Real -Time Communication System Powered byAI for Specially Abled

TEAM ID: PNT2022TMID38653

A PROJECT REPORT

Submitted by

- TEAM LEADER :S.YOGESHWARI
- TEAM MEMBER1:M.AISHWARYA
- TEAM MEMBER2:S.HEMAPRIYA
- TEAM MEMBER3:D.SIVADHARSHINI
- TEAM MEMBER4:S.SUVATHINI

ADHIPARASAKTHI

ENGINEERING COLLEGE

MELMARUVATHUR-603319

(Affiliated by Anna University)

CONTENTS

ABSTRACT :

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. Communications 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 situationswhere other forms like speech cannot be used. Voice Conversion System with 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.

New system that helps convert sign language to text and speech for easiercommunication with audience.

1.INTRODUCTION:

1.1Project Overview:

Gesture is a non-verbal means of communication. It refers to expressing an idea using position, orientation or movement of a body part. Gesture recognition is the mathematical interpretation of orientation or motionof human body by a computational system. In this project, the words expressedby hand gestures by the speech and hearing impaired are converted into verbal means of communication. The translated output is displayed on a screen and “spoken” on a speaker.

Sign Language is the well-structured code, which uses hand gestures instead of sound to convey meaning, simultaneously combining hand shapes, orientations and movement of the hands. Communicative hand glove is anelectronic device that can translate sign language into speech and text in order to make the communication possible between the deaf and/or mute with the general public. This technology has been used in a variety of application areas, which demands accurate

interpretation of sign language. In this project, the words/lettersconveyed by the disabled person are displayed on a screen and also spoken on a speaker.

1.1 Purpose

The project aims to develop a system that converts the sign languageinto a human hearing voice in the desired language to convey a message to normal people, as well as convert speech into understandable sign language for the deaf and dumb using the convolutional neural network.

An app is built which enables the deaf and dumb people to convey their information using signs which is converted to human understandablelanguage and output is given as speech.

2.LITERATURE SURVEY:

2.1 Existing problem:

Communications 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. Only specially abled people are taught sign language and the commonperson is unaware its working causing a communication gap.Under emergency situations, it is even more difficult for specially abled people to get help. Non-Emergency normal environments can also behardfor them to navigate needing special assistance.

2.2References:

1 Koufos, K., EL Haloui, K., Dianati, M., Higgins, M., Elmirghani, J., Imran, M. A., & Tafazolli, R. (2021). Trends in Intelligent Communication Systems: Review of Standards, Major Research Projects,and Identification of Research Gaps. Journal of Sensor and Actuator Networks, 10(4), 60.ibm.com/blogs/internet-of-things/connected-trains- rail-travel/

2 Panda, G., Upadhyay, A. K., & Khandelwal, K. (2019). Artificial intelligence: A strategic disruption in public

relations. Journal of Creative Communications, 14(3), 196-213.

3 Xu, G., Mu, Y., & Liu, J. (2017). Inclusion of artificial intelligence incommunication networks and services. ITU J. ICT Discov. Spec, 1, 1-6.

4 Verma, P., Shimi S. L. and Priyadarshani, R., "Design of CommunicationInterpreter for Deaf and Dumb Person", Vol.4, no.1,2013.

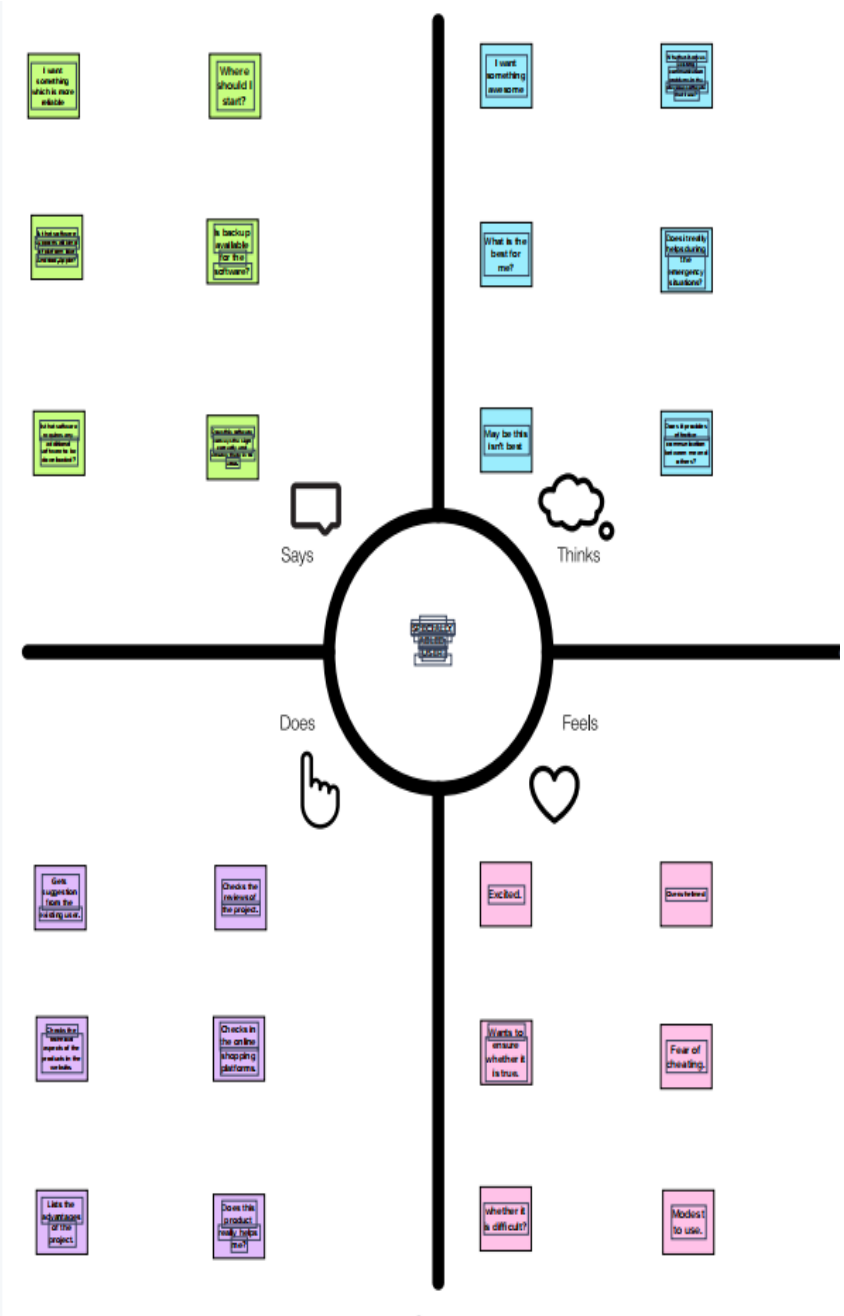
2.1 Problem Statement Definition:

Only specially abled people are taught sign language and the commonperson is unaware its working causing a communication gap. Under emergency situations, it is even more difficult for specially abled people to get help. Non-Emergency normal environments can also behard for them to navigate needing special assistance.

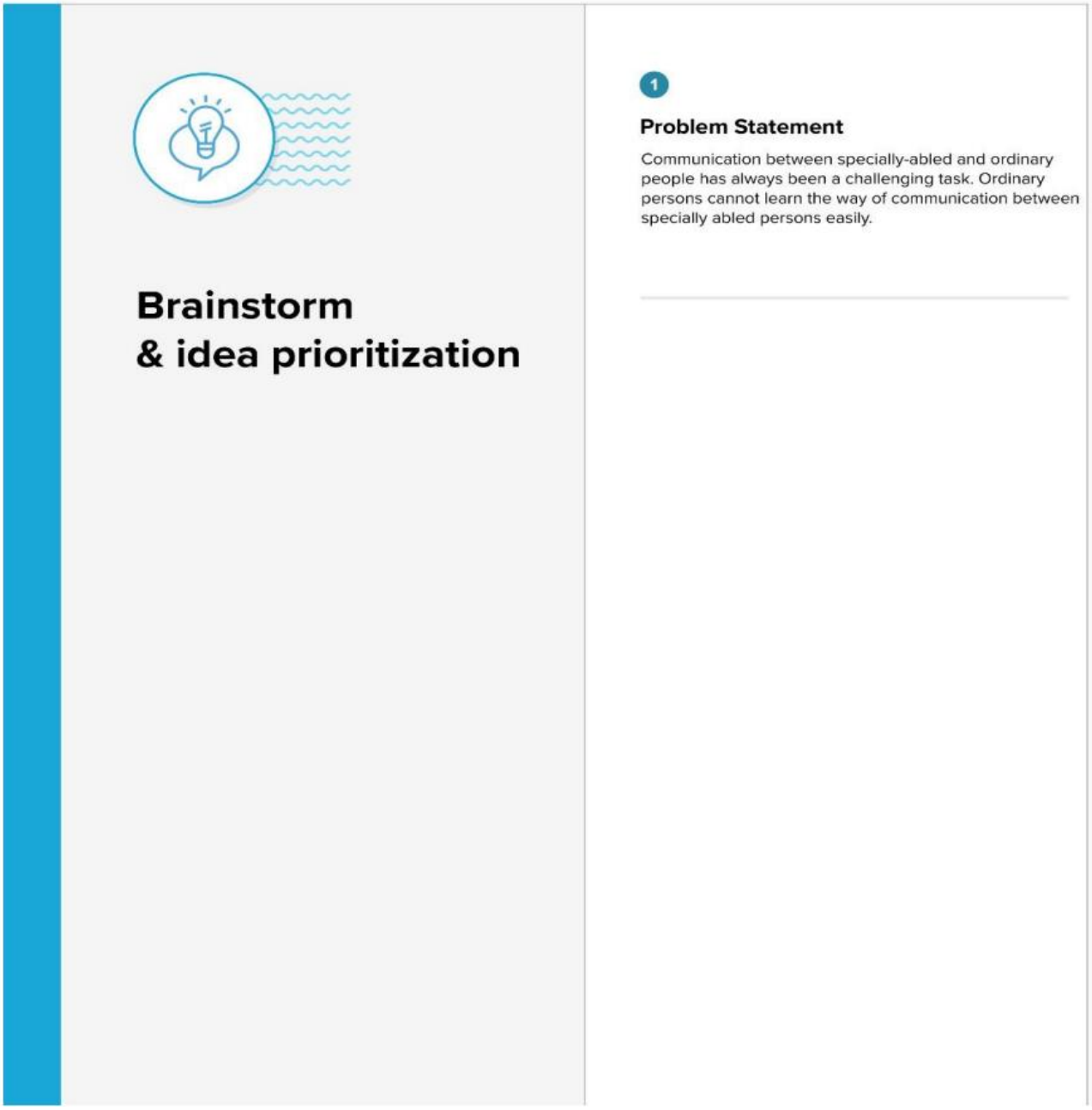
Communications between deaf-mute and a normal person has always been a challenging task. It is very difficult for mute people to convey their message to normal people. Since normal people are not trained on hand sign language.

