ABSTRACT

Character recognition plays an important role in the modern world. It can solve more complex problems and makes humans' job easier. An example is handwritten character recognition. This is a system widely used in the world to recognize zip code or postal code for mail sorting. There are different techniques that can be used to recognize handwritten characters. Two techniques researched in this paper are Pattern Recognition and Artificial Neural Network (ANN). Both techniques are defined and different methods for each technique is also discussed. Bayesian Decision theory, Nearest Neighbor rule, and Linear Classification or Discrimination is types of methods for Pattern Recognition. Shape recognition, Chinese Character and Handwritten Digit recognition uses Neural Network to recognize them. Neural Network is used to train and identify written digits. After training and testing, the accuracy rate reached 99%.This accuracy rate is very high.

SI NO	TABLE OF CONTENTS	PAGE NO

1. INTRODUCTION

1.1 Project Overview :

Character recognition is becoming more and more important in the modern world. It helps humans ease their jobs and solve more complex problems. An example is handwritten character recognition which is widely used in the world. This system is developed for zip code or postal code recognition that can be employed in mail sorting. This can help humans to sort mails with postal codes that are difficult to identify. For more than thirty years, researchers have been working on handwriting recognition. Over the past few years, the number of companies involved in research on handwriting recognition has continually increased. The advance of handwriting processing results from a combination of various elements, for example: improvements in the recognition rates, the use of complex systems to integrate various kinds of information, and new technologies such as high quality high speed scanners and cheaper and more powerful CPUs. Some handwriting recognition system allows us to input our handwriting into the system. This can be done either by controlling a mouse or using a third-party drawing tablet. The input can be converted into typed text or can be left as an “ink object” in our own handwriting. We can also enter the text we would like the system to recognize into any Microsoft Office program file by typing. We can do this by typing 1s and 0s. This works as a Boolean variable. Handwriting recognition is not a new technology, but it has not gained public attention until recently. The ultimate goal of designing a handwriting recognition system with an accuracy rate of 100% is quite illusionary, because even human beings are not able to recognize every handwritten text without any doubt. For example, most people can not even read their own notes. Therefore there is an obligation for a writer to write clearly. In this paper, both Pattern Recognition and Neural Networks will be defined. Examples of types of Pattern Recognition and Neural Networks will be discussed. The advantages of using Neural Networks][2]to recognize handwritten characters will be listed. Finally, Artificial Neural Networks, using back-Propagation method will be used to train and identify handwritten digits.

1.2 Purpose :

Handwritten digit recognition is a created system that is used to recognize handwritten digits. The handwritten digit images get transformed into histograms and these histograms are fed into a neural network. This neural network outputs scores for matching the input digit against the ten possible digits (0-9). The data is trained and tested and it outputs the accuracy rate. The results can show us which numeral needs more training to reach high accuracies and which numeral the system had a difficulty to identify.

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 :

TEAM MEMBERS : C.Raguraman, M.Muthu, B.Magesh, T.Sabastin

PROJECT : A Novel Method for Handwritten Digit Recognition System

1. **TensorFlow**. MNIST for ML Beginners. 2017. Available online: https://www.tensorflow.org/get_started/mnist/beginners (accessed on 20 April 2018).

2. **LeCun, Y.; Cortes, C.; Burges, C.J.C.** The MNIST Database of Handwritten Digits. 2012. Available online: http://yann.lecun.com/exdb/mnist/ (accessed on 25 April 2018).

3. **Benenson, R**. Classification Datasets Results. 2016. Available online: http://rodrigob.github.io/are_we_Thereyet/build/classification_datasets_results.html (accessed on 21 May 2018).

4. **LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P**. Gradient-based learning applied to document recognition. Proc. IEEE 1998, 86, 2278–2324. [CrossRef]

5. **Belongie, S.; Malik, J.; Puzicha, J**. Shape matching and object recognition using shape contexts. IEEE Trans. Pattern Anal. Mach. Intell. 2002, 24, 509–522. [CrossRef]

6. **Keysers, D.; Deselaers, T.; Gollan, C.; Ney, H. Deformation** models for image recognition. IEEE Trans. Pattern Anal. Mach. Intell. 2007, 29, 1422–1435. [CrossRef] [PubMed]

7. **Kegl, B.; Busa-Fekete, R**. Boosting products of base classifiers. In Proceedings of the 26th Annual International Conference on Machine Learning, Montreal, QC, Canada, 14–18 June 2009; pp. 497–504.

8. **Decoste, D.; Schölkopf, B.** Training invariant support vector machines. Mach. Learn. 2002 ,46, 161–190.[CrossRef]

9. **Simard, P.; Steinkraus, D.; Platt, J.C**. Best Practices for Convolutional Neural Networks Applied to Visual Document Analysis. In Proceedings of the 7th International Conference on Document Analysis and Recognition, Edinburgh, UK, 3–6 August 2003; Volume 2, pp. 958–963.

10. **Deng, L.; Yu, D. Deep Convex Net**: A Scalable Architecture for Speech Pattern Classification. In Proceedings of the 12th Annual Conference of the International Speech Communication Association, Florence, Italy, 27–31 August 2011; pp. 2285–2288

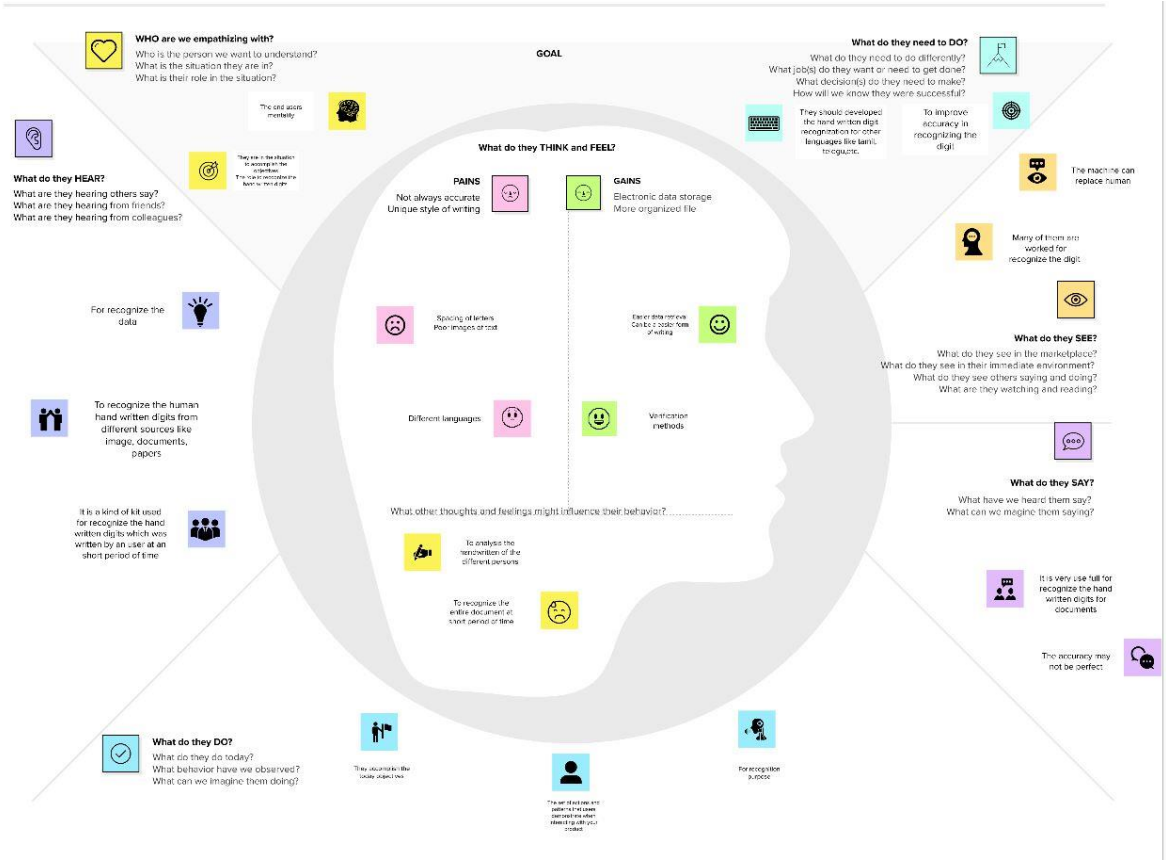
2.3 Problem Statement Definition :

