K.L.N College of Information

Technology, Pottapalayam

Department of CSE

Sub.Code & Sub.Name: HX 8001 & Professional Readiness for Innovation,Employability and Entrepreneurship

"Project Report"

REAL-TIME COMMUNICATION SYSTEM POWERED BY AI FOR SPECIALLY ABLED

Guided By,

Ms.B.K.Hemalatha,

HoD(i/c)/ECE,

KLNCIT

Submitted by,

K.SuryaPrakash(910719104028)

M.V.Varsha(910719104030)

I.Priya Lakshmi(910719104019)

R.Rajesh(910719104020)

TABLE OF CONTENTS

1. INTRODUCTION :

1.1.Overview :

In our society, we have people with disabilities. The technology is developing day by day but no significant developments are undertaken for the betterment of these people. Communication between deaf-mute and a normal person has always been a challenging task. It is very difficult for mute people to convey their message to normal people. Since normal people are not trained on hand sign language. In emergency times conveying their message is very difficult. The human hand has remained a popular choice to convey information in situations where other forms like speech cannot be used. Hand Gesture Recognition will be very useful to have a proper conversation between a normal person and an impaired person in any language.

1.2.Purpose:

The project aims to develop a system that converts the sign language into a text to convey a message to normal people. We are making use of a convolution neural network to create a model that is trained on different hand gestures. A Web Application is built which uses this model. This application enables deaf and dumb people to convey their information using signs which get converted to human-understandable language.

2. LITERATURE SURVEY:

2.1. Existing problem:

A Specially Abled Person (deaf and dumb) who needs to communication with others easily. Generally these people use sign language for communication, but they find difficulty in communicating with others who don't understand sign language. So there is a barrier in communication between these two communities. Image processing with Hand Gesture Recognition will be very useful to have a proper conversation between a normal person and an impaired person in any language.

2.2. References:

1. Ong Chin Ann, Marlene lu, Bee Theng Lau(January 2011), A Face Based Real Time Communication for Physically and Speech Disabled People.DOI:10.4018/978-1-60960-541-4.ch00

2.Bayan Mohammed Saleh, Reem Ibrahim Al-Beshr ,MuhammadUsman Tariq (September 2020), D-Talk: Sign Language Recognition System for People with Disability using Machine Learning and Image Processing, International Journal ofAdvanced Trends in Computer Science

andEngineering 9(4):4374-4382DOI:10.30534/ijatcse/2020/29942020

3. Boon-Giin Lee, Su Min Lee (February 2018), Smart Wearable Hand Device for Sign Language Interpretation System With Sensors Fusion, IEEE Sensors JournalDOI:10.1109/JSEN.2017.2779466 Corpus ID: 45573954

4. B.K. Sy, J.R. Deller, An AI-based communication system for motor and speech disabled persons: design methodology and prototype testing , IEEE Transactions on BiomedicalEngineering (Volume: 36, Issue: 5, May 1989)DOI: 10.1109/10.24260

5. Yukai Song; William Taylor; Yao Ge; Kia Dashtipour; MuhammadAli Imran; Qammer H. Abbasi, Design and Implementation of a Contactless AI-enabled Human Motion Detection System for Next_Generation Healthcare, 2021 IEEE International Conference on Smart Internet of Things (SmartIoT),DOI: 10.1109/SmartIoT52359.2021.00027

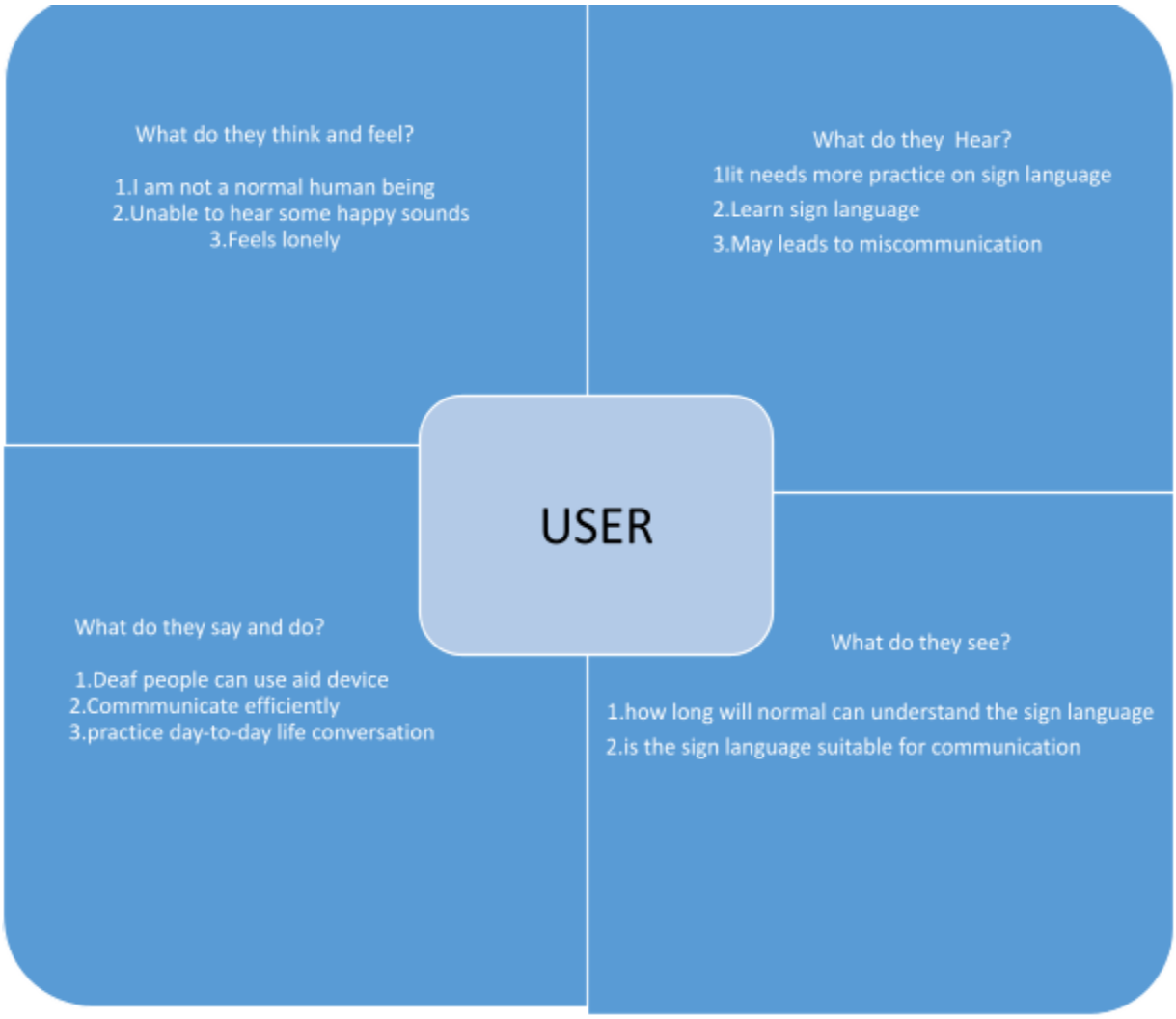
2.3.Definition of problem statement:

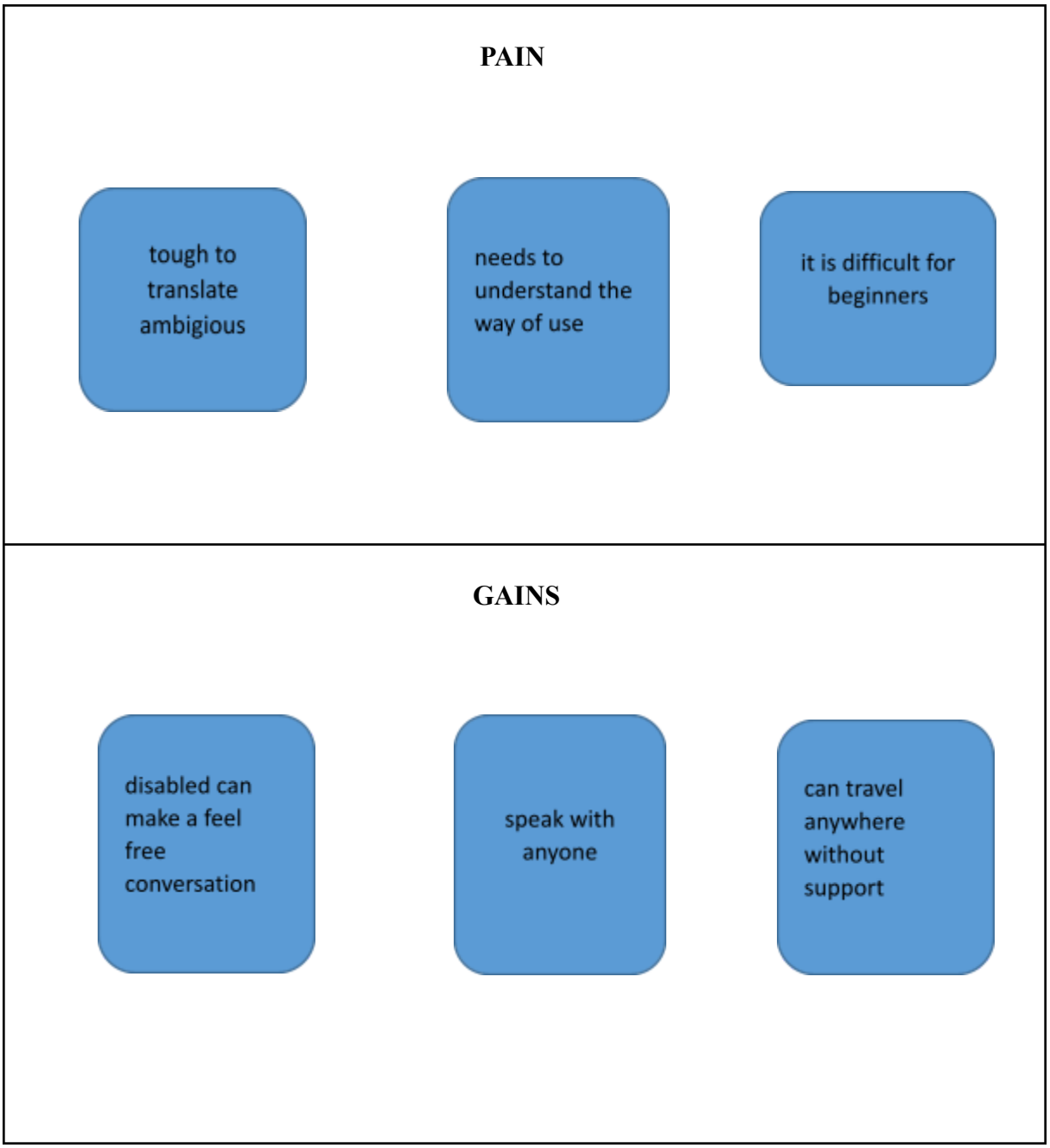
In our society, we have people with disabilities. The technology is developing day by day but no significant developments are undertaken for the betterment of these people. Communication between deaf-mute and a normal person has always been a challenging task. It is very difficult for mute people to convey their message to normal people. Since normal people are not trained on hand sign language. In emergency times conveying their message is very difficult. The human hand has remained a popular choice to convey information in situations where other forms like speech cannot be used.Hand Gesture Recognition and translation will be very useful to have a proper conversation between a normal person and an impaired person in any language.

The project aims to develop a system that converts the sign language into a human understandable texts in the desired language to convey a message to normal people, as well as convert text into understandable sign language for the deaf and dumb. We are making use of a convolution neural network to create a model that is trained on different hand gestures. An app is built which uses this model. This app enables deaf and dumb people to convey their information using signs which get converted to human-understandable language and speech is given as output.

