A NOVEL METHOD FOR HADNWRITTEN DIGIT RECOGNITION SYSTEM

THE PROJECT REPORT

Submitted by

Team ID : PNT2022TMID22012

KANIMOZHI.M(113019104042)

DHIVYA.V(113019104025)

INDHIRA.K(113019104035)

ANKAM USHA SHREE(113019104009)

BACHELOR OF ENGINEEERING

IN

COMPUTER SCIENCE AND ENGINEERING

TABLE OF CONTENTS

ABSTRACT

Handwritten digit 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,from data entry,etc. The heart of the problem lies within the ability to users by the way of a scanner,tablet and other digital devices.This paper presents an approach to off-line handwritten digit recognition based on different techniques The main objective of this paper is to ensure effective and reliable approaches for recognition of handwritten digits

LIST OF FIGURES

Chapter 1
INTRODUCTION

1.1Project Overview:

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.

1.2 Purpose :

The applications of digit recognition include in **postal mail sorting, bank check processing, form data entry**, etc. The main problem lies within the ability on developing an efficient algorithm that can recognize hand written digits, which is submitted by users by the way of a scanner, tablet, and other digital devices.

Chapter 2

LITERATURE SURVEY

2.1 Existing problem:

In this digital age, many companies are working to reduce the amount of paper driving the business. A natural extension of this is to tackle handwritten documents and forms that are still needed today. Many forms are still being sent out via post where electronic methods are practical. There are a significant number of people who do not have access to a personal computer, the internet, websites and on-line portals and there are some cases where email addresses are not known or don't exist. In these cases accurate handwriting recognition can still be a problem for many businesses.

2.2 References:

1. 12 June 2020 Improved Handwritten Digit Recognition Using Convolutional Neural Networks (CNN), Savita Ahlawat,Amit Choudhary, Anand Nayyar,Saurabh Singh,and Byungun Yoon.

2. 4 July 2020 Handwritten Digit Recognition Using Various Machine Learning Algorithms and Models,Pranit Patil and Bhupinder Kaur.

3. 2020 Handwritten Digit Recognition Using Computer Vision,Ashish Shekhar and Ajay Kaushik

4. 6 June 2019 Handwritten Digit Recognition using CNN, Vijayalaxmi R Rudraswamimath and Bhavanishankar K.

5. 2019 Recognition of Handwritten Digit using Convolutional Neural Network (CNN), Md. Anwar Hossain & Md. Mohon Ali.

6. 31 August 2019 An efficient and improved scheme for handwritten digit recognition based on convolutional neural network,Saqib Ali, Zeeshan Shaukat, Muhammad Azeem, Zareen Sakhawat, Tariq Mahmood & Khalil ur Rehman.

7. 2018 Handwritten Digit Recognition using Machine Learning Algorithms, S M Shamim, Mohammad Badrul Alam Miah, Angona Sarker, Masud Rana & Abdullah Al Jobair.

8. 2018 A Review of Various Handwriting Recognition Methods,Salma Shofia Rosyda and Tito Waluyo Purboyo.

9. April, 2013 [9] Improving Offline Handwritten Digit Recognition Using Concavity-Based Features, M. Karic and G. Martinovic.

10. 2011 A Statistical Approach for Latin Handwritten Digit Recognition, Ihab Zaqout.

2.3 Problem Statement Definition:

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 thisproblem which uses the image of a digit and recognizes the digit present in the image. Convolutional Neural Network model created using flask over the MNIST dataset to recognize handwritten digits .

Chapter 3

IDEATION AND PROPOSED SOLUTION

3.1 Empathy Map Canvas :

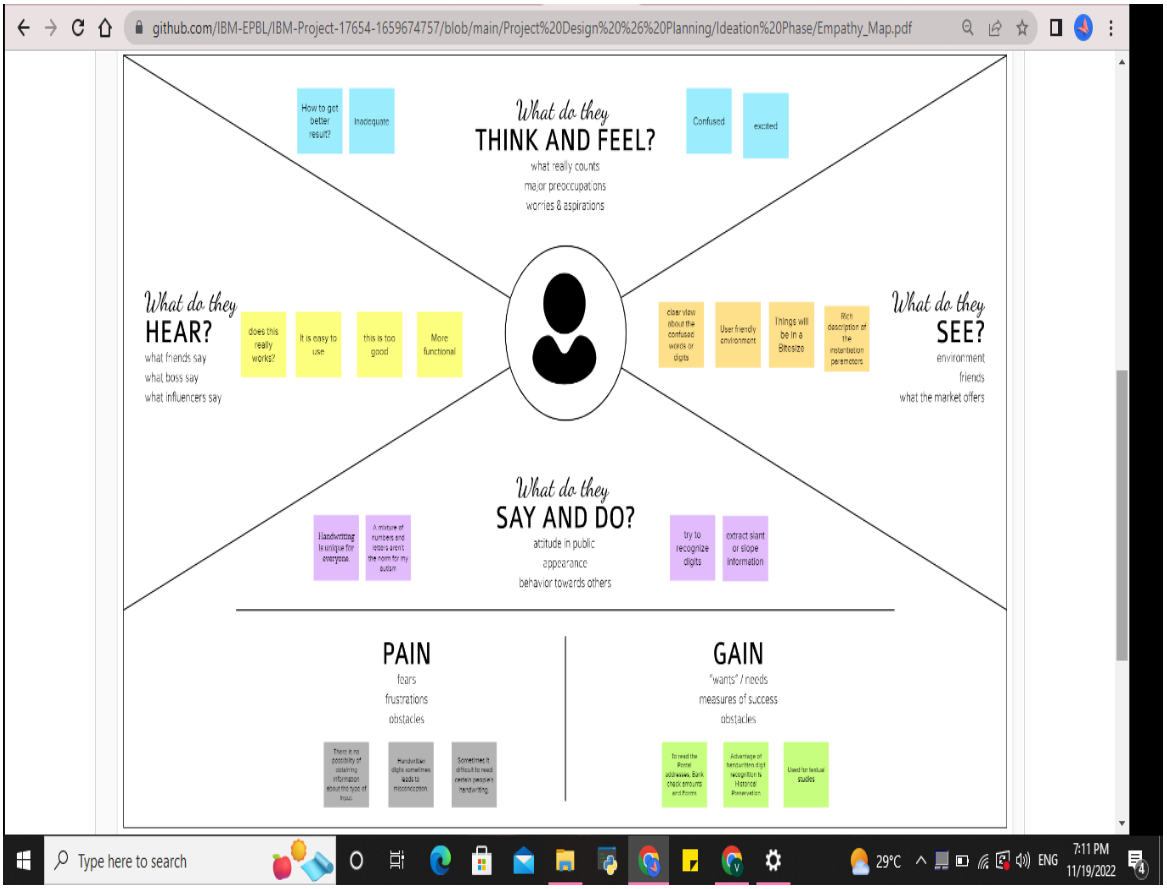
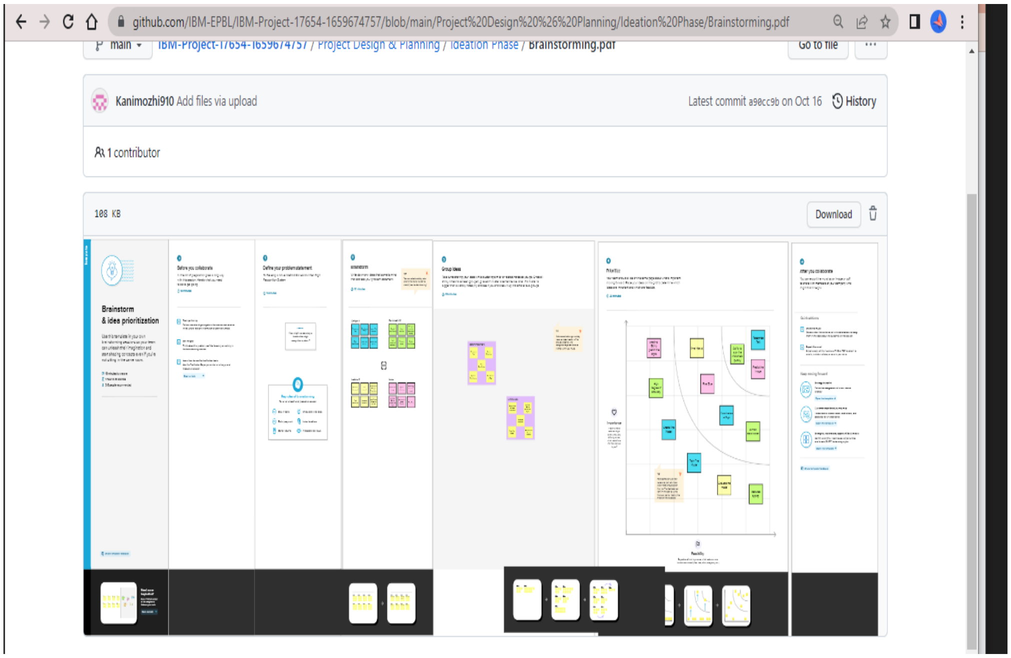

