

PROJECT REPORT

A NOVEL METHOD FOR HANDWRITTEN DIGIT RECOGNITION

**A Novel Method for Handwritten Digit**

**Recognition System**

***submitted by***

***PNT2022TMID15689***

Gowthamen Raj R- 111719104056 Eashwar Karthik V-111719104051 Hareesh R – 111719104058 Hemesh M - 111719104059

**TABLE OF CONTENTS**

|  |  |  |  |
| --- | --- | --- | --- |
| **1** | **INTRODUCTION** | **1** |  |
|  | PROJECT OVERVIEW |  | 1 |
|  | PURPOSE |  | 1 |
|  | **2 LITERATURE SURVEY** |  | **2** |
|  | EXISTING PROBLEM |  | 2 |
|  | REFERENCES |  | 2 |
|  | PROBLEM STATEMENT DEFINITION |  | 5 |
|  | **3 IDEATION AND PROPOSED SOLUTION** |  | **6** |
|  | EMPATHY MAP CANVAS |  | 6 |
|  | IDEATION & BRAINSTORMING |  | 7 |
|  | PROPOSED SOLUTION |  | 8 |
|  | PROBLEM SOLUTION FIT |  | 9 |
|  | **4 REQUIREMENT ANALYSIS** |  | **10** |
|  | FUNCTIONAL REQUIREMENTS |  | 10 |
|  | NON FUNCTIONAL REQUIREMENTS |  | 11 |
|  | **5 PROJECT DESIGN** |  | **12** |
|  | DATA FLOW DIAGRAM |  | 12 |
|  | SOLUTION & TECHNICAL ARCHITECTURE |  | 13 |
|  | USER STORIES |  | 15 |

1. PROJECT PLANNING AND SCHEDULING 16

[SPRINT PLANNING AND ESTIMATION 16](#_TOC_250009)

[SPRINT DELIVERY SCHEDULE 17](#_TOC_250008)

1. CODING & SOLUTIONING 18
2. TESTING 20

[TEST CASES 20](#_TOC_250007)

USER ACCEPTANCE TESTING 22

DEFECT ANALYSIS 22

[TEST CASE ANALYSIS 22](#_TOC_250006)

1. RESULTS 23

[PERFORMANCE METRICS 23](#_TOC_250005)

1. ADVANTAGES & DISADVANTAGES 25

[ADVANTAGES 25](#_TOC_250004)

[DISADVANTAGES 25](#_TOC_250003)

1. CONCLUSION 26
2. FUTURE SCOPE 27

[APPENDIX 28](#_TOC_250002)

[SOURCE CODE 28](#_TOC_250001)

[GITHUB 37](#_TOC_250000)

PROJECT DEMO 37

# CHAPTER 1

## INTRODUCTION

### PROJECT OVERVIEW

Machine learning and deep learning play an important role in computer technology and artiﬁcial intelligence. With the use of deep learning and machine learning, human effort can be reduced in recognizing, learning, predictions and in many more areas.

Handwritten Digit Recognition is the ability of computer systems to recognise handwritten digits from various sources, such as images, documents, and so on. This project aims to let users take advantage of machine learning to reduce manual tasks in recognizing digits.

### PURPOSE

Digit recognition systems are capable of recognizing the digits from different sources like emails, bank cheque, papers, images, etc. and in different real-world scenarios for online handwriting recognition on computer tablets or system, recognize number plates of vehicles, processing bank cheque amounts, numeric entries in forms ﬁlled up by hand (tax forms) and so on.

# CHAPTER 2

## LITERATURE SURVEY

### EXISTING PROBLEM

The fundamental problem with handwritten digit recognition is that handwritten digits do not always have the same size, width, orientation, and margins since they vary from person to person. Additionally, there would be issues with identifying the numbers because of similarities between numerals like 1 and 7, 5 and 6, 3 and 8, 2 and 5, 2 and 7, etc. Finally, the individuality and variation of each individual's handwriting inﬂuence the structure and appearance of the digits.

### REFERENCES

#### **R. Bajaj, L. Dey, S. Chaudhari et al**, employed three different kinds of features, namely, the density features, moment features and descriptive component features for c lassification of Devanagari Numerals. They proposed multi classifier connectionist architecture for increasing the recognition reliability and they obtained 89.6% accuracy for handwritten Devanagari numerals.

**Aparna et al,** proposed a method to construct a handwritten Tamil character by executing a sequence of strokes. A structure or shape-based representation of a stroke was used in which a stroke was represented as a string of shape features. Using this string

#### representation, an unknown stroke was identified by comparing it with a database of strokes using a flexible string matching procedure.

**Renata F. P. Neves** has proposed SVM based offline handwritten digit recognition. Authors claim that SVM outperforms the Multilayer perceptron classifier. Experiment is carried out on NIST SD19 standard dataset. Advantage of MLP is that it is able to segment non-linearly separable classes. However, MLP can easily fall into a region of local minimum, where the training will stop assuming it has achieved an optimal point in the error surface. Another hindrance is defining the best network architecture to solve the problem, considering the number of layers and the number of perceptrons in

#### each hidden layer. Because of these disadvantages, a digit recognizer using the MLP structure may not produce the desired low error rate.

**Improved Handwritten Digit Recognition Using Quantum K-Nearest Neighbor Algorithm (2019)**

*Wang, Yuxiang and Wang, Ruijin and Li, Dongfen and Adu-Gyamﬁ, Daniel and Tian, Kaibin and Zhu, Yixin*

The KNN classical machine learning technique is used in this research to enable quantum parallel computing and superposition. They used the KNN algorithm with quantum acceleration to enhance handwritten digit recognition. When dealing with more complicated and sizable handwritten digital data sets, their suggested method considerably lowered the computational time complexity of the traditional KNN algorithm. The paper offered a theoretical investigation of how quantum concepts can be applied to machine learning. Finally, they established a fundamental operational concept and procedure for machine learning with quantum acceleration.

### PROBLEM STATEMENT DEFINITION

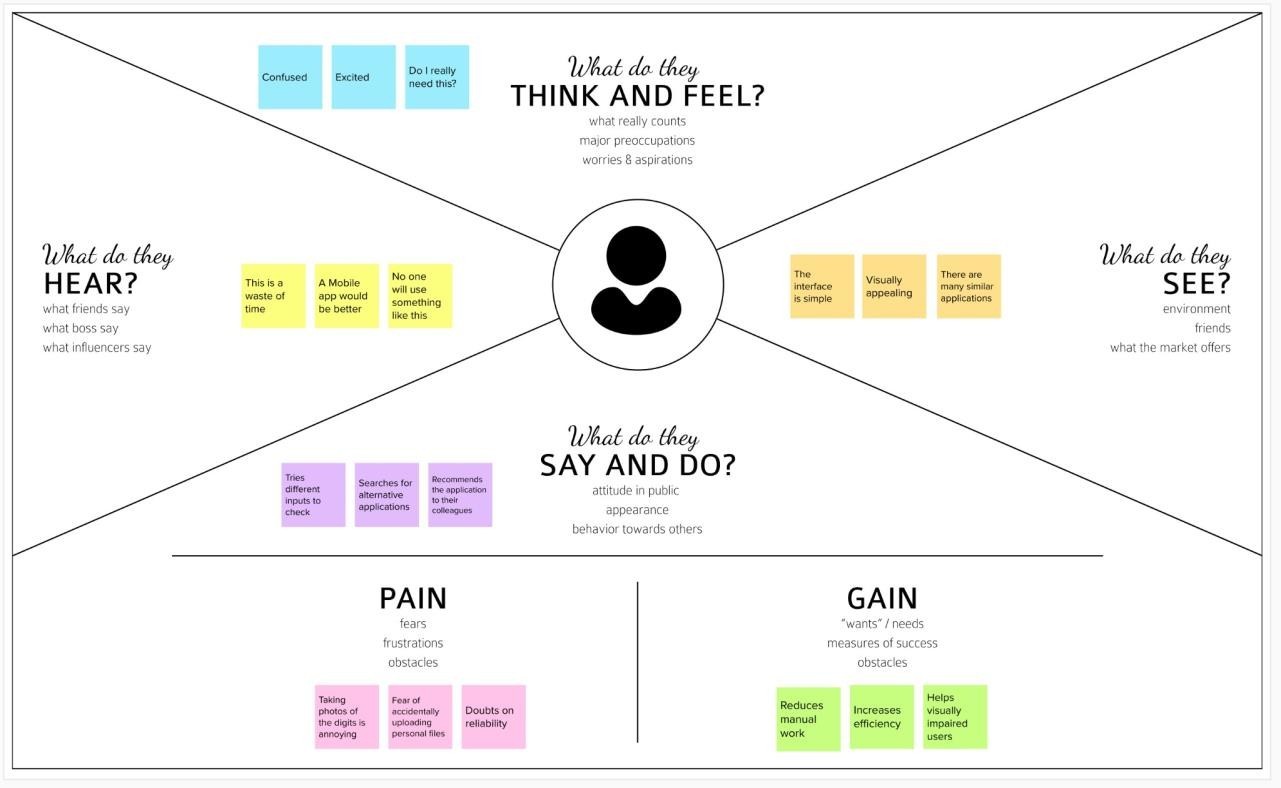
For years, the trafﬁc department has been combating trafﬁc law violators.

