Student Name

Student ID

Facial Data Collection

Requirements Specification Document

# Document Information

|  |  |
| --- | --- |
| Category | Information |
| Customer | - |
| Project | Facial Data Collection App |
| Document | Requirements Specification Document |
| Document Version | 1.0 |
| Identifier | - |
| Status | Complete |
| Author(s) | Student Name |
| Approver(s) | Supervisor Name |
| Issue Date | June 10, 2021, 2021 |
| Document Location | United Kingdom |
| Distribution | - |

Table of Contents

[Document Information 1](#_Toc74124996)

[Introduction 3](#_Toc74124997)

[Purpose 3](#_Toc74124998)

[Scope 3](#_Toc74124999)

[Project Modules 3](#_Toc74125000)

[Background 3](#_Toc74125001)

[Development Technologies 4](#_Toc74125002)

[Overall Description 4](#_Toc74125003)

[Specific Requirements 4](#_Toc74125004)

[Functional Requirements 4](#_Toc74125005)

[Non-functional Requirements 4](#_Toc74125006)

[Availability 4](#_Toc74125007)

[Usability 4](#_Toc74125008)

[Performance 4](#_Toc74125009)

[Design Constraints 4](#_Toc74125010)

[Interfaces 5](#_Toc74125011)

[Architecture Diagram 5](#_Toc74125012)

[Use Cases 6](#_Toc74125013)

[Use Case Diagram 6](#_Toc74125014)

[Use Case Descriptions 6](#_Toc74125015)

[UC01 – Initiate the Machine Learning Model 6](#_Toc74125016)

[Sequence Diagrams 7](#_Toc74125017)

[Sequence Diagram 1 – Take Image and Save to Local Storage 7](#_Toc74125018)

[Sequence Diagram 2 – Machine Learning Model Prediction 7](#_Toc74125019)

# Introduction

This section sheds light on the purpose of this document to provide a basic understanding of the system for a head start. For vocabulary that might be too technical, an appendix section will be provided at the end of the document.

## Purpose

With technological development at its all-time high, the environment now adapts to humans rather than the humans adapting to the environment. Experience defines everything and humans are constantly striving to create a better experience of every single activity performed. This requires an in-depth analysis of the way humans react and interact with the environment. One of the methods to study this is human involuntary reactions. Biomechanical researchers strive to collect facial data for involuntary expressions from observing the way humans react to several stimuli from the environment. This project makes it easier for the researchers to record facial data and classify it and store it in the database by automating the complete process using Machine Learning.

The purpose this document carries is to document the functionalities of the proposed system along with the design process required to build the system. The system is created from Flask and Jinja, and consists of a front-end module and the business logic integrated into it.

The document also provides the non-functional requirements for the system along with an overview of the system constraints and the major stakeholders in the system.

## Scope

* The proposed system consists of a single interface with the local system storage treated as the database.
* The system is server-based and the server is located on the local machine the code is executed on.
* The primary language would be English.
* The system would be tested in a controlled environment as well as in a real environment.

### Project Modules

#### Front End

Made with Jinja, which is the template engine of choice for this project paired with Flask. It uses basic HTML and inline CSS to provide a highly basic and functional UI which caters to the requirements.

#### Flask Server

Made with Flask, the server is based in the local machine and is created with Python. The back-end server includes the routes for the application and the business logic which includes data preprocessing, model initialization, prediction, webcam control and file I/O as functions that can be called once a specific route is accessed.

## Background

Biomechanical researchers require a lot of facial data from involuntary expressions made by humans upon sudden interaction from environmental stimulus. Currently there are high-end systems available like iMotion that perform this on a very high level but none that are easily accessible and require less work input and are easier to use.

This project provides the ease of access to a good quality facial recognition algorithm that will perform with excellent accuracy and will provide the data required for biomechanical research.

## Development Technologies

* Flask
* Jinja
* Bootstrap
* HTML
* Python

# Overall Description

This app is built to provide a UI to the Machine Learning algorithm responsible for the classification of involuntary human facial expression into various emotions. The domain for this project would be Machine Learning and Facial Recognition. Although this is the current extent of the system, the system would be capable of scaling and expanding according to the requirements if necessary.

* Stakeholders: Biomechanical Researchers, Students, Data Scientists, Researchers
* Major Product Functions:
  + Data collection
  + Prediction using Machine Learning

# Specific Requirements

The following are the functionalities of the system under consideration:

## Functional Requirements

1. Provide a way to receive live camera feed to record user facial expressions.
2. Provide a way to initialize the algorithm prediction procedure.
3. Provide a slideshow of images that will act as a surprise external stimulus for the user.
4. Take images of the user and store it in the local storage of the device that it is running on.
5. Save the prediction made by the model in a text file for future reference.