github.com/IBM-EPBL/IBM-Project-17654-1659674757/blob/main/Project%20Design%20%26%20Planning/Ideation%20Phase/Empathy_Map.pdf
What do they
THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
How to got better result?
Inadequate
Confused
excited
What do they
HEAR?
what friends say
what boss say
what influencers say
does this really works?
It is easy to use
this is too good
More functional
What do they
SEE?
environment
friends
what the market offers
clear view about the confused words or digits
User friendly environment
Things will be in a Bitesize
Rich description of the instantiation paremotors
What do they
SAY AND DO?
attitude in public
appearance
behavior towards others
Handwriting is unique for everyone.
A mixture of numbers and letters aren't the norm for my autism
try to recognize digits
extract slant or slope information
PAIN
fears
frustrations
obstacles
There is no possibility of obtaining information about the type of input.
Handwritten digits sometimes leads to misconception.
Sometimes it difficult to read certain people's handwriting.
GAIN
"wants" / needs
measures of success
obstacles
To read the Postal addresses, Bank check amounts and Forms
Advantage of handwritten digit recognition is Historical Preservation
Used for textual studies
Type here to search
29°C
ENG
7:11 PM
11/19/2022

Fig3.1 : Empathy map canvas

3.2 Ideation & Brainstorming:



github.com/IBM-EPBL/IBM-Project-17654-1659674757/blob/main/Project%20Design%20%26%20Planning/Ideation%20Phase/Brainstorming.pdf

main ▾ IBM-Project-17654-1659674757 / Project Design & Planning / Ideation Phase / Brainstorming.pdf

Go to file

Kanimozhi910 Add files via upload

Latest commit a9ecc9b on Oct 16

History

1 contributor

108 KB

Download

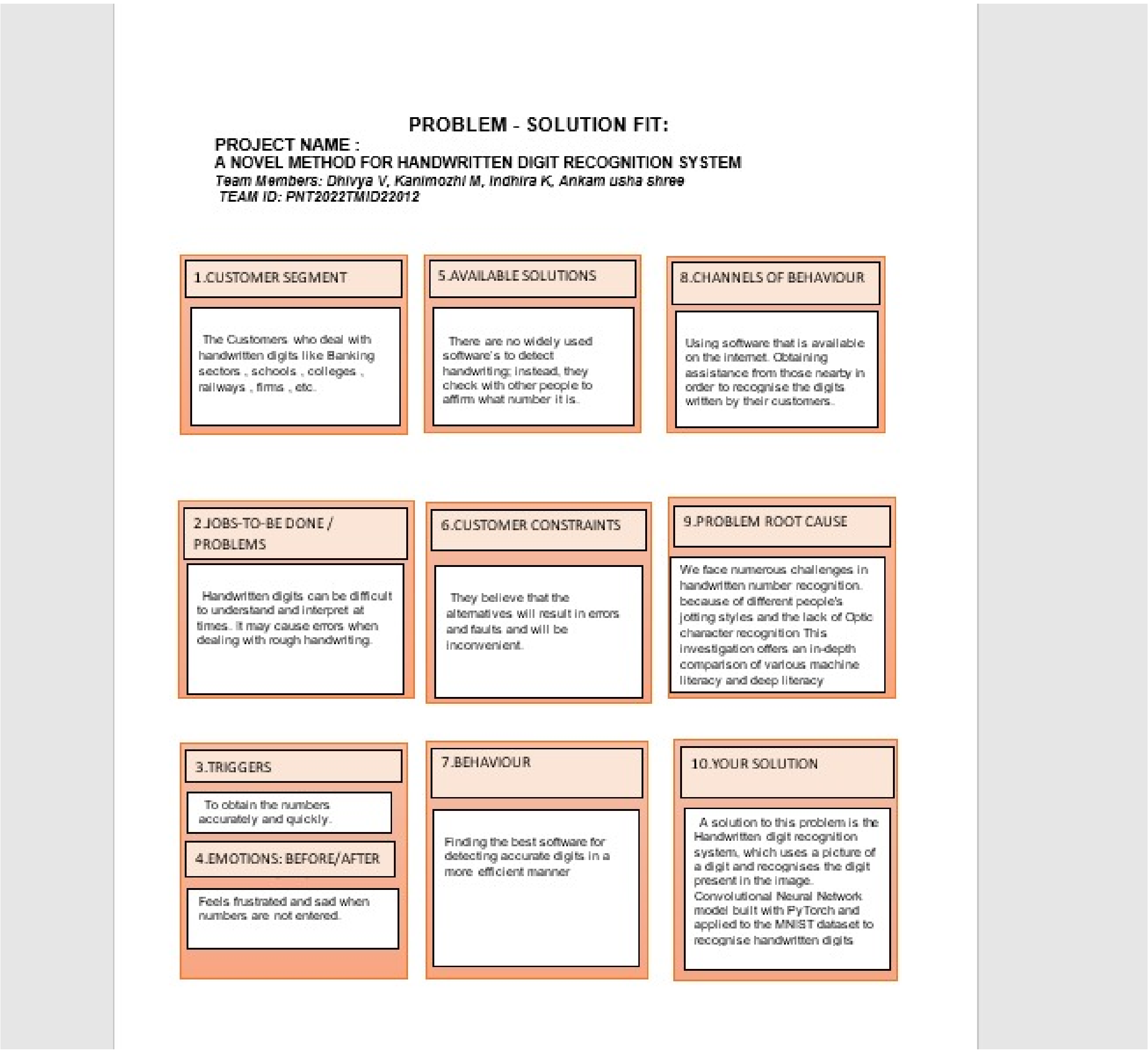
Fig .3.2: **Ideation And Brainstorming**

3.3 Proposed Solution :

S.NO	PARAMETERS	DESCRIPTION
1	Problem Statement (Problem to be solved)	Physically composed digits are of an alternate size, thickness, position and bearing. The machine has a troublesome obligation on the grounds that written by hand digits are not perfect and can be created with various flavours. The Handwritten digit recognition is the capacity of PC applications to perceive the human transcribed digits. The solution for this issue is written by hand digit acknowledgment, which utilizes a picture of a digit and distinguishes the digit addressed in the picture.
2	Idea / Solution description	It is the capacity of a PC to fete the mortal manually written whole numbers from various sources like pictures, papers, contact guards. It permits client to decipher every one of those mark furthermore, notes into electronic words in a text report organization and this information just expects far less actual space than the capacity of the actual duplicates.
3	Novelty / Uniqueness	• To enhance the accuracy, we provided more datasets. • The uniqueness and combination in the synthesis styles of different people furthermore impact the model and presence of the digits.

4	Social Impact / Customer Satisfaction	• The really friendly effect of this work is to guarantee successful and solid methodologies for acknowledgment of transcribed digits and make banking activities simpler and mistake free. • Clients will feel calm, as it is simple and advantageous to utilize. • As the precision is acceptable, this can have numerous applications.
5	Business Model (Revenue Model)	This novel method for Handwritten Digit Recognition System can be approached by many industries which needs this application including, programmed bank checks, postal locations and tax documents and so on. Humans recognizing the handwritten digits with their naked eye can be difficult at times as it of different sizes, thickness, direction and ca also lead to making errors due to these factors. This is when our proposed solutions comes into help. We provide different data sets which helps in recognition with accuracy, so that human making errors can be avoided respectively.
6	Scalability of the Solution	• Ability to recognise digits in more noisy environments. • Financial and other business organizations such as banks are facing issues in Recognizing written digits such as in cheques etc. • Our proposed solution is scalable as it is dynamic and also trained using AI and deep learning Models.