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 AND PROPOSED SYSTEM

3.1 Empathy Map Canvas :

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.



3.1 Empathy Map Canvas

3.2 Brainstorm & Idea Prioritization Template :

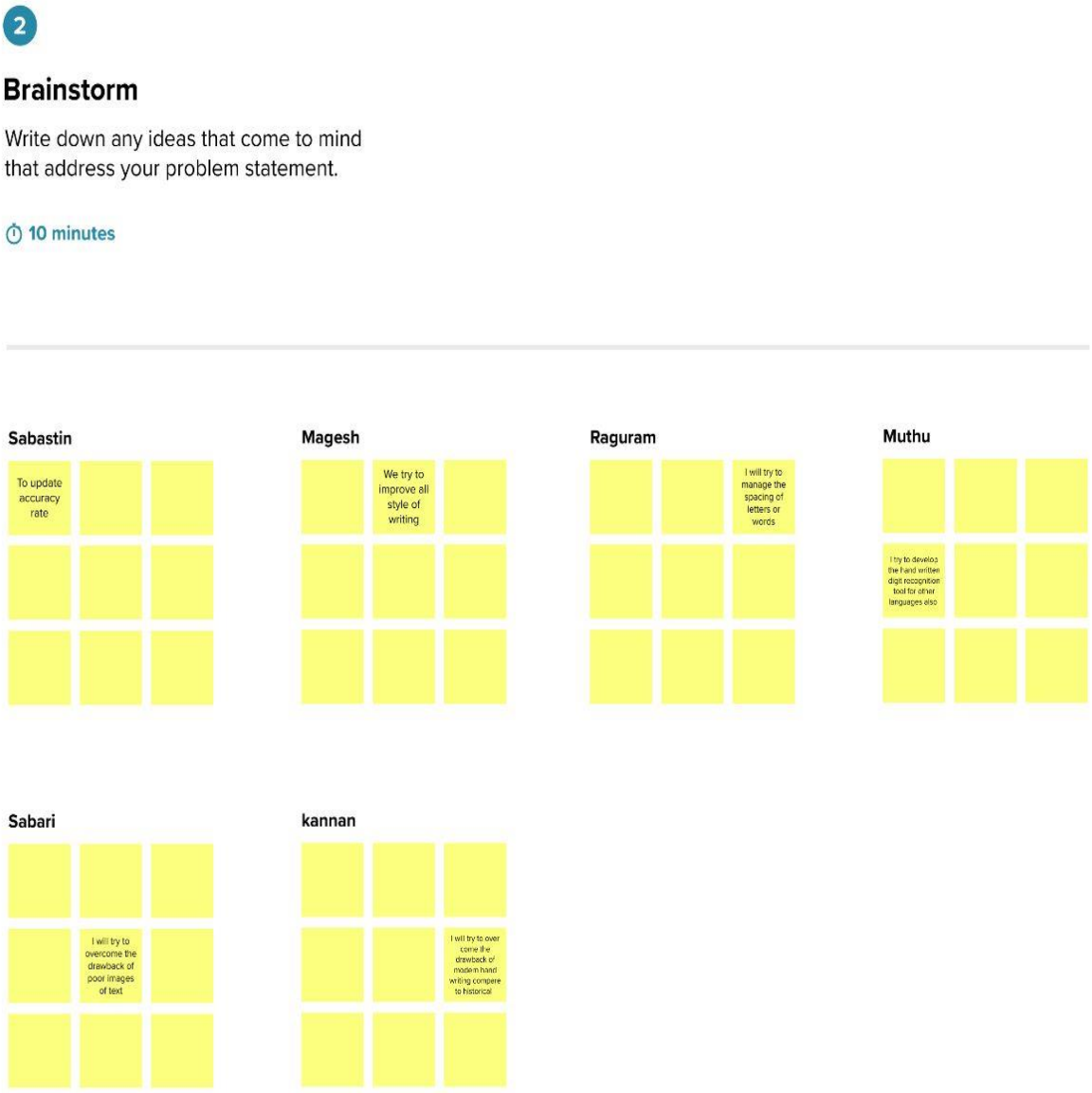
Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions. Use this template in your own brainstorming sessions so your team can

unleash their imagination and start shaping concepts even if you're not sitting in the same room.