3. IDEATION & PROPOSED SOLUTION

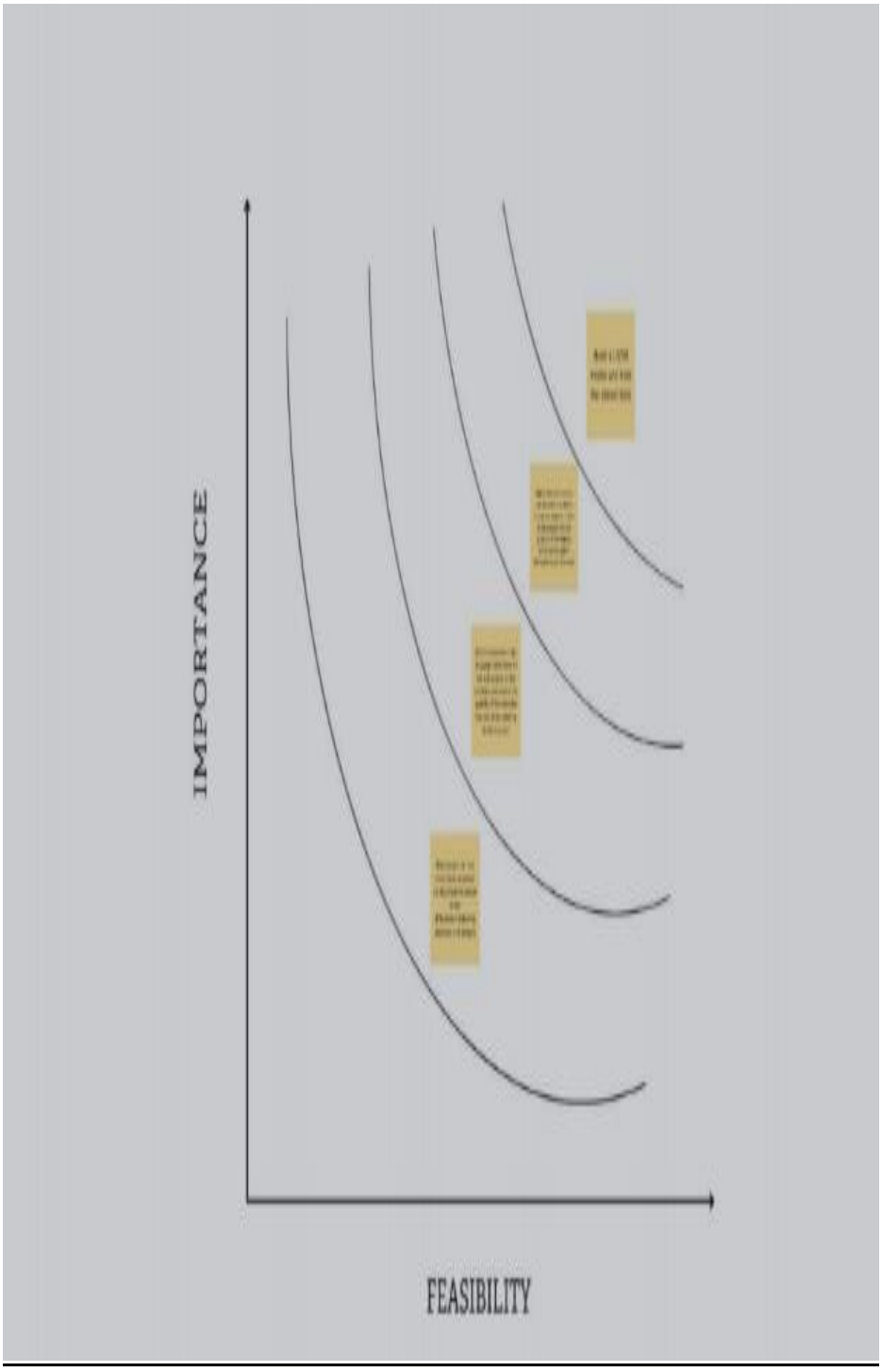
3.1 Empathy Map Canvas



3.2 Brain& Idea Prioritization:



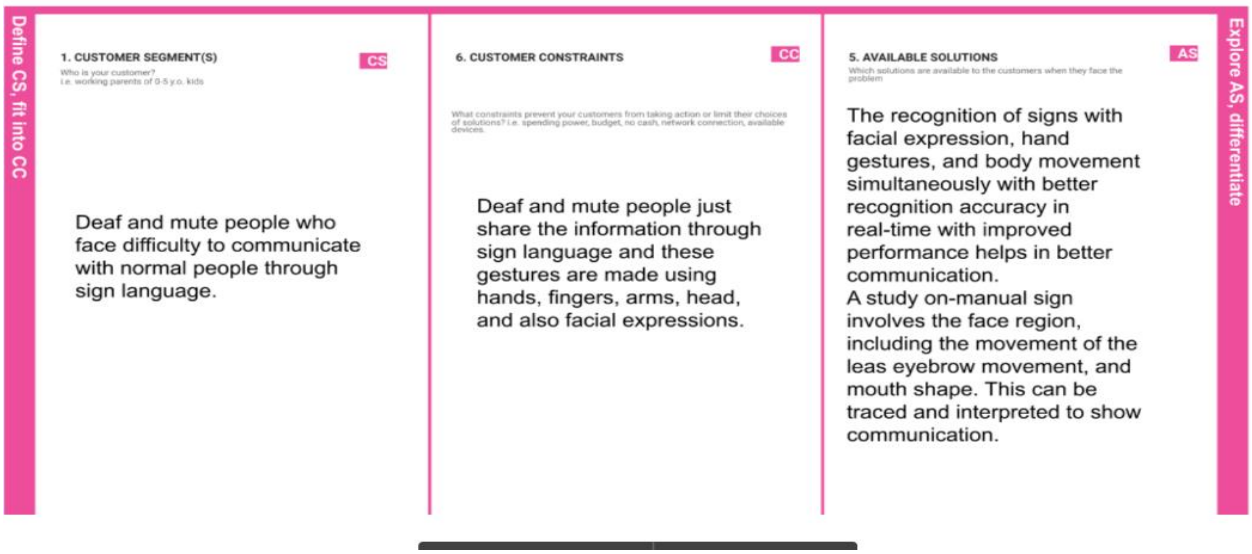
3.3 Prioritize

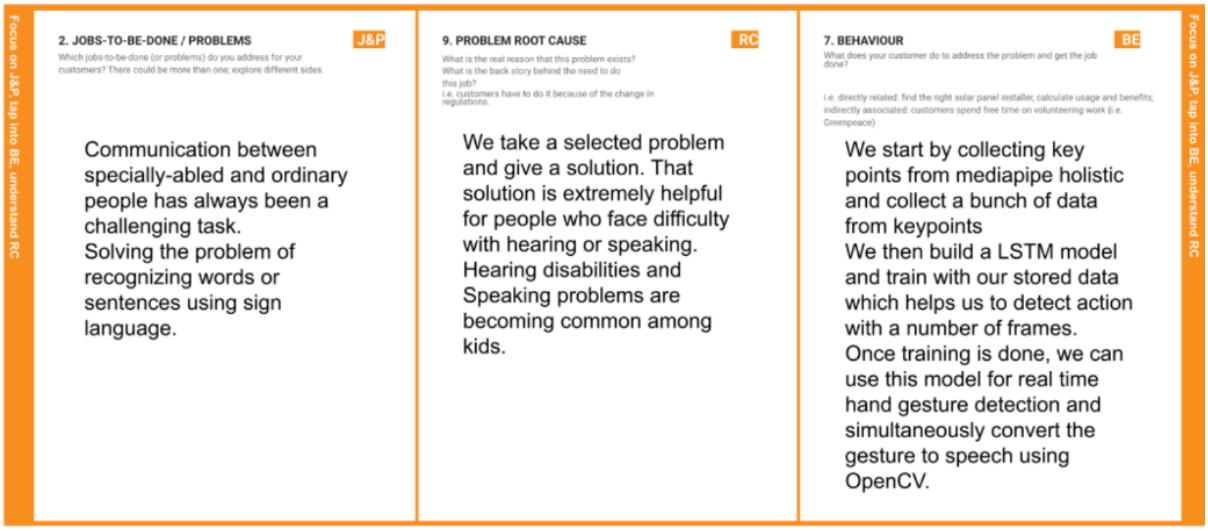


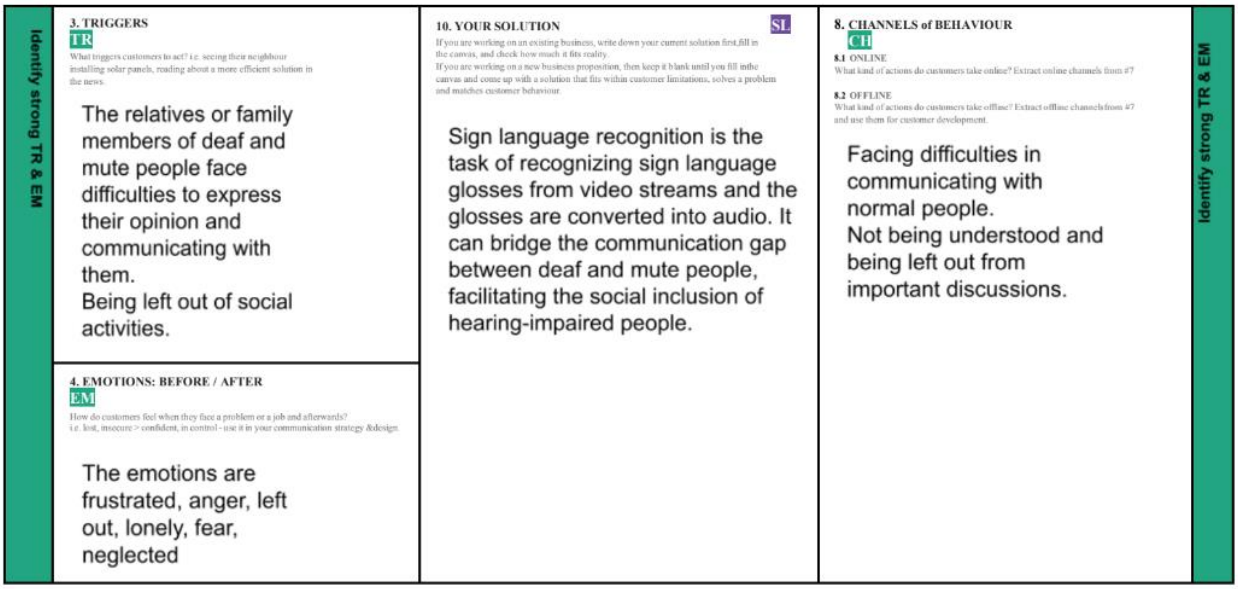
3.4 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Communication between deaf-mute and a normal person. It is often difficult for mute person to convey their information to normal people in emergency as well as in normal times since normal people are not trained in sign language. Hence there is a need for a system which enables them to communicate with normal people.
2.	Idea / Solution description	To develop a system that converts sign language into human hearing voice so that it can be conveyed to normal people.
3.	Novelty / Uniqueness	Convolution neural network is used to create the model and it is trained on different hand gestures and an app is built.
4.	Social Impact / Customer Satisfaction	Communication process is carried without the help of additional human intervention and there is no additional hardware support needed.
5.	Business Model (Revenue Model)	The app can be made available to more groups which will increase its growth.
6.	Scalability of the Solution	Easy to handle and it can be accessed from any device and by everyone as it is hosted in IBM cloud.

3.5 Problem Solution fit







4. REQUIREMENT ANALYSIS

4.1 Functional requirements

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement(Epic)	Sub Requirement(Story/Sub-Task)
FR-1	User Registration	Registration is done through the google forms.
FR-2	User Confirmation	Confirmation is done via EmailConfirmation by OTP.
FR-3	Authentication	Authentication through Facial recognition. Authentication through Password authentication protocol.
FR-4	External interfaces	Microphone ,Camera Ethernet , Wi-Fi and USB dongle to provide internet facilities.
FR-5	Reporting	If there are any issues faced by the customer or user it will be directly notified to the developer.
FR-6	Compliance to Rules or Laws	Privacy policy , Terms and Conditions , End user agreement.