3.IDEATION AND PROPOSED SOLUTION:

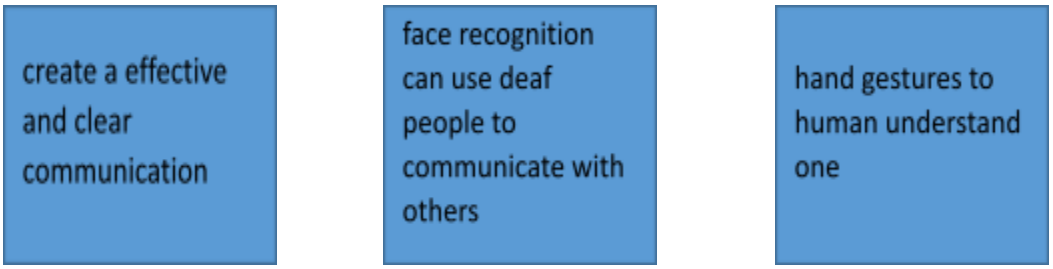
3.1.Empathy map canvas:



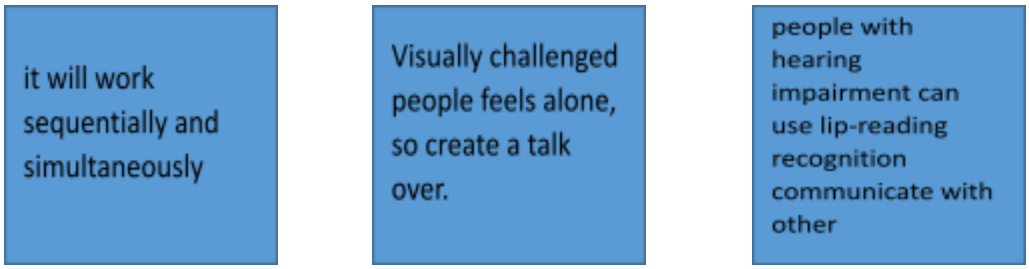

PAIN
tough to translate ambigious
needs to understand the way of use
it is difficult for beginners
GAINS
disabled can make a feel free conversation
speak with anyone
can travel anywhere without support

3.2.Ideation and brainstorming:

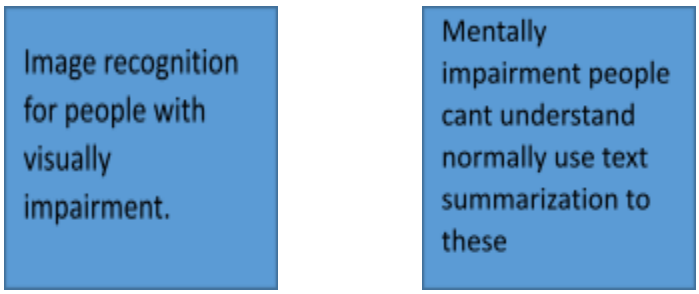
Surya Prakash.K



M.V.Varsha



I.Priya Lakshmi



R.Rajesh

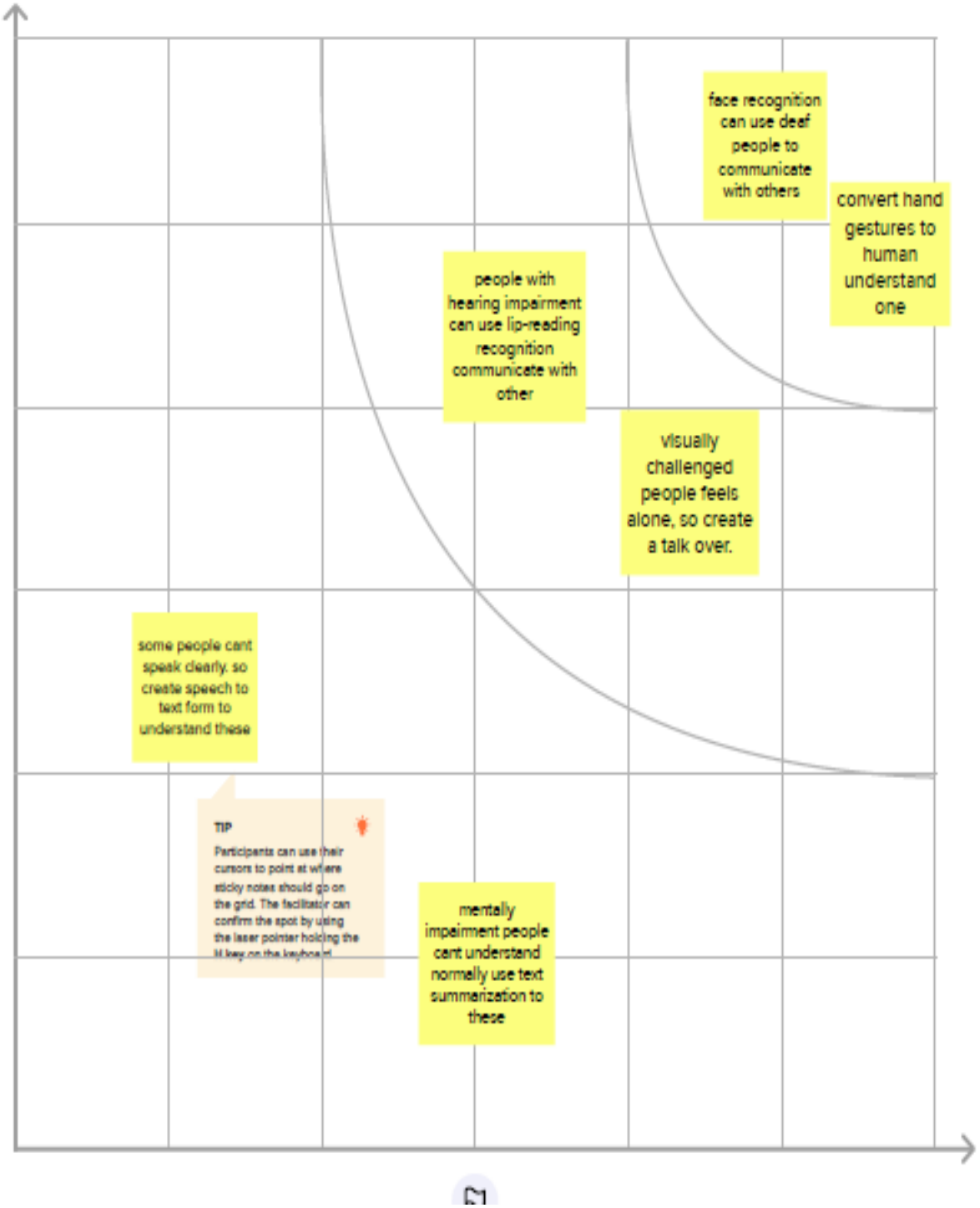
it can use anywhere at anytime they want .

some people cant speak clearly. So create speech to text form to understand these

Group ideas:

Deaf people	Dumb people	Blind people
convert hand gestures to human understand one	convert hand gestures to human understand one	Visually challenged people feels alone, so create a talk over
face recognition can use deaf people to communicate with others	face recognition can use deaf people to communicate with others	Image recognition for people with visually impairment.
people with hearing impairment can use lip-reading recognition communicate with other		

Prioritization:

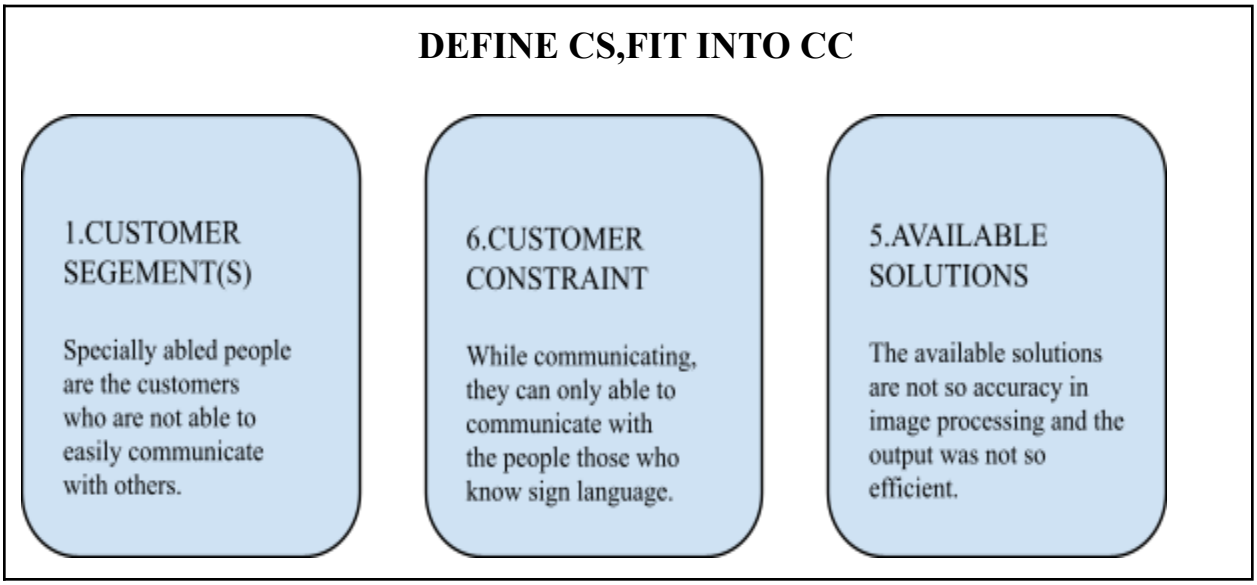


3.3.Proposed solution:

S.No.	Parameter	Description
1.	ProblemStatement(Problem Solved)	Designed an application for Deaf and Dumb people to communicate easier as Normal people Using text Conversion System.
2.	Idea/Solution Description	Real-Time Communication helps User to Communicate easily without their hesitation and Mental distraction by using AI technology.
3.	Novelty/Uniqueness	Process makes clear the two-way communication with the help of AI.It also helps in emerge by quick response
4.	SocialImpact/CustomerSatisfaction	Differently abled people feel free to communicate and it bring a huge Difference comparing to past.So,they Can live their life independently and break the social barrier

5. .	Business Model(Revenue Model)	The device we marketing will be the latest and Portable for the user. This is at free of cost which makes all the user make use of it.
6.	ScalabilityoftheSolution	Thus, this would bring a new evolution in Real Time Communication System Powered byAI for Specially Abled with less time and safe enough resources.

3.4.Problem solution fit :



FOCUS ON J&P ,TAP INTO BE,UNDERSTAND RC

2.JOBS-TO-BE-DONE/PROBLEMS

Only sign language known people can communicate so we introduced a new system to communicate all specially abled people.

9.PROBLEM ROOT CAUSE

Due to the inability to communicate with others by the specially abled people's

7.BEHAVIOUR

Finding the right signs and converting into correct communication between the people's

IDENTIFY STRONG ER TM

3.TRIGGERS

Some of the triggers are introducing in all hospitals, medical trusts and also in advertisements.

10.YOUR SELECTION

Created an application using AI , that will able to convert the sign language by image processing of the specially abled people.

8. CHANNELS of BEHAVIOUR

8.1 ONLINE

We can update our application and use it in a very efficient way.

8.2 OFFLINE

In offline mode we use it but not so efficient we can use it with a recently updated application.

4.EMOTIONS:BEFORE/ AFTER

specially abled people hesitate to communicate with others but know using this system they can easily communicate with others.

4.REQUIREMENT ANALYSIS

4.1.Non-Functional Requirements:

Usability- Non-functional requirements are the constraints of the software.The requirements deal with issues like scalability, maintainability,performance, portability, security, reliability and a lot.

Security-It provides cyber security systems with up-to-date and relevant knowledge of Industry specific and global threats, which help teams make critical decisions.

Reliability-AI technology can empower people living with limited physical mobility. Microsoft's AI for Accessibility program uses the potential of Artificial Intelligence to develop solutions to many physical and cognitive challenges disabled individuals face at work and in daily life to promote social inclusion for them.

Performance - AI enables people with disabilities to step into a world where their difficulties are understood and taken into account.Technology adapts and helps transform the world into an inclusive place with artificial intelligence accessibility.

Availability-Using driverless cars enables disabled people to leave the house, get around their communities, interact with people and even find jobs. Once autonomous vehicles are fully integrated into society, they could ease independent mobility, and increase accessibility adapted to each user's abilities and needs.

Scalability-Scalability is a non-functional property of a system that describes the ability to appropriately handle increasing workloads.