Step-1: Team Gathering, Collaboration and Select the Problem Statement

3.2 Brainstorm & Idea Prioritization Template

Step-2: Brainstorm, Idea Listing and Grouping



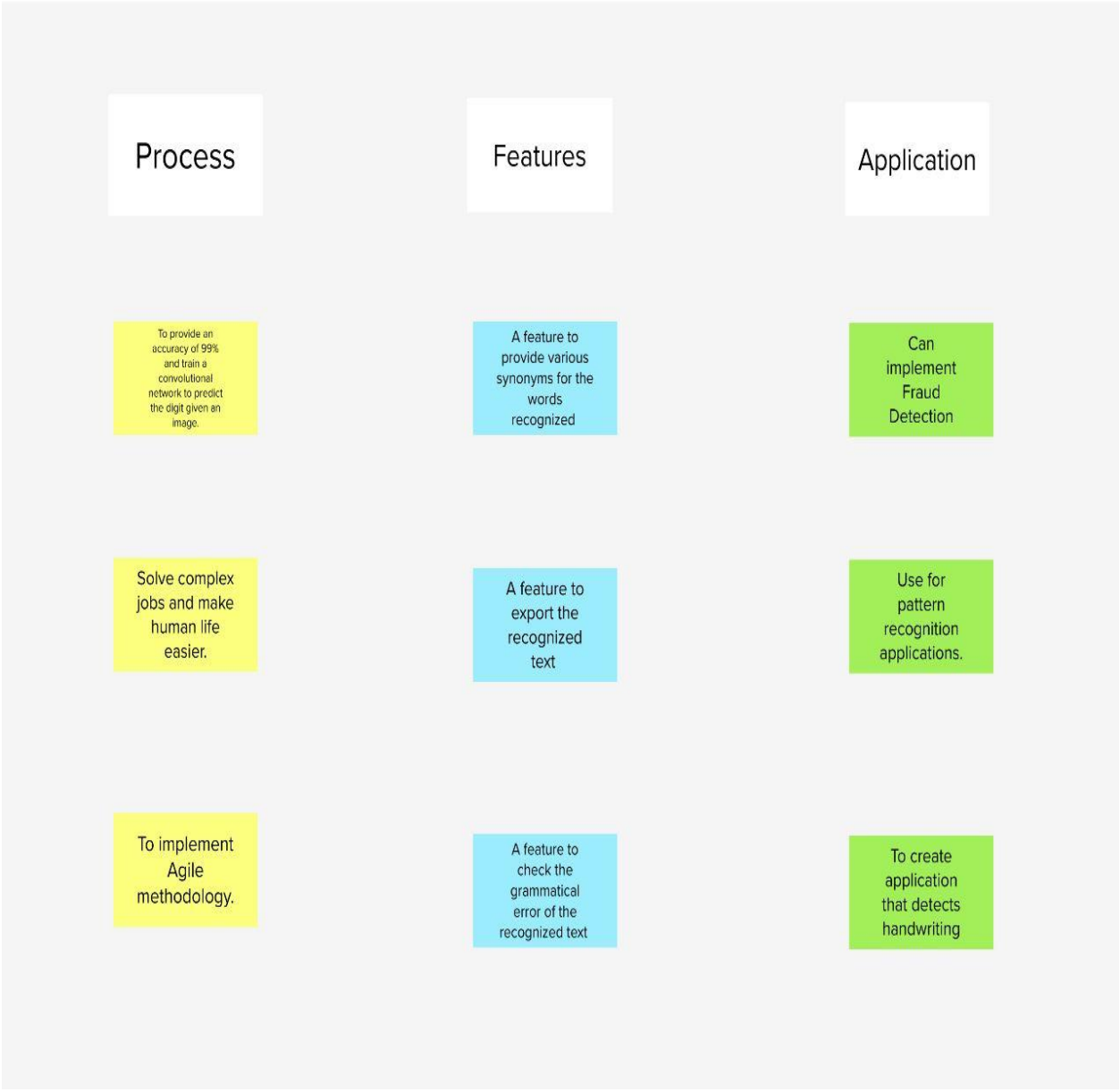
3.3 Brainstorm & Idea Prioritization Template

3

Group ideas

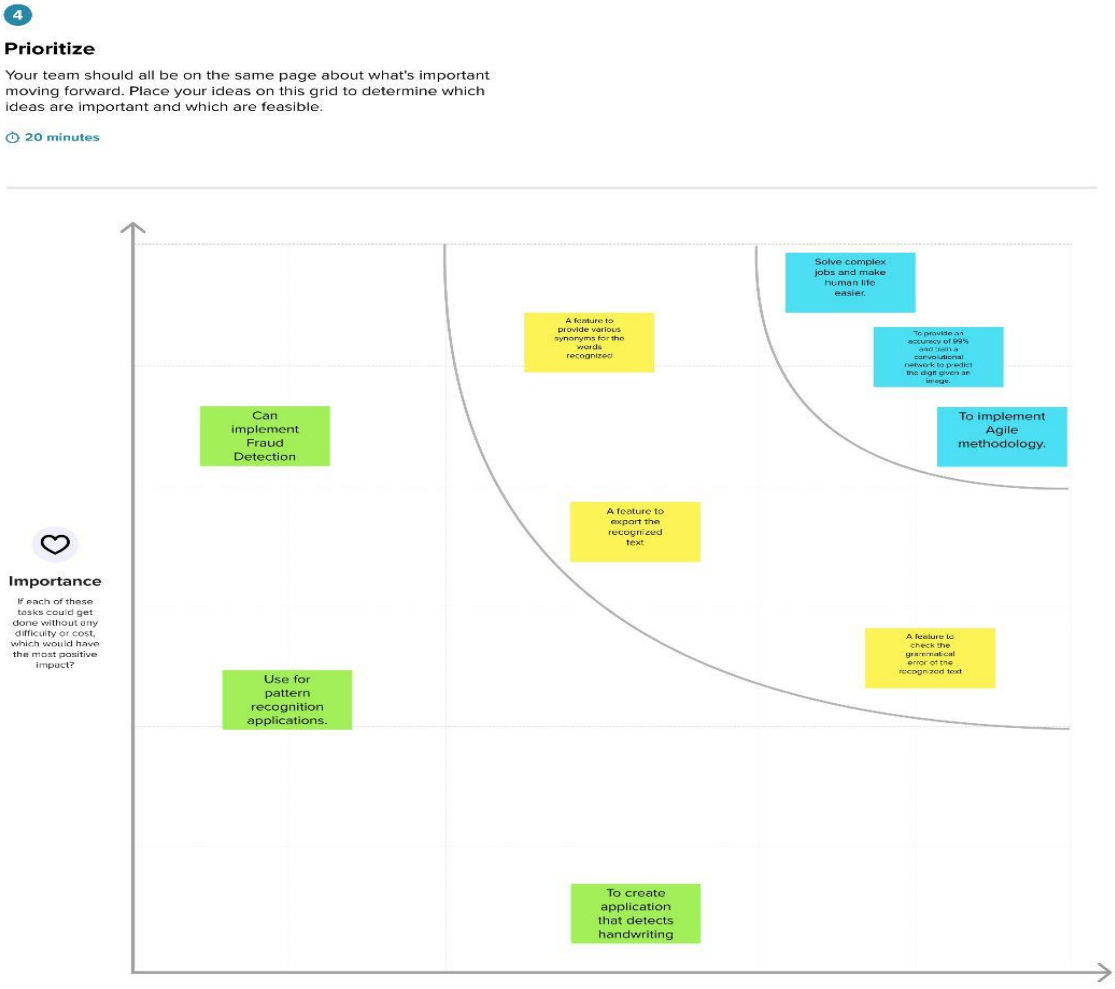
Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes



3.4 Brainstorm & Idea Prioritization Template

Step-3: Idea Prioritization



3.5 Brainstorm & Idea Prioritization Template

3.3 Proposed Solution Template :

Project team shall fill the following information in the proposed solution template

