# PROJECT REPORT

# A NOVEL METHOD FOR

# HANDWRITTEN DIGIT RECOGNITION

*submitted by*

**PNT2022TMID05025**

NISHANTHINI S - 921319104136

NESIKA P - 921319104134

SABITHA J - 921319104162

SANGEETHA M - 921319104165

# TABLE OF CONTENTS

# CHAPTER 1

# INTRODUCTION

## 1.1 PROJECT OVERVIEW

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.

The 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 the UI. In the Banking Sector too where more handwritten numbers are involved like account number, figure of cash and checks.

## 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, 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

1. FaisalTehseen Shah, Kamran Yousaf,"Handwritten Digit Recognition Using Image Processing and Neural Networks", Proceedings of the World Congress on Engineering, vol., 2007.
2. Dr.Kusumgupta2 ,"a comprehensive review on handwritten digit recognition using various neural network approaches", international journal of enhanced research in management & computer applications, vol. 5, no. 5, pp. 22-25, 2016.
3. Saeed AL-Mansoori,"Intelligent Handwritten Digit Recognition using Artificial Neural Network", Int. Journal of Engineering Research and Applications, vol. 5, no. 5, pp. 46-51, 2015.
4. Viragkumar N. Jagtap , Shailendra K. Mishra,"Fast Efficient Artificial Neural Network for Handwritten Digit Recognition", International Journal of Computer Science and Information Technologies, vol. 52, no. 0975- 9646, pp. 2302-2306, 2014.

5. Ishani Patel, ViragJagtap and OmpriyaKale."A Survey on Feature Extraction Methods for Handwritten Digits Recognition", International Journal of Computer Applications, vol. 107, no. 12, pp. 11-17, 2014.
6. Dutt A, Dutt A (2017) Handwritten digit recognition using deep learning. Int J AdvRes Comput Eng Technol 6(7):990–997

