

PROJECT REPORT

A NOVEL METHOD FOR HANDWRITTEN DIGIT RECOGNITION

submitted by

PNT2022TMID13400

Rajageethaanjali M B – 951919CS076

Srivarthini J – 951919CS100

Pavithra S – 951919CS066

Uthra V – 951919CS108

TABLE OF CONTENTS

| | |
|--|-----------|
| 1.1 PROJECT OVERVIEW | 1 |
| 1.2 PURPOSE | 1 |
| 2 LITERATURE SURVEY | 2 |
| 2.1 EXISTING PROBLEM | 2 |
| 2.2 REFERENCES | 2 |
| 2.3 PROBLEM STATEMENT DEFINITION | 5 |
| 3 IDEATION AND PROPOSED SOLUTION | 6 |
| 3.1 EMPATHY MAP CANVAS | 6 |
| 3.2 IDEATION & BRAINSTORMING | 7 |
| 3.3 PROPOSED SOLUTION | 8 |
| 3.4 PROBLEM SOLUTION FIT | 9 |
| 4 REQUIREMENT ANALYSIS | 10 |
| 4.1 FUNCTIONAL REQUIREMENTS | 10 |
| 4.2 NON FUNCTIONAL REQUIREMENTS | 11 |
| 5 PROJECT DESIGN | 12 |
| 5.1 DATA FLOW DIAGRAM | 12 |
| 5.2 SOLUTION & TECHNICAL ARCHITECTURE | 13 |
| 5.3 USER STORIES | 15 |
| 6 PROJECT PLANNING AND SCHEDULING | 16 |
| 6.1 SPRINT PLANNING AND ESTIMATION | 16 |
| 6.2 SPRINT DELIVERY SCHEDULE | 17 |
| 7 CODING & SOLUTIONING | 18 |
| 8 TESTING | 20 |
| 8.1 TEST CASES | 20 |

| | | |
|-------|----------------------------|----|
| 8.2 | USER ACCEPTANCE TESTING | 22 |
| 8.2.1 | DEFECT ANALYSIS | 22 |
| 8.2.2 | TEST CASE ANALYSIS | 22 |
| 9 | RESULTS | 23 |
| 9.1 | PERFORMANCE METRICS | 23 |
| 10 | ADVANTAGES & DISADVANTAGES | 25 |
| | ADVANTAGES | 25 |
| | DISADVANTAGES | 25 |
| 11 | CONCLUSION | 26 |
| 12 | FUTURE SCOPE | 27 |
| | APPENDIX | 28 |
| | SOURCE CODE | 28 |
| | GITHUB | 37 |
| | PROJECT DEMO | 37 |

CHAPTER 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 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.

1.2 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

2.1 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.

2.2 REFERENCES

Improved Handwritten Digit Recognition Using Convolutional Neural Networks (CNN) (2020)

Ahlawat, Savita and Choudhary, Amit and Nayyar, Anand and Singh, Saurabh and Yoon, Byungun

This paper's primary goal was to enhance handwritten digit recognition ability. To avoid difficult pre-processing, expensive feature extraction, and a complex ensemble (classifier combination) method of a standard recognition system, they examined different convolutional neural network variations. Their current work makes suggestions on the function of several hyper-parameters through thorough evaluation utilizing an MNIST dataset. They also confirmed that optimizing hyper-parameters is crucial for enhancing CNN architecture performance. With the Adam optimizer for the MNIST database, they were able to surpass many previously published results with a recognition rate of 99.89%. Through the trials, it is made abundantly evident how the performance of handwritten digit recognition is affected by the number of convolutional layers in CNN architecture. According to the paper, evolutionary algorithms can be explored for optimizing convolutional filter kernel sizes, CNN learning parameters, and the quantity of layers and learning rates.

An Efficient And Improved Scheme For Handwritten Digit Recognition Based On Convolutional Neural Network (2019)

Ali, Saqib and Shaukat, Zeeshan and Azeem, Muhammad and Sakhawat, Zareen and Mahmood, Tariq and others

This study uses rectified linear units (ReLU) activation and a convolutional neural network (CNN) that incorporates the Deeplearning4j (DL4J) architecture to recognize handwritten digits. The proposed CNN framework has all the necessary parameters for a high level of MNIST digit classification accuracy. The system's training takes into account the time factor as well. The system is also

tested by altering the number of CNN layers for additional accuracy verification. It is important to note that the CNN architecture consists of two convolutional layers, the first with 32 filters and a 5x5 window size and the second with 64 filters and a 7x7 window size. In comparison to earlier proposed systems, the experimental findings show that the proposed CNN architecture for the MNIST dataset demonstrates great performance in terms of time and accuracy. As a result, handwritten numbers are detected with a recognition rate of 99.89% and high precision (99.21%) in a short amount of time.

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.

Handwritten Digit Recognition Using Machine And Deep Learning Algorithms (2021)

Pashine, Samay and Dixit, Ritik and Kushwah, Rishika

In this study, they developed three deep and machine learning-based models for handwritten digit recognition using MNIST datasets. To determine which model was the most accurate, they compared them based on their individual properties. Support vector machines are among the simplest classifiers, making them faster than other algorithms and providing the highest training accuracy rate in this situation. However, due to their simplicity, SVMs cannot categorize complicated and ambiguous images as accurately as MLP and CNN algorithms can. In their research, they discovered that CNN produced the most precise outcomes for handwritten digit recognition. This led them to the conclusion that

