

# **PROJECT REPORT**

## **A NOVEL METHOD FOR HANDWRITTEN DIGIT RECOGNITION SYSTEM**

**Submitted By**

***PNT2022TMID15485***

B. Siva Varun Kumar - 111519205004

D. Naga Datta - 111519205028

T. Krishna - 111519205053

T. Deepu Chowdary - 111519205052

# **TABLE OF CONTENTS**

## **1. INTRODUCTION**

1.1Project Overview

1.2Purpose

## **2. LITERATURE SURVEY**

2.1Existing problem

2.2References

2.3Problem Statement Definition

## **3. IDEATION & PROPOSED SOLUTION**

3.1 Empathy Map Canvas

3.2Ideation & Brainstorming

3.3Proposed Solution

3.4Problem Solution fit

## **4. REQUIREMENT ANALYSIS**

4.1Functional requirement

4.2Non-Functional requirements

## **5. PROJECT DESIGN**

5.1 Data Flow Diagrams

5.2Solution & Technical Architecture

5.3User Stories

## **6. PROJECT PLANNING & SCHEDULING**

6.1 Sprint Planning & Estimation

6.2 Sprint Delivery Schedule

## **7. CODING & SOLUTIONING (Explain the features added in the project along with code)**

## **8. TESTING**

8.1 Test Cases

8.2 User Acceptance Testing

## **9. RESULTS**

9.1 Performance Metrics

## **10. ADVANTAGES & DISADVANTAGES**

## **11. CONCLUSION**

## **12. FUTURE SCOPE**

## **13. APPENDIX**

Source Code

GitHub & Project Demo Link

# **1. INTRODUCTION**

## **1.1 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 recognize 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.

## **1.2 Purpose**

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

# **2. LITERATURE SURVEY**

## **2.1 Existing problem**

- ✓ The different architectures of CNN, hybrid CNN, CNN - RNN and CNNHMM models, and domain - specific recognition system, are not thoroughly inquired and evolutionary algorithms are not clearly explored for optimizing CNN learning parameters ,the number of layers, learning rate and kernel sizes of convolutional filters.

- ✓ The fluctuation of accuracies for handwritten digits was observed for 15 epochs by varying the hidden layers. There is no clear explanation given for observing variation in the overall classification accuracy by varying the number of hidden layers and batch size.

## 2.2 References

S.NO	Author Name	Paper Title	Journal/Conference title	Page No/Volume No	Year of Publication	Description
	Savita Ahlawat , Amit Choudhary, Anand Nayyar, Saurabh Singh and Byungun Yoon.	Improved Handwritten Digit Recognition Using Convolutional Neural Networks (CNN)	IEEE Sensors Journal		2020	In this paper, with the aim of improving the performance of handwritten digit recognition, they evaluated variants of a convolutional neural network to avoid complex preprocessing, costly feature extraction and a complex ensemble (classifier combination) approach

						of a traditional recognition system.
	Vijayalaxmi R Rudraswamimath, Bhavani shankar and Channasandra.	Handwritten Digit Recognition using CNN	International Journal of Innovative Science and Research Technology	Volume -4 Issue-6	2019	In this paper, the most widely used Machine learning algorithms, KNN, SVM, RFC and CNN have been trained and tested on the same data in order to acquire the comparison between the classifiers
	Fathma Siddique, Shadman Sakib and Md. Abu Bakr Siddique.	Recognition of Handwritten Digit using Convolutional Neural Network in Python with Tensorflow and Comparison of Performance for Various Hidden Layers	5th International Conference on Advances in Electrical Engineering (ICAEE)		2019	In this paper, they observed the variation of accuracies of CNN to classify handwritten digits for 15 epochs using various numbers of hidden layers and epochs and

						to make the comparison between the accuracies. For this performance evaluation of CNN, they performed the experiment using Modified National Institute of Standards and Technology( MN IST) dataset.
	Akanks ha Gupta, Ravindr a Pratap Narwari a and Madhav Singh	Review on Deep Learning Handwritten Digit Recognition using Convolutiona l Neural Network	International Journal of Recent Technology and Engineering (IJRTE)	Volume -9 Issue- 5	2021	In this paper, Object Character Recognition (OCR) is used on printed or documented letters to convert them into text. The database has training image database of 60,000 images and

						testing image database of 10,000 images. The KNN algorithm describes categorical value by making use of majority of votes of K - nearest neighbors, the K value used to differ here.
	Md. Anwar Hossain and Md. Mohon Ali	Recognition of Handwritten Digit using Convolutional Neural Network (CNN)	Global Journal of Computer Science and Technology: D Neural & Artificial Intelligence	Volume 19 Issue2	2019	The goal of this work will be to create a model that will be able to identify and determine the handwritten digit from its image with better accuracy using the concepts of Convolutional Neural Network and MNIST



						dataset. Later it can be extended for character recognition and real- time person's handwriting. The results can be made more accurate with more convolution layers and more number of hidden neurons.
--	--	--	--	--	--	---