4.2.Functional Requirements:

User registration:

Registration through email .Registration through mobile no

User confirmation:

Confirmation via email.Confirmation via otp

Data collection:

Collecting data for building our project .Creating two folders one for training and the other for Testing.Images present in the training folder will be used for building the model and the testing images will be used for validating our model.

Model building:

Initializing the model.Adding convolution layers.Adding pooling layers.Fully connection layers which includes hidden layer.Flatten layer.Compile the model with layers we added to complete the neural network structure

Test the model:

Test the model by passing an image to get predictions.Make sure that the dimensions,rescaling,target size are correct while testing the model.

Train the model:

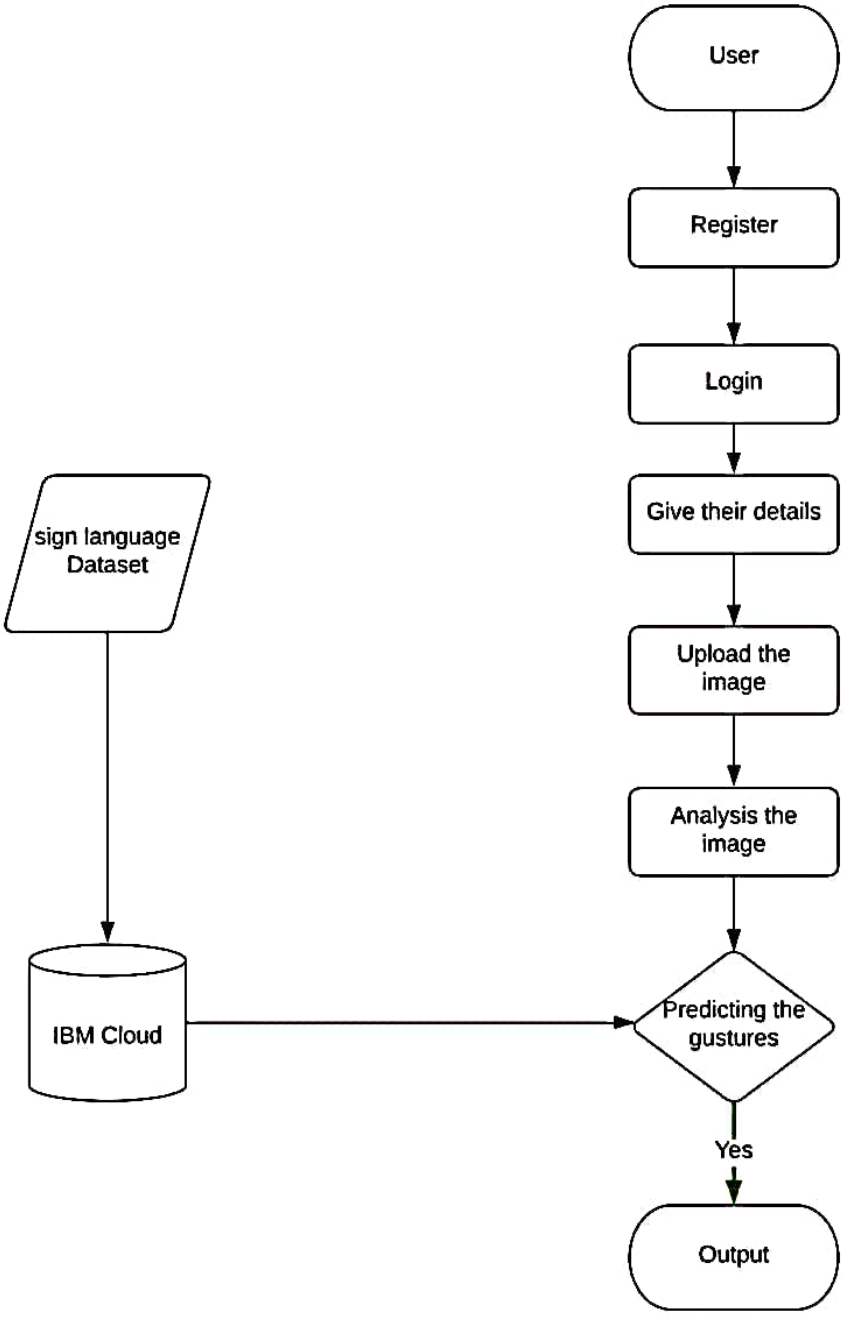
You can also train your Image classification Models on IBM Cloud.Train the model .Store the Model .Download the Stored model to the Localsystem

Train image classification model :

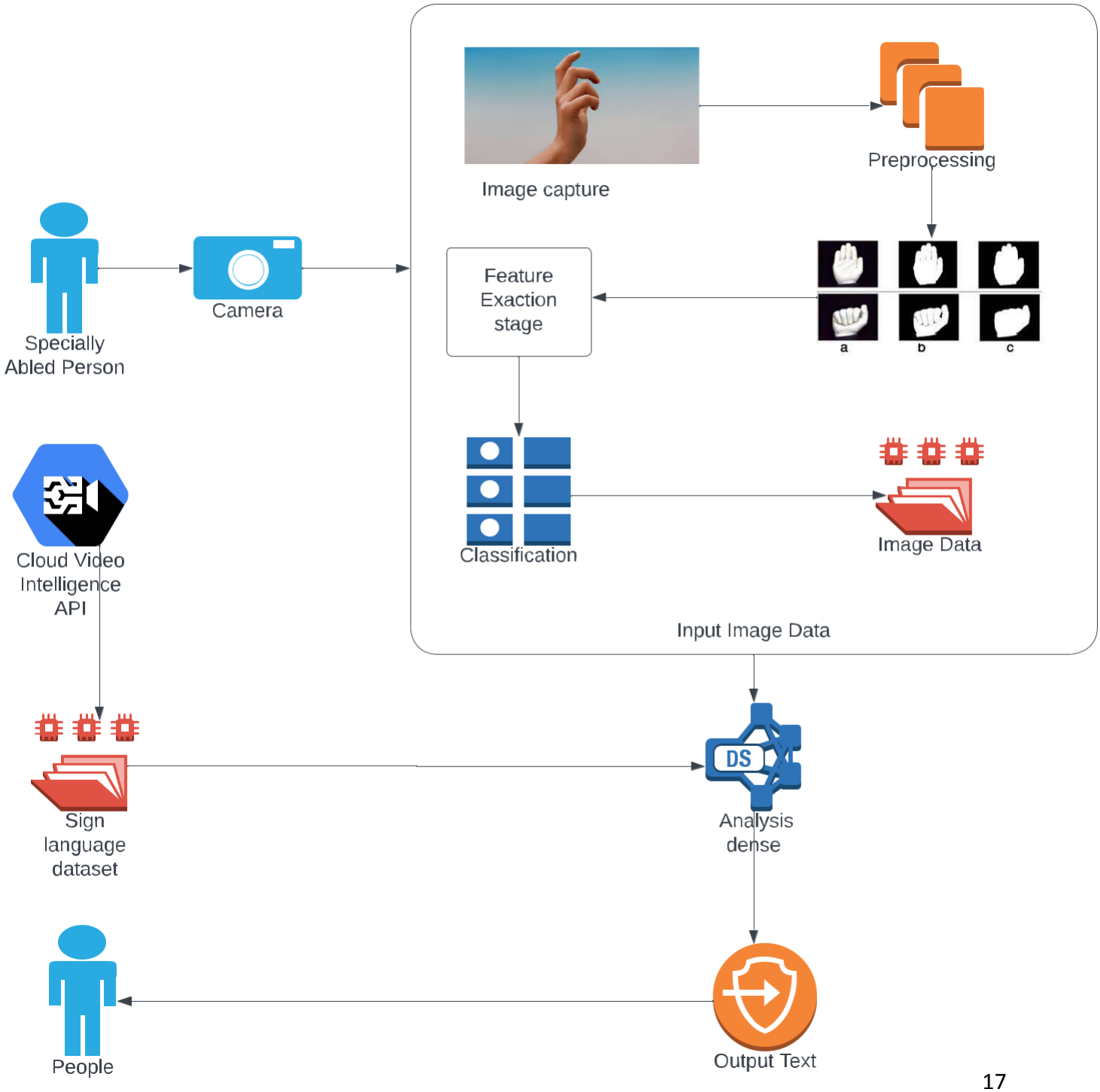
Train the model on IBM.Store the Model.Download the model to local system.Test the model locally.

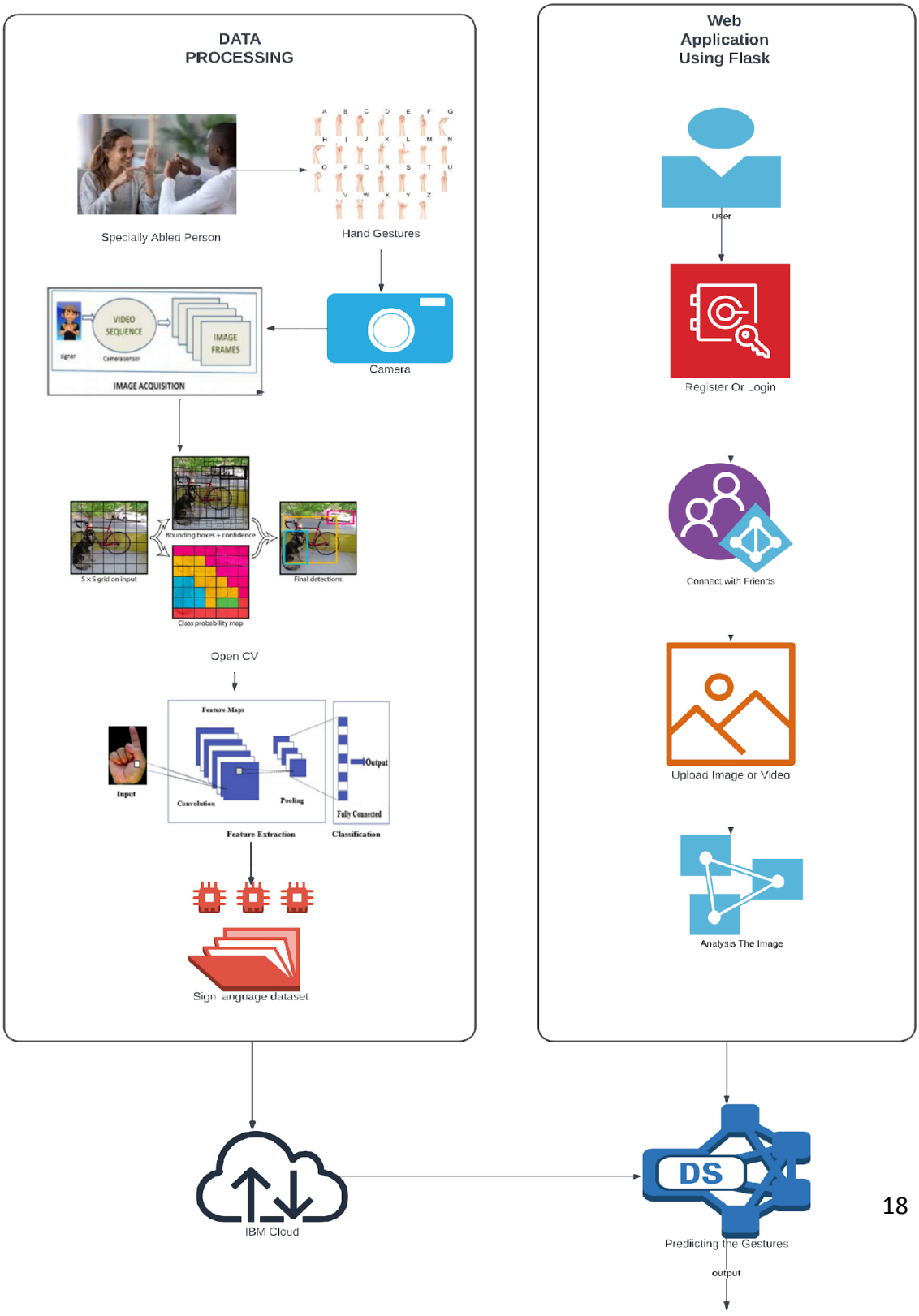
5.PROJECT DESIGN:

5.1 Data Flow Diagrams:



5.2 Solution & Technical Architecture :




DATA
PROCESSING
Specially Abled Person
Hand Gestures
VIDEO
SEQUENCE
IMAGE
FRAMES
IMAGE ACQUISITION
Camera
Bounding boxes + confidence
Class probability map
Final detections
Open CV
Feature Maps
Input
Convolution
Pooling
Output
Fully Connected
Feature Extraction
Classification
Sign language dataset
Web
Application
Using Flask
User
Register Or Login
Upload Image or Video
Analysis The Image
IBM Cloud
DS
Predicting the Gestures
output

User Stories

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

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web 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-2
		USN-3	As a user, I can register for the application through Gmail		Medium	Sprint-2
	Login	USN-4	As a user, I can log into the application by entering email & password		High	Sprint-1
	Dashboard	USN-5	As a user, display all information on the site.		High	Sprint-1
		USN-6	As a user, give their details for future use.		Medium	Sprint-2
	Recognition	USN-7	As a user, record a video or image and upload it..		High	Sprint-3
		USN-8	As a user, view the results of analysis and communicate to others.		High	Sprint-3
Customer Care Executive			create bot chat and collect the queries.		low	Sprint-4
			get feedback from customers.		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	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Priya Lakshmi I
Sprint-1	Login	USN-2	As a user, I can log into the application by entering email & password	2	High	Rajesh R
Sprint-1	Dashboard	USN-3	As a user, display all information of site.	3	High	Varsha MV
Sprint-2		USN-4	As a user, I will receive confirmation email once I have registered for the application	3	High	Rajesh R
Sprint-2		USN-5	As a user, give their details for future use.	3	Medium	Varsha MV
Sprint-3	Recognition	USN-6	As a user, record a video or image and upload it.	2	High	Surya Prakash K
Sprint-3		USN-7	As a user, view the results of analysis and communicate to others.	5	High	Varsha MV
Sprint-4	Customer Care		create bot chat and collect the queries.	5	Low	Surya Prakash K
Sprint-4			get feedback from customers.	2	Medium	Priya Lakshmi I

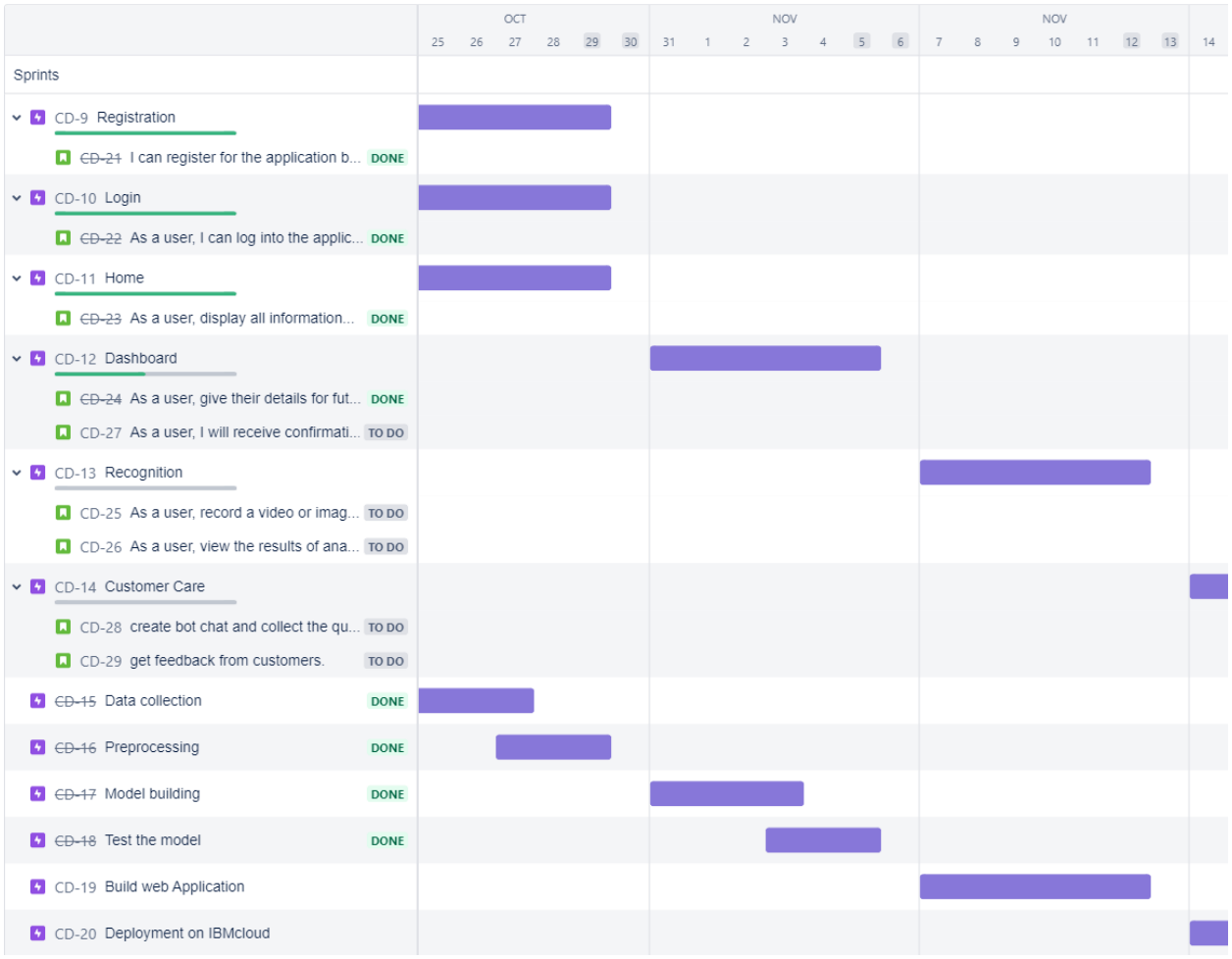
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Data collection		1.Collecting data for building our project . 2.Creating two folders one for training and the other for testing.	2	High	Rajesh R
Sprint-1	Preprocessing		Preprocessing the Dataset	3	High	Varsha MV
Sprint-2	Model building		1.Initializing the model 2.Adding convolution layers 3.adding pooling layers 4.Full connection layers which includes hidden layers 5.Flatten layer 6.Compile the model with layers we added to complete the neural network structure	3	High	Surya Prakash K
Sprint-2	Test the model		Test the model by passing an image to get predictions. Make sure that the dimensions,rescaling, target size are correct while testing the model	3	High	Priya Lakshmi I
Sprint-3	Build web Application		Create a web application for recognition using Flask.	5	High	Surya Prakash K , Varsha MV, Priya Lakshmi I ,Rajesh.R

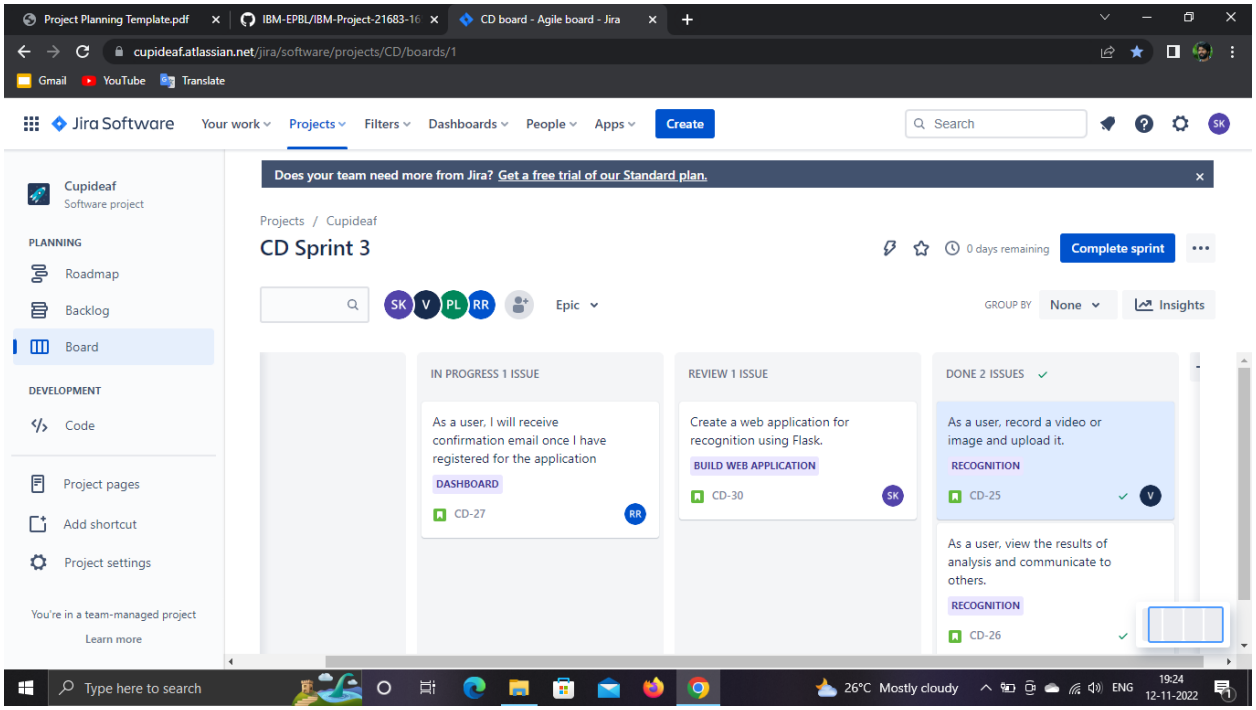
Sprint-4	Deployment on IBMcloud		Integrating the Model which trains and Web application are deployed on cloud.	5	High	Surya Prakash K , Varsha MV, Priya Lakshmi I , Rajesh R

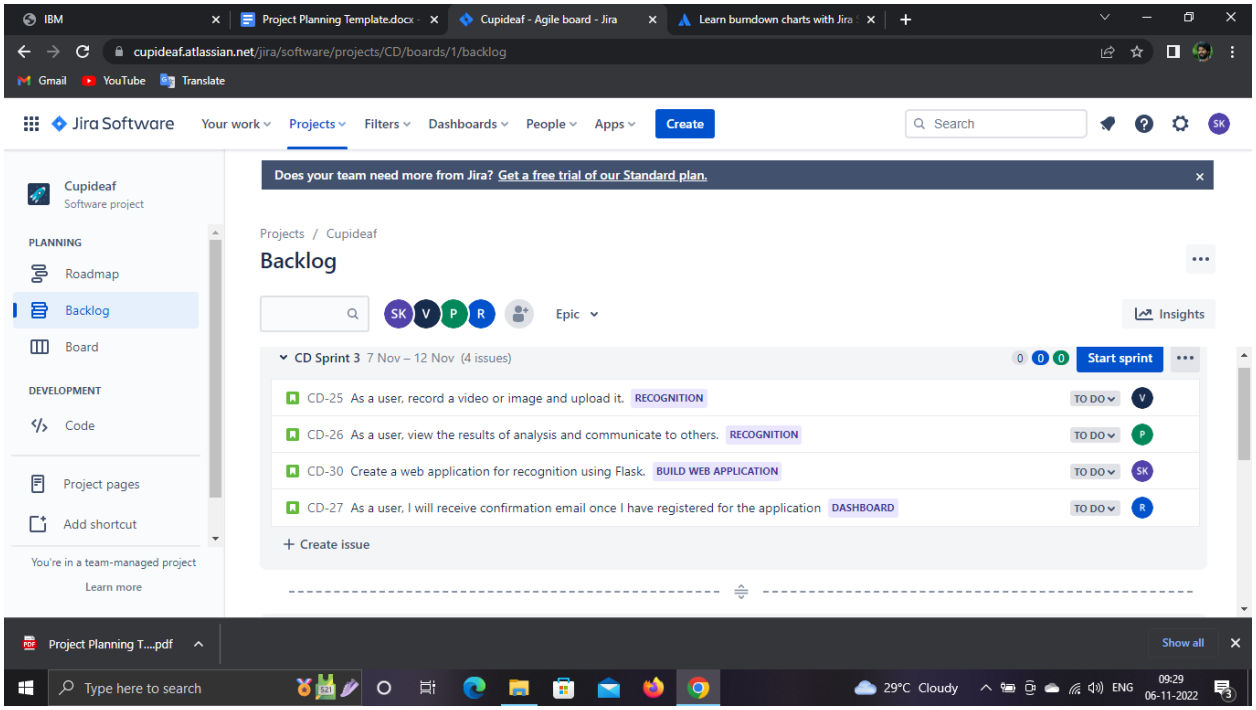
6.2 Sprint Delivery Schedule:

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	12	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	12	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	12	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	12	19 Nov 2022

6.3 Reports from JIRA:




Project Planning Template.pdf
IBM-EPBL/IBM-Project-21683-16
CD board - Agile board - Jira
cupideaf.atlassian.net/jira/software/projects/CD/boards/1
Gmail
YouTube
Translate
Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
Cupideaf
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / Cupideaf
CD Sprint 3
0 days remaining
Complete sprint
Epic
GROUP BY
None
Insights
IN PROGRESS 1 ISSUE
As a user, I will receive confirmation email once I have registered for the application
DASHBOARD
CD-27
REVIEW 1 ISSUE
Create a web application for recognition using Flask.
BUILD WEB APPLICATION
CD-30
DONE 2 ISSUES
As a user, record a video or image and upload it.
RECOGNITION
CD-25
As a user, view the results of analysis and communicate to others.
RECOGNITION
CD-26
Type here to search
26°C Mostly cloudy
ENG
19:24
12-11-2022


IBM
Project Planning Template.docx
Cupideaf - Agile board - Jira
Learn burndown charts with Jira
cupideaf.atlassian.net/jira/software/projects/CD/boards/1/backlog
Gmail
YouTube
Translate
Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
Cupideaf
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
You're in a team-managed project
Learn more
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / Cupideaf
Backlog
Epic
Insights
CD Sprint 3 7 Nov – 12 Nov (4 issues)
Start sprint
CD-25 As a user, record a video or image and upload it. RECOGNITION
CD-26 As a user, view the results of analysis and communicate to others. RECOGNITION
CD-30 Create a web application for recognition using Flask. BUILD WEB APPLICATION
CD-27 As a user, I will receive confirmation email once I have registered for the application DASHBOARD
TO DO
Create issue
Project Planning T....pdf
Show all
Type here to search
29°C Cloudy
ENG
09:29
06-11-2022

7. CODING & SOLUTIONING:

7.1.Feature 1:

Depending on the features given to the classifier, it accumulates a knowledge base for classification purposes.It identifies the edges and after analysing it reflects the hand gesture's corresponding words. It would in fact be irrelevant if all the patterns are shuffled in the same manner and presented to the classifier for classification purposes. If such information is given to the classifier, its performance can be improved. We are doing image processing and predicting the model displaying words.

MODEL BUILDING

```
from keras.models import
Sequential from keras.layers import
Dense from keras.layers import
Convolution2D from tensorflow.keras.layers import
Conv2D, MaxPooling2D from keras.layers import
Dropout from keras.layers import Flatten
#Creating the model
model=Sequential()
#Adding the layers
model.add(Convolution2D(32,(3,3), input_shape=(64,64,1), activation = 'relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())
#adding hidden layers
model.add(Dense(400, activation='relu'))
model.add(Dense(200, activation='relu'))
model.add(Dense(100, activation='relu'))
#Adding the output layer
```

```
model.add(Dense(9, activation='softmax'))
```

In [102]:

```
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
```

In [157]:

```
model.fit_generator(x_train, steps_per_epoch=30, epochs=10,
validation_data=x_test,validation_steps=50)
```

```
Epoch 1/10
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: UserWarning:
`Model.fit_generator` is deprecated and will be removed in a future version. Please use
`Model.fit`, which supports generators.
"""Entry point for launching an IPython kernel.
30/30 [==============================] - ETA: 0s - loss: 0.0083 - accuracy: 0.9957
WARNING:tensorflow:Your input ran out of data; interrupting training. Make sure that your
dataset or generator can generate at least `steps_per_epoch * epochs` batches (in this case, 50
batches). You may need to use the repeat() function when building your dataset.
30/30 [==============================] - 18s 587ms/step - loss: 0.0083 - accuracy:
0.9957 - val_loss: 0.2910 - val_accuracy: 0.9693
Epoch 2/10
30/30 [==============================] - 12s 402ms/step - loss: 0.0081 - accuracy:
0.9980
Epoch 3/10
30/30 [==============================] - 12s 400ms/step - loss: 0.0102 - accuracy:
0.9963
Epoch 4/10
```

30/30 [==============================] - 12s 402ms/step - loss: 0.0049 - accuracy:

0.9993

Epoch 5/10

30/30 [==============================] - 12s 402ms/step - loss: 0.0030 - accuracy:

0.9997

Epoch 6/10

30/30 [==============================] - 12s 394ms/step - loss: 0.0019 - accuracy:

0.9997

Epoch 7/10

30/30 [==============================] - 12s 401ms/step - loss: 0.0081 - accuracy:

0.9973

Epoch 8/10

30/30 [==============================] - 12s 402ms/step - loss: 0.0124 - accuracy:

0.9960

Epoch 9/10

30/30 [==============================] - 12s 401ms/step - loss: 0.0070 - accuracy:

0.9987

Epoch 10/10

30/30 [==============================] - 12s 399 ms/step - loss: 0.0089 - accuracy:

0.9973

model.save('Real_time.h5')

7.2 Feature 2:

From previous, We are doing image processing and predicting the model displaying words.The extra feature added here is the elaboration of dataset and we converted the already trained dataset into some essential words used by the deaf mute people for better prediction lively using convolutional neural networks.

TEST THE MODEL

```
from tensorflow.keras.models import load_model

from tensorflow.keras.preprocessing import image

import numpy as np

import cv2

In [105]:

model = load_model('/content/Real_time.h5')

In [151]:

img = image.load_img('/content/Dataset/test_set/H/107.png',target_size = (100,100))img

from skimage.transform import resize

def detect(frame):

img=image.img_to_array(frame)

img = resize(img,(64,64,1))

img = np.expand_dims(img,axis=0)

pred=np.argmax(model.predict(img))

op=['LIVE LONG,THUMBS DOWN,THUMBS UP,ROCK,STOP,OKAY,CALL ME,FIST']

print("THE PREDICTED WORD IS ",op[pred])

In [150]:

img=image.load_img("/content/Dataset/test_set/H/107.png")

detect(img)
```

1/1 [==============================] - 0s 28ms/step

THE PREDICTED WORD IS LIVE LONG

In [155]:

```
img = image.load_img('/content/Dataset/test_set/A/110.png')
pred=detect(img)
```

1/1 [==============================] - 0s 26ms/step

THE PREDICTED WORD IS THUMBS DOWN

In [158]:

```
img=image.load_img('/content/Dataset/test_set/E/111.png')
detect(img)
```

1/1 [==============================] - 0s 30ms/step

THE PREDICTED WORD IS THUMBS UP

7.3 Database Schema :

Explore the data:

Plot the images to better understand the kind of data we are working with

In [71]:
```
import numpy as np
from keras.models import load_model
import cv2
```

In [72]:
```
model = load_model('/content/aslpng1.h5')
```

In [75]:
```
from skimage.transform import resize
def detect(frame):
  img = resize(frame,(64,64,1))
  img = np.expand_dims(img,axis=0)
  if(np.max(img)>1):
    img = img/255.0
  pred = np.argmax(model.predict(img))
  prediction = model.predict(img)
  print(prediction)
  print(pred)
```

Pre-Process data:

We read the training and validation dataset using Image Data Generator

In [56]:
```
# Importing req. lib

from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

In [57]:
```
# Initializing data augmentation to training variable

train_datagen = ImageDataGenerator(rescale=1./255,
                                   shear_range=0.2,
                                   zoom_range=0.2,
                                   horizontal_flip=True)
```

In [58]:
```
# Initializing data augmentation to testing variable

test_datagen = ImageDataGenerator(rescale=1./255)
```

Data Augmentation:

In [59]:
```
# Data augmentation on training data

x_train = train_datagen.flow_from_directory('Dataset/training_set',
                                            target_size=(64,64),
                                            class_mode='categorical',
                                            batch_size=300,
                                             color_mode = "grayscale")
```

```
Found 15750 images belonging to 9 classes.
```

In [60]:
```
#Data augmentation on testing

x_test = test_datagen.flow_from_directory('Dataset/test_set',
                                            target_size=(64,64),
                                            class_mode='categorical',
                                            batch_size=300,
                                             color_mode = "grayscale")
```