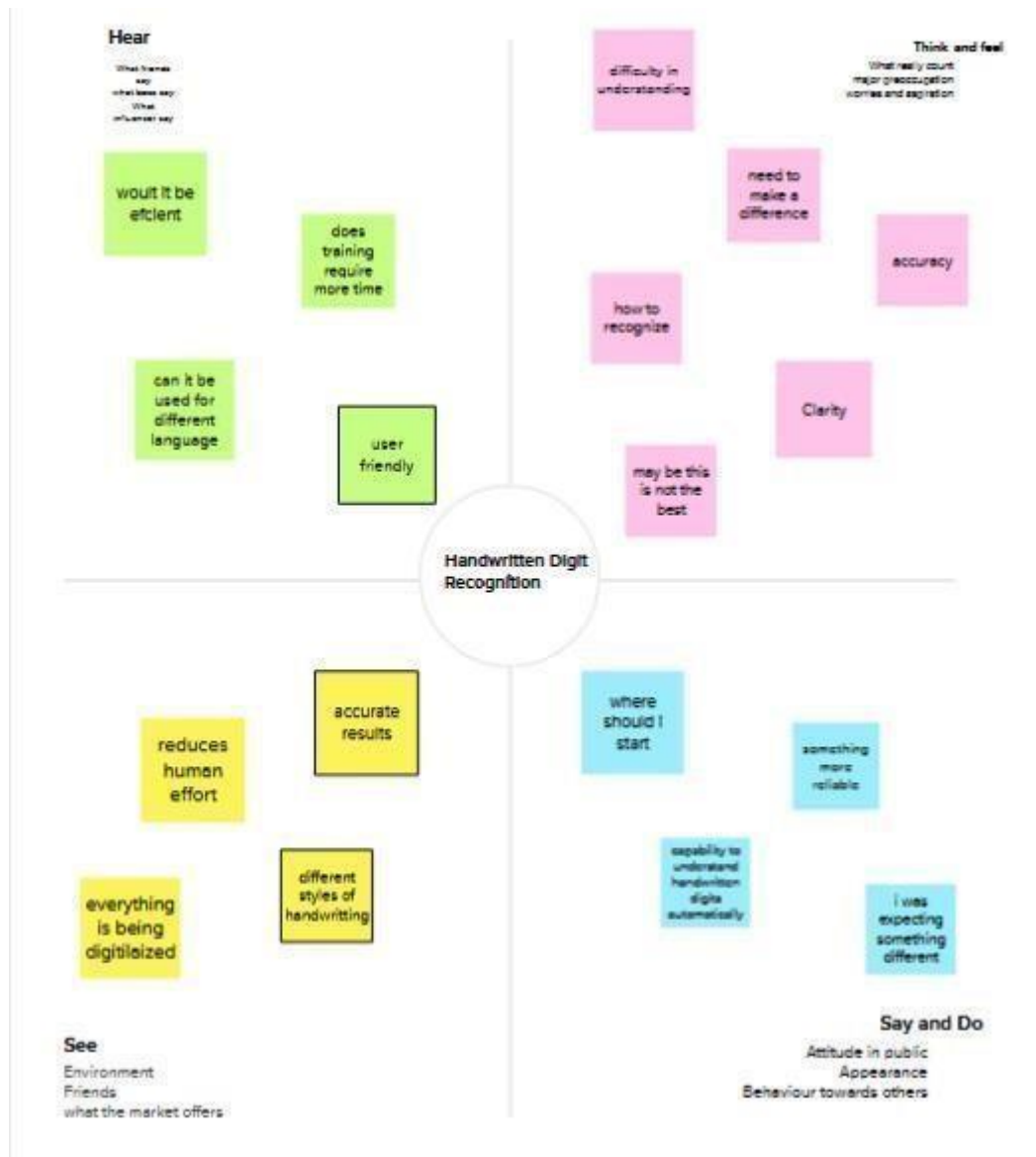
CNN is the most effective solution for all types of prediction issues, including those using picture data. Next, by comparing the execution times of the algorithms, they determined that increasing the number of epochs without changing the configuration of the algorithm is pointless due to the limitation of a certain model, and they discovered that beyond a certain number of epochs, the model begins over-fitting the dataset and provides biased predictions.

2.3 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

3.1 EMPATHY MAP CANVAS



3.2 IDEATION & BRAINSTORMING

2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

🕒 10 minutes

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

Member 1

| | | |
|--|---|---|
| Deletes the scanned images of handwritten digits | Improve response system | Plays an important role in modern world |
| Image augmentation | | Feature extraction from processed image |
| CNN has hidden layers that detects the digits | Using CNN to predict real time hand written value | Creating and training the model |

Member 2

| | | |
|---|---|--|
| Import libraries and loading the dataset | Feature extraction from processed image | Preprocessing of dataset |
| Train the predictive model with the large dataset | | the model takes images and classifies them in a certain category |
| accurate and effective | accurate and effective | makes human job easier in testing system |

Member 3

| | | |
|--|--|---|
| Using CNN predict real hand written digits | classification and recognition | Convert handwritten digits into machine readable format |
| Preprocessing the data | | analyse the image clearly |
| train with many variety of handwritings | Applying statistical techniques for preprocessing and recognising the sign | High accuracy |

Member 4

| | | |
|---|--|---|
| Importing large dataset | Evaluating the model | Acquisition of image |
| Can be used in many applications like postal mail sorting | | the digits can be classified from words, letters and characters |
| use different algorithms | the digits can be identified from the given test | Identifies digits automatically |