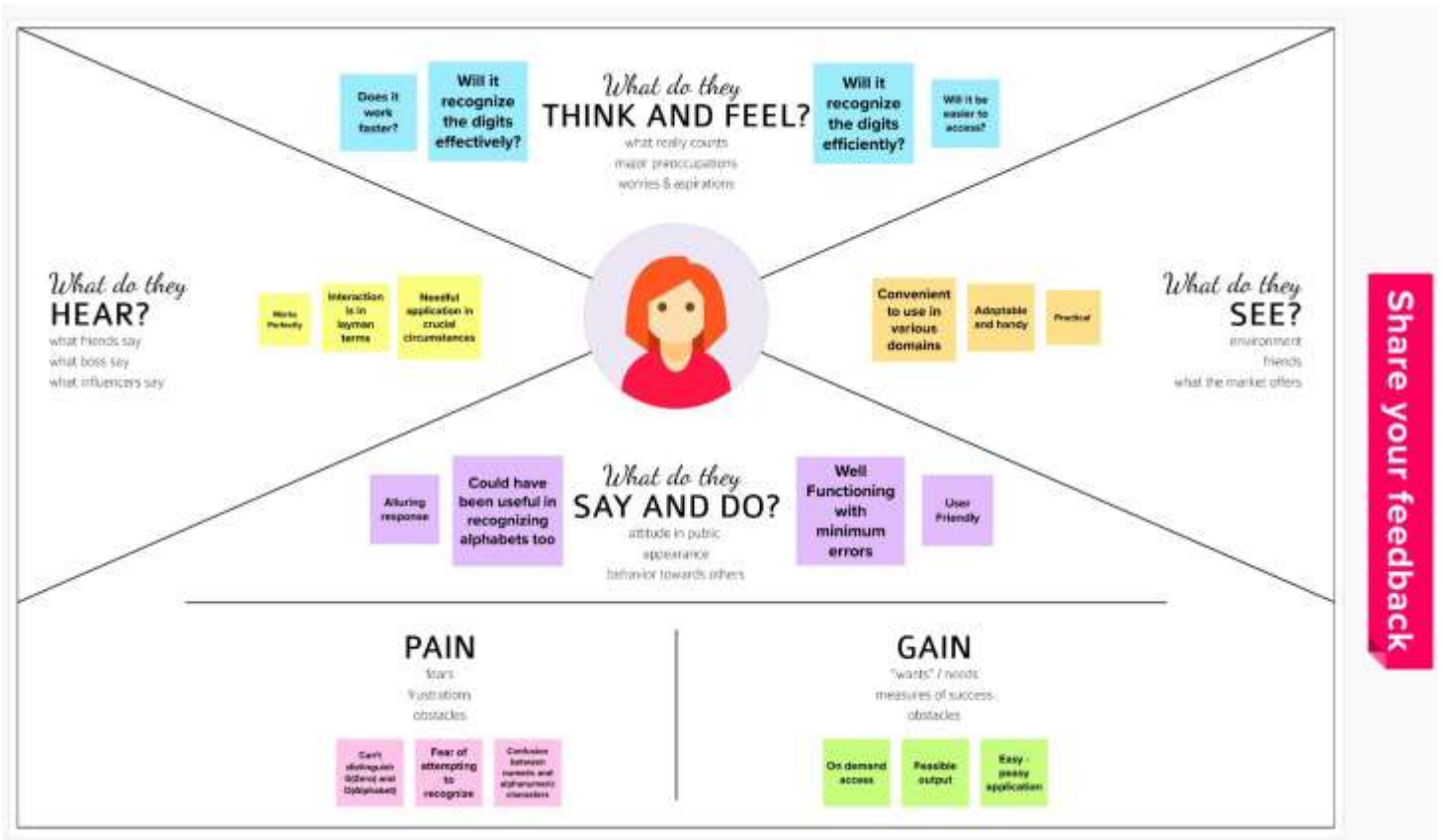
## 2.3 Problem Statement Definition

Handwriting recognition is one of the compelling research works going on because every individual in this world has their own style of writing. It is the capability of the computer to identify and understand handwritten digits or characters automatically. Because of the progress in the field of science and technology, everything is being digitalized to reduce human effort.

Hence, there comes a need for handwritten digit recognition in many real time applications. MNIST data set is widely used for this recognition process and it has 70000 handwritten digits. We use Artificial neural networks to train these images and build a deep learning model. Web application is created where the user can upload an image of a handwritten digit. this image is analyzed by the model and the detected result is returned on to UI(User Interface).

### 3. IDEATION & PROPOSED SOLUTION

#### 3.1 Empathy Map Canvas



#### 3.2 Ideation & Brainstorming



### 3.3 Proposed Solution

S.NO	Parameter	Description
1.	Problem Statement (Problem to be solved)	<p><b>Statement:</b> The handwritten digit recognition is the capability of computer applications to recognize the human handwritten digits.</p> <p><b>Description:</b> It is a hard task for the machine because handwritten digits are not perfect and can be made with many different shapes and sizes.</p>
2.	Idea / Solution description	<p>1. It is the capability of a computer to fetch the mortal handwritten integers from different sources like images, papers, touch defences.</p> <p>2. It allows user to translate all those signature and notes into electronic words in a text document format and this data only requires far less physical space than the storage of the physical copies.</p>

3.	Novelty / Uniqueness	Accurately recognize the digits rather than recognizing all the characters like OCR.
4.	Social Impact / Customer Satisfaction	<p>1. Artificial Intelligence developed the app called Handwritten digit Recognizer.</p> <p>2. It converts the written word into digital approximations and utilizes complex algorithms to identify characters before churning out a digital approximation.</p>
5.	Business Model (Revenue Model)	<p>1. This system can be integrated with traffic surveillance cameras to recognize the vehicle's number plates for effective traffic management.</p> <p>2. Can be integrated with Postal system to identify and recognize the pin-code details easily.</p>
6.	Scalability of the Solution	<p>1. Ability to recognise digits in more noisy environments.</p> <p>2. There is no limit in the number of digits it can be recognized.</p>

### 3.4 Problem Solution fit

<p><b>Define CS, fit into CC</b></p> <p><b>1. CUSTOMER SEGMENT(S)</b> <b>CS</b></p> <p>Who is your customer? i.e. working parents of 0-5 y.o. Kids</p> <p>Organizations who want to recognize the handwritten digits of people</p> <p>Example:</p> <ul style="list-style-type: none"> <li>✓ Post office,</li> <li>✓ Data entry offices,</li> <li>✓ Forensic Departments.</li> </ul>	<p><b>6. CUSTOMER CONSTRAINTS</b> <b>CC</b></p> <p>What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.</p> <p>In mobiles and laptop, there are possibilities for lack of stable internet connections and unavailability of devices. It is hard task for the machine to recognize the handwritten digits which are not perfect.</p>	<p><b>5. AVAILABLE SOLUTIONS</b> <b>AS</b></p> <p>Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros &amp; cons do these solutions have? i.e. pen and paper is an alternative to digital note taking. Already there are existing solutions available for handwritten recognition. But, most of them are inaccurate.</p> <p>The solution proposed by our system has more accuracy and it is efficient in recognition of manually written digits.</p> <p><b>Explore AS, differentiate</b></p>
<p><b>Focus on J&amp;P, tap into BE, understand RC</b></p> <p><b>2. JOBS-TO-BE-DONE / PROBLEMS</b> <b>J&amp;P</b></p> <p>Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.</p> <p>Jobs to be done: To identify the digits in the manually written forms, Cheques filled by people in banks, Phone numbers written manually in register notebook of hospitals.</p> <p>Problems: Dim lighting and weak eyesight</p>	<p><b>9. PROBLEM ROOT CAUSE</b> <b>RC</b></p> <p>What is the real reason that this problem exists? What is the backstory behind the need to do this job?</p> <p>i.e. customers have to do it because of the change in regulations.</p> <p>Handwritten digits are in different fonts and sizes, hard to recognize the digits due to various factors such as dim lighting, weakening eyesight.</p>	<p><b>7. BEHAVIOUR</b> <b>BE</b></p> <p>What does your customer do to address the problem and get the job done? i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)</p> <p>customer wants available devices with stable internet connection and quality cameras.</p> <p><b>Focus on J&amp;P, tap into BE, understand RC</b></p>

