HANDWRITTEN DIGIT RECOGNITION

TEAM ID - PNT2022TMID01309

TABLE OF CONTENTS

1. INTRODUCTION

1.1 PROJECT OVERVIEW

Traditional methods of recoganize handwriting rely heavily on a lot of prior knowledge like Optical Character Recognition (OCR). Since the style of handwriting changes with every individual, it is a challenging task in identifying the characters correctly. The thickness of stroke, style carries uniqueness with different person depending on them. The rapid growth in the need for digitizing handwritten data and the availability of massive processing power demands improvement in recognition accuracy. Hence a highly proficient algorithm is required when dealing with handwriting recognition. Handwritten digit recognition can be done using deep learning methods effectively. The Convolutional Neural Networks (CNN) is a deep learning algorithm that is highly suitable for image recognition and those tasks involving processing of pixel data.MNIST data set is widely used for this recognition process and it has 70000 handwritten digits. Those images are split as train set and test set images. Artificial neural networks is used to train these images and build a deep learningmodel. 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

Each individual has a unique handwriting style which makes it a bit complex to identify the digits. If the handwritten digit recognition becomes an efficient practice, this will help digitize number processing. Huge amounts of data can be processed by machine which will save loads of time. In today's world, technology plays a major role in handling data, therefore it is important to bring this system in managing data. Workers at the postal office sorting throughs mails using the postal code can be helped using this. This also comes handy while arranging records and huge amounts of information. Manual labour is eased and it saves upa lot of time. It can be used in programming checks and in case of tax documentation. The Labour cost will also be reduced with the help of machines. There are also the activities of processing bank checks and tax documentations. Large piles of records and archives can be arranged and sorted well easing the stress and work load from manual labourers

2. LITERATURE SURVEY

2.1 EXISTING PROBLEM

Because of the progress in science and technology everything is being digitalized to reduce human effort. It takes a lot of time and effort on the side ofmanual workers when sorting through mails by postal codes. It is not an easy taskto handle data by human worker. There is also the possibility of human error while processing huge amount of data. Therefore, digitizing these will help reduce time and labour. The labour cost will also be reduced with the help of machines. There are also the activities of processing bank checks and tax documentations. Large piles of records and archives can be arranged and sorted well easing the stress and work load from manual labourers. The problem with handwriting is that every individual has different style of writing. There is a differing thickness of stroke, style and general uniqueness that just brings a levelof hardness in identifying the handwriting. The machine must be capable of picking up the digits correctly with a good accuracy rate. Hence a highly proficient algorithm is required when dealing with handwriting recognition. Handwritten digit recognition can be done using deep learning methods effectively.

2.2 REFERNCES

[1] Handwritten Digit Recognition using Machine and Deep Learning Algorithms by Ritik Dixit of Computer Science and Engineering, Rishika Kushwah of Computer Science and Engineering, Samay Pashine of Computer Science and Engineering, Acropolis Institute of Technology & Research, Indore, India, 23 June 2021

[2] Recognition of Handwritten Digit using Convolutional Neural Network (CNN), By Md. Anwar Hossain & Md. Mohon Ali, Pabna University of Science& Technology, 2019

[3] On the Brittleness of Handwritten Digit Recognition Models, Alexander K. Seewald, 30 Nov 2011

[4] A Novel Handwritten Digit Classification System Based on Convolution al Neural Network Approach, Ali Abdullah Yahya, 18 September 2021

[5] Handwritten Digit Recognition Using Machine Learning Algorithms, S. M. Shamim, Md Badrul Alam Miah, Angona Sarker, Masud Rana, Abdullah Al Jobair, March 2018

[6]. Handwritten digit recognition using deep learning, International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), vol. 6,no. 7, A. Dutta and A. Dutta, July 2017

[7]. Automatic prediction of age, gender, and nationality in offline handwriting. EURASIP Journal on Image and Video Processing, no. 1, Al Maadeed, Somaya,and Abdelaali Hassaine, 2014.

[8]. Handwritten Digits Recognition, Project Report, Gaurav Jain, Jason Ko, University of Toronto, 11/21/2008.

[9]. Handwritten recognition using SVM, KNN an neural network, arXiv preprintarXiv:1702.00723. Hamid, Norhidayu Abdul, and NilamNur Amir

Sjarif, 2017.

[10]. Digit Classification using the MNIST Dataset, M. Wu and Z. Zhang, Handwritten, 2010.

[11]. Handwritten digit classification using support vector machines, R.G.Mihalyi, 2011.

2.3 PROBLEM STATEMENT DEFINITION

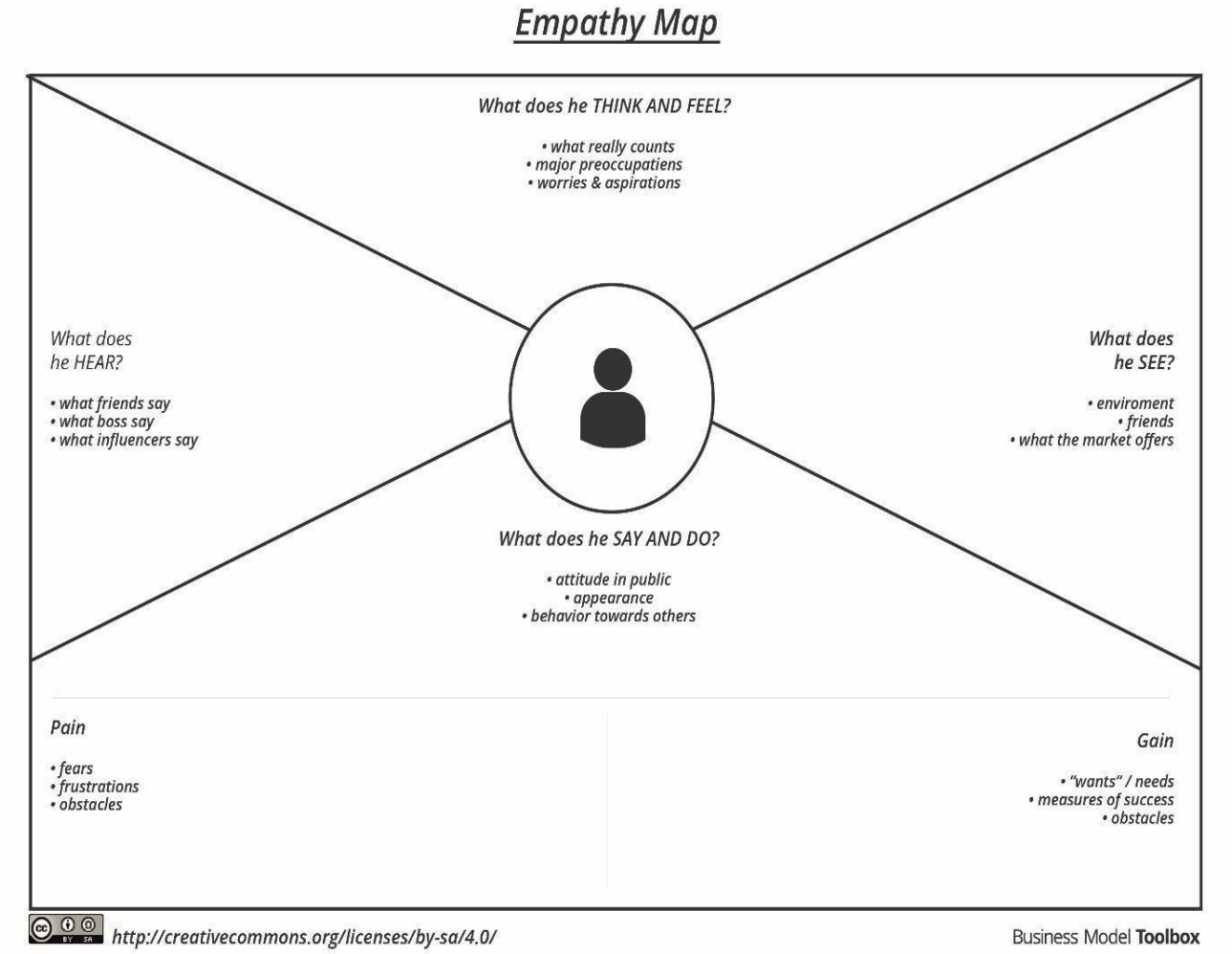
Handwriting recognition is one of the compelling research works going on because every individual in this world has their own style of writing. Since the style of handwriting changes with every individual, it is a challenging task in identifying the characters correctly. The thickness of stroke, style carries uniqueness with different person depending on them. 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 needfor handwritten digit recognition in many real-time applications. MNIST data set is widely used for this recognition process and it has 70000 handwritten digits. Artificial neural network is used to train these images and build a deep learning model. The Convolutional Neural Networks (CNN) is a deep learning algorithmthat is highly suitable for image recognition and those tasks involving processing of pixel data. Convolutional neural networks (CNNs) are very effective in perceiving the structure of handwritten characters/words in ways that help in automatic extraction of distinct features and make CNN the most suitable approach for solving handwriting recognition problems. Our aim in the proposed work is to deploy the CNN model effectively and produce a good result with better accuracy. The main objective was to actualize a pattern characterization method to perceive the handwritten digits provided in the MINIST data set of images of handwritten digits (0-9). Web application is created where the user canupload an image of a handwritten digit. This image is analyzed by the model andthe detected result is returned on to UI.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

