
High Level Design

MEXICAN SIGN LANGUAGE ALPHABET IMAGE CLASIFICATION SYSTEM

Erick Ibarra
A00959090

February 13, 2017

PURPOSE

The development of this document has as an objective to establish the ideas behind what the Mexican Sign Language Alphabet Image Classification System should do and how it should make them possible at a high level.

1 FUNCTIONAL DESCRIPTION

1.1 OVERVIEW

The Mexican Sign Language Alphabet Image Classification System shall be able to take an image as input and predict the letter the hand is representing from the possible alphabet letters. To achieve this the system must know how a hand is represented in an image and the different signs it can make. In order to understand at a greater detail please refer to the Functional Requirements Document.

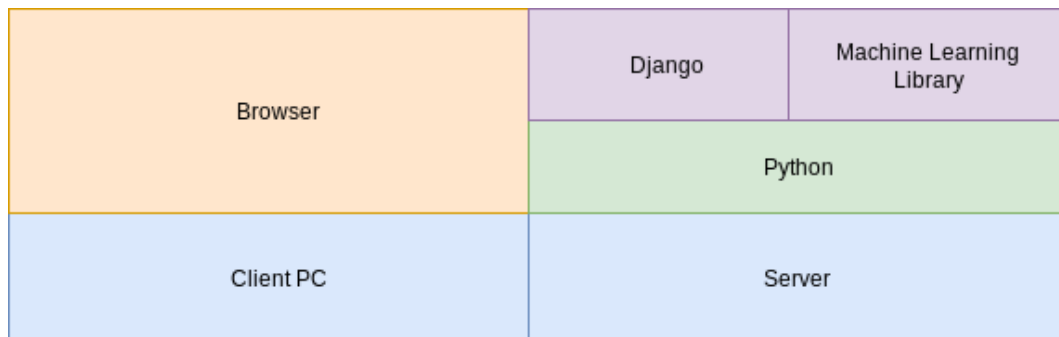
1.2 LOGICAL FLOW AND BUSINESS RULES

In order to achieve the business goals listed on the Business Proposal and detailed on the Functional Requirements Document the application should count with the following functionalities:

- `processImage(img)`
- `createMask(img)`
- `findHand(mask)`
- `normalizeImage(mask)`
- `storeImage(normalizedMask)`
- `predictClass(normalizedMask)`
- `train(model)`

1.3 ARCHITECTURAL OVERVIEW

The general architecture decided to build the system is described in the following diagram some of the details on how this components are used are described in the Functional Requirements Document.



1.4 APPLICATION COMPONENT ADDITIONS OR MODIFICATIONS

In order to develop the system and achieve the minimum accuracy desired the system must be trained with a huge dataset of images as described in the Functional Requirements Document. This may require the elaboration of a different mini-system which is not included in the scope of the project.

1.5 PAGES OR FORMS

For the client portion of the system a single form to upload the image shall be made. This form should also display the output and it also has to receive the feedback. The process is going to be shown in the same page this form resides.

1.6 ERROR CONDITIONS

The most common error which will happen is that when the image is uploaded the hand may not be very distinguishable from the image and hence not predicting at all any solution. A message stating this should be shown to the user in case it happens.

1.7 SYSTEM DEPENDENCIES/INTERACTION

All the system dependencies and interfaces are listed and described in the Functional Requirements Document. As it is shown in the architecture overview the system depends on the python programming language and some of its libraries.

2 CONFIGURATION GUIDE

The following section will list the steps in order to test the system for a successful deployment. To be able to successfully test the functionalities of the system the following steps should be followed:

- Take a picture with the format specified in the Functional Requirements Document.
- Open a browser and visit the URL for the system, this url may end up being a page in my website.
- Login to the system with the credentials generated by the administrator.
- Upload the image to the classifier
- Give accurate feedback to the system

In order to deploy the system in a new server the guide will be outlined in the repository of the project depending on the final configuration.

3 TEST PLAN

The different tests which the system should be able to pass will be briefly described in the following section. Although the real test will be when the users interact with the system some of test have to be passed before even going into production. This list of tests are the ones which the system should achieve in order to consider the solution deployable.

3.1 MODEL TESTS

- Achieve 90% or more accuracy in predicting the correct image label
- The model should be able to store new data which meets the conditions

3.2 USER TESTS

- A user shall be able to login correctly to the system
- An administrator shall be able to create users

4 PROJECT TIMELINE

As stated on the business document, the budget for this project is around 30 to 40 hours of work. This section will show a more detailed structure in which this hours will be used.

4.1 RESEARCH

Most of the time on this project will be destined to the research of possible solutions which help improve the accuracy of the system. I figure 10 to 15 hours will be needed in order to get a general view of the area and the possible methods to implement.

4.2 DOCUMENTATION

As demonstrated by this delivery, a minimum of 10 hours should be dedicated to the documentation of the code and the project itself.

4.3 CODE AND TESTING

For the actual development of the system the remaining 20 hours will be used. This includes the actual tests developed for the code and the different iterations of the same code with some tweaks being tested. Most of the time will be used in the iteration of new techniques and different parameters.