These offenders endanger not only their own lives, but also the lives of other individuals. Punishing these offenders is critical to ensuring that others do not become like them. Identiﬁcation of these offenders is next to impossible because it is impossible for the average individual to write down the license plate of a reckless driver. Therefore, the goal of this project is to help the trafﬁc department identify these offenders and reduce trafﬁc violations as a result.

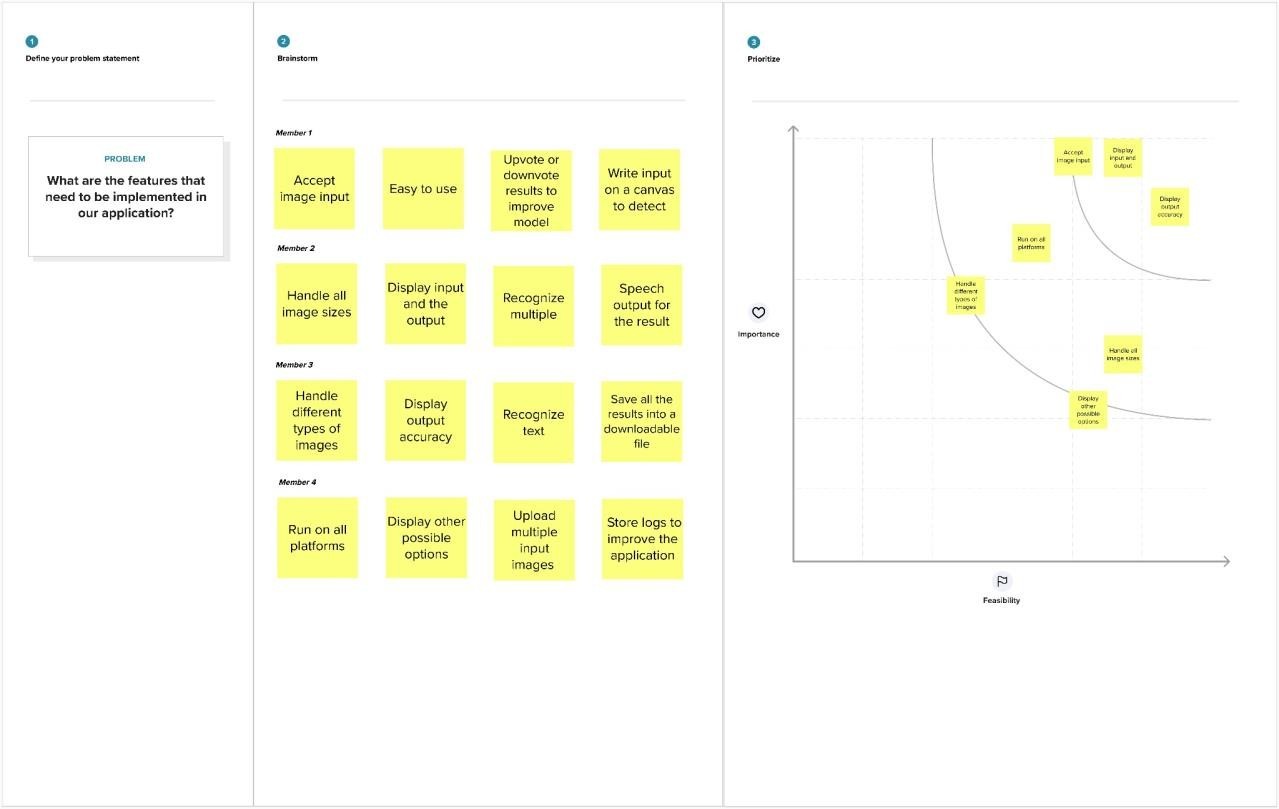
# CHAPTER 3

## IDEATION AND PROPOSED SOLUTION

### EMPATHY MAP CANVAS



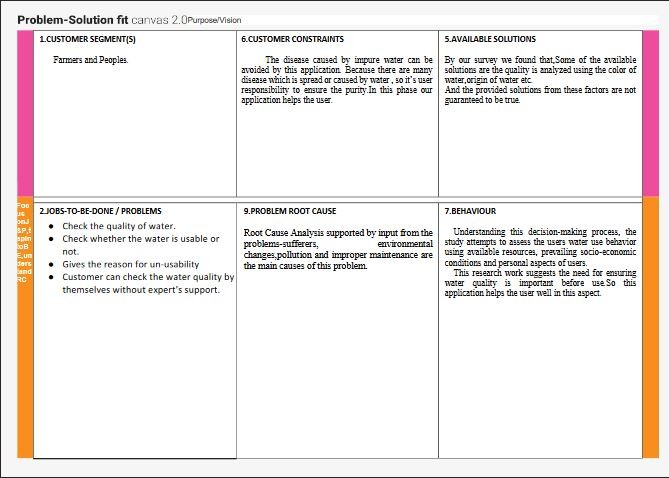
**IDEATION & BRAINSTORMING**

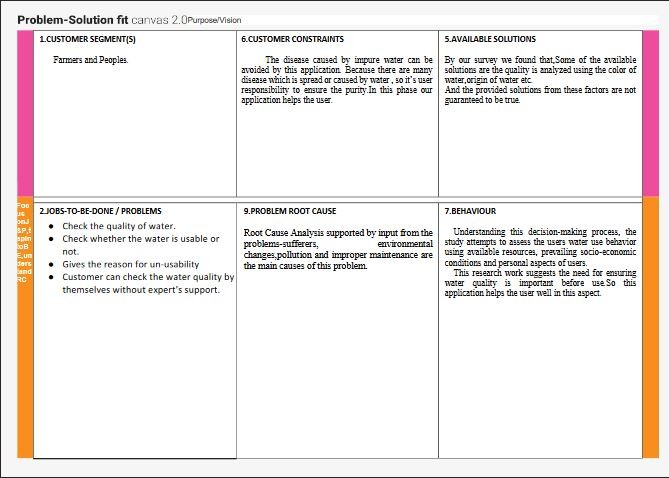


### PROPOSED SOLUTION

|  |  |  |
| --- | --- | --- |
| **S.NO** | **PARAMETER** | **DESCRIPTION** |
| 1 | Problem Statement | To create an application that recognizes handwritten digits |
| 2 | Idea / Solution Description | The application takes an image as the input and accurately detects the digits in it. |
| 3 | Novelty / Uniqueness | Instead of recognizing every text, the application accurately recognizes only the digits |
| 4 | Social Impact / Customer Satisfaction | This application reduces the manual tasks that need to be performed. This improves productivity in the workplace. |
| 5 | Business Model | The application can be integrated with traffic surveillance cameras to recognize vehicle number plates  The application can be integrated with Postal systems to recognize the pin codes effectively |
| 6 | Scalability of the Solution | The application can easily be scaled to accept multiple inputs and process them parallelly to further increase efficiency |