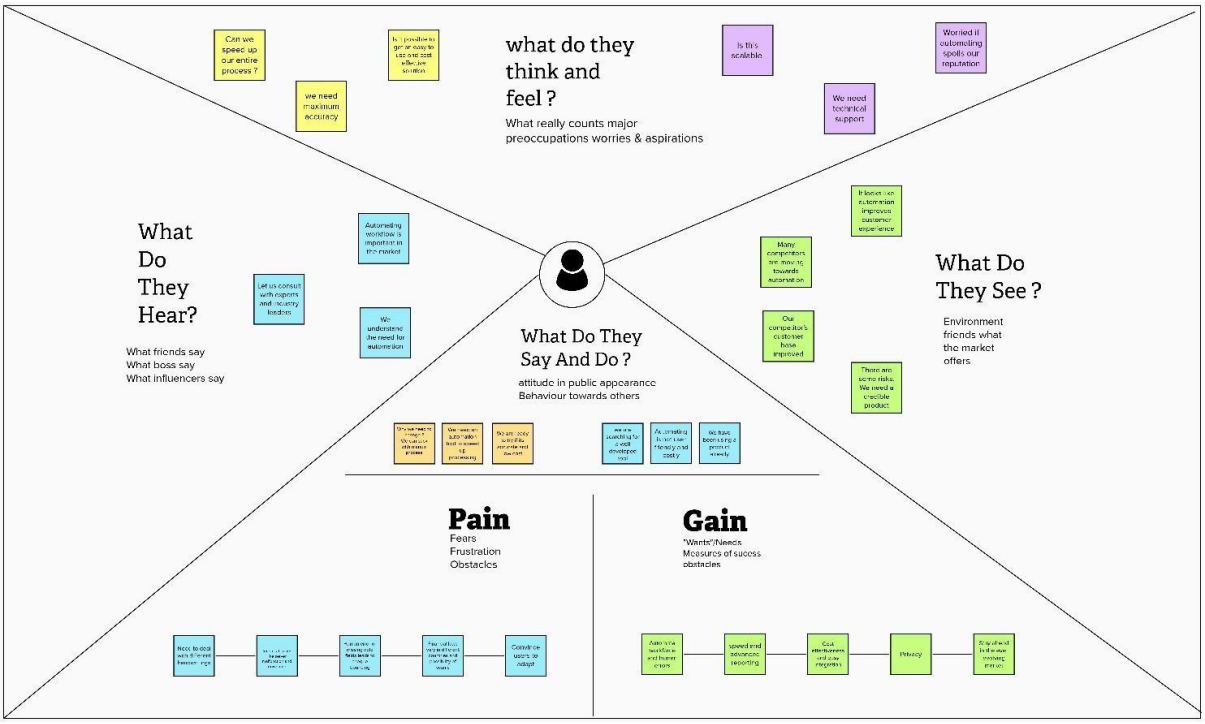
An Empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise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