## Non-functional Requirements

### Availability

1. Be available to all potential stakeholders [99.03%](https://medium.com/pingdom/downtime-for-the-worlds-top-50-e-commerce-websites-d6d4865deb0#:~:text=The%20average%20uptime%20for%20the,a%2099.99%25%20uptime%20or%20better.) of the time.

### Usability

1. Implement the principle of locality for the complete website.
2. Be easy to use and remember.
3. The brief and simple language will be used on the website.
4. Use meaningful images on the website.

### Performance

1. Load the system webpages within 3 seconds on average.
2. Implement server-side rendering for the interface due to the dynamic nature of the majority of the data on the website.

### Design Constraints

1. Be a web-based system.

### Interfaces

#### User Interface

The proposed system will have only one page with the following contents on it:

* Slideshow with images used as External Stimulus for the User.
* Webcam Live Feed.
* A Button for Initiating the Prediction Procedure.

# Architecture Diagram

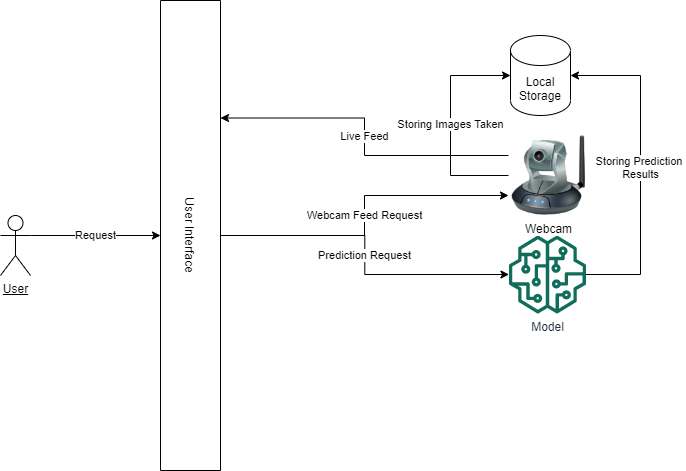


Figure 1 - Architecture Diagram

*Figure 1* provides a bird’s eye view of the proposed system. There is only a single User Interface for the user. All the requests to different functions and routes of the project are accessed in this User Interface. The User Interface houses the slideshow that provides the external stimulus for the user. It also houses the live feed from the webcam and a button that acts as a trigger for the Machine Learning Model. The request for the live feed from the camera is continuously generated and handled by the User Interface and the business logic. The prediction button on the User Interface sends a trigger request to the Machine Learning model through routing and the Model starts predicting results by ingesting images taken by the camera in the local storage.

# Use Cases

## Use Case Diagram

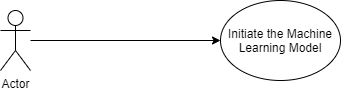


Figure 2 - Use Case Diagram (Incomplete)

## Use Case Descriptions

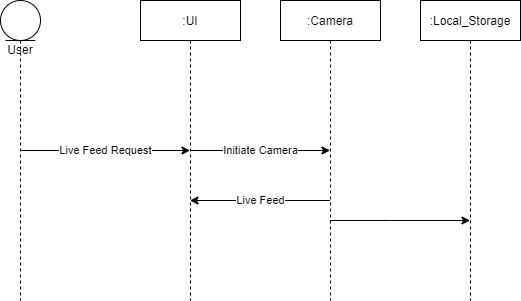
### UC01 – Initiate the Machine Learning Model

|  |  |  |  |
| --- | --- | --- | --- |
| **Initiate the Machine Learning Model** | | | |
| **Actor: User** | | | |
| **Feature:** User wants to predict the emotions in the images of the user taken in the local storage. | | | |
| **Requirement ID** | | FR-02 | |
| **Use-Case ID** | | UC01 | |
| **Preconditions** | | The customer must be on the main page. | |
| **Step #** | **User Action** | | **System Response** |
| 1. | Navigate to main screen. | | System displays the main screen. |
| 5. | User waits for the Images to be Taken for each External Stimulus in the Slideshow. | |  |
| 6. | Click the ‘Prediction’ button. | | System produces the Prediction Result file in the Local Storage. |
| **Post Condition** | | | |
| The user will be able to access the prediction results. | | | |

Figure 3 - UI 1

# Sequence Diagrams

## Sequence Diagram 1 – Take Image and Save to Local Storage



## Sequence Diagram 2 – Machine Learning Model Prediction

