PROJECT REPORT

A NOVEL METHOD FOR HANDWRITTEN DIGIT RECOGNITION

Submitted by

PNT2022TMID51312

VISHAL V U	**960319104039**
BIJU PATSON	**960319104010**
VIVEK A	**960319104040**
LIGIN ANTO	**960319104023**
ANANTHAN M	**960319104005**

TABLE OF CONTENTS

1. **INTRODUCTION** **4**
   1.1 PROJECT OVERVIEW 4
   1.2 PURPOSE 4

2. **LITERATURE SURVEY** **5**
   2.1 EXISTING PROBLEM 5
   2.2 REFERENCES 5
   2.3 PROBLEM STATEMENT DEFINITION 9

3. **IDEATION AND PROPOSED SOLUTION** **10**
   3.1 EMPATHY MAP 10
   3.2 IDEATION & BRAINSTORMING 11
   3.3 PROPOSED SOLUTION 12
   3.4 PROBLEM SOLUTION FIT 13

4. **REQUIREMENT ANALYSIS** **19**
   4.1 FUNCTIONAL REQUIREMENTS 19
   4.2 NON-FUNCTIONAL REQUIREMENTS 20

5. **PROJECT DESIGN** **21**
   5.1 DATA FLOW DIAGRAM 21
   5.2 SOLUTION & TECHNICAL ARCHITECTURE 22
   5.3 USER STORIES 24

6. **PROJECT PLANNING** **25**
   6.1 SPRINT PLANNING AND ESTIMATION 25
   6.2 SPRINT DELIVERY SCHEDULE 26
   6.3 REPORT FROM JIRA 27

7. **CODING** **28**
   7.1 PROGRAM 28

8. **TESTING** **30**
   8.1 TEST CASES 30
   8.2 USER ACCEPTANCE TESTING 34

9. **RESULTS** **36**

9.1 OUTPUT 36

10 ADVANTAGES & DISADVANTAGES 37

11 CONCLUSION 38

12 FUTURE SCOPE

13 APPENDIX 28

SOURCE CODE 28

GITHUB LINK 37

CHAPTER 1

INTRODUCTION

1.1.PROJECT OVERVIEW

Machine learning and deep learning play an important role in computer technology and artificial intelligence. With the use of deep learning and machine learning, human effort can be reduced in recognizing, learning, predictions and in many more areas.

Handwritten Digit Recognition is the ability of computer systems to recognise handwritten digits from various sources, such as images, documents, and so on. This project aims to let users take advantage of machine learning to reduce manual tasks in recognizing digits

1.2. PURPOSE

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.

Digit recognition systems are capable of recognizing the digits from different sources like emails, bank cheque, papers, images, etc. and in different real-world scenarios for online handwriting recognition on computer tablets or system, recognize number plates of vehicles, processing bank cheque amounts, numeric entries in forms filled up by hand (tax forms) and so on.

CHAPTER 2

LITERATURE SURVEY

2.1. EXISTING PROBLEM

The issue is that there's a wide range of handwriting – good and bad. This makes it tricky for programmers provide enough examples of how every character might look. Sometimes, characters look very similar, making hard for a computer to recognize accurately.

The fundamental problem with handwritten digit recognition is that handwritten digits do not always have same size, width, orientation, and margins since they vary from person to person. Additionally, there would be issues with identifying the numbers because of similarities between numerals like 1 and 7, 5 and 6, 3 and 8, 2 a 5, 2 and 7, etc. Finally, the individuality and variation of each individual's handwriting influence the structure appearance of the digits.

2.2. REFERENCES

Recognition of Handwritten Digit using Convolutional Neural Network in Python with TensorFlow Comparison of Performance for Various Hidden Layers (2019)

F. Siddique, S. Sakib and M. A. B. Siddique

The increase of Artificial Neural Network (ANN), deep learning has brought a dramatic twist in the field of machine learning by making it more artificially intelligent. Deep learning is remarkably used in vast ranges of fields because of its diverse range of applications such as surveillance, health, medicine, sports, robotics, drones, etc. In deep learning, Convolutional Neural Network (CNN) is at the center of spectacular advances that mixes Artificial Neural Network (ANN) and up to date deep learning strategies. It has been used broadly in pattern recognition, sentence classification, speech recognition, face recognition, text categorization, document analysis, scene, and handwritten digit recognition. The goal of this paper is to observe the variation of accuracies of CNN to classify handwritten digits using various numbers of hidden layers and epochs and to make the comparison between the accuracies. For this performance evaluation of CNN, we performed our experiment using Modified National Institute of Standards and Technology (MNIST) dataset. Further, the network is trained using stochastic gradient descent and the back-propagation algorithm.