Example**:**



Reference: https://www.mural.co/templates/empathy-map-canvas

Example: A Novel Method for Handwritten Digit Recognition System



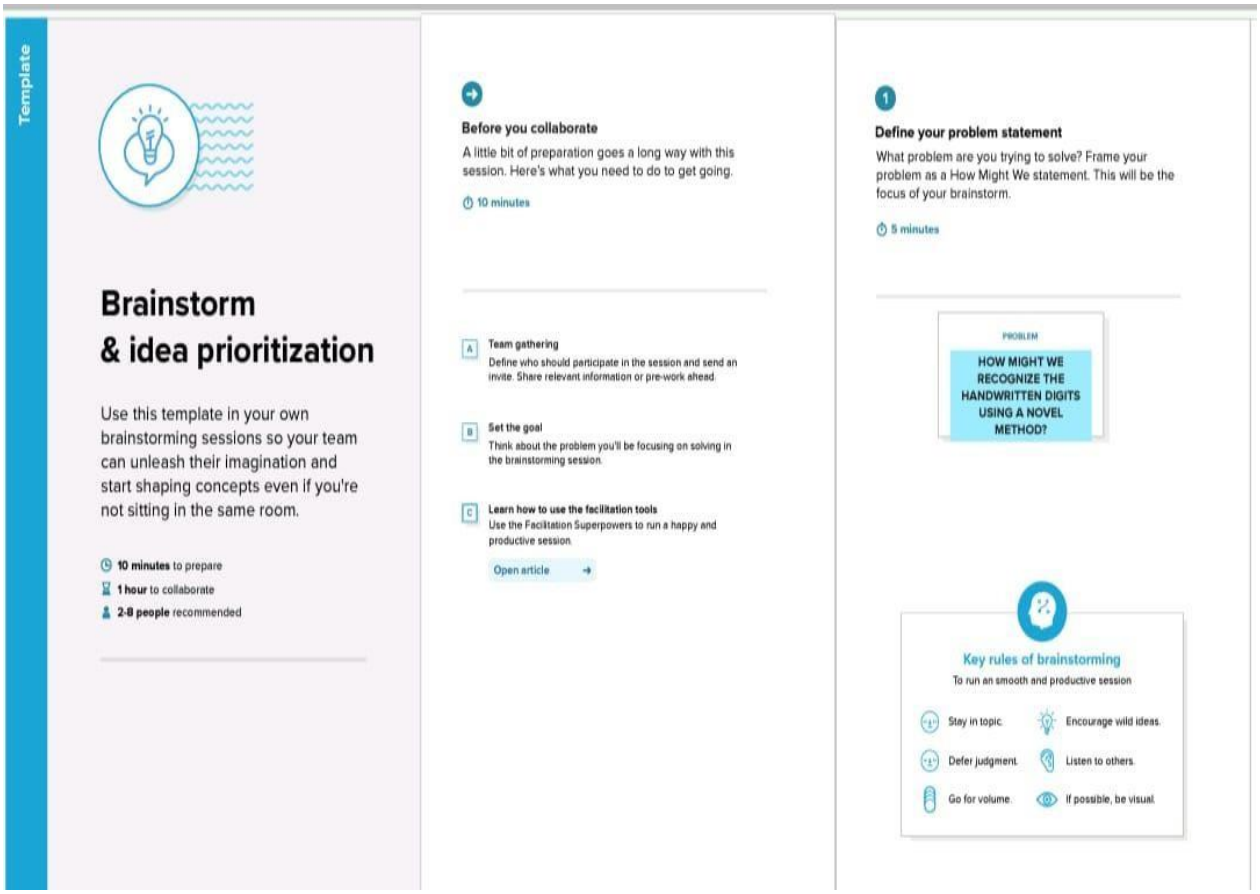
3.2 Ideation & Brainstorming

Brainstorming provides a free and open environment that encourages everyone within a teamto participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants areencouraged to collaborate, helping each other develop a rich Amount of creative solutions.

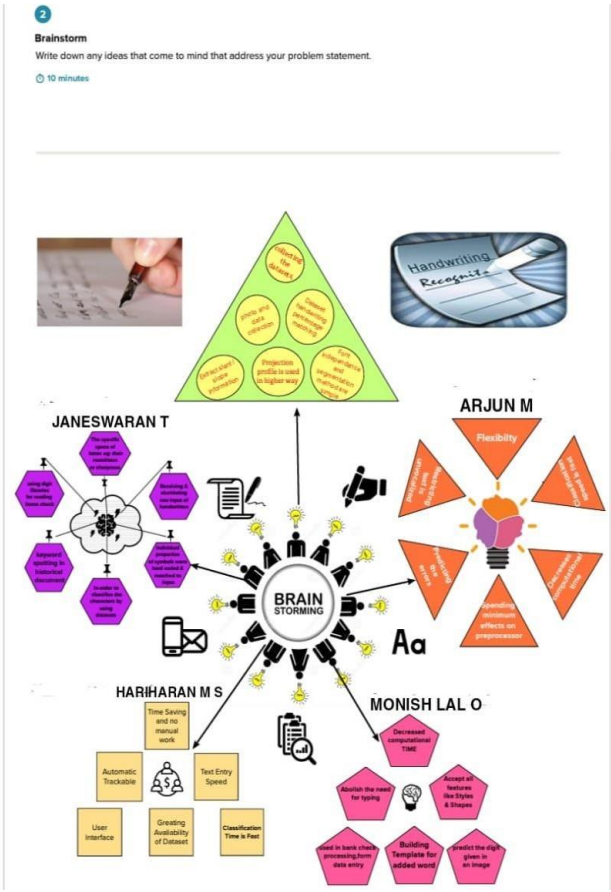
Use this template in your own brainstorming sessions so your team can unleash theirimagination and start shaping concepts even if you're not sitting in the same room.

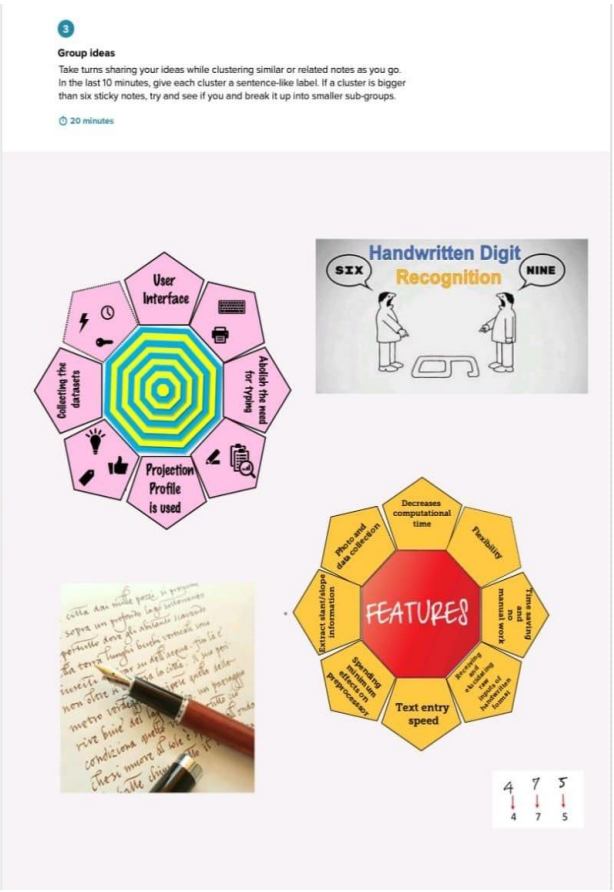
Reference**:** https://www.mural.co/templates/empathy-map-canvas