3.4Problem Solution Fit :



PROBLEM - SOLUTION FIT:

PROJECT NAME :
A NOVEL METHOD FOR HANDWRITTEN DIGIT RECOGNITION SYSTEM
Team Members: Dhivya V, Kanimozhi M, Indhira K, Ankam usha shree
TEAM ID: PNT2022TMID22012

1.CUSTOMER SEGMENT

The Customers who deal with handwritten digits like Banking sectors , schools , colleges , railways , firms , etc.

2.JOBS-TO-BE DONE / PROBLEMS

Handwritten digits can be difficult to understand and interpret at times. It may cause errors when dealing with rough handwriting.

3.TRIGGERS

To obtain the numbers accurately and quickly.

4.EMOTIONS: BEFORE/AFTER

Feels frustrated and sad when numbers are not entered.

5.AVAILABLE SOLUTIONS

There are no widely used software's to detect handwriting; instead, they check with other people to affirm what number it is.

6.CUSTOMER CONSTRAINTS

They believe that the alternatives will result in errors and faults and will be inconvenient.

7.BEHAVIOUR

Finding the best software for detecting accurate digits in a more efficient manner

8.CHANNELS OF BEHAVIOUR

Using software that is available on the internet. Obtaining assistance from those nearby in order to recognise the digits written by their customers.

9.PROBLEM ROOT CAUSE

We face numerous challenges in handwritten number recognition. because of different people's jotting styles and the lack of Optic character recognition This investigation offers an in-depth comparison of various machine literacy and deep literacy

10.YOUR SOLUTION

A solution to this problem is the Handwritten digit recognition system, which uses a picture of a digit and recognises the digit present in the image. Convolutional Neural Network model built with PyTorch and applied to the MNIST dataset to recognise handwritten digits

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 Functional requirement:

Following are the functional requirements of the proposed solution.

FR. NO	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
1	User Input	GUI allows the user to input image by browsing the device storage
2	Model	The MNIST dataset should be trained using CNN to create a trained model
3	Prediction	The trained model has to be tested by using the test data provided by MNIST and the accuracy of the model should be above 90%
4	Evaluation	Ensure that the output produced by the model is correct

4.2 Non-Functional requirements :

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

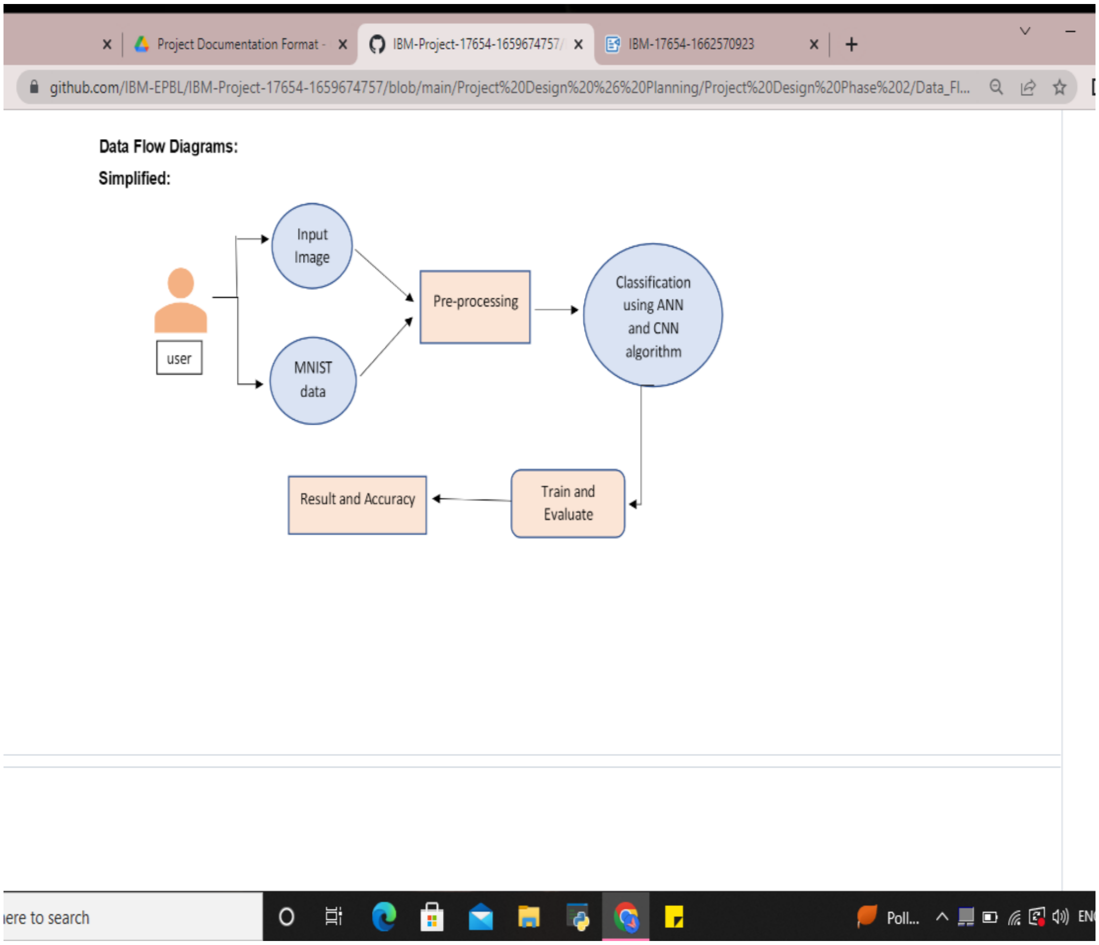
FR. NO	Non-Functional Requirement	Description
1	Usability	Hand written digit recognition is one of the major important issues in pattern recognition application.

		Some of the applications for digit recognition include data entry forms, Bank check processing etc.
2	Security	The application of hand written digit recognition can be used in the banking sector where it can be used to maintain the security pin numbers safely.it can be also used for blind-people by using sound output.
3	Reliability	Reliability indicates the probability that the system will perform its intended function for a large period of sufficient time and also it will operate in a secured environment without any failures.
4	Performance	The standard implementations of neural networks achieve an accuracy of approximately(98-99)

CHAPTER 5

PROJECT DESIGN

5.1 Data Flow Diagrams:



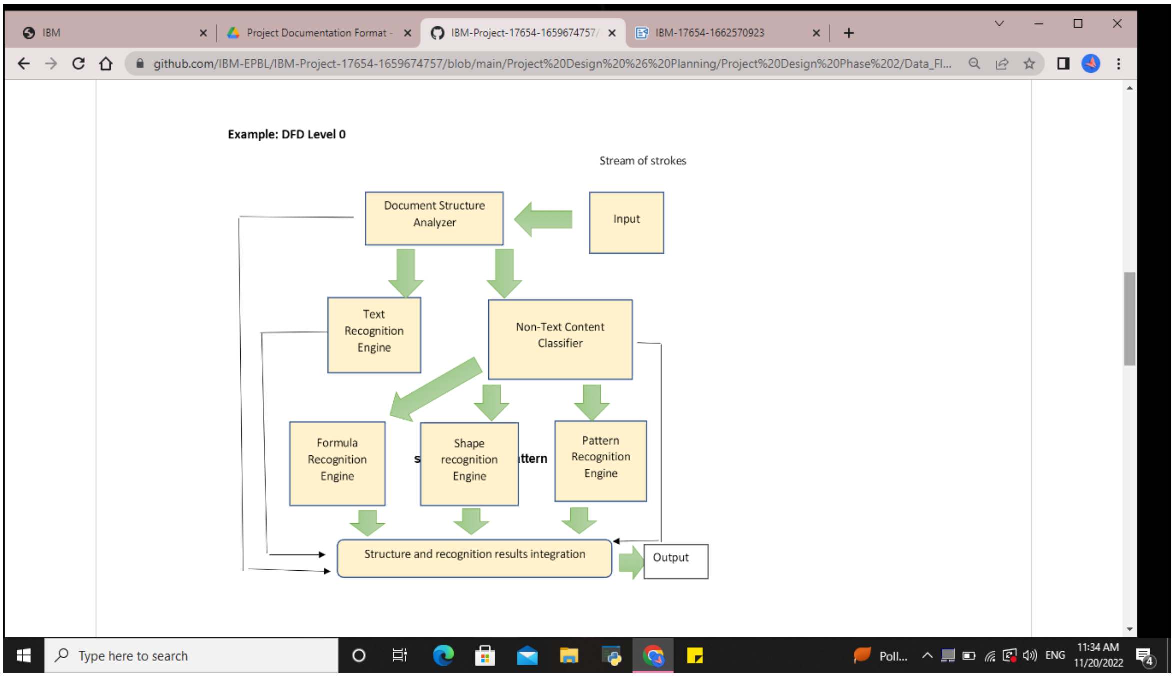
Data Flow Diagrams:

Simplified:

Project Documentation Format -
IBM-Project-17654-1659674757/
IBM-17654-1662570923
github.com/IBM-EPBL/IBM-Project-17654-1659674757/blob/main/Project%20Design%20%26%20Planning/Project%20Design%20Phase%202/Data_Fl...
user
Input Image
MNIST data
Pre-processing
Classification using ANN and CNN algorithm
Train and Evaluate
Result and Accuracy
ere to search
Poll...
EN

Fig 5.1: Data flow diagrams

Industry Standard:


IBM
Project Documentation Format -
IBM-Project-17654-1659674757/
IBM-17654-1662570923
github.com/IBM-EPBL/IBM-Project-17654-1659674757/blob/main/Project%20Design%20%26%20Planning/Project%20Design%20Phase%202/Data_Fl...