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

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

This study uses rectified linear units (ReLU) activation and a convolutional neural network (CNN) that incorporates the Deeplearning4j (DL4J) architecture to recognize handwritten digits. The proposed CNN framework has all the necessary parameters for a high level of MNIST digit classification accuracy. The system's training takes into account the time factor as well. The system is also tested by altering the number of CNN layers for additional accuracy verification. It is important to note that the CNN architecture consists of two convolutional layers, the first with 32 filters and a 5x5 window size and the second with 64 filters and a 7x7 window size. In comparison to earlier proposed systems, the experimental findings show that the proposed CNN architecture for the MNIST dataset demonstrates great performance in terms of time and accuracy. As a result, handwritten numbers are detected with a recognition rate of 99.89% and high precision (99.21%) in a short amount of time.

Handwritten Digit Recognition using Convolutional Neural Network in Python with TensorFlow and Observe the Variation of Accuracies for Various Hidden Layers (2019)

Fathma Siddique, Shadman Sakib, Md. Abu Bakr Siddique

In recent times, with the increase of Artificial Neural Network (ANN), deep learning has brought a dramatic twist in the field of machine learning by making it more Artificial Intelligence (AI). Deep learning is remarkably used in vast ranges of fields because of its diverse range of applications such as surveillance, health, medicine, sports, robotics, drones etc. In deep learning, Convolutional Neural Network (CNN) is at the center of spectacular advances that mixes Artificial Neural Network (ANN) and up to date deep learning strategies. It has been used broadly in pattern recognition, sentence classification, speech recognition, face recognition, text categorization, document analysis, scene, and handwritten digit recognition. The goal of this paper is to observe the variation of accuracies of CNN to classify handwritten digits using various numbers of hidden layers and epochs and to make the comparison between the accuracies. For this performance evaluation of CNN, we performed our experiment using the Modified National Institute of Standards and Technology (MNIST) dataset. Further, the network is trained using stochastic gradient descent and the backpropagation algorithm.

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

Pashine, Samay and Dixit, Ritik and Kushwah, Rishika

In this study, they developed three deep and machine learning-based models for handwritten digit recognition using MNIST datasets. To determine which model was the most accurate, they compared them based on their individual properties.

Support vector machines are among the simplest classifiers, making them faster than other algorithms and providing the highest training accuracy rate in this situation. However, due to their simplicity, SVMs cannot categorize complicated and ambiguous images as accurately as MLP and CNN algorithms can. In their research, they discovered that CNN produced the most precise outcomes for handwritten digit recognition. This led them to the conclusion that CNN is the most effective solution for all types of prediction issues, including those using picture data. Next, by comparing the execution times of the algorithms, they determined that increasing the number of epochs without changing the configuration of the algorithm is pointless due to the limitation of a certain model, and they discovered that beyond a certain number of epochs, the model begins over- fitting the dataset and provides biased predictions.

An Enhanced Handwritten Digit Recognition Using Convolutional Neural Network (2021)

M. S, C. N. Vanitha, N. Narayan, R. Kumar and G. R

Handwritten digit recognition has a great impact in the applications of deep learning. Convolutional Neural Network in deep learning has become one of the major methods and one of the important factors in the various success in recent times and deep learning is used majorly in the area of object recognition. In the paper work, the speech output feature is integrated along with the text output. Convolutional Neural Network model is applied in the image classification. The dataset used to train and test is the MNIST dataset. There are various applications of handwritten digit recognition in real time. It is applied in detection of vehicle number, reading of bank cheques, the arrangement of letters in the post office.