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



Step-2: Brainstorm, Idea Listing and Grouping





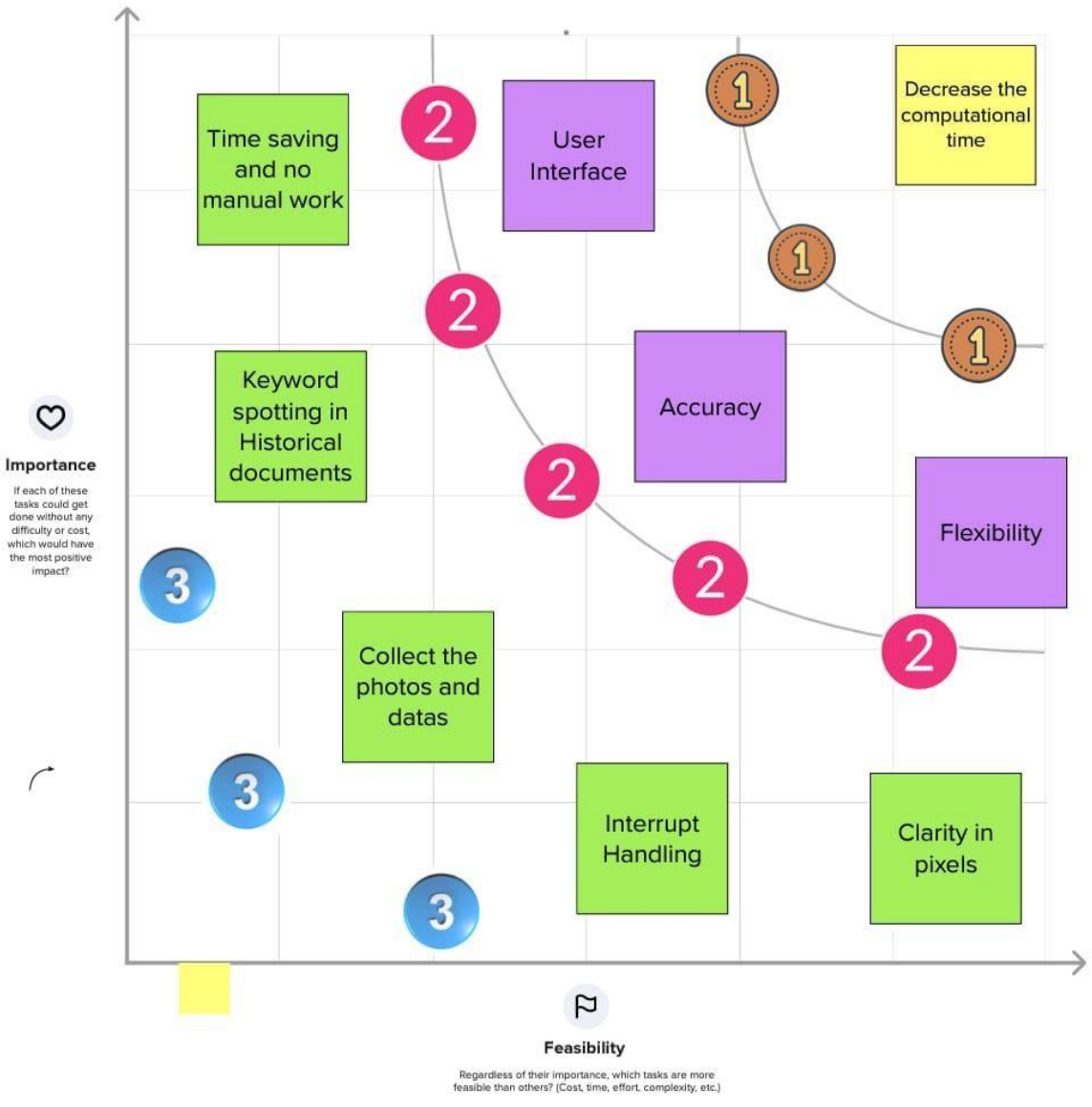
Step-3: Idea Prioritization

4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

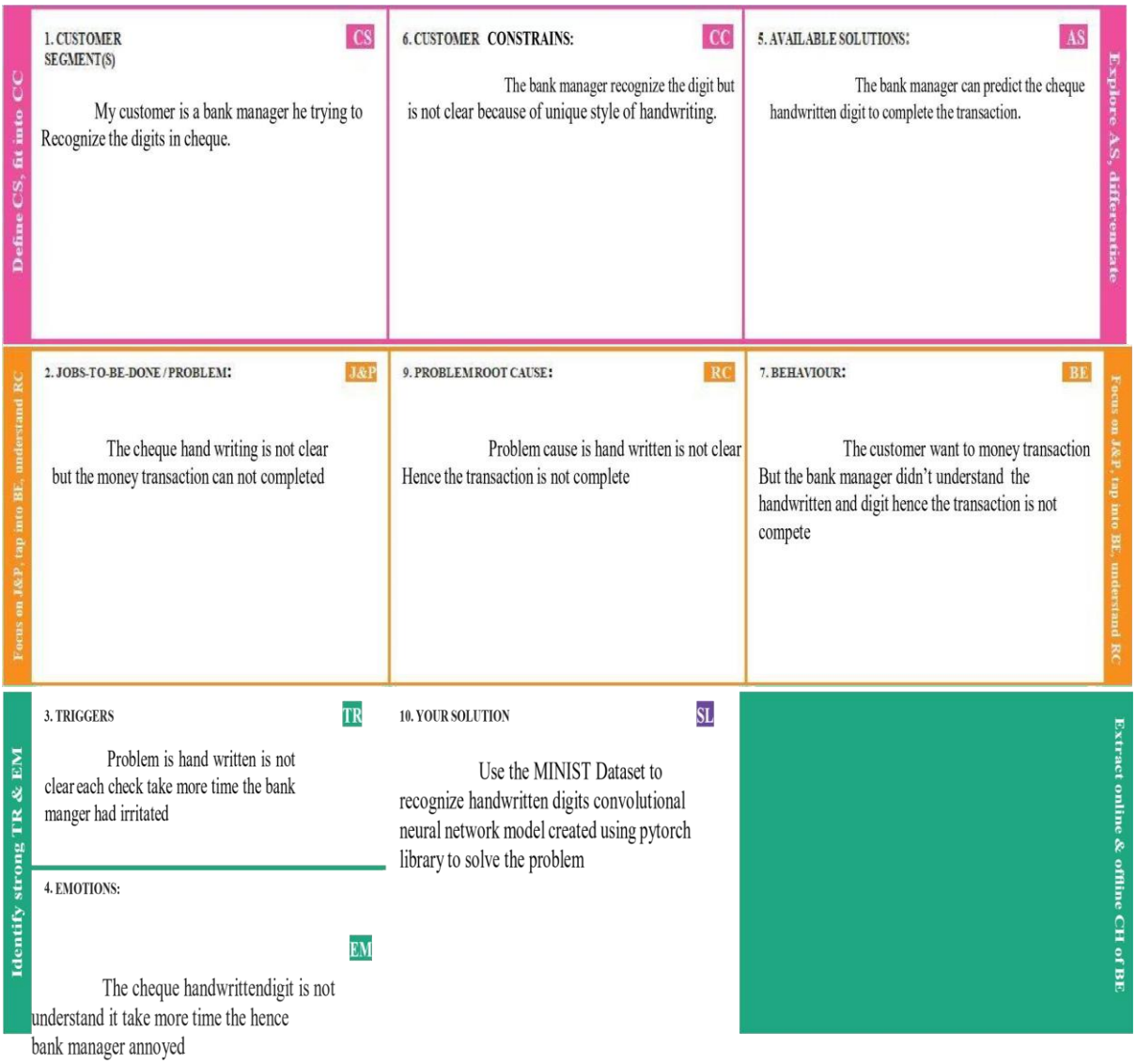
20 minutes



3.3 Proposed Solution

S.NO	Parameter	Description
1.	Problem Statement (Problem to be solved)	**Statement**-The handwritten digit recognition is the capability of computer applications to recognize the human handwritten digits. **Description**: It is a hard task for the machine because handwritten digits are not perfect and can be made with many different shapes and sizes.
2.	Idea / Solution description	1. It is the capability of a computer to fete the mortal handwritten integers from different sources like images, papers, touch defences. 2. It allows user to translate all those signature and notes into electronic words in a text document format and this data only requires far less physical space than the storage of the physical copies.
3.	Novelty / Uniqueness	Accurately recognize the digits rather than recognizing all the characters like OCR.
4.	Social Impact / Customer Satisfaction	1. Artificial Intelligence developed the app calledHandwritten digit Recognizer. 2. It converts the written word into digital approximations and utilizes complex algorithms to identify characters before churning out a digital approximation.
5.	Business Model (Revenue Model)	• This system can be integrated with traffic surveillance cameras to recognize the vehicle's number plates for effective traffic management.
		• Can be integrated with Postal system to identify and recognize the pin-code details easily.
6.	Scalability of the Solution	• Ability to recognize digits in more noisyenvironments. • There is no limit in the number of digits it can be recognized.

