

# **IBM PROJECT DOCUMENTATION**

## **1 INTRODUCTION**

- 1.1 Project Overview
- 1.2 Purposes

## **2 LITERATURE SURVEY**

- 2.1 Existing problem
- 2.2 References
- 2.3 Problem Statement Definition

## **3 IDEATION & PROPOSED SOLUTION**

- 3.1 Empathy Map Canvas
- 3.2 Ideation & Brainstorming
- 3.3 Proposed Solution
- 3.4 Problem Solution fit

## **4 REQUIREMENT ANALYSIS**

- 4.1 Functional requirement
- 4.2 Non-Functional requirements

## **5 PROJECT DESIGN**

- 5.1 Data Flow Diagrams
- 5.2 Solution & Technical Architecture
- 5.3 User Stories

## **6 PROJECT PLANNING & SCHEDULING**

- 6.1 Sprint Planning & Estimation
- 6.2 Sprint Delivery Schedule
- 6.3 Reports from JIRA

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

- 7.1 Feature 1

7.2 Feature 2  
7.3 Database Schema (if Applicable)

## **8 TESTING**

8.1 Test Cases  
8.2 User Acceptance Testing

## **9 RESULTS**

9.1 Performance Metrics

## **10 ADVANTAGES & DISADVANTAGES**

## **11 CONCLUSION**

## **12 FUTURE SCOPE**

## **13 APPENDIX**

Source Code & GitHub & Project Demo Link

# **1.INTRODUCTION**

## **1.1 Project Overview**

The handwritten digit recognition is the ability of computers to recognize human handwritten digits. It is a hard task for the machine because handwritten digits are not perfect and can be made with many different flavors. The handwritten digit recognition is the solution to this problem which uses the image of a digit and recognizes the digit present in the image.

All the Python codes are first written in Jupyter notebook supplied along with Anaconda Python and then the codes are tested in IBM cloud. Finally a web based framework is designed with help Flask a Python library. There are 2 html files and CSS are created in templates folder along with their

associated files in static folder. The Python program 'app.py' used to interface with these two webpages is written in Spy-der-Anaconda python and tested.

## **1.2 Purpose**

Handwritten character recognition is one of the practically important issues in pattern recognition applications. The applications of digit recognition includes in postal mail sorting, bank check processing, form data entry.

## **2.LITERATURE SURVEY**

### **2.1 Existing problem**

The handwritten digit recognition is the capability of computer applications to recognize the human handwritten digits. It is a hard task for the machine because handwritten digits are not perfect and can be made with many different shapes and sizes. The handwritten digit recognition system is a way to tackle this problem which uses the image of a digit and recognizes the digit present in the image. Convolutional Neural Network model created using PyTorch library over the MNIST dataset to recognize handwritten digits .

Handwritten Digit Recognition is the capability of a computer to fete the mortal handwritten integers from different sources like images, papers, touch defenses, etc, and classify. them into 10 predefined classes (0-9). This has been a Content of bottomless- exploration in the field of deep literacy. Number recognition has numerous operations like number plate recognition, postal correspondence sorting, bank check processing, etc . (2). In Handwritten number recognition, we face numerous challenges . because of different styles of jotting of different peoples as it . is not an Optic character recognition. This exploration provides a comprehensive comparison between different machine literacy and deep literacy algorithms for the purpose of handwritten number recognition. For this, we've used Support .

Vector Machine, Multilayer Perceptron, and Convolutional . Neural Network. The comparison between these algorithms is carried out on the base of their delicacy, crimes, and .testing- training time corroborated by plots and maps that have been constructed using matplotlib for visualization.

