Real-Time Communication System Powered byAI for Specially Abled

TEAM ID: PNT2022TMID26243

P.PRIYAADARSHAN TEAM LEADER	211519205116
S. YUVARAJ	211519205184
SURYAPRAKASH K S	211519205164
LOKESHWARAN.V	211519205088
THARUN KUMAR R	211519205173

CONTENTS

1. INTRODUCTION:

1.1 Project Overview:

The improvement of communication is essential to a better world. People become more bonded and connected as a result. People with disabilities exist in our society. Though technology is constantly evolving, no big advancements are being made for the benefit of these people. It has always been difficult to communicate with a deaf-mute and a normal person. Mute persons find it incredibly challenging to communicate with non-mute people. due to the fact that most people are not trained in hand sign language. It is exceedingly difficult for them to communicate during times of emergency. In circumstances where other modes of communication, like speech, are not possible, the human hand has remained a common alternative for information transmission. To have a proper communication between a normal person and a handicapped person in any language, a voice conversion system with hand gesture recognition and translation will be very helpful. The project intends to create a system that can translate speech into acceptable sign language for the deaf and dumb as well as translate sign language into a human hearing voice in the desired language to communicate a message to normal people. A convolution neural network is being used to build a model that is trained on various hand motions. On the basis of this model, an app is created.

1.2 Purpose:

The goal of the project is to create a system that can translate speech into understandable sign language for the deaf and dumb. It also seeks to construct a system that can translate sign language into a human hearing voice in the appropriate language to communicate a message to normal people. This project is developed to support the following parameters

- Create a barrier-free environment for people with disabilities and the general public.

- Making use of technology to solve societal issues.
- Making a contribution to the society's inclusive growth.
- To influence/change society in a way that is positive.

2. LITERATURE SURVEY:

2.1 Existing Problems:

In this world , it has always been difficult to communicate with someone who is deaf-mute. It is quite challenging for silent persons to communicate with non-mute people. because hand sign language is not taught to the general public. Only those with special needs are taught sign language, and because the average person has no idea how it works, there is a communication gap. Specially abled individuals find it extremely harder to receive aid in an emergency. They may also need help to traverse non-emergency typical situations.

The following study shows the technology advancements made in this aspect.

[1] Title	Messaging and Video Calling Application for Specially Abled people using Hand Gesture Recognition
Author:	Rachana R. Chhajed , Komal P. Parmar , Manvi D. Pandya , Neha G. Jaju
Journal / Conference	2021 6th International Conference for Convergence in Technology (I2CT) Pune, India. Apr 02-04, 2021
Methodology:	The app will basically convert the sign language to text while messaging and to voice while video calling for a normal person and vice versa for deaf-mute people. It uses vision-based approach CNN applied to ISL Dataset and for object detection YOLO v4 has been used. The application will recognize deaf-mute speech samples of alphabets (A–Z), digits (0 to 9) and various common sentences. Other different types of gestures can also be added to the database.
Advantage:	It will provide messaging and video calling service for both deaf-mutes and normal people.

Limitation:	scope can be extended for blind people

[2] Title	Predicting Sentiments to an accuracy matching the gesture recognized for the specially-abled
Author:	Jaganath Prasad Mohanty , Ayas Kanta Swain, Kamalakanta Mahapatra
Journal / Conference	2020 IEEE International Symposium on Smart Electronic Systems(iSES) (Formerly iNiS)
Methodology:	In this work, Natural Language Processing (NLP) is done to accurately recognize the sentiment behind the gestures of a specially abled individual, to analyse their behavior in real time considering different application platforms.
Advantage:	A new paradigm shifts in acknowledging the services offered by an all-inclusive world and consumer industry to specially abled public by making their lives more comfortable. It can be linked to industry inputs to provide services adherent to their needs.
Limitation:	classification for improved analysis and creating statements that match with the behaviour of the sign language user.

[3] Title	Portable Communication Aid for Specially Challenged : Conversion of Hand Gestures into Voice and Vice Versa
Author:	T Meera Devi, K M Shravan Raju
Journal / Conference	2018 International Conference on Intelligent Computing and Communication for Smart World (I2C2SW), 2018, pp. 306-310, doi: 10.1109/I2C2SW45816.2018.8997140.
Methodology:	The work is to develop a portable device for the disabled people who are not able to communicate with the normal persons properly. There are various steps involved in recognising the feature distinguishing hand gesticulation. The collected gesticulation is trained using Neural Network. The hand movement pattern is separated from a continuous recording of gestures. Low-Level understanding for the feature pattern comprises the gestural segment
Advantage:	This will be useful for the normal people to communicate with differently abled people and vice versa.

Limitation:	Separation of the hand movements from continuous hand gestures may result in accuracy issues.

[4] Title	Survey on sign language recognition in context of vision-based and deep learning
Author:	S. Subburaj , S. Murugavalli
Journal / Conference	Measurement: Sensors, Volume 23, 2022, 100385,ISSN 2665-9174, https://doi.org/10.1016/j.measen.2022.100385.
Methodology:	It examine the methods employed within the SLR systems, and the classification methods used, and to propose the most promising technique for future research. This paper specializes in the classification strategies utilized in earlier Sign Language Recognition.
Advantage:	shared a quantitative study of different methods used in sign language recognition

[5] Title	Artificial Intelligence enabled virtual sixth sense application for the disabled
Author:	Aditya Sharma |Aditya Vats |Shiv Shankar Dash|Surinder Kaur
Journal / Conference	Fusion: Practice and Applications (FPA), 1(1), 32–39. https://doi.org/10.5281/zenodo.3825929
Methodology:	The main highlight of the project is an application that provided a one-stop-shop solution to all the sections of differently-abled people. Integration has provided a seamless User interface/experience for the initial setup. Another point achieved here was no extra hardware; hence, no additional cost to utilize the service. The application still does depend on the camera picture quality for object detection and OCR but is still high enough in confidence level (70%) for most of the cases which were covered.
Advantage:	The major contribution of the work is an innovative approach for text to speech is implemented to provide a faster and convenient approach for mute to communicate through SAM (Speech Assisted for Mute).

[6] Title	Real-Time Sign Language Detection using TensorFlow, OpenCV and Python
Author:	Prashant Verma, Khushboo Badli
Journal / Conference	International Journal for Research in Applied Science & Engineering Technology (IJRASET) ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538 Volume 10 Issue V May 2022-
Methodology:	This project falls under the category of human-computer interaction (HCI) and tries to recognise multiple alphabets (a-z), digits (0-9) and several typical ISL hand gestures. To apply Transfer learning to the problem, we used a Pre-Trained SSD Mobile net V2 architecture trained on our own dataset.
Advantage:	a computer-based intelligent system that will allow deaf persons to interact effectively with others by utilising hand gestures based on technologies tensor flow, object detection, open cv, labelling
Limitation:	environmental conditions such as low light intensity and an unmanaged backdrop, which reduce detection accuracy.

2.2 References

1. Rachana R. Chhajed , Komal P. Parmar , Manvi D. Pandya , Neha G. Jaju , “ Messaging and Video Calling Application for Specially Abled people using Hand Gesture Recognition “, 2021 6th International Conference for Convergence in Technology (I2CT) Pune, India. Apr 02-04, 2021
2. Jaganath Prasad Mohanty , Ayas Kanta Swain, Kamalakanta Mahapatra , “ Predicting Sentiments to an accuracy matching the gesture recognized for the specially-abled “ , 2020 IEEE International Symposium on Smart Electronic Systems(iSES) (Formerly iNiS)

3. T. MeeraDevi and K. M. S. Raju, "Portable Communication Aid for Specially Challenged : Conversion of Hand Gestures into Voice and ViceVersa," 2018 International Conference on Intelligent Computing and Communication for Smart World (I2C2SW), 2018, pp. 306-310, doi: 10.1109/I2C2SW45816.2018.8997140.

4. S. Subburaj, S. Murugavalli, ,” Survey on sign language recognition in context of vision-based and deep learning,”, Measurement: Sensors, Volume 23, 2022, 100385,ISSN 2665-9174, https://doi.org/10.1016/j.measen.2022.100385.

5. Aditya Sharma , Aditya Vats , Shiv Shankar Dash and Surinder Kaur , (2020). Artificial Intelligence enabled virtual sixth sense application for the disabled. Fusion: Practice and Applications (FPA), 1(1), 32–39. https://doi.org/10.5281/zenodo.3825929

6. Prashant Verma, Khushboo Badli , “Real-Time Sign Language Detection using TensorFlow, OpenCV and Python “ , International Journal for Research in Applied Science & Engineering Technology (IJRASET) ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538 Volume 10 Issue V May 2022

7. Kothadiya, Deep & Bhatt, Chintan & Sapariya, Krenil & Patel, Kevin & Gil, Ana & Corchado Rodríguez, Juan. (2022). Deepsign: Sign Language Detection and Recognition Using Deep Learning. Electronics. 11. 1780. 10.3390/electronics11111780.

8. Prof. P.G. Ahire, K.B. Tilekary,T.A. Jawake, P.B. Warale, “Two Way Communicator between Deaf and Dumb People and Normal People", 978-1-4799-6892-3/15 31.00 c 2015 IEEE.

9. Shreyashi Narayan Sawant, "Sign Language recognition System to aid Deaf- dumb People Using PCA", IJCSET ISSN : 2229-3345 Vol. 5 No. 05 May 2014.

10. Matusiak, K., Skulimowski, P., & Strurniłło, P. (2013, June). Object recognition in a mobile phone application for visually impaired users. In 2013 6th International Conference on Human System Interactions (HSI) (pp. 479-484). IEEE.

2.3 Problem Statement Definition:

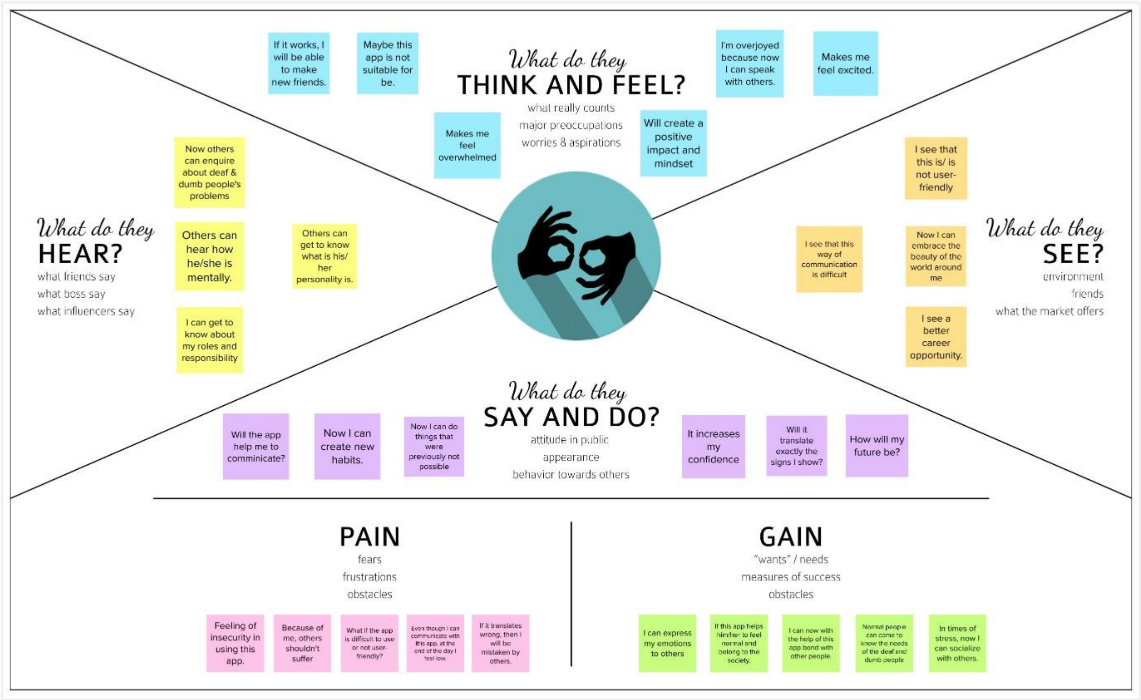
In our society, we have people with disabilities. The technology is developing day by day but no significant developments are undertaken for the betterment of these people. Communications between deaf-mute and a normal person has always been a challenging task. It is very difficult for mute people to 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 situations where 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.