**PROBLEM SOLUTION FIT**

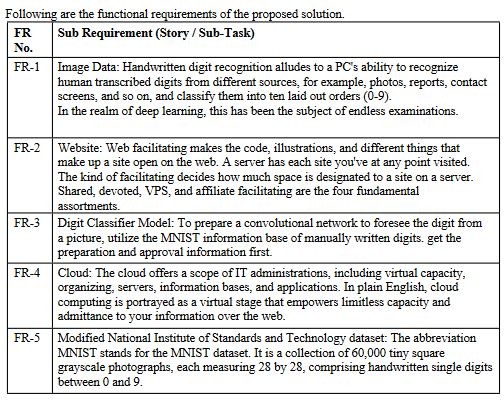




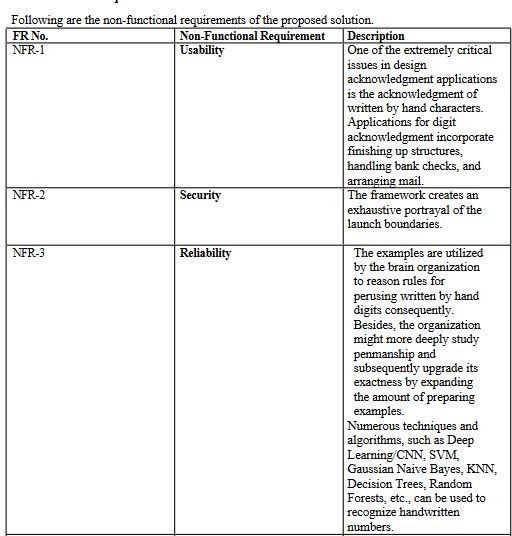
# CHAPTER 4

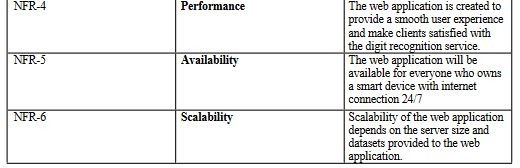
## REQUIREMENT ANALYSIS

### FUNCTIONAL REQUIREMENTS



**NON FUNCTIONAL REQUIREMENTS**

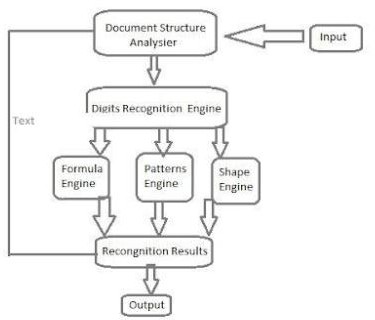




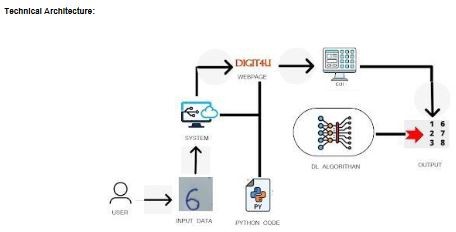
# CHAPTER 5

## PROJECT DESIGN

### DATA FLOW DIAGRAM



**SOLUTION & TECHNICAL ARCHITECTURE**



**USER STORIES**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| User Type | Functional Requirements | User Story Number | User Story / Task | Acceptance Criteria | Priority | Release |
| Customer | Accessing the Application | USN-1 | As a user, I should be able to access the application from anywhere and use on any devices | User can access the application using the browser on any device | High | Sprint-4 |
| Uploading Image | USN-2 | As a user, I should be able to upload images to predict the digits | User can upload images | High | Sprint-3 |
| Viewing the Results | USN-3 | As a user, I should be able to view the results | The result of the prediction is displayed | High | Sprint-3 |
| Viewing Other Prediction | USN-4 | As a user, I should be able to see other close predictions | The accuracy of other values must be displayed | Medium | Sprint-4 |
| Usage Instruction | USN-5 | As a user, I should have a usage instruction to know how to use the application | The usage instruction is displayed on the home page | Medium | Sprint-4 |

# CHAPTER 6

## PROJECT PLANNING AND SCHEDULING

### SPRINT PLANNING AND ESTIMATION

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SPRINT** | **USER STORY / TASK** | **STORY POINTS** | **PRIORITY** | **TEAM MEMBERS** |
| Sprint - I | Get the dataset | 3 | High | Eashwar Karthik V |
| Explore the data | 2 | Medium | Eashwar Karthik V Hemesh M |
| Data Pre-Processing | 3 | High | Hareesh R GowthamenRaj R |
| Prepare training and testing data | 3 | High | Hareesh R Gowthamen Raj R |
| Sprint - II | Create the model | 3 | High | Hareesh R |
| Train the model | 3 | High | Gowthamen Raj R |
| Test the model | 3 | High | Hemesh M |
| Sprint - III | Improve the model | 2 | Medium | Hareesh R Gowthamen Raj R |
| Save the model | 3 | High | Eashwar Karthik |
| Build the Home Page | 3 | High | Hemesh M Eashwar Karthik V |
| Setup a database to store input images | 2 | Medium | Hareesh R |
| Sprint - IV | Build the results page | 3 | High | Gowthamen Raj R Hareesh R |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Integrate the model with the application | 3 | High | Hemesh M Hareesh R |
| Test the application | 3 | High | Hareesh R |