Identify strong TR & EM	<p><b>3. TRIGGERS</b> <span>TR</span></p> <p>What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.</p> <p>Advertisement in the market about the efficient recognition of digits.</p> <p>Articles about the achievements made by our project.</p>	<p><b>10. YOUR SOLUTION</b> <span>SL</span></p> <p>If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality.</p> <p>If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.</p> <p>Our solution aims to recognize handwritten digits using machine learning techniques thereby saving costs to the organization improving employee productivity.</p> <p>In our model we use AlexNet, which is one of the CNN architectures. AlexNet allows for multi-GPU training by putting half of the model's neurons on one GPU and the other half on another GPU. Not only does this mean that a bigger model can be trained, but it also cuts down on the training time. It also reduces the overfitting problem by Data Augmentation and Dropout.</p>	<p><b>8. CHANNELS of BEHAVIOUR</b> <span>CH</span></p> <p><b>8.1 ONLINE</b></p> <p>What kind of actions do customers take online? Extract online channels from #7</p> <p>Requires Stable internet connection for image processing.</p> <p><b>8.2 OFFLINE</b></p> <p>What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.</p> <p>Obtain modern electronic devices and check they are working</p>	Identify strong TR & EM
	<p><b>4. EMOTIONS: BEFORE / AFTER</b> <span>EM</span></p> <p>How do customers feel when they face a problem or a job and afterwards?</p> <p>i.e. lost, insecure &gt; confident, in control - use it in your communication strategy &amp; design.</p> <p>Defects are common and our project is not an exception</p> <p>When the system failed to recognize the digit,</p> <p>Customer Mentality:</p> <p>Before:(Failure)</p> <p>We would give guarantee that it would work most of the time and if any error occurs, they can contact us at any time.</p> <p>So, customers can feel at ease.</p> <p>After:(Failure)</p> <p>They have no need to panic when the failure occurs</p> <p>They can easily contact us to rectify the error.</p> <p>We would solve the defect as soon as possible.</p>			

## 4. REQUIREMENT ANALYSIS

### 4.3 Functional requirement

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Input Correlation	Digital image correlation is a technique that combines image registration and tracking methods for accurate 2D measurements of changes in images and recognizes the characters from the images.
FR-2	Data Preparation	Data preparation is the process of preparing raw data so that it is suitable for further processing and analysis.
FR-3	Feature Extraction	Feature extraction refers to the process of transforming raw data into numerical features that can be processed while preserving the information in the original data set.
FR-4	Character Classification	In character classification phase, the attributes of the data in the picture are compared to the classes in the database to determine in which class the picture belongs to.

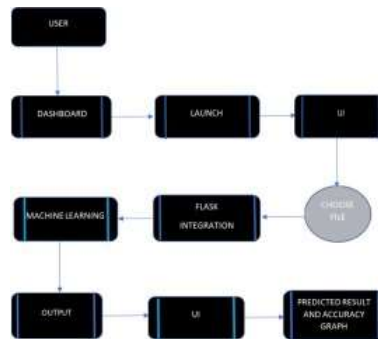
## 4.4 Non-Functional requirements

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	<b>Usability</b>	Handwritten digit recognition is one of the major important issues in pattern recognition applications. Some of the applications for digit recognition include data entry forms, Bank check processing etc.,
NFR-2	<b>Security</b>	The applications of handwritten digit recognition can be used in the banking sector where it can be used to maintain the security pin numbers safely. It can be also used for blind-people by using sound output.
NFR-3	<b>Reliability</b>	Reliability indicates the probability that the system will perform its intended function for a larger period of sufficient time and also it will operate in a secured environment without any failures.
NFR-4	<b>Performance</b>	The standard implementations of neural networks achieve an accuracy of approximately (98–99)
		percent in correctly classifying the handwritten digits.
NFR-5	<b>Availability</b>	The features for handwritten digit recognition have been Acquainted. These features are based on shape analysis of the digit image and extract slant or slope information. They are effective in obtaining good recognition of accuracy.
NFR-6	<b>Scalability</b>	The scalability in the task of handwritten digit recognition, using a classifier, has great importance and it makes use of online handwriting recognition on computer tablets, recognizing zip codes on mail for postal mail sorting, processing bank check amounts, numeric entries in forms filled up manually(for example - tax forms) and so on.

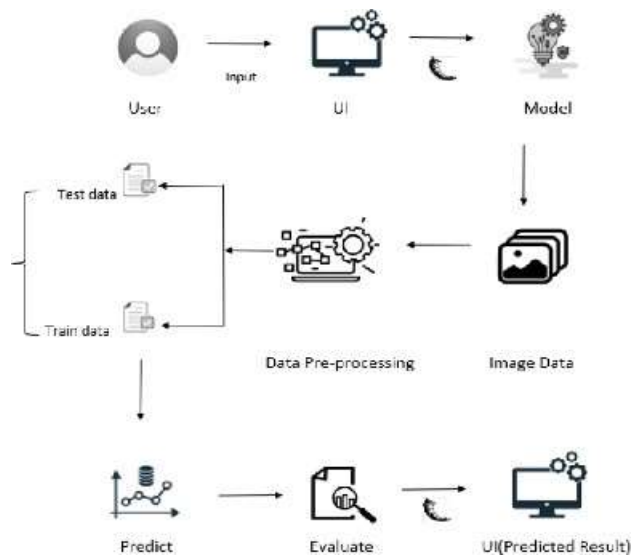
## 5. PROJECT DESIGN

### 5.1 Data Flow Diagrams



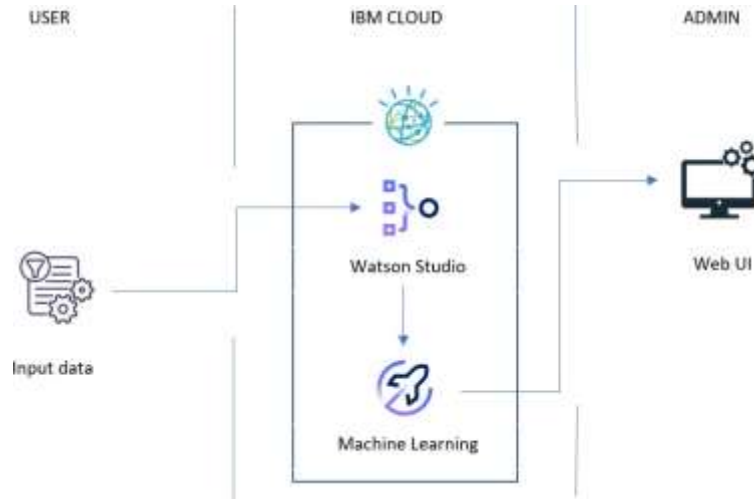
### 5.2 Solution & Technical Architecture

#### Solution Architecture





# Technology Architecture



**Table-1 : Components & Technologies:**

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Web UI	HTML, CSS, JavaScript
2.	Application Logic-1	Model is built	Python
3.	Application Logic-2	Python model is deployed	IBM Watson Studio
4.	File Storage	Predicted outputs of the image are stored in a local folder.	Local Filesystem
5.	Machine Learning Model	To predict the image uploaded by the user.	Image Recognition Model
6.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Flask Cloud Server Configuration : IBM Watson Studio	Local, Cloud Foundry.