Example: DFD Level 0

Stream of strokes
Input
Document Structure Analyzer
Text Recognition Engine
Non-Text Content Classifier
Formula Recognition Engine
Shape recognition Engine
Pattern Recognition Engine
Structure and recognition results integration
Output
Type here to search
Poll...
ENG
11:34 AM
11/20/2022

Fig 5.2-Industry Standard

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

Example: A Novel Method For Handwritten And Digit Recognition

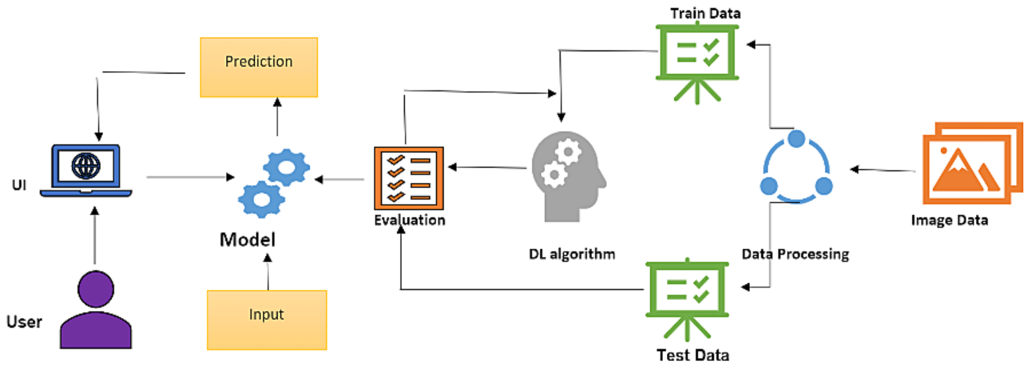
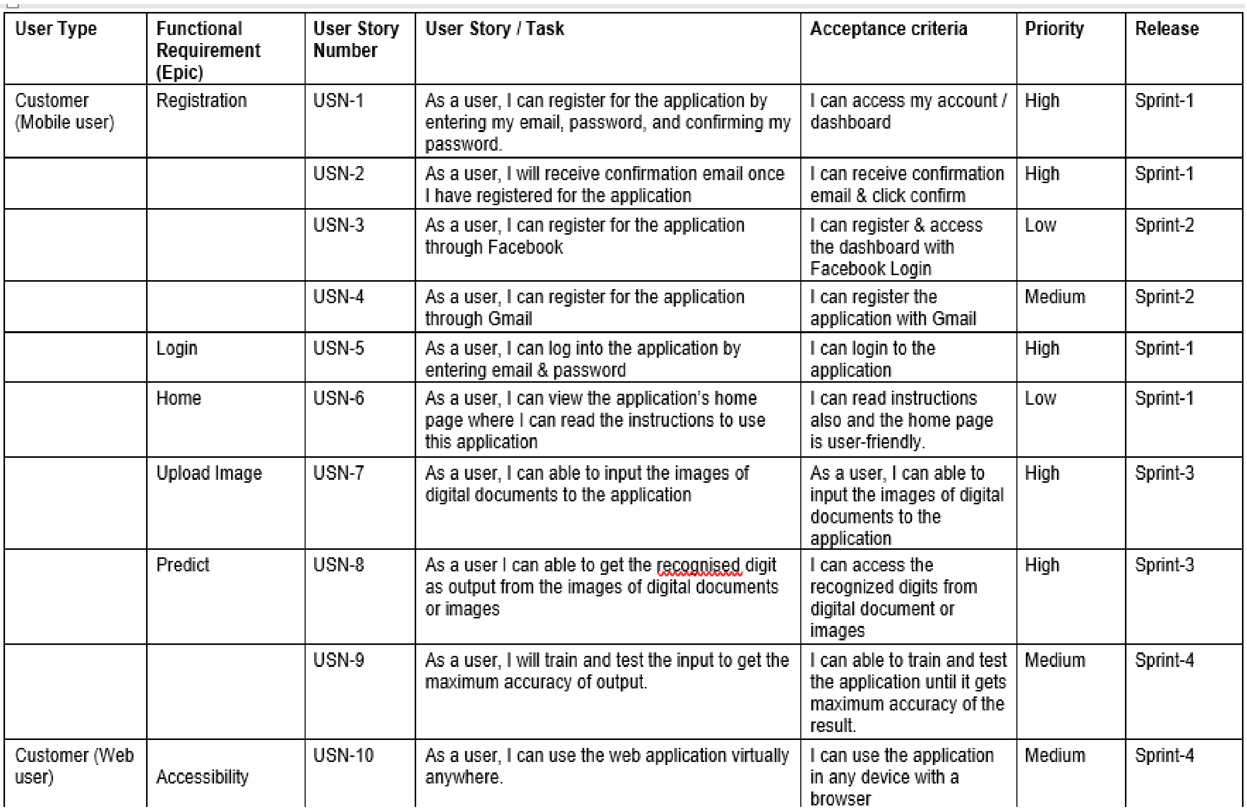


Fig 5.3-Technical Architecture

5.4 User Stories :

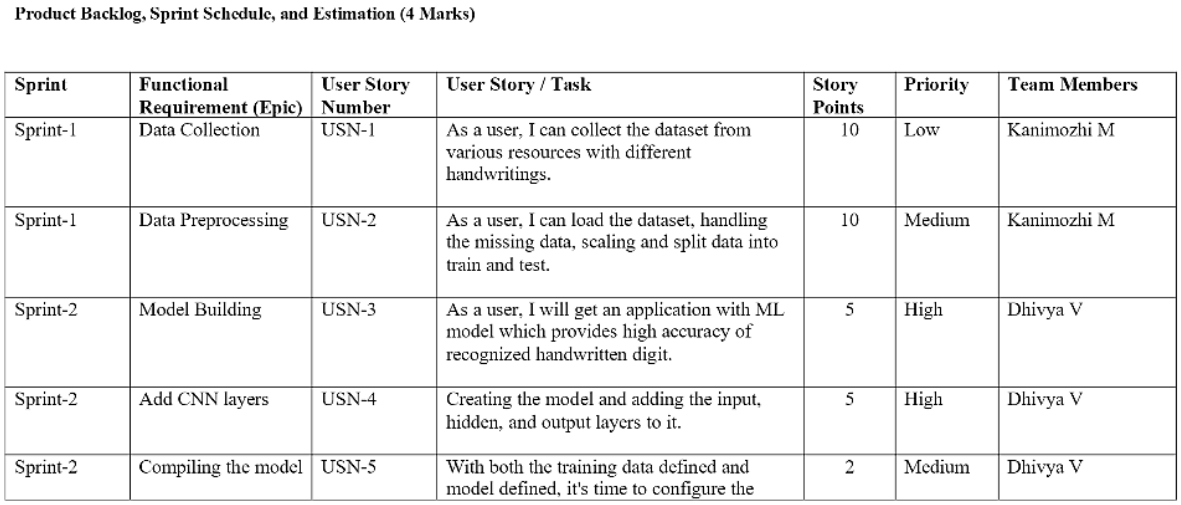


User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	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