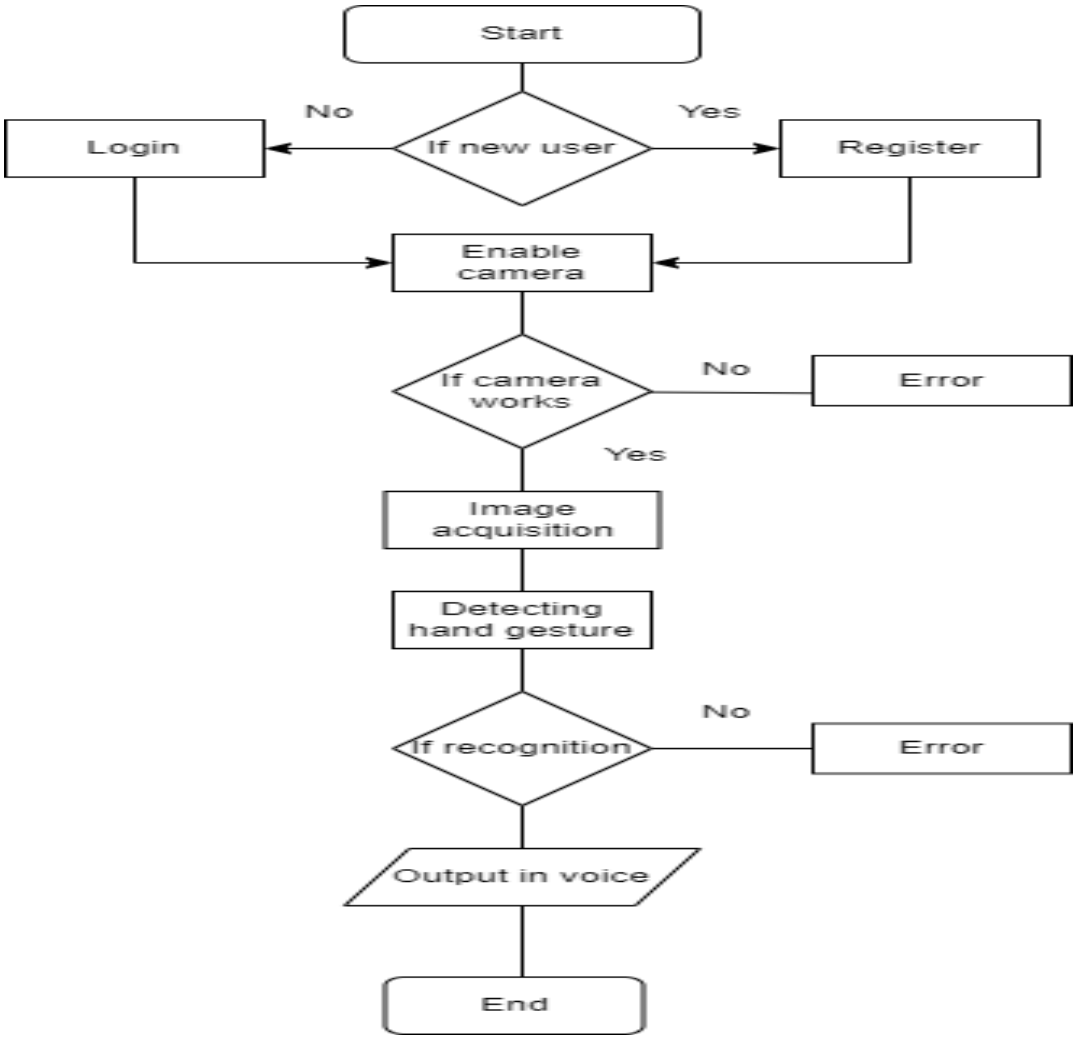
4.2 Non-functional Requirements:

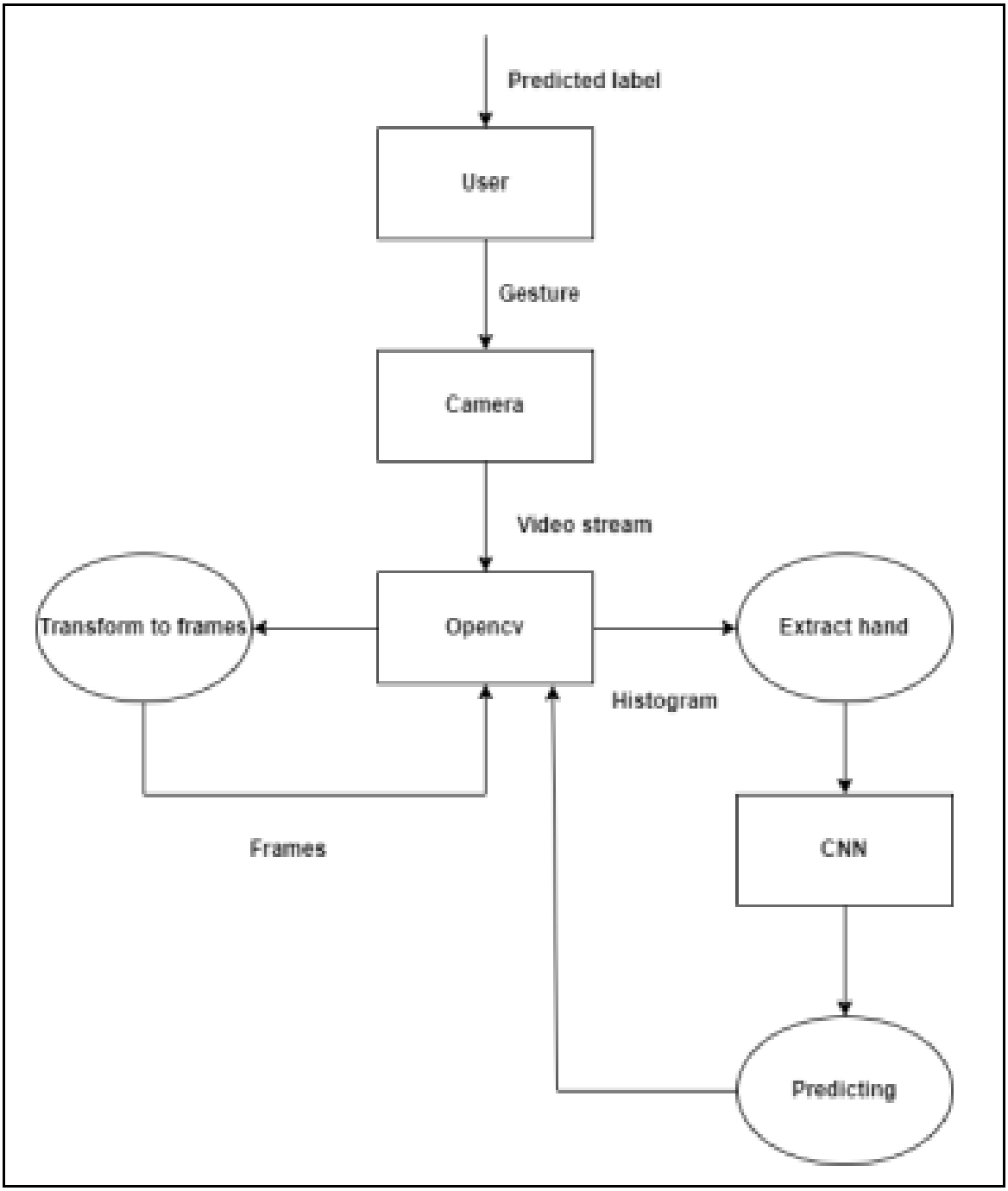
Following are the non-functional requirements of the proposed solution.

NFR No.	Non- functional Requirements	Description
NFR-1	**Usability**	The system should be user friendly .
NFR-2	**Security**	All the communication information is accessed only by the user.
NFR-3	**Reliability**	It sets the pace for the future and helps the people indeed.
NFR-4	**Performance**	Enables people with disabilities to step into a world where their difficulties are understood and taken into account.
NFR-5	**Availability**	Provides automatic recovery as much as possible.
NFR-6	**Scalability**	This enables the specially abled interaction with the environment.

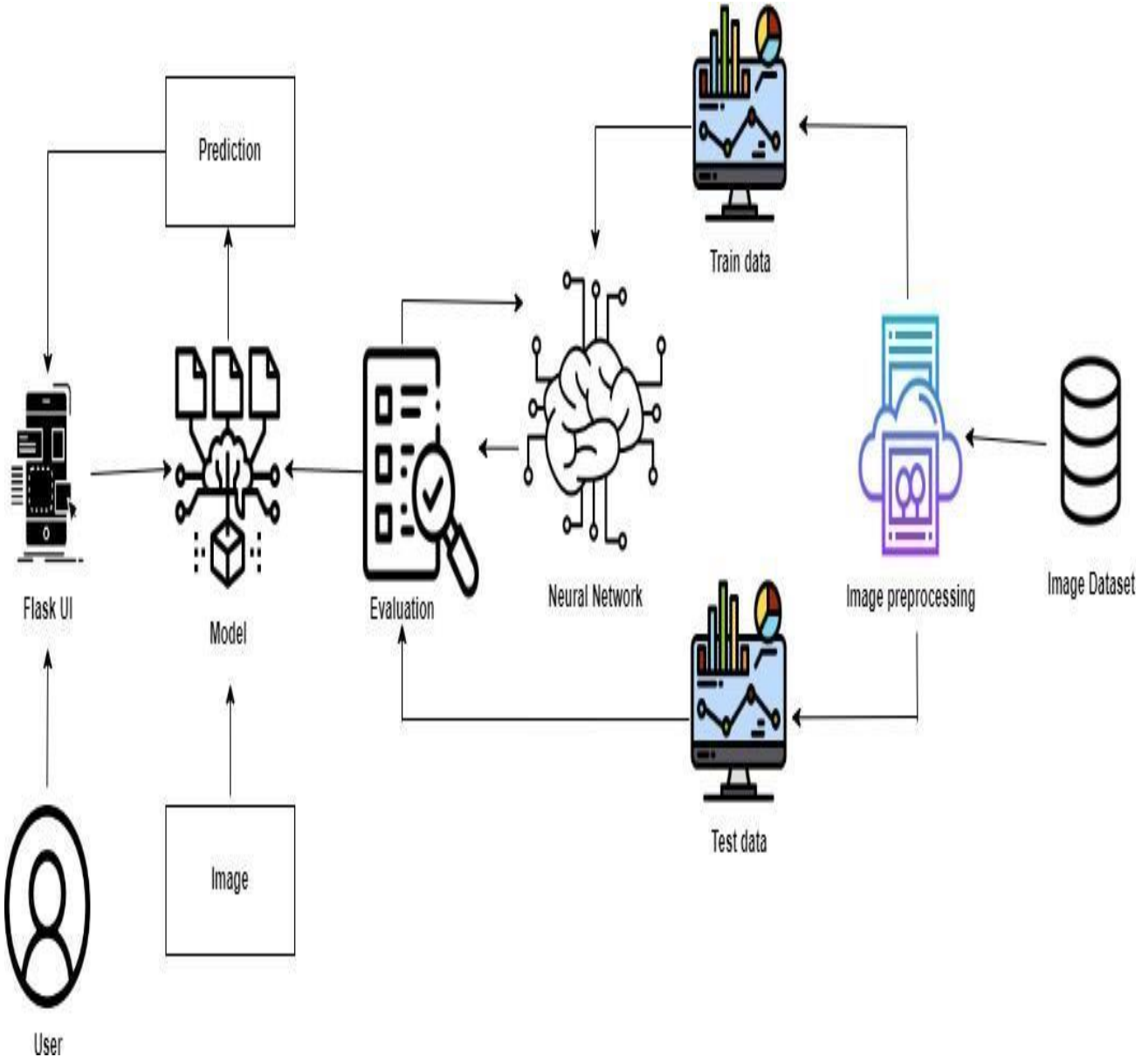
5.PROJECT DESIGN

5.1 Data Flow Diagrams




Predicted label
User
Gesture
Camera
Video stream
Transform to frames
Opencv
Extract hand
Histogram
Frames
CNN
Predicting