## 2.2 References

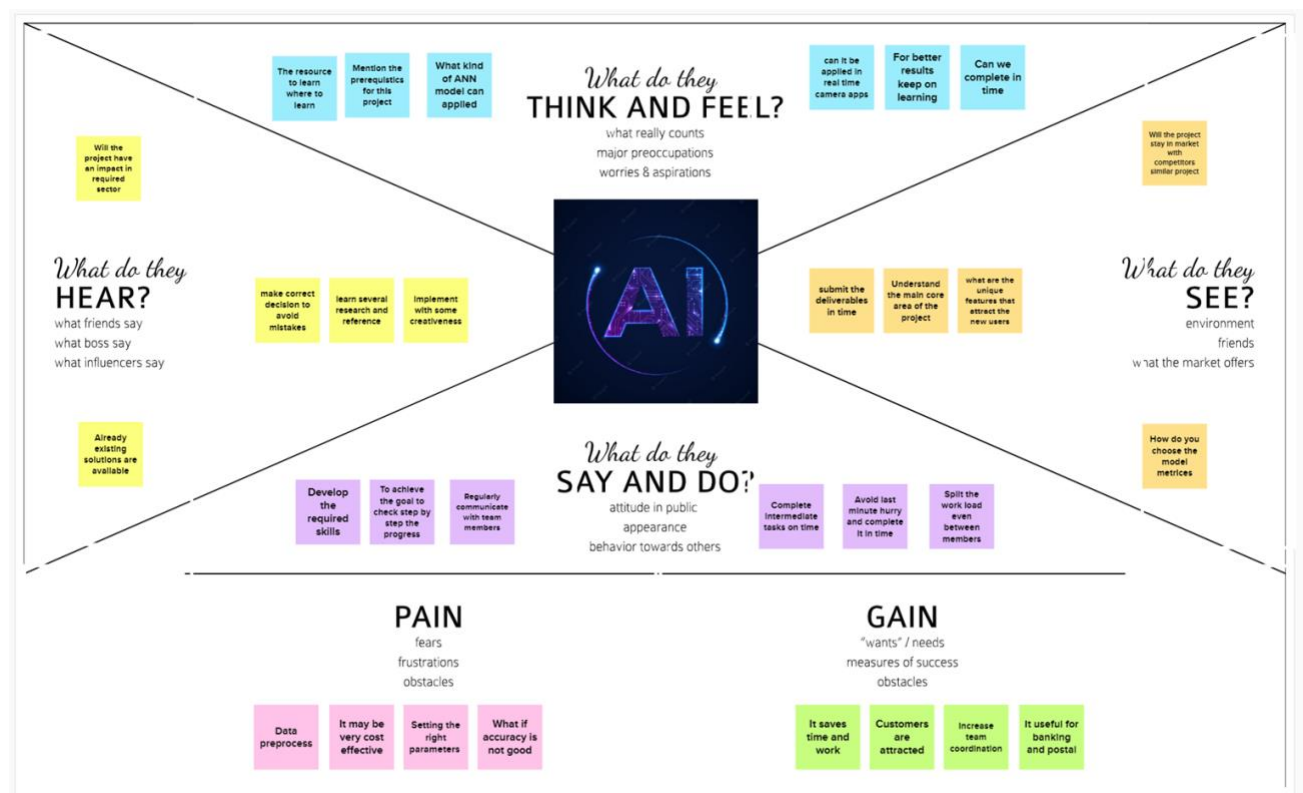
- [1] R. Jayadevan, S. R. Kolhe, P. M. Patil, and U. Pal, "Automatic processing of handwritten bank cheque images: a survey," Int. J. Doc. Anal. Recognit. IJDAR, vol. 15, no. 4, pp. 267–296, 2012.
- [2] N. M. Nasrabadi, "Pattern recognition and machine learning," J. Electron. Imaging, vol. 16, no. 4, p. 049901, 2007.

## 2.3 Problem Statement Definitions

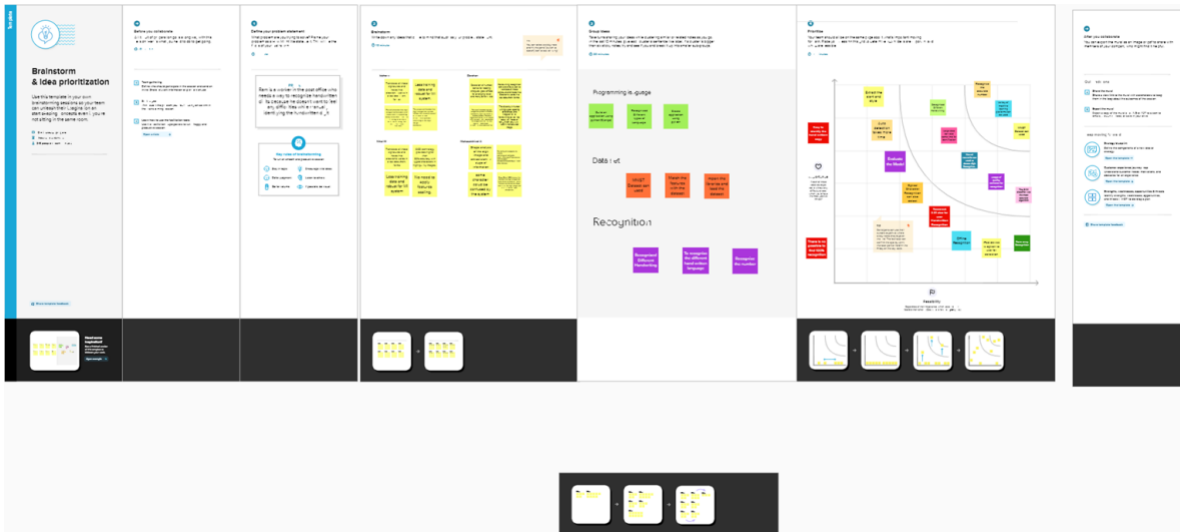
The problem statement is to classify handwritten digits. The goal is to take an image of a handwritten digit and determine what that digit is The digits range from zero (0) through nine (9).