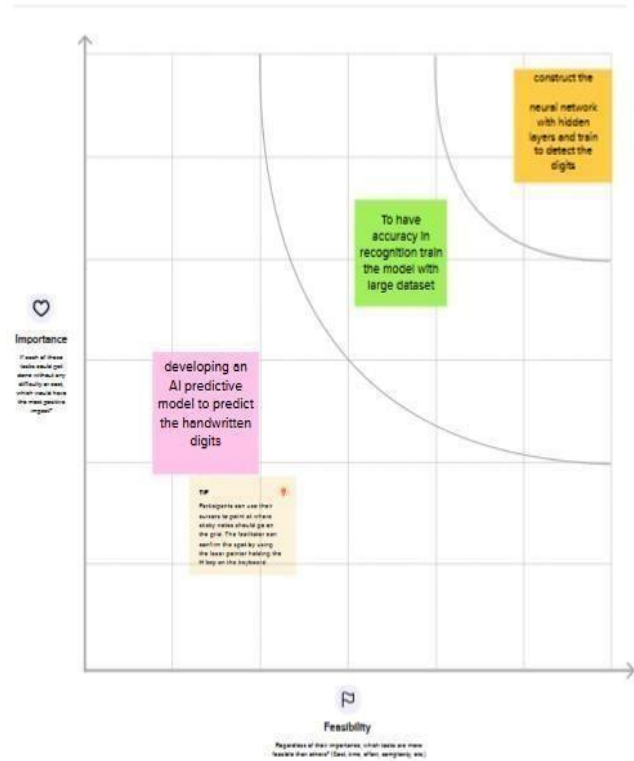
Group ideas

10 minutes

TIP
Add numbers with tags to study notes to make it easier to find relevant, supported, and interesting important ideas as themes within your mind.

Prioritize

20 minutes



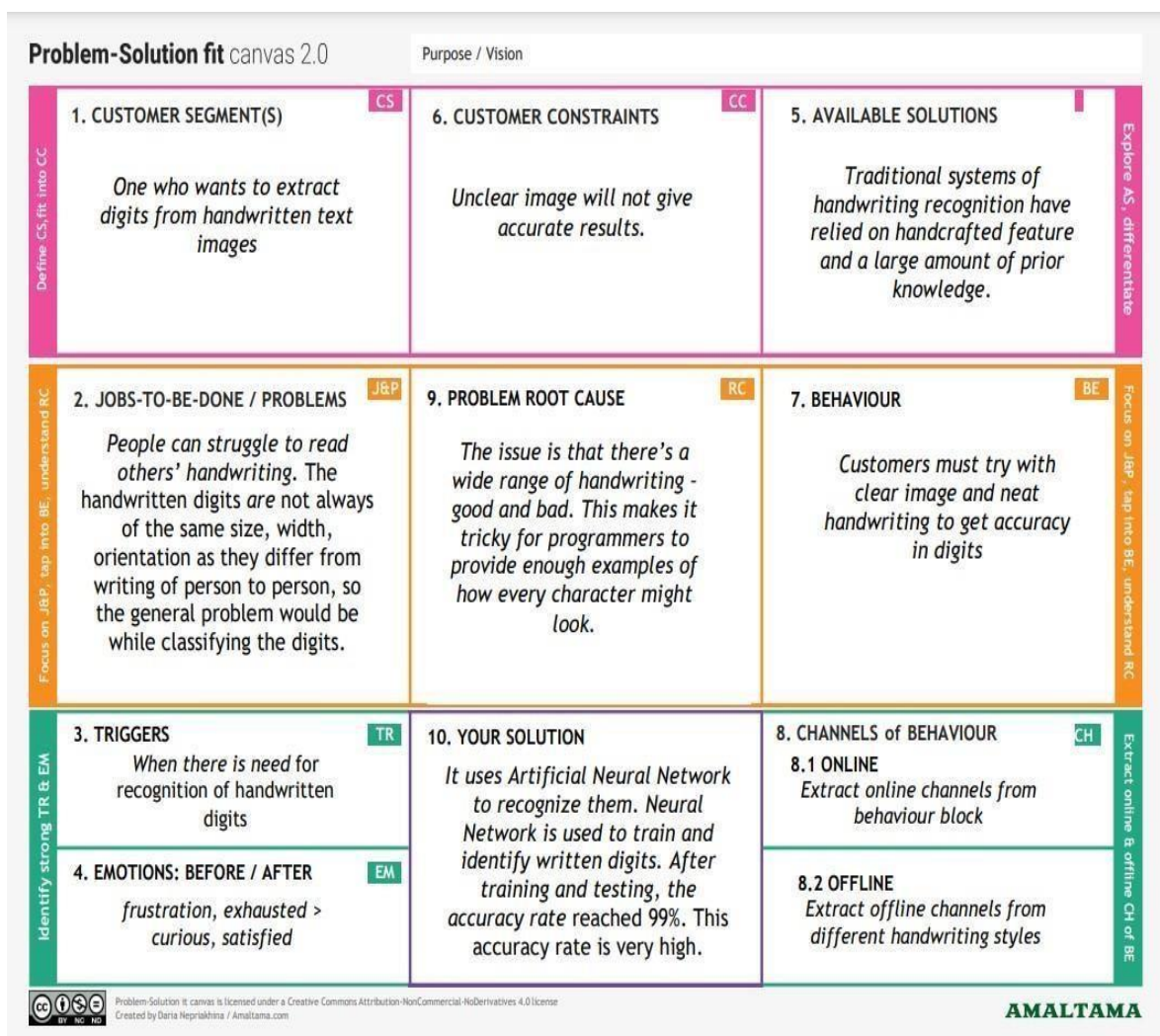
7

| S.No. | Parameter | Description |
|-------|--|---|
| 1. | Problem Statement (Problem to be solved) | The problem statement aims at developing a novel handwritten recognition system using ML .The handwritten digit recognition system is a way to tackle the problem which uses the image of a digit and recognizes the digit present in the image . |
| 2. | Idea / Solution description | Developing an AI predictive model to predict the handwritten digits and to construct a neural network with hidden layers and train to detect the digits. |
| 3. | Novelty / Uniqueness | The system not only produces a classification of the digit but also a rich description of the instantiation parameters which can yield information such as the writing style |

| | | |
|----|---------------------------------------|---|
| 4. | Social Impact / Customer Satisfaction | Handwritten digits can be recognised easily without any strenuous efforts. This reduces time and improves productivity for people. |
| 5. | Business Model (Revenue Model) | It is used in the detection of vehicle numbers, banks for reading cheques, post offices for arranging letters, and many other tasks. |
| 6. | Scalability of the Solution | To attain higher performances in the domain of character recognition and pattern recognition, due to its excellent feature extraction and working as best classifier characteristics. |

| | | |
|--|--|---|
| | | There is no limit in the number of digits that can be recognized. |
|--|--|---|