3.4 Problem Solution fit



4. REQUIREMENT ANALYSIS

4.1 Functional requirement

Following are the functional requirements of the proposed solution.

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 categorize them into ten established classifications (0-9). In the realm of deep learning, this has been the subject of countless studies.
FR-2	Website: Web hosting makes the code, graphics, and other items that make up a website accessible online. A server hosts every website you've ever visited. The type of hosting determines how much space is allotted to a website on a server. Shared, dedicated, VPS, and reseller hosting are the four basic varieties.
FR-3	Digit Classifier Model: To train a convolutional network to predict the digit from an image, use the MNIST database of handwritten digits and get the training and validation data first.
FR-4	Cloud: The cloud offers a range of IT services, including virtual storage, networking, servers, databases, and applications. In plain English, cloud computing is described as a virtual platform that enables unlimited storage and access to your data over the internet.
FR-5	Modified National Institute of Standards and Technology dataset: The abbreviation MNIST stands for the MNIST dataset. It is a collection of 60,000 tiny square grayscale photographs, each measuring 28 by 28, comprising handwritten single digits between 0 and 9.

4.2 Non-Functional requirements

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

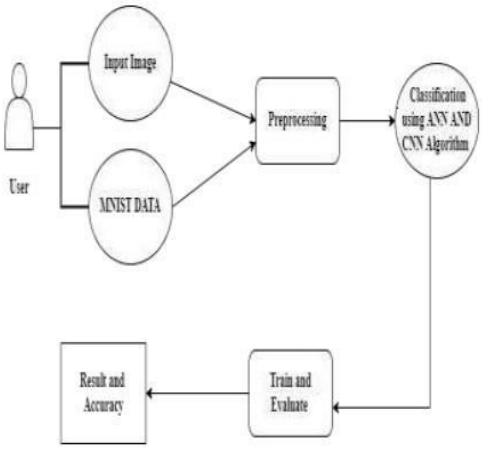
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	One of the very significant problems in pattern recognition applications is the recognition of handwritten characters. Applications for digit recognition include filling out forms, processing bank checks, and sorting mail.
NFR-2	Security	1. The system generates a thorough description of the instantiation parameters, which might reveal information like the writing style, in addition to a categorization of the digit. 2. The generative models are capable of segmentation driven by recognition. 3. The procedure uses a relatively.
NFR-3	Reliability	The samples are used by the neural network to automatically deduce rules for reading handwritten digits. Furthermore, the network may learn more about handwriting and hence enhance its accuracy by increasing the quantity of training instances. Numerous techniques and algorithms, such as Deep Learning/CNN, SVM, Gaussian Naive Bayes, KNN, Decision Trees, Random Forests, etc., can be used to recognize handwritten numbers.
FR-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.

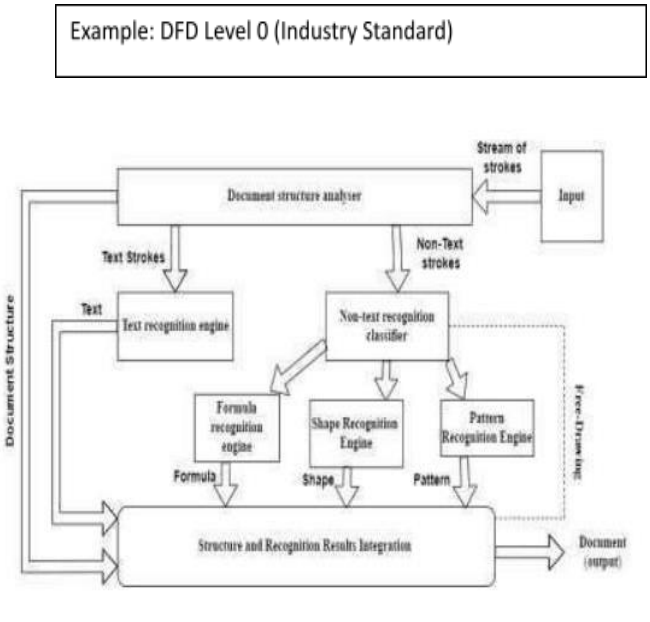
NFR-5	Availability	The features for handwritten digit recognition have been Acquainted. These features are based on shape analysis of the digit image and extract slant or slope information. They are effective in obtaining good recognition of accuracy.
NFR-6	Scalability	The scalability in the task of handwritten digit recognition, using a classifier, has great importance and it makes use of online handwriting recognition on computer tablets, recognizing zip codes on mail for postal mail sorting, processing bank check amounts, numeric entries in forms filled up manually(for example - tax forms) and so on.

5. PROJECT DESIGN

5.1 Data Flow Diagrams

Simplified:





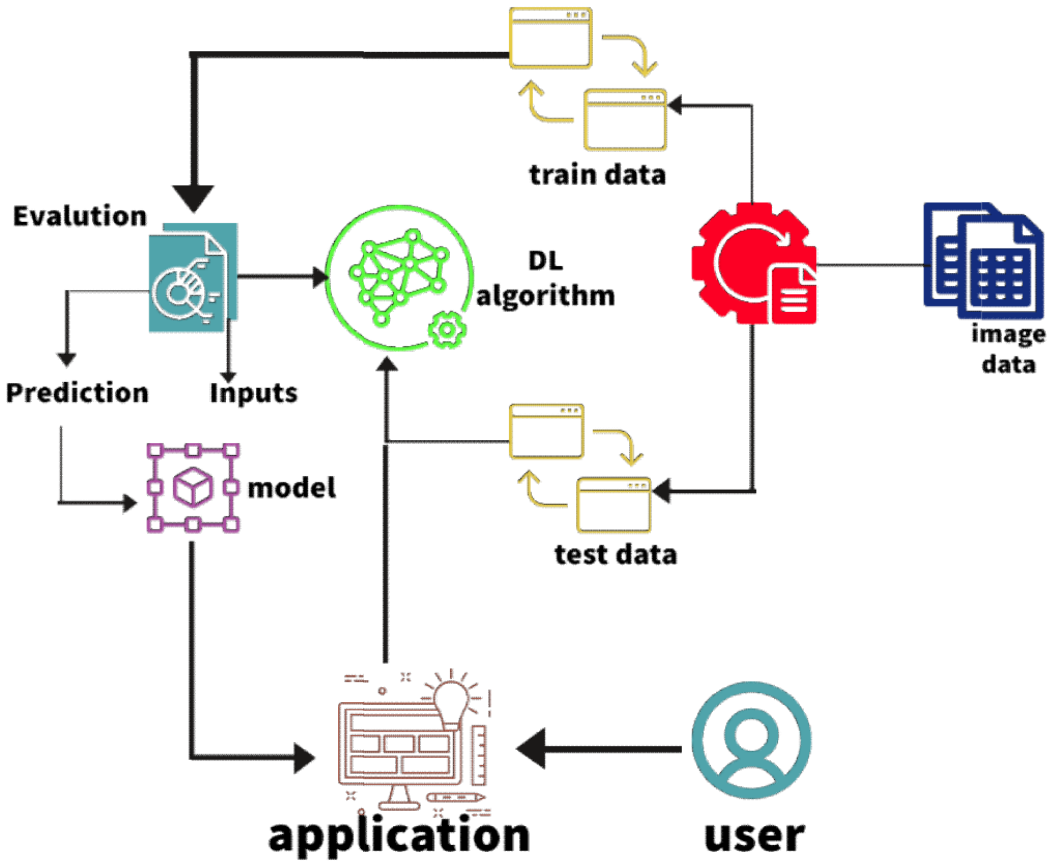
5.2 Solution & Technical Architecture

Solution Architecture:

Solution architecture is a complex process – with many sub-processes – that bridges thegap between business problems and technology solutions. Its goals are to:

- Find the best tech solution to solve existing business problems.
- Describe the structure, characteristics, Behaviour, and other aspects of the software to project stakeholders
- Define features, development phases, and solution requirements
- Provide specifications according to which the solution is defined, managed, anddelivered.

Solution Architecture Diagram:



Architecture -A Novel Method for Handwritten Digit Recognition System

Technical Architecture:

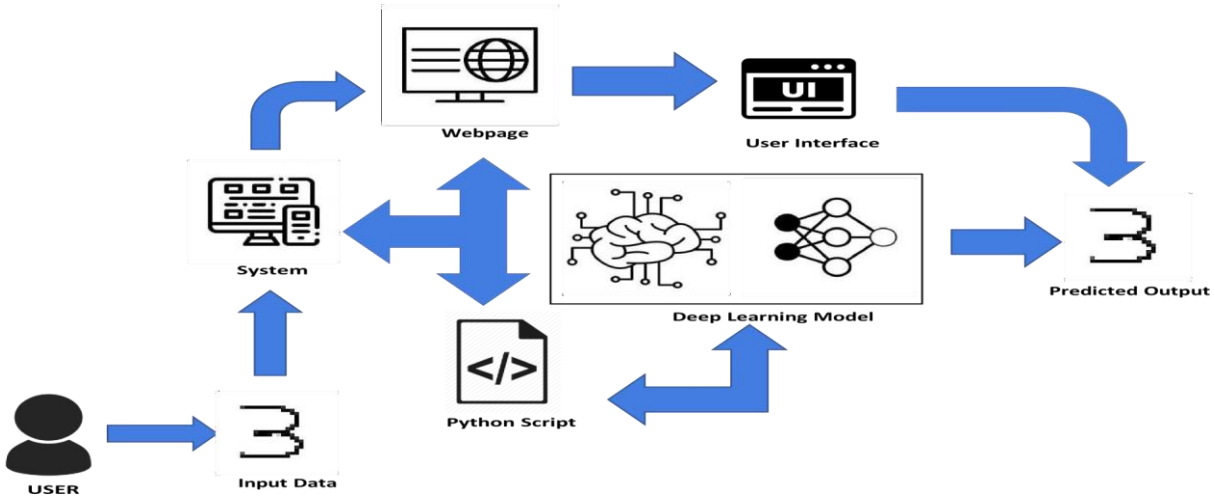


Table-1: Components & Technologies:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	Technology of Opensource framework
2.	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	SHA-256, Encryptions, IAM Controls, OWASP

S. No	Component	Description	Technology
1.	User Interface	How user interacts with application e g... Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript
2.	Application Logic-1	Logic for a process in the application	Python
3.	Application Logic-2	Logic for a process in the application	IBM Watson STT service
4.	Application Logic-3	Logic for a process in the application	IBM Watson Assistant
5.	Database	Data Type, Configurations etc.	MySQL, NoSQL, etc.
6.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloud
7.	File Storage	File storage requirements	IBM Block Storage
8.	External API-1	Purpose of External API used in the application	IBM Weather API
9.	External API-2	Purpose of External API used in the application	Aadhar API
10.	Machine Learning Model	Purpose of Machine Learning Model	Object Recognition Model
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration Cloud Server Configuration.	Local, Cloud Foundry

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
3.	Scalable Architecture	Justify the scalability of architecture	3 – tier, Micro-services
4.	Availability	Abstract and Figures. The features for handwrittendigit recognition have been introduced. These features are based on shape analysis of the digit image and extract slant or slope information. Theyare effective in obtaining good recognition accuracies	Distributed servers, IBM cloud
5.	Performance	The standard implementations of neural networksachieve an accuracy of ~ (98–99) percent in correctly classifying the handwritten digits.	number of requests per sec, useof Cache, use of CDN's

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 onceI 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 & accessthe dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail	I can register the applicatio n 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 appli catio n	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 ofdigital 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 recognized 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 getsmaximum accuracy of the result.	Medium	Sprint-4
Customer (Webuser)	Accessibility	USN-10	As a user, I can use the web application virtually anywhere.	I can use the applicationin any device with a browser	Medium	Sprint-4

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Data Collection & preprocessing	USN-1	As a user, I can upload any kind of image with the pre-processing step is involved in it.	10	High	Hariharan MS,Monish Lal O
Sprint-1		USN-2	As a user, I can upload the image in any resolution .	10	High	Janeswaran T,Arjun M
Sprint-2	Building the Machine learning model	USN-3	As a user, I will get a application with ML model which provides high accuracy of recognized handwritten digit	3	Medium	Janeswaran T,Arjun M
Sprint-2		USN-4	As a user, I can pass the handwritten digit image for recognizing the digit.	2	Medium	Janeswaran T, Arjun M

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-2		USN-5	As a user, I can get the most suitable recognized digit.	10	High	Hariharan MS,Monish Lal O
Sprint-3	Building User Interface Application	USN-6	As a user, I will upload the handwritten digit image to the application by clicking a upload button.	8	Medium	Janeswaran T,Arjun M
Sprint-3		USN-7	As a user, I can know the details of the fundamental usage of the application.	2	High	Hariharan MS, Janeswaran T
Sprint-3		USN-8	As a user, I can see the predicted / recognizeddigits in the application	10	Medium	Arjun M, Monish Lal O
Sprint-4	Train and deployment of modelin IBM Cloud	USN-9	As a user, I can access the web application andmake the use of the product from anywhere	20	High	Janeswaran T,Arjun M, Hariharan MS,Monish Lal O

6.2 SPRINT DELIVERY SCHEDULE

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

6.3 REPORTS FROM JIRA

Velocity:

Imagine we have a 6-day sprint duration, and the velocity of the team is 20 (points persprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

AV = *sprint duration / velocity*

=20/6=3.33

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 time

7. CODING AND SOLUTIONING

7.1 FEATURE-1 MODEL BUILDING

ML depends heavily on data, without data, it is impossible for a machine tolearn. It is the most crucial aspect that makes algorithm training possible. In Machine Learning projects, we need a training data set. It is the actual data set used to train the model for performing various actions. TensorFlow already hasMNIST Data set so there is no need to explicitly download or create Dataset.
The MNSIT dataset contains ten classes: Digits from 0-9. Each digit is taken asa class. The required libraries are imported which are required for the model torun. The dataset for this model is imported from the Keras module. The data is split into train and test. Using the training dataset, the model is trained and the testing dataset is used to predict the results. Basically, the pixel values range from 0-255. The value of each image is stored is y train. The model is built with convolutional, pooling and dense layers. The created model is then compiled and saved.