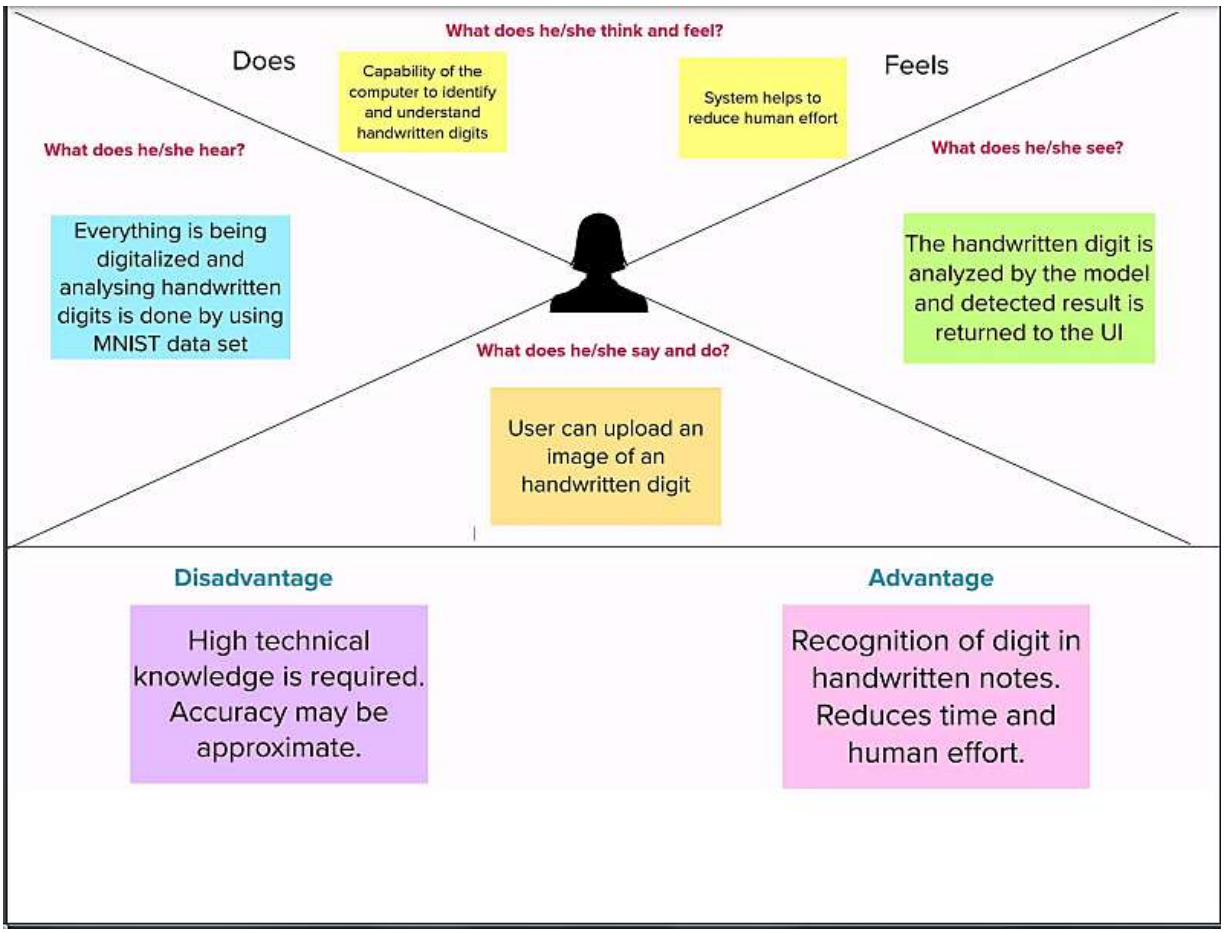
## 2.3 PROBLEM STATEMENT DEFINITION

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. The 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 the UI. In the Banking Sector too where more handwritten numbers are involved like account number, figure of cash and checks.