## 3.IDEATION & PROPOSED SOLUTION

### 3.1 Empathy Map Canvas



## 3.2 Ideation & Brainstorming



## 3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Probleto be solved)	<p>Statement-The handwritten digit recognition is the capability of computer applications to recognize the human handwritten digits.</p> <p>Description: To make the machine to get recognize the available digits written in any form of descriptions by giving many sample images with various constraints.</p>
2.	Idea/Solution description	<p>It is the capability of a computer to feed the mortal handwritten integers from various input devices.</p> <p>It allows users to define the available solutions to find the answers on the search engine to improve knowledge of the students.</p>
3.	Novelty/Uniqueness	Accurately recognize the digits like OCR, specifically for numbers.

<b>4.</b>	Social Impact/ CustomerSatisfaction	Difficulties for the people during the recognition of digits is reduced. Rather than identifying the characters, it identifies specifically numbersbefore definig the recognition.
<b>5.</b>	Business Model (RevenueModel)	It is used to recognize the register number of the candidates while writing the exams. It can be integrated with banking systems to identify and recognize the account numbers and amount details easily.

## 4.REQUIREMENT ANALYSIS

### 4.1 Functional Requirements

Following are the functional requirements of the proposed solution.

FR No	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	No Registration needed.
FR-2	Image	Image of the paper and any documents to check for the right digits.
FR-3	PC / Desktop or Mobile	Device needed to run the application

### 4.2 Non-Functional Requirement

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

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Used to find the exact digit.
NFR -2	Security	No security needed while using this application
NFR -3	Reliability	Reliable on anytime anywhere with the usage

NFR -4	Performance	High performance to find the digit as very clear and exactly
NFR -5	Availability	Available for customers on both web and mobile
NFR -6	Scalability	Scalable with finding the digit as earlier as possible and detect all digits.

## 5.PROJECT DESIGN

### 5.1 Data Flow Diagram

#### User Stories :

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

Customer usage types	Functional Requirement	User Story Numbers	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile View)	Home	USN-1	As a user, I can view the guide and awareness to use this application.	I can view the awareness to use this application and its limitations.	Low	Sprint-1
		USN-2	As a user, I'm allowed to view the guided video to use the interface of this application.	I can gain knowledge to use this application by a practical method.	Low	Sprint-1
		USN-3	As a user, I can read the instructions to use this application.	I can read instructions also to use it in a user-friendly method.	Medium	Sprint-2
	Recognize	USN-4	As a user, In this prediction page I get to choose the image.	I can choose the image from our local system and predict the output.	High	Sprint-2
	Predict	USN-6	As a user, I'm Allowed to upload and choose the image to be uploaded	I can upload and choose the image from the system storage and also in any virtual storage.	High	Sprint-3
		USN-7	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.	High	Sprint-4
		USN-8	As a user, I can access the MNIST data set	I can access the MNIST data set to produce the accurate result.	Medium	Sprint-3

Customer (Web View)	Home	USN-9	As a user, I can view the guide to use the web app.	I can view the awareness of this application and its limitations.	Low	Sprint-1
<b>Customer usage types</b>	<b>Functional Requirement</b>	<b>User Story Number</b>	<b>User Story / Task</b>	<b>Acceptance criteria</b>	<b>Priority</b>	<b>Release</b>
Customer (Mobile View)	House	USN-1	As a user, I can view the guide and awareness to use this application.	I can view the awareness to use this application and its limitations.	Low	Sprint-1
		USN-2	As a user, I'm allowed to view the guided video to use the interface of this application.	I can gain knowledge to use this application by a practical method.	Low	Sprint-1
		USN-3	As a user, I can read the instructions to use this application.	I can read instructions also to use it in a user-friendly method.	Low	Sprint-2
	Recognize	USN-10	As a user, I can use the web application virtually anywhere.	I can use the application portably anywhere.	High	Sprint-1
		USN-11	As it is an open source, can use it cost freely.	I can use it without any payment to be paid for it to access.	Medium	Sprint-2
		USN-12	As it is a web application, it is installation free	I can use it without the installation of the application or any software.	Medium	Sprint-4
	Predict	USN-13	As a user, I'm Allowed to upload and choose the image to be uploaded	I can upload and choose the image from the system storage and also in any virtual storage.	Medium	Sprint-3