**Table-2: Application Characteristics:**

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	Flask
2.	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	e.g. SHA-256, Encryptions, IAM Controls, OWASP etc.
3.	Scalable Architecture	High workload can be supported without undergoing any major changes.	Technology used in the architecture is that with Python and the IBM cloud.
4.	Availability	Readily available enables the IT Infrastructure to function when some of the components fail.	Technology used is IBM cloud.
5.	Performance	Performance technology is a field which uses various tools, processes and procedures in a systematic and efficient manner to improve the desired outcomes of individuals and organizations.	Technology used is python.

## 5.3 User Stories

### User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	Dashboard	USN-1	As a user, they can see the information regarding the prediction of handwritten digit recognition.	I can see the information regarding digit recognition.	High	Sprint 1
	Launch	USN-2	On clicking the launch button, it will redirect the user to a page where the images to be predicted can be uploaded.	I can see the launch button.	High	Sprint 1
	Upload	USN-3	Users can select the image from the local storage.	I can upload the image.	High	Sprint 2
	Predict	USN-4	Once the image is uploaded, it will predict the respective image.		High	Sprint 3
	Display	USN-5	The predicted image will be displayed with the accuracy chart.	I can see the result with accuracy.	High	Sprint 4

## 6. PROJECT PLANNING & SCHEDULING

### 6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Dashboard	USN-1	As a user, they can see the information regarding the prediction of handwritten digit recognition.	2	High	B S Varun Kumar D Naga Datta T Deepu Chowdary T Krishna
Sprint-1	Launch	USN-2	On clicking the launch button, it will redirect the user to a page where the images to be predicted can be uploaded.	2	High	B S Varun Kumar D Naga Datta T Deepu Chowdary T Krishna
Sprint-2	Upload	USN-3	Users can select the image from the local storage.	2	High	B S Varun Kumar D Naga Datta
Sprint-3	Predict	USN-4	Once the image is uploaded, it will predict the respective image.	2	High	B S Varun Kumar D Naga Datta
Sprint-4	Display	USN-5	The predicted image will be displayed with the accuracy chart.	2	High	B S Varun Kumar D Naga Datta T Deepu Chowdary T Krishna

## 6.2 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

he

## 7. CODING & SOLUTIONING (Explain the features added in the project along with code)

```
File Edit Selection Find View Goto Tools Project Preferences Help
1 from flask import Flask, render_template, request
2 from scipy.misc import image, imread, imresize
3 import numpy as np
4 import keras.models
5 import re
6 import base64
7
8 import sys
9 import os
10 sys.path.append(os.path.abspath("../model"))
11 from load import *
12
13 app = Flask(__name__)
14 global model, graph
15 model, graph = init()
16
17 @app.route('/')
18 def index():
19     return render_template("index.html")
20
21 @app.route('/predict/', methods=['GET', 'POST'])
22 def predict():
23     # get data from drawing canvas and save as image
24     parnelimage(request.get_data())
25
26     # read parnel image back in 8-bit, black and white mode (L)
27     x = imread('output.png', mode='L')
28     x = np.invert(x)
29     x = imresize(x, (28,28))
30
31     # reshape image data for use in neural network
32     x = x.reshape(1,28,28,1)
33     with graph.as_default():
34         out = model.predict(x)
35         print(out)
36         print(np.argmax(out, axis=-1))
37         response = np.array_str(np.argmax(out, axis=-1))
38         return response
39
40 def parnelimage(imgData):
41     # parse canvas bytes and save as output.png
42     imgstr = re.search(b'base64,(.*)', imgData).group(1)
43     with open('output.png', 'wb') as output:
44         output.write(base64.decodebytes(imgstr))
45
46 if __name__ == '__main__':
47     app.debug = True
48     port = int(os.environ.get("PORT", 5000))
49     app.run(host='0.0.0.0', port=port)
```

## 8. TESTING

### 8.1 Test Cases

Test case ID	Feature Type	Component	Test Scenario	Expected Result	Actual Result	Status
Homepage_TC_OO1	Functional	Home Page	Verify user is able to see the Homepage when clicked on the link	Home Page should be displayed.	Working as expected	Pass
Homepage_TC_OO2	UI	Home Page	Verify the UI elements in Homepage	Application should show below UI elements: a.choose file button b.predict button c.clear button	Working as expected	Pass
Homepage_TC_OO3	Functional	Home Page	Verify user is able to choose file from the local system and click on predict	Choose file popup screen must be displayed and user should be able to click on predict button	Working as expected	Pass
Homepage_TC_OO4	Functional	Home page	Verify user able to select invalid file format	Application won't allow to attach formats other than ".png, .jiff, .pjp, .jpeg, .jpg, .jpeg"	Working as expected	Pass
Predict_TC_OO5	Functional	Predict page	Verify user is able to navigate to the predict to and view the predicted result	User must be navigated to the predict page and must view the predicted result	Working as expected	Pass

## 8.2 User Acceptance Testing

### Defect Analysis

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	0	0	0	0	0
Duplicate	0	0	0	0	0
External	0	0	0	0	0
Fixed	0	0	0	0	0
Not Reproduced	0	0	0	0	0
Skipped	0	0	0	0	0
Won't Fix	0	0	0	0	0
Totals	0	0	0	0	0

### Test Case Analysis

Section	Total Cases	Not Tested	Fail	Pass
Client Application	5	0	0	5
Security	5	0	0	5
Final Report Output	5	0	0	5
Performance	5	0	0	5

## 9. RESULTS

### 9.1 Performance Metrics

Model Summary:

```
Model: "sequential"
```

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 26, 26, 64)	640
conv2d_1 (Conv2D)	(None, 24, 24, 32)	18464
flatten (Flatten)	(None, 18432)	0
dense (Dense)	(None, 10)	184330