# CHAPTER 3

# IDEATION AND PROPOSED SOLUTION

## 3.1 EMPATHY MAP CANVAS

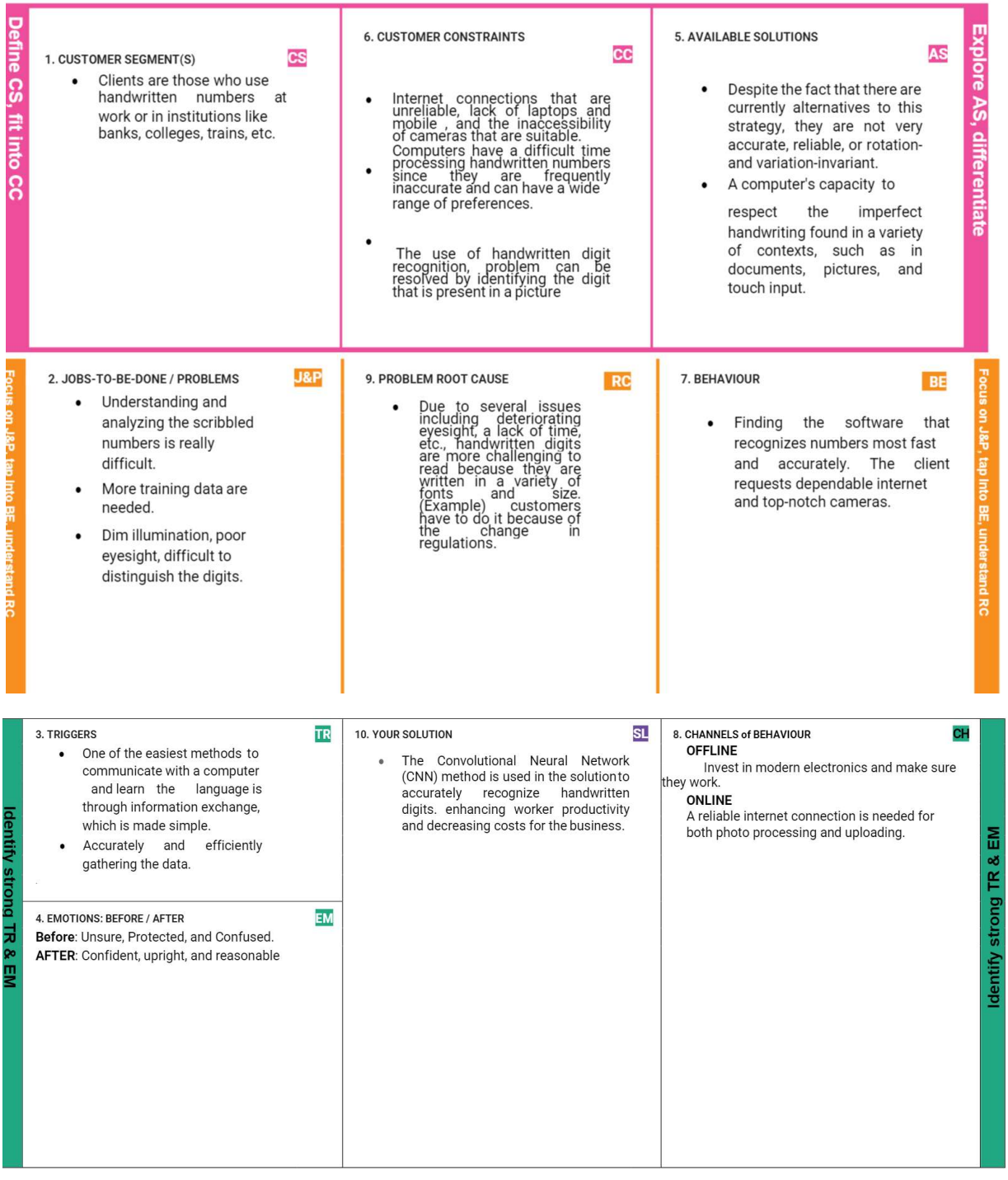


## 3.2 IDEATION & BRAINSTORMING

## 3.3 PROPOSED SOLUTION

| S.No. | Parameter | Description |
|---|---|---|
| 1. | Problem Statement (Problem to be solved) | To recognize the handwritten digits from a wide variety of sources like emails, papers, images, letters etc. It can solve more complex problems and makes humans' jobs easier. |
| 2. | Idea / Solution description | This system is built by using (CNN) Convolutional Neural Network. By using this system, we can capture the image of handwritten digits and can predict the digits. |
| 3. | Novelty / Uniqueness | Here users can upload the handwritten digits from anywhere and it gives accurate results. |
| 4. | Social Impact / Customer Satisfaction | The feasibility of implementing this idea is moderate neither easy nor tough because the system needs to satisfy the basic requirements of the customer and should give accurate results. |
| 5. | Business Model (Revenue Model) | By using this website, the users can predict and analyze the handwritten digits of the user. The website can be developed at minimum cost with high performance and interactive user interface. |
| 6. | Scalability of the Solution | The solution can be made scalable. The people can easily understand the digits. This system can also be integrated with future technologies |

## 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 | MNIST Dataset | A training set of 60,000 instances and a test set are included in the modified National Institute of Standards and Technology dataset (MNIST) database of handwritten digits.10,000 examples in a set. |
| FR-2 | Data pre-processing | Enhances the image by applying a few adjustments to the input image to get it ready for segmentation. |
| FR-3 | Image Data | Handwritten digit recognition refers to a computer's capacity to identify human handwritten digits from a variety of sources, such as photographs, documents, touch screens, etc., and categorise them into ten established classifications (0-9). In the realm of deep learning, this has been the subject of countless studies |
| FR-4 | Website | Web hosting makes the code, graphics, and other items that make up a website accessible online. A server hosts every website you have ever visited. The type of hosting determineshow much space is allotted to a website on a server. Shared, dedicated, VPS, and reseller hosting are the fourbasic varieties. |
| FR-5 | Digit Classifier Model | To train a convolutional network to predict the digit from an image, use the MNIST database of handwritten digits. get the training and validation datafirst |
| FR-6 | Evaluation | Make that the model recognizes the digit correctly and generates the correct result. |