## 6.PROJECT PLANNING & SCHEDULING

### 6.1 Sprint Planning

1. Planning is a crucial role in project management because it allows team members to schedule their time on the project.
2. This activity demonstrates how the team members assigned and completed various tasks!
3. In Project we can Split into the Four Step of Phrases are

- Phrase 1: Information Collection and Requirement Analysis
- Phrase 2: Project Planning and Developing Modules
- Phrase 3: Implementing the High Accuracy Machine Learning Algorithm to Perform
- Phrase 4: Deploying the Model on Cloud and Testing the Model and UI Performance

### 6.1 Estimation

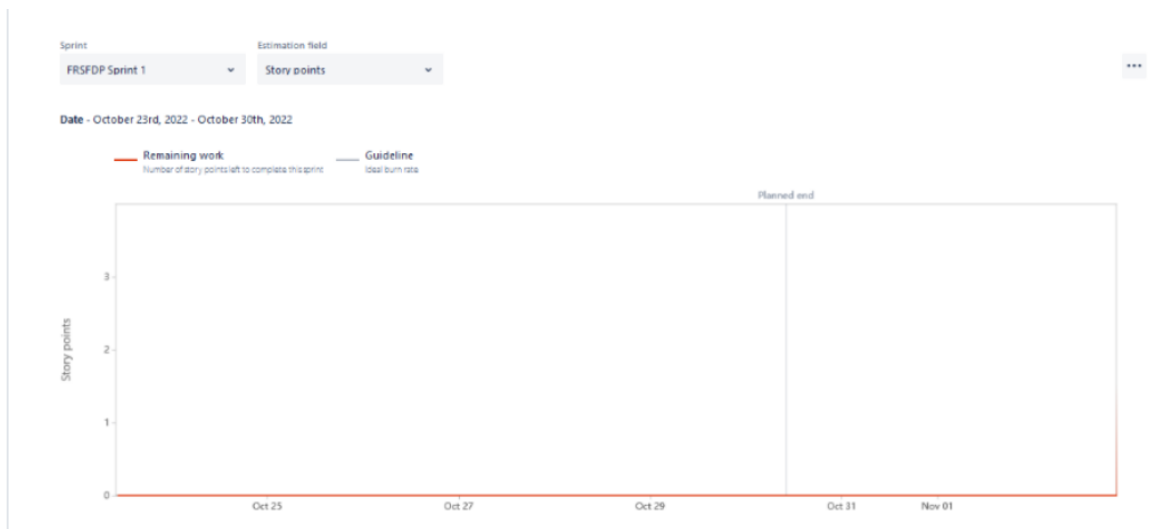
<b>Sprint</b>	<b>Functional Requirement (Epic)</b>	<b>User Story Number</b>	<b>User Story / Task</b>	<b>Team Members</b>
Sprint-1	Registration	USN-1	Each user should have the phone number to use this service and verified using otp and passwords should be created.	4
Sprint-2	Login	USN-2	User can Login by using their respective user id and password.	4
Sprint-3	Upload Image of hand written document	USN-3	The input from the user to find the digits should be uploaded.	4
Sprint-4	Recognition and Results.	USN-4	The userinput got processed and the result outcome will be the clear digits.	4