Problem Statement (PS)	I am (Customer)	I’m trying to	But	Because	Which makes me feel
PS-1	Teacher	Present a topic	Using hand gesture	But Not able to talk	Disappointed , since not able to complete the topic on time.

PS-2	HR Manager	Interview a deaf and dumb person	Watching hand gestures of interview candidate	Unable to understand their hand gesture	Frustrated , unable to judge a candidate
PS-3	Radha (a deaf- mute person)	Express my feelings to others	I can't do so like the normal people	I can't communicate ina way they can easily comprehend	Depressedand sad
PS-4	Ahmed (a deaf-mute person)	Complain about an issueto a public authority	I find it difficultto do so	The public authority finds it difficult to understand me	Inferior and incompetent

3. IDEATION & PROPOSED SOLUTION:

3.1 Empathy Map Canvas:

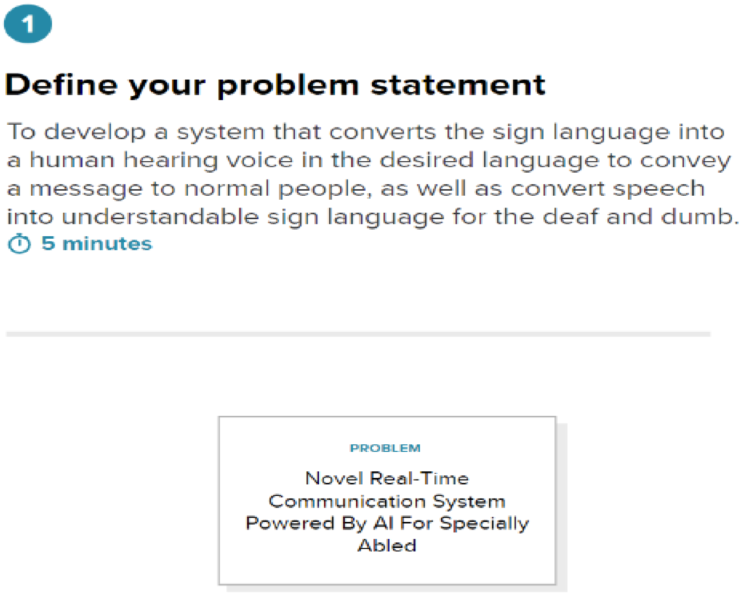


What do they
THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
If it works, I will be able to make new friends.
Maybe this app is not suitable for be.
I'm overjoyed because now I can speak with others.
Makes me feel excited.
Makes me feel overwhelmed
Will create a positive impact and mindset
What do they
HEAR?
what friends say
what boss say
what influencers say
Now others can enquire about deaf & dumb people's problems
Others can hear how he/she is mentally.
Others can get to know what is his/ her personality is.
I can get to know about my roles and responsibility
What do they
SEE?
environment
friends
what the market offers
I see that this is/ is not user-friendly
I see that this way of communication is difficult
Now I can embrace the beauty of the world around me
I see a better career opportunity.
What do they
SAY AND DO?
attitude in public
appearance
behavior towards others
Will the app help me to comminicate?
Now I can create new habits.
Now I can do things that were previously not possible
It increases my confidence
Will it translate exactly the signs I show?
How will my future be?
PAIN
fears
frustrations
obstacles
Feeling of insecurity in using this app.
Because of me, others shouldn't suffer
What if the app is difficult to use or not user-friendly?
Even though I can communicate with this app, at the end of the day I feel low.
If it translates wrong, then I will be mistaken by others.
GAIN
"wants" / needs
measures of success
obstacles
I can express my emotions to others
If this app helps him/her to feel normal and belong to the society.
I can now with the help of this app bond with other people.
Normal people can come to know the needs of the deaf and dumb people
In times of stress, now I can socialize with others.

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes.

It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. Creating the map helps participants consider things from the user's perspective along with his or her goals and challenges as shown in the above diagram

3.2 Ideation & Brainstorming:



1

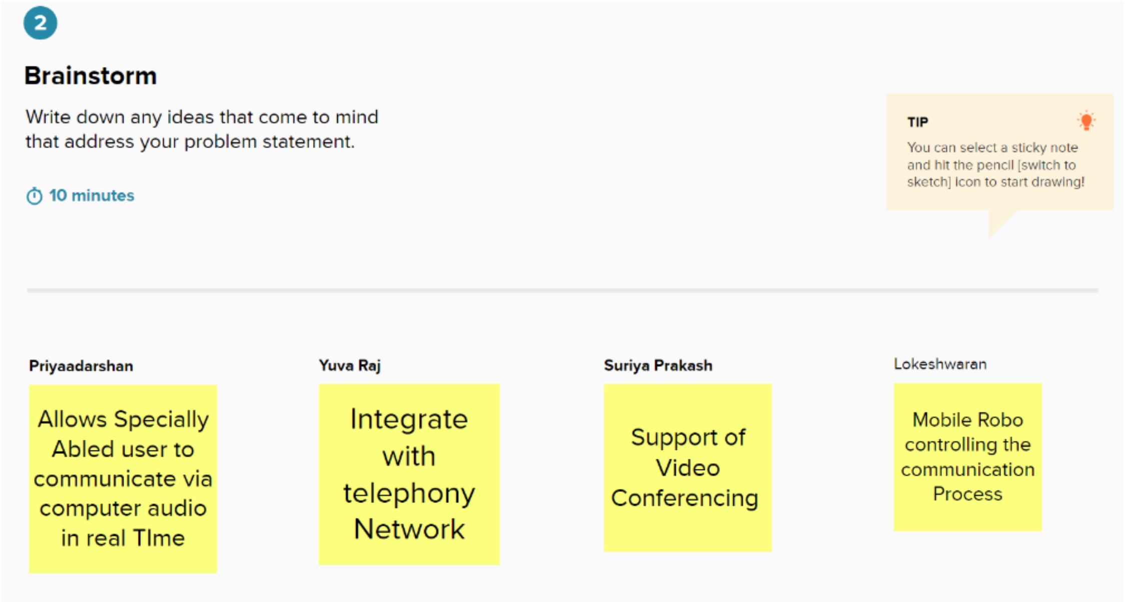
Define your problem statement

To develop a system that converts the sign language into 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.

5 minutes

PROBLEM

Novel Real-Time Communication System Powered By AI For Specially Abled



2

Brainstorm

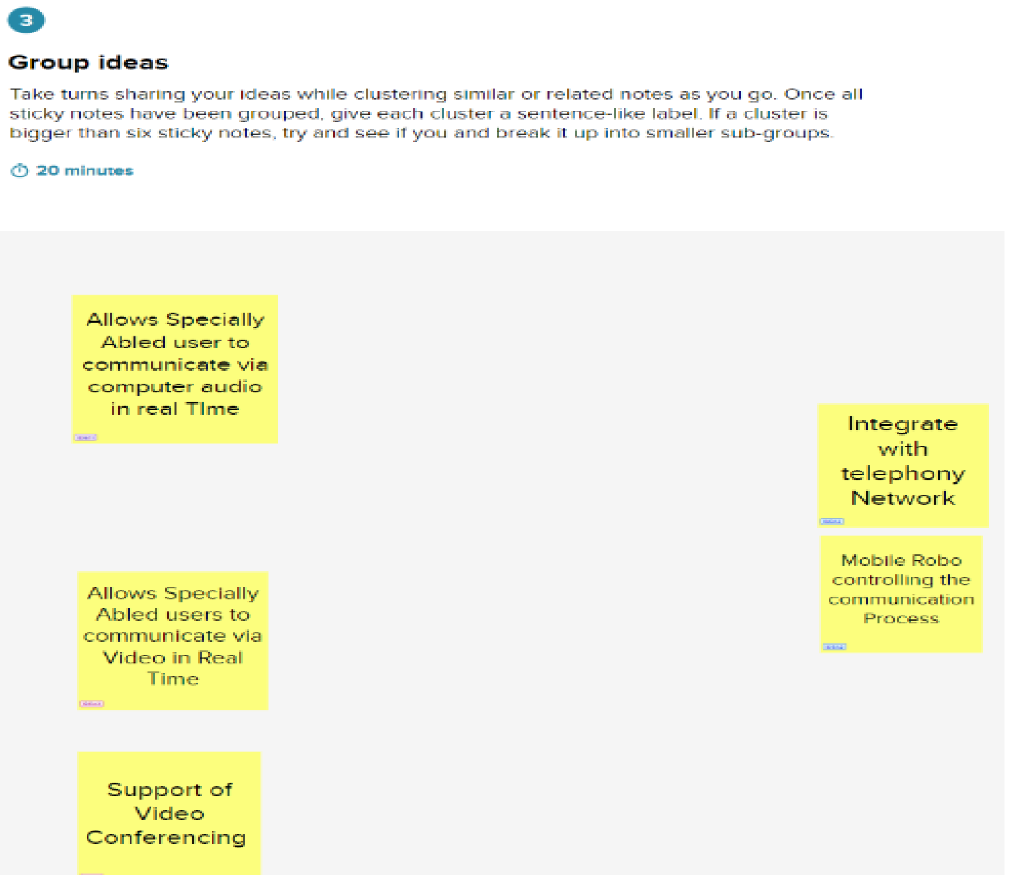
Write down any ideas that come to mind that address your problem statement.

10 minutes

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

Priyaadarshan	Yuva Raj	Suriya Prakash	Lokeshwaran
Allows Specially Abled user to communicate via computer audio in real TIme	Integrate with telephony Network	Support of Video Conferencing	Mobile Robo controlling the communication Process



3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

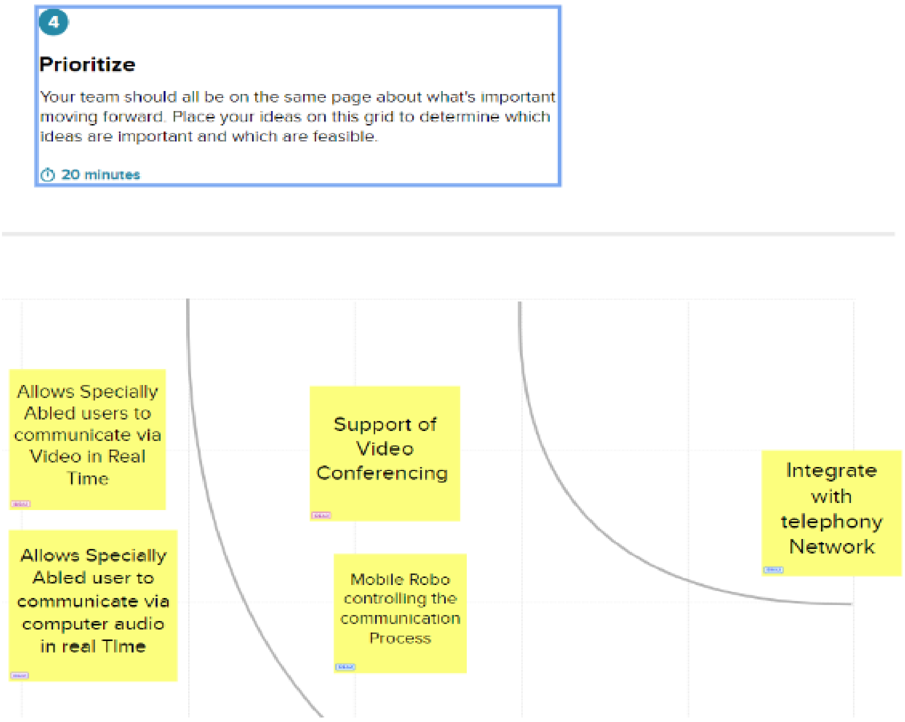
Allows Specially Abled user to communicate via computer audio in real TIme