5.2 TECHNICAl ARCHITECTURE



5.3User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Normal people and Deaf-mute people	Registration	USN-1	As a user, I can register for the application by entering my email, and password, and confirming my password	I can access my account/dashboard	High	Sprint-1
		USN-2	As a user, I will receive a confirmation email once I have registered for the application	I can receive a confirmation email & click confirm	High	Sprint-1
	login	USN-3	As a user I can log into the application by entering the registered Email id and Password	If the registered information matches the given information accept login.	High	Sprint-1
	Dashboard		Two options available Choose the option based on who uses the app. If you are normal person click the "normal people" option. If you are deaf-mute people click "deaf-mute" Option	I can access my dashboard	High	Sprint-2
Normal people			Give access to camera to recognize the gestures Give access to microphone to give our message through voice	I can access messages given by the Deaf-mute people	High	Sprint-2
Deaf-mute people			Give access to display to view the message sent by normal people.	I can access messages given by the Normal people	High	Sprint-2

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

TITLE	DESCRIPTION	DATE
Literature Survey & Information Gathering	Literature survey on the selected project & gathering information by referring the, technical papers , research publications etc.	11 SEPTEMBER 2022
Prepare Empathy Map	Prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare list of problem statements	10 SEPTEMBER 2022
Ideation	List the by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.	24 OCTOBER 2022

Proposed Solution	Prepare the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.	24 OCTOBER 2022
Problem Solution Fit	Prepare problem - solution fit document.	24 OCTOBER 2022
Solution Architecture	Prepare solution architecture document.	24 OCTOBER 2022
Customer Journey	Prepare the customer journey maps to understand the user interactions & experiences with the application (entry to exit).	28 OCTOBER 2022
Functional Requirement	Prepare the functional requirement document.	28 OCTOBER 2022
Data Flow Diagrams	Draw the data flow diagrams and submit for review.	28 OCTOBER 2022
Technology Architecture	Prepare the technology architecture diagram.	28 OCTOBER 2022
Prepare Milestone & Activity List	Prepare the milestones & activity list of the project.	29 OCTOBER 2022
Project Development - Delivery of Sprint-1, 2, 3 & 4	Develop & submit the developed code by testing it.	19 NOVEMBER 2022

6.2 Sprint Delivery Schedule

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

To create product backlog and sprint schedule

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	Hemapriya
Sprint-1	Registration	USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	Suvathini
Sprint-2	Registration	USN-3	As a user, I can register for the application through phone number	2	Medium	Sivadarshini
Sprint-2	User interface	USN-4	Professional responsible for user requirements & needs	2	Medium	Aishwarya
Sprint-3	Login	USN-5	As a user, I can log into the application by entering email & password	1	High	Yogeshwari
Sprint-3	Dashboard	USN-6	As a user, I must receive any updates or pop ups in my dashboard	2	High	Yogeshwari
Sprint-4	Details	USN-7	As a user, I should get notification about the progress and any updates via email or sms	1	Medium	Suvathini

Sprint-4	Privacy	USN-8	The developed application should be secure for the users	2	High	Aishwarya

Project Tracker, Velocity & Burndown Chart: (4 Marks)

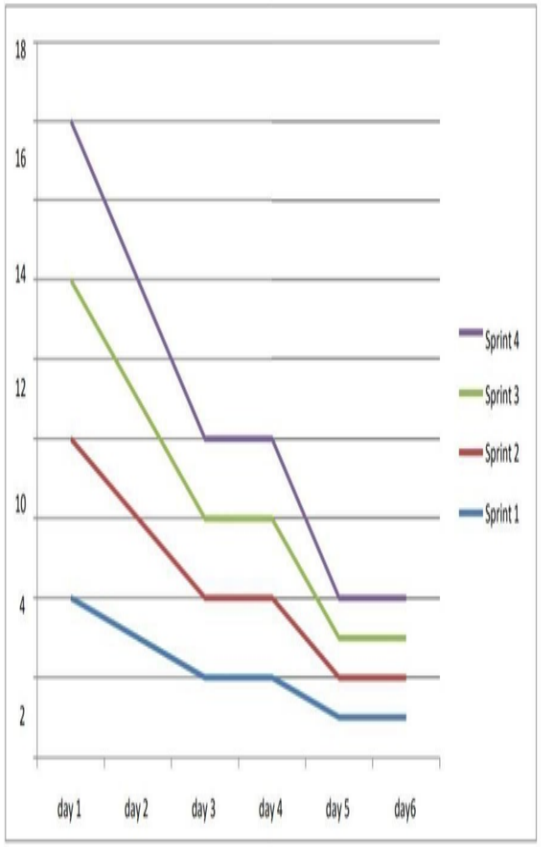
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	30 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	13 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	29 Nov 2022

$$AV = 6/10 = 0.6$$

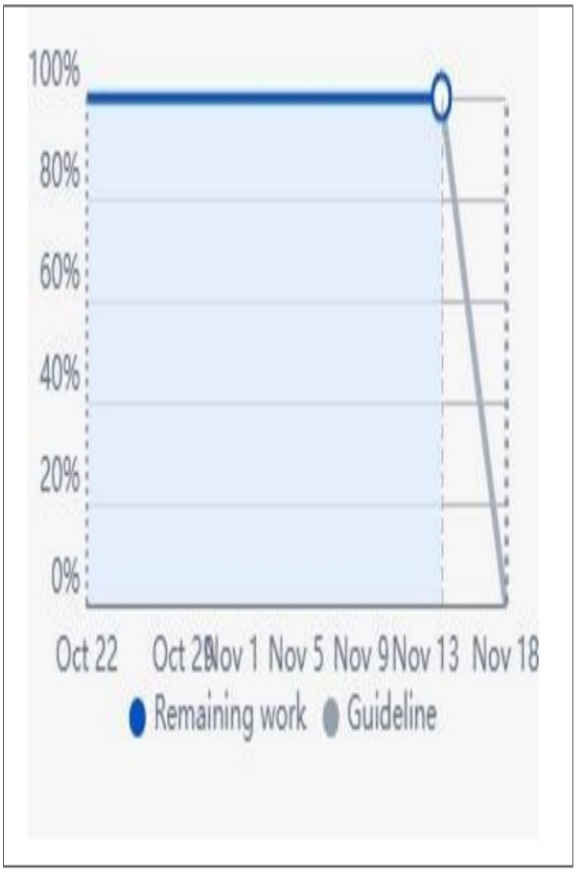
Burndown Chart:

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.

SPRINT SCHEDULE CHART:



SPRINT BURNDOWN CHART:



7.CODING & SOLUTION(Explain the features added in the project along with code)

7.1 Model Building

Importing The Required Model Building Libraries

In [ ]:
```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

In [ ]:
```
from keras.models import Sequential, load_model
from keras.layers.core import Dense, Dropout, Activation
from keras.utils import np_utils
```

In [ ]:
```
# Training Datagen
train_datagen = ImageDataGenerator(rescale=1/255,zoom_range=0.2,horizontal_flip=True,vertical_flip=False)
# Testing Datagen
test_datagen = ImageDataGenerator(rescale=1/255)
```

In [ ]:
```
# Training Dataset
x_train=train_datagen.flow_from_directory(r'/content/drive/MyDrive/Dataset/training_set',target_size=(64,64), class_mode='categorical',batch_size=900)
# Testing Dataset
x_test=test_datagen.flow_from_directory(r'/content/drive/MyDrive/Dataset/test_set',target_size=(64,64), class_mode='categorical',batch_size=900)
```

```
Found 15760 images belonging to 9 classes.
Found 2250 images belonging to 9 classes.
```

In [ ]:
```
print("Len x-train : ", len(x_train))
print("Len x-test : ", len(x_test))
```

```
Len x-train :  18
Len x-test :  3
```

In [ ]:
```
# The Class Indices in Training Dataset
x_train.class_indices
```

Out[ ]:
```
{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}
```

```
model=Sequential()
```

Initializing The Model

In [ ]:
```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

In [ ]:
```
spatial_dropout=0.05
recurrent_dropout=0.1
```

In [ ]:
```
# Training Datagen
train_datagen = ImageDataGenerator(rescale=1/255,zoom_range=0.2,horizontal_flip=True,vertical_flip=False)
# Testing Datagen
test_datagen = ImageDataGenerator(rescale=1/255)
```

In [ ]:
```
# Training Dataset
x_train=train_datagen.flow_from_directory(r'/content/drive/MyDrive/Dataset/training_set',target_size=(64,64), class_mode='categorical',batch_size=900)
# Testing Dataset
x_test=test_datagen.flow_from_directory(r'/content/drive/MyDrive/Dataset/test_set',target_size=(64,64), class_mode='categorical',batch_size=900)
```

```
Found 15760 images belonging to 9 classes.
Found 2250 images belonging to 9 classes.
```

In [ ]:
```
print("Len x-train : ", len(x_train))
print("Len x-test : ", len(x_test))
```

```
Len x-train :  18
Len x-test :  3
```

In [ ]:
```
# The Class Indices in Training Dataset
x_train.class_indices
```

Out[ ]:
```
{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}
```

Model Creation

In [ ]:
```
# Importing Libraries
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Convolution2D,MaxPooling2D,Flatten,Dense
```

In [ ]:
```
dataset = pd.read_csv('E:\Datasets\Mall_Customers.csv')
```

Adding The Convolution Layer

In [ ]:
```
import numpy as np
import matplotlib.pyplot as plt
```

In [ ]:
```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

In [ ]:
```
# Training Datagen
train_datagen = ImageDataGenerator(rescale=1/255,zoom_range=0.2,horizontal_flip=True,vertical_flip=False)
# Testing Datagen
test_datagen = ImageDataGenerator(rescale=1/255)
```

In [ ]:
```
# Training Dataset
x_train=train_datagen.flow_from_directory(r'/content/drive/MyDrive/Dataset/training_set',target_size=(64,64), class_mode='categorical',batch_size=900)
# Testing Dataset
x_test=test_datagen.flow_from_directory(r'/content/drive/MyDrive/Dataset/test_set',target_size=(64,64), class_mode='categorical',batch_size=900)
```

```
Found 15760 images belonging to 9 classes.
Found 2250 images belonging to 9 classes.
```

In [ ]:
```
# let img1 be an image with no features
img1 = np.array([np.array([200, 200]), np.array([200, 200])])
img2 = np.array([np.array([200, 200]), np.array([0, 0])])
img3 = np.array([np.array([200, 0]), np.array([200, 0])])

kernel_horizontal = np.array([np.array([2, 2]), np.array([-2, -2])])
print(kernel_horizontal, 'is a kernel for detecting horizontal edges')

kernel_vertical = np.array([np.array([2, -2]), np.array([2, -2])])
print(kernel_vertical, 'is a kernel for detecting vertical edges')
```

In [ ]:
```
# We will apply the kernels on the images by
# elementwise multiplication followed by summation
def apply_kernel(img, kernel):
    return np.sum(np.multiply(img, kernel))

# Visualizing img1
plt.imshow(img1)
plt.axis('off')
plt.title('img1')
plt.show()

# Checking for horizontal and vertical features in image1
print('Horizontal edge confidence score:', apply_kernel(img1,
                                        kernel_horizontal))
print('Vertical edge confidence score:', apply_kernel(img1,
                                        kernel_vertical))
```

In [ ]:
```
# Visualizing img2
plt.imshow(img2)
plt.axis('off')
plt.title('img2')
plt.show()