## 6.2 Sprint Delivery Schedule

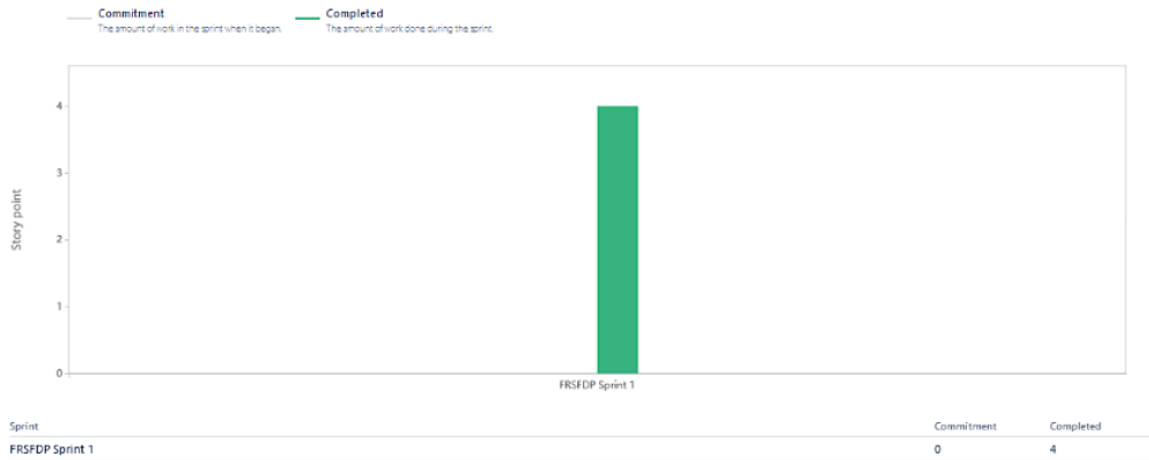
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022

## 6.3 Reports From JIRA



## Velocity report

[How to read this report](#)



## 7.CODING & SOLUTIONING

### HOME

IBM x IBM-12295-1662558326 x Inbox (1,734) - kishorkity278@gn x Digit Recognition WebApp x +

← → ↻ 127.0.0.1:5000 🔍 📄 ☆ 🖨️ ⓘ ⋮

IBM PROJECT

Handwritten Digit Recognition Website

The website is designed to predict the handwritten digit.

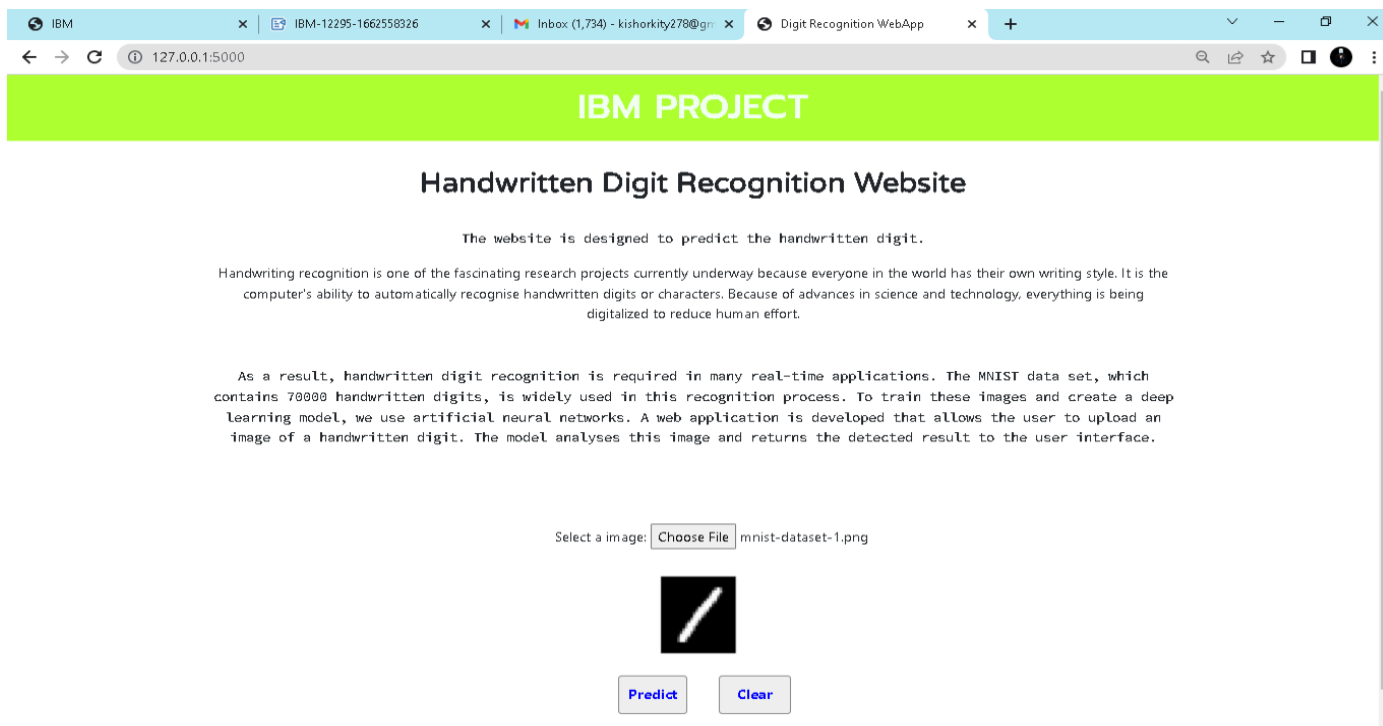
Handwriting recognition is one of the fascinating research projects currently underway because everyone in the world has their own writing style. It is the computer's ability to automatically recognise handwritten digits or characters. Because of advances in science and technology, everything is being digitalized to reduce human effort.

