential impact on jobs and society.So we thought in making ways of interaction more natural and intuitive of interacting with this technology such as non-verbal cues and gestures/poses that are optimal but developing them can be a bit of a challenge. With the development of this ways of interaction he can be made by the use of an approach with machine learning.The recent approach's are obliged to build the entire pipeline for every case, that result in small databases and they cause problems in it's expansion namely for derive from conditions of acquisition very specific. It would be interesting to have a more systematic way to be able to use this approach in a way to build databases more systematically and be willing to explore new ways to interact.

The modular data acquisition framework described in this report is designed to facilitate the making of data acquisition systems for interaction scenarios. This system involves the acquisition of data from various sources, as well as the real-time processing and analysis of that data to enable interactive behavior.

There are many reasons why it is useful to create a modular data acquisition framework. Some possible motivations for doing so include:

- Flexibility: A modular framework allows to add or remove different components of the system without affecting the other parts.

- Re-usability: By breaking a system into modular components, you can reuse those components in other applications or projects.

- Maintainability: Modular frameworks are typically easy to maintain because each component can be updated or replaced independently. This can reduce the overall complexity of the system and make it easier to troubleshoot and fix issues.

- Collaboration: Modular frameworks can make it easier for teams of developers to work on different parts of the system simultaneously. This is very efficient to upgrade the productivity

Our modular data acquisition framework can be used in a wide range of applications such as research and development and consumer-facing systems. In research and development, it can be used to study and evaluate different interaction scenarios and algorithms. And in consumer-facing systems, it can be used to enable interactive and personalized experiences for users.

There are several challenges that may be found when implementing a modular data acquisition framework. Some of these challenges include:

- Making the modules compatible between themselves:This means that the modules must be able to communicate and exchange data with each other in a seamless manner

- Managing the complexity of the system: As the number of modules in a data acquisition framework increases, the overall complexity of the system also increases. This will make it difficult to manage and maintain the system.

- Supporting real-time data processing: The data that is collected by a data acquisition framework needs to be processed in real-time in order to be useful. This can be a challenge, especially if the framework is handling large volumes of data or if the modules are distributed across a wide geographic area.

Clarifying the **structure** of this report, firstly, we are going to address Background, Related Work at the section 2 which is where is present the work we searched for and we adapt them to our own needs. Secondly, will be displayed the conceptual Modelling and on the chapter State of the Art where all the requirements, the creation fictional characters to define goals and understanding the needs for this project, the technologies that were chosen for the execution of this project, by making a succinct description of their purpose and the main reasons leading to their choice as well as the architecture since a well-designed architecture can help ensure that the system is scalable, maintainable, and extensible and also it can help the stakeholders understand and agree on the goals and direction of the project (sections 3 & 4). At the section 5 where it is presented the implementation of the features needed to obtain a working platform.

Chapter 2

Background and 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*[1] 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, along with *Google ML Kit Pose Detection API*[2] 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* [1] 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 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 [2] and for organization and methodology was used FAIR and Data Driven SCRUM (DDS) focused on the improvement of our data science team's communication.

[1] `https://cutt.ly/S1WXm1Z`

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

FAIR data [3] refers to content that follows the principles Findable, Accessible, Interoperable and Reusable that pushes towards the goal of standardizing 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" [4]. 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.

Chapter 3

Conceptual Modelling / Theoretical Analysis

For this project, relying on the information we gathered from related work (2), the next sections will expose the main reasons leading to the choice of *technologies*, as well as the *requirements* needed for the development of this project and the design of the *framework's architecture.*

In the early stages of this project we found it hard to understand the 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.

Chapter 4

State of the Art

4.1 Personas/Actors and Use Cases

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 sub-chapter, 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.

1. **Sérgio Paulo creates a new dataset**

Title: Creation of a gesture dataset (*USE CASE*)
The researcher accesses our system/platform to create their models, after opening the project he selects the gesture features and set up some base classes of gestures(thumbs up, thumbs down, point up, point down). Then he proceeds to select thumbs up and record ten examples of that gesture, doing the same for the remaining classes. The researcher can now test the accuracy of his new dataset.

2. **DoTaSet's Owner (Start-Up) completes dataset**

Title: Complete dataset (*USE CASE*)

DoTaSet, despite being a startup, the owner has already started developing a few data analisys projects.