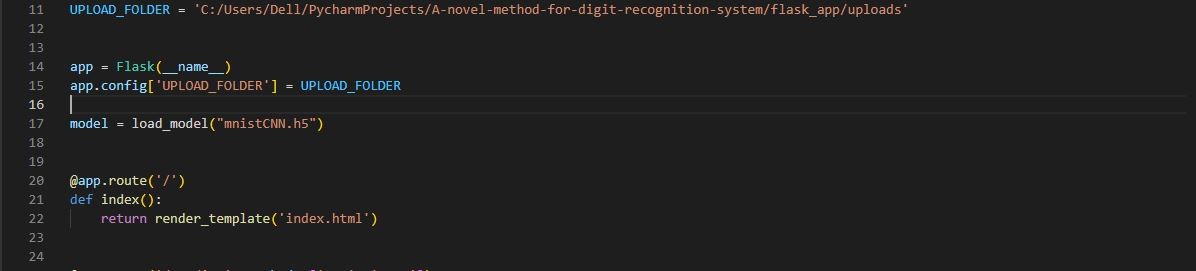
### SPRINT DELIVERY SCHEDULE

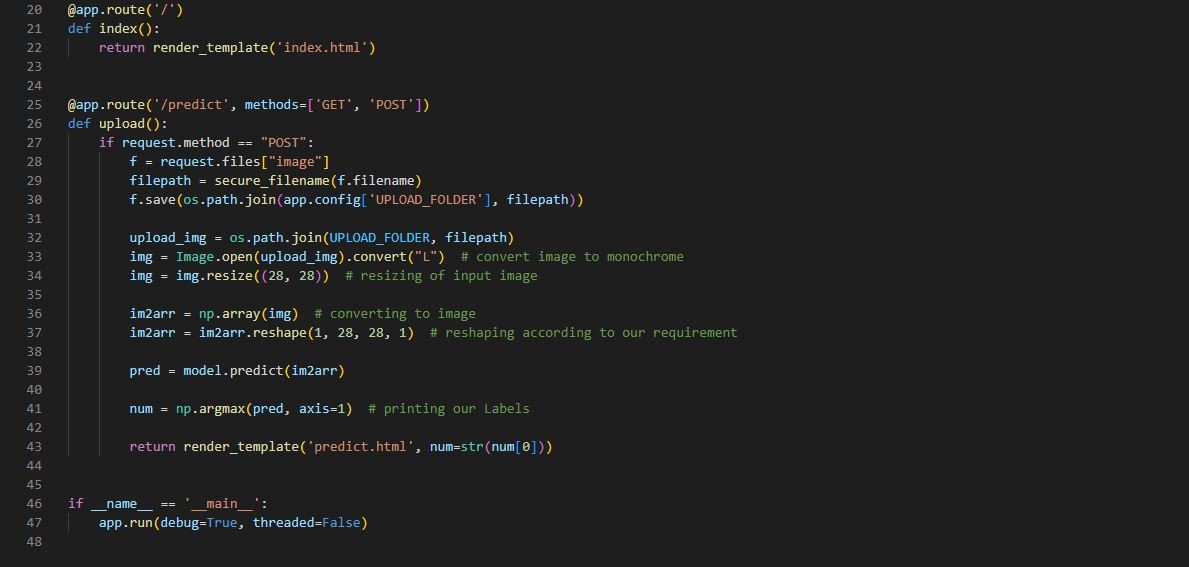
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SPRINT** | **TOTAL STORY POINTS** | **DURATION** | **SPRINT START DATE** | **SPRINT END DATE (PLANNED)** | **STORY POINTS COMPLETED (AS ON PLANNED DATE)** | **SPRINT RELEASE DATE (ACTUAL)** |
| Sprint - I | 11 | 6 Days | 24 Oct  2022 | 29 Oct  2022 | 11 | 29 Oct  2022 |
| Sprint - II | 9 | 6 Days | 31 Oct  2022 | 05 Nov  2022 | 9 | 05 Nov  2022 |
| Sprint - III | 10 | 6 Days | 07 Oct  2022 | 12 Nov  2022 | 10 | 12 Nov  2022 |
| Sprint - IV | 9 | 6 Days | 14 Nov  2022 | 19 Nov 2022 | 9 | 19 Nov 2022 |

# CHAPTER 7

**CODING & SOLUTIONING**







# CHAPTER 8

## TESTING

### TEST CASES

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test case ID** | **Feature Type** | **Component** | **Test Scenario** | **Expected Result** | **Actual Result** | **Status** |
| HP\_TC\_001 | UI | Home Page | Verify UI elements in the Home Page | The Home page must be displayed properly | Working as expected | FAIL |
| HP\_TC\_002 | UI | Home Page | Check if the UI elements are displayed properly in different screen sizes | The Home page must be displayed properly in all sizes | The UI is not displayed properly in screen size 2560 x 1801  and 768 x 630 | FAIL |
| HP\_TC\_003 | Functional | Home Page | Check if user can upload their ﬁle | The input image should be uploaded to the application successfully | Working as expected | PASS |
| HP\_TC\_004 | Functional | Home Page | Check if user cannot upload unsupported ﬁles | The application should not allow user to select a non image ﬁle | User is able to upload any ﬁle | FAIL |
| HP\_TC\_005 | Functional | Home Page | Check if the page redirects to the result page once the input is given | The page should redirect to the results page | Working as expected | PASS |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| BE\_TC\_001 | Functional | Backend | Check if all the routes are working properly | All the routes should properly work | Working as expected | PASS |
| M\_TC\_001 | Functional | Model | Check if the model can handle various image sizes | The model should rescale the image and predict the results | Working as expected | PASS |
| M\_TC\_002 | Functional | Model | Check if the model predicts the digit | The model should predict the number | Working as expected | PASS |
| M\_TC\_003 | Functional | Model | Check if the model can handle complex input image | The model should predict the number in the complex image | The model fails to identify the digit since the model is not built to handle such data | FAIL |
| RP\_TC\_001 | UI | Result Page | Verify UI elements in the Result Page | The Result page must be displayed properly | Working as expected | PASS |
| RP\_TC\_002 | UI | Result Page | Check if the input image is displayed properly | The input image should be displayed properly | The size of the input image exceeds the display container | FAIL |
| RP\_TC\_003 | UI | Result Page | Check if the result is displayed properly | The result should be displayed properly | Working as expected | PASS |
| RP\_TC\_004 | UI | Result Page | Check if the other predictions are displayed properly | The other predictions should be displayed properly | Working as expected | PASS |

**USER ACCEPTANCE TESTING DEFECT ANALYSIS**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Resolution | Severity 1 | Severity 2 | Severity 3 | Severity 4 | Total |
| By Design | 1 | 0 | 1 | 0 | 2 |
| Duplicate | 0 | 0 | 0 | 0 | 0 |
| External | 0 | 0 | 2 | 0 | 2 |
| Fixed | 4 | 1 | 0 | 1 | 6 |
| Not Reproduced | 0 | 0 | 0 | 1 | 1 |
| Skipped | 0 | 0 | 0 | 1 | 1 |
| Won’t Fix | 1 | 0 | 1 | 0 | 2 |
| Total | 6 | 1 | 4 | 3 | 14 |