As a result, handwritten digit recognition is required in many real-time applications. The MNIST data set, which contains 70000 handwritten digits, is widely used in this recognition process. To train these images and create a deep learning model, we use artificial neural networks. A web application is developed that allows the user to upload an image of a handwritten digit. The model analyses this image and returns the detected result to the user interface.

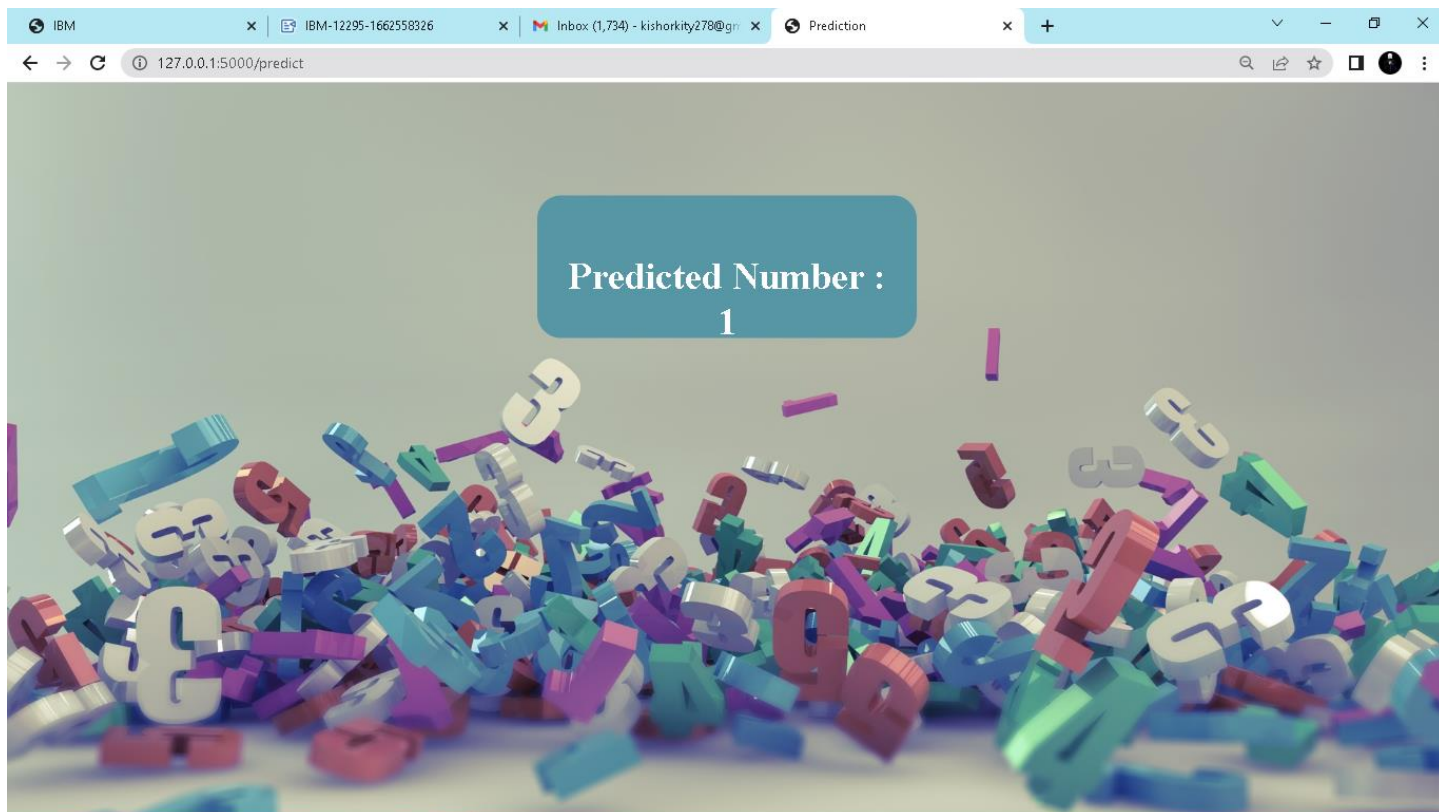
Select a image: Choose File No file chosen