3.4 PROBLEM SOLUTION FIT



CHAPTER

4

REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

| FR No. | Functional Requirement (Epic) | Sub Requirement (Story / Sub-Task) |
|--------|-------------------------------|--|
| FR-1 | User Registration | Registration through Form Registration through Gmail Registration through LinkedIn |
| FR-2 | User Confirmation | Confirmation via Email Confirmation via OTP |
| FR-3 | Upload image | Image upload via files Image upload via folders Image upload via drive Image upload via web Image upload via scan/camera |
| FR-4 | Spelling support | Identifies handwriting of different styles and fonts Spelling check |
| FR-5 | Translation | Handwritten digits from the image are extracted. Conversion of handwritten digits into machine readable form |

| | | |
|------|---------|---------------------|
| FR-6 | Log out | Log out / sign out. |
|------|---------|---------------------|

4.2 NON FUNCTIONAL REQUIREMENTS

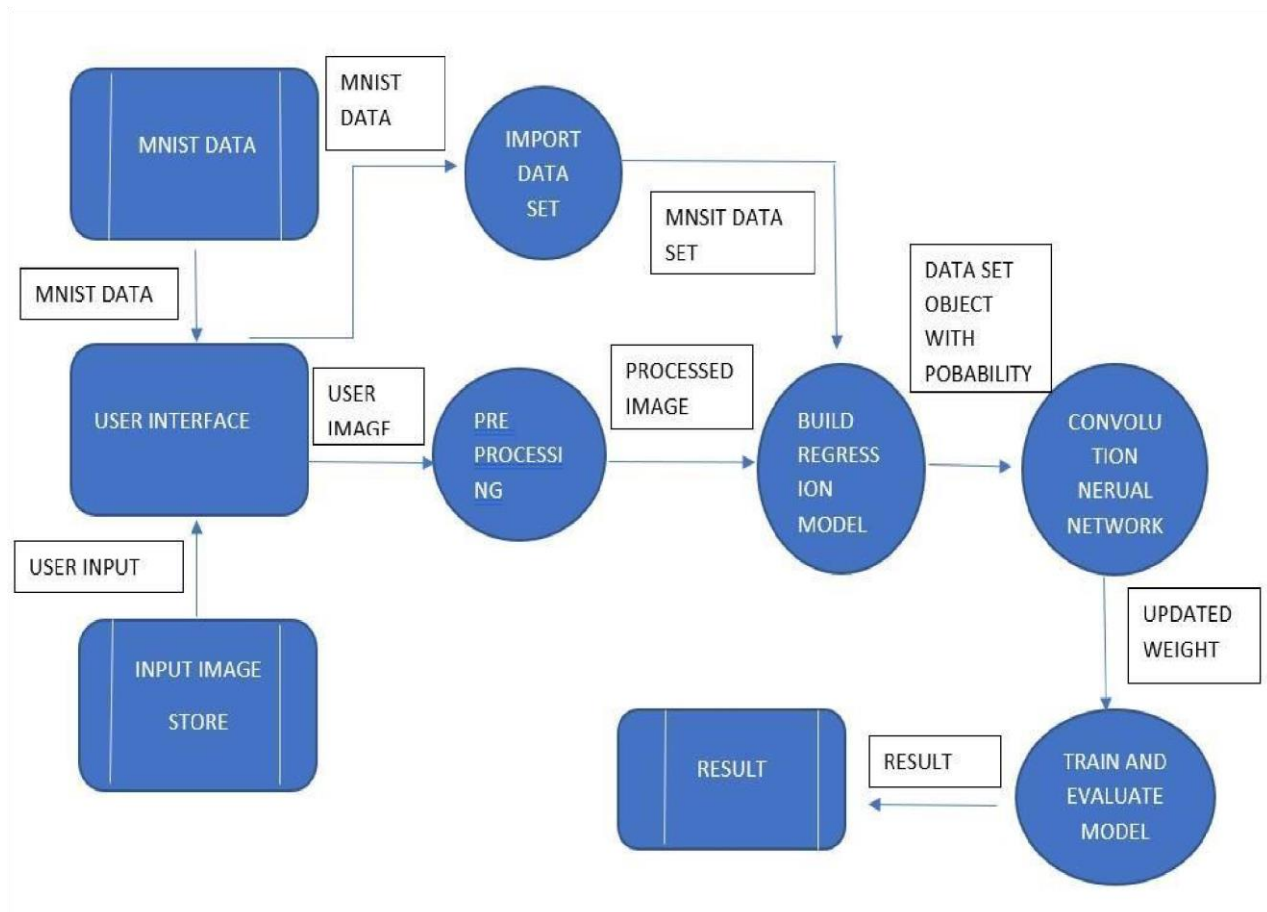
| NFR No. | Non-Functional Requirement | Description |
|---------|----------------------------|---|
| NFR-1 | Usability | The proposed system gives good results for images that contain handwritten text written in different styles, different size and alignment with varying background |
| NFR-2 | Security | Only authorized people can access the system data and modify the database. |

| | | |
|-------|--------------|---|
| NFR-3 | Reliability | The Database is frequently updated with handwriting of different styles and size and will rollback when any update fails. |
| NFR-4 | Performance | The proposed system is advantageous as it uses fewer features to train the neural network, which results in faster convergence. |
| NFR-5 | Availability | The system functionality and services are available for use with all operations. |
| NFR-6 | Scalability | The website traffic limit must be scalable enough to support 2 lakhs users at a time |