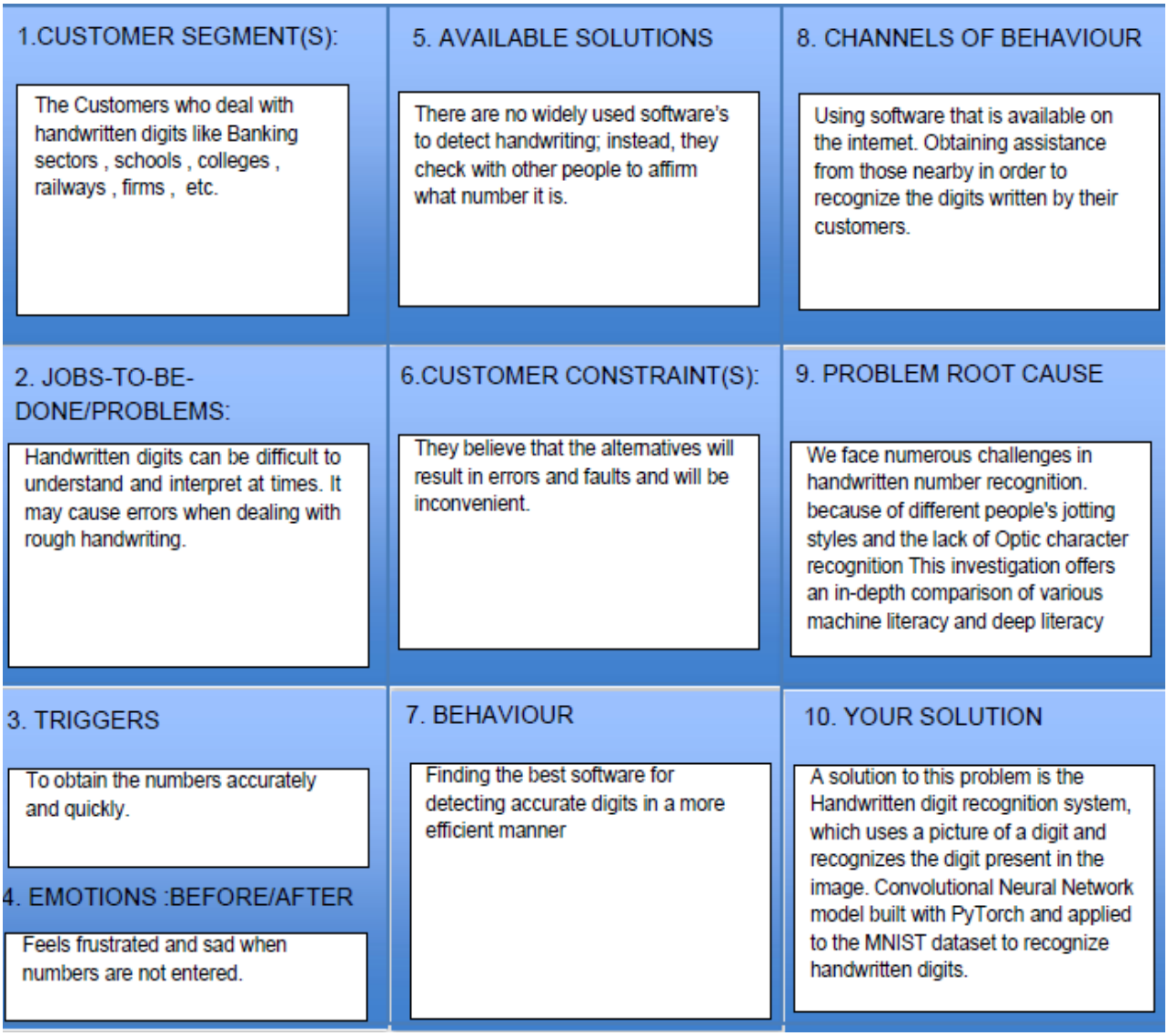
S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	In the modern world, digit recognition is crucial. It is capable of solving increasingly difficult problems and making humans' jobs easier. Handwritten digit recognition is one example. This is a worldwide system for recognizing zip codes or postal codes for mail sorting. Handwritten digit recognition can be accomplished using a variety of approaches. The machine has a difficult duty because handwritten

		digits are not flawless and can be generated with a variety of flavors. The solution to this issue is handwritten digit recognition, which uses an image of a digit and identifies the digit represented in the image.
2.	Idea / Solution description	Handwritten digit recognition is performed using the MNIST dataset which contains 60,000 training images of handwritten digits from zero to nine and 10,000 images for testing. So, the MNIST dataset has 10 different classes. In this project, we are going to implement a handwritten digit recognition application trained using the Convolutional Neural Networks model. In the end, a GUI is built where the user gives the handwritten digit as input where it is recognized and the result is displayed immediately.
3.	Novelty / Uniqueness	This project introduces an operative strategy for dealing with novelty in the handwritten visual recognition domain. A perfect transcription agent would be able to distinguish known and unknown characters in a picture, as well as determine any aesthetic variations that may occur inside or between texts. The existence of novelty has shown to be a major stumbling block for even the most robust machine learning-based algorithms for these activities. Novelty in handwritten papers might include, among other things, a change in the writer, character properties, writing attributes, or overall document appearance. Instead of examining each element separately, we believe that an integrated agent capable of processing known characters and novelties concurrently is a superior technique. The handwritten digit recognition problem can be seen as a subtask of the optical character recognition (OCR)

		problem.
4.	Social Impact/ Customer Satisfaction	There are many benefits associated with the handwriting recognition system. In addition to reading postal addresses and bank check amounts, it is also useful for reading forms. Furthermore, it's used in fraud detection because it makes it easy to compare two texts and determine which one is a copy. As a result, this system fulfills customers' expectations, as it is a novel method for recognizing handwritten digits, ensuring high accuracy for the model and meeting all customer expectations. Users will save a lot of time and effort if the system provides various synonyms for the words recognized. Due to the fact that the users in rural areas will be using their own regional language, this proposed system should be able to detect those digits as well. As the system is being used in socially crowded places such as banks to check amounts, it should be fast and reliable. As it is designed to solve real-world problems, it should be highly reliable and trustworthy in every way, and users throughout the world should be able to use it effectively
5.	Business Model (Revenue Model)	A revenue model means understanding how a startup can make money. Our major revenue sources consist of sales, government funds, and public donations. The introduction of novel ideas increases revenue streams, such as introducing gesture or touch features , voice read out of recognised digits, etc..
6.	Scalability of the Solution	One of the approaches to make the handwritten digit recognition system scalable is to make use of cloud-native methods. For example, one of the cloud solutions for making AI scalable is IBM Cloud. IBM Cloud Build helps run and manage AI models,

		optimize decisions at scale across any cloud. The advantage of using cloud to make solutions scalable is that we can deploy our AI application on the specific cloud environment that best supports our business needs. We can take advantage of builtin security capabilities and AI model monitoring. We can Automate AI lifecycles with ModelOps pipelines, deploy and run models through one-click integration and also prepare and build models visually and programmatically. Looking at these advantages, we can drive better business outcomes by optimizing our decisions and also make our solution scalable using cloud.