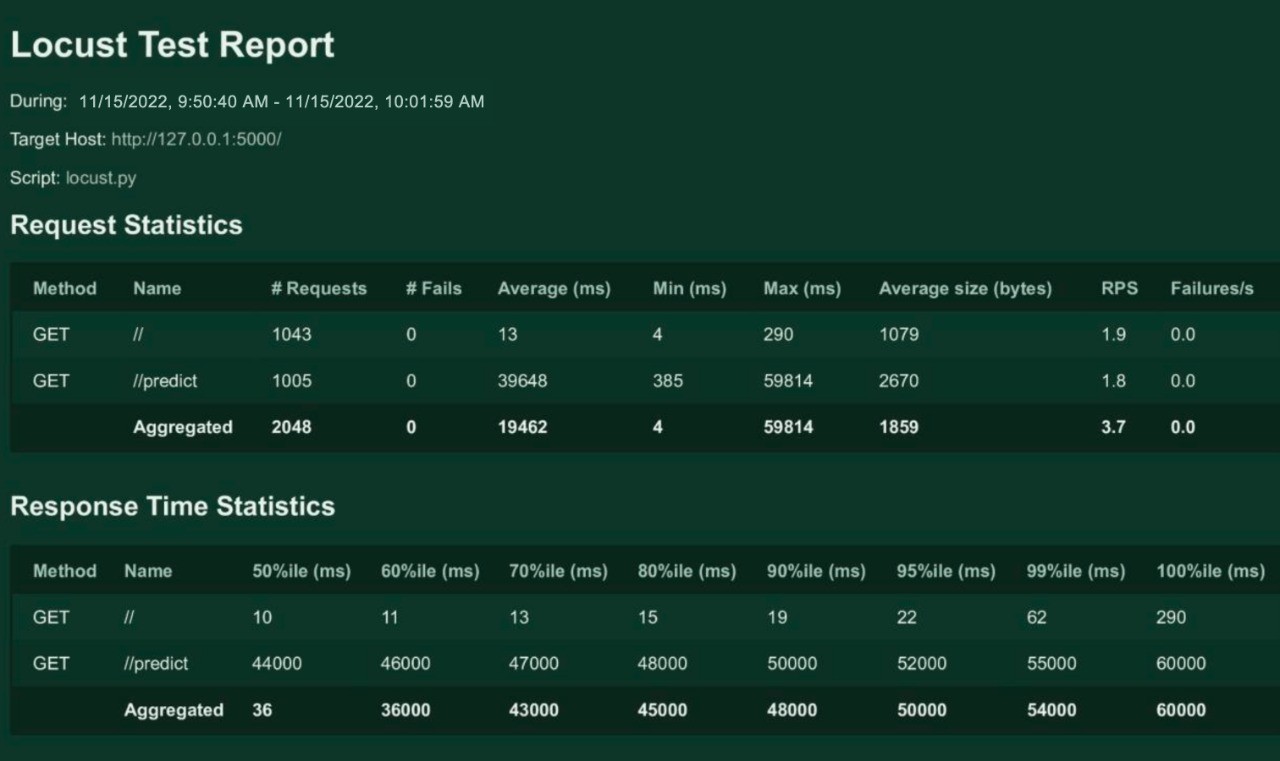
### TEST CASE ANALYSIS

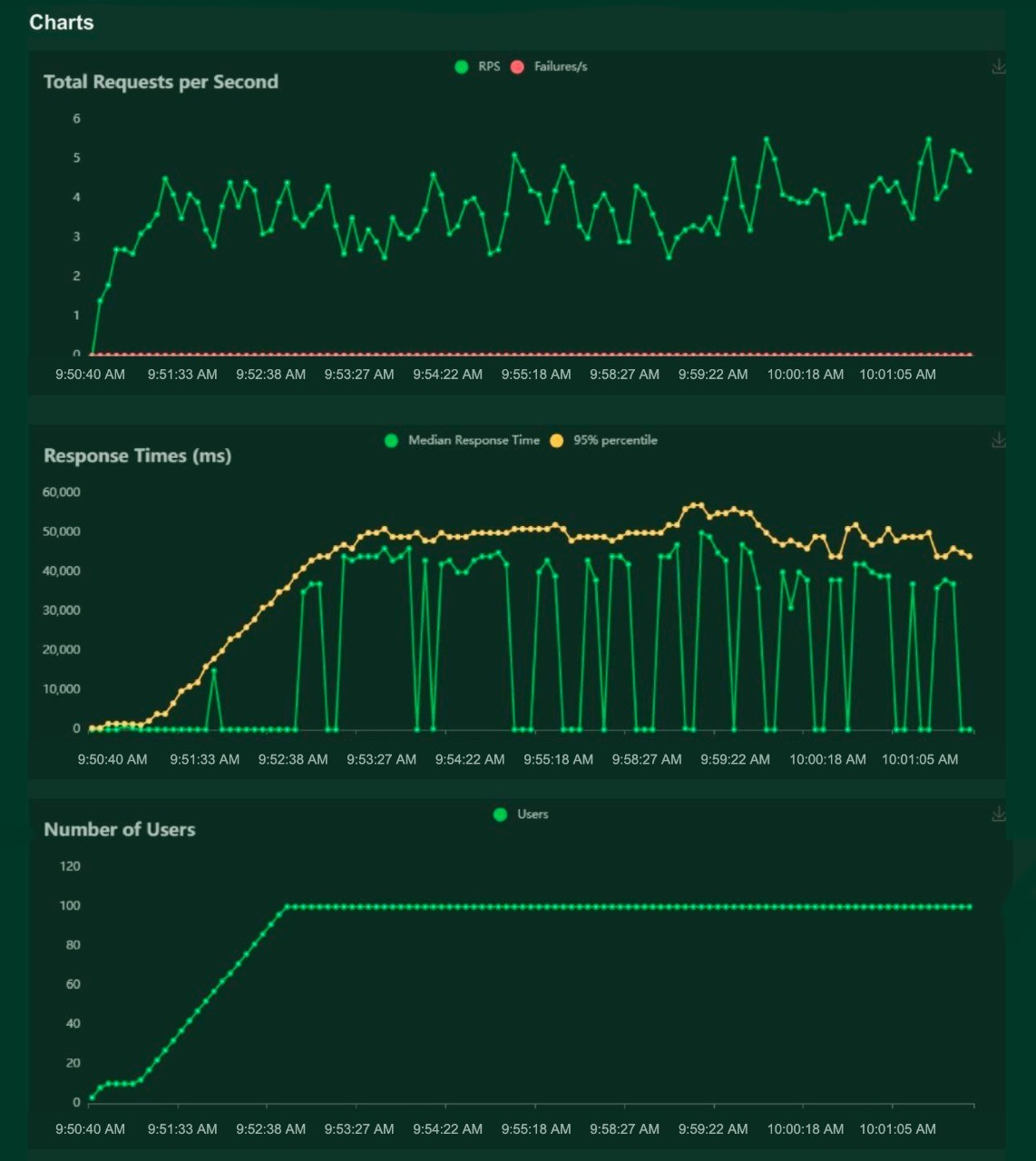
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Section | Total Cases | Not Tested | Fail | Pass |
| Client Application | 10 | 0 | 3 | 7 |
| Security | 2 | 0 | 1 | 1 |
| Performance | 3 | 0 | 1 | 2 |
| Exception Reporting | 2 | 0 | 0 | 2 |

# CHAPTER 9

## RESULTS

### PERFORMANCE METRICS





# CHAPTER 10

## ADVANTAGES & DISADVANTAGES

### ADVANTAGES

* Reduces manual work
* More accurate than average human
* Capable of handling a lot of data
* Can be used anywhere from any device

### DISADVANTAGES

* Cannot handle complex data
* All the data must be in digital format
* Requires a high performance server for faster predictions
* Prone to occasional errors

**CHAPTER 11**

## CONCLUSION

This project demonstrated a web application that uses machine learning to recognise handwritten numbers. Flask, HTML, CSS, JavaScript, and a few other technologies were used to create this project. The model predicts the handwritten digit using a CNN network. During testing, the model achieved a 99.61% recognition rate. The proposed project is scalable and can easily handle a huge number of users. Since it is a web application, it is compatible with any device that can run a browser. This project is extremely useful in real-world scenarios such as recognizing number plates of vehicles, processing bank cheque amounts, numeric entries in forms ﬁlled up by hand (tax forms) and so on. There is so much room for improvement, which can be implemented in subsequent versions.

# CHAPTER 12

## FUTURE SCOPE

This project is far from complete and there is a lot of room for improvement.

Some of the improvements that can be made to this project are as follows:

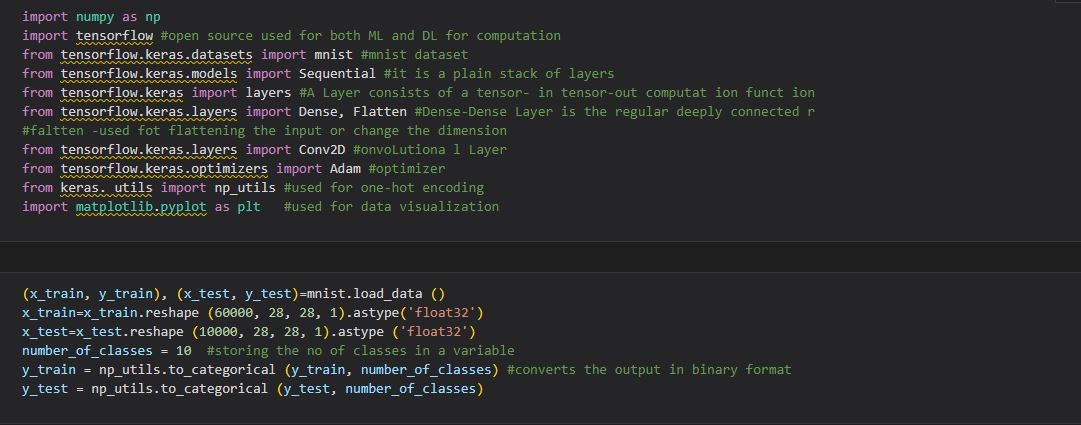
* Add support to detect from digits multiple images and save the results
* Add support to detect multiple digits
* Improve model to detect digits from complex images
* Add support to different languages to help users from all over the world

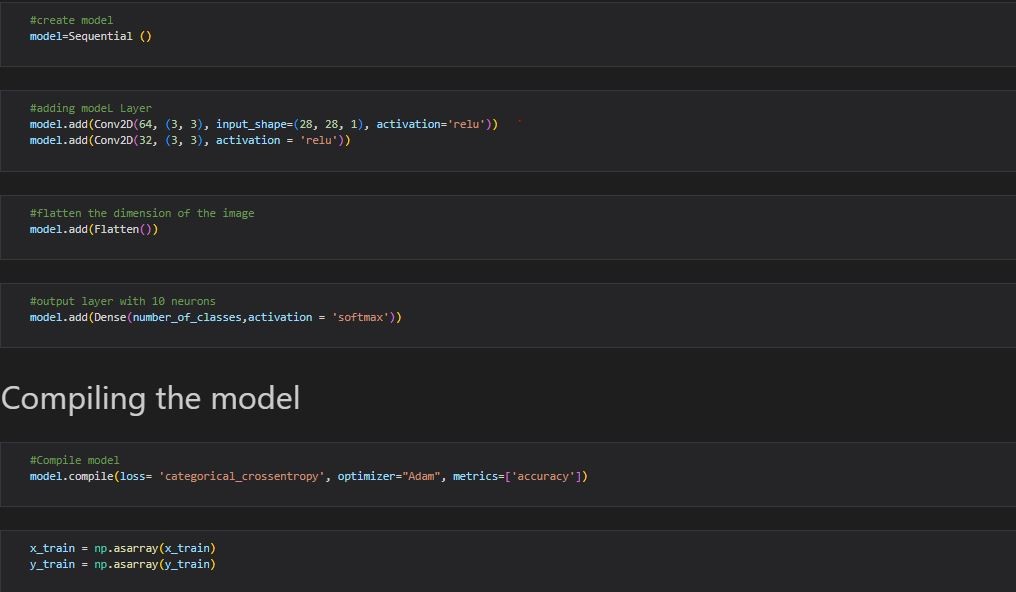
This project has endless potential and can always be enhanced to become better. Implementing this concept in the real world will beneﬁt several industries and reduce the workload on many workers, enhancing overall work efﬁciency.