In his most recent work, he is developing a 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.

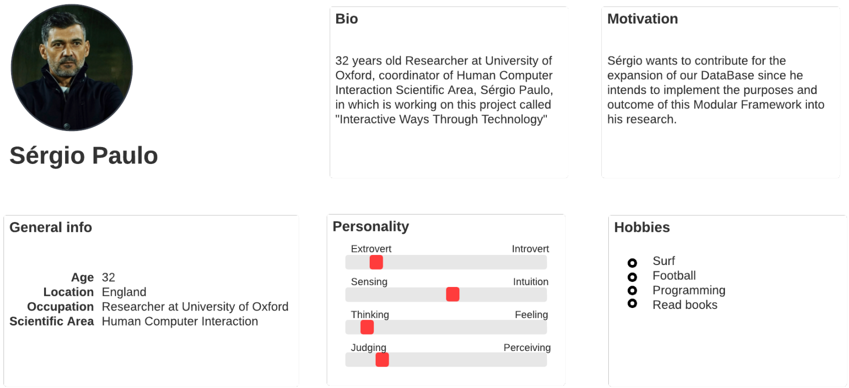
These third-parties would then have the ability to choose from whichever iteration of the database development fits them the most, that is, if they conclude DoTaSeT's version has too much data that they wouldn't find useful, they could simply go for the previous iteration and adapt the rest themselves.

3. **Rui Veloso uses dataset**

Title: Use of dataset (*USE CASE*)

After a long day of brainstorming for his start-up, Rui Veloso, decided to approach an area that is not as developed as other technologies, which have a highly interest for a variety of enterprises, such as google, apple, governments and among others. Rui chose to do a project, that will focus on giving functionalities to gestures, poses and many other queues.

However the main asset to have in order to produce a reliable application with those features is having a considerable dataset with accurate data to make an efficiency detection of gesture and poses. Fortunately, our framework, will provide to the users, such as Rui, the opportunity to download an open-source dataset to keep on improving his work.



Sérgio Paulo

Bio

32 years old Researcher at University of Oxford, coordinator of Human Computer Interaction Scientific Area, Sérgio Paulo, in which is working on this project called "Interactive Ways Through Technology"

Motivation

Sérgio wants to contribute for the expansion of our DataBase since he intends to implement the purposes and outcome of this Modular Framework into his research.

General info

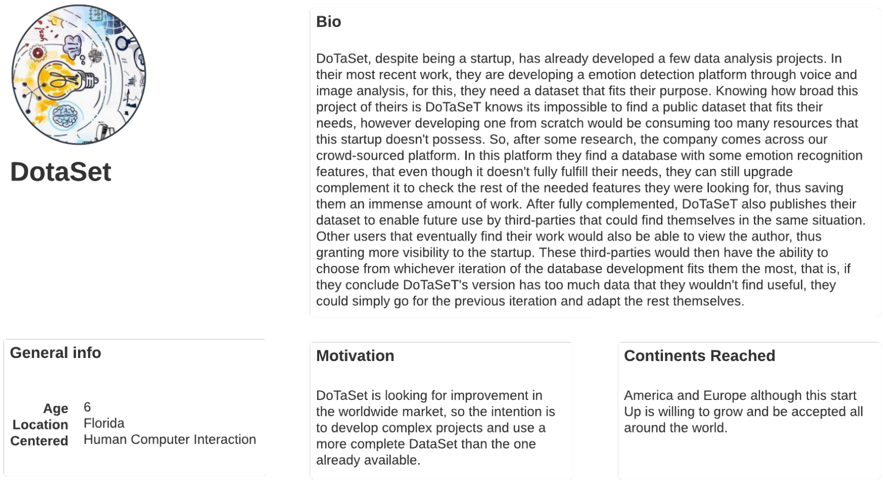
Age	32
Location	England
Occupation	Researcher at University of Oxford
Scientific Area	Human Computer Interaction

Personality

Hobbies

- Surf
- Football
- Programming
- Read books

Figure 4.1: Sérgio Paulo - Persona



DotaSet

Bio

DoTaSet, despite being a startup, has already developed a few data analysis projects. In their most recent work, they are developing a 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 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 they find a database with some emotion recognition features, that even though it doesn't fully fulfill their needs, they can still upgrade complement it to check the rest of the needed features they were looking for, thus saving them an immense amount of work. After fully complemented, DoTaSeT also 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.