```
Found 2250 images belonging to 9 classes.
```

8. TESTING:

8.1 Test Cases:

- ☐ Verify if user can see the options when user clicks the URL
- ☐ Verify if the UI elements are getting displayed properly
- ☐ Verify if the user can choose any languages
- ☐ Verify if the user is getting redirected to the sign to text page
- ☐ Verify if the web application can convert the sign to relevant word text
- ☐ Verify if the user can exit the sign to text page
- ☐ Verify if the user is getting redirected to the speech to sign page
- ☐ Verify if the application can convert sign language to text through image processing to text on detecting the hand gestures to text button.

8.2 User Acceptance Testing:

Defect Analysis:

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

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	11	7	4	2	24
Duplicate	1	0	2	0	3
External	2	3	2	1	8
Fixed	10	5	3	14	32
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	1	0	0	0	1
Totals	25	15	13	18	71

Test Case Analysis:

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

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	15	0	0	15
Security	2	0	0	2
Outsource Shipping	2	0	0	2
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

9. RESULTS:

OpenCV:

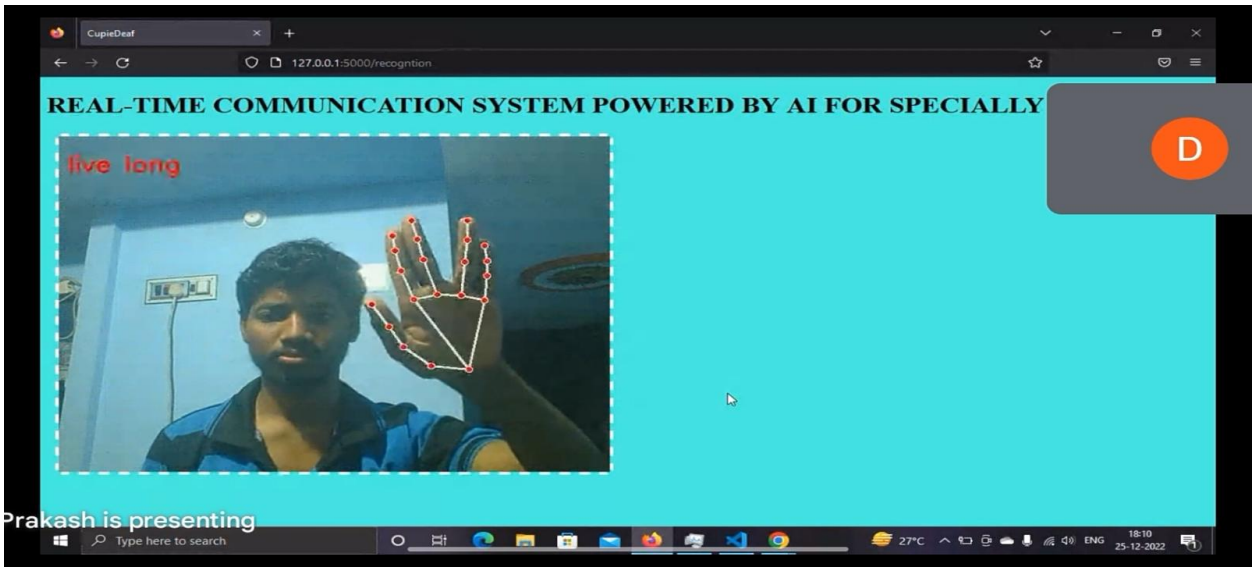
OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being an Apache 2 licensed product, OpenCV makes it easy for businesses to utilize and modify the code.

The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces, identify objects, classify human actions in videos, track camera movements, track moving objects, extract 3D models of objects, produce 3D point clouds from stereo cameras, stitch images together to produce a high resolution image of an entire scene, find similar images from an image database, remove red eyes from images taken using flash, follow eye movements, recognize scenery and establish markers to overlay it with augmented reality, etc. OpenCV has more than 47 thousand users and an estimated number of downloads exceeding in millions. The library is used extensively in companies, research groups and by governmental bodies.

Output snapshot:

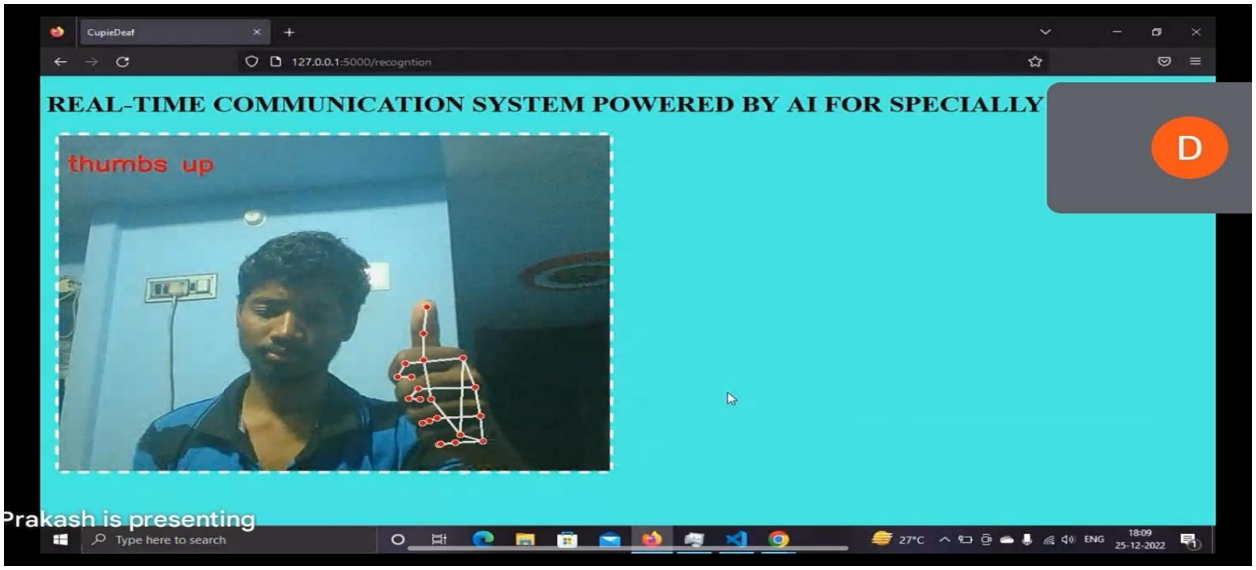
OUTPUT SNAPSHOT:1

LIVE LONG



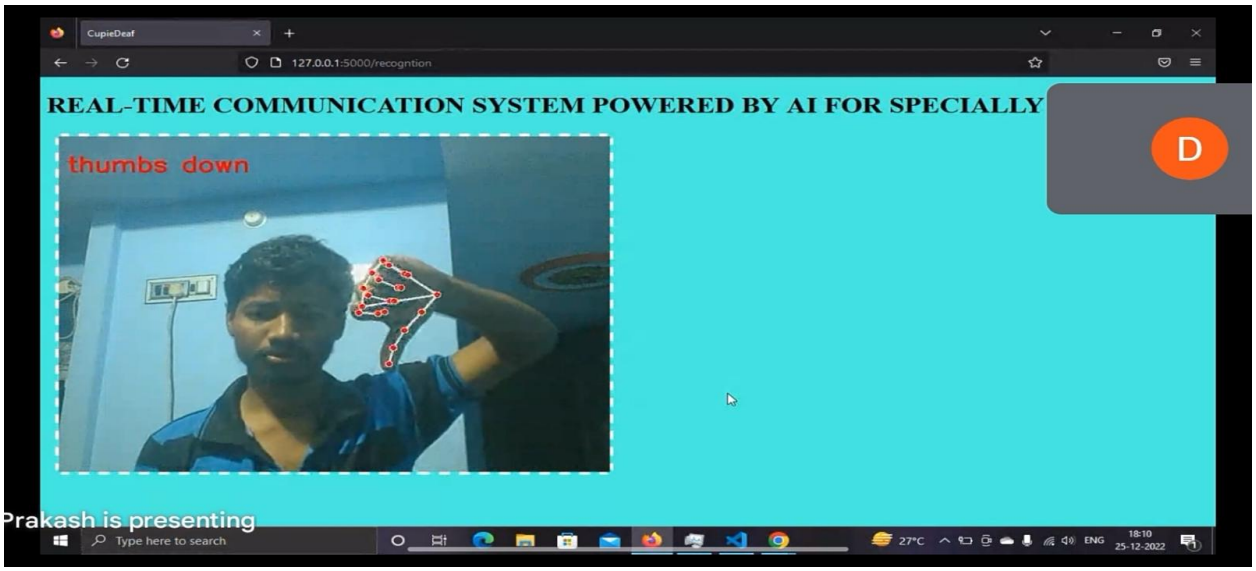
OUTPUT SNAPSHOT:2

THUMBS UP



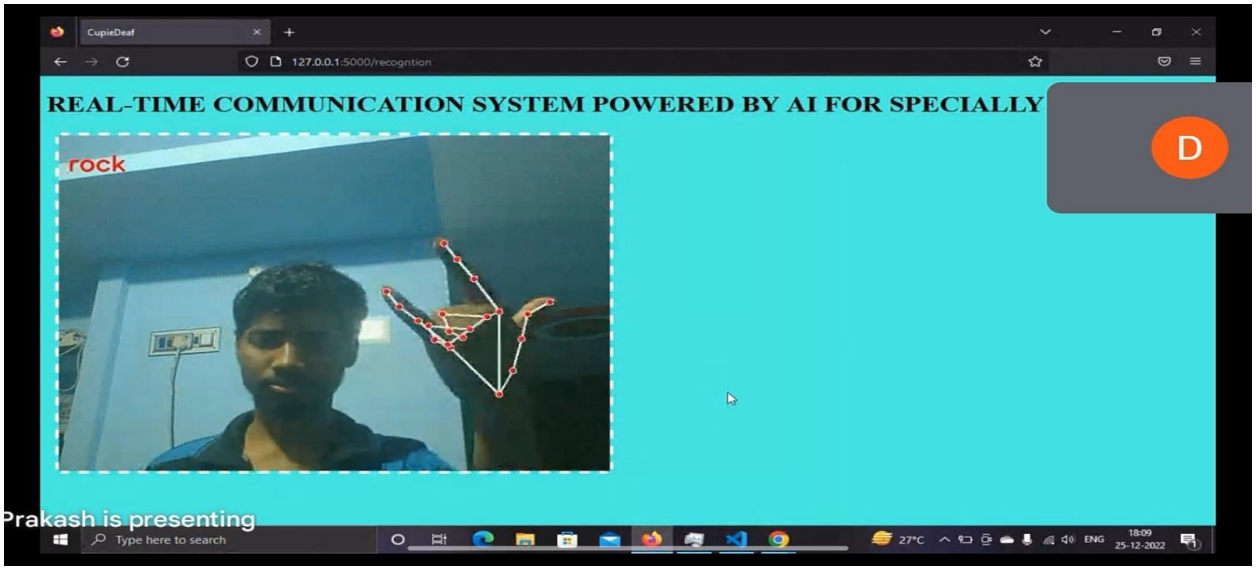
OUTPUT SNAPSHOT:3

THUMBS DOWN



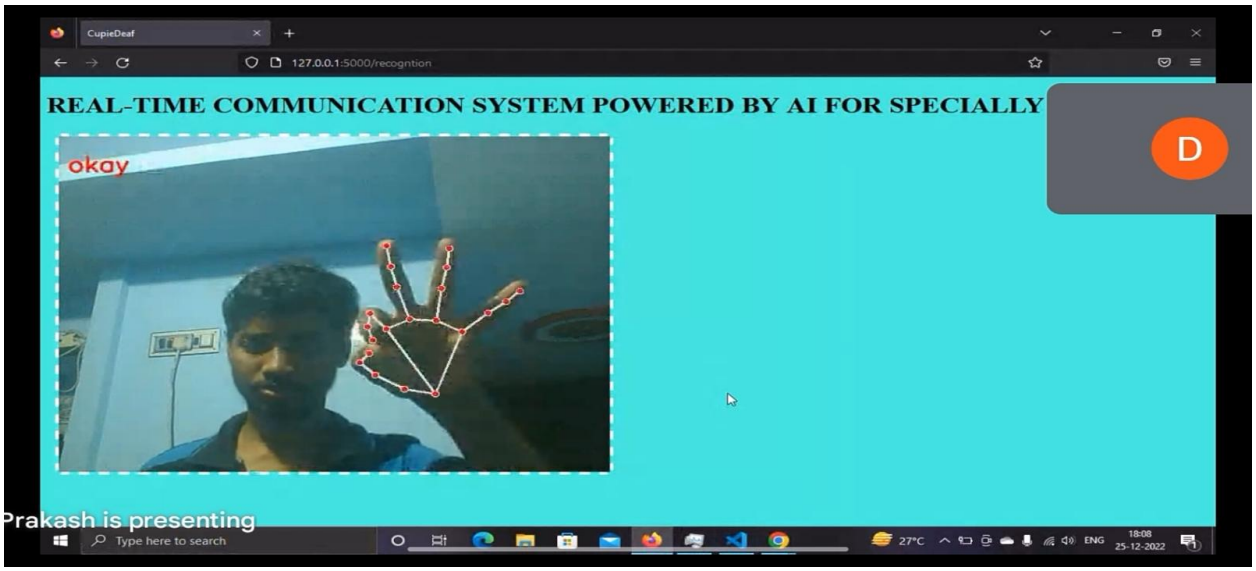