3.4 PROBLEM - SOLUTION FIT :



3.6 Problem - Solution Fit

4. REQUIREMENT ANALYSIS

4.1 Functional requirements :

Following are the functional requirements of the proposed solution.

FR No.	Sub Requirement (Story / Sub-Task)
FR-1	mage Data: Handwritten digit recognition refersto a computer's capacity to identify human handwritten digits from a variety of sources, such as photographs, documents, touch screens, etc., and categorize them into ten established classifications (0-9). In the realm of deep learning, this has been the subject of countless studies.
FR-2	Website: Web hosting makes the code, graphics, and other items that make up a website accessible online. A server hosts every website you've ever visited. The type of hosting determines how much space is allotted to a website on a server. Shared, dedicated, VPS, and reseller hosting are the four basic varieties.
FR-3	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 data first
FR-4	Cloud: The cloud offers a range of IT services, including virtual storage, networking, servers, databases, and applications. In plain English, cloud computing is described as a virtual platform that enables unlimited storage and access to your data over the internet.
FR-5	Modified National Institute of Standards and Technology dataset: The abbreviation MNIST stands for the MNIST dataset. It is a collection of 60,000 tiny square grayscale photographs, each measuring 28 by 28, comprising handwritten single digits between 0 and 9

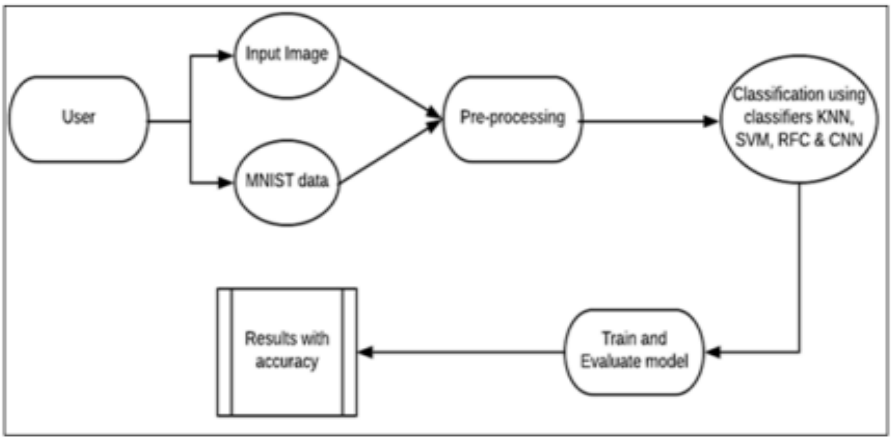
4.2 Non functional requirements :

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

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 bank checks, and sorting mail.
NFR-2	Security	1) The system generates a thorough description of the instantiation parameters, which might reveal information like the writing style, in addition to a categorization of the digit. 2) The generative models are capable of segmentation driven by 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 recognize handwritten numbers
NFR-4	Accuracy	With typed text in high -quality photos, optical character recognition (OCR) technology offers accuracy rates of greater than 99%. However, variances in spacing, abnormalities in handwriting, and the variety of human writing styles result in less precise character identification.
NFR-5	Availability	Open source

5. PROJECT DESIGN

5.1 Data flow diagram :



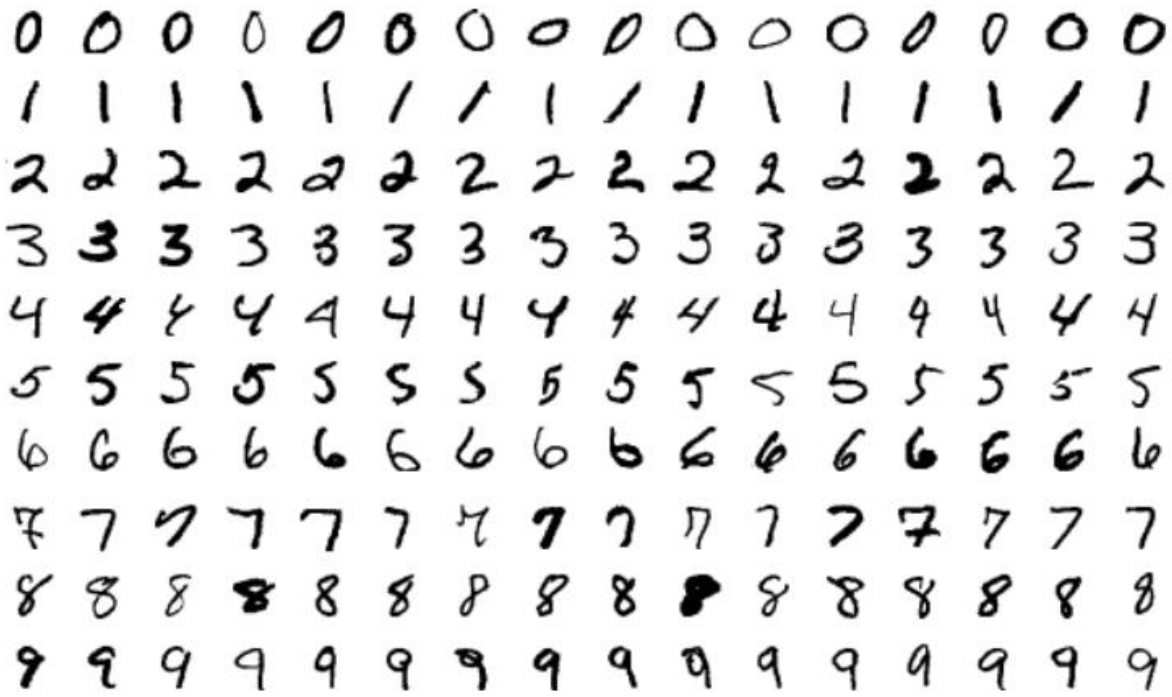
5.1 Data flow diagram

5.2 Solution & Technical Architecture :

Solution:

MNIST Dataset Description

The MNIST Handwritten Digit Recognition Dataset includes 60,000 training and 10,000 testing handwritten digit images. Each image has a height of 28 pixels and a width of 28 pixels, for a total of 784 (2828) pixels. Each pixel is connected with a single pixel value. It indicates how bright or dark that pixel is (larger numbers indicates darker pixel). This pixel value is an integer between 0 and 255.