CHAPTER 6

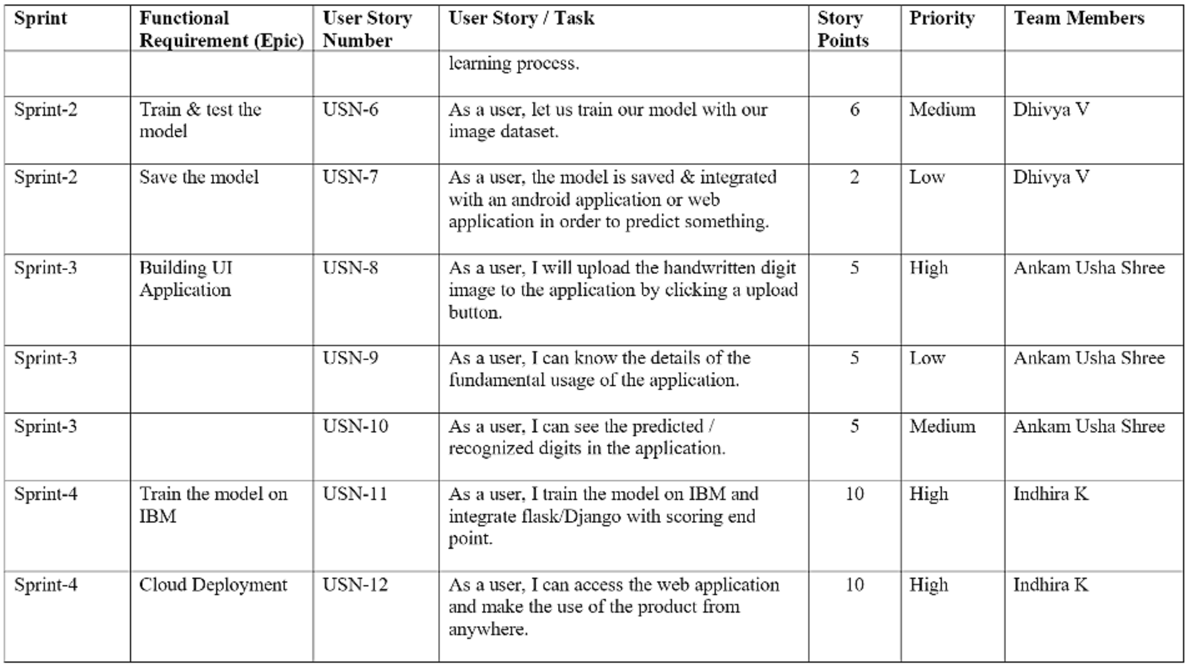
PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation:



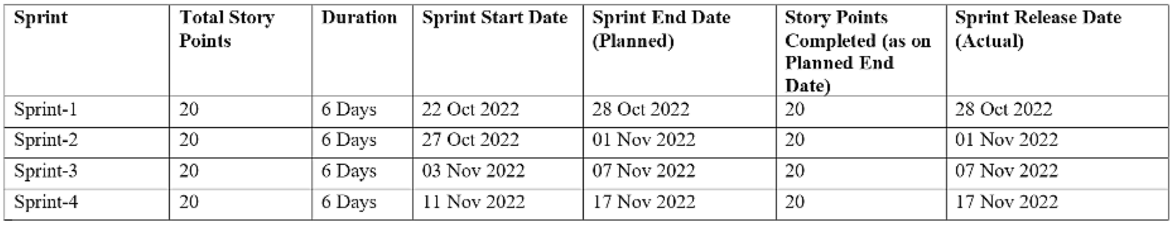
Product Backlog, Sprint Schedule, and Estimation (4 Marks)

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	Kanimozhi M
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	Kanimozhi 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	Dhivya V
Sprint-2	Add CNN layers	USN-4	Creating the model and adding the input, hidden, and output layers to it.	5	High	Dhivya V
Sprint-2	Compiling the model	USN-5	With both the training data defined and model defined, it's time to configure the	2	Medium	Dhivya V



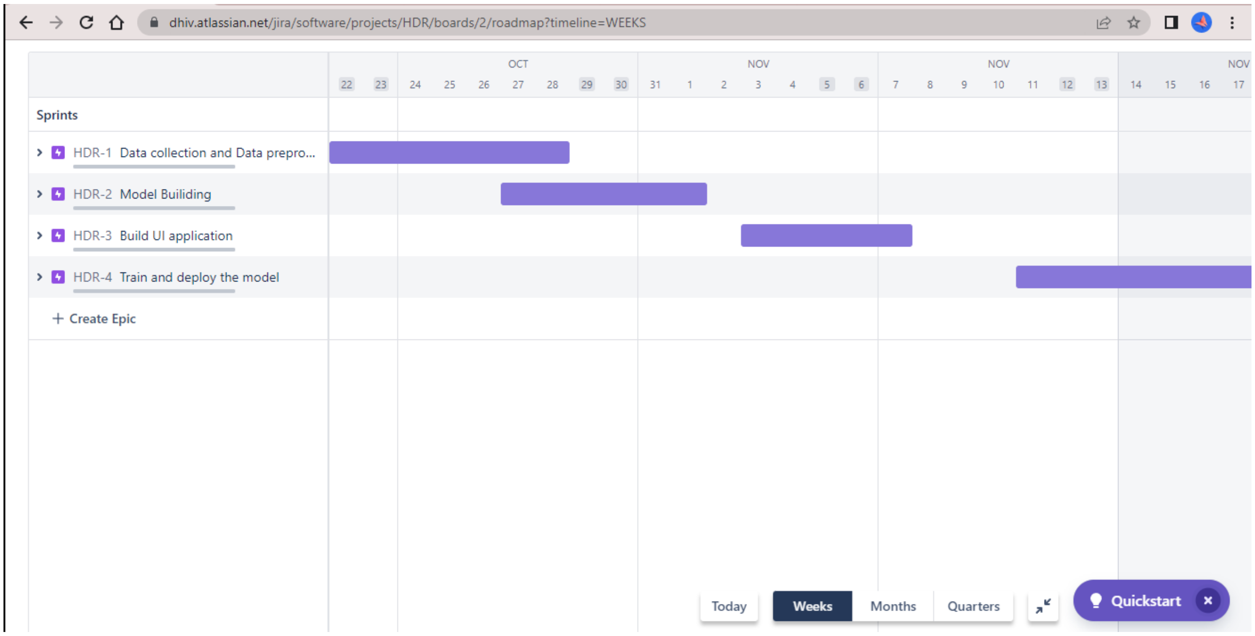
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
			learning process.			
Sprint-2	Train & test the model	USN-6	As a user, let us train our model with our image dataset.	6	Medium	Dhivya V
Sprint-2	Save the model	USN-7	As a user, the model is saved & integrated with an android application or web application in order to predict something.	2	Low	Dhivya V
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	Ankam Usha Shree
Sprint-3		USN-9	As a user, I can know the details of the fundamental usage of the application.	5	Low	Ankam Usha Shree
Sprint-3		USN-10	As a user, I can see the predicted / recognized digits in the application.	5	Medium	Ankam Usha Shree
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	Indhira K
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	Indhira K

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	22 Oct 2022	28 Oct 2022	20	28 Oct 2022
Sprint-2	20	6 Days	27 Oct 2022	01 Nov 2022	20	01 Nov 2022
Sprint-3	20	6 Days	03 Nov 2022	07 Nov 2022	20	07 Nov 2022
Sprint-4	20	6 Days	11 Nov 2022	17 Nov 2022	20	17 Nov 2022

6.3 Reports from JIRA :



dhiv.atlassian.net/jira/software/projects/HDR/boards/2/roadmap?timeline=WEEKS

Sprints

- HDR-1 Data collection and Data prepro...
- HDR-2 Model Builiding
- HDR-3 Build UI application
- HDR-4 Train and deploy the model

+ Create Epic

CHAPTER 7

CODING & SOLUTIONING

7.1 Feature 1 :

```
   <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-
UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>
  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>
  <script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs@latest"></script>

</head>

<script>
  function preview() {
    frame.src=URL.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
  <div id="team_id">TEAM ID : PNT2022TMID22012</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="buttons_div">
<form action = "/upload">
            <button type="submit" class="btn btn-white"
id="predict_button">Predict</button></form>
            <button type="button" class="btn btn-white" id="clear_button">&nbsp Clear
&nbsp</button>
        </form>
        </div>
 </section>
</body>
</html>
```