OUTPUT SNAPSHOT:4

ROCK



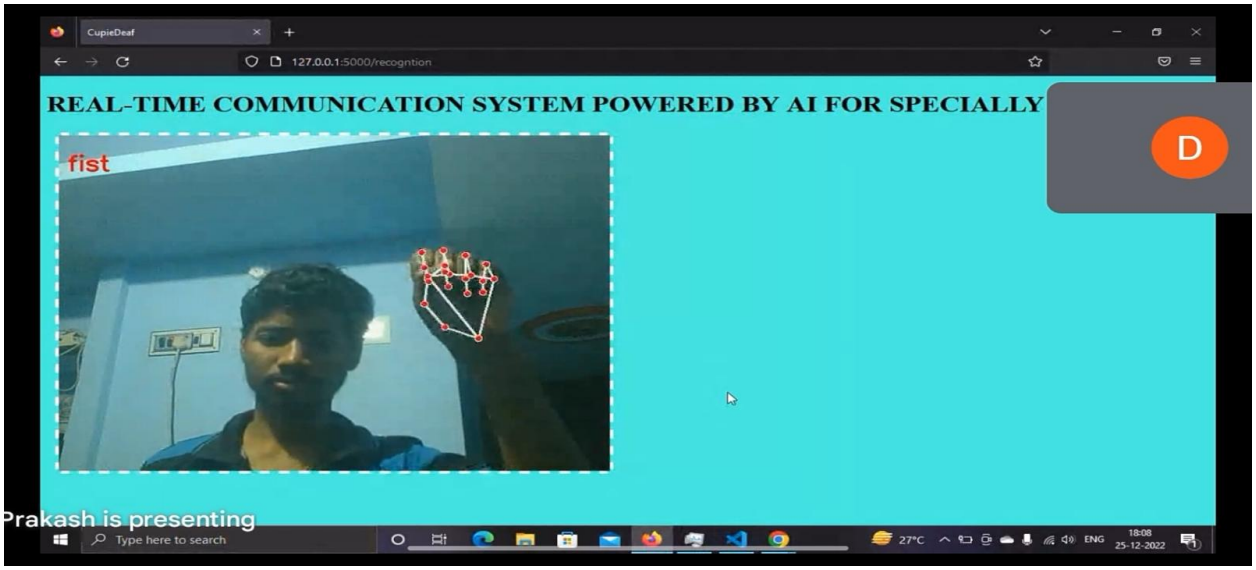
OUTPUT SNAPSHOT:5

OKAY



OUTPUT SNAPSHOT:6

FIST

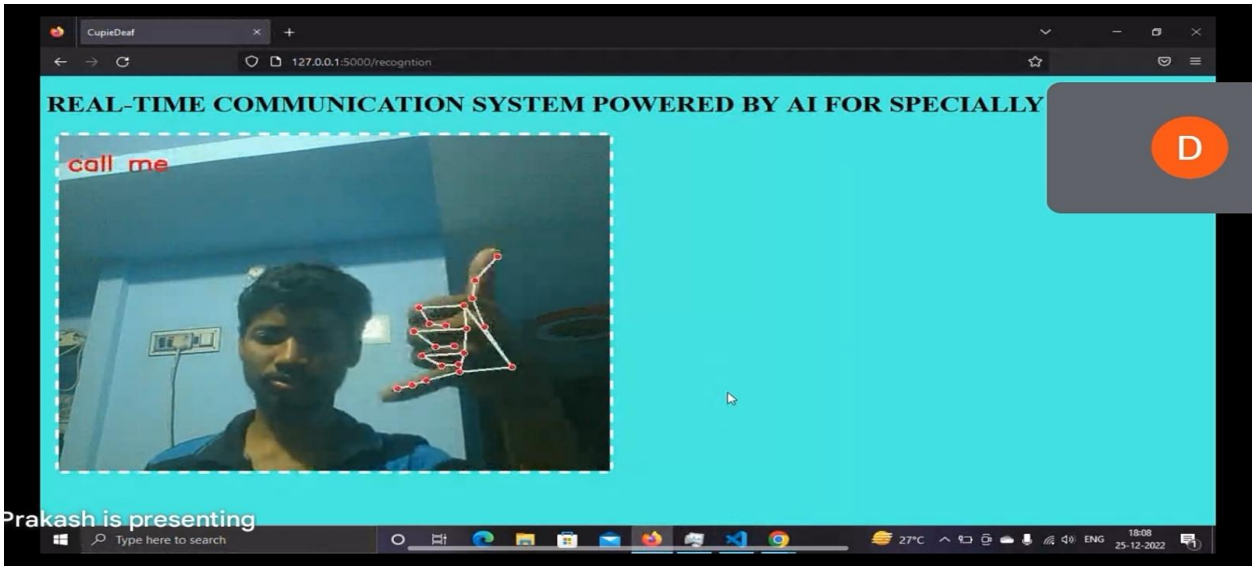


OUTPUT SNAPSHOT:7

STOP

OUTPUT SNAPSHOT:8

CALL ME

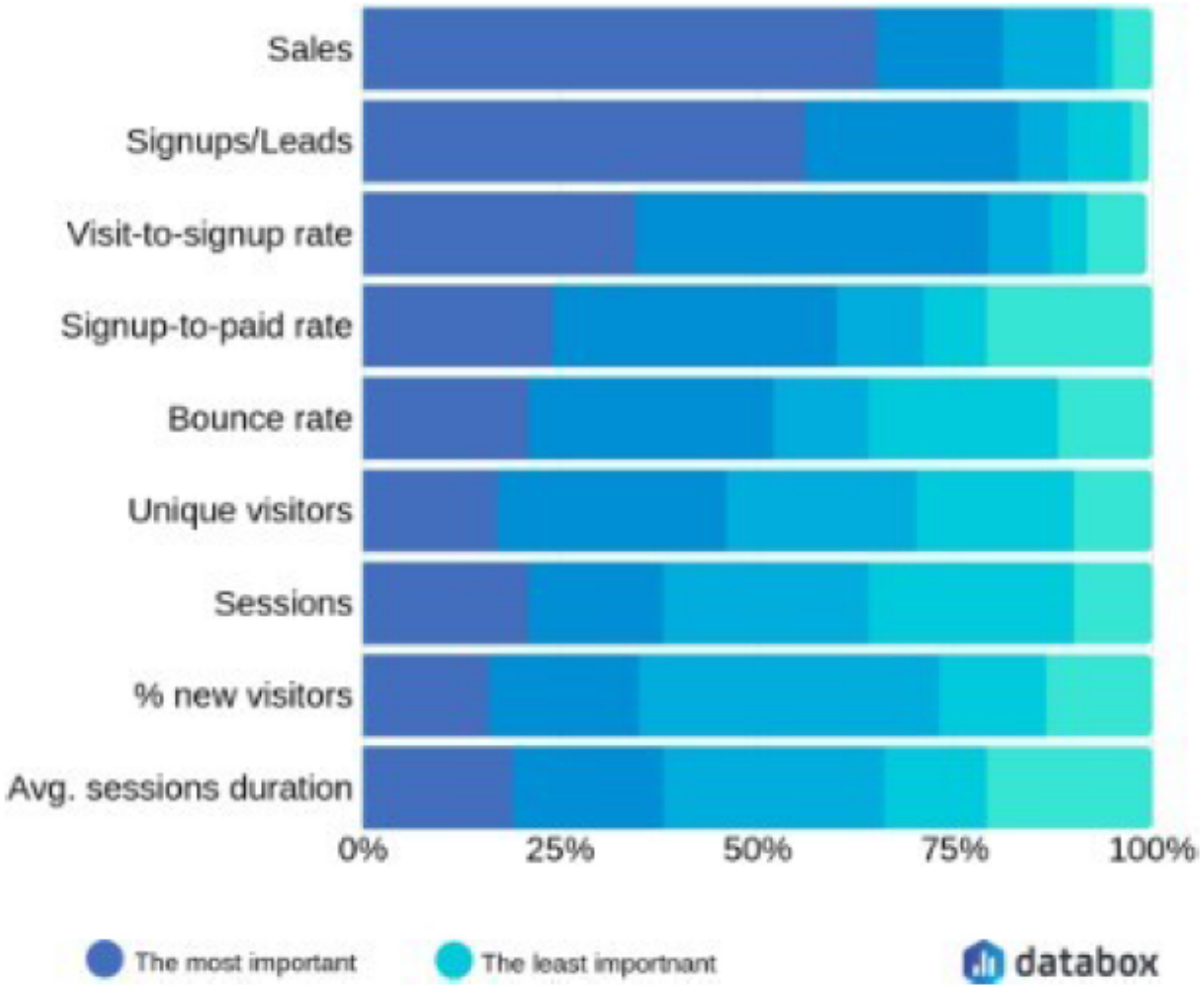


9.1 Performance Metrics:

The proposed procedure was implemented and tested .

● The training database consists of images of some words, while the testing database consists of images of same words.

● Once the gesture is recognized the equivalent words is shown on the screen.



10. ADVANTAGES & DISADVANTAGES:

Advantages:

1. It is possible to create a web application to bridge the communication gap between deaf and dumb persons and the general public.

2. As different sign language standards exist, their dataset can be added, and the user can choose which sign language to read.

Disadvantages:

1.The current model only works for a few words.

2.As the quantity/quality of images in the dataset is low, the accuracy is not great, but that can easily be improved by change in dataset.

11. CONCLUSION:

Sign language is a useful tool for facilitating communication between deaf and dumb people. Because it allows for two-way communication, the system aims to bridge the communication gap between deaf people and the rest of society. The proposed methodology predicts sign language and figures out the words that are understandable to humans. This system sends hand gestures to the model, who recognises them and displays the equivalent words on the screen. Deaf-mute people can use their hands to perform sign language, which will then be converted into commonly used wordings, thanks to this project.

12. FUTURE SCOPE:

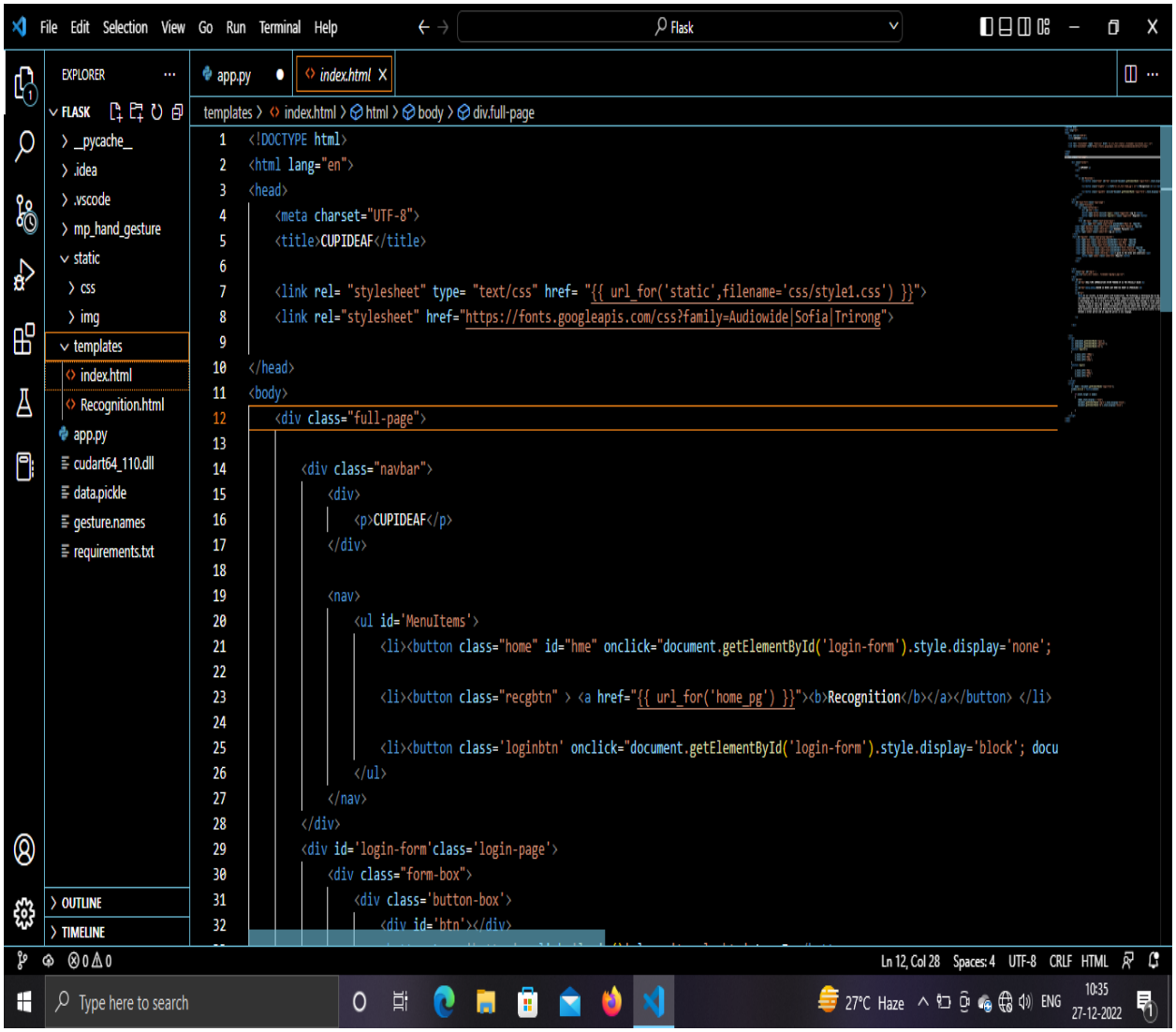
Having a technology that can translate hand sign language to its corresponding word is a game changer in the field of communication and Ai for the specially abled people such as deaf and dumb. With the introduction of gesture recognition, the web application can easily be expanded to recognize words more than this, digits and other symbols plus gesture recognition can also allow controlling of software/hardware interfaces. As the quantity and quality of images in the dataset is low, the accuracy is not great, but that can easily be improved by change in dataset.If it is done quite clearly with all efficient words then it would reach its goal in communication efficiently.