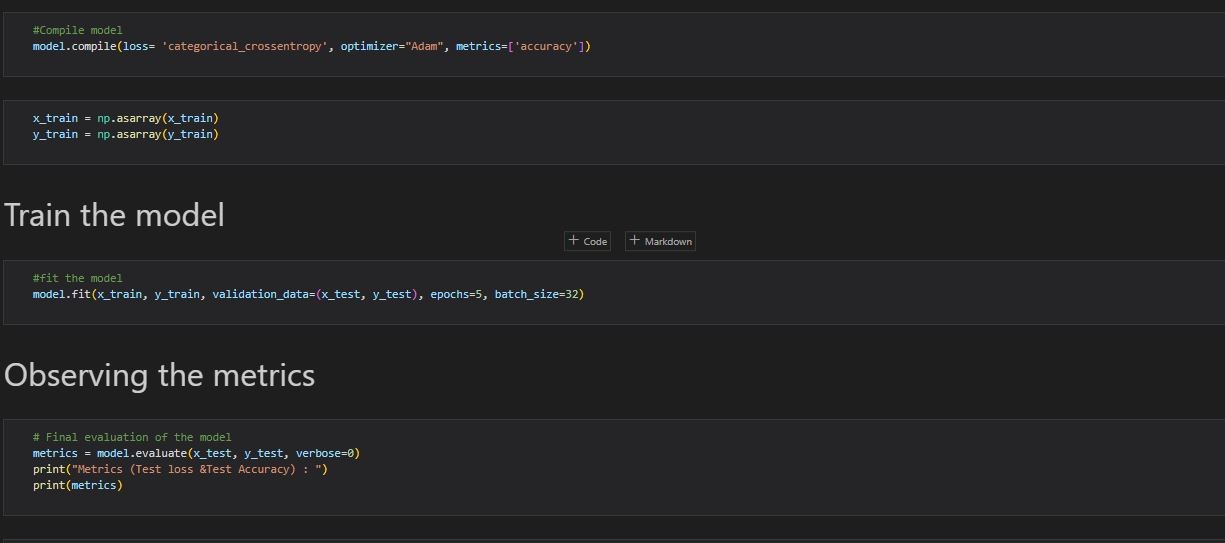
## APPENDIX

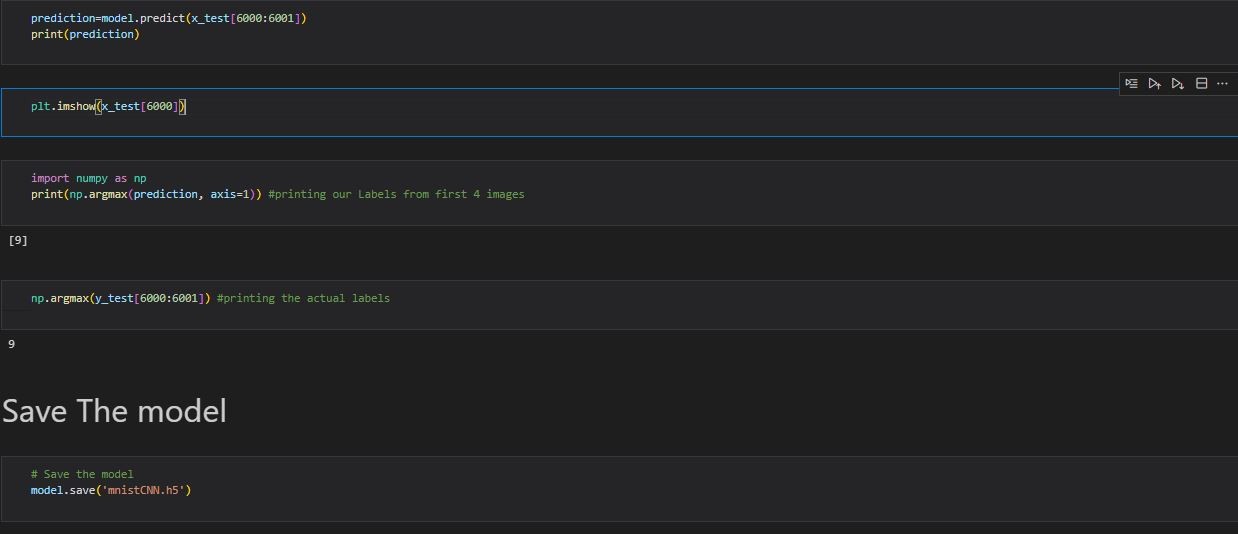
### SOURCE CODE

MODEL CREATION

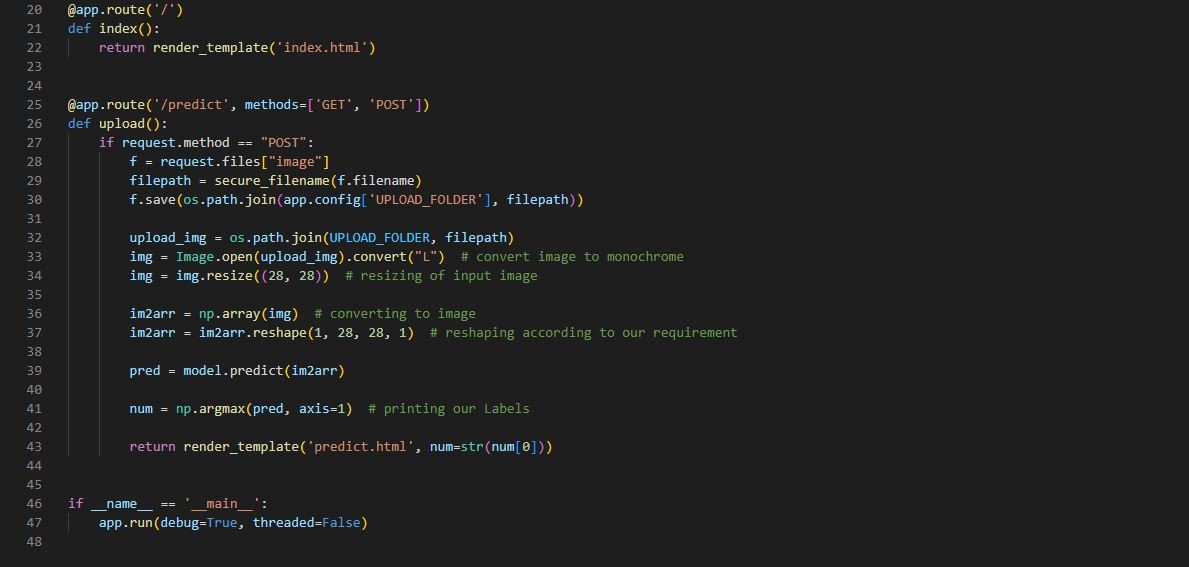