# Checking for horizontal and vertical features in image2
print('Horizontal edge confidence score:', apply_kernel(img2,
                                        kernel_horizontal))
print('Vertical edge confidence score:', apply_kernel(img2,
                                        kernel_vertical))
```

In [ ]:
```
# Visualizing img3
plt.imshow(img3)
plt.axis('off')
plt.title('img3')
plt.show()

# Checking for horizontal and vertical features in image3
print('Horizontal edge confidence score:', apply_kernel(img3,
                                        kernel_horizontal))
print('Vertical edge confidence score:', apply_kernel(img3,
                                        kernel_vertical))
```

In [ ]:
```
print("Len x-train : ", len(x_train))
print("Len x-test : ", len(x_test))
```

```
Len x-train :  18
Len x-test :  3
```

In [ ]:
```
# The Class Indices in Training Dataset
x_train.class_indices
```

Out[ ]:
```
{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}
```

Model Creation

In [ ]:
```
# Importing Libraries
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Convolution2D,MaxPooling2D,Flatten,Dense
```

In [ ]:
```
# Creating Model
model=Sequential()
```

In [ ]:
```
# Adding Layers
model.add(Convolution2D(32,(3,3),activation='relu',input_shape=(64,64,3)))
```

Adding The Pooling Layer

In [ ]:
```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

In [ ]:
```
import numpy as np
from keras.models import Sequential
from keras.layers import MaxPooling2D
```

In [ ]:
```
# define input image
image = np.array([[2, 2, 7, 3],
                  [9, 4, 6, 1],
                  [8, 5, 2, 4],
                  [3, 1, 2, 6]])
image = image.reshape(1, 4, 4, 1)
```

In [ ]:
```
# define model containing just a single max pooling layer
model = Sequential(
        [MaxPooling2D(pool_size = 2, strides = 2)])

# generate pooled output
output = model.predict(image)
```

In [ ]:
```
# print output image
output = np.squeeze(output)
print(output)
```

In [ ]:
```
# Training Datagen
train_datagen = ImageDataGenerator(rescale=1/255,zoom_range=0.2,horizontal_flip=True,vertical_flip=False)
# Testing Datagen
test_datagen = ImageDataGenerator(rescale=1/255)
```

In [ ]:
```
# Training Dataset
x_train=train_datagen.flow_from_directory(r'/content/drive/MyDrive/Dataset/training_set',target_size=(64,64), class_mode='categorical',batch_size=900)
# Testing Dataset
x_test=test_datagen.flow_from_directory(r'/content/drive/MyDrive/Dataset/test_set',target_size=(64,64), class_mode='categorical',batch_size=900)
```

```
Found 15760 images belonging to 9 classes.
Found 2250 images belonging to 9 classes.
```

In [ ]:
```
print("Len x-train : ", len(x_train))
print("Len x-test : ", len(x_test))
```

```
Len x-train :  18
Len x-test :  3
```

In [ ]:
```
# The Class Indices in Training Dataset
x_train.class_indices
```

Out[ ]:
```
{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}
```

Model Creation

In [ ]:
```
# Importing Libraries
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Convolution2D,MaxPooling2D,Flatten,Dense
```

In [ ]:
```
# Creating Model
model=Sequential()
```

In [ ]:
```
# Adding Layers
model.add(Convolution2D(32,(3,3),activation='relu',input_shape=(64,64,3)))
```

In [ ]:
```
model.add(MaxPooling2D(pool_size=(2,2)))
```

Adding The Flatten Layer

In [ ]:
```
# importing numpy as np
import numpy as np
```

In [ ]:
```
# declare flatten np
gfg = np.array([[6, 9, 12], [8, 5, 2], [18, 21, 24]])

# using array.flatten() method
flat_gfg = gfg.flatten(order='A')
print(flat_gfg)
```

In [ ]:
```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

In [ ]:
```
# Training Datagen
train_datagen = ImageDataGenerator(rescale=1/255,zoom_range=0.2,horizontal_flip=True,vertical_flip=False)
# Testing Datagen
test_datagen = ImageDataGenerator(rescale=1/255)
```

In [ ]:
```
# Training Dataset
x_train=train_datagen.flow_from_directory(r'/content/drive/MyDrive/Dataset/training_set',target_size=(64,64), class_mode='categorical',batch_size=900)
# Testing Dataset
x_test=test_datagen.flow_from_directory(r'/content/drive/MyDrive/Dataset/test_set',target_size=(64,64), class_mode='categorical',batch_size=900)
```

```
Found 15760 images belonging to 9 classes.
Found 2250 images belonging to 9 classes.
```

In [ ]:
```
print("Len x-train : ", len(x_train))
print("Len x-test : ", len(x_test))
```

```
Len x-train :  18
Len x-test :  3
```

In [ ]:
```
# The Class Indices in Training Dataset
x_train.class_indices
```

Out[ ]:
```
{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}
```

Model Creation

In [ ]:
```
model = Sequential()
for i, feat in enumerate(args.conv_f):
    if i==0:
        model.add(Conv2D(feat, input_shape=x[0].shape, kernel_size=3, padding = 'same',use_bias=False))
    else:
        model.add(Conv2D(feat, kernel_size=3, padding = 'same',use_bias=False))
        model.add(BatchNormalization())
        model.add(LeakyReLU(alpha=args.conv_act))
        model.add(Conv2D(feat, kernel_size=3, padding = 'same',use_bias=False))
        model.add(BatchNormalization())
        model.add(LeakyReLU(alpha=args.conv_act))
        model.add(Dropout(args.conv_do[i]))
```

In [ ]:
```
model.add(Flatten())

#Input code here

denseArgs = {'use_bias':False}
for i,feat in enumerate(args.dense_f):
    model.add(Dense(feat,**denseArgs))
    model.add(BatchNormalization())
    model.add(LeakyReLU(alpha=args.dense_act))
    model.add(Dropout(args.dense_do[i]))
model.add(Dense(1))
```

In [ ]:
```
# Importing Libraries
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Convolution2D,MaxPooling2D,Flatten,Dense
```

In [ ]:
```
# Importing Libraries
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Convolution2D,MaxPooling2D,Flatten,Dense
```

In [ ]:
```
# Creating Model
model=Sequential()
```

In [ ]:
```
# Adding Layers
model.add(Convolution2D(32,(3,3),activation='relu',input_shape=(64,64,3)))
```

In [ ]:
```
model.add(MaxPooling2D(pool_size=(2,2)))
```

In [ ]:
```
model.add(Flatten())
```

In [ ]:
```
# Adding Dense Layers
model.add(Dense(300,activation='relu'))
model.add(Dense(150,activation='relu'))
model.add(Dense(9,activation='softmax'))
```

Adding The Dense Layers

In [ ]:
```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

In [ ]:
```
model.add(Dense(units=512, activation='relu'))
model.add(Dense(units=9, activation='softmax'))
```

In [ ]:
```
print("Adding dense layer on top")
model.add(layers.Flatten())
model.add(layers.Dense(64, activation='relu'))
model.add(layers.Dense(10))
```

In [ ]:
```
print("Complete architecture of the model")
model.summary()
```

In [ ]:
```
# Training Datagen
train_datagen = ImageDataGenerator(rescale=1/255,zoom_range=0.2,horizontal_flip=True,vertical_flip=False)
# Testing Datagen
test_datagen = ImageDataGenerator(rescale=1/255)
```

In [ ]:
```
# Training Dataset
x_train=train_datagen.flow_from_directory(r'/content/drive/MyDrive/Dataset/training_set',target_size=(64,64), class_mode='categorical',batch_size=900)
# Testing Dataset
x_test=test_datagen.flow_from_directory(r'/content/drive/MyDrive/Dataset/test_set',target_size=(64,64), class_mode='categorical',batch_size=900)
```

```
Found 15760 images belonging to 9 classes.
Found 2250 images belonging to 9 classes.
```

In [ ]:
```
print("Len x-train : ", len(x_train))
print("Len x-test : ", len(x_test))
```

```
Len x-train :  18
Len x-test :  3
```

In [ ]:
```
# The Class Indices in Training Dataset
x_train.class_indices
```

Out[ ]:
```
{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}
```

Model Creation

In [ ]:
```
# Importing Libraries
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Convolution2D,MaxPooling2D,Flatten,Dense
```

In [ ]:
```
# Creating Model
model=Sequential()
```

In [ ]:
```
# Adding Layers
model.add(Convolution2D(32,(3,3),activation='relu',input_shape=(64,64,3)))
```

In [ ]:
```
model.add(MaxPooling2D(pool_size=(2,2)))
```

In [ ]:
```
# Adding Dense Layers
model.add(Dense(300,activation='relu'))
model.add(Dense(150,activation='relu'))
model.add(Dense(9,activation='softmax'))
```

In [ ]:
```
# Compiling the Model
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
```

Compile To The Model

In [ ]:
```
from tensorflow.keras.preprocessing.image
import ImageDataGenerator
```

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

In [ ]:
```
# Creating sample sourcecode to multiply two variables
# x and y.
srcCode = 'x = 10\ny = 20\nmul = x * y\nprint("mul =", mul)'

# Converting above source code to an executable
execCode = compile(srcCode, 'mulstring', 'exec')

# Running the executable code.
exec(execCode)
```

In [ ]:
```
# Training Datagen
train_datagen = ImageDataGenerator(rescale=1/255,zoom_range=0.2,horizontal_flip=True,vertical_flip=False)
# Testing Datagen
test_datagen = ImageDataGenerator(rescale=1/255)
```

In [ ]:
```
# Training Dataset
x_train=train_datagen.flow_from_directory(r'/content/drive/MyDrive/Dataset/training_set',target_size=(64,64), class_mode='categorical',batch_size=900)
# Testing Dataset
x_test=test_datagen.flow_from_directory(r'/content/drive/MyDrive/Dataset/test_set',target_size=(64,64), class_mode='categorical',batch_size=900)
```

```
Found 15760 images belonging to 9 classes.
Found 2250 images belonging to 9 classes.
```

In [ ]:
```
def compile_model_results(model, root="./"):

    listing = glob.glob(root + '/models/' + model + '/*/best_pars.pkl')

    dic_list = []
    for file in listing:
        tmp = hyper_parameters_load(file)
        dic_list.append(tmp.to_dictionary())

    df = pd.DataFrame(dic_list)
    df['diff'] = df.test_F1 - df.forecast_F1
    df['pci'] = abs(df.test_F1 - df.forecast_F1)

    if not os.path.exists(root + '/figures/' +  model ):
        os.makedirs(root + '/figures/' +  model )

    df.to_csv(root + '/figures/' +  model + '/results.csv', index=False)

    return df
```

In [ ]:
```
# Set optimizer loss and metrics
    opt = Adam(lr=args.initial_lr, beta_1=0.99, beta_2=0.999, decay=1e-6)
    if args.net.find('caps') != -1:
        metrics = {'out_seg': dice_hard}
    else:
        metrics = [dice_hard]

    loss, loss_weighting = get_loss(root=args.data_root_dir, split=args.split_num, net=args.net,
                                    recon_wei=args.recon_wei, choice=args.loss)

    # If using CPU or single GPU
    if args.gpus <= 1:
        uncomp_model.compile(optimizer=opt, loss=loss, loss_weights=loss_weighting, metrics=metrics)
        return uncomp_model
    # If using multiple GPUs
    else:
        with tf.device("/cpu:0"):
            uncomp_model.compile(optimizer=opt, loss=loss, loss_weights=loss_weighting, metrics=metrics)
            model = multi_gpu_model(uncomp_model, gpus=args.gpus)
            model.__setattr__('callback_model', uncomp_model)
        model.compile(optimizer=opt, loss=loss, loss_weights=loss_weighting, metrics=metrics)

X = array[:,0:8]
Y = array[:,8]
test_size = 0.33
seed = 7
X_train, X_test, Y_train, Y_test = model_selection.train_test_split(X, Y, test_size=test_size,
random_state=seed)
```