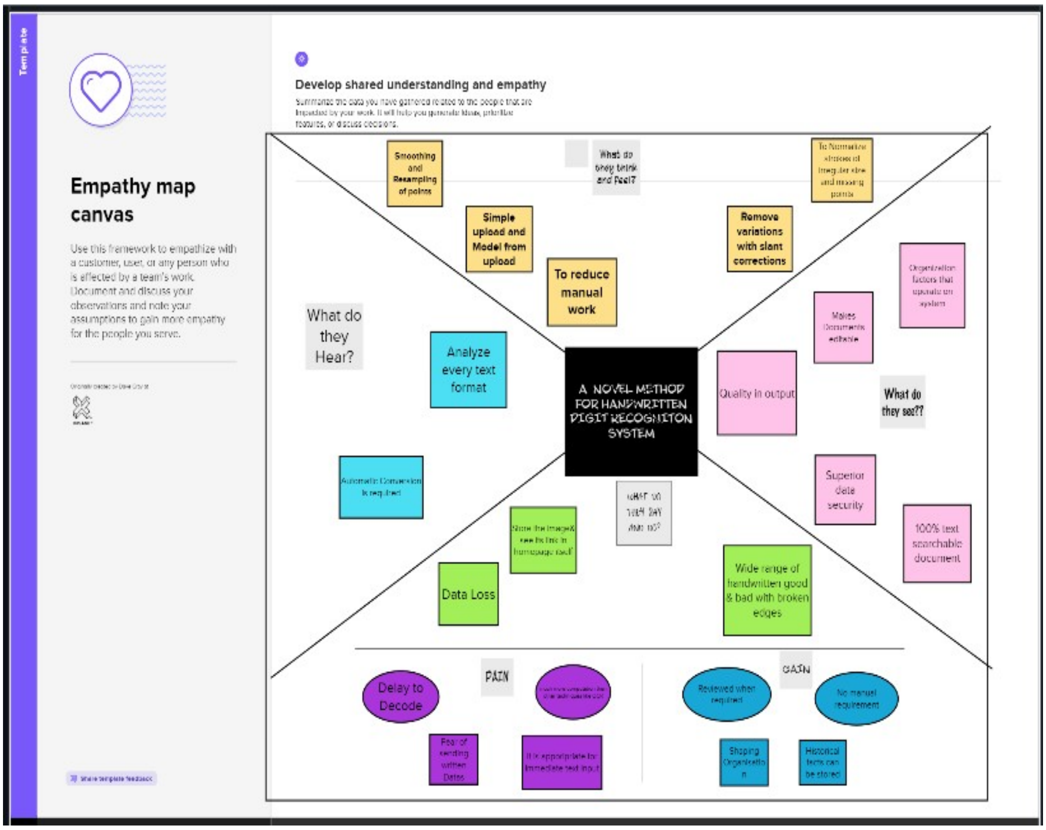
2.3. PROBLEM STATEMENT DEFINITION

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 digitized to reduce human effort. Hence, there comes a need for handwritten digit recognition in many real-time applications. The MNIST data set is widely used for this recognition process and it has 70000 handwritten digits. We use Artificial neural networks to train these images and build a deep learning model. Web application is created where the user can upload an image of a handwritten digit. This image is analyzed by the model and the detected result is returned to the UI. MNIST ("Modified National Institute of Standards and Technology") is considered an unofficial computer vision "hello-world" dataset. This is a collection of thousands of handwritten pictures used to train classification models using Machine Learning techniques.