13. APPENDIX:

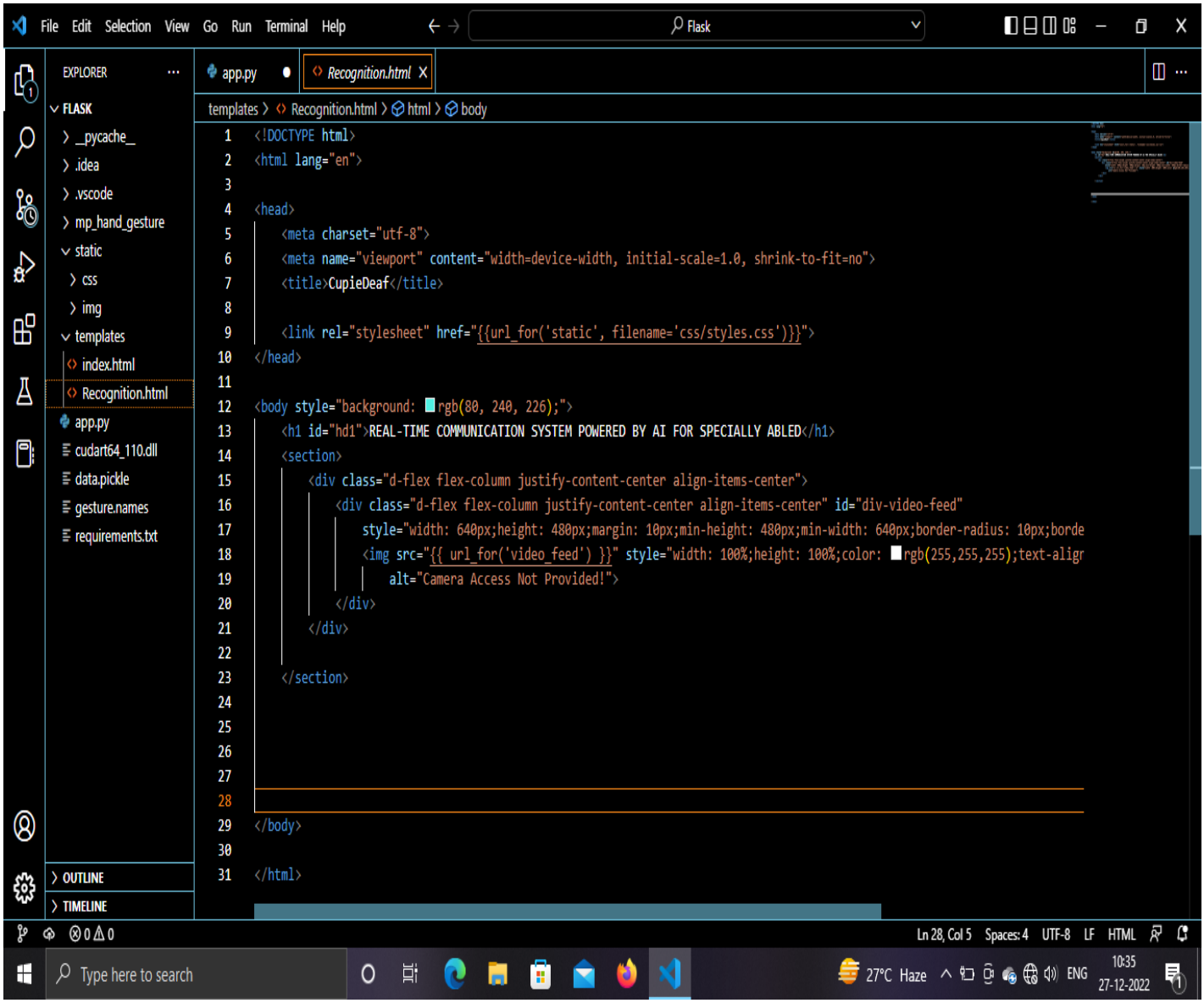
13.1.Source Code:

HTML:

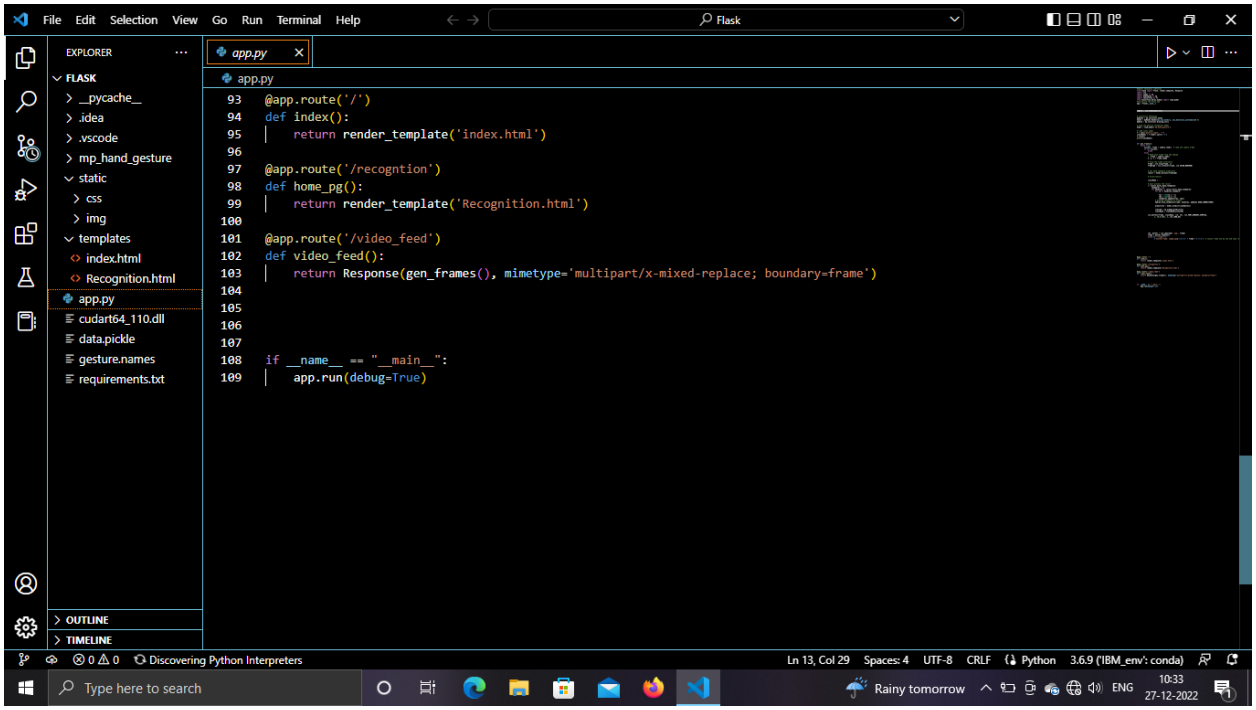
INDEX.HTML:

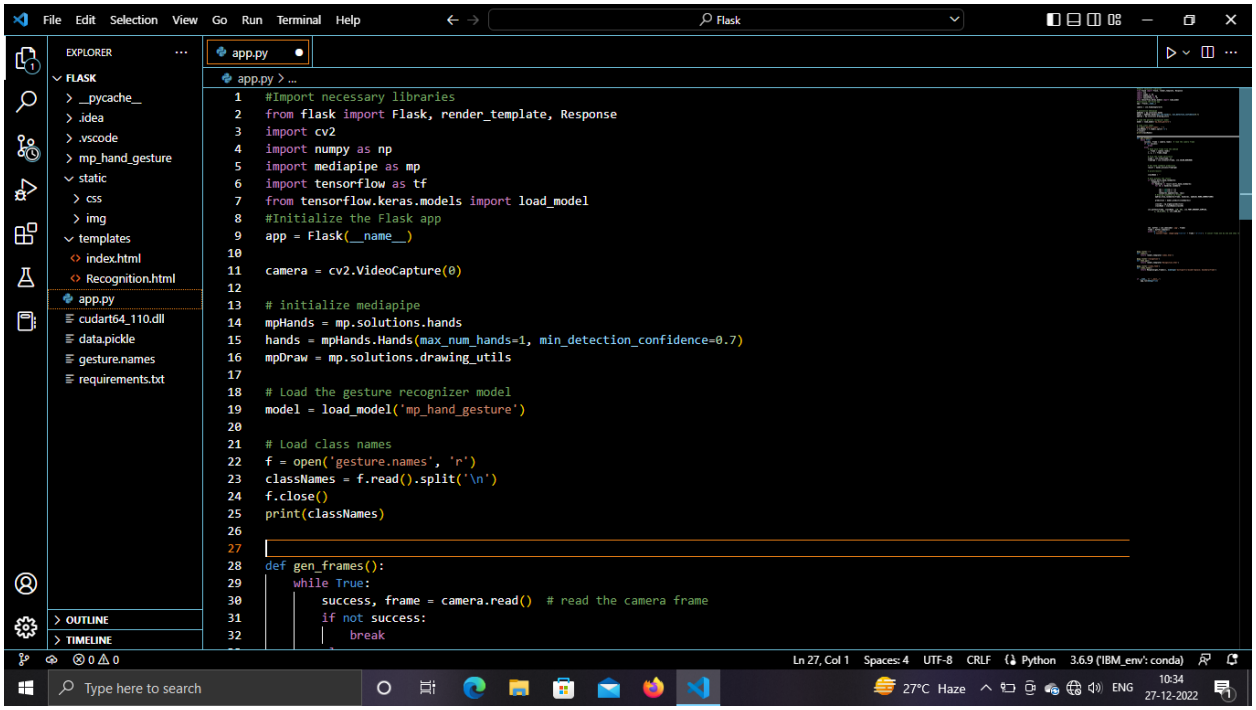


RECOGNIZATION.HTML:



APP:


File Edit Selection View Go Run Terminal Help
Flask
EXPLORER
app.py
FLASK
__pycache__
.idea
.vscode
mp_hand_gesture
static
css
img
templates
index.html
Recognition.html
app.py
cudart64_110.dll
data.pickle
gesture.names
requirements.txt
OUTLINE
TIMELINE
93 @app.route('/')
94 def index():
95 return render_template('index.html')
96
97 @app.route('/recogntion')
98 def home_pg():
99 return render_template('Recognition.html')
100
101 @app.route('/video_feed')
102 def video_feed():
103 return Response(gen_frames(), mimetype='multipart/x-mixed-replace; boundary=frame')
104
105
106
107
108 if __name__ == "__main__":
109 app.run(debug=True)
Discovering Python Interpreters
Ln 13, Col 29 Spaces: 4 UTF-8 CRLF Python 3.6.9 ('IBM_env': conda)
Type here to search
Rainy tomorrow
ENG
10:33
27-12-2022


File Edit Selection View Go Run Terminal Help
Flask
EXPLORER
app.py
FLASK
__pycache__
.idea
.vscode
mp_hand_gesture
static
css
img
templates
index.html
Recognition.html
app.py
cudart64_110.dll
data.pickle
gesture.names
requirements.txt
OUTLINE
TIMELINE
1 #Import necessary libraries
2 from flask import Flask, render_template, Response
3 import cv2
4 import numpy as np
5 import mediapipe as mp
6 import tensorflow as tf
7 from tensorflow.keras.models import load_model
8 #Initialize the Flask app
9 app = Flask(__name__)
10
11 camera = cv2.VideoCapture(0)
12
13 # initialize mediapipe
14 mpHands = mp.solutions.hands
15 hands = mpHands.Hands(max_num_hands=1, min_detection_confidence=0.7)
16 mpDraw = mp.solutions.drawing_utils
17
18 # Load the gesture recognizer model
19 model = load_model('mp_hand_gesture')
20
21 # Load class names
22 f = open('gesture.names', 'r')
23 classNames = f.read().split('\n')
24 f.close()
25 print(classNames)
26
27
28 def gen_frames():
29 while True:
30 success, frame = camera.read() # read the camera frame
31 if not success:
32 break
Ln 27, Col 1 Spaces: 4 UTF-8 CRLF Python 3.6.9 ('IBM_env': conda)
Type here to search
27°C Haze
ENG
10:34
27-12-2022

13.2.GitHub & Project Demo Link:

https://github.com/IBM-EPBL/IBM-Project-21683-1659787710

DEMO LINK:

https://drive.google.com/file/d/1SDb9kNhejKTk3M1YI6qPZneFizkN2l8R/view?usp=drivesdk