In [ ]:
```
print("Len x-train : ", len(x_train))
print("Len x-test : ", len(x_test))
```

```
Len x-train :  18
Len x-test :  3
```

In [ ]:
```
# The Class Indices in Training Dataset
x_train.class_indices
```

Out[ ]:
```
{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}
```

Model Compilation

In [ ]:
```
# Importing Libraries
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Convolution2D,MaxPooling2D,Flatten,Dense
```

In [ ]:
```
# Creating Model
model=Sequential()
```

In [ ]:
```
# Adding Layers
model.add(Convolution2D(32,(3,3),activation='relu',input_shape=(64,64,3)))
```

In [ ]:
```
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())
```

In [ ]:
```
# Adding Dense Layers
model.add(Dense(300,activation='relu'))
model.add(Dense(150,activation='relu'))
model.add(Dense(9,activation='softmax'))
```

In [ ]:
```
# Compiling the Model
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
```

In [ ]:
```
# reading code from a file
f = open('main.py', 'r')
temp = f.read()
f.close()

code = compile(temp, 'main.py', 'exec')
exec(code)
```

Saving the Model

In [ ]:
```
model.save('asl_model_84_54.h5')
```

Fit And Save The Model

In [ ]:
```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

In [ ]:
```
# Training Datagen
train_datagen = ImageDataGenerator(rescale=1/255,zoom_range=0.2,horizontal_flip=True,vertical_flip=False)
# Testing Datagen
test_datagen = ImageDataGenerator(rescale=1/255)
```

In [ ]:
```
# Training Dataset
x_train=train_datagen.flow_from_directory(r'/content/drive/MyDrive/Dataset/training_set',target_size=(64,64), class_mode='categorical',batch_size=900)
# Testing Dataset
x_test=test_datagen.flow_from_directory(r'/content/drive/MyDrive/Dataset/test_set',target_size=(64,64), class_mode='categorical',batch_size=900)
```

```
Found 15760 images belonging to 9 classes.
Found 2250 images belonging to 9 classes.
```

In [ ]:
```
# Save Model Using Pickle
import pandas
from sklearn import model_selection
from sklearn.linear_model import LogisticRegression
import pickle
```

In [ ]:
```
url = "https://raw.githubusercontent.com/jbrownlee/Datasets/master/pima-indians-
diabetes.data.csv"
names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']
dataframe = pandas.read_csv(url, names=names)
array = dataframe.values
X = array[:,0:8]
Y = array[:,8]
test_size = 0.33
seed = 7
X_train, X_test, Y_train, Y_test = model_selection.train_test_split(X, Y, test_size=test_size,
random_state=seed)
```

In [ ]:
```
# Fit the model on training set
model = LogisticRegression()
model.fit(X_train, Y_train)
# save the model to disk
filename = 'finalized_model.sav'
pickle.dump(model, open(filename, 'wb'))

# load the model from disk
loaded_model = pickle.load(open(filename, 'rb'))
result = loaded_model.score(X_test, Y_test)
print(result)
```

In [ ]:
```
# Fitting the Model Generator
model.fit_generator(x_train,steps_per_epoch=len(x_train),epochs=10,validation_data=x_test,validation_steps=len(x_test))
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:2: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future versio
n. Please use `Model.fit`, which supports generators.

Epoch 1/10
18/18 [==============================] - 92s 5s/step - loss: 0.0049 - accuracy: 0.9994 - val_loss: 0.2635 - val_accuracy: 0.9773
Epoch 2/10
18/18 [==============================] - 90s 5s/step - loss: 0.0040 - accuracy: 0.9995 - val_loss: 0.2074 - val_accuracy: 0.9773
Epoch 3/10
18/18 [==============================] - 87s 5s/step - loss: 0.0041 - accuracy: 0.9995 - val_loss: 0.2460 - val_accuracy: 0.9773
Epoch 4/10
18/18 [==============================] - 91s 5s/step - loss: 0.0041 - accuracy: 0.9992 - val_loss: 0.2470 - val_accuracy: 0.9782
Epoch 5/10
18/18 [==============================] - 88s 5s/step - loss: 0.0037 - accuracy: 0.9993 - val_loss: 0.2439 - val_accuracy: 0.9782
Epoch 6/10
18/18 [==============================] - 88s 5s/step - loss: 0.0024 - accuracy: 0.9997 - val_loss: 0.2852 - val_accuracy: 0.9782
Epoch 7/10
18/18 [==============================] - 91s 5s/step - loss: 0.0023 - accuracy: 0.9997 - val_loss: 0.2589 - val_accuracy: 0.9782
Epoch 8/10
18/18 [==============================] - 93s 5s/step - loss: 0.0014 - accuracy: 1.0000 - val_loss: 0.2523 - val_accuracy: 0.9782
Epoch 9/10
18/18 [==============================] - 92s 5s/step - loss: 0.0013 - accuracy: 0.9999 - val_loss: 0.2269 - val_accuracy: 0.9778
Epoch 10/10
18/18 [==============================] - 91s 5s/step - loss: 0.0012 - accuracy: 0.9999 - val_loss: 0.2968 - val_accuracy: 0.9782
```

Out[ ]:

Saving the Model

In [ ]:
```
model.save('asl_model_84_54.h5')
```

In [ ]:
```
print("Len x-train : ", len(x_train))
print("Len x-test : ", len(x_test))
```

```
Len x-train :  18
Len x-test :  3
```

In [ ]:
```
# The Class Indices in Training Dataset
x_train.class_indices
```

Out[ ]:
```
{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}
```

Model Creation

In [ ]:
```
# Importing Libraries
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Convolution2D,MaxPooling2D,Flatten,Dense
```

In [ ]:
```
# Creating Model
model=Sequential()
```

In [ ]:
```
# Adding Layers
model.add(Convolution2D(32,(3,3),activation='relu',input_shape=(64,64,3)))
```

In [ ]:
```
model.add(MaxPooling2D(pool_size=(2,2)))
```

In [ ]:
```
model.add(Flatten())
```

In [ ]:
```
# Adding Dense Layers
model.add(Dense(300,activation='relu'))
model.add(Dense(150,activation='relu'))
model.add(Dense(9,activation='softmax'))
```

In [ ]:
```
# Compiling the Model
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
```

8.TESTING

8.1Test Cases

Loading the Dataset & Image Data Generation

In [14]:
```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

In [15]:
```
# Training Datagen
train_datagen = ImageDataGenerator(rescale=1/255,zoom_range=0.2,horizontal_flip=True,vertical_flip=False)
# Testing Datagen
test_datagen = ImageDataGenerator(rescale=1/255)
```

In [25]:
```
# Training Dataset
x_train=train_datagen.flow_from_directory(r'C:\Users\india\Desktop\Final_Project\Dataset\test_set',target_size=(64,64), class_mode='categorical',batch
# Testing Dataset
x_test=test_datagen.flow_from_directory(r'C:\Users\india\Desktop\Final_Project\Dataset\training_set',target_size=(64,64), class_mode='categorical',bat
```

```
Found 4969 images belonging to 9 classes.
Found 4969 images belonging to 9 classes.
```

In [26]:
```
print("Len x-train : ", len(x_train))
print("Len x-test : ", len(x_test))
```

```
Len x-train :  6
Len x-test :  6
```

In [27]:
```
# The Class Indices in Training Dataset
x_train.class_indices
```

Out[27]:
```
{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}
```

Model Creation

In [28]:
```
# Importing Libraries
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Convolution2D,MaxPooling2D,Flatten,Dense
```

In [29]:
```
# Creating Model
model=Sequential()
```

In [30]:
```
# Adding Layers
model.add(Convolution2D(32,(3,3),activation='relu',input_shape=(64,64,3)))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())

# Adding Hidden Layers
model.add(Dense(300,activation='relu'))
model.add(Dense(150,activation='relu'))

# Adding Output Layer
model.add(Dense(9,activation='softmax'))
```

In [31]:
```
# Compiling the Model
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
```

In [32]:
```
# Fitting the Model Generator
model.fit(x_train,steps_per_epoch=len(x_train),epochs=10,validation_data=x_test,validation_steps=len(x_test))
```

```
Epoch 1/10
6/6 [==============================] - 23s 4s/step - loss: 5.1206 - accuracy: 0.1690 - val_loss: 3.6505 - val_accuracy: 0.3119
Epoch 2/10
6/6 [==============================] - 22s 4s/step - loss: 2.3945 - accuracy: 0.3266 - val_loss: 1.5087 - val_accuracy: 0.4991
Epoch 3/10
6/6 [==============================] - 22s 4s/step - loss: 1.4384 - accuracy: 0.4037 - val_loss: 1.0430 - val_accuracy: 0.5836
Epoch 4/10
6/6 [==============================] - 23s 4s/step - loss: 1.0761 - accuracy: 0.6488 - val_loss: 0.7109 - val_accuracy: 0.7955
Epoch 5/10
6/6 [==============================] - 27s 5s/step - loss: 0.7835 - accuracy: 0.7774 - val_loss: 0.4046 - val_accuracy: 0.9501
Epoch 6/10
6/6 [==============================] - 25s 5s/step - loss: 0.5470 - accuracy: 0.8756 - val_loss: 0.2540 - val_accuracy: 0.9752
Epoch 7/10
6/6 [==============================] - 22s 4s/step - loss: 0.4018 - accuracy: 0.9090 - val_loss: 0.1675 - val_accuracy: 0.9799
Epoch 8/10
6/6 [==============================] - 22s 4s/step - loss: 0.2862 - accuracy: 0.9406 - val_loss: 0.1185 - val_accuracy: 0.9847
Epoch 9/10
6/6 [==============================] - 22s 4s/step - loss: 0.2108 - accuracy: 0.9612 - val_loss: 0.0880 - val_accuracy: 0.9863
Epoch 10/10
6/6 [==============================] - 22s 4s/step - loss: 0.1548 - accuracy: 0.9738 - val_loss: 0.0736 - val_accuracy: 0.9843
```

Out[32]:

8.2 User Acceptance Testing

1. Purpose of Document

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

2. Defect Analysis

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

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	11	2	3	2	18
Duplicate	1	3	4	0	8
External	3	5	0	0	8
Fixed	12	2	5	22	41
Not Reproduced	0	1	0	0	1
Skipped	0	0	1	2	3
Won't Fix	0	4	1	1	7
Totals	27	17	14	27	86

3. Test Case Analysis

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

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	8	0	0	8
Client Application	49	0	0	49
Security	4	0	0	4

Outsource Shipping	4	0	0	4
Exception Reporting	11	0	0	11
Final Report Output	2	0	0	2
Version Control	1	0	0	1

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	11	2	3	2	18
Duplicate	1	3	4	0	8
External	3	5	0	0	8
Fixed	12	2	5	22	41
Not Reproduced	0	1	0	0	1
Skipped	0	0	1	2	3
Won't Fix	0	4	1	1	7
Totals	27	17	14	27	86

3. Test Case Analysis

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

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	8	0	0	8
Client Application	49	0	0	49
Security	4	0	0	4