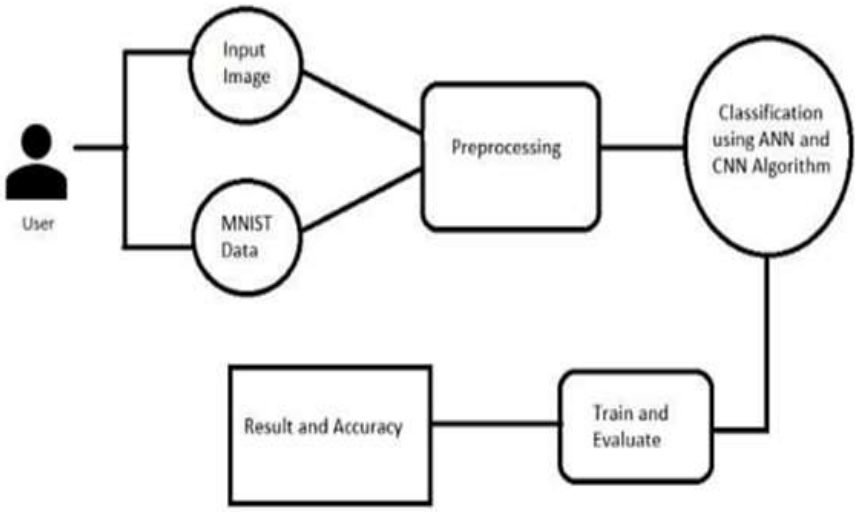
### 4.2 NON FUNCTIONAL REQUIREMENTS

| FR No. | Non-Functional Requirement | Description |
|---|---|---|
| NFR-1 | **Usability** | One of the very significant problems in pattern recognition applications is the recognition of handwritten characters. Applications for digit recognition include filling out forms, processing bankcheques, and sorting mail. |
| NFR-2 | **Security** | 1) The system generates a thorough description of the instantiation parameters, which mightreveal information likethe writing style, in addition to a categorization of the digit. 2) The generative models are capable of segmentation drivenby recognition. 3) The procedure uses a relatively. |
| NFR-3 | **Reliability** | The samples are used by the neural network to automatically deduce rules for reading handwritten digits. Furthermore, the network may learn more about handwriting and hence enhance its accuracy by increasing the quantity of training instances. Numerous techniques and algorithms, such as Deep Learning/CNN, SVM, Gaussian Naive Bayes, KNN, Decision Trees, Random Forests,etc., can be used to recognise handwritten numbers |
| NFR-4 | **Performance** | Information is restricted to each userslimited access |
| NFR-5 | **Availability** | Applications for digit recognition include filling out forms, processing bankchecks, and sorting mail. |
| NFR-6 | **Scalability** | Thesystem should be able to handle 1000 users accessing the website at the sametime |