Predict Clear

### INSERTION



## PREDICT



## 8. TESTING

### 8.1 Test Cases

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	BUG ID	Executed By
Test Case - D1	Functional	web Page	Verify user is able to see the page popup when user they enter to web page		1.Enter URL and click go		webpage should display	Working as expected	Pass				Darshan.B. Nihal.M
Test Case - D2	UI	Home Page	Verify the UI elements in Homepage		1.Enter URL and click go 2.webpage displayed		Application should show below UI elements: a.Homepage b.Introduction c.Launch	Working as expected	Pass				Mahasakthivel.A Kishor.V
Test Case - D3	Functional	Home page	Verify user is able to See the details about the webpage		1.Enter URL and click go 2.Webpage Displayed 3.Displays necessary details		User should navigate to homepage	Working as expected	Pass				Nihal.M. Mahasakthivel.A
Test Case - D4	Functional	Introduction	Verify user is able to details about uses of the gesture based tool and its importance		1.Enter URL and click go 2.Webpage Displayed 3.Displays necessary details 4.click introduction to go Displays about the uses of the gesture based tool		user should navigate to introduction	Working as expected	Pass				Kishor.V.      Nihal.M
Test Case - D5	Functional	Launch	Verify user is able to navigate to launch		1.Enter URL and click go 2.Webpage Displayed 3.Displays necessary details 4.click introduction to go Displays about the uses of the gesture based tool 5.Click Launch to navigates to launch page		user should navigate to Launch	Working as expected	Pass				Kishor.V. Darshan.B
Test Case - D6	Functional	Launch	Verify user is able to upload image and predicts the Handwritten Digit.		1.Enter URL and click go 2.Webpage Displayed 3.Displays necessary details 4.click introduction to go Displays about the uses of the gesture based tool 5.Click Launch to navigates to launch page 6.uploads the image and predicts using the hand gesture	Upload image : img1.jpg	Application should Display the images in different types of images like blurred , rotated image	Working as expected	Pass				Darshan.B. Nihal.M

### 8.2 User Acceptance Testing

## 1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [ProductName] project at the time of the release to User Acceptance Testing (UAT).

## 2. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	6	2	0	1	9
Duplicate	2	0	3	0	5
External	3	2	0	0	5
Fixed	10	3	2	16	31
Not Reproduced	0	0	0	0	0
Skipped	0	0	1	1	2
Won't Fix	0	2	1	2	5
Totals	21	9	7	20	57

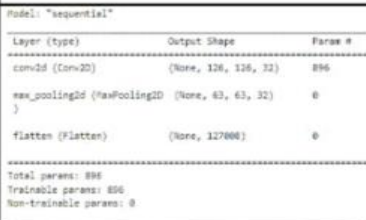
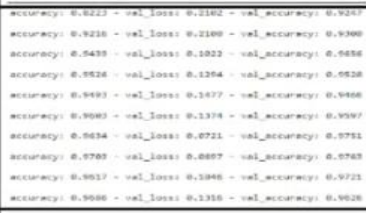
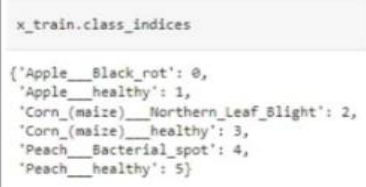
## 3. Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	10	2	0	10
Security	0	0	0	0
Outsource Shipping	6	0	0	6
Exception Reporting	6	0	0	6
Final Report Output	5	0	0	5
Version Control	1	0	0	1

## 9.RESULTS

### 9.1Performance Metrics

S.No.	Parameter	Values	Screenshot
1.	Model Summary	Model value - 896	 <pre> Model: "sequential" Layer (type)                Output Shape                 Param # ----- conv2d (Conv2D)              (None, 128, 128, 32)        896 max_pooling2d (MaxPooling2D) (None, 63, 63, 32)          0 flatten (Flatten)             (None, 127680)              0 Total params: 896 Trainable params: 896 Non-trainable params: 0           </pre>
2.	Accuracy	Training Accuracy – 0.9686 Validation Accuracy – 0.9626	 <pre> accuracy: 0.6223 - val_loss: 0.2102 - val_accuracy: 0.9267 accuracy: 0.9236 - val_loss: 0.2100 - val_accuracy: 0.9300 accuracy: 0.5439 - val_loss: 0.1022 - val_accuracy: 0.9456 accuracy: 0.9526 - val_loss: 0.1294 - val_accuracy: 0.9526 accuracy: 0.9493 - val_loss: 0.1477 - val_accuracy: 0.9406 accuracy: 0.9080 - val_loss: 0.1374 - val_accuracy: 0.9597 accuracy: 0.9634 - val_loss: 0.0721 - val_accuracy: 0.9751 accuracy: 0.9769 - val_loss: 0.0887 - val_accuracy: 0.9769 accuracy: 0.9617 - val_loss: 0.1046 - val_accuracy: 0.9721 accuracy: 0.9686 - val_loss: 0.1316 - val_accuracy: 0.9626           </pre>
3.	Confidence Score (Only Yolo Projects)	Class Detected Confidence Score - 96	 <pre> x_train.class_indices {'Apple__Black_rot': 0,  'Apple__healthy': 1,  'Corn_(maize)__Northern_Leaf_Blight': 2,  'Corn_(maize)__healthy': 3,  'Peach__Bacterial_spot': 4,  'Peach__healthy': 5}           </pre>