PROCEDURE:

1. Install the latest TensorFlow library.

2. Prepare the dataset for the model.

3. Develop Single Layer Perceptron model for classifying the handwritten digits.

4. Plot the change in accuracy per epochs.

5. Evaluate the model on the testing data.

6. Analyse the model summary.

7. Add hidden layer to the model to make it Multi-Layer Perceptron.

8. Add Dropout to prevent overfitting and check its effect on accuracy.

9. Increasing the number of Hidden Layer neuron and check its effect on accuracy.

10. Use different optimizers and check its effect on accuracy.

11. Increase the hidden layers and check its effect on accuracy.

12. Manipulate the batch size and epochs and check its effect on accuracy.

MNIST is a dataset which is widely used for handwritten digit recognition. The dataset consists of 60,000 training images and 10,000 test images. The artificial neural

networks can all most mimic the human brain and are a key ingredient in image processing field. Handwritten digit recognition using MNIST dataset is a major project made with the help of Neural Network. It basically detects the scanned images of handwritten digits. We've taken it a step further, and our handwritten digit recognition technology not only recognizes scanned images of handwritten numbers, but also allows you to write digits on the screen and have them recognized using an integrated GUI.

Approach:

We will approach this project by using a three-layered Neural Network.

• The input layer: It distributes the features of our examples to the next layer for calculation of activations of the next layer.

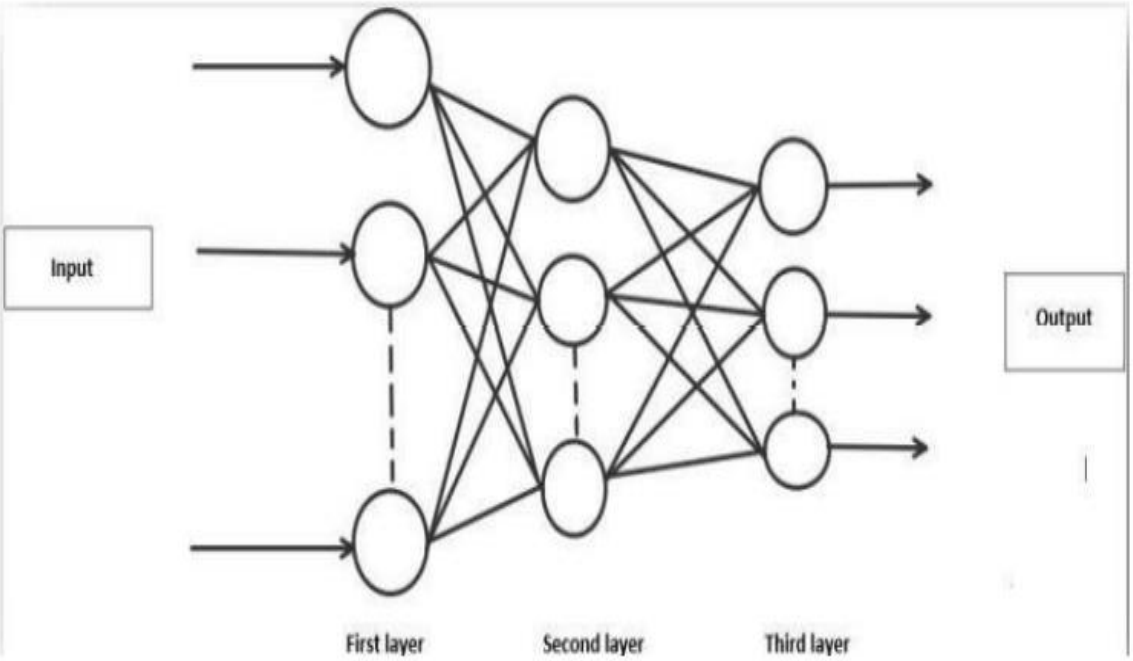
• The hidden layer: They are made of hidden units called activations providing nonlinear ties for the network. A number of hidden layers can vary according to our requirements.

• The output layer: The nodes here are called output units. It provides us with the final prediction of the Neural Network on the basis of which final predictions can be made.

A neural network is a model based on how the brain functions. It is made up of several layers with numerous activations, which mirror neurons in our brain. A neural network attempts to learn a set of parameters from a set of data, which may aid in recognising underlying links. Because neural networks can adapt to changing input, they can produce the best possible results without having to rethink the output criteria.

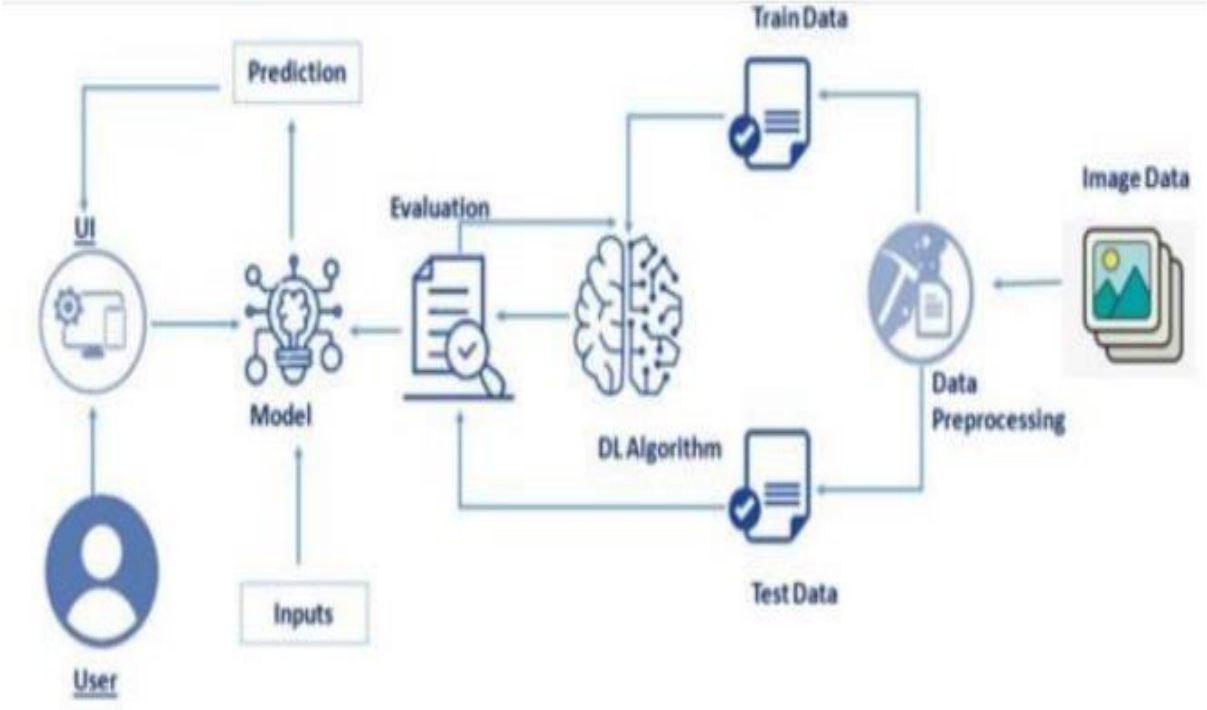
Methodology:

We created a Neural Network with one hidden layer and 100 activation units (excluding bias units). Data is loaded from a mat file, then features (X) and labels (Y) are extracted. Then, to avoid overflow during computation, features are divided by 255 and rescaled into a range of [0,1]. The data is divided into 60,000 training instances and 10,000 testing examples. Feedforward is used with the training set to calculate the hypothesis, followed by backpropagation to reduce the error between the layers. To solve the issue of overfitting, the regularization parameter lambda is adjusted to 0.1. The optimizer is run 70 times to get the best fit model.