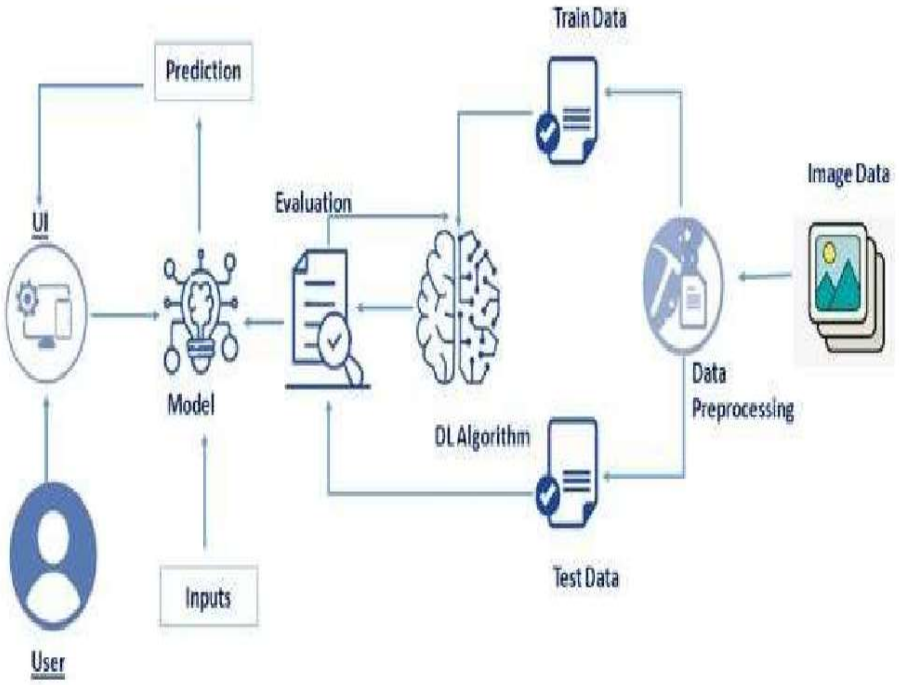
# CHAPTER 5

# PROJECT DESIGN

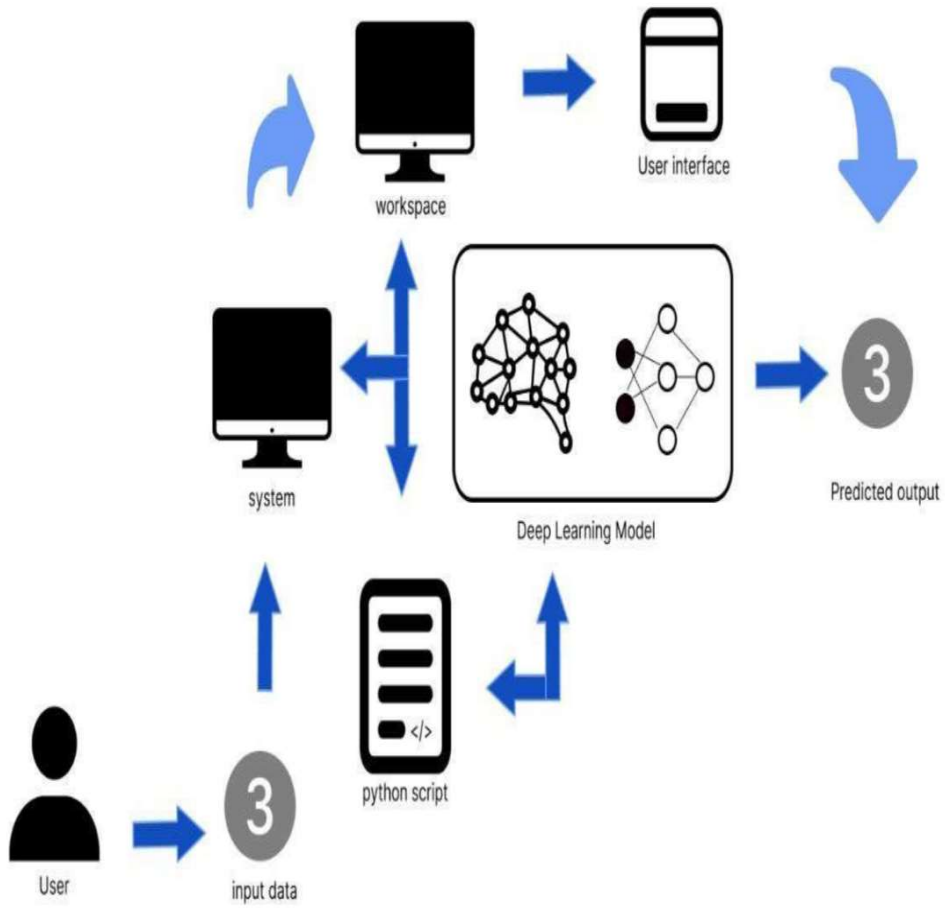
## 5.1 DATA FLOW DIAGRAM



## 5.2 SOLUTION & TECHNICAL ARCHITECTURE




Technology Architecture




Process Flow Diagram

## 5.3 USER STORIES

| User Type | Functional Requirement (Epic) | User StoryNumber | UserStory /Task | Acceptance criteria | Priority | Release |
|---|---|---|---|---|---|---|
| Customer (Mobileuser) | Home | USN-1 | I can view the user manual and am aware of how to use this application. | I can see the limitation of this software & the awarenes of how to use it. | Low | Sprint-1 |
| | | USN-2 | I'm authorized to watch the instructional film to learn how to utilize this application's interface as a user. | I can learn how to use this application through a hands-on approach. | Low | Sprint-1 |
| | | USN-3 | I am able to understand the directions for using thisapplication as a user. | I am able to read the directions and utilize it. | Low | Sprint-2 |
| | Recognize | USN-4 | I get to select the image on this prediction page asa user. | I am able to select an image from our local system. | High | Sprint-2 |
| | Predict | USN-5 | I am permitted to upload and pick the image that willbe submitted as a user. | From the system storage as well as any virtual storage, I may upload and select an image. | Medium | Sprint-3 |
| | | USN-6 | I will train and evaluate the input as a user to ensurethe output is as accurate as possible. | I am able to test and train the application. | High | Sprint-4 |
| | | USN-7 | I have access to the MNIST data collection asa user. | To produce the precise output, I may access it. | Medium | Sprint-3 |

# 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 | Sabitha J<br>Nesika P |
| 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 | Nishanthini S<br>Sangeetha M |
| Sprint-2 | Model Building | USN-3 | As a user, I will get an application with ML model which provides high accuracy of recognized handwritten digit. | 5 | High | Sabitha J<br>Nesika P<br>Nishanthini S<br>Sangeetha M |
| Sprint-2 | Add CNN layers | USN-4 | Creating the model and adding the input, hidden, and output layers to it. | 5 | High | Nishanthini S<br>Sabitha J<br>Nesika P<br>Sangeetha M |