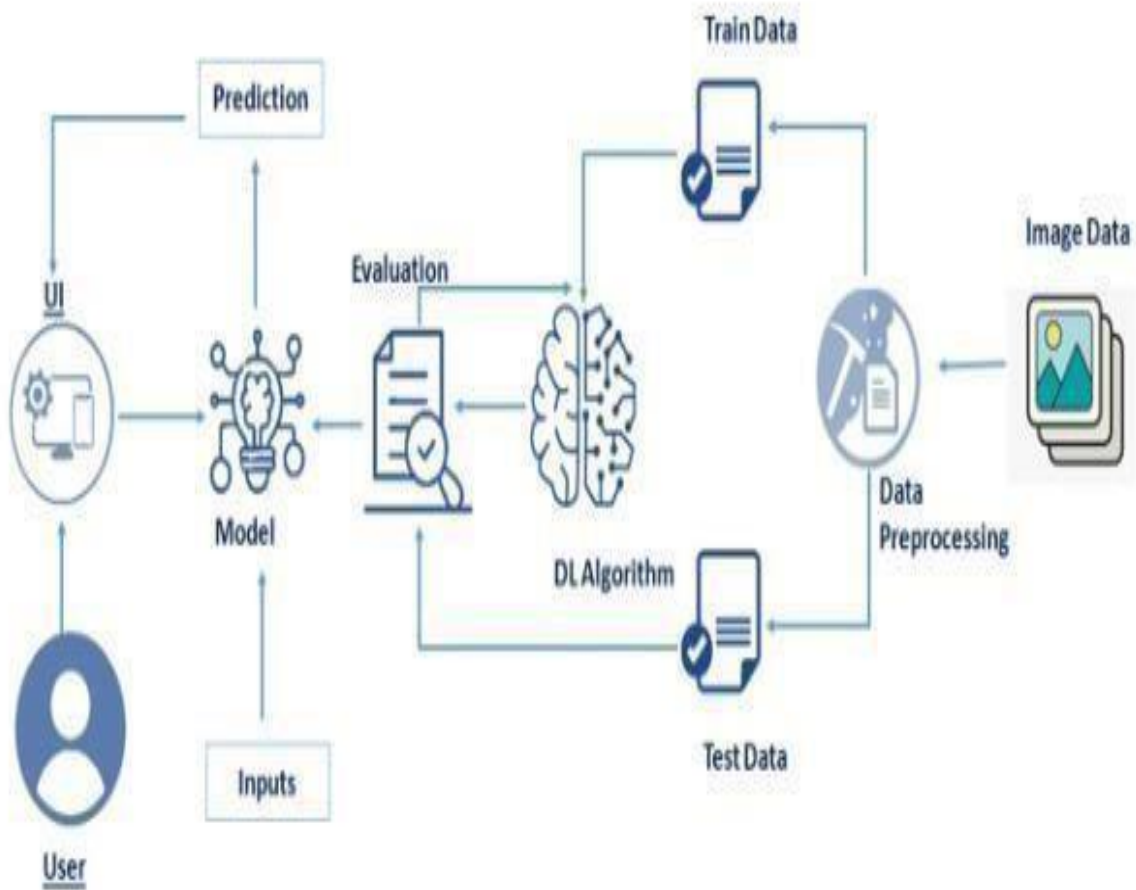
CHAPTER 5

PROJECT DESIGN

5.1 DATA FLOW DIAGRAM



5.2 SOLUTION & TECHNICAL ARCHITECTURE



5.3 USER STORIES

| User Type | Functional Requirement (Epic) | User Story Number | User Story / Task | Acceptance criteria | Priority | Release |
|-----------|-------------------------------|-------------------|-------------------|---------------------|----------|---------|
| | | | | | | |

| | | | | | | |
|---------------------------|--------------|-------|---|---|--------|----------|
| Customer (Mobile user) | Registration | USN-1 | As a user, I can register for the application by entering my email, password, and confirming my password. | I can access my account / dashboard | High | Sprint-1 |
| | | USN-2 | As a user, I will receive confirmation email once I have registered for the application | I can receive confirmation email & click confirm | High | Sprint-2 |
| | | USN-3 | As a user, I can register for the application through gmail or facebook | I can register & access the dashboard with Facebook Login | Medium | Sprint-2 |

| | | | | | | |
|--|-------|-------|---------------------------|--------------------|------|----------|
| | Login | USN-4 | As a user, I can log into | I can login to the | High | Sprint-1 |
|--|-------|-------|---------------------------|--------------------|------|----------|

| | | | | | | |
|--|--------------|-------|---|---|------|----------|
| | | | the application by entering email & password | application | | |
| | Dashboard | USN-5 | Go to dashboard and refer the content about our project | I can read instructions also and the home page is userfriendly. | Low | Sprint-1 |
| | Upload Image | USN-6 | As a user, I can able to input the images of digital documents to the application | As a user, I can able to input the images of digital documents to the application | High | Sprint-3 |

| | | | | | | |
|--|---------|-------|---|--|--------|----------|
| | Predict | USN-7 | As a user I can able to get the recognised digit as output from the images of digital documents or images | I can access the recognized digits from digital document or images | High | Sprint-3 |
| | | USN-8 | As a user, I will train and test the | I can able to train and test the | Medium | Sprint-4 |

| | | | | | | |
|--|--|--|--|---|--|--|
| | | | input to get the maximum accuracy of output. | application until it gets maximum accuracy of the result. | | |
|--|--|--|--|---|--|--|

| | | | | | | |
|-------------------------------|-----------|--------|---|------------------------------------|--------|----------|
| Customer (Web user) | Login | USN-9 | As a user, I can use the application by entering my email, password. | I can access my account | Medium | Sprint-4 |
| Customer Care Executive | Dashboard | USN-10 | upload the image | Recognize and get the output | High | Sprint-1 |
| Administrator | Security | USN-11 | updated the features | checking the security | Medium | Sprint-1 |