5.2 Solution & Technical Architecture

Technical Architecture :



5.3 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-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail	I can register the application with Gmail	Medium	Sprint-2
	Login	USN-5	As a user, I can log into the application by entering email & password	I can login to the application	High	Sprint-1
	Home	USN-6	As a user, I can view the application's home page where I can read the instructions to use this application	I can read instructions also and the home page is user-friendly.	Low	Sprint-1
	Upload Image	USN-7	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-8	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-9	As a user, I will train and test the input to get the maximum accuracy of output.	I can able to train and test the application until it gets maximum accuracy of the result.	Medium	Sprint-4
Customer (Web user)	Accessibility	USN-10	As a user, I can use the web application virtually anywhere.	I can use the application in any device with a browser	Medium	Sprint-4

5.4 User Stories

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	Image Data	USN-1	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 categorize them into ten established classifications (0-9). In the realm of deep learning, this has been the subject of countless studies	1	High	Raguraman.C
Sprint-2	Website	USN-2	Web hosting makes the code, graphics, and other items that make up a website accessible online. A server hosts every website you've ever visited. The type of hosting determines how much space is	1	Medium	Muthu.M

			allotted to a website on a server. Shared, dedicated, VPS, and reseller hosting are the four basic varieties			
Sprint-3	Digit Classifier Model	USN-3	To train a convolutional network to predict the digit from an image, use the MNIST database of handwritten digits. get the training and validation data first.	1	High	Sabastin.T
Sprint-4	Cloud	USN-4	The cloud offers a range of IT services, including virtual storage, networking, servers, databases, and applications. In plain English, cloud computing is described as a virtual platform that enables unlimited storage and access to your data over the internet	1	Medium	Magesh.B
Sprint-5	MNIST	USN-1	The abbreviation MNIST stands for the MNIST dataset.	1	High	Raguraman.C

			It is a collection of 60,000 tiny square grayscale photographs, each measuring 28 by 28, comprising handwritten single digits between 0 and 9			

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	1	3 Days	24 Oct 2022	26 Oct 2022	1	26 Oct 2022
Sprint-2	1	3 Days	31 Oct 2022	02 Nov 2022	1	02 Nov 2022
Sprint-3	1	3 Days	07 Nov 2022	09 Nov 2022	1	09 Nov 2022
Sprint-4	1	3 Days	14 Nov 2022	16 Nov 2022	1	16 Nov 2022
Sprint-5	1	3 Days	17 Nov 2022	19 Nov 2022	1	19 Nov 2022

7. CODING & SOLUTIONS

7.1 Feature 1

Index.html

```
<html>

<head>

  <title>HDR</title>

  <meta name="viewport" content="width=device-width">

  <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">

  <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') }}">

  <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>

  <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/css/bootstrap.min.css">

  <script src="https://cdn.jsdelivr.net/npm/jquery@3.6.0/dist/jquery.slim.min.js"></script>

  <script
src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js"></script>

  <script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/js/bootstrap.bundle.min.js"></script>

</head>

<style>

    body{

     background-image: url('static/images/bc1.jpg');

     background-repeat: no-repeat;

     background-size: cover;

    }

</style>
```

```
<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>HandWritten Digit Recognition System</h1>
        <div class="container p-3 my-3 bg-dark text-white">
            <p>Handwritten Digit Recognition is a technology that is much needed in this world as of Today.This Digit Recognition System is used to recognize the digits from different sources like email, posts, cheque etc. Before proper implementation of this technology we have relied on writing text with our own hands which can result in error.It's difficult to store and access physical data with efficiency.The project presents in representing the recognization of handwritten digits (0 - 9) from the famous MNIST dataset. Here we will be using AlexNet which is an architecture of Convolutional Neural Network.</p>
        </div>
        <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" width="100px" height="100px"/>
          <div class="buttons_div">
            <button type="submit" class="btn btn-light">Predict</button>
            <button type="button" class="btn btn-light">  Clear  </button>
          </div>
        </form>
        </div>
    </section>
<!--
  <h1 class="welcome">IBM PROJECT
  <div id="team_id">TEAM ID : PNT2022TMID28666</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.

 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

</section>

-->

<!--<section id="content">

<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()">

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

<button type="button" class="btn btn-dark" id="clear_button"> Clear </button>

<button type="submit" class="btn btn-light">Predict</button>

<button type="button" class="btn btn-light"> Clear </button>

```
        </div>
      </form>
      </div>
  </section>-->
</body>
</html>
```

Predit.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:400px;
     height:150px;
     background-color: #000000;
```

```
    border-radius: 15px;
    position:absolute;
    box-shadow: 0px 0px 10px 5px white;
    top:25%;
    left:50%;
    transform:translate(-50%,-50%);
   }
   #head{
  text-align: center;
  font-size: 30px;
  margin: 0 auto;
  padding: 3% 5%;
  font-family: Arial, Helvetica, sans-serif;
  color: white;
   }
   #num{
       font-size: 50px;
   }
</style>
<body>
   <div id="rectangle">
       <h1 id="head">Predicted Number : <br><center id="num">{{num}}</center></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%;
    }
  }
```

7.2 Feature 2

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("./models/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)
```

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, .pjpeg"	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

8.1 Test

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

8.2 User Acceptance Testing

9. RESULTS

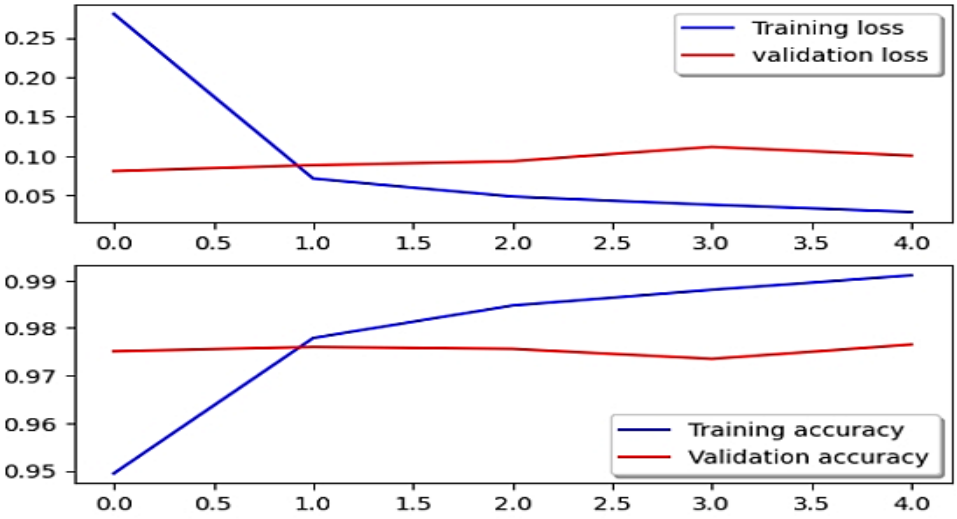
9.1 Performance Metrics :

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
```