Integrate with telephony Network

Mobile Robo controlling the communication Process

Allows Specially Abled users to communicate via Video in Real Time

Support of Video Conferencing



4

Prioritize

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

20 minutes

Allows Specially Abled users to communicate via Video in Real Time

Allows Specially Abled user to communicate via computer audio in real TIme

Support of Video Conferencing

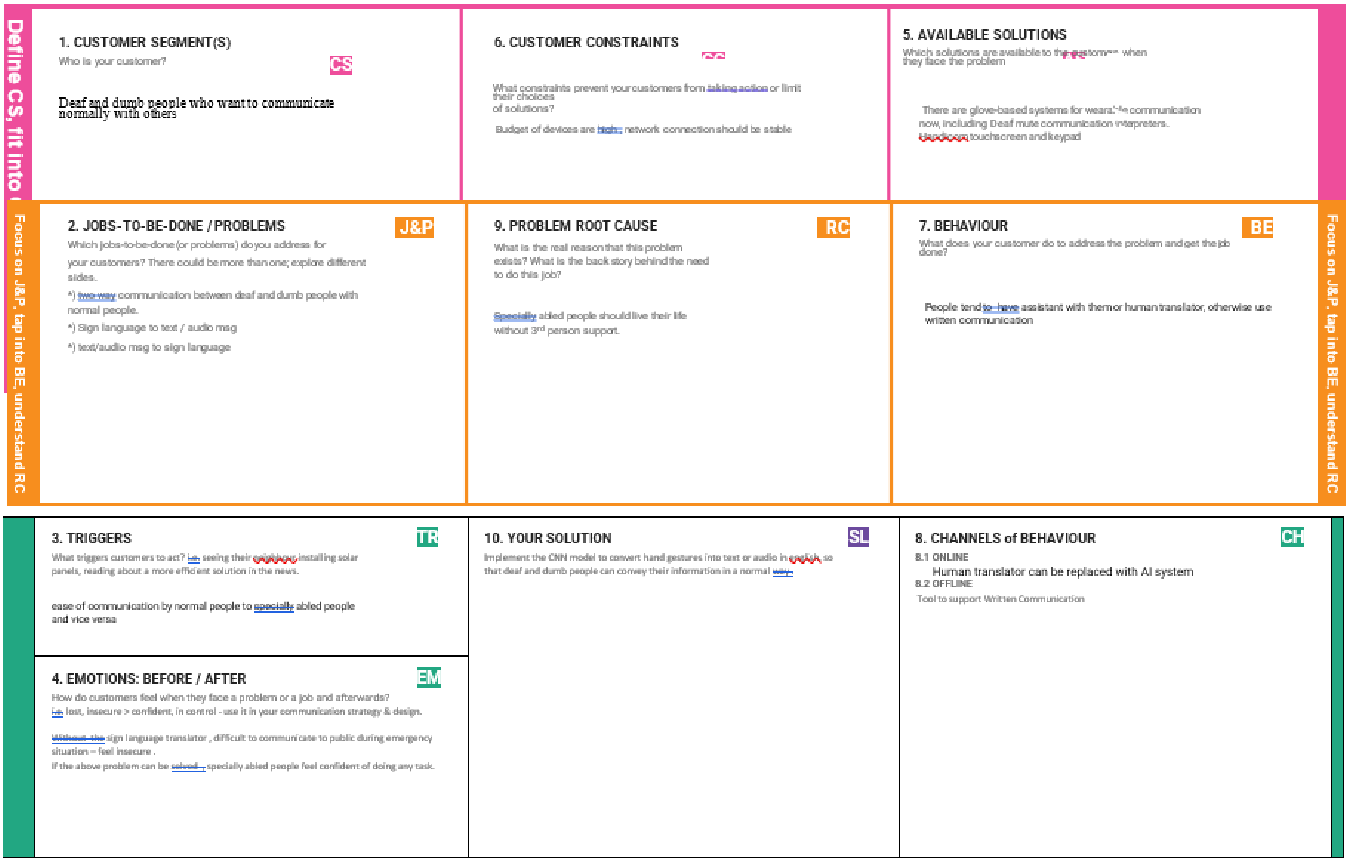
Mobile Robo controlling the communication Process

Integrate with telephony Network

3.3 Proposed Solution:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	People who are deaf or have cognitive disabilities find it difficult to communicate with others. To solve the problems of deaf-dumb people in order to communicate with others and make them feel confident.
2.	Idea / Solution description	Using Machine Learning Algorithm , sign language is converted into voice and text in the desired language with two-way communication.
3.	Novelty / Uniqueness	Enhancing our solution by implementing an alert system in the desired language.
4.	Social Impact / Customer Satisfaction	Improves the communication between deaf & dumb with normal people , so they live independently. Increases the scope of Life skill development
5.	Business Model (Revenue Model)	As many Specially Abled people need of effective interface for their communication , by implementing latest ML algorithm for converting sign to voice & text and voice to sign gives good outcome.
6.	Scalability of the Solution	As Machine Learning algorithm Is used for this implementation, Scalability is easily achieved.

3.4 Problem Solution Fit:



Define CS, fit into C

1. CUSTOMER SEGMENT(S)

CS

Who is your customer?

Deaf and dumb people who want to communicate normally with others

6. CUSTOMER CONSTRAINTS

CC

What constraints prevent your customers from taking action or limit their choices of solutions?

Budget of devices are ~~high ,~~ network connection should be stable

5. AVAILABLE SOLUTIONS

AS

Which solutions are available to the customers when they face the problem

There are glove-based systems for wearable communication now, including Deaf mute communication interpreters. Handicom touchscreen and keypad

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

2. JOBS-TO-BE-DONE / PROBLEMS

J&P

Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.

*) ~~two way~~ communication between deaf and dumb people with normal people.

*) Sign language to text / audio msg

*) text/audio msg to sign language

9. PROBLEM ROOT CAUSE

RC

What is the real reason that this problem exists? What is the back story behind the need to do this job?

~~Specially~~ abled people should live their life without 3rd person support.

7. BEHAVIOUR

BE

What does your customer do to address the problem and get the job done?

People tend ~~to have~~ assistant with them or human translator, otherwise use written communication

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

3. TRIGGERS

TR

What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.

ease of communication by normal people to ~~specially~~ abled people and vice versa

4. EMOTIONS: BEFORE / AFTER

EM

How do customers feel when they face a problem or a job and afterwards? i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

~~Without the~~ sign language translator , difficult to communicate to public during emergency situation – feel insecure .

If the above problem can be ~~solved ,~~ specially abled people feel confident of doing any task.

10. YOUR SOLUTION

SL

Implement the CNN model to convert hand gestures into text or audio in english, so that deaf and dumb people can convey their information in a normal way.

8. CHANNELS of BEHAVIOUR

CH

8.1 ONLINE

Human translator can be replaced with AI system

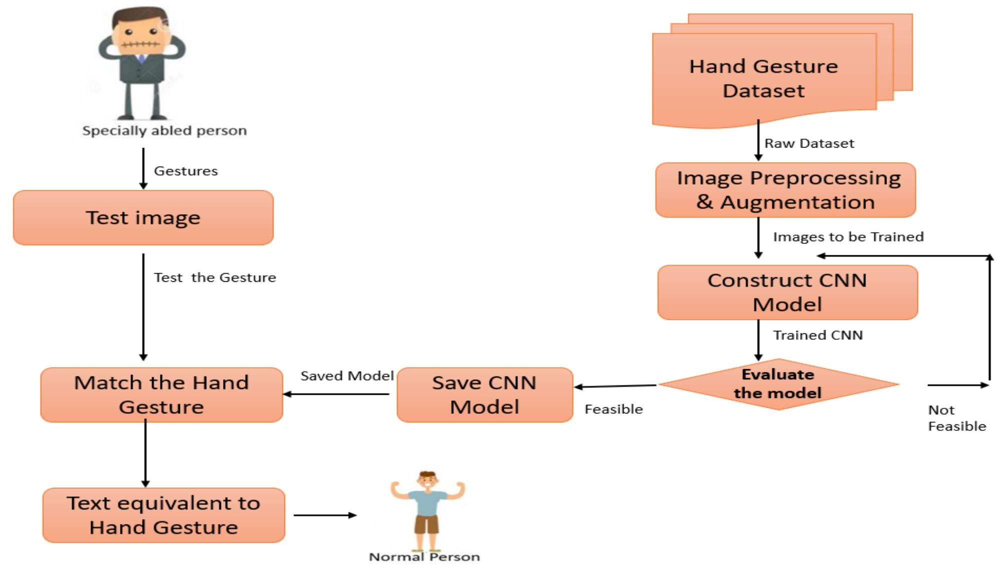
8.2 OFFLINE

Tool to support Written Communication

4. REQUIREMENT ANALYSIS:

4.1 Functional Requirements:

The functional Requirements gives major functionalities of the project work .



FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form
FR-2	User Confirmation	Confirmation via mail
FR-3	User Login	Enter Login id & password to enter into the application
FR-4	Dashboard User Interface	Main Screen to give the image input
FR-5	User Input	Input image representing sign language can be given
FR-6	Display Sign Language Gallery	User can easily select from the gallery for immediate communication
FR-7	Hand Gesture Recognition	Sub Task1: Collect dataset Sub Task2: Train the Model Sub Task3: Fit , Evaluate Model Sub Task4: Test the Model
FR-8	Translation Process	Option available to convert to English Text or Speech.
FR-9	Output Process	Based on the option selected either text output or Audio output is given
FR-10	Admin Console	Admin can maintain the app , login credentials , language for translation, uploading images in gallery

4.2 Non-functional Requirements:

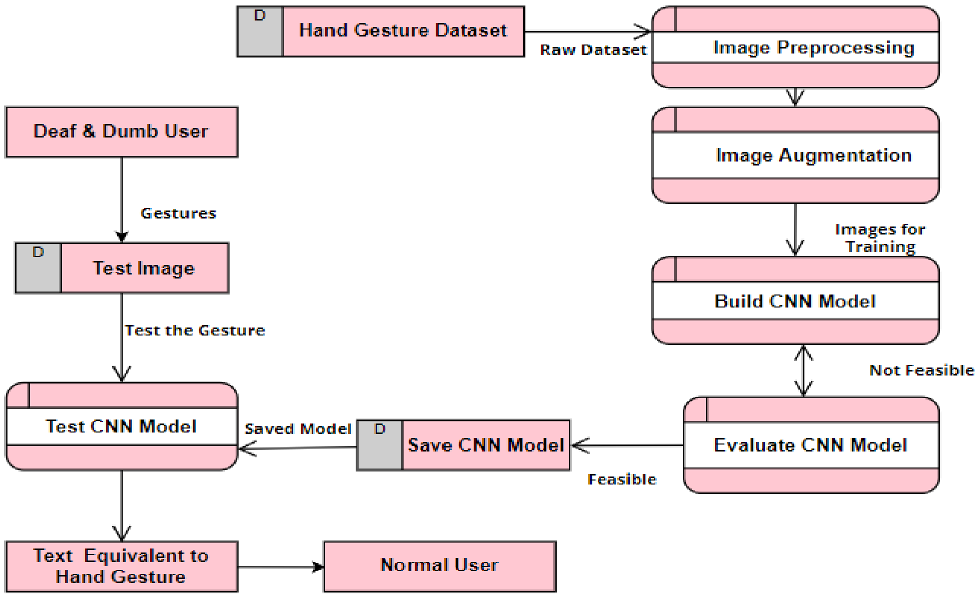
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	User Interface designed to provide easy mechanism to give sign image and get the text or audio output.
NFR-2	**Security**	When this app installed , login details are approved. Thus allowing Only Authorised persons to use this app.
NFR-3	**Reliability**	Sign Language used in this app is universally accepted one , so deaf and dumb people can use this app confidently. Even if a bug does exist, this application has strong fault tolerance and recovers quickly.
NFR-4	**Performance**	This app can process the input image quickly to predict the sign language using ML algorithm which is of high accuracy.
NFR-5	**Availability**	This app can be made available in Google app store to give accessibility to all people who require this
NFR-6	**Scalability**	Since this app is running as standalone application with its own dataset , any number of users can install this app and get the services.