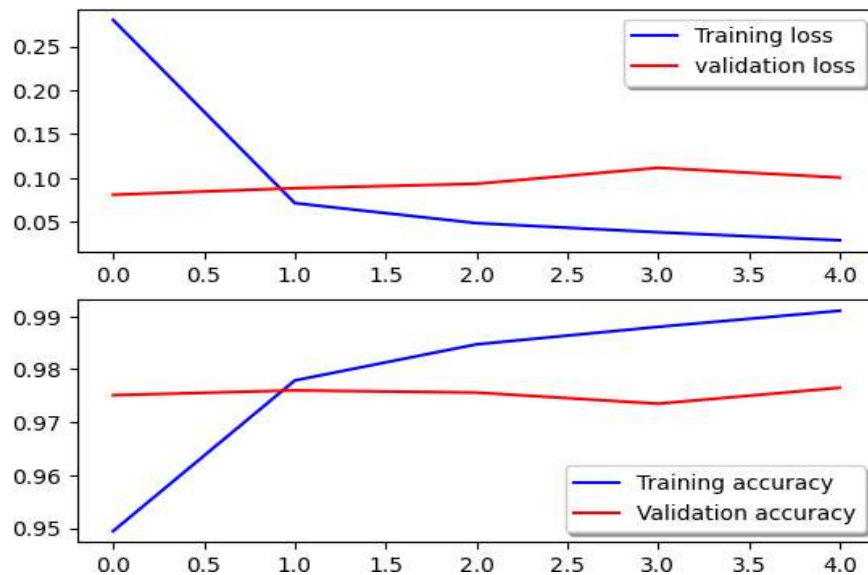
```
=====
```

Total params: 203,434  
Trainable params: 203,434  
Non-trainable params: 0

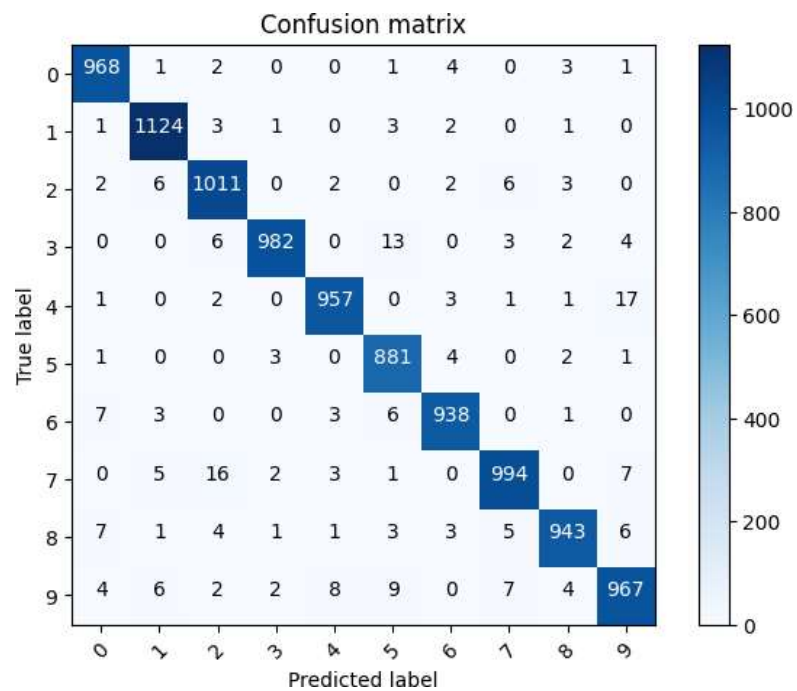
```
=====
```

None

Accuracy:



## Confusion Matrix:

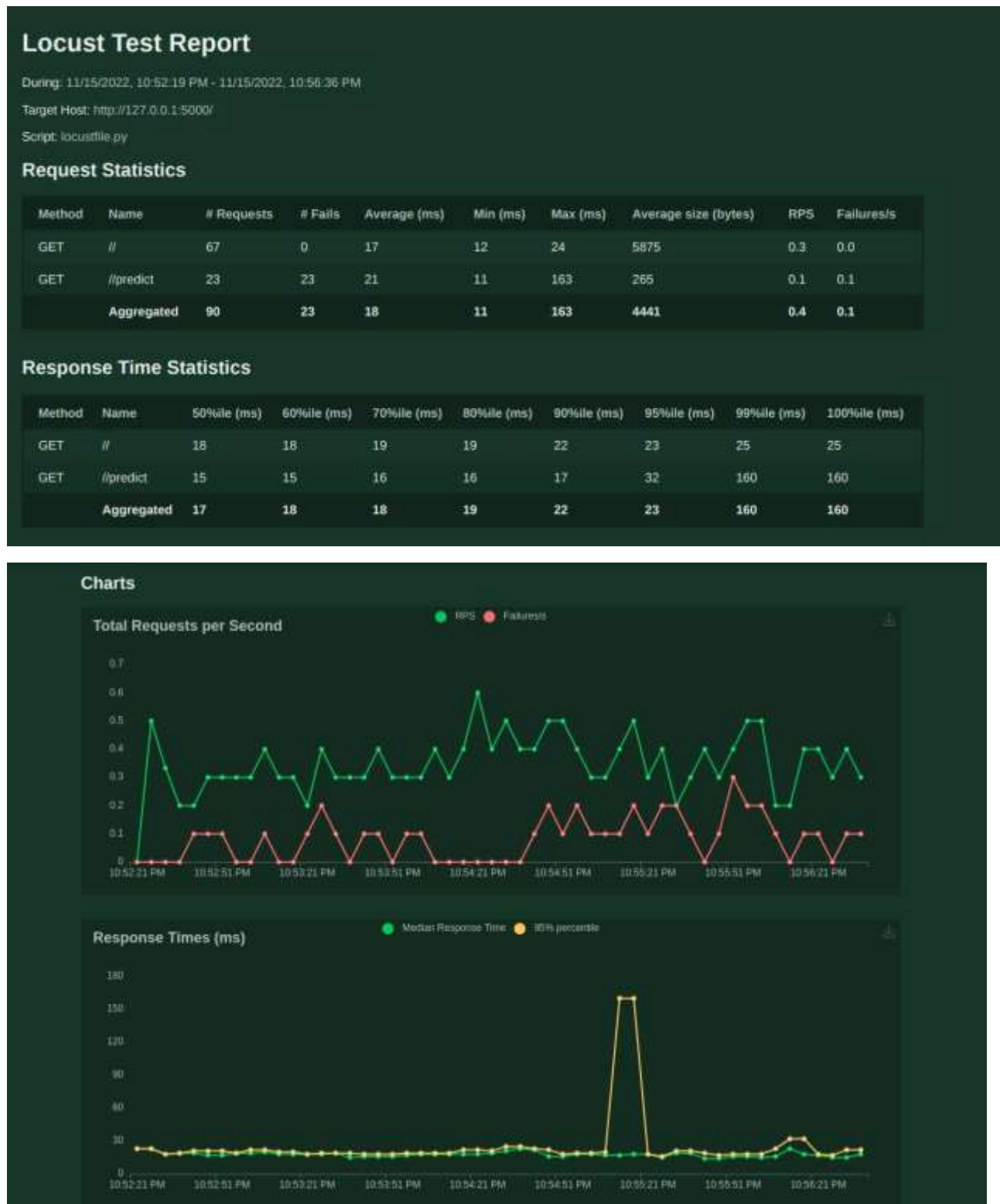


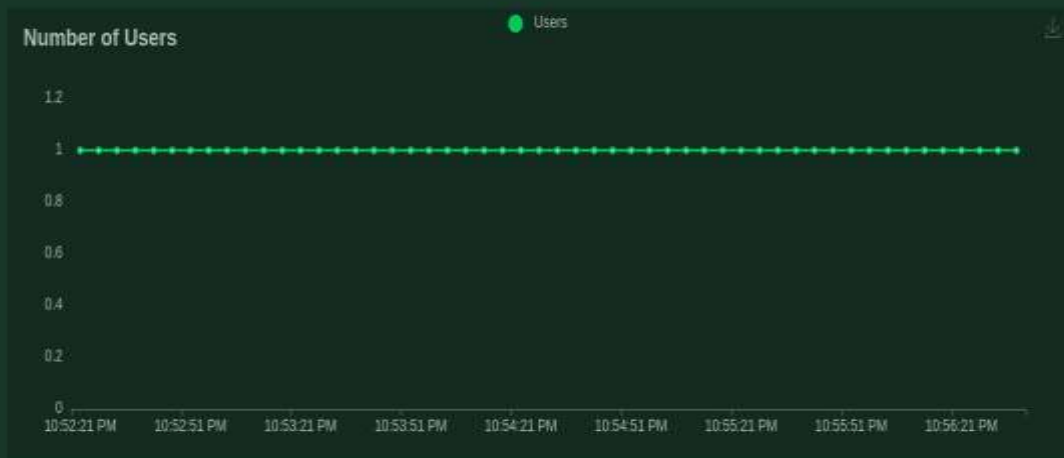
## Classification Report:

	precision	recall	f1-score	support
0	0.98	0.99	0.98	980
1	0.98	0.99	0.99	1135
2	0.97	0.98	0.97	1032
3	0.99	0.97	0.98	1010
4	0.98	0.97	0.98	982
5	0.96	0.99	0.97	892
6	0.98	0.98	0.98	958
7	0.98	0.97	0.97	1028
8	0.98	0.97	0.98	974
9	0.96	0.96	0.96	1009
accuracy			0.98	10000
macro avg	0.98	0.98	0.98	10000
weighted avg	0.98	0.98	0.98	10000



## Performance Metrics Result:





## Final ratio

### Ratio per User class

- 100.0% QuickstartUser
  - 80.0% index
  - 20.0% about

### Total ratio

- 100.0% QuickstartUser
  - 80.0% index
  - 20.0% about

## **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 high performance server for faster predictions.
- ✓ Prone to occasional errors.

## **11.CONCLUSION**

This project demonstrated a web application that uses machine learning to recognize 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.

## **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.

## **13.APPENDIX**

**Source Code**

**HTML AND CSS:**

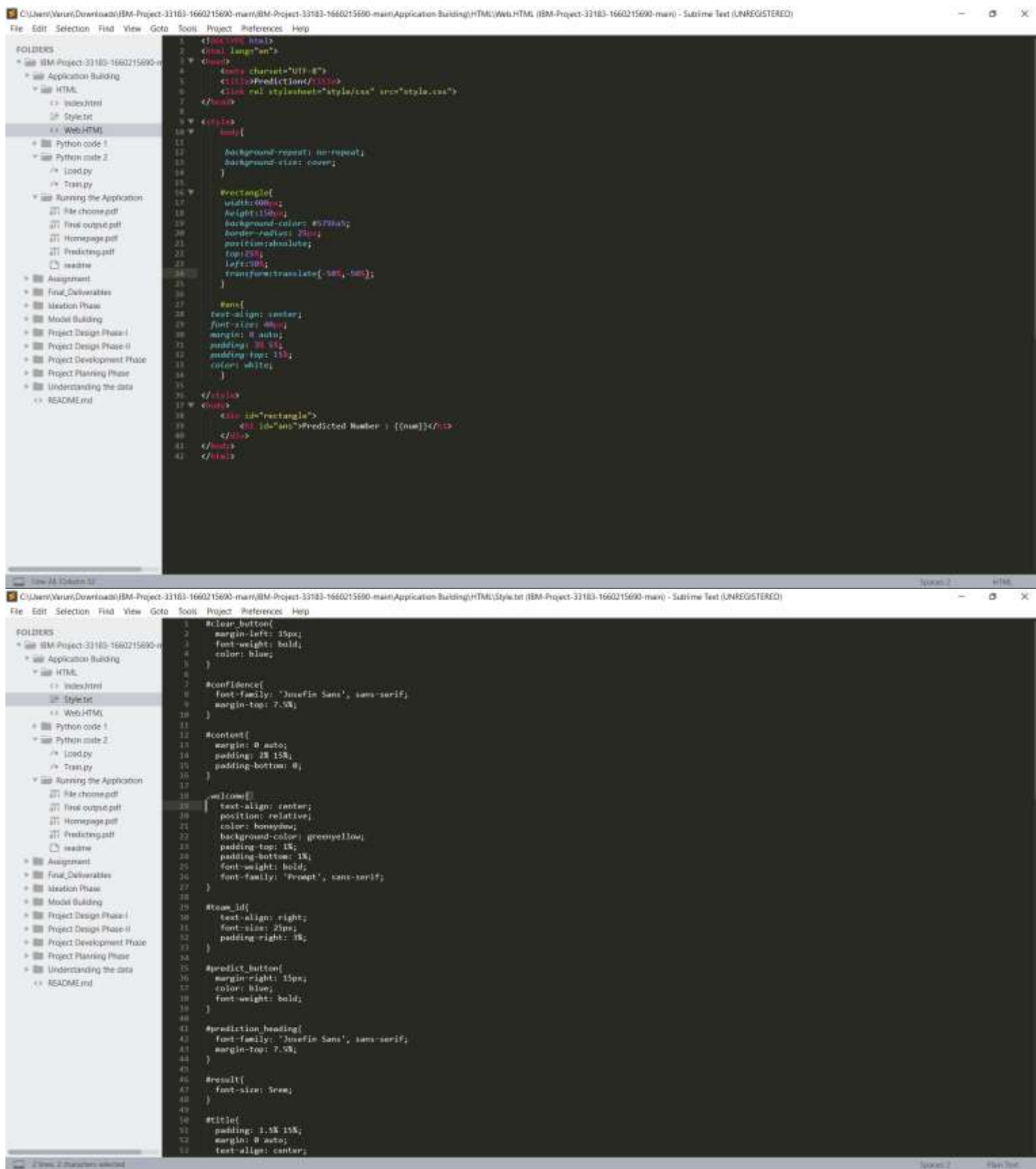
**index.html:**

```
C:\Users\Varun\Downloads\IBM-Project-33183-1660215600-main\IBM-Project-33183-1660215600-main\Application Building\HTML\index.html (IBM-Project-33183-1660215600-main) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

1 <html>
2
3 <head>
4 <title>Digit Recognition WebApp</title>
5
6 <meta name="viewport" content="width=device-width">
7 <!-- Google font -->
8 <link href="https://fonts.googleapis.com/css2?family=Prompt:wght@400&display=swap" rel="stylesheet">
9 <link href="https://fonts.googleapis.com/css2?family=VarelaRound&display=swap" rel="stylesheet">
10 <link href="https://fonts.googleapis.com/css2?family=SourceCodePro:wght@400&display=swap" rel="stylesheet">
11 <link href="https://fonts.googleapis.com/css2?family=Calistoga|Josefin+Sans:400,700|Pacifico&display=swap" rel="stylesheet">
12 <!-- Bootstrap -->
13 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbBKlrgt/FOcVYVhJlrp0p76tH6pKd/7Ut+FzopV/iTbq6wz7HvYj/28LrQ==" crossorigin="anonymous">
14 <link rel="stylesheet" type="text/css" href="style.css">
15 <!-- Font awesome -->
16 <script src="https://kit.fontawesome.com/hand9icb07.js" crossorigin="anonymous"></script>
17
18 <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+9654zD80zT4oYwuETgS3425Xdf4QqauNy7p76p2zgZ3d2F7A330" crossorigin="anonymous"></script>
19 <script src="https://cdn.jsdelivr.net/npm/popper.js@1.14.7/umd/popper.min.js" integrity="sha384-Uh2uYkmJ48PSzfj4C1YjE732FtgoK0E/t6KwTb8BBXPFofVMC8XRKFab40Eq1w==" crossorigin="anonymous"></script>
20 <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-ggOyR0iXCbBKlrgt/FOcVYVhJlrp0p76tH6pKd/7Ut+FzopV/iTbq6wz7HvYj/28LrQ==" crossorigin="anonymous"></script>
21 <script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs@latest"></script>
22
23 </head>
24
25 <body>
26 <function preview() {
27   frame.src = URL.createObjectURL(event.target.files[0]);
28 }
29
30 <[document].ready(function() {
31   $('#clear_button').on('click', function() {
32     $('#image').val('');
33     $('#frame').attr('src','');
34   });
35 });
36
37 </script>
38
39 <div>
40 <div class="welcome">IBM PROJECT</div>
41 </div>
42
43 <section id="title">
44 <div class="heading">Handwritten Digit Recognition</div>
45 </div>
46 </div>
47 The website is designed to predict the handwritten digit.</div>
48 </div>
49
50 </body>
51 </html>
```

```
C:\Users\Varun\Downloads\IBM-Project-33183-1660215600-main\IBM-Project-33183-1660215600-main\Application Building\HTML\index.html (IBM-Project-33183-1660215600-main) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

51 </div>
52
53 <section id="content">
54
55 <div class="leftside">
56 <form action="/predict" method="POST" enctype="multipart/form-data">
57 <div>Select a Image</div>
58 <input id="image" type="file" name="image" accept="image/png, image/jpeg" onchange="preview()"></div>
59 <div id="frame" src="" width="300px" height="100px"></div>
60 <div class="button_div">
61 <button type="submit" class="btn btn-dark" id="predict_button" onclick="predict.html">Predict</button>
62 <button type="button" class="btn btn-dark" id="clear_button">Clear</button>
63 </div>
64 </form>
65 </div>
66 </section>
67
68 </body>
69
70 </html>
```