7.2 Feature 2:

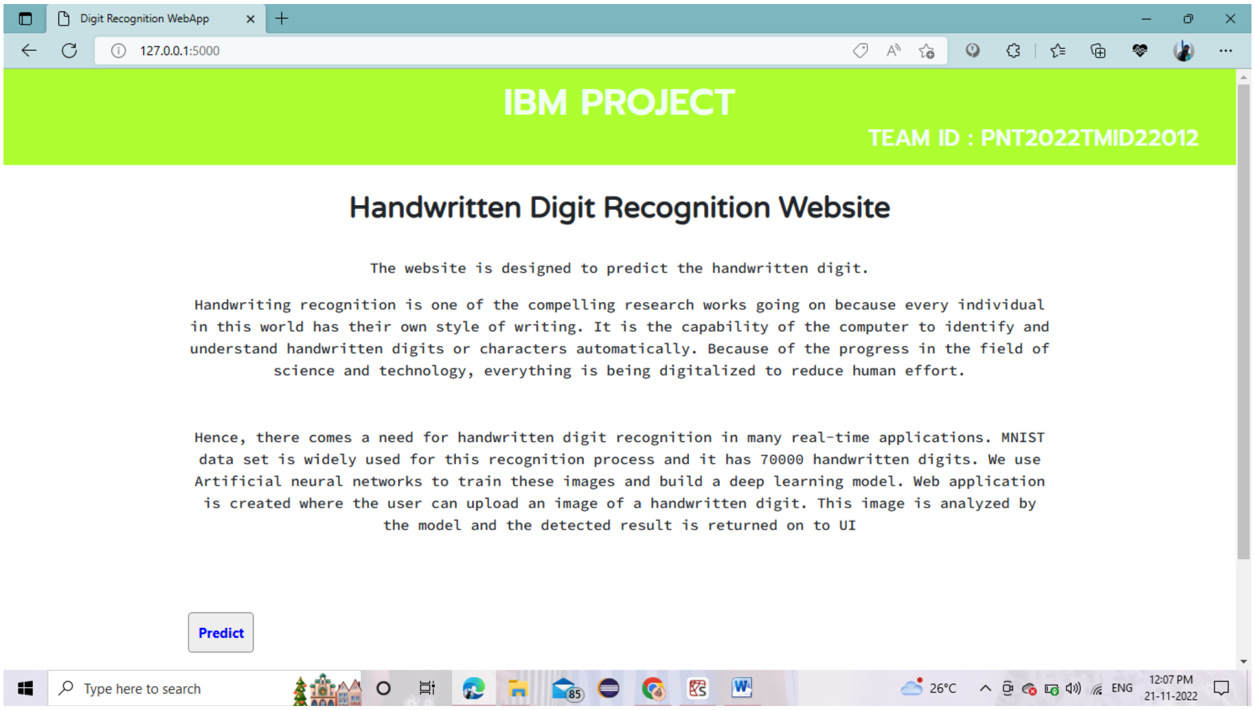
```
<!DOCTYPE html>
<html lang="en">
<head>
<link rel="shortcut icon" href="data:image/x-icon;," type="image/x-icon">
   <meta charset="UTF-8">
   <title>Prediction</title>
</head>

<style>
   body{
    background-image: url('static/images/image1.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>
 <form action = "/upload" method = "POST"
          enctype = "multipart/form-data">
         <label>Select a image:</label>
         <input id="image" type="file" name="file" accept="image/png, image/jpeg"
onchange="preview()"><br><br>
  <input type = "submit" name="Recognize">
</form>
    <div id="rectangle">
          <h1 id="ans">Predicted Number : {{msg}}</h1>
    </div>
</body>
</html>
```

RESULT:



IBM PROJECT

TEAM ID : PNT2022TMID22012

Handwritten Digit Recognition Website

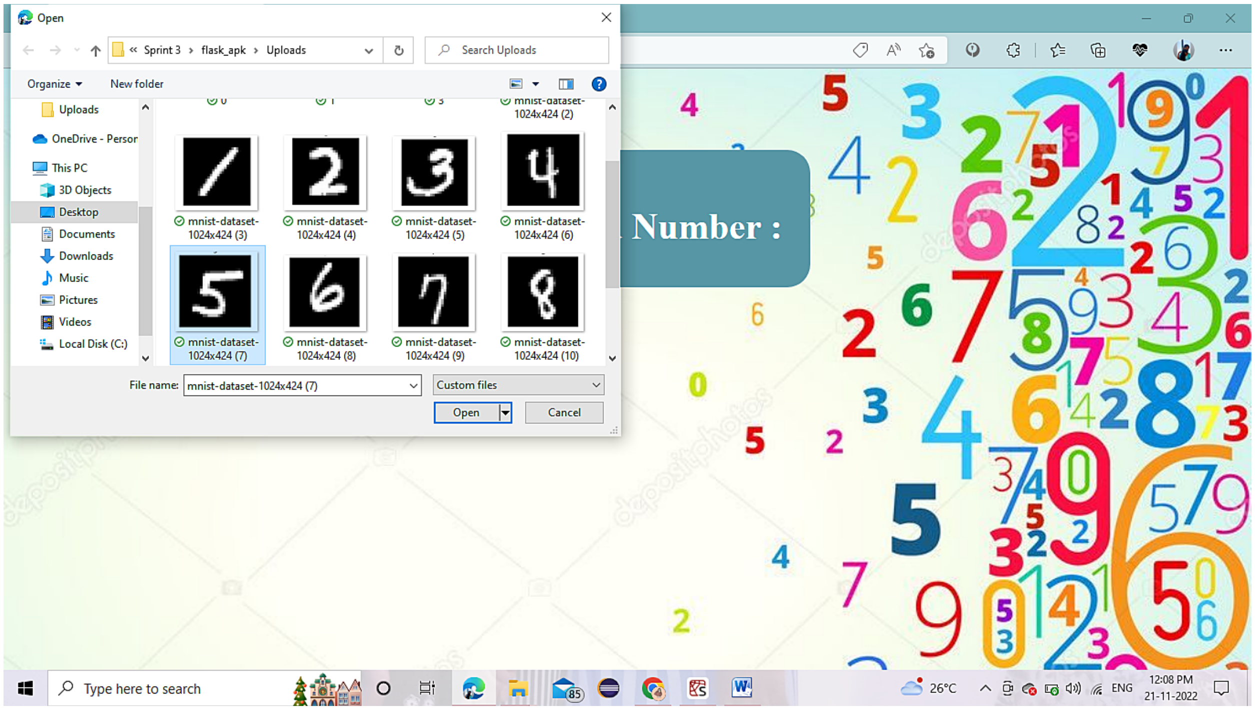
The website is designed to predict the handwritten digit.