## 10.ADVANTAGES & DISADVANTAGES

### Advantages:

- ✓ The main purpose is to recognize corrupted handwritten digits using recognition method. Image of connected digits, disjoint digits and overlapped digits was divided into sub-images.
- ✓ Our study serves as a reference for aspiring researchers in automatic handwritten digits or hand recognition and classification.
- ✓ The experimental results of this study showed that conventional neural network is the best suitable algorithm or classifier for automatic handwritten digits on documented images accuracy rate is also high when compared with other algorithms like SVM and ANN .

### **Disadvantages:**

- ✓ Computation cost of the proposed system
- ✓ It could be indeployed in connected digit , overlapped digit and spaced digits.
- ✓ Some of the threats need to be solve within the given validity.

## **11.CONCLUSION**

In this paper an extensive review of recent advancement in the field of handwritten numeric digit classification and recognition has been presented. The review presented covers all the aspects for handwritten as well as printed digit recognition like off-line and on-line recognition, different features used, and finally various types of classifies recently used for digit classification. Moreover, all the important and recent works have been discussed with their advantages and limitations. It has been also discussed, most of the available systems were developed for particular database and yet to analyze over real time handwritten digits.

## **12.FUTURE SCOPE**

The proposed model in this project work can be extended to image recognition. The entire model can be converted to application software using python to exe software. The real time image classification, image recognition and vide processing are possible with help OpenCV python library This project work can be extended for security applications such as figure print recognition, inis recognition and face recognition.

## 13.APPENDIX

### 13.1 SOURCE CODE

#### INDEX.HTML

```
<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/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-
```



```
U02eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIly6OrQ6VrjIEaFf/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">IBM PROJECT
</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>
  Handwriting recognition is one of the fascinating research projects currently underway because
  everyone
  in the world has their own writing style. It is the computer's ability to automatically recognise
  handwritten
  digits or characters. Because of advances in science and technology, everything is being digitalized
  to reduce human effort.

  </p>

  <br>
```

<p> As a result, handwritten digit recognition is required in many real-time applications. The MNIST data set, which contains 70000 handwritten digits, is widely used in this recognition process. To train these images and create a deep learning model, we use artificial neural networks. A web application is developed that allows the user to upload an image of a handwritten digit. The model analyses this image and returns the detected result to the user interface.</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>

## PREDICT.HTML

<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=s
wap" rel="stylesheet">
<!-- bootstrap -->
<link
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-JjSmVgyd0p3pXB1rRibZUAYyoly6OrQ6VrjIEaFf/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">IBM PROJECT
</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>
  Handwriting recognition is one of the fascinating research projects currently underway because
  everyone
  in the world has their own writing style. It is the computer's ability to automatically recognise
  handwritten
  digits or characters. Because of advances in science and technology, everything is being digitalized
  to reduce human effort.

  </p>

  <br>
  <p> As a result, handwritten digit recognition is required in many real-time applications. The MNIST
  data set,
  which contains 70000 handwritten digits, is widely used in this recognition process. To train these
  images
  and create a deep learning model, we use artificial neural networks. A web application is developed
  that allows
  the user to upload an image of a handwritten digit. The model analyses this image and returns the
  detected result
  to the user interface.</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>
```

## STYLE.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;  
}
```

```
.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 PIL import Image
from flask import Flask, request, render_template
from keras.models import load_model
from werkzeug.utils import secure_filename

UPLOAD_FOLDER = './uploads'

```

```

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

model = load_model("digit-recognition.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)

```

**Github Link :** <https://github.com/IBM-EPBL/IBM-Project-12295-1659446816>