5. PROJECT DESIGN:

5.1 Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



5.2 Solution & Technical Architecture:

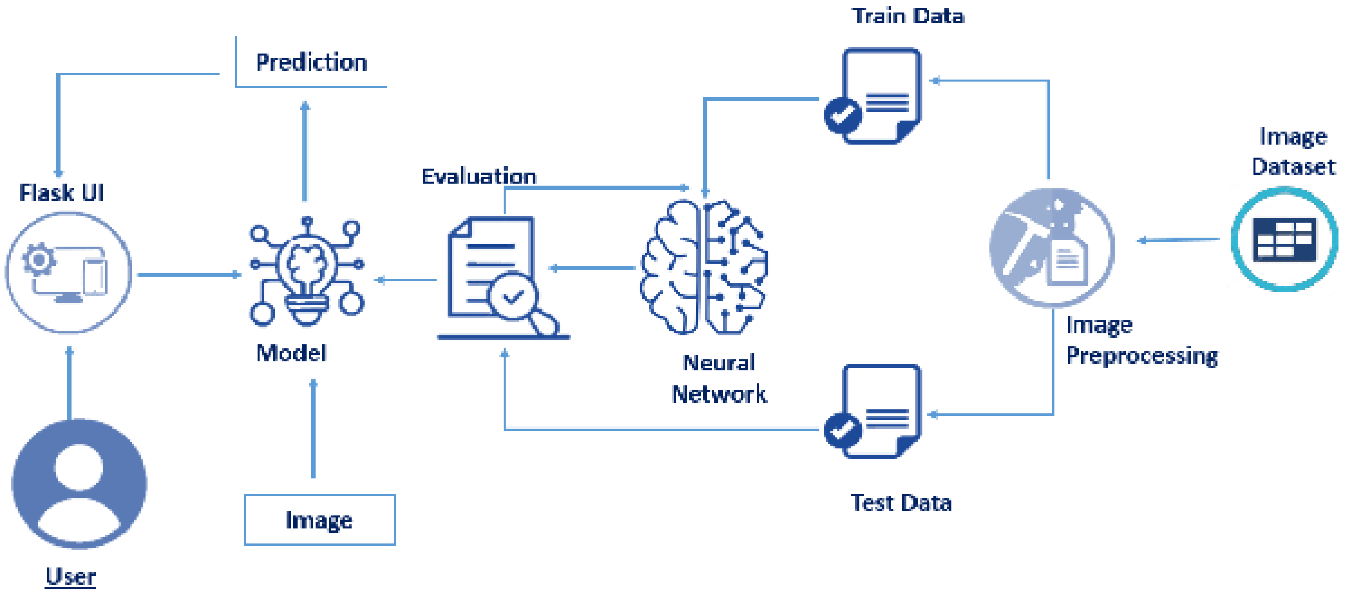


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	user interacts with Mobile Application	HTML, Flask UI
2.	Application Logic-1 Hand Gesture Image Dataset	Collecting the dataset for Hand Gesture	Kaggle & IBM Login
		Preparing Dashboard with Predefined hand gestures	
3.	Application Logic-2 Image Pre-processing	Image Pre-processing steps to create train and test data set	IBM Watson , Python, OpenCV
4.	Application Logic-3	CNN Model Building	IBM Watson Python
5.	Database	Data Type – Images of Hand Gestures	MYSQL
6.	Cloud Database	Database Service on Cloud	--
7.	File Storage	File storage requirements for hand gesture images and also for audio	IBM Block Storage , Local Filesystem
8.	External API-1	focus on enabling fast experimentation	Keras API
9.	External API-2	an approachable, highly-productive interface for solving machine learning problems	Tensor flow 2
10.	External API-3	Text to voice converter	IBM Watson TTS /Text to Speech /Rev.ai
11.	Machine Learning Model	Including required layers in CNN model	Tensor Flow 2, IBM watson
12.	Infrastructure (Server / Cloud)	Application Deployment on Local System & Cloud	Local & IBM Cloud & Watsan Studio Service

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	Keras, Tensor Flow
2.	Security Implementations	Only registered users are allowed to access the application.	--
3.	Scalable Architecture	Allows admin to add more templates for sign language	MYSQL
4.	Availability	Always this application is available to user as it is using open source framework	Keras, Tensor Flow
5.	Performance	CNN model get the image & identifies the sign language and gives the result with high accuracy	Deep learning

5.3 User Stories:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	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
	User Confirmation	USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	Low	Sprint-1
	User Login	USN-3	As a user, I can log into the application by entering email & password	Application validates Login Id , Password and display the dashboard on entering valid details otherwise error msg displayed to user	High	Sprint-1
	Dashboard User Interface	USN-4	As a user, I can select buttons available in the dashboard to give Image input	I can access the relevant screen to give input	High	Sprint-2
	User Input	USN-6	As a user, I can give image input	I can click Image submit button for further processing	High	Sprint-2
	Display Sign Language Gallery	USN-5	As a user, I can select the sign from the gallery	I can access the sign from gallery	Low	Sprint 4
	Hand Gesture Recognition	USN-7	As a user, I can select hand gesture recognition to get the equivalent text or audio	I can click the Hand Gesture Recognition button	High	Sprint-3
	Translation Process	USN-8	As a user, I can select translate option to get text or audio	I can click the translate option button	High	Sprint-3
	Output Process	USN-9	As a user, I can select the output button to get the response	I can get the translated content	High	Sprint-4

6. PROJECT PLANNING & SCHEDULING:

6.1 Sprint Planning & Estimation:

Product Backlog, Sprint Schedule, and Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Data Collection	USN-1	Collect the dataset required for alphabets and numbers	4	High	PRIYAADARSHAN B YUVARAJ S SURYAPRAKASH K S LOKESHWARAN V THARUNKUMAR R
Sprint-1	Pre-processing	USN-2	Do Image preprocessing	2	Medium	PRIYAADARSHAN B YUVARAJ S SURYAPRAKASH K S LOKESHWARAN V THARUNKUMAR R
Sprint-2	Model Building	USN-3	Import libraries ,Initialize the model Add the convolution layer ,Add the pooling layer ,Add the flatten layer Adding the dense layers ,Compile the model ,Fit and save the model.	5	High	PRIYAADARSHAN B YUVARAJ S SURYAPRAKASH K S LOKESHWARAN V THARUNKUMAR R
Sprint-3	Test the Model	USN-4	Pass an image to get predictions, Import the packages and load the saved model , Load the test image, pre-process it and predict.	5	High	PRIYAADARSHAN B YUVARAJ S SURYAPRAKASH K S LOKESHWARAN V THARUNKUMAR R
Sprint-4	Train CNN Model on IBM & Application Development	USN-5	Register for IBM Cloud. Train your model on IBM Cloud.	6	High	PRIYAADARSHAN B YUVARAJ S

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
			Store your Model on IBM Cloud.			SURYAPRAKASH K S LOKESHWARAN V THARUNKUMAR R

Project Tracker, Velocity & Burndown Chart:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	6	6 Days	24 Oct 2022	29 Oct 2022	6	29 Oct 2022
Sprint-2	5	6 Days	31 Oct 2022	05 Nov 2022	5	05 Nov 2022
Sprint-3	5	6 Days	07 Nov 2022	12 Nov 2022	5	12 Nov 2022
Sprint-4	6	6 Days	14 Nov 2022	19 Nov 2022	6	19 Nov 2022

Velocity :

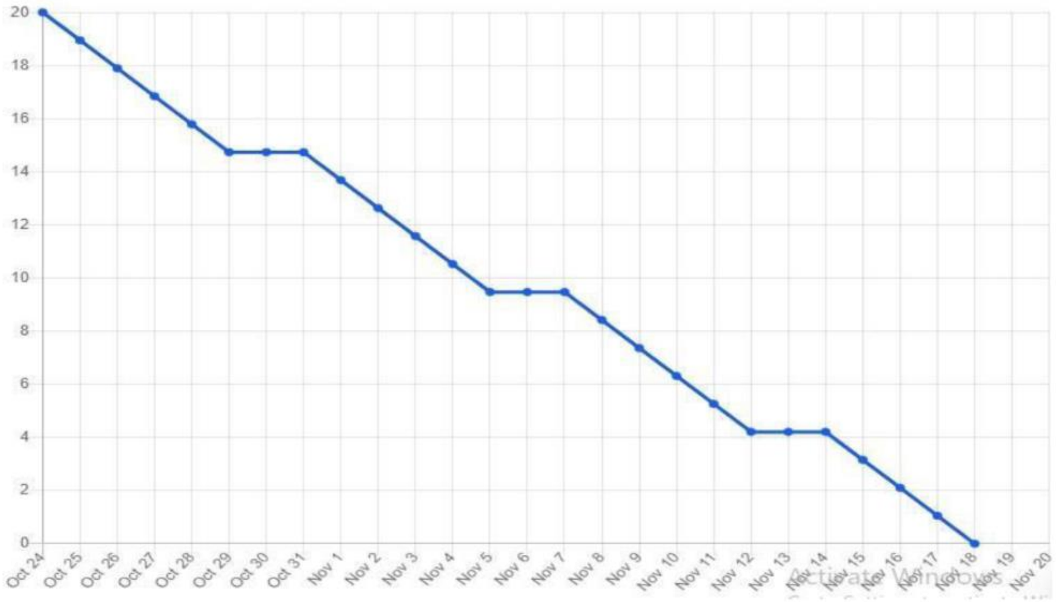
Average Velocity = Velocity / Sprint Duration

- Average Velocity → AV
- Velocity → Points per sprint
- Sprint Duration → Number of days per sprint

1. Sprint – 1: AV = 6/6 = 1
2. Sprint – 2: AV = 5/6 = 0.834
3. Sprint – 3: AV = 5/6 = 0.834
4. Sprint – 4: AV = 6/6 = 1

Burndown Chart:

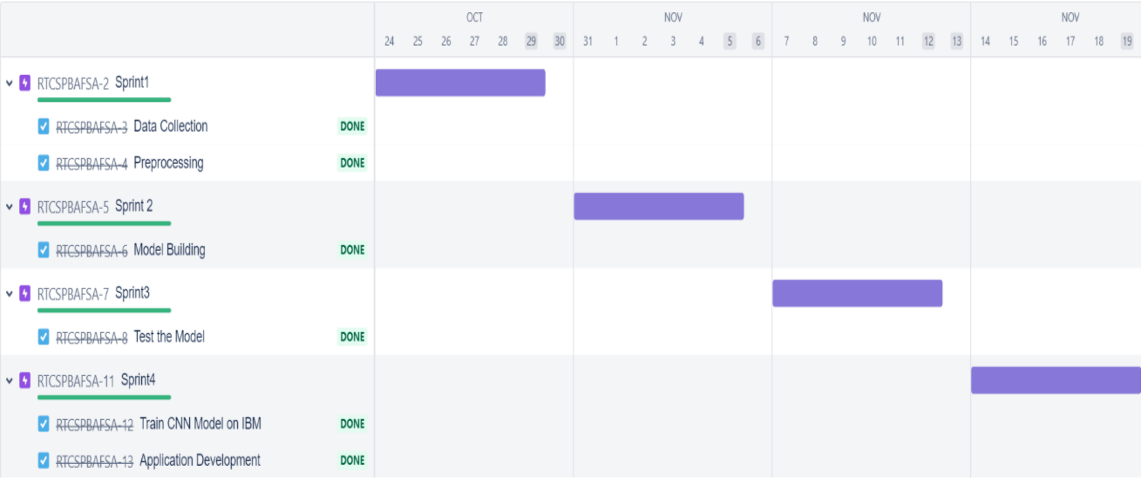
A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurableprogress over time.