General info

Age	6
Location	Florida
Centered	Human Computer Interaction

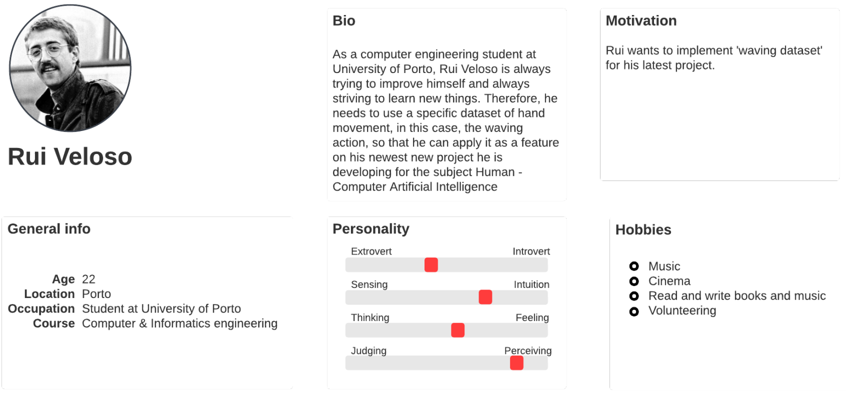
Motivation

DoTaSet is looking for improvement in the worldwide market, so the intention is to develop complex projects and use a more complete DataSet than the one already available.

Continents Reached

America and Europe although this start Up is willing to grow and be accepted all around the world.

Figure 4.2: DoTaSet - Persona



Rui Veloso

Bio

As a computer engineering student at University of Porto, Rui Veloso 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 'waving dataset' for his latest project.

General info

Age	22
Location	Porto
Occupation	Student at University of Porto
Course	Computer & Informatics engineering

Personality

Hobbies

- Music
- Cinema
- Read and write books and music
- Volunteering

Figure 4.3: Rui Veloso - Persona

4.2 Requirements

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

4.2.1 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...)

⇒ 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

⇒ Framework must support submission of contents and input data

⇒ System must be multi platform supporting *Windows*, *Linux* and *macOS*.

⇒ The framework should provide a user-friendly dashboard to ease the process of interacting with the data and the system

⇒ Data should be formatted in a standardized way

⇒ The system should be capable of extracting relevant features of collected data

4.2.2 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 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

⇒ 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

4.3 Architecture

This modular framework for data acquisition and annotation to support interaction scenarios is a type of software architecture that allows for the flexible and extensible integration of various data sources and annotation tools. It is design in a way that its pieces can work completely apart from each other allowing for addition or removal of components as needed.

As such, the framework offers several advantages over more straight forward approaches. It allows for greater flexibility and adaptability, as new data sources can be easily integrated without affecting the overall architecture while promoting code re-usability and maintainability, as the modular components can be developed and tested independently and shared among different projects.

Thus, this architecture is capable of supporting a wide range of interaction scenarios and different types of data.

A high-level view of the intended architecture is presented in the following figure.

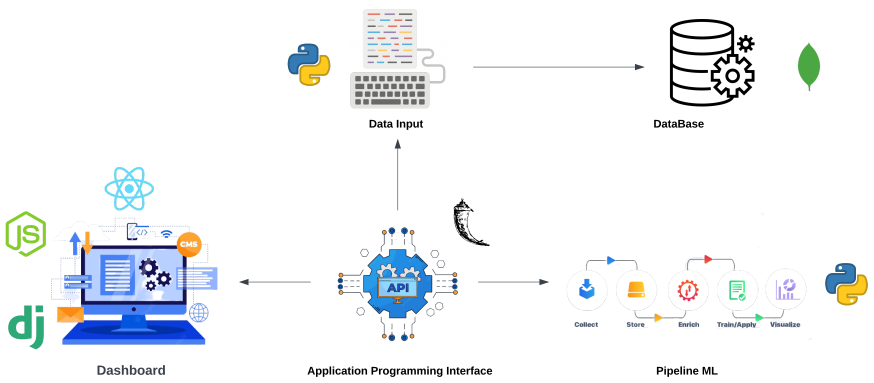


Figure 4.4: Framework Architecture

- **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 (processing services, data treatment modules and dashboard). It is in charge of displaying info needed to the users and manage the flow of acquired data to the insert module and filtered data to the pipeline. It also allows a clear and more versatile separation of the backend services from the user interface.