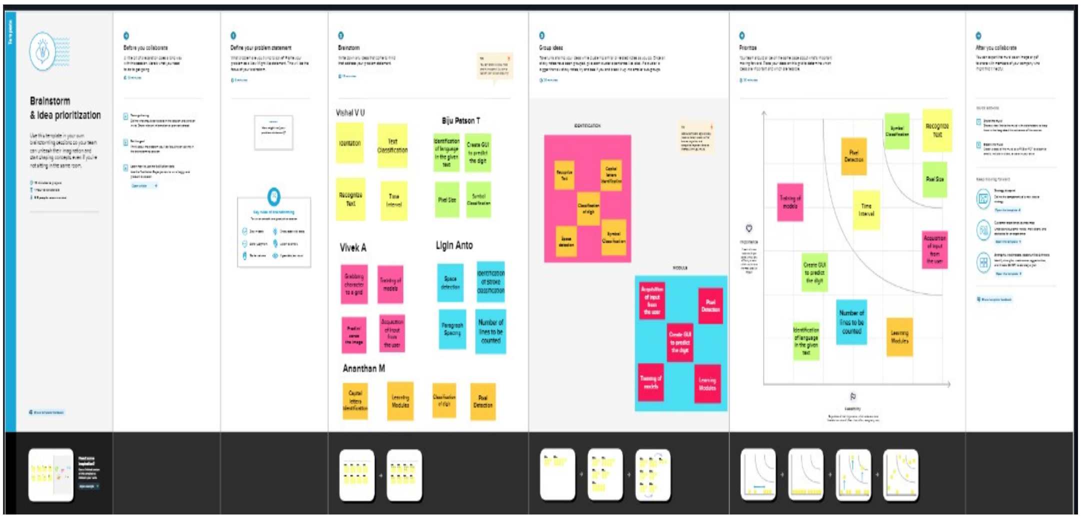
CHAPTER 3

IDEATION AND PROPOSED SOLUTION

3. 3.1. EMPATHY MAP



3.2. IDEATION & BRAINSTORMING



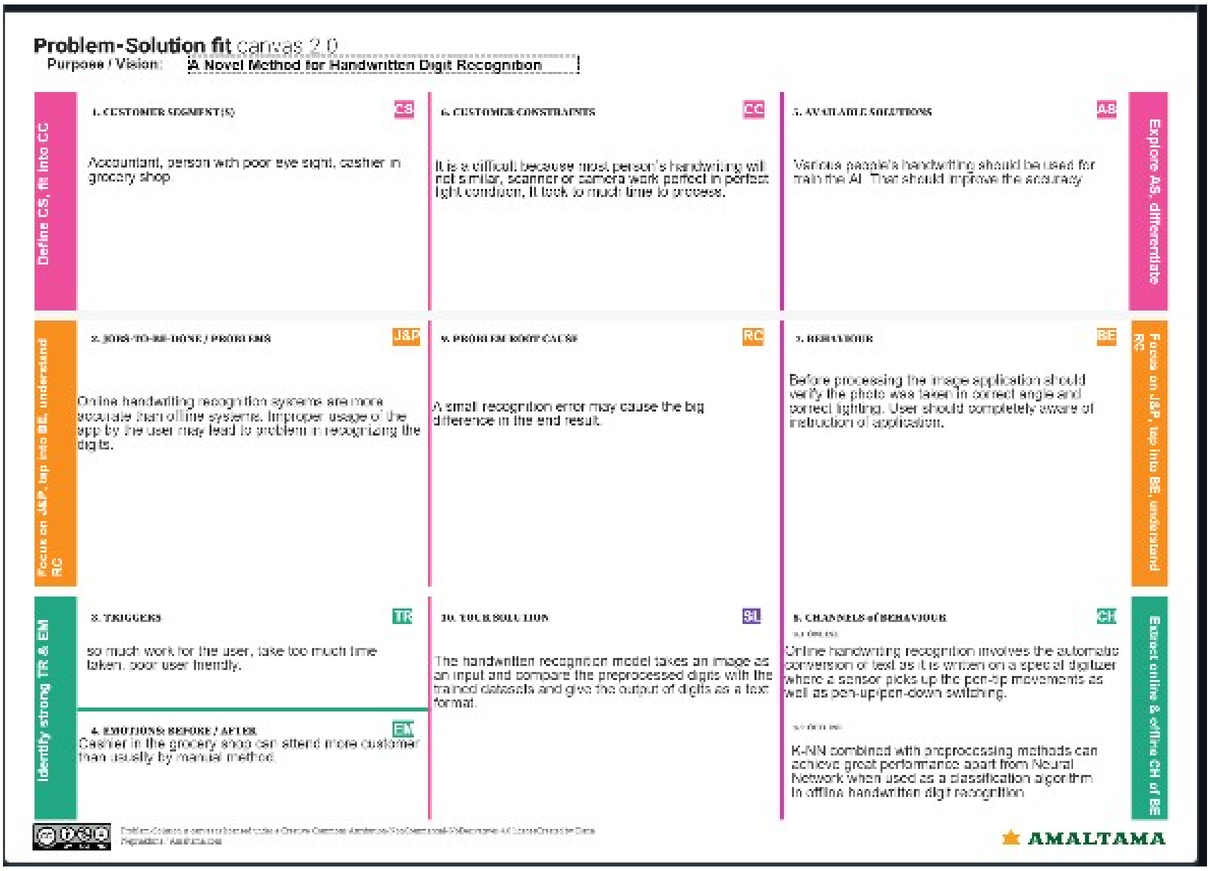
3.3. PROPOSED SOLUTION

PROPOSED SOLUTION

PROPOSED SOLUTION TEMPLATE:

S.No.	Parameter	Description
1.	PROBLEM STATEMENT	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 styles. 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.
2.	IDEA / SOLUTION DESCRIPTION	The idea is simple as that we have to provide an image as the input. The image which is given as the input will be scanned. After the process of scanning, the input image will be converted into digits which will be accurate and easy to access.
3.	NOVELTY	• Recognition of numbers are more accurate. • It will be an faster process to scan the digits instead of entering it manually.
4.	SOCIAL IMPACT	• It reduces the unnecessary writing of digits repeatedly. • Any person even without the knowledge of this process can use this. • The writings can be printed as many times as it is needed.
5.	BUSINESS MODEL	• We can provide pop-up ads, overlay ads, and other advertising services from third party advertisers. • An account can hold multiple profiles at different subscription levels.
6.	FEASIBILITY OF IDEA	• It can be used in laptop, pc and mobile phones. • There is no leakage of data as it can be used offline.
7.	SCALABILITY OF SOLUTION	• Accuracy scores can be enhanced in the later versions. • Enhancing easy recognition in any type of lightings.

4. PROBLEM SOLUTION FIT



Problem-Solution fit canvas 2.0

Purpose / Vision: A Novel Method for Handwritten Digit Recognition

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

Accountant, person with poor eye sight, cashier in grocery shop

6. CUSTOMER CONSTRAINTS CC

It is a difficult because most person's handwriting will not similar, scanner or camera work perfect in perfect light condition, It took to much time to process.

5. AVAILABLE SOLUTIONS AS

Various people's handwriting should be used for train the AI. That should improve the accuracy

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Online handwriting recognition systems are more accurate than offline systems. Improper usage of the app by the user may lead to problem in recognizing the digits.

9. PROBLEM ROOT CAUSE RC

A small recognition error may cause the big difference in the end result.

7. BEHAVIOUR BE

Before processing the image application should verify the photo was taken in correct angle and correct lighting. User should completely aware of instruction of application.

Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

3. TRIGGERS TR

so much work for the user, take too much time taken. poor user friendly.

4. EMOTIONS: BEFORE / AFTER EM

Cashier in the grocery shop can attend more customer than usually by manual method

10. YOUR SOLUTION SL

The handwritten recognition model takes an image as an input and compare the preprocessed digits with the trained datasets and give the output of digits as a text format.

8. CHANNELS of BEHAVIOUR CH

8.1 ONLINE

Online handwriting recognition involves the automatic conversion of text as it is written on a special digitizer where a sensor picks up the pen-tip movements as well as pen-up/pen-down switching.

8.2 OFFLINE

K-NN combined with preprocessing methods can achieve great performance apart from Neural Network when used as a classification algorithm in offline handwritten digit recognition

Extract online & offline CH of BE

AMALTAMA

CHAPTER 4

REQUIREMENT ANALYSIS

1. FUNCTIONAL REQUIREMENTS

FR No.	Sub Requirement (Story / Sub-Task)
FR-1	Image Data: Handwritten digit recognition refers to a computer's capacity to identify human handwritten digits from a variety of sources, such as photographs, documents, touch screens, etc., and categorise them into ten established classifications (0-9). In the realm of deep learning, this has been the subject of countless studies.
FR-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.