9. RESULTS

9.1 Performance Metrics

Technical Skills Evaluation Metrics

S. No	Metrics	Weightage (%)	Description	Sub-Evaluation Metrics & Scoring Criteria	Score	Description	Standard Template	Scoring Guidelines
1	**Technical Training & Assignments**	5%	This metric will be scored based on the following 1. Participation in the Quiz at the end of each training session 2. Submission of completed assignments	**Number of Tests / Quizzes Attempted (Total – 4 Nos.)** **Assignment-1** **Assignment-2** **Assignment-3** **Assignment-4**	Quiz will be of 40(10 type) with 10 questions 1 1 1 1	Weekly Assignment Weekly Assignment Weekly Assignment Weekly Assignment	No	Each Quiz Carries - 2.5 Score Completed - 2, Partial -1, Not Submitted -0 Completed - 2, Partial -1, Not Submitted -0 Completed - 2, Partial -1, Not Submitted -0 Completed - 2, Partial -1, Not Submitted -0
2	**Innovation & Problem Solving**	10%	This metric will be assessed and scored based on the unique innovative features incorporated in the proposed solution. The team shall submit the following templates 1. Define the Problem Statement 2. Empathize & Discover 3. Brainstorm & Prioritize Ideas	**1. Empathize & Discover** **2. Define the Problem Statement** **3. Brainstorm & Prioritize Ideas**	3 4 4	A problem statement often touches on the 5 w's (who, what, where, when, why) of the problem. Submit the template provided When designing a solution for a problem, it's important to understand who you're designing it for, so you can keep their needs, challenges, behaviors, and attitudes in mind while you design. Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving.	Yes Yes Yes	Completed - 3, Partial -1, Not Submitted -0 Completed - 4, Partial -2, Not Submitted -0 Completed - 4, Partial -2, Not Submitted -0
3	**Requirement Analysis using Critical Thinking**	10%	This metric will be scored based on the Submission of Project Requirement Analysis Template with valid information	**Generate the Requirements Customer Journey Maps** **Requirement Analysis (Functional, Operational, Technical) / Flow Charts** **Interpretation & Modelling**	4 3 3	Create customer journey maps to understand the requirements of customer Analyze the requirements, categorize them into Functional, Operational, and technical. Flow charts can be employed for each requirement. Decide which of the categorized requirements can be achieved and how they will be delivered. Find out and document what is expected, or wanted from the product, project, or application.	Yes Yes No	Completed - 4, Partial -2, Not Submitted -0 Completed - 3, Partial -1, Not Submitted -0 Completed - 2, Partial -1, Not Submitted -0
4	**Project Design using Design Thinking**	10%	This metric will be assessed and scored based on the submission of project design template which includes **1. Proposed solution** - Novelty - Feasibility of Concept - Social / Business Impact **2. Problem - Solution Fit** **3. Solution Architecture**	**Proposed solution** Novelty Feasibility of Concept Social / Business Impact **Problem - Solution Fit** **Solution Architecture**	3 4 3	Shall include the uniqueness of proposed solution, novelty, social or business impact and feasibility of concept. It is a template to help identify solutions with high chances of solution adoption, not a lot time spent on testing and get a better overview of the current situation Shall cover the architectural aspect of project from concept to deployment	Yes Yes Yes	Completed - 3, Partial -1, Not Submitted -0 Completed - 3, Partial -1, Not Submitted -0 Completed - 2, Partial -1, Not Submitted -0
5	**Technology Stack**	5%	This metric will be assessed and scored based on the selection of technology in building the proposed solution.	**Technical Architecture** **- Is the system robust** **- Is it highly modifiable?** **- Is it scalable?** **- Is it available (wrt time & budget constraints)** **Open Source Frameworks** **Third-party API's** **Cloud Deployment**	4 3 2 3	Detailed technical architecture of the solution shall be developed. Samples of code patterns will be introduced here for the reference Assess the utilization of open source frameworks in the solution Assess the number of APIs integrated in the solution Check whether the solution migrated on cloud or developed locally. Also check have they implemented micro-service architecture.	No No No No	Not Submitted – 0, Partial – 1 to 2, Complete – 4 Open source frameworks used Only one - 1 Mark More than 2 - 2 Marks 1-2 APIs - 1 Mark >2 APIs - 3 Mark Simple Architecture – 1 Containerized Containers – 2
6	**Project Planning using Agile Methodologies**	5%	This metric will be assessed and scored based on the detailed project planning templates submitted by the project team. The template shall cover 1. Project Milestones & Tasks 2. Sprint Delivery Schedule 3. Distribution of Tasks among team members 4. Team management tool for agile planning	**Project Milestones & Tasks** **Sprint Delivery Plan** **Project Progress Tracking** **Team management tool for agile planning (e.g. Trello, Jira, etc.)**	4 3 2 1	Ensure the detailed planning of project milestones, the most important features, expected outputs, amount of work, list stories & epics, Acc iterations Sprint planning is an event to ensure that teams off to a sprint. The purpose of sprint planning is to define what can be delivered in the sprint Distribution of tasks to team members, Prioritization of tasks. Track the progress of all tasks. Submit velocity & burn down chart Utilization of project / team management tools for the project delivery	Yes Yes No No	Not Submitted – 0, Partial – 1 to 2, Complete – 4 Number of Sprint planning meetings organized, Minutes of meeting recorded Scale up Data 3 – 4, 2-3 – 2, 1 – 1 Not Submitted – 0, Partial – 1, Complete – 2 No tools used – 0 marks Any one tool – 2 marks
7	**Coding & Autodocking**	15%	This metric will be scored based on the following parameters of the code 1.No. of Functional Features 2.Code Efficiency 3.Utilization of algorithms 4.Coding Layout / Self-explanation & re-usability 5.Debugging & Traceability 6.Exception Handling	**No. of Functional Features included in the solution** **Code Layout, Readability and Reusability** **Utilization of Algorithms, Dynamic Programming, Optimal Memory Utilization** **Debugging & Traceability** **Exception Handling**	3 3 3 3 3	Assess the feature presented vs. implemented Assess the code layout, readability and reusability Assess the utilization of Algorithms, Dynamic Programming, Optimal Memory Utilization Assess the Debugging & Traceability Capability in the core Assess the Exception Handling capability of the code developed	No No No No No	0 - 33% Mapping - 0 Marks 31 – 70 % Mapping - 1 Mark >70% Mapping- 2 Marks Average - 0, Good - 1, Excellent - 2 Average - 0, Good - 1, Excellent - 2 Average - 0, Good - 1, Excellent - 2 Average - 0, Good - 1, Excellent - 2
8	**Acceptance Testing**	5%	This metric will be assessed and scored based on the following 1.UAT Initiation 2.UAT Design 3.UAT Execution 4.UAT Report Submission 5.Utilization of Test Management Tools	**UAT Initiation** **UAT Design** **UAT Execution** **UAT Report Submission** **Utilization of Test management Tools**	2 2 2 2 2	Ensure the availability of UAT checklist (Gates Approach, test data requirements, etc.) Assess the UAT design document Assess the list of bugs identified and fixes implemented Assess the report submitted Assess the utilization of Tools like Gitea, Apollon, Listit e.t.c etc.	No No No Yes No	Not Submitted – 0, Partial – 1, Complete - 2 Not Submitted – 0, Partial – 1, Complete - 2 Not Submitted – 0, Partial – 1, Complete - 2 Not Submitted – 0, Partial – 1, Complete - 2 Not Submitted – 0, Partial – 1, Complete - 2
9	**Performance Testing**	5%	This metric will be assessed and scored based on the following 1.Application Performance (for Cloud & IoT Projects) 2.Model Performance (for ML/AI Projects) 3.Dashboard & User Stories (for Data Analytics Projects)	**Application Performance Metrics** Error rates Response times Request rates Customer experience **Model Performance Metrics** Confusion matrix Accuracy Precision Recall Specificity F1 score Precision-Recall or PR curve ROC (Receiver Operating Characteristic) curve PR or ROC score **Dashboard & User Story Metrics** Dashboard design Data Responsiveness Amount Data is Rendered Utilization of Data Filters Correlation Efficiency Effective User Story	10 10 10	Assess the application performance metrics submitted or verify using performance testing tools Assess the model performance metrics submitted or verify using model performance testing tools Assess the visualization dashboard based on the metrics provided	Yes Yes Yes	Below Average - 3 - 25% Satisfactory – 25% - 50% Good – 51% - 75% Excellent - 76% - 100% Below Average - 3 - 25% Satisfactory – 25% - 50% Good – 51% - 75% Excellent - 76% - 100% Below Average - 3 - 25% Satisfactory – 25% - 50% Good – 51% - 75% Excellent - 76% - 100%

10. ADVANTAGES & DISADVANTAGES

Advantages:

- It is a cost-effective way of getting several people from differentlocations to attend meetings and conferences.
- It enables employees from across the world to communicate with each other 24×7 and share ideas or solve problems quickly.

Disadvantages:

- Also accuracy depends upon distance between camera andobject.
- It takes a lot of time to listen, speak, read, or write to someone.

11.CONCLUSION

The proposed communication system between Deaf and Dumb peopleand ordinary people are aiming for it when bridging the communication gap between two societies. It provides complete two -sided communication in an efficient manner between the disabled andthe normal person.

For communication between deaf person and a second person, a mediator is required to translate sign language of deaf person. But a mediator is required to know the sign language used by deaf person. But this is not always possible since there are multiple sign languages for multiple languages

So to understand all sign languages, Hand gestures of deaf peoples by normal peoples this system is proposed.

12.FUTURE SCOPE

The speech-to-text and text-to-speech technologies helped those people who had difficulties in communicating or expressing their feelings to the normal people.

This reduces the communication gap between the normal people andthe specially abled people.

Using image pre-processing and Artificial Intelligence it is easy to understand the context of objects and clearly explains it to the peoplewho use it for communication.