## Predict.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Prediction</title>
</head>

<style>
  body{
    background-image: url('static/images/bc1.jpg');
```

```

    background-repeat: no-repeat;
    background-size: cover;
}

#rectangle{
    width:600px;
    height:150px;
    background-color: #000000;
    border-radius: 25px;
    position:absolute;
    box-shadow: 0px 0px 10px 5px white;
    top:25%;
    left:50%;
    transform:translate(-50%,-50%);
}

#num{
text-align: center;
font-size: 30px;
margin: 0 auto;
padding: 3% 5%;
padding-top: 8%;
color: white;
}

</style>

<body>

    <div id="rectangle">
        <h1 id="num">Predicted Number is {{num}}</h1>
    </div>

</body>
</html>

```

## Style.css

```

#clear_button{
    margin-left: 15px;
    font-weight: bold;
    color: rgb(0, 174, 255);
}

#confidence{
    font-family: 'Josefin Sans', sans-serif;

```



```

    margin-top: 7.5%;
}

#content{
    margin: 0 auto;
    padding: 2% 15%;
    padding-bottom: 0;
}

.welcome{
    text-align: center;
    position: relative;
    color: rgb(0, 32, 112);
    background-color: skyblue;
    padding-top: 1%;
    padding-bottom: 1%;
    font-weight: bold;
    font-family: 'Bookman', 'URW Bookman L', serif;
}

#team_id{
    text-align: right;
    font-size: 25px;
    padding-right: 3%;
}

#predict_button{
    margin-right: 15px;
    color: rgb(0, 255, 72);
    font-weight: bold;
}

#prediction_heading{
    font-family: 'Josefin Sans', sans-serif;
    margin-top: 7.5%;
}

#result{
    font-size: 5rem;
}

#title{
    padding: 1.5% 15%;
    margin: 0 auto;
    text-align: center;
}

.btn {

```

```

    font-size: 15px;
    padding: 10px;
    /* -webkit-appearance: none; */
    background: #eee;
    border: 1px solid #888;
    margin-top: 20px;
    margin-bottom: 20px;
}

.buttons_div{
    margin-bottom: 30px;
    margin-right: 80px;
}

.heading{
    font-family: "American Typewriter", serif;
    font-weight: 700;
    font-size: 2rem;
    display: inline;
}

.leftside{
    text-align: center;
    margin: 0 auto;
    margin-top: 2%;
    /* padding-left: 10%; */
}

#frame{
    margin-right: 10%;
}

.predicted_answer{
    text-align: center;
    margin: 0 auto;
    padding: 3% 5%;
    padding-top: 0;
    /* padding-left: 10%; */
}

h1{
    text-align: center;
    color: aliceblue;
    padding: 100px 50px 65px 100px;
}

@media (min-width: 720px) {
    .leftside{

```

```
padding-left: 10%;  
}  
}
```

## FLASK:

### app.py:

```
import numpy as np  
import os  
from PIL import Image  
from flask import Flask, request, render_template, url_for  
from werkzeug.utils import secure_filename, redirect  
#from gevent.pywsgi import WSGIServer  
from keras.models import load_model  
from keras.preprocessing import image  
from flask import send_from_directory  
  
UPLOAD_FOLDER = 'D:/ibm/data'  
  
app = Flask(__name__)  
app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER  
  
model = load_model("./DigitRecog_IBM_model/mnistCNN.h5")  
  
@app.route('/')  
def index():  
    return render_template('index.html')  
  