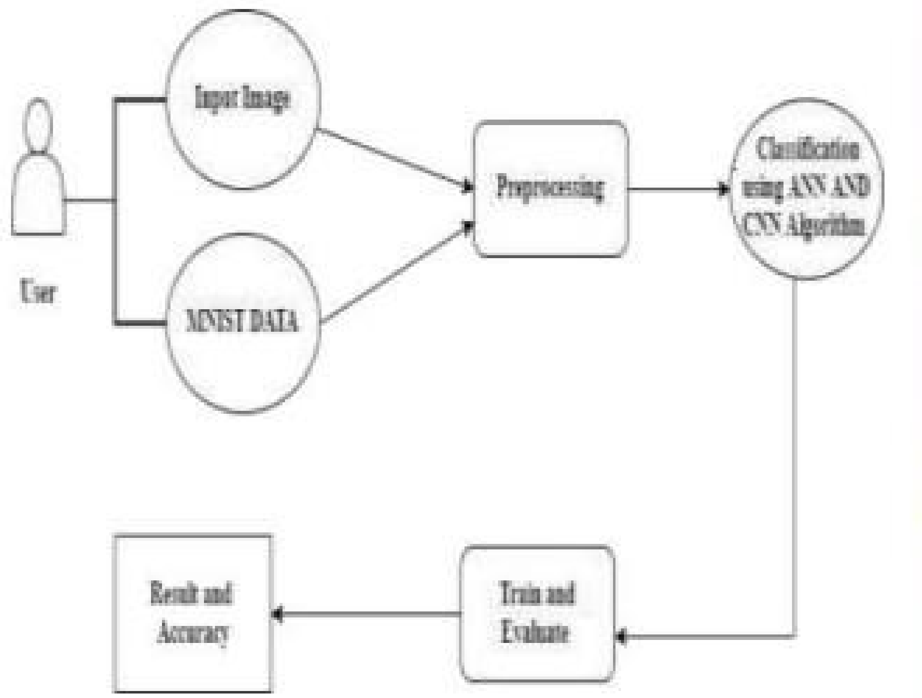
2. NON FUNCTIONAL REQUIREMENTS

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	One of the very significant problems in pattern recognition applications is the recognition of handwritten characters. Applications for digit recognition include filling out forms, processing 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 recognise 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**	

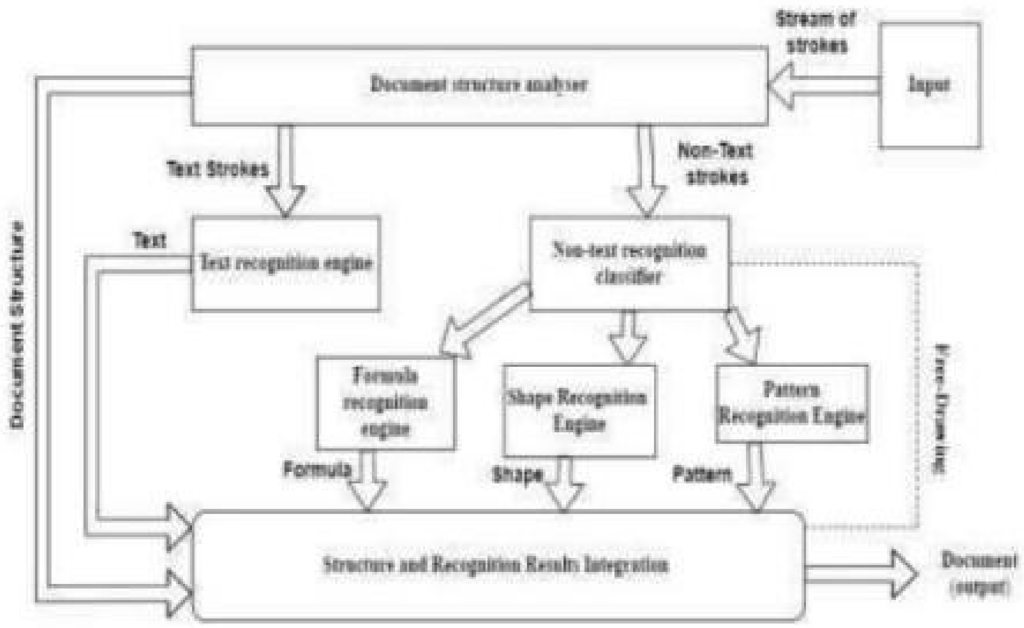
CHAPTER 5

PROJECT DESIGN

1. DATA FLOW DIAGRAM



2. SOLUTION & TECHNICAL ARCHITECTURE



3. USER STORIES

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 AND SCHEDULING

1. SPRINT PLANNING AND ESTIMATION

Sprint Delivery Plan

Project Tracker, Velocity & Burndown Chart:

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	10	5 Days	24 Oct 2022	29 Oct 2022	10	29 Oct 2022
Sprint-2	10	5 Days	31 Oct 2022	05 Nov 2022	10	05 Nov 2022
Sprint-3	10	5 Days	07 Nov 2022	12 Nov 2022	10	12 Nov 2022
Sprint-4	10	5 Days	14 Nov 2022	19 Nov 2022	10	19 Nov 2022

2. SPRINT DELIVERY SCHEDULE

Sprint Delivery Plan

Project Tracker, Velocity & Burndown Chart:

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	10	5 Days	24 Oct 2022	29 Oct 2022	10	29 Oct 2022
Sprint-2	10	5 Days	31 Oct 2022	05 Nov 2022	10	05 Nov 2022
Sprint-3	10	5 Days	07 Nov 2022	12 Nov 2022	10	12 Nov 2022
Sprint-4	10	5 Days	14 Nov 2022	19 Nov 2022	10	19 Nov 2022

Velocity:

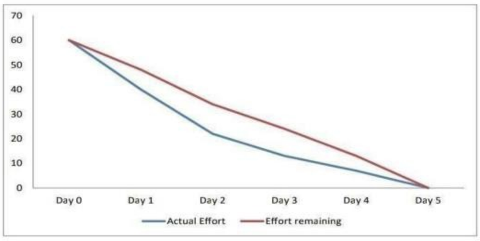
Imagine we have a 5-day sprint duration, and the velocity of the team is 10 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