CHAPTER 6

PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION

| Sprint | Functional Requirement (Epic) | User Story Number | User Story / Task | Story Points | Priority | Team Members |
|--------|-------------------------------------|----------------------|----------------------|-----------------|----------|-----------------|
|--------|-------------------------------------|----------------------|----------------------|-----------------|----------|-----------------|

| | | | | | | |
|----------|--------------------|-------|---|----|--------|---|
| Sprint-1 | Data Collection | USN-1 | As a user, I can collect the dataset from various resources with different handwritings. | 10 | Low | Srivarthini J Rajageethaanjali M B |
| Sprint-1 | Data Preprocessing | USN-2 | As a user, I can load the dataset, handling the missing data, scaling and split data into train and test. | 10 | Medium | Srivarthini J Rajageethaanjali M B |
| Sprint-2 | Model Building | USN-3 | As a user, I will get an application with ML model which provides high | 5 | High | Srivarthini J Rajageethaanjali M B Pavithra S Uthra V |

| | | | | | | |
|----------|---------------------|-------|---|---|--------|---|
| | | | accuracy of recognized handwritten digit. | | | |
| Sprint-2 | Add CNN layers | USN-4 | Creating the model and adding the input, hidden, and output layers to it. | 5 | High | Srivarthini J Rajageethaanjali M B Pavithra S Uthra V |
| Sprint-2 | Compiling the model | USN-5 | With both the training data defined and model defined, it's time to configure the learning process. | 2 | Medium | Srivarthini J Rajageethaanjali M B Pavithra S Uthra V |

| | | | | | | |
|----------|------------------------|-------|---|---|--------|---|
| Sprint-2 | Train & test the model | USN-6 | As a user, let us train our model with our image dataset. | 6 | Medium | Srivarthini J Rajageethaanjali M B Pavithra S Uthra V |
|----------|------------------------|-------|---|---|--------|---|

| | | | | | | |
|----------|-------------------------|-------|--|---|------|---|
| Sprint-2 | Save the model | USN-7 | As a user, the model is saved & integrated with an android application or web application in order to predict something. | 2 | Low | Pavithra S |
| Sprint-3 | Building UI Application | USN-8 | As a user, I will upload the handwritten digit image to the application by clicking a upload button. | 5 | High | Srivarthini J Rajageethaanjali M B Pavithra S Uthra V |

| | | | | | | |
|----------|------------------------|--------|--|----|--------|---|
| Sprint-3 | | USN-9 | As a user, I can know the details of the fundamental usage of the application. | 5 | Low | Uthra V |
| Sprint-3 | | USN-10 | As a user, I can see the predicted / recognized digits in the application. | 5 | Medium | Srivarthini J Rajageethaanjali M B Pavithra S Uthra V |
| Sprint-4 | Train the model on IBM | USN-11 | As a user, I train the model on IBM and integrate flask/Django with scoring end point. | 10 | High | Srivarthini J Rajageethaanjali M B Pavithra S Uthra V |

| | | | | | | |
|----------|------------------|--------|--|----|------|--|
| Sprint-4 | Cloud Deployment | USN-12 | As a user, I can access the web application and make the use of the product from anywhere. | 10 | High | Srivarthini J Rajageethaanjali M B |
|----------|------------------|--------|--|----|------|--|

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 | 29 Oct 2022 | 03 Nov 2022 | 20 | 03 Nov 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 |

CHAPTER 7 CODING & SOLUTIONING

```
import os

import numpy as np
from flask import Flask, render_template, request, send_from_directory, url_for
from gevent.pywsgi import WSGIServer
from keras.models import load_model
from keras.preprocessing import image
from PIL import Image
from werkzeug.utils import redirect, secure_filename
💡
UPLOAD_FOLDER = 'C:/Users/Downloads/ibmproject/flask_app/upload'
```

```

app = Flask(__name__, template_folder='template')
app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER

model = load_model("mnistCNN.h5")

@app.route('/', methods=['GET'])
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)

```

CHAPTER 8

TESTING

8.1 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 | PASS |
| 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 |

8.2 USER ACCEPTANCE TESTING

8.2.1 DEFECT ANALYSIS

| Resolution | Severity 1 | Severity 2 | Severity 3 | Severity 4 | Total |
|------------|------------|------------|------------|------------|-------|
| By Design | 1 | 0 | 1 | 0 | 2 |
| | 0 | 0 | 0 | 0 | 0 |

| | | | | | |
|----------------|---|---|---|---|---|
| Duplicate | | | | | |
| 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 |