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

Predict

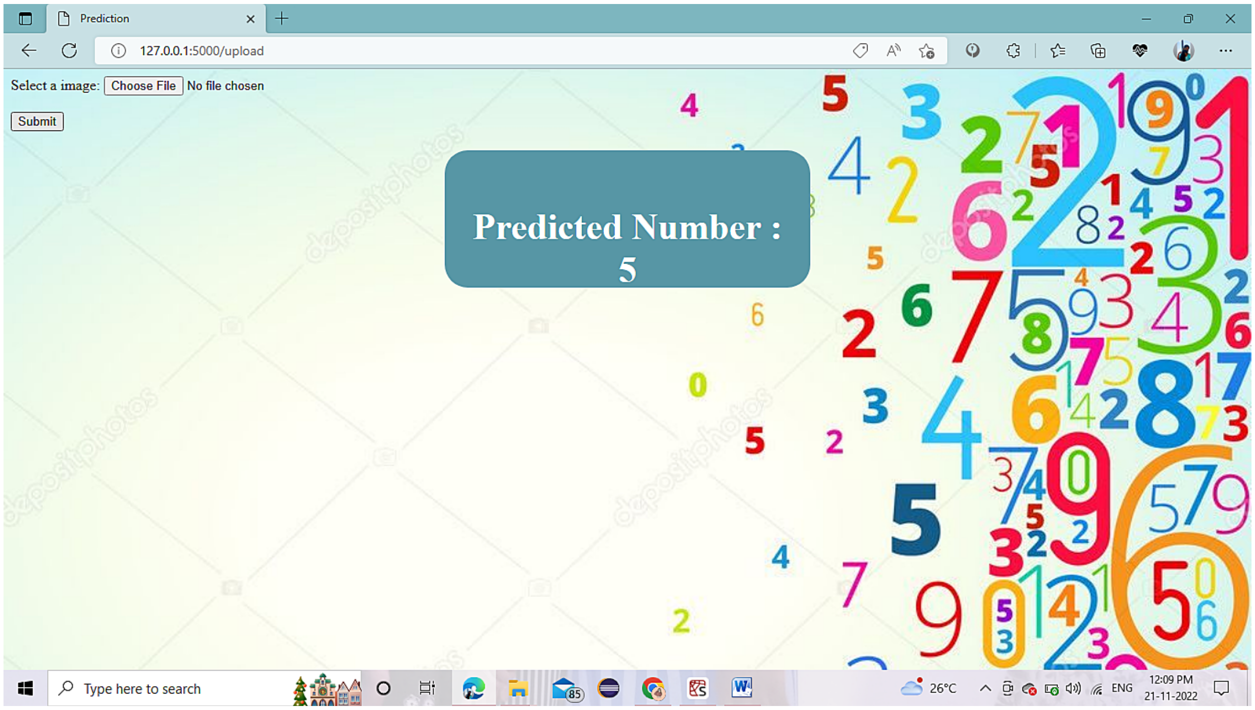
Fig 7.1 :Feature7.1


Open
Sprint 3 › flask_apk › Uploads
Search Uploads
Organize
New folder
Uploads
OneDrive - Person
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
mnist-dataset-1024x424 (2)
mnist-dataset-1024x424 (3)
mnist-dataset-1024x424 (4)
mnist-dataset-1024x424 (5)
mnist-dataset-1024x424 (6)
mnist-dataset-1024x424 (7)
mnist-dataset-1024x424 (8)
mnist-dataset-1024x424 (9)
mnist-dataset-1024x424 (10)
Open
Cancel
Number :
Type here to search
26°C
ENG
12:08 PM
21-11-2022

File name: mnist-dataset-1024x424 (7)

Custom files

Fig 7.2:Feature7.2



127.0.0.1:5000/upload

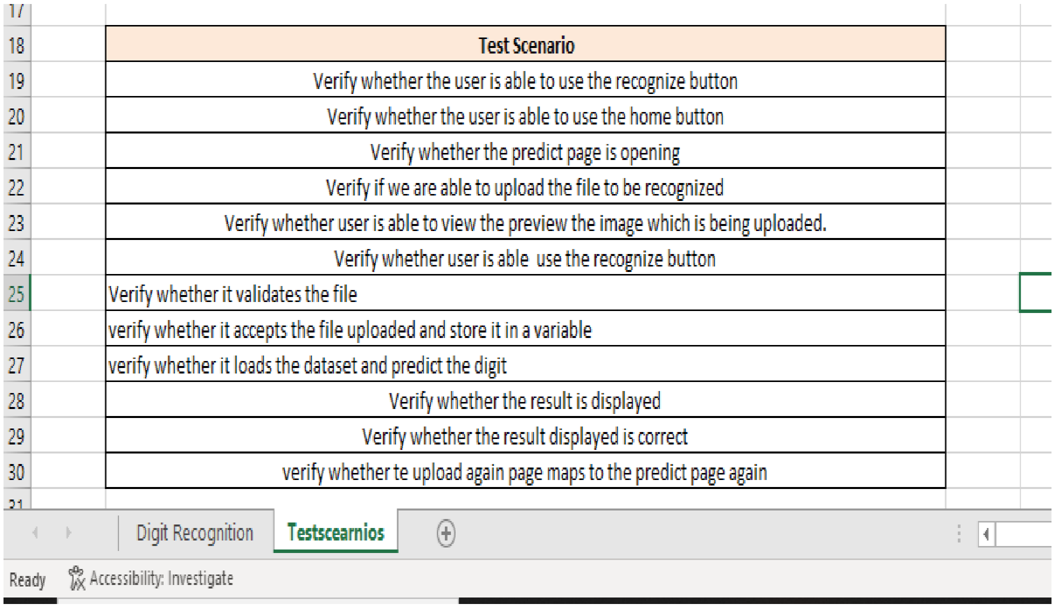
Select a image: Choose File No file chosen

Predicted Number : 5

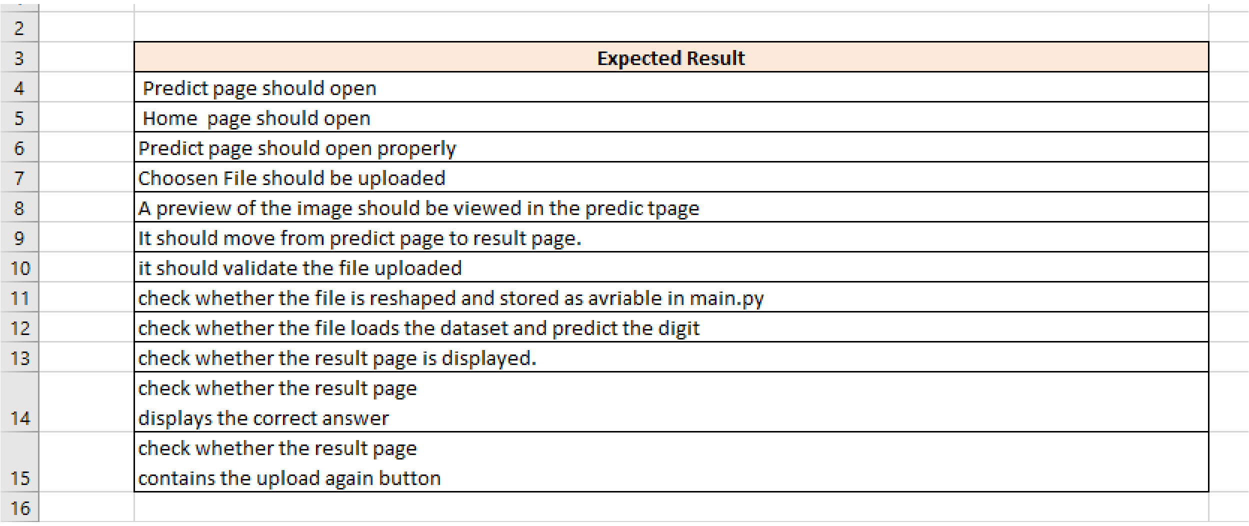
CHAPTER 8

TESTING

8.1 Test Cases :



	Test Scenario
19	Verify whether the user is able to use the recognize button
20	Verify whether the user is able to use the home button
21	Verify whether the predict page is opening
22	Verify if we are able to upload the file to be recognized
23	Verify whether user is able to view the preview the image which is being uploaded.
24	Verify whether user is able use the recognize button
25	Verify whether it validates the file
26	verify whether it accepts the file uploaded and store it in a variable
27	verify whether it loads the dataset and predict the digit
28	Verify whether the result is displayed
29	Verify whether the result displayed is correct
30	verify whether te upload again page maps to the predict page again



	Expected Result
4	Predict page should open
5	Home page should open
6	Predict page should open properly
7	Choosen File should be uploaded
8	A preview of the image should be viewed in the predic tpage
9	It should move from predict page to result page.
10	it should validate the file uploaded
11	check whether the file is reshaped and stored as avriable in main.py
12	check whether the file loads the dataset and predict the digit
13	check whether the result page is displayed.
14	check whether the result page displays the correct answer
15	check whether the result page contains the upload again button

8.2 User Acceptance Testing:

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issuesof the A Novel Method for Handwritten Digit Recognition project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

This reportshows the numberof 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	0	4	2	3	9
Duplicate	0	0	3	0	3
External	0	0	0	1	1
Fixed	0	4	5	4	13
Not Reproduced	0	0	0	0	0
Skipped	0	0	0	1	1
Won't Fix	0	0	0	1	1
Totals	0	8	11	10	26