- **The Data Insert** module is a more complex feature that is represented in a very high-level in this architecture model. However it is charged with doing the following:

  - **Data Acquisition -** receives inputed data from the API and properly stores it in the database.
  - **Data Annotation -** adds labels and tags defining what the acquired data relates to in order to later improve the accuracy of data analysis.
  - **Data Filtering -** receives specified attributes to be searched for in the collected data to than choose the best fitting content to be fed into the ML pipeline for analysis. This process mainly works to reduce the ammount of unnecessary information based on the labels associated with the data.

- **The Database** is projected to be a non-relational storage unity to provide a secure and efficient flow of data between passive and active services.

- **The ML Pipeline** is the main processing feature of the architecture that iterates data through a series of stages in order to produce accurate models of various data types. In this module the content goes through an analysis process that is adaptative to the type of data to be worked with as well as to the goal that is meant to serve, thus this procedure can make use of various techniques such as regression, classification, clustering and dimensionality reduction. The output of this evaluation will be a set of trends, patterns and relationships to support better decision-making from the remaining pipeline

4.4 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.

4.4.1 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 **React**, an open-source, declarative, and flexible JavaScript library, since it is a efficient, declarative and flexible programming language in a way that it can be used on a huge variety of platforms to build scalable and richer interactive user interfaces once the more the User Interface (UI) is attractive the more they will be attached to our framework. But the question is,

Why choose ReactJS when you have PHP, HTML 5, Flutter, or others?

That's because React makes it easier to create a clean and better architecture that is friendly to testing and another advantage is that this technology allows data changes to be made without reloading the page and another important fact is that React 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.

4.4.2 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 an easier connection with the *frontend* and *API's.*

For the database we opted to go with *MongoDB*[2] 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.

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

[2] https://www.mongodb.com/home

Chapter 5

Procedure / Implementation

In virtue of the personas and use cases mentioned in the previous chapter (3), the team started the implementation of the *framework* developing a *web platform* that allows the user to capture video with or without audio as well as a simple client-server connection. At the end of the capture, the user might desire to download their recent capture, start over or upload it to the *framework's* ***DataBase***. Under those circumstances, it could be tested by anyone with this link[1]or instead, on the next present figures will be possible to observe the simple web platform implemented and it's features.

[1] `https://github.com/PECI2022/demo-repository`

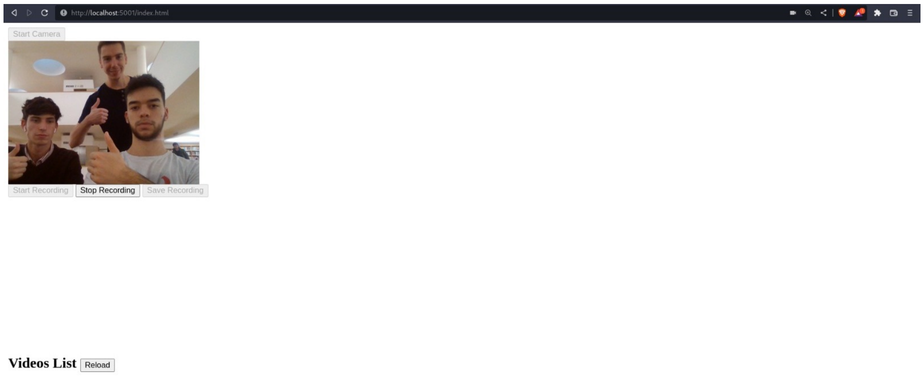
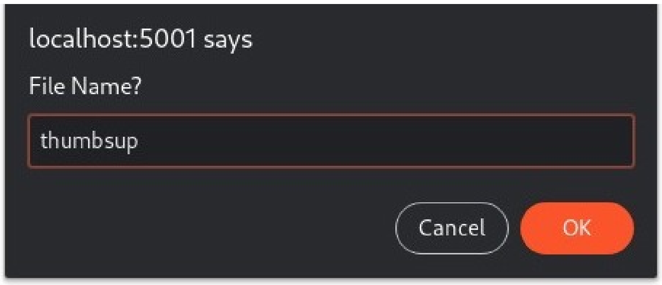


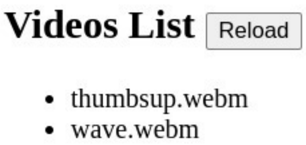
Figure 5.1: Recording data


localhost:5001 says

File Name?