13 APPENDIX:

SourceCode

```
<div id="content" style="margin-top:2em">
<div class="container">
  <div class="row">
    <div class="col-sm-6 bd">
      <h2 color="white">Real Time Communication System Powered By AI For Specially Abled: </h2>
      <b>
      <br>
      <p> Designing and implementing a system using artificial intelligence, Deep Learning algorithms and image processing concepts to take input as hand gestures (or) sign language and It gene
      </b>
        <img src="https://png.pngtree.com/element_our/png_detail/20181129/aphabets-by-fingers-from-american-sign-language-png_251331.jpg" height="50%",width="20%">
    </div>
    <div class="col-sm-6">
        <div>
            <h3>Upload Image Here To Identify the Sign Language</h3>
    <form action = "C:\Users\aishu\Desktop\login page1.html" id="upload-file" method="post" enctype="multipart/form-data">
        <label for="imageUpload" class="upload-label">
            Choose...
        </label>
        <input type="file" name="image" id="imageUpload" accept=".png, .jpg, .jpeg">
    </form>


    <div class="image-section" style="display:none;">
        <div class="img-preview">
            <div id="imagePreview">
            </div>
        </div>
        <div>
            <button type="button" class="btn btn-info btn-lg " id="btn-predict">Predict!</button>
        </div>
    </div>

    <div class="loader" style="display:none;"></div>

    <h3>
        <span id="result"> </span>
    </h3>
<img src="C:\Users\aishu\Desktop\New folder\training_set\A\1.png"height="50%",width="20%">
</div>
    </div>

  </div>
</div>
</div>
</div>
</body>

<footer>
    <script src="{{ url_for('static', filename='cfile.js') }}" type="text/javascript"></script>
```

```
}
 table{text-align: center;}

    </style>
</head>
    <body>
    <h1> <nav class="red">
         <div class="container">
<center>
             <a class="pink" href="#"><font color="pink">  Real Time Communication System Powered By AI For Specially Abled Using CNN</font></a>
</center>
        </div>
     </nav>
     </h1>
     <br>
      <br>
        <form action="C:\Users\aishu\Desktop\login page1.html"method="get">
           <div>
              <table align="center">

<h1>
                <br>
                <br>
                        <tr>
                    <td>Username</td>
                    <td><input type='text' name="uname"></td>
                </tr>
                <br>
                <br>
                        <tr>
                    <td>Password</td>
                    <td><input type='password' name="pass"></td>
                </tr>
                <br>
                <br>
</h1>
                        <tr>
                    <td>
                        
                    <a class ="button" href="index page11.html">click here</a></td>
                </tr>

            </table>
            <p>                                     &
 <img src="https://th.bing.com/th/id/R.7d1ae245d03510eb47b035b22221c808?rik=p6PzkRhZi878pw&riu=http%3a%2f%2fcdn.greatnonprofits.org%2fimages%2flogos%2fNoLimitsNewLogo2017.png&ehk=Lb3kDoL3mgvqrDNQGqN%2f
            </p> </div>
        </form>
    </body>
</html>
```

```html
}
 table{text-align: center;}

    </style>
</head>
    <body>
    <h1> <nav class="red">
         <div class="container">
<center>
              <a class="pink" href="#"><font color="pink">  Real Time Communication System Powered By AI For Specially Abled Using CNN</font></a>
</center>
        </div>
     </nav>
     </h1>
     <br>
      <br>
        <form action="C:\Users\aishu\Desktop\login page1.html"method="get">
           <div>
               <table align="center">

<h1>
                <br>
                <br>
                        <tr>
                    <td>Username</td>
                    <td><input type='text' name="uname"></td>
                </tr>
                <br>
                <br>
                        <tr>
                    <td>Password</td>
                    <td><input type='password' name="pass"></td>
                </tr>
                <br>
                <br>
</h1>
                        <tr>
                    <td>
                        
                    <a class ="button" href="index page11.html">click here</a></td>
                </tr>

            </table>
            <p>                                     &
 <img src="https://th.bing.com/th/id/R.7d1ae245d03510eb47b035b22221c808?rik=p6PzkRh7i878pw&riu=http%3a%2f%2fcdn.greatnonprofits.org%2fimages%2flogos%2fNoLimitsNewLogo2017.png&ehk=Lb3kDoL3mgvqrDNQGqN%2f
           </p> </div>
        </form>
    </body>
</html>
```

```html
    </h1>
    <br>
     <br>
<u>
        <h1> Hi ! User </h1>
        <center>
        <h2> You have login successfully</h2>
        </center>
        </u>
        <u>
    <b>
     <br><font color="white">
     <center>
        <h2>   Welcome you to The Real Time Communication System Powered By AI For Spcially Abled</h2></font>
        </center>
         <br>
         </b>
         </u>
         <b>
        <h2><div><p>  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.
         Communications between deaf-mute and a normal person has always been a challenging task. </p>
         <br>
         <p>  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
         The human hand has remained a popular choice to convey information in situations where other forms like speech cannot be used. </p>
          <br>
         <p>  Text Conversion System with 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 text in the desired language to convey a message to normal people.</p>
          <br>
          <p>  We are making use of a convolution neural network to create a model that is trained on different hand gestures. An Web Application is built which uses this model. This app enables de
         people to convey their information using signs which get converted to human-understandable language and text is given as output.</p></div>
         <br> <br>
          <br>
        <div>
       </h2></b> <p>                                    &
        <img src="https://leewilsonaccess.files.wordpress.com/2014/03/cartoon-with-multiple-peopel-standing-in-a-row-titled-not-all-disabilities-are-visible.png?w=1024==" height="60%",width="30%">
         <br>
          <br><br>
          <h2>
        <p>            Click the below link to be directed to The Real Time Communication System Powered By AI For Spcially Abled</p>
        </h2>
        </h4>
        </h4>
         <br>
        <h1><a class ="button" href="after login1.html">Click Here</a></h1>
         <br>
        </div>
    </body>
</html>
```

```
<html>
<head>
<title>Home page</title>
 <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <link href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.cfile.css" rel="stylesheet">
    <script src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.cfile.js"></script>
    <script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.cfie.js"></script>
    <script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.cfile.js"></script>
    <link href="/static/cfile.css" rel="stylesheet">
<style>

    .bg-dark {
        background-color:   #21618C!important;
    }
    #result {
        color: #ffffff;
    }
body
{
 background-image: url("https://i.pinimg.com/originals/58/28/c5/5828c5a104f147e30ef1bed19798c2a8.jpg");


    background-size: cover;
}
<style>
        h2{text-align: center;}
        h3{text-align: center;}
        img{text-align: center;}
         h4{text-align: center;}
        h1{text-align: center;}
        p{text-align: justify;}
    </style>
    </style>
</head>
    <body>
   <h1> <cya class="Cyan">
        <div class="container">
        <u>
            <a class="pink" href="#"><font color="pink">  Real Time Communication System Powered By AI For Specially Abled Using CNN</font></a>
            </u>
        </div>
    </cya>
    </h1>
```

Line wrap

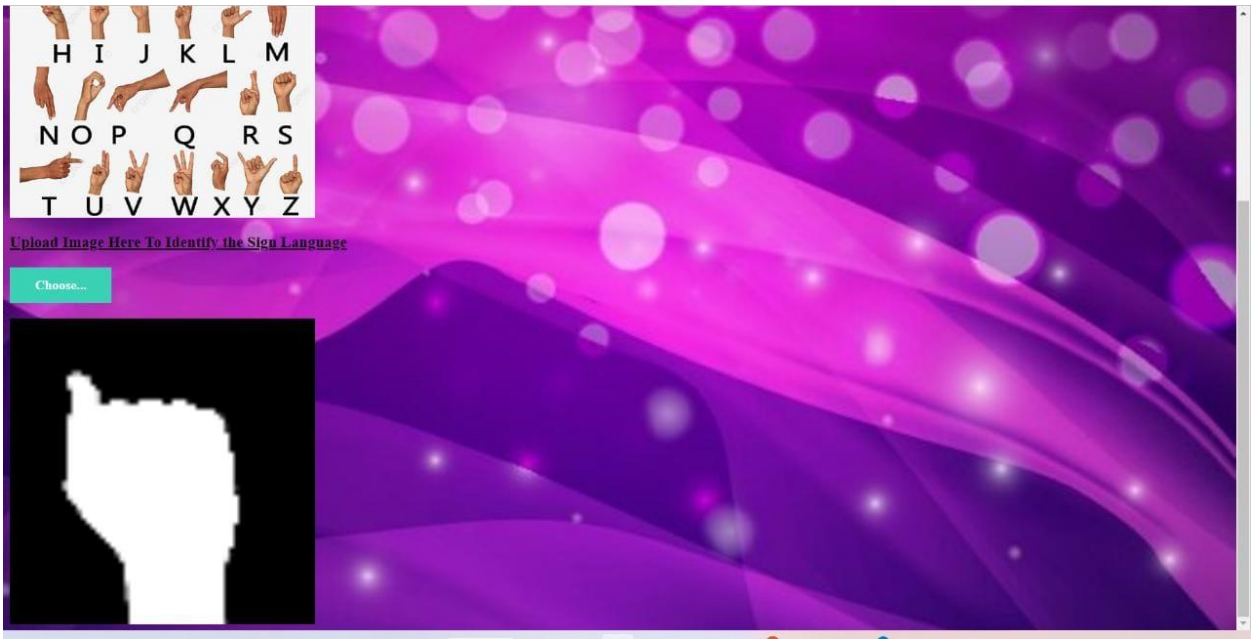
```
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Real Time Communication System Powered By AI For Specially Abled</title>
    <link href="C:\Users\aishu\Desktop\cfile.css" rel="stylesheet">
    <script src="C:\Users\aishu\Desktop\cfile.js"></script>
    <script src="C:\Users\aishu\Desktop\cfile.js"></script>
    <script src="C:\Users\aishu\Desktop\cfile.js"></script>
    <link href="/static/cfile.css" rel="stylesheet">
<style>

    .bg-dark {
        background-color:   #21618C!important;
    }
    #result {
        color: #ffffff;
    }
    body
{
    background-image: url("https://i.pinimg.com/originals/58/28/c5/5828c5a104f147e30ef1bed19798c2a8.jpg");

    background-size: cover;
}

    </style>
</head>

<body>

    <h1>
<b>
<u>
<center>
    <nav class="">
        <div class="container">
<a class="yellow" href="#">Real Time Communication System Powered By AI For Specially Abled Using CNN</a>
        </div>
        </center>
<u>
<b>
</h1>

    </nav>
    <div class="container">
        <div id="content" style="margin-top:2em">
```

```
        <div id="content" style="margin-top:2em">
        <div class="container">
          <div class="row">
            <div class="col-sm-6 bd">
              <h2 color="white">Real Time Communication System Powered By AI For Specially Abled: </h2>
              <b>
              <br>
              <p> Designing and implementing a system using artificial intelligence, Deep Learning algorithms and image processing concepts to take input as hand gestures (or) sign language and It gene
              </b>
                <img src="https://png.pngtree.com/element_our/png_detail/20181129/aphabets-by-fingers-from-american-sign-language-png_251331.jpg" height="50%",width="20%">
            </div>
            <div class="col-sm-6">
                <div>
                    <h3>Upload Image Here To Identify the Sign Language</h3>
            <form action = "C:\Users\aishu\Desktop\login page1.html" id="upload-file" method="post" enctype="multipart/form-data">
                <label for="imageUpload" class="upload-label">
                    Choose...
                </label>
                <input type="file" name="image" id="imageUpload" accept=".png, .jpg, .jpeg">
            </form>


            <div class="image-section" style="display:none;">
                <div class="img-preview">
                    <div id="imagePreview">
                    </div>
                </div>
                <div>
                    <button type="button" class="btn btn-info btn-lg " id="btn-predict">Predict!</button>
                </div>
            </div>

            <div class="loader" style="display:none;"></div>

            <h3>
                <span id="result"> </span>
            </h3>
<img src="C:\Users\aishu\Desktop\New folder\training_set\A\1.png"height="50%",width="20%">
        </div>
            </div>

          </div>
        </div>
        </div>
    </div>
</body>

<footer>
    <script src="{{ url_for('static', filename='cfile.js') }}" type="text/javascript"></script>
```


```
        <div id="content" style="margin-top:2em">
        <div class="container">
          <div class="row">
            <div class="col-sm-6 bd">
              <h2 color="white">Real Time Communication System Powered By AI For Specially Abled: </h2>
              <b>
              <br>
              <p> Designing and implementing a system using artificial intelligence, Deep Learning algorithms and image processing concepts to take input as hand gestures (or) sign language and It gene
              </b>
                <img src="https://png.pngtree.com/element_our/png_detail/20181129/aphabets-by-fingers-from-american-sign-language-png_251331.jpg" height="50%",width="20%">
            </div>
            <div class="col-sm-6">
                <div>
                    <h3>Upload Image Here To Identify the Sign Language</h3>
            <form action = "C:\Users\aishu\Desktop\login page1.html" id="upload-file" method="post" enctype="multipart/form-data">
                <label for="imageUpload" class="upload-label">
                    Choose...
                </label>
                <input type="file" name="image" id="imageUpload" accept=".png, .jpg, .jpeg">
            </form>


            <div class="image-section" style="display:none;">
                <div class="img-preview">
                    <div id="imagePreview">
                    </div>
                </div>
                <div>
                    <button type="button" class="btn btn-info btn-lg " id="btn-predict">Predict!</button>
                </div>
            </div>

            <div class="loader" style="display:none;"></div>

            <h3>
                <span id="result"> </span>
            </h3>
<img src="C:\Users\aishu\Desktop\New folder\training_set\A\1.png"height="50%",width="20%">
        </div>
            </div>

          </div>
        </div>
        </div>
    </div>
</body>

<footer>
    <script src="{{ url_for('static', filename='cfile.js') }}" type="text/javascript"></script>
```


Real Time Communication System Powered By AI For Specially Abled Using CNN
Username
Aishu
Password
click here
I CAN DO IT!
No Limits
for deaf children and families


NOT ALL DISABILITIES ARE VISIBLE
Click the below link to be directed to The Real Time Communication System Powered By AI For Specially Abled
Click Here



GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-14795-1659589960