8.2.2 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

9.1 PERFORMANCE METRICS

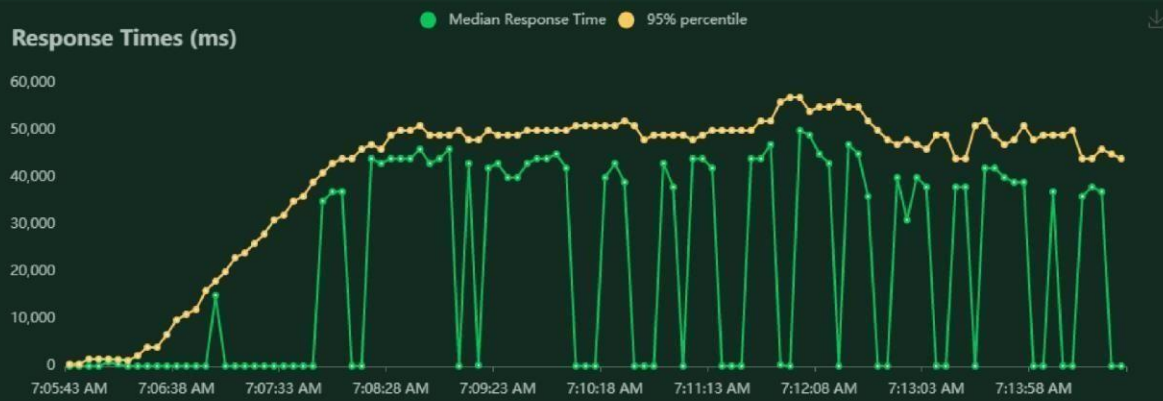
| Locust Test Report | | | | | | | | | |
|---|----------|-------------|-------------|--------------|-------------|-------------|----------------------|-------------|--------------|
| During: 11/12/2022, 7:05:40 AM - 11/12/2022, 7:14:47 AM | | | | | | | | | |
| Target Host: http://127.0.0.1:5000/ | | | | | | | | | |
| Script: locust.py | | | | | | | | | |
| Request Statistics | | | | | | | | | |
| Method | Name | # Requests | # Fails | Average (ms) | Min (ms) | Max (ms) | Average size (bytes) | RPS | Failures/s |
| GET | / | 1043 | 0 | 13 | 4 | 290 | 1079 | 1.9 | 0.0 |
| GET | /predict | 1005 | 0 | 39648 | 385 | 59814 | 2670 | 1.8 | 0.0 |
| Aggregated | | 2048 | 0 | 19462 | 4 | 59814 | 1859 | 3.7 | 0.0 |
| Response Time Statistics | | | | | | | | | |
| Method | Name | 50%ile (ms) | 60%ile (ms) | 70%ile (ms) | 80%ile (ms) | 90%ile (ms) | 95%ile (ms) | 99%ile (ms) | 100%ile (ms) |
| GET | / | 10 | 11 | 13 | 15 | 19 | 22 | 62 | 290 |
| GET | /predict | 44000 | 46000 | 47000 | 48000 | 50000 | 52000 | 55000 | 60000 |
| Aggregated | | 36 | 36000 | 43000 | 45000 | 48000 | 50000 | 54000 | 60000 |

Charts

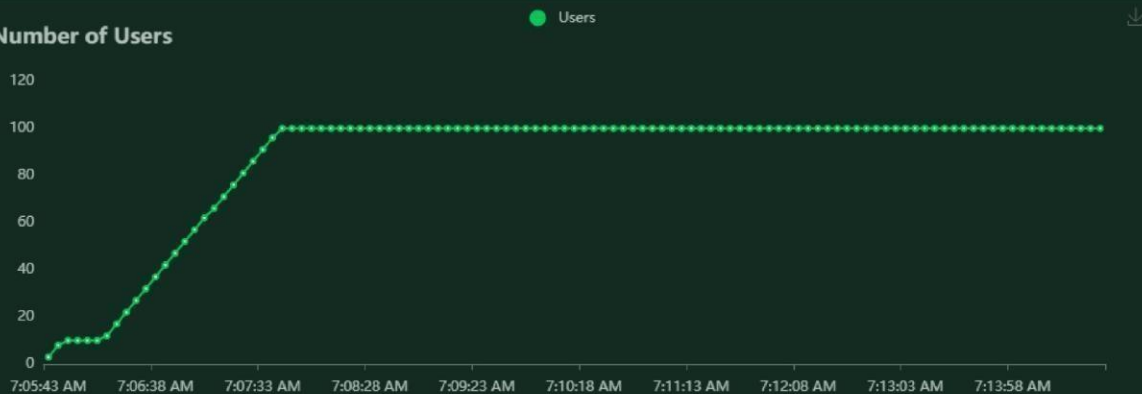
Total Requests per Second



Response Times (ms)



Number of Users



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, 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

```
<html>
<head>
<title>Digit Recognition Web Application</title>
<meta name="viewport" content="width=device-width">
<!-- GoogleFont -->
<link href="https://fonts.googleapis.com/css2?family=Prompt:wght@600&display=swap" rel="stylesheet">
<link href="https://fonts.googleapis.com/css2?family=Varela+Round&display=swap" rel="stylesheet">
```



```

<link href="https://fonts.googleapis.com/css2?family=Source+Code+Pro:wght@500&display=swap"
rel="stylesheet">

<link href="https://fonts.googleapis.com/css?family=Calistoga|Josefin+Sans:400,700|Pacifico&display=swap"
rel="stylesheet"> <!-- bootstrap -->

<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">

<link rel="stylesheet" type="text/css" href="{ { url_for('static',filename='css/style.css') } }"> <!--
fontawesome -->

<script src="https://kit.fontawesome.com/b3aed9cb07.js" crossorigin="anonymous"></script>

<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-
UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs@latest"></script>

</head>

<script> function
preview() {
frame.src=URL.cre
ateObjectURL(even
t.target.files[0]); }

$(document).ready(function() {
$('#clear_button').on('click', function() {
$('#image').val("");
$('#frame').attr('src','');
});
});