7.2 FEATURE-2 WEB APP

HTML, CSS and JavaScript are used to create the web pages for the front end. An html page that takes in image files as input using form and submits to back end is created. A flask app is created using python flask, where it receivesthe image files from the templates, html pages and the prediction operation is done over this image. Later the predicted output is sent to the result page.

8. TESTING

8.1 TEST CASES

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)
HomePage_TC_001	Functional	Home Page	Verify user is able to see the navigation bar on top		1.Enter URL and click go 2.Click on the Recognise button on navigation bar	Move to recognise page	Working as expected	Pass		Y
RecognizePage_TC_002	Functional	RecognizePage	Verify user is able to move to recognise page		1.Enter URL and click go 2.Click on the Recognise button on navigation bar 3.Click on select file button on the view page	1.user should be navigate to our computer image folder.	Working as expected	pass		Y
RecognizePage_TC_003	Functional	Recognizepage	Verify user is moved to predict page.		1.Enter URL and click go 2.Click on the Recognise button on navigation bar 3.Click on select file button on the view page. 4.click on the recognise button.	1.move to predict page.	Working as expected	Pass		N
PredictPage_TC_004	Non-Functional	Predictpage	Verify whether digit is predicted correctly.		1.Enter URL and click go 2.Click on the Recognize button on the navigaor bar 3.click on select file button on the view page 4.Choose on image as you want to predict. 5.Click on recognise button	1.User expects the image to be predicted correctly.	There are incorrect predictions at times	Fail	The accuracy of the system affects the results	N
BackPage_TC_005	Functional	BackPage	In case of incorrect prediction or user wants another image predicted, then user clicks on back button.		1.Enter URL and click go 2.Click on the Recognize button on the navigaor bar 3.click on selct file button on the view page 4.Choose on image as you want to predict. 5.click on recognise Button. 6.click on Back Button.	1.user is moved back to recognise page	Working as expected	Pass		Y

8.2 USER ACCEPTANCE TEST

Defect Analysis

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

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	13	2	1	2	18
Duplicate	4	0	2	0	6
External	3	2	1	0	6
Fixed	12	3	2	17	34
Not Reproduced	0	2	0	0	2
Skipped	0	0	2	1	3
Won't Fix	0	3	4	1	8
Totals	32	12	13	21	77

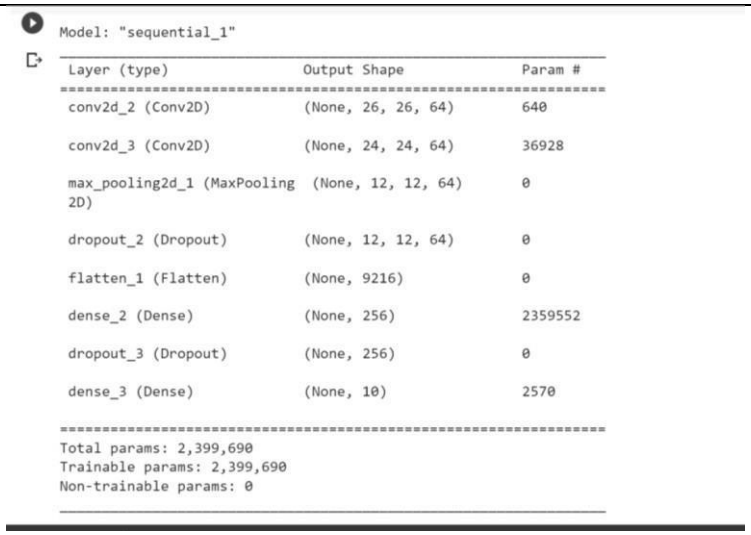
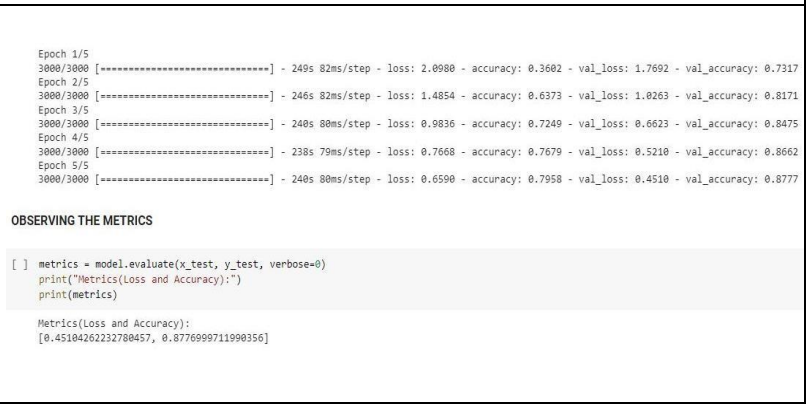
Test Case Analysis

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

Section	Total Cases	Not Tested	Fail	Pass
Client Application	37	0	0	37
Image	14	0	0	14
Prediction	5	0	2	3
Exception Reporting	7	0	0	7
Final Report Output	4	0	0	4
Version Control	2	0	0	2

9. RESULTS

9.1 Performance Metrics

S.No	Parameter	Values	Screenshot
1.	Model Summary		
2.	Accuracy	Training Accuracy –79.58 Validation Accuracy -87.76	

10. ADVANTAGES & DISADVANTEGES

ADVANTAGES:

- It saves times for arranging and sorting huge amount of data
- Only requires far less physical space than the storage of the physicalcopies
- It reduces human effort and Labour cost
- This can be used for sorting through mail by postal code

DISADVANTAGES

- The handwriting of every individual varies which proves to be achallenge for the system to predict
- The system build is complex and holds difficulty
- The accuracy is not guarantees and there are risk of errors
- Possible unemployment of labour that is typical of technology growth

11. CONCLUSION

Handwritten digit recognition has immense applications in the field of medical, banking, student management, and taxation process etc. Many classifiers like KNN, SVM, and CNN are used to identify the digit from the handwritten image. Here we've used CNN for implementation. Convolutional Neural Network gets trained from the real-time data and makes the model verysimple by reducing the number of variables and gives relevant accuracy.
MNIST dataset consist of handwritten numbers from 0-9 and it is a standarddataset used to find performance of classifiers.