Test Case Analysis

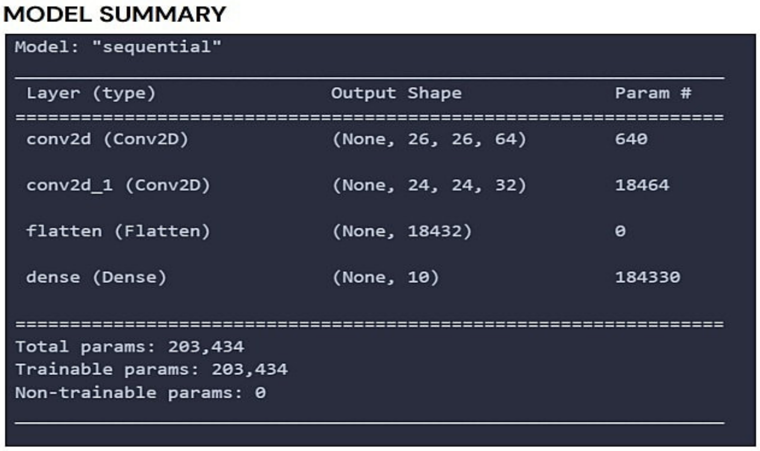
This report shows the numberof test cases that have passed, failed,and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	10	0	0	10
Security	1	0	0	1
Outsource Shipping	2	0	0	2
Exception Reporting	9	0	0	9
Final ReportOutput	4	0	0	4
Version Control	2	0	0	2

CHAPTER 9

RESULTS

9.1 Performance Metrics :

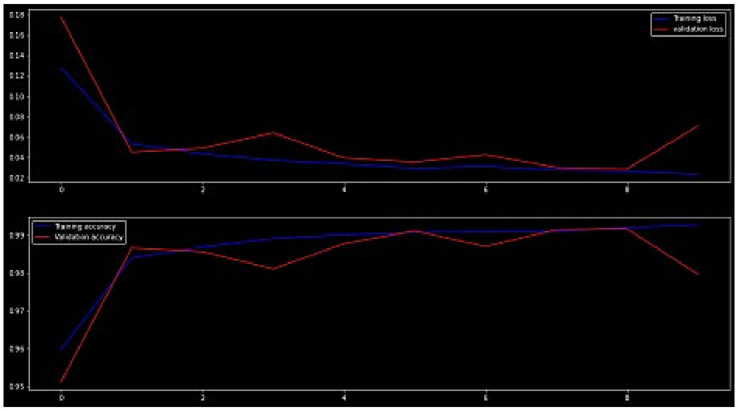


MODEL SUMMARY

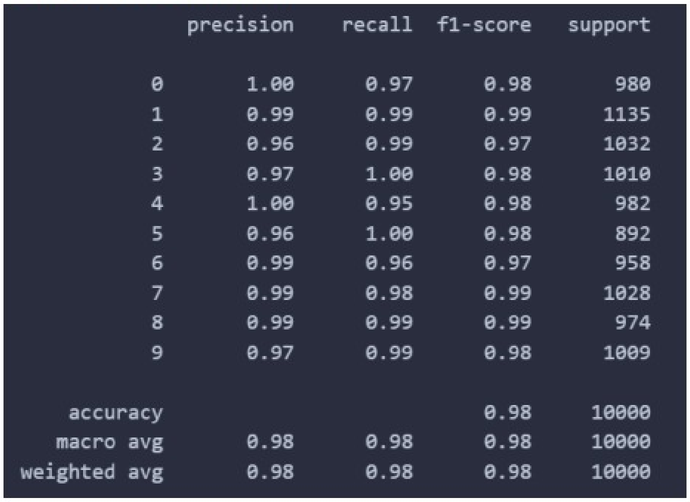
```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 conv2d (Conv2D)             (None, 26, 26, 64)        640       
                                                                 
 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
_________________________________________________________________
```

ACCURACY

CONTENT	VALUE
Training Accuracy	99.14%
Training Loss	2.70%
Validation Accuracy	97.76%
Validation Loss	10.36%



CLASSIFICATION REPORT



```
              precision    recall  f1-score   support

           0       1.00      0.97      0.98       980
           1       0.99      0.99      0.99      1135
           2       0.96      0.99      0.97      1032
           3       0.97      1.00      0.98      1010
           4       1.00      0.95      0.98       982
           5       0.96      1.00      0.98       892
           6       0.99      0.96      0.97       958
           7       0.99      0.98      0.99      1028
           8       0.99      0.99      0.99       974
           9       0.97      0.99      0.98      1009

    accuracy                           0.98     10000
   macro avg       0.98      0.98      0.98     10000
weighted avg       0.98      0.98      0.98     10000
```

CHAPTER 10
ADVANTAGES & DISADVANTAGES

10.1. ADVANTAGES:

1. The system not only produces a classification of the digit but also a rich description of the instantiation parameters which can yield information such as writing style.

2. The generative models can perform recognition driven segmentation.

3.Handwriting forces your brain to mentally engage with the information, improving both literacy and reading comprehension.

4.Digitalization

5.Data Collection.

10.2. DISADVANTAGES:

1. Despite that there are enormous convolutional neural network algorithms proposed for handwritten digit recognition, issues such as recognition accuracy.

2. Get alternative, less likely predictions when available.

3.Anyway Higher processor is required.

4.High cost

5.Time consuming

6.computation time still require for further improvement.

CHAPTER 11

CONCLUSION

We successfully built a Python artificial intelligence project for handwritten digit recognition. We built and trained a Convolutional Neural Network (CNN) that excels at image classification. Deep Learning Methods for Handwritten Digit Recognition have been developed. The most commonly used Machine learning algorithms CNN were trained and tested on the same data to provide a comparison of classifiers. These deep learning approaches can achieve a high level of accuracy. Unlike other research methodologies, this one is concerned with determining which classifier performs best by increasing accuracy. A CNN model that uses Keras as the backend as well as TensorFlow as the software can achieve 98% accuracy.

CHAPTER 12

FUTURE SCOPE

As a result of the above differentiation and future development, we can achievehigh-level functioning applicationsthat can be used in classified or government agencies as well as for the general public.

We can add fuzzification with Back propagation algoritm.This algorithm can be used to recognize multiple digits at a time and also to recognize characters

CHAPTER 13

APPENDIX

13.1 Source Code :

main.py

```
from flask import Flask,render_template,request

from PIL import Image

import numpy as np

from tensorflow.keras.models import load_model

import tensorflow as tf

app = Flask(__name__)

@app.route('/')

def upload_file():

    return render_template('index.html')


@app.route('/about')

def upload_file1():

    return render_template('predict.html')


@app.route('/upload')

def upload_file2():
    return render_template('predict.html')


@app.route('/upload', methods = ['POST'])
def upload_image_file():
    if request.method=='POST':
        img = Image.open(request.files['file'].stream).convert("L")
        img = img.resize((28,28))
        img2arr = np.array(img)
        img2arr = img2arr.reshape(1,28,28,1)
        model = load_model('mnistCNN.h5')
        #y_pred = model.predict_classes(img2arr)
        y_pred = np.argmax(model.predict(img2arr), axis=1)
        print(y_pred)
        if(y_pred == 0):
            return render_template("predict.html",msg="0")
        elif(y_pred == 1):
            #return render_template("1.html",shoechase = str(y_pred))
            return render_template("predict.html",msg="1")
        elif(y_pred == 2):
            #return render_template("2.html",shoechase = str(y_pred))
            return render_template("predict.html",msg="2")
        elif(y_pred == 3):
            #return render_template("3.html",shoechase = str(y_pred))
            return render_template("predict.html",msg="3")
        elif(y_pred == 4):
            #return render_template("4.html",shoechase = str(y_pred))
            return render_template("predict.html",msg="4")
        elif(y_pred == 5):
            #return render_template("5.html",shoechase = str(y_pred))
            return render_template("predict.html",msg="5")
        elif(y_pred == 6):
            #return render_template("6.html",shoechase = str(y_pred))
            return render_template("predict.html",msg="6")
        elif(y_pred == 7):
            #return render_template("7.html",shoechase = str(y_pred))
            return render_template("predict.html",msg="7")
        elif(y_pred == 8):
            #return render_template("8.html",shoechase = str(y_pred))
            return render_template("predict.html",msg="8")
        else:
            #return render_template("9.html",shoechase = str(y_pred))
            return render_template("predict.html",msg="9")
    else:
        return None


if __name__=='__main__':
    app.run()
```

13.2 Github And Demo Link:

Youtube Demo Link:

https://youtu.be/MmPPID4ZJQ

Github Link:

https://github.com/IBM-EPBL/IBM-Project-17654-1659674757

Source Code Link:

https://github.com/IBM-EPBL/IBM-Project-17654-1659674757/tree/main/Final%20Deliverables41