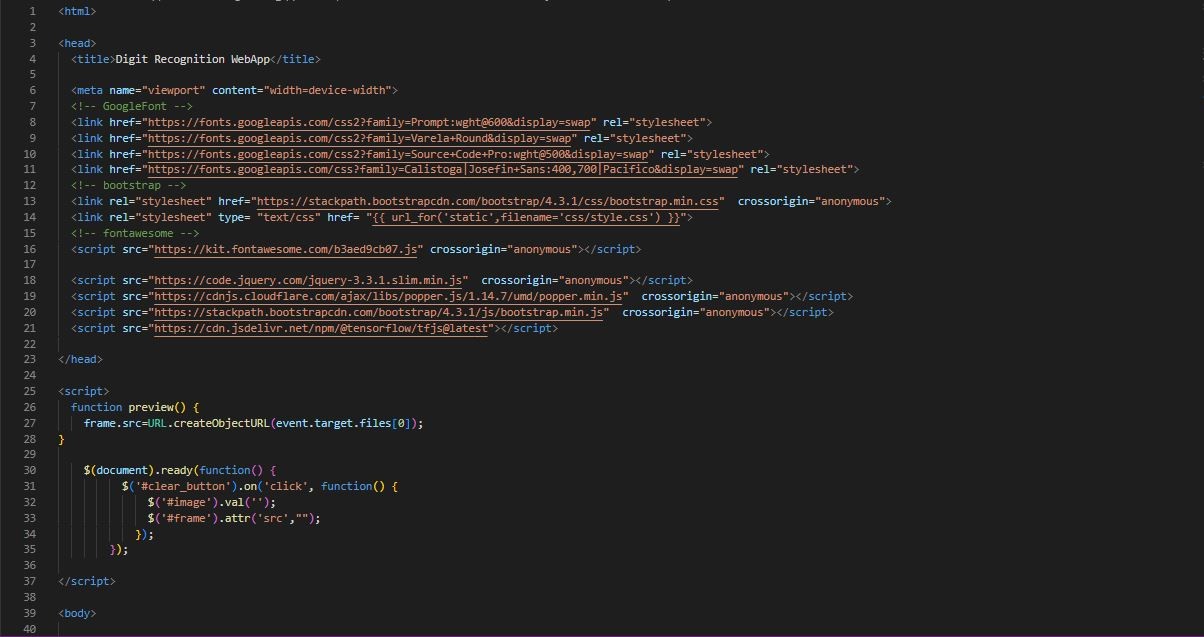


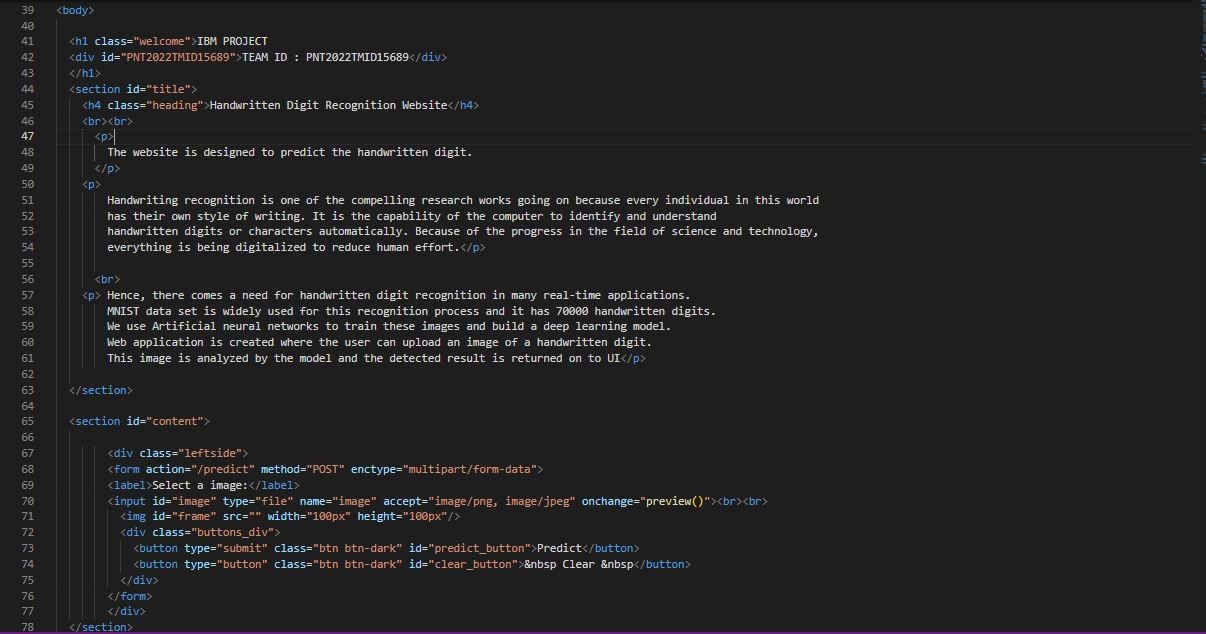


FLASK APP

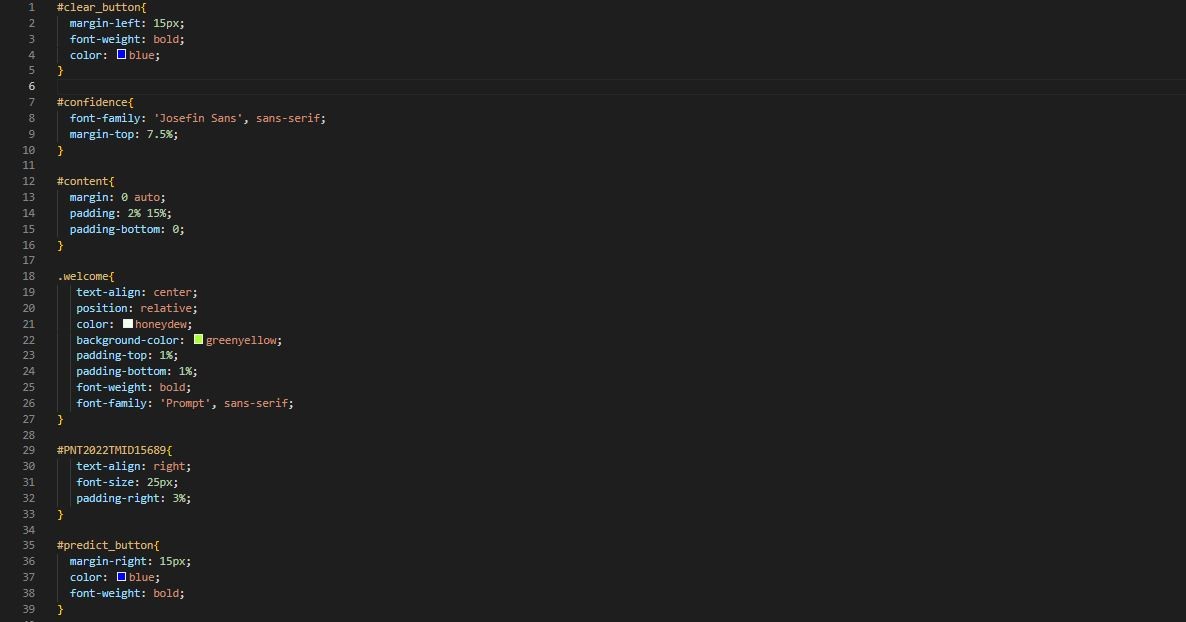


HOME PAGE (HTML)