Results of HDR is improved a lot by using CNN classifier but it can be improved further in terms of complexity, duration of execution and accuracy ofresults by making combination of classifiers or using some additional algorithmwith it. More accurate results can be established with more convolution layers and more number of hidden neurons. It can completely abolish the need for typing. Digit recognition is an excellent way to develop more advanced techniques of deep learning.

12. FUTURE SCOPE

In future, different architectures of CNN, namely, hybrid CNN, viz., CNN-RNN and CNN-HMM models, and domain-specific recognition systems,can be investigated. Evolutionary algorithms can be explored for optimizing CNN learning parameters, namely, the number of layers, learning rate and kernel sizes of convolutional filters. The future development of the applicationsbased on algorithms of deep and machine learning is practically boundless.

In the future, we can work on a denser or hybrid algorithm than the current set of algorithms with more manifold data to achieve the solutions to many problems. In future, the application of these algorithms lies from the public to high-level authorities, as from the differentiation of the algorithms above and with future development we can attain high-level functioning applications which can be used in the classified or government agencies as wellas for the common people. Currently only the digits are recognized. In future the all the characters in all the language can be predicted with high accuracy rate.

13. APPENDIX

Source Code

The necessary libraries are imported.

```
import keras
import tensorflow as tf
from keras.datasets import mnist
from keras.layers import Dense, Dropout, Flatten
from keras.layers import Conv2D, MaxPooling2D
from keras import backend as K
from tensorflow.keras.utils import to_categorical
import matplotlib.pyplot as plt
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D
from tensorflow.keras.layers import MaxPooling2D
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Flatten
from tensorflow.keras.optimizers import SGD
from keras.utils import np_utils
from keras.utils.np_utils import to_categorical
```

The MNIST dataset is downloaded from the keras library and the data is analyzed.

```
print(x_train.shape,y_train.shape)
print(x_test.shape,y_test.shape)
x_train[0]
```

The data is pre-processed and reshaped

```
num_classes=10
x_train=x_train.reshape(x_train.shape[0],28,28,1)
x_test=x_test.reshape(x_test.shape[0],28,28,1)
input_shape = (28,28,1)
```

Applying one-hot encoding. The class vectors are converted to binary class matrices.

```
y_train=keras.utils.np_utils.to_categorical(y_train,num_classes)
y_test=keras.utils.np_utils.to_categorical(y_test,num_classes)
x_train=x_train.astype('float32')
x_test=x_test.astype('float32')
x_train=x_train/255
x_test=x_test/255
print('x_train shape:',x_train.shape)
print(x_train.shape[0],'train samples')
```

The CNN model is created. The activation function is Rectified linearunit(ReLU). The pooling layers, dense layers are added and flattened.

```
batch_size=128
num_classes=10
epochs=20
model = Sequential()
model.add(Conv2D(32,
kernel_size=(3,3),activation='relu',input_shape=input_shape))
model.add(Conv2D(64,(3,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Dropout(0.25))
model.add(Flatten())
model.add(Dense(256,activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(num_classes,activation='softmax'))
```

The model is then compiled

```
model.compile(loss=keras.losses.categorical_crossentropy,optimizer=tf.keras
.optimizers.Adadelta(),metrics=['accuracy'])
```

The model is trained

```
hist = model.fit(x_train,
y_train,batch_size=20,epochs=5,verbose=1,validation_data=(x_test, y_test))
```

Observing the metrics and testing the model

```
metrics = model.evaluate(x_test, y_test, verbose=0)
print("Metrics(Loss and Accuracy):")
print(metrics)
prediction = model.predict(x_test[:4])
print(prediction)
```

The model is saved and then tested. A sample image is given in to test the savedmodel. The image is reshaped and then predicted

```
model.save('digit_classifier.h5')
from keras.utils.image_utils import img_to_array
from tensorflow.keras.models import load_model
model = load_model('/content/digit_classifier.h5')
from PIL import Image
import numpy as np

img = Image.open('/content/sample.jpg').convert("L")
img = img.resize((28,28))
im2arr = np.array(img)
im2arr = im2arr.reshape(1,28,28,1)


#display the image
import matplotlib.pyplot as plt
plt.imshow(img)


#predict the image
y_predict = model.predict(im2arr)
print(np.argmax(y_predict[0]))
```

The recognize page where the user can upload the image for prediction

recognise.html

```
<html>
    <head>
        <link rel="stylesheet" href="./style/rstyle.css">
        <title>Digit Recognition</title>
        <script src="./script/script.js">
        </script>
        <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="anonymo
        <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz
        <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM
        <script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs@latest"></script>
    </head>
```

```
    <body>
        <div class="details">
            <p style="font-size:30px;font-weight: bolder;">Handwritten Digit Recognition</p>
        <p style="font-size: 20px;">Handwriting recognition is one of the compelling research works going on because every
            individual in this world has their own style of writing.<br><br>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>
        </div>

        <h1 style = "color: rgb(2, 60, 0)">Digit Recognition
        <br>
        <br>

        <form action="/recognise" method="POST" enctype="multipart/form-data">
        <input id="image" type="file" name="image" accept="image/png, image/jpeg" onchange="preview()"><br><br>
        <img id="frame" src="" width="100px" height="100px"/>
        </h1>
        <button type="submit" class="button button2" id="predict_button" onclick="predict.html">Recognise</button>
        </form>
    </body>
</html>
```

The page where the predicted output is displayed

predict.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Prediction</title>
    <link rel="stylesheet" href="./style/style.css">
</head>
<body>
        <p id="pred">PREDICTION : {{num}}</p>
</body>
</html>
```

The flask app.py python code to calculate the prediction value from processingthe image uploaded by the user

app.py

```
import numpy as np
import os
# import tensorflow
from PIL import Image
from flask import Flask, request, render_template, url_for
from werkzeug.utils import secure_filename, redirect
from gevent.pywsgi import WSGIServer
from keras.models import load_model
from keras.preprocessing import image
from flask import send_from_directory
# import tensorflow.compat.v2 as tf

FOLDER =r'C:\Users\Janeswaran\OneDrive\Desktop\img'
app = Flask(__name__)
app.config['UPLOAD_FOLDER'] = FOLDER

model = load_model("digit_classifier.h5")

@app.route('/')
def index():
    return render_template('recognise.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))
        uploading_img = os.path.join(FOLDER, filepath)
        img = Image.open(uploading_img).convert("L")
        img = img.resize((28, 28))
        im2arr = np.array(img)
        im2arr = im2arr.reshape(1, 28, 28, 1)
        predict = model.predict(im2arr)
        num = np.argmax(predict, axis=1)
        return render_template('predict.html', num=str(num[0]))

if __name__ == '__main__':
    app.run(debug=True, threaded=False)
```

GitHub:

https://github.com/IBM-EPBL/IBM-Project-7448-1658857113

Demo Link:

https://clipchamp.com/watch/PDniDY4kdmH

YouTube Link:

https://www.youtube.com/watch?v=pIv9H0KVO4o