| 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 | Nesika P<br>Sangeetha M<br>Nishanthini S |

| Sprint | Functional Requirement (Epic) | User Story Number | User Story / Task | Story Points | Priority | Team Members |
|---|---|---|---|---|---|---|
| Sprint-2 | Train & test the model | USN-6 | As a user, let us train our model with our image dataset. | 6 | Medium | Sabitha J<br>Nishanthini S<br>Nesika P |
| 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 | Sangeetha M |
| 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 | Sabitha J<br>Nesika P<br>Sangeetha M |
| Sprint-3 | | USN-9 | As a user, I can know the details of the fundamental usage of the application. | 5 | Low | Nishanthini S<br>Sabitha J |
| Sprint-3 | | USN-10 | As a user, I can see the predicted / recognized digits in the application. | 5 | Medium | Nishanthini S<br>Sangeetha M |
| 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 | Nesika P<br>Sabitha J<br>Nishanthini S<br>Sangeetha M |
| 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 | Sabitha J<br>Nesika P |

## 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 |

# CHAPTER 7

# CODING & SOLUTIONING

```
import numpy as np
import tensorflow #open source used for both ML and DL for computation
from tensorflow.keras.datasets import mnist #mnist dataset
from tensorflow.keras.models import Sequential #it is a plain stack of layers
from tensorflow.keras import layers #A Layer consists of a tensor- in tensor-out computat ion funct ion
from tensorflow.keras.layers import Dense, Flatten #Dense-Dense Layer is the regular deeply connected r
#faltten -used fot flattening the input or change the dimension
from tensorflow.keras.layers import Conv2D #convolutional Layer
from keras.utils import np_utils #used for one-hot encoding
import matplotlib.pyplot as plt  #used for data visualization
```

```
(x_train, y_train), (x_test, y_test)=mnist.load_data () #splitting the mnist data into train and test
```

```
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/mnist.npz
11490434/11490434 [==============================] - 0s 0us/step
```

```
print (x_train.shape)  #shape is used for give the dimens ion values #60000-rows 28x28-pixels
print (x_test.shape)
```

```
(60000, 28, 28)
(10000, 28, 28)
```