AV=Sprint Duration/Velocity=10/5=2

3. REPORTS

Burndown Chart:

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over tim



CHAPTER 7

CODING

PROGRAM

app.py

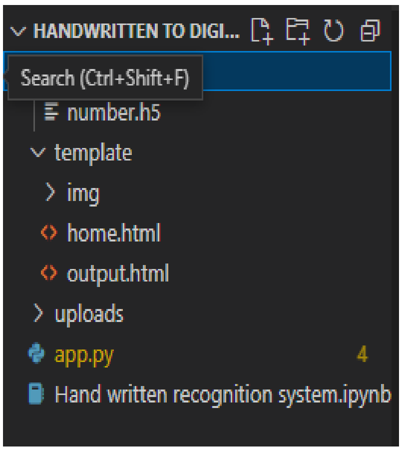
```python
import os
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
from flask import Flask, render_template, request

app = Flask(__name__)
model=load_model("model/number.h5")

@app.route('/')
def home() :
    return render_template('home.html')

@app.route('/getdata',methods=['GET','POST'])
def upload() :
    if request.method=='POST':
        f=request.files['image']
        basepath=os.path.dirname(__name__)
        filepath=os.path.join(basepath,'uploads',f.filename)
        f.save(filepath)
        img=image.load_img(filepath,target_size=(64,64))
        x=image.img_to_array(img)
        x=np.expand_dims(x,axis=0)
        pred=np.argmax(model.predict(x),axis=1)
        index=['.', '*', '8', '=', '5', '4','-', '9', '1', '+', '7', '6', '3', '*', '2', '0']
        text="Your Output is : "+str(index[pred[0]])
    return render_template('output.html',output=text)

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



INPUT HTML PAGE:

template > home.html > ...

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>A Novel Method for Handwritten Digit Recognition System</title>
</head>
<script>
    var loadFile = function(event) {
    var image = document.getElementById('output');
    image.src = URL.createObjectURL(event.target.files[0]);
};
</script>
<style>
    body{
    margin: 0px;
    margin-top: -27px;

    }
    .topic{
    background-color: brown;
    color: white;
    font-size: 40px;
    width:100%
    }
    .content{
        width: 100%;
        display: flex;
        background-image: url('img/writting.avif');
        height: 750px;

        background-repeat: no-repeat;
```

OUTPUT HTML PAGE:

template > output.html > ...

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Output</title>
</head>
<style>
    body{
    margin: 0px;
    margin-top: -27px;

    }
    .topic{
    background-color: brown;
    color: white;
    font-size: 40px;
    width:100%
    }
    .content{
        width: 100%;
        display: flex;
        background-image: url('img/AI.webp');
        height: 650px;
        background-repeat: no-repeat;
        background-size: cover;
        background-position: center;
        margin-top: -27px;
    }
    .content1{
```

CHAPTER 8

TESTING

1. TESTING CASE

In [ ]:
```
model.predict(x)
```

```
1/1 [==============================] - 0s 15ms/step
```

Out[ ]:
```
array([[0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.]],
      dtype=float32)
```

In [ ]:
```
xtrain.class_indices
```

Out[ ]:
```
{'decimal': 0,
 'div': 1,
 'eight': 2,
 'equal': 3,
 'five': 4,
 'four': 5,
 'minus': 6,
 'nine': 7,
 'one': 8,
 'plus': 9,
 'seven': 10,
 'six': 11,
 'three': 12,
 'times': 13,
 'two': 14,
 'zero': 15}
```

In [ ]:
```
op=['decimal', 'div', 'eight', 'equal', 'five', 'four','minus', 'nine', 'one', 'plus', 'seven', 'six', 'three', 'times', 'two', 'zero']
pred = np.argmax(model.predict(x))
op[pred]
```

```
1/1 [==============================] - 0s 15ms/step
```

Out[ ]:
```
'eight'
```

test

In [ ]:
```
img = image.load_img('/content/dataset/test/minus/141.jpg',target_size=(64,64))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred= np.argmax(model.predict(x))
op[pred]
```