```

```
</script>

<body>

<h1 class="welcome">TEAM ID : PNT2022TMID13400

<div id="team_id">DIGIT RECOGNIZING WEBSITE</div>

</h1>

<section id="title">

<h4 class="heading">HANDWRITTEN DIGIT RECOGNITION</h4>

<br><br>

<p>

The website is made for mainly predicting the handwritten digit.

</p>

</section>

<section id="content">

<div class="leftside">

<form action="/predict" method="POST" enctype="multipart/form-data">

<label>Select a image:</label>

<input id="image" type="file" name="image" accept="image/png, image/jpeg" onchange="preview()"><br><br>

<img id="frame" src="" width="100px" height="100px"/>

<div class="buttons_div">

<button type="submit" class="btn btn-dark" id="predict_button">PREDICT</button>

<button type="button" class="btn btn-dark" id="clear_button">&nbsp; CLEAR &nbsp;</button></div></div>

</section>

</body>

</html>
```

```

<link href="https://fonts.googleapis.com/css?family=Calistoga|Josefin+Sans:400,700|Pacifico&display=swap"
rel="stylesheet"> <!-- bootstrap -->

<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">

<link rel="stylesheet" type="text/css" href="{ { url_for('static',filename='css/style.css') } }"> <!--
fontawesome -->

<script src="https://kit.fontawesome.com/b3aed9cb07.js" crossorigin="anonymous"></script>

<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-
UO2eT0CpHqdSJK6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs@latest"></script>

</head>

<script> function preview() {
frame.src=URL.createObjectURL(event.target.files[0]);
}

$(document).ready(function() {
$('#clear_button').on('click', function() {
$('#image').val("");
$('#frame').attr('src','');
});
});

</script>

<body>

<h1 class="welcome">TEAM ID : PNT2022TMID13400

```

```

<div id="team_id">DIGIT RECOGNIZING WEBSITE</div>

</h1>

<section id="title">

<h4 class="heading">HANDWRITTEN DIGIT RECOGNITION</h4>

<br><br>

<p>

The website is made for mainly predicting the handwritten digit.

</p>

</section>

<section id="content">

<div class="leftside">

<form action="/predict" method="POST" enctype="multipart/form-data">

<label>Select a image:</label>

<input id="image" type="file" name="image" accept="image/png, image/jpeg" onchange="preview()"><br><br>

<img id="frame" src="" width="100px" height="100px"/>

<div class="buttons_div">

<button type="submit" class="btn btn-dark" id="predict_button">PREDICT</button>

<button type="button" class="btn btn-dark" id="clear_button">&nbsp; CLEAR &nbsp;</button>

</div>

</form>

</div>

</section>

</body>

</html>

</div>

</form>

</div>

</section>

</body> </html>

```

Predict.html

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

<style> body{
background-image: url('static/images/image01.jpg');
background-repeat: no-repeat; background-size:
cover; }

#rectangle{ width:400px;
height:150px;
background-color:
#5796a5; border-radius:
25px; position:absolute;
top:25%; left:50%;
transform:translate(-
50%,-50%); }

#ans{ text-align:
center; font-size:
40px; margin: 0
auto; padding: 3%
5%; padding-top:
15%; color: white;
}

</style>
<body>
<div id="rectangle">
<h1 id="ans">Predicted Number : { {num}} </h1>
```

```
</div>
</body> </html>
```

Style.Css

```
#clear_button{ margin-
left: 17px; font-weight:
bold; color: blue;
}

#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: honeydew;
background-color: black;
padding-top: 1%; padding-
bottom: 1%; font-weight: bold;
font-family: 'Prompt', sans-serif;
}

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

#predict_button{
margin-right: 15px;
```

```
color: blue; 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: 'Varela Round',
sans-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%; */
} p{ font-family: 'Source Code Pro', monospace,sans-
serif; margin-top: 1%; }

@media (min-width: 720px) {
.leftside{ padding-left:
10%;
}

```

App.Py

```

import os

import numpy as np from flask import Flask, render_template, request,
send_from_directory, url_for from gevent.pywsgi import WSGIServer from
keras.models import load_model from keras.preprocessing import image from
PIL import Image

from werkzeug.utils import redirect, secure_filename

UPLOAD_FOLDER = 'C:/Users/Happy/Desktop/flask_app' app

= Flask(__name__,template_folder='template')

```



```

app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER model
= load_model("mnistCNN.h5")

@app.route('/', methods=['GET'])
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)

```


HOME PAGE (CSS)

```
#predict_button{
  margin-right: 15px;
  color: blue;
  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: 'Varela Round', sans-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%; */
}

p{
  font-family: 'Source Code Pro', monospace, sans-serif;
  margin-top: 1%;
}

@media (min-width: 720px) {
  .leftside{
    padding-left: 10%;
  }
}
```


PREDICT PAGE (HTML)

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

<style>
  body{
    background-image: url('static/images/image01.jpg');
    background-repeat: no-repeat;
    background-size: cover;
  }

  #rectangle{
    width:400px;
    height:150px;
    background-color: #5796a5;
    border-radius: 25px;
    position:absolute;
    top:25%;
    left:50%;
    transform:translate(-50%,-50%);
  }
```



```

        #ans{
        text-align: center;
        font-size: 40px;
        margin: 0 auto;
        padding: 3% 5%;
        padding-top: 15%;
        color: ■white;
        }
    </style>
    <body>
        <div id="rectangle">
            <h1 id="ans">Predicted Number : {{num}}</h1>
        </div>
    </body>
</html>

```

Application.py

```

import os

import numpy as np
from flask import Flask, render_template, request, send_from_directory, url_for
from gevent.pywsgi import WSGIServer
from keras.models import load_model
from keras.preprocessing import image
from PIL import Image
from werkzeug.utils import redirect, secure_filename
💡
UPLOAD_FOLDER = 'C:/Users/Downloads/ibmproject/flask_app/upload'

```

```

app = Flask(__name__, template_folder='template')
app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER

model = load_model("mnistCNN.h5")

@app.route('/', methods=['GET'])
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)

```



<https://github.com/IBM-EPBL/IBM-Project-21281-1659776755>



https://drive.google.com/file/d/1XL0wBCW9jAS6VFZzRPSx2HSBam-N-WMc/view?usp=share_link