# 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 |
| BE_TC_001 | Functional | Backend | Check if all the routes are working properly | All the routes should properly work | Working as expected | PASS |
| RP_TC_001 | UI | Result Page | Verify UI elements in the Result Page | The Result page must 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 |
| 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 |
| Total | 5 | 1 | 3 | 3 | 12 |

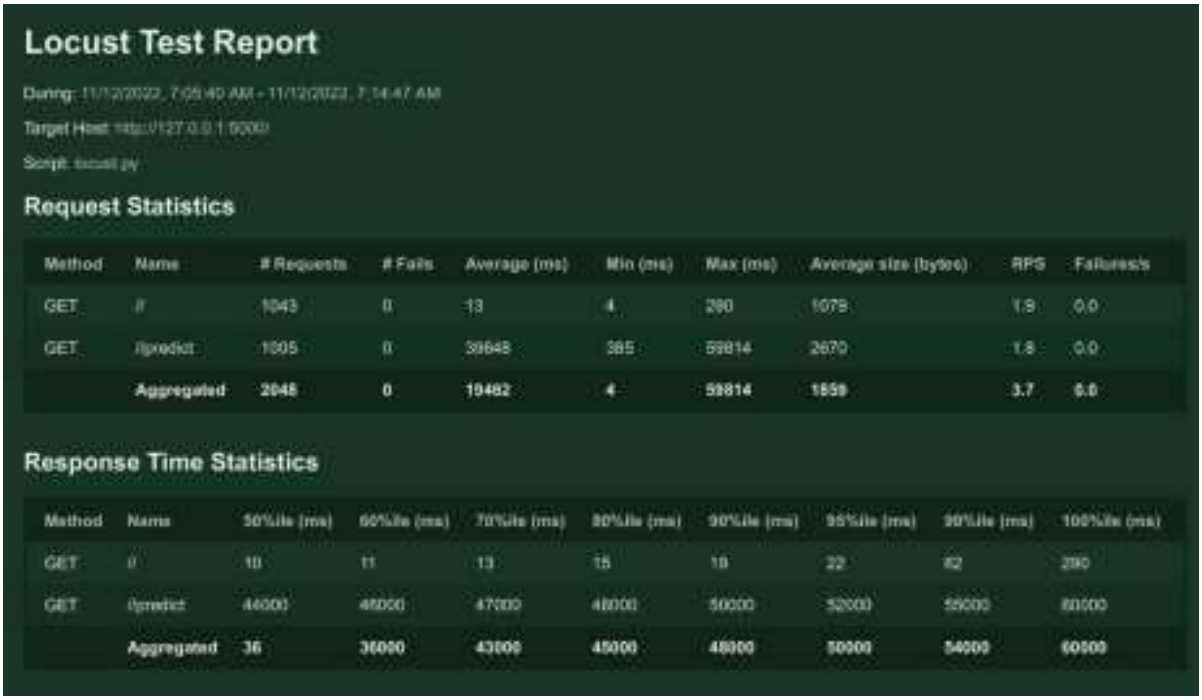
### 8.2.2 TEST CASE ANALYSIS

| 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 | 280 | 1079 | 1.9 | 0.0 |
| GET | /predict | 1005 | 0 | 39648 | 385 | 59814 | 2670 | 1.8 | 0.0 |
| | Aggregated | 2048 | 0 | 19482 | 4 | 59814 | 1859 | 3.7 | 0.0 |

Response Time Statistics

| Method | Name | 50%ile (ms) | 66%ile (ms) | 75%ile (ms) | 80%ile (ms) | 90%ile (ms) | 95%ile (ms) | 98%ile (ms) | 100%ile (ms) |
|---|---|---|---|---|---|---|---|---|---|
| GET | / | 10 | 11 | 13 | 15 | 19 | 22 | 62 | 280 |
| GET | /predict | 44000 | 46000 | 47000 | 48000 | 50000 | 52000 | 56000 | 60000 |
| | Aggregated | 36 | 36000 | 43000 | 45000 | 48000 | 50000 | 54000 | 60000 |