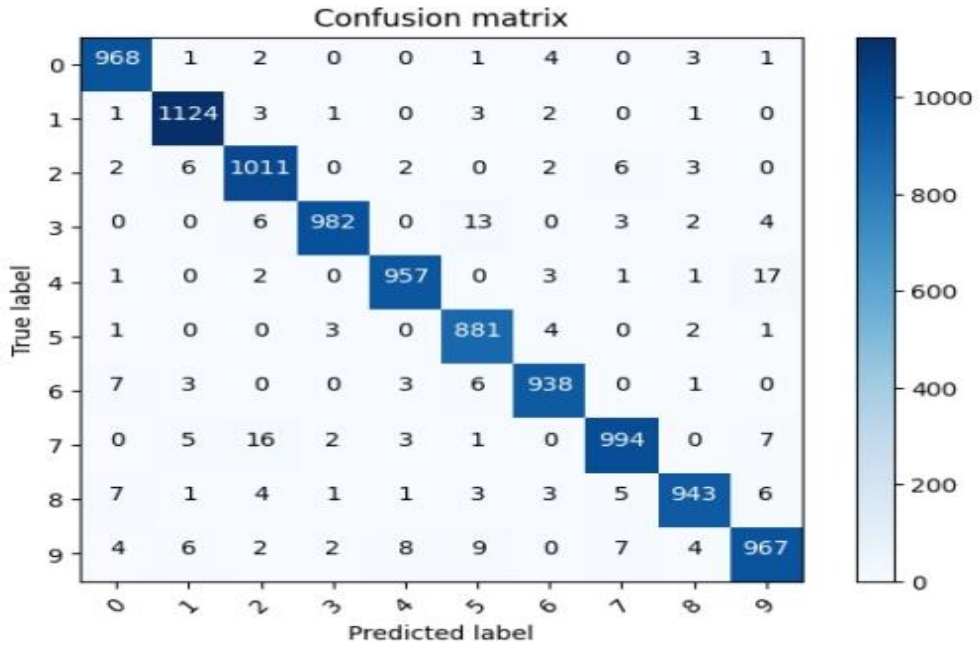
9.1 Performance Metrics (Model Summary)

Accuracy :



9.2 Performance Metrics (Accuracy)

Confusion Matrix :



9.3 Performance Metrics (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

9.4 Performance Metrics (Classification Report)

10. ADVANTAGES & DISADVANTAGES

Advantages :

1. Electronic data storage
2. More organized files
3. Easier data retrieval
4. Can be easier form of writing
5. Verification methods

Disadvantages :

1. Not always accurate
2. Unique style of writing
3. Spacing of letters or words
4. Different languages
5. Modern handwriting compared to historical

11. CONCLUSION

Using Neural Network system, back-propagation learning, to recognize handwritten digits was very successful. An image, which contained 100 samples of each number, was trained and tested. The accuracy rate of recognizing the number was 99%. This accuracy rate is very high. From the training and testing results, it was concluded that the system had more trouble identifying numeral 5“. This maybe caused by the fact that the digit is running together or maybe it is not fully connected. The system was not stable. It gave different training and testing results every day for each numeral. It will need to take a close look at the system and should look for improvements for the future. From the net-file, the system was able to produce an image-file. The image-file produced showed the recognized number. By looking at figure 5.2, it is concluded that the image-file produced does not show the numeral 5“ clear enough. This part will also need more improvements. Apart from the above problems and parts that need improvements, the overall recognition system was successful.

12. FUTURE SCOPE

1. Try to enhance the accuracy in future while recognizing the digit
2. Try to overcome the style of writing the digit
3. Try to recognizing the historical style

13. APPENDIX

13.1 Source code :

Index.html

```
<html>

<head>

  <title>HDR</title>

  <meta name="viewport" content="width=device-width">

  <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">

  <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') }}">

  <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>

  <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/css/bootstrap.min.css">

  <script src="https://cdn.jsdelivr.net/npm/jquery@3.6.0/dist/jquery.slim.min.js"></script>

  <script
src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js"></script>

  <script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/js/bootstrap.bundle.min.js"></script>

</head>

<style>

    body{

     background-image: url('static/images/bc1.jpg');

     background-repeat: no-repeat;

     background-size: cover;

    }

</style>

<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>HandWritten Digit Recognition System</h1>
      <div class="container p-3 my-3 bg-dark text-white">
         <p>Handwritten Digit Recognition is a technology that is much needed in this world as of Today.This Digit Recognition System is used to recognize the digits from different sources like email, posts, cheque etc. Before proper implementation of this technology we have relied on writing text with our own hands which can result in error.It's difficult to store and access physical data with efficiency.The project presents in representing the recognization of handwritten digits (0 - 9) from the famous MNIST dataset. Here we will be using AlexNet which is an architecture of Convolutional Neural Network.</p>
      </div>
      <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" width="100px" height="100px"/>
```

```
            <div class="buttons_div">
              <button type="submit" class="btn btn-light">Predict</button>
              <button type="button" class="btn btn-light">  Clear  </button>
            </div>
          </form>
          </div>
      </section>
<!--
    <h1 class="welcome">IBM PROJECT
  <div id="team_id">TEAM ID : PNT2022TMID28666</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" 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>
        <button type="submit" class="btn btn-light">Predict</button>
        <button type="button" class="btn btn-light">  Clear  </button>
      </div>
    </form>
```

```
        </div>
  </section>-->
</body>
</html>
```

Predit.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:400px;
     height:150px;
     background-color: #000000;
     border-radius: 15px;
     position:absolute;
```

```
    box-shadow: 0px 0px 10px 5px white;
    top:25%;
    left:50%;
    transform:translate(-50%,-50%);
   }
   #head{
  text-align: center;
  font-size: 30px;
  margin: 0 auto;
  padding: 3% 5%;
  font-family: Arial, Helvetica, sans-serif;
  color: white;
   }
   #num{
       font-size: 50px;
   }
</style>
<body>
   <div id="rectangle">
       <h1 id="head">Predicted Number : <br><center id="num">{{num}}</center></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%;
    }
  }
```

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("./models/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)
```

13.2 GitHub & Project Demo Link

GitHub Link :

https://github.com/IBM-EPBL/IBM-Project-47249-1660797600

Project Demo Link :

HWDR.mp4 - Google Drive