@app.route('/predict', methods=['GET', 'POST'])  
def upload():  
    if request.method == "POST":  
        f = request.files["image"]  
        filepath = secure_filename(f.filename)  
        f.save(os.path.join(app.config['UPLOAD_FOLDER'], filepath))  
  
        upload_img = os.path.join(UPLOAD_FOLDER, filepath)  
        img = Image.open(upload_img).convert("L") # convert image to monochrome  
        img = img.resize((28, 28)) # resizing of input image  
  
        im2arr = np.array(img) # converting to image  
        im2arr = im2arr.reshape(1, 28, 28, 1) # reshaping according to our requirement  
  
        pred = model.predict(im2arr)
```

```
num = np.argmax(pred, axis=1) # printing our Labels
```

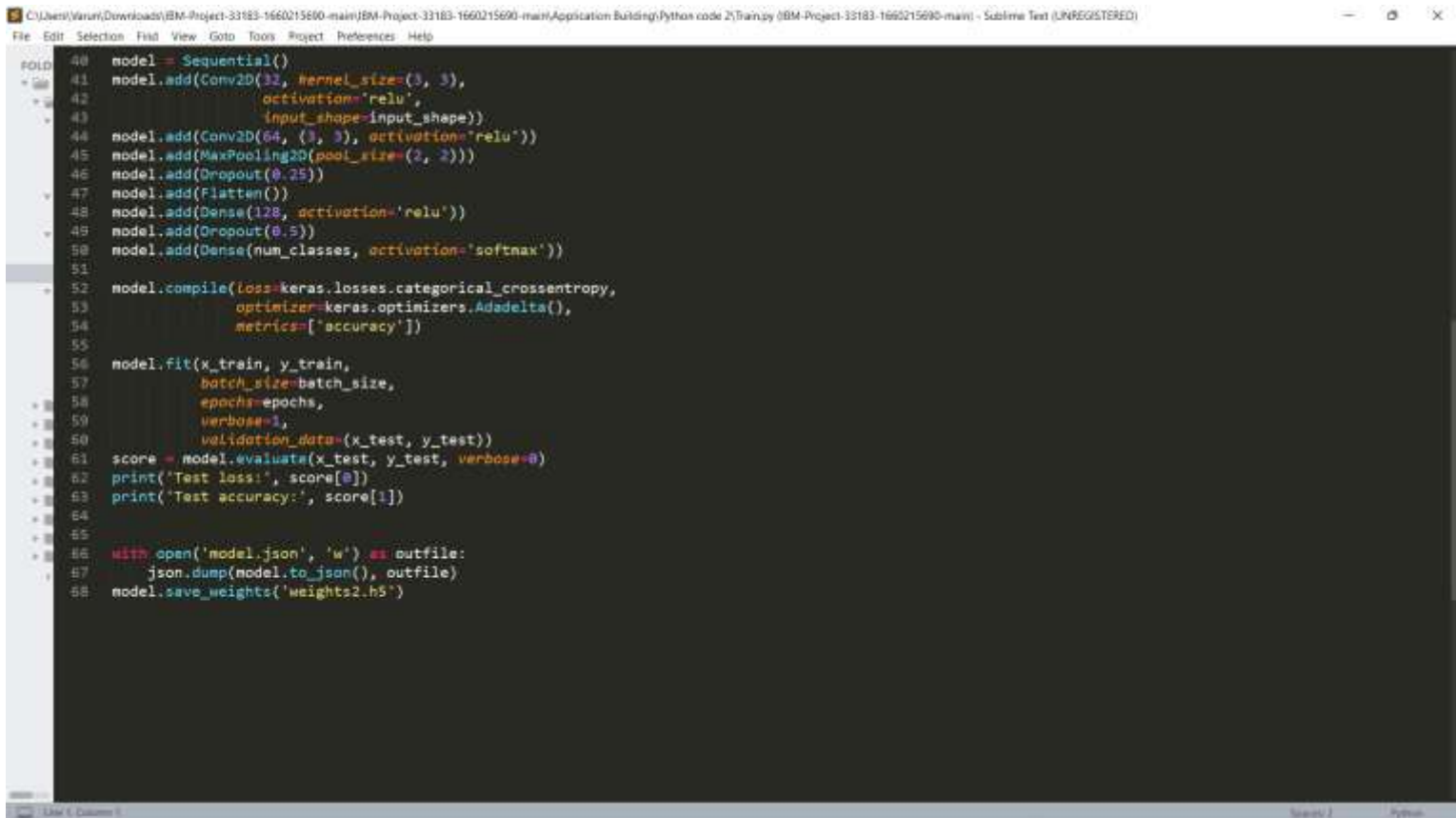
```

        return render_template('predict.html', num=str(num[0]))

if __name__ == '__main__':
    app.run(debug=True, threaded=False)

```

## MODEL CREATION:



The screenshot shows a Sublime Text editor window with a Python script for creating and training a Keras model. The script is as follows:

```

40 model = Sequential()
41 model.add(Conv2D(32, kernel_size=(3, 3),
42                 activation='relu',
43                 input_shape=input_shape))
44 model.add(Conv2D(64, (3, 3), activation='relu'))
45 model.add(MaxPooling2D(pool_size=(2, 2)))
46 model.add(Dropout(0.25))
47 model.add(Flatten())
48 model.add(Dense(128, activation='relu'))
49 model.add(Dropout(0.5))
50 model.add(Dense(num_classes, activation='softmax'))
51
52 model.compile(loss=keras.losses.categorical_crossentropy,
53              optimizer=keras.optimizers.Adadelta(),
54              metrics=['accuracy'])
55
56 model.fit(x_train, y_train,
57         batch_size=batch_size,
58         epochs=epochs,
59         verbose=1,
60         validation_data=(x_test, y_test))
61 score = model.evaluate(x_test, y_test, verbose=0)
62 print('Test loss:', score[0])
63 print('Test accuracy:', score[1])
64
65
66 with open('model.json', 'w') as outfile:
67     json.dump(model.to_json(), outfile)
68 model.save_weights('weights2.h5')

```

```
1 import keras
2 from keras.datasets import mnist
3 from keras.models import Sequential
4 from keras.layers import Dense, Dropout, Flatten
5 from keras.layers import Conv2D, MaxPooling2D
6 from keras import backend as K
7 import json
8
9 batch_size = 32
10 num_classes = 10
11 epochs = 68
12
13 # input image dimensions
14 img_rows, img_cols = 28, 28
15
16 # the data, shuffled and split between train and test sets
17 (x_train, y_train), (x_test, y_test) = mnist.load_data()
18
19 if K.image_data_format() == 'channels_first':
20     x_train = x_train.reshape(x_train.shape[0], 1, img_rows, img_cols)
21     x_test = x_test.reshape(x_test.shape[0], 1, img_rows, img_cols)
22     input_shape = (1, img_rows, img_cols)
23 else:
24     x_train = x_train.reshape(x_train.shape[0], img_rows, img_cols, 1)
25     x_test = x_test.reshape(x_test.shape[0], img_rows, img_cols, 1)
26     input_shape = (img_rows, img_cols, 1)
27
28 x_train = x_train.astype('float32')
29 x_test = x_test.astype('float32')
30 x_train /= 255
31 x_test /= 255
32 print('x_train shape:', x_train.shape)
33 print(x_train.shape[0], 'train samples')
34 print(x_test.shape[0], 'test samples')
35
36 # convert class vectors to binary class matrices
37 y_train = keras.utils.to_categorical(y_train, num_classes)
38 y_test = keras.utils.to_categorical(y_test, num_classes)
39
40 model = Sequential()
```