# 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

```
import numpy as np
import tensorflow #open source used for both ML and DL for computation
from tensorflow.keras.datasets import mnist #mnist dataset
from tensorflow.keras.models import Sequential #it is a plain stack of layers
from tensorflow.keras import layers #A Layer consists of a tensor- in tensor-out computat ion funct ion
from tensorflow.keras.layers import Dense, Flatten #Dense-Dense Layer is the regular deeply connected r
#faltten -used fot flattening the input or change the dimension
from tensorflow.keras.layers import Conv2D #convolutional Layer
from keras.utils import np_utils #used for one-hot encoding
import matplotlib.pyplot as plt  #used for data visualization
```

```
(x_train, y_train), (x_test, y_test)=mnist.load_data () #splitting the mnist data into train and test
```

```
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/mnist.npz
11490434/11490434 [==============================] - 0s 0us/step
```

```
print (x_train.shape)  #shape is used for give the dimens ion values #60000-rows 28x28-pixels
print (x_test.shape)
```

```
(60000, 28, 28)
(10000, 28, 28)
```

```
#Reshaping to format which CNN expects (batch, height, width, channels)
x_train=x_train.reshape (60000, 28, 28, 1).astype('float32')
x_test=x_test.reshape (10000, 28, 28, 1).astype ('float32')
```

HOME PAGE (HTML)

```
<style>
    body{
     background-image: url('static/images/index6.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>
```

HOME PAGE (CSS)

```css
#clear_button{
  margin-left: 15px;
  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: greenyellow;
   padding-top: 1%;
   padding-bottom: 1%;
   font-weight: bold;
   font-family: 'Prompt', sans-serif;
}

#team_id{
   text-align: right;
   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;
}
```

```css
.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

```
13
14  app = Flask(__name__)
15  app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
16
17  model = load_model("mnistCNN.h5")
18
19
20  @app.route('/')
21  def index():
22      return render_template('index.html')
23
24
25  @app.route('/predict', methods=['GET', 'POST'])
26  def upload():
27      if request.method == "POST":
28          f = request.files["image"]
29          filepath = secure_filename(f.filename)
30          f.save(os.path.join(app.config['UPLOAD_FOLDER'], filepath))
31
32          upload_img = os.path.join(UPLOAD_FOLDER, filepath)
33          img = Image.open(upload_img).convert("L")  # convert image to monochrome
34          img = img.resize((28, 28))  # resizing of input image
35
36          im2arr = np.array(img)  # converting to image
37          im2arr = im2arr.reshape(1, 28, 28, 1)  # reshaping according to our requirement
38
39          pred = model.predict(im2arr)
40
41          num = np.argmax(pred, axis=1)  # printing our Labels
42
43          return render_template('predict.html', num=str(num[0]))
44
45
46  if __name__ == '__main__':
47      app.run(debug=True, threaded=False)
```

## index.html

```
<head>
  <title>Digit Recognition WebApp</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/iJTQUOhcWr7x9
  <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 src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1" c
  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM" cros
  <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',"");
          });
       });

```

```
<h1 class="welcome">IBM PROJECT
<div id="team_id">TEAM ID : PNT2022TMID05025</div>
</h1>
<section id="title">
  <h4 class="heading">Handwritten Digit Recognition Website</h4>
  <br><br>
    <p>
      The website is designed to predict the handwritten digit.
    </p>
  <p>
      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.</p>

    <br>
  <p> 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</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">  Clear  </button>
        </div>
```

## GITHUB

https://github.com/IBM-EPBL/IBM-Project-8922-1658938041

## PROJECT DEMO LINK

https://drive.google.com/file/d/1b8egZ_U12YF5csdvgfOsa56kmNLFX5NV/view?usp=sharing

32