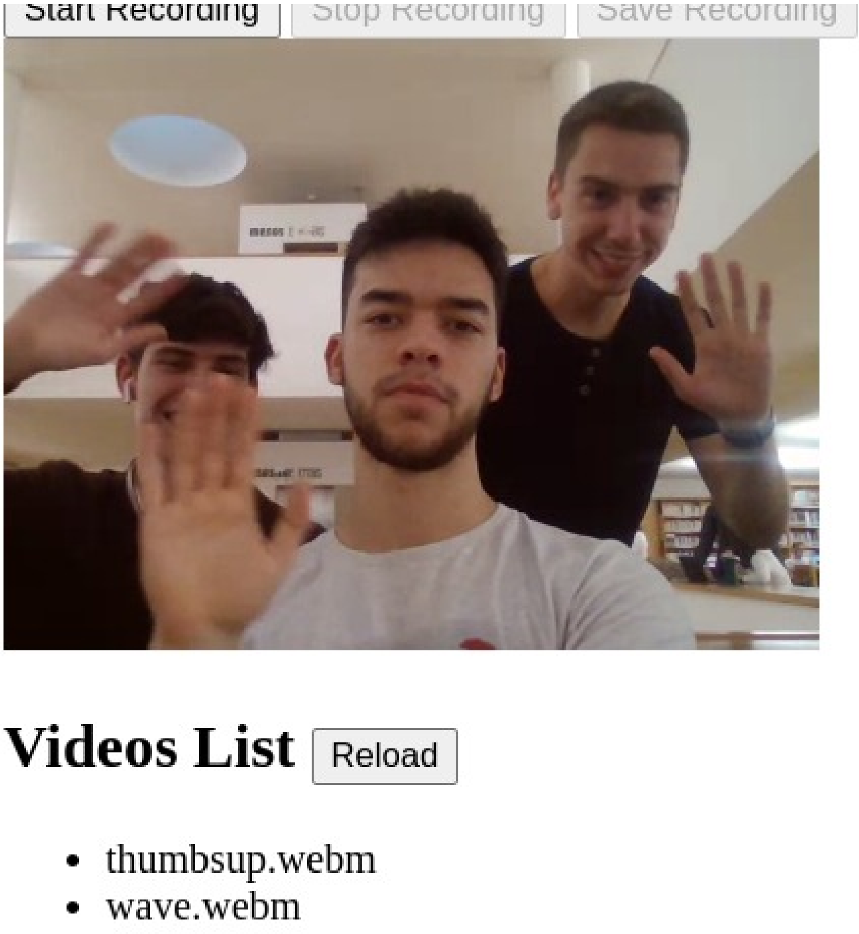
thumbsup

Figure 5.2: Caption/ Annotation the recorded video


Videos List

- thumbsup.webm
- wave.webm

Figure 5.3: Videos List of the recorded videos



- thumbsup.webm
- wave.webm

Figure 5.4: Present the video chosen from the list of videos above the list

Chapter 6

Conclusion / summary

In conclusion, the Modular Framework for Data Acquisition and Annotation to Support Interaction Scenarios will be a versatile and powerful tool to collect and analyse data in a wide range of interaction scenarios, thus building models 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 will allow 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.

In future iterations we expect to implement algorithms into our Machine Learning pipeline, supporting various data analysis techniques, as well as improving the content acquisition, annotation and filtering capabilities. Other points to work on would be the scalability of the database and the interface of the dashboard

Chapter 7

Work Distribution

- Eduardo Fernandes: 20% - Procedure / implementation and Demonstration, Related Work
- Guilherme Claro: 20% - Introduction
- João Afonso Ferreira: 20% - Background, Related work and State of the Art, Technologies, Requirements, Personas/Actors, Use Cases, Architecture
- Pedro Durval: 20% - Related Work, Conclusion, Architecture, Abstract, Requirements, Technologies
- Tiago Mostardinha: 20% - Personas/ use cases

Bibliography

[1] 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.

[2] NetApp, “Artificial intelligence, machine learning”, 2022.

[3] OpenAIRE, online from `https://www.openaire.eu/how-to-make-your-data-fair`, 2022.

[4] 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.