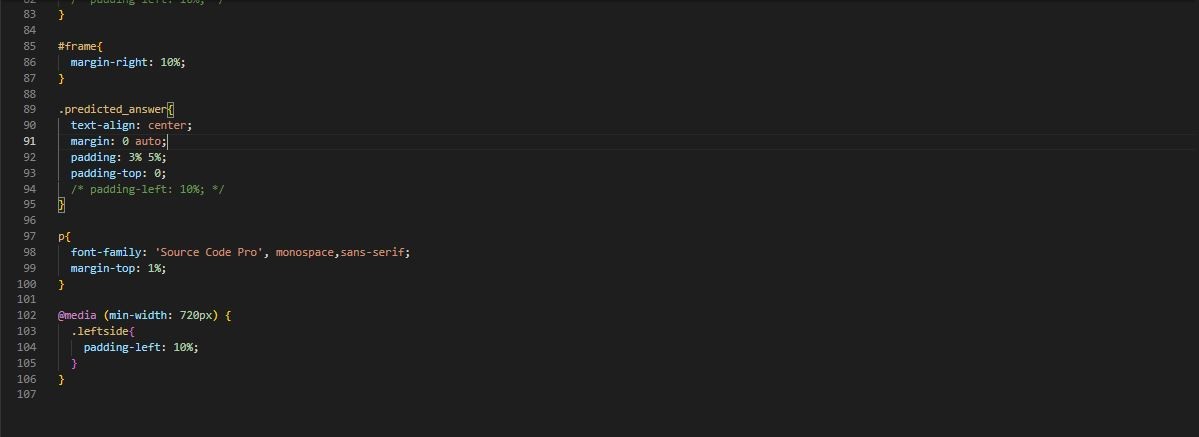




HOME PAGE (CSS)



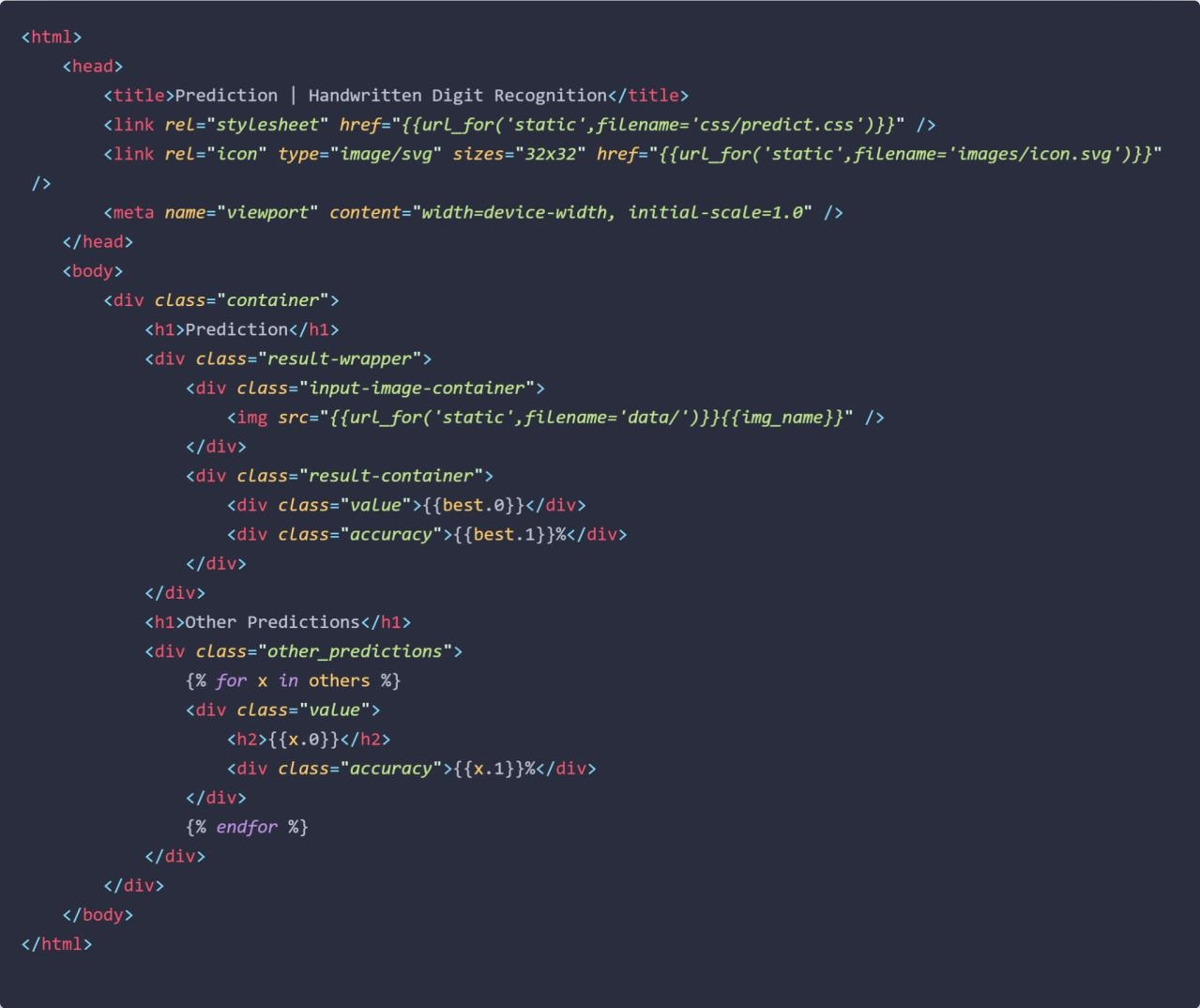




HOME PAGE (JS)



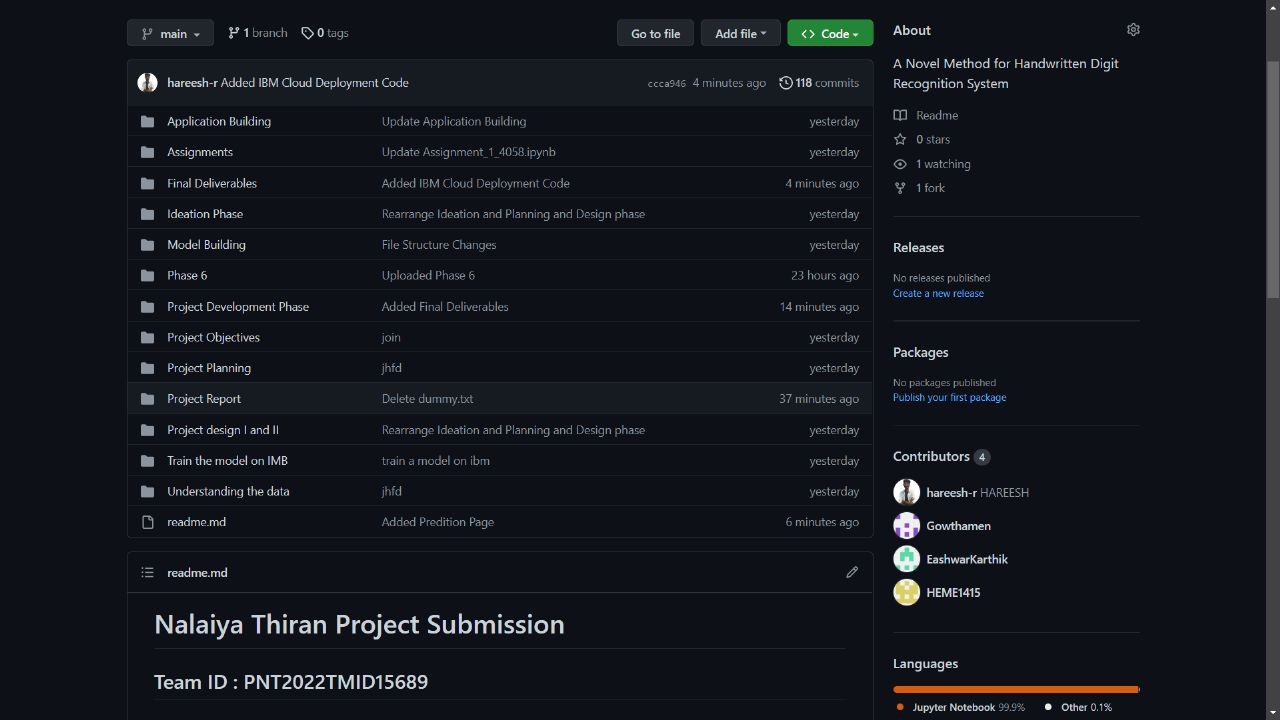
PREDICT PAGE (HTML)







**GITHUB**



**GITHUB LINK:**

<https://github.com/IBM-EPBL/IBM-Project-25872-1659975512>