6.2 Sprint Delivery Schedule:

Sprint	**Total Story Points**	**Duration**	**Sprint Start Date**	**Sprint End Date (Planned)**
Sprint-1	6	6 Days	24 Oct 2022	29 Oct 2022
Sprint-2	5	6 Days	31 Oct 2022	05 Nov 2022
Sprint-3	5	6 Days	07 Nov 2022	12 Nov 2022
Sprint-4	6	6 Days	14 Nov 2022	19 Nov 2022

6.3 Reports From JIRA:



RTCSPBAFSA-2 Sprint1

~~RTCSPBAFSA-3~~ Data Collection DONE

~~RTCSPBAFSA-4~~ Preprocessing DONE

RTCSPBAFSA-5 Sprint 2

~~RTCSPBAFSA-6~~ Model Building DONE

RTCSPBAFSA-7 Sprint3

~~RTCSPBAFSA-8~~ Test the Model DONE

RTCSPBAFSA-11 Sprint4

~~RTCSPBAFSA-12~~ Train CNN Model on IBM DONE

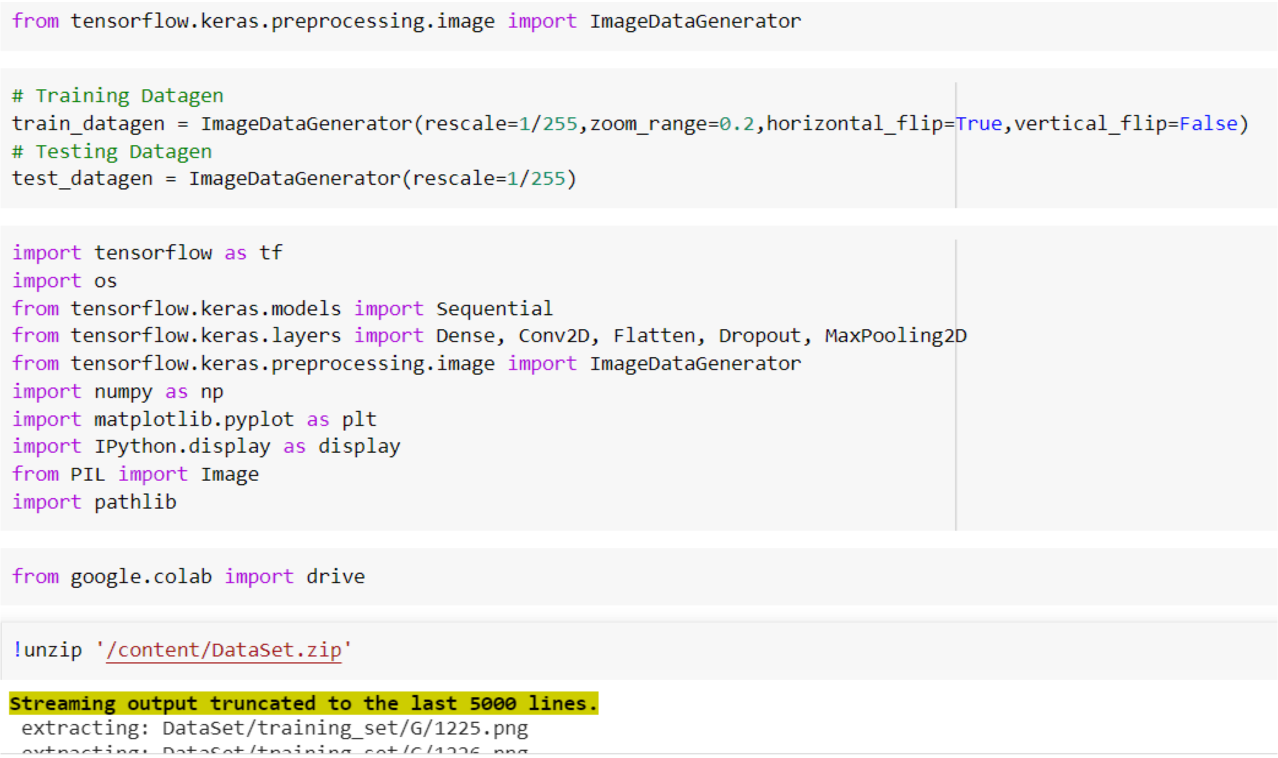
~~RTCSPBAFSA-13~~ Application Development DONE

7. CODING & SOLUTIONING:

7.1 Dataset Collection & Preprocessing

The first stage of any machine learning algorithm starts with dataset collection. And preprocessing so as to develop CNN model and the save it . We have used a dataset from IBM – guided project section to classify the ISL (Indian Sign Language) alphabets (A-I).

Step1 : Import Image Generator Library and configure it



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

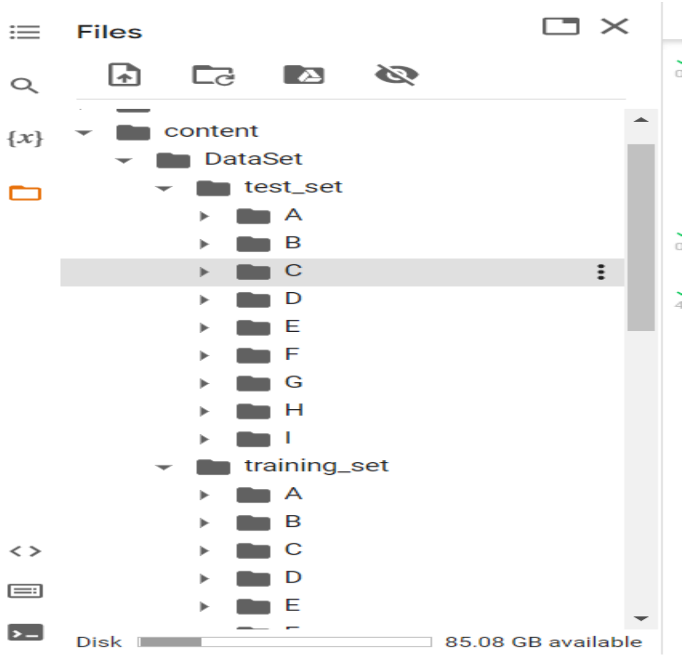
```
# Training Datagen
train_datagen = ImageDataGenerator(rescale=1/255,zoom_range=0.2,horizontal_flip=True,vertical_flip=False)
# Testing Datagen
test_datagen = ImageDataGenerator(rescale=1/255)
```

```
import tensorflow as tf
import os
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Conv2D, Flatten, Dropout, MaxPooling2D
from tensorflow.keras.preprocessing.image import ImageDataGenerator
import numpy as np
import matplotlib.pyplot as plt
import IPython.display as display
from PIL import Image
import pathlib
```

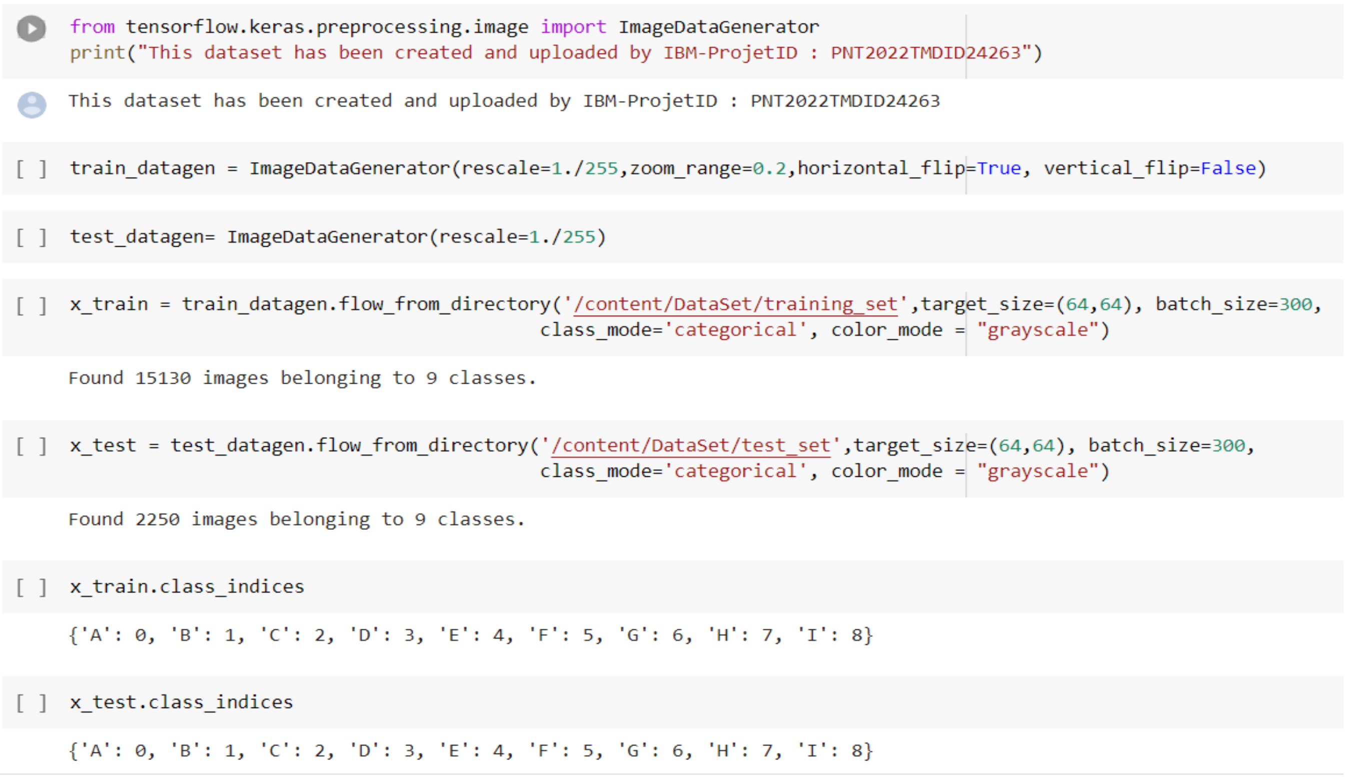
```
from google.colab import drive
```

```
!unzip '/content/DataSet.zip'
```

```
Streaming output truncated to the last 5000 lines.
 extracting: DataSet/training_set/G/1225.png
 extracting: DataSet/training_set/G/1226.png
```



Step2 : Apply Image Data generator Functionality to Train and Test set



```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
print("This dataset has been created and uploaded by IBM-ProjetID : PNT2022TMDID24263")
```

```
This dataset has been created and uploaded by IBM-ProjetID : PNT2022TMDID24263
```

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

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

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

```
Found 15130 images belonging to 9 classes.
```

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

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

```
x_train.class_indices
```

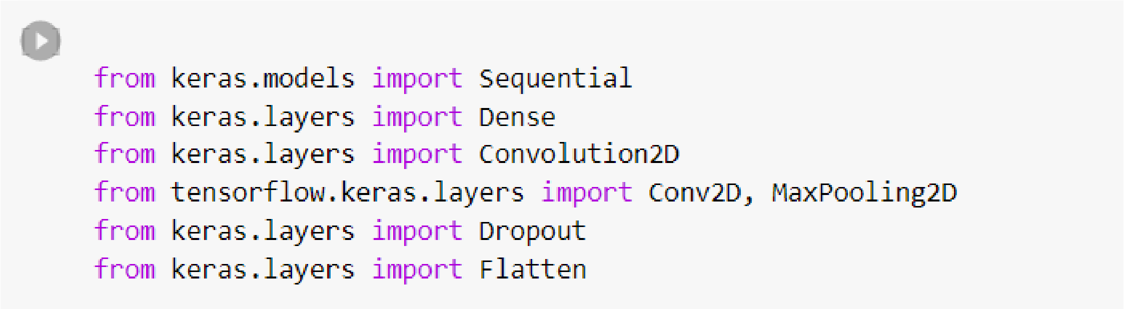
```
{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}
```

```
x_test.class_indices
```

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

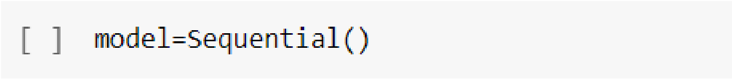
7.2 MODEL BUILDING

1.Import libraries



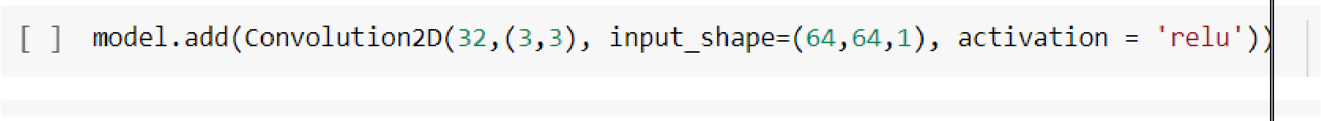
```python
from keras.models import Sequential
from keras.layers import Dense
from keras.layers import Convolution2D
from tensorflow.keras.layers import Conv2D, MaxPooling2D
from keras.layers import Dropout
from keras.layers import Flatten
```

2.Initialize the model



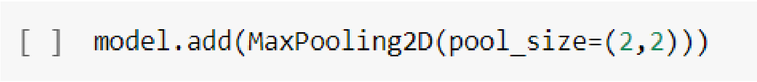
```python
model=Sequential()
```

3.Add the convolution layer



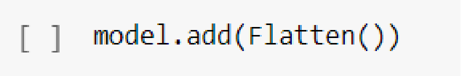
```python
model.add(Convolution2D(32,(3,3), input_shape=(64,64,1), activation = 'relu'))
```

4.Add the pooling layer



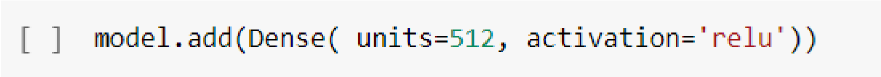
```python
model.add(MaxPooling2D(pool_size=(2,2)))
```

5.Add the flatten layer

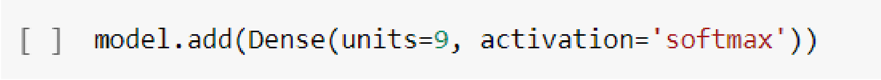


```python
model.add(Flatten())
```

6.Adding the dense layers

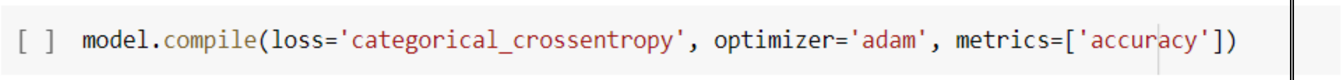


```python
model.add(Dense( units=512, activation='relu'))
```



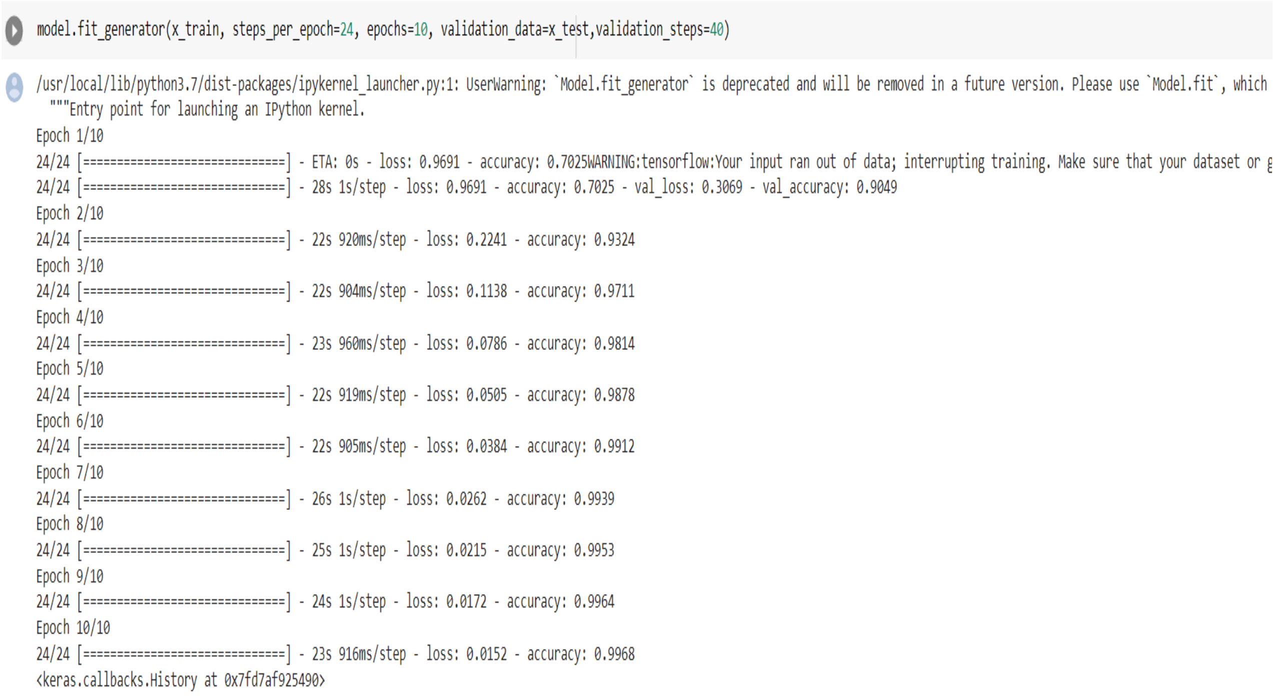
```python
model.add(Dense(units=9, activation='softmax'))
```

7.Compile the model

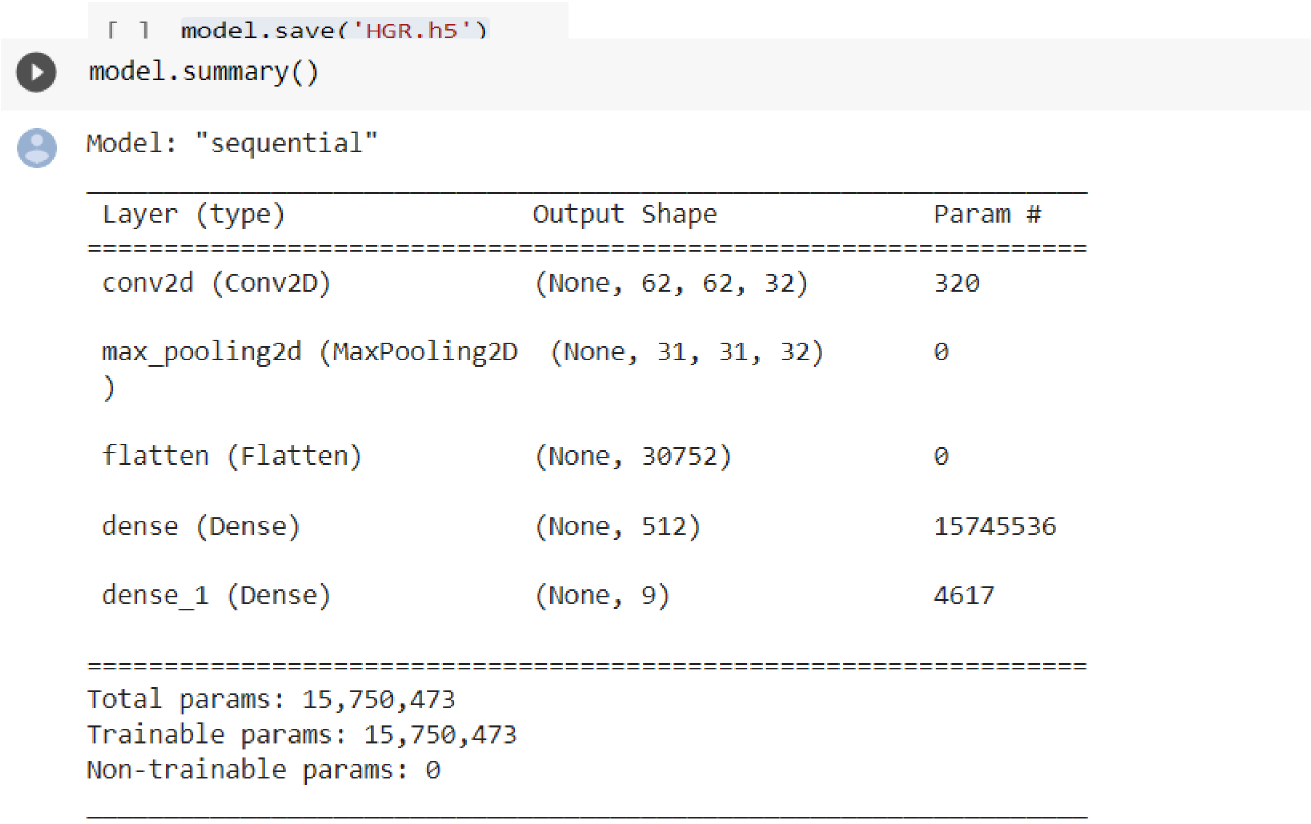


```python
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
```

8.Fit and save the model.



```
model.fit_generator(x_train, steps_per_epoch=24, epochs=10, validation_data=x_test,validation_steps=40)

/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
  """Entry point for launching an IPython kernel.
Epoch 1/10
24/24 [==============================] - ETA: 0s - loss: 0.9691 - accuracy: 0.7025WARNING:tensorflow:Your input ran out of data; interrupting training. Make sure that your dataset or g
24/24 [==============================] - 28s 1s/step - loss: 0.9691 - accuracy: 0.7025 - val_loss: 0.3069 - val_accuracy: 0.9049
Epoch 2/10
24/24 [==============================] - 22s 920ms/step - loss: 0.2241 - accuracy: 0.9324
Epoch 3/10
24/24 [==============================] - 22s 904ms/step - loss: 0.1138 - accuracy: 0.9711
Epoch 4/10
24/24 [==============================] - 23s 960ms/step - loss: 0.0786 - accuracy: 0.9814
Epoch 5/10
24/24 [==============================] - 22s 919ms/step - loss: 0.0505 - accuracy: 0.9878
Epoch 6/10
24/24 [==============================] - 22s 905ms/step - loss: 0.0384 - accuracy: 0.9912
Epoch 7/10
24/24 [==============================] - 26s 1s/step - loss: 0.0262 - accuracy: 0.9939
Epoch 8/10
24/24 [==============================] - 25s 1s/step - loss: 0.0215 - accuracy: 0.9953
Epoch 9/10
24/24 [==============================] - 24s 1s/step - loss: 0.0172 - accuracy: 0.9964
Epoch 10/10
24/24 [==============================] - 23s 916ms/step - loss: 0.0152 - accuracy: 0.9968
<keras.callbacks.History at 0x7fd7af925490>
```



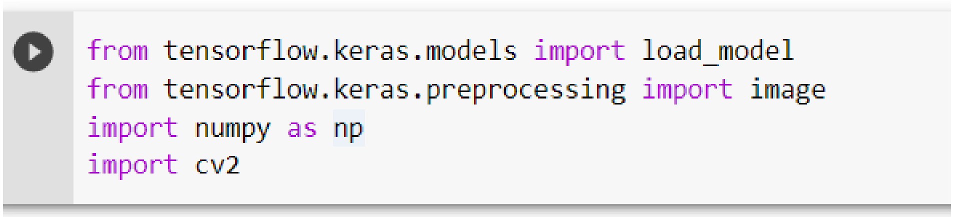
```
[ ]  model.save('HGR.h5')

model.summary()

Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 62, 62, 32)        320

 max_pooling2d (MaxPooling2D  (None, 31, 31, 32)       0
 )

 flatten (Flatten)           (None, 30752)             0

 dense (Dense)               (None, 512)               15745536

 dense_1 (Dense)             (None, 9)                 4617

=================================================================
Total params: 15,750,473
Trainable params: 15,750,473
Non-trainable params: 0
_________________________________________________________________
```

Model summary generates the complete details of model saying convolution layer , max pool layer , flatten , dense and parameters trainable and non trainable

7.3 TESTING THE MODEL

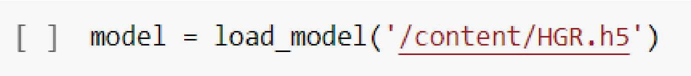
Testing of a built model as important as building the application itself. Here, we test the built model with our testing dataset images. Steps are given below

1. Import the packages



```
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
import numpy as np
import cv2
```

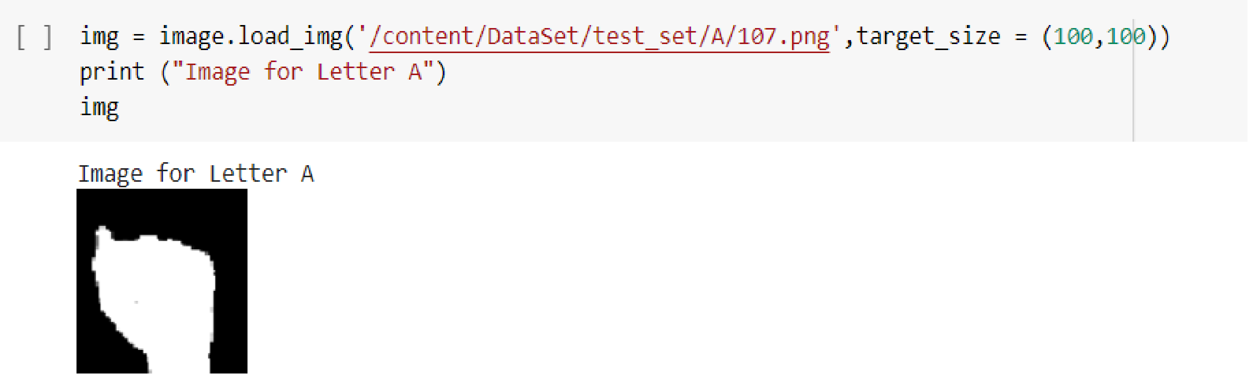
2 . load the saved model



```
[ ]  model = load_model('/content/HGR.h5')
```

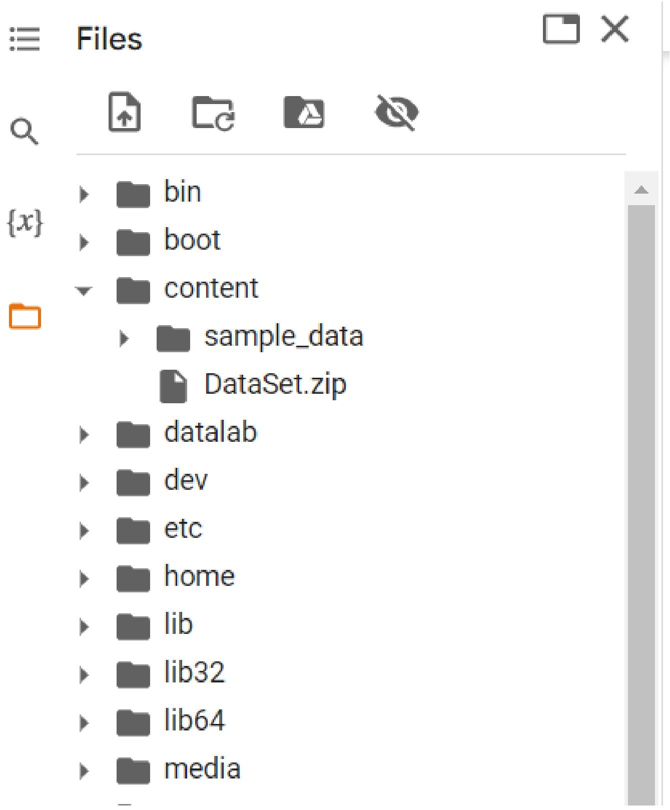
3. .Load the test image & predict the image

Test Case – 1: ISL Alphabet ‘A’:



```
[ ]  img = image.load_img('/content/DataSet/test_set/A/107.png',target_size = (100,100))
     print ("Image for Letter A")
     img

     Image for Letter A
```

7.4 DATA SET Collection

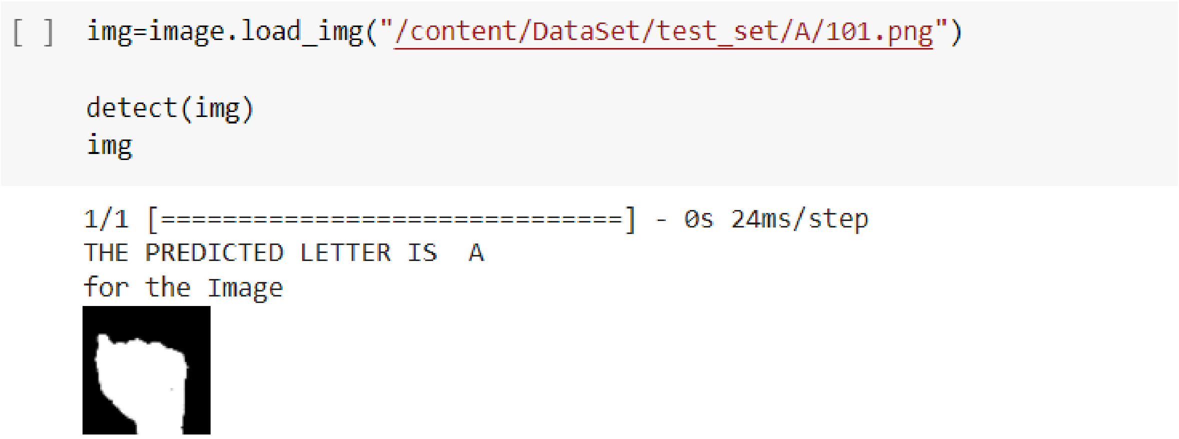


8. TESTING:

8.1 Test Cases:

Testing of a built model as important as building the application itself.Here, we test the built model with our testing dataset images.

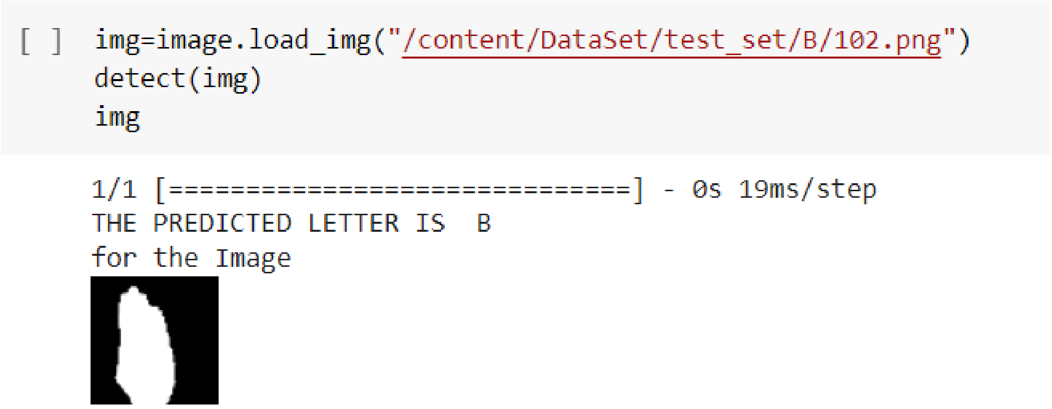
Test Case – 1: ISL Alphabet ‘A’:



```
[ ]  img=image.load_img("/content/DataSet/test_set/A/101.png")

     detect(img)
     img
```

```
1/1 [==============================] - 0s 24ms/step
THE PREDICTED LETTER IS  A
for the Image
```

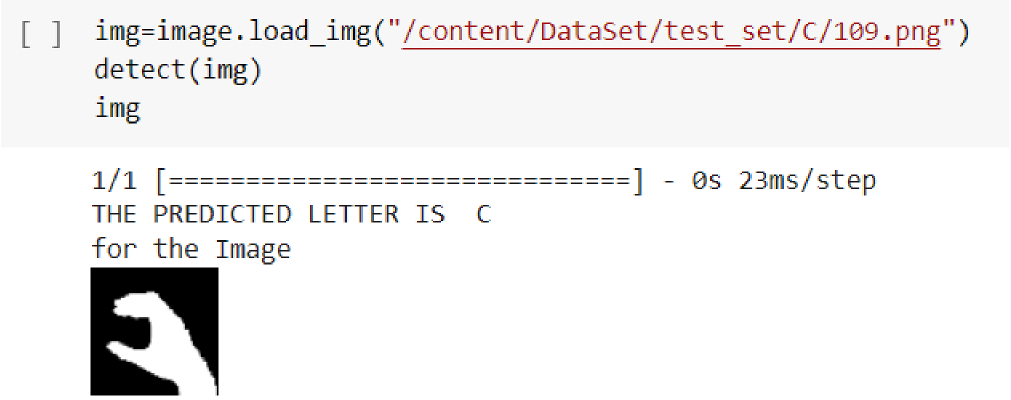
Test Case – 2: ISL Alphabet ‘B’:



```
[ ]  img=image.load_img("/content/DataSet/test_set/B/102.png")
     detect(img)
     img
```

```
1/1 [==============================] - 0s 19ms/step
THE PREDICTED LETTER IS  B
for the Image
```

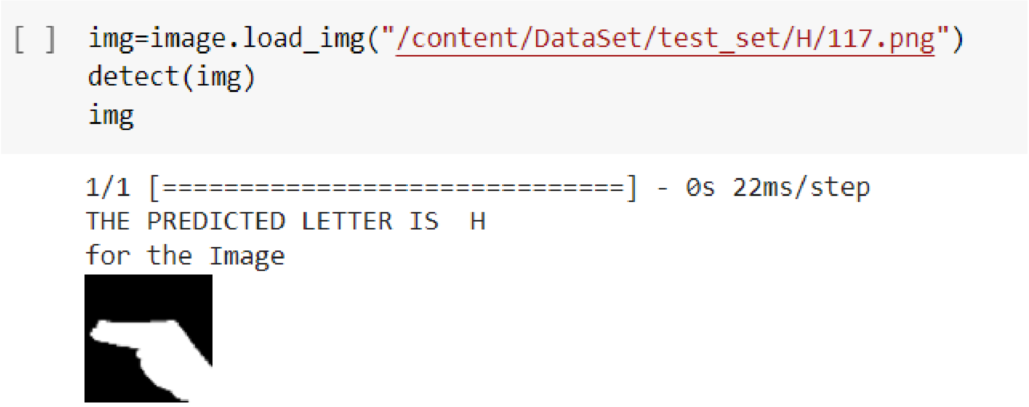
Test Case – 3: ISL Alphabet ‘C’:



```
[ ]  img=image.load_img("/content/DataSet/test_set/C/109.png")
     detect(img)
     img
```

```
1/1 [==============================] - 0s 23ms/step
THE PREDICTED LETTER IS  C
for the Image
```

Test Case – 4: ISL Alphabet ‘H’:

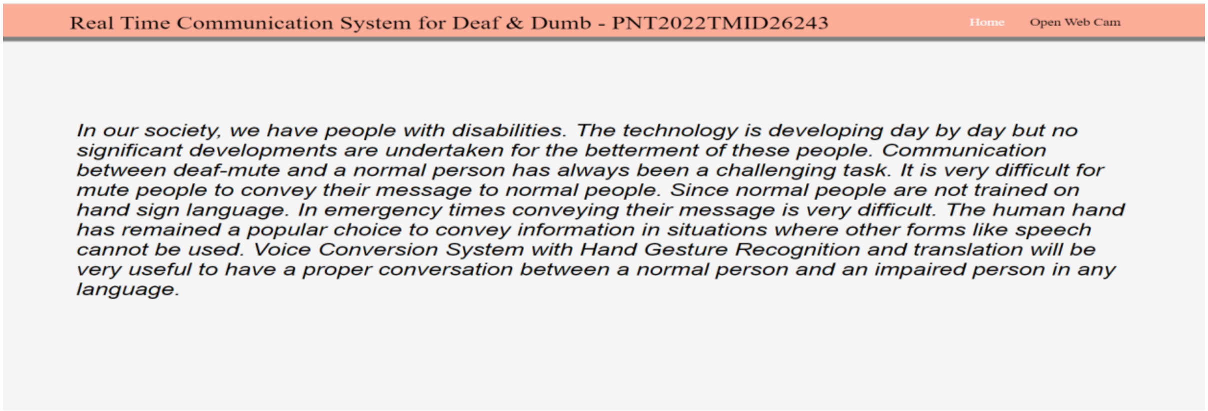


```
[ ]  img=image.load_img("/content/DataSet/test_set/H/117.png")
     detect(img)
     img
```

```
1/1 [==============================] - 0s 22ms/step
THE PREDICTED LETTER IS  H
for the Image
```

8.2 User Acceptance Testing

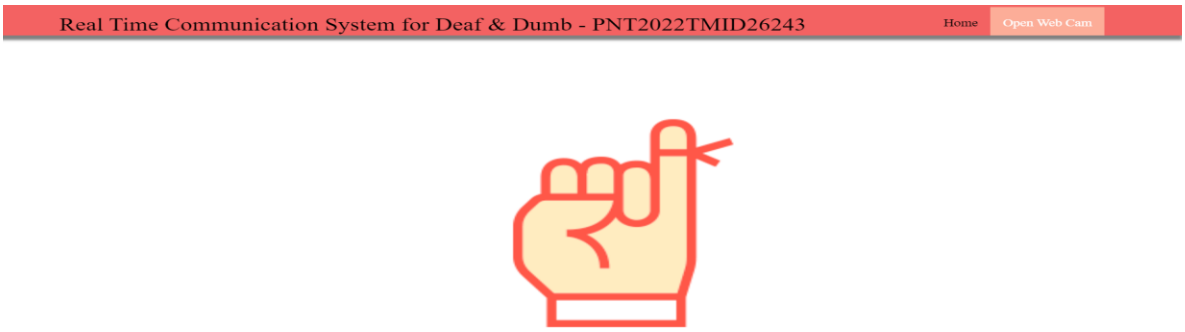
Home Page:



Real Time Communication System for Deaf & Dumb - PNT2022TMID26243

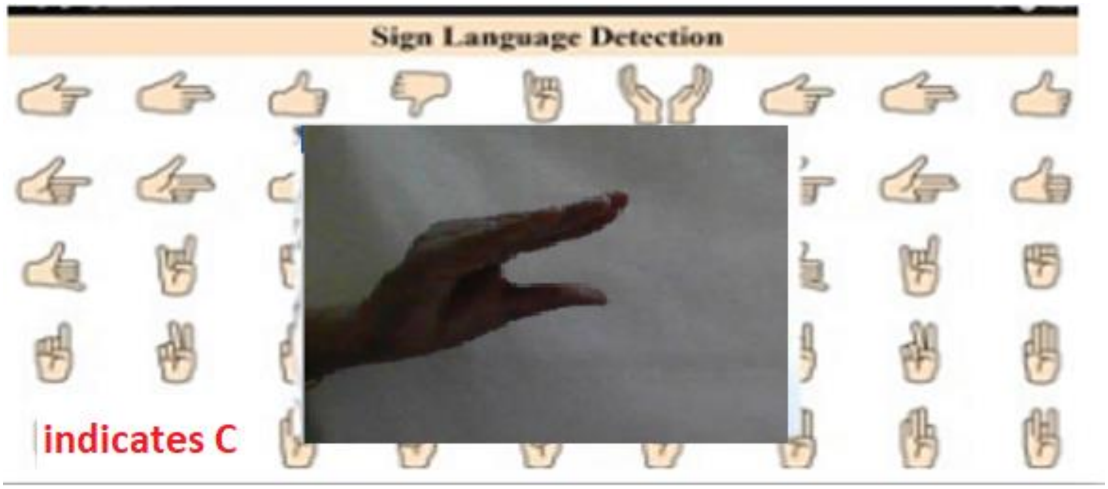
In our society, we have people with disabilities. The technology is developing day by day but no significant developments are undertaken for the betterment of these people. Communication between deaf-mute and a normal person has always been a challenging task. It is very difficult for mute people to 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. 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.

Open WebCam Page:



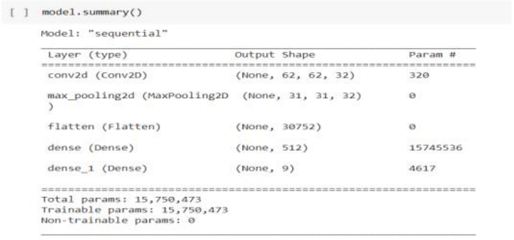
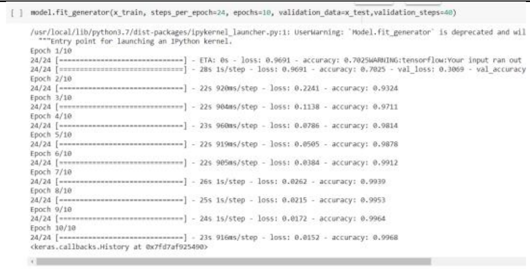
Real Time Communication System for Deaf & Dumb - PNT2022TMID26243

Conversion Process :



9. RESULTS :

9.1 Performance Metrics

S.No.	Parameter	Values	Screenshot
1.	Model Summary	**Model : Sequential** **Convolution2D: (None , 62,62, 32)** **Maxpooling 2D: (None , 31,31, 32)** **Trainable Parameter : 15750473** **Non Trainable Parameter: 0**	
2.	Accuracy	**Training Accuracy - 0.9968** **Loss : 0.0152**	

```
[ ]  model.summary()

     Model: "sequential"
     _________________________________________________________________
      Layer (type)                Output Shape              Param #
     =================================================================
      conv2d (Conv2D)             (None, 62, 62, 32)        320

      max_pooling2d (MaxPooling2D  (None, 31, 31, 32)       0
      )

      flatten (Flatten)           (None, 30752)             0

      dense (Dense)               (None, 512)               15745536

      dense_1 (Dense)             (None, 9)                 4617

     =================================================================
     Total params: 15,750,473
     Trainable params: 15,750,473
     Non-trainable params: 0
     _________________________________________________________________
```

```
[ ]  model.fit_generator(x_train, steps_per_epoch=24, epochs=10, validation_data=x_test,validation_steps=40)

     /usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: UserWarning: `Model.fit_generator` is deprecated and wil
       """Entry point for launching an IPython kernel.
     Epoch 1/10
     24/24 [==============================] - ETA: 0s - loss: 0.9691 - accuracy: 0.7025WARNING:tensorflow:Your input ran out
     24/24 [==============================] - 28s 1s/step - loss: 0.9691 - accuracy: 0.7025 - val_loss: 0.3069 - val_accuracy
     Epoch 2/10
     24/24 [==============================] - 22s 920ms/step - loss: 0.2241 - accuracy: 0.9324
     Epoch 3/10
     24/24 [==============================] - 22s 904ms/step - loss: 0.1138 - accuracy: 0.9711
     Epoch 4/10
     24/24 [==============================] - 23s 960ms/step - loss: 0.0786 - accuracy: 0.9814
     Epoch 5/10
     24/24 [==============================] - 22s 919ms/step - loss: 0.0505 - accuracy: 0.9878
     Epoch 6/10
     24/24 [==============================] - 22s 905ms/step - loss: 0.0384 - accuracy: 0.9912
     Epoch 7/10
     24/24 [==============================] - 26s 1s/step - loss: 0.0262 - accuracy: 0.9939
     Epoch 8/10
     24/24 [==============================] - 25s 1s/step - loss: 0.0215 - accuracy: 0.9953
     Epoch 9/10
     24/24 [==============================] - 24s 1s/step - loss: 0.0172 - accuracy: 0.9964
     Epoch 10/10
     24/24 [==============================] - 23s 916ms/step - loss: 0.0152 - accuracy: 0.9968
     <keras.callbacks.History at 0x7fd7af925490>
```

10, ADVANTAGES & DISADVANTAGES:

Advantages:

- It enables people with disabilities to interact with the general public using their native sign language.
- When utilized appropriately, the software can eliminate barriers between people with special needs and the general population.
- The application can be scaled up to recognize words, numbers, and other objects.

Disadvantages:

- To use the programme, people with special needs must receive training.
- Because the trained model's accuracy is not 100 percent, there may occasionally be instances where the model generates incorrect results.
- For the model to accurately categorize, the input image must be of high quality.

11. CONCLUSION:

- The project can be enhanced with additional functions in the future and can close the communication gap between deaf-mute persons and hearing people.
- The project also has commercial potential that might be used.

12. FUTURE SCOPE:

Future development on this topic has a huge amount of potential. This programme provides components that are incorporated into a Flask web application, so components can be modified or replaced in the future to meet changing requirements.Many new functionalities can be added like:

- Making the application predict words by combing multiple signs.
- Using NLP (Natural Language Processing) & ML (Machine Learning) tocombine predicted words to create a meaningful sentence.
- Add many User Experience (UX) enhancing changes.
- Making the web application’s UI (User Interface) look more appealing anduser-friendly.

13. APPENDIX:

Source Code: App.py:

```
# import the necessary packages
from flask import Flask,render_template,request
# Flask-It is our framework which we are going to use to run/serve our
application.
#request-for accessing file which was uploaded by the user on our
application.
import cv2 # opencv library
from tensorflow.keras.models import load_model#to load our trained model
import numpy as np
from gtts import gTTS #to convert text to speech
from skimage.transform import resize
import os
from keras.preprocessing import image
from playsound import playsound
'''
def playaudio(text):
    speech=gTTS(text)
    print(type(speech))
    speech.save("output1.mp3")
    playsound("output1.mp3")
    return
'''
app = Flask(__name__,template_folder="templates") # initializing a flask app
# Loading the model
model=load_model('HGR.h5')
print("Loaded model from disk")
vals = ['A', 'B','C','D','E','F','G','H','I']

#app=Flask(__name__,template_folder="templates")
@app.route('/', methods=['GET'])
def index():
    return render_template('home.html')
@app.route('/home', methods=['GET'])
def home():
    return render_template('home.html')
@app.route('/upload', methods=['GET', 'POST'])
```

```
def predict():

        # Get a reference to webcam #0 (the default one)

        print