```
1/1 [==============================] - 0s 16ms/step
```

Out[ ]:
```
'minus'
```

In [ ]:

2. USER ACCEPTANCE TESING

1. DEFECT ANALYSIS

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Total
By Design	1	0	1	0	2
Duplicate	0	0	0	0	0
External	0	0	2	0	2
Fixed	4	1	0	1	6
Not Reproduced	0	0	0	1	1
Skipped	0	0	0	1	1
Won’t Fix	1	0	1	0	2
Total	6	1	4	3	14

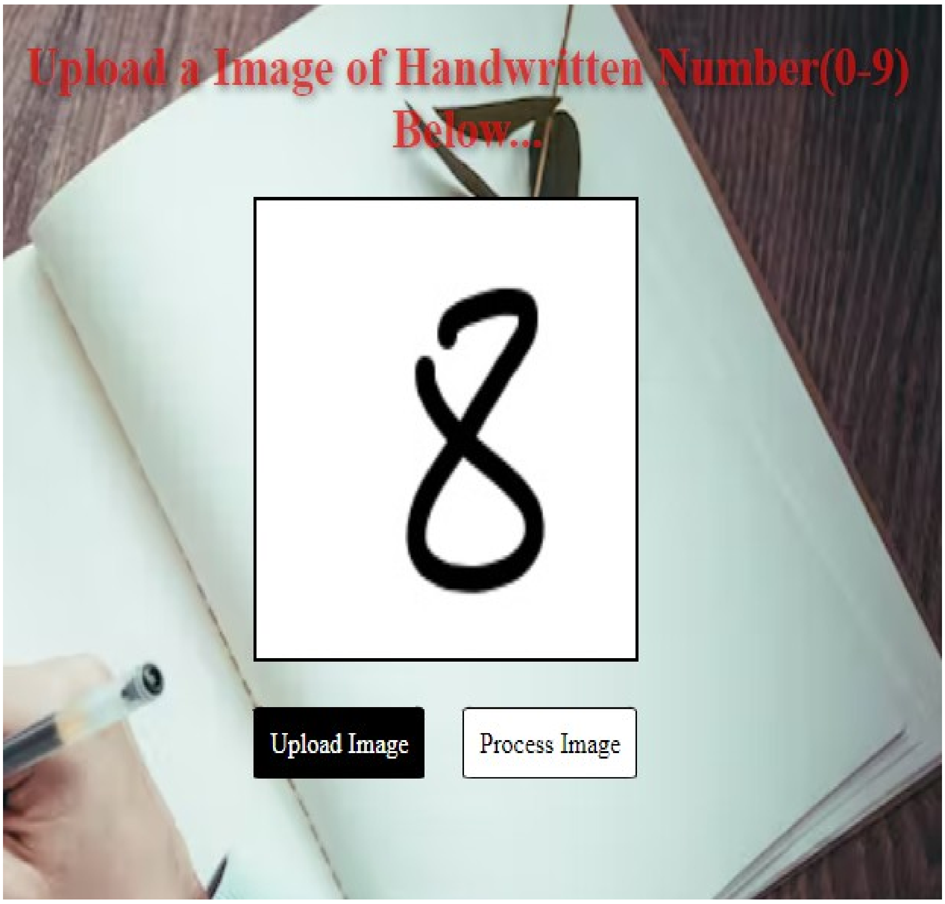
2. TESTING CASE ANALAYSIS

Section	Total Cases	Not Tested	Fail	Pass
Client Application	10	0	3	7
Security	2	0	1	1
Performance	3	0	1	2
Exception Reporting	2	0	0	2

CHAPTER 9

RESULT

9.1. OUTPUT



CHAPTER 10

ADVANTAGES & DISADVANTADES

ADVATAGES

- Reduces manual work
- Backups
- More accurate than average human
- Capable of handling a lot of data
- Can be used anywhere from any device

DISADVANTAGES

- Cannot handle complex data
- Low retention
- All the data must be in digital format
- Requires a high-performance server for faster predictions
- Prone to occasional errors

CHAPTER 11

CONCLUSION

This project demonstrated a web application that uses machine learning to recognise handwritten numbers. Flask, HTML, CSS, JavaScript, and a few other technologies were used to create this project. The model predicts the handwritten digit using a CNN network. During testing, the model achieved a 99.61% recognition rate. The proposed project is scalable and can easily handle a huge number of users. Since it is a web application, it is compatible with any device that can run a browser. This project is extremely useful in real-world scenarios such as recognizing number plates of vehicles, processing bank cheque amounts, numeric entries in forms filled up by hand (tax forms) and so on. There is so much room for improvement, which can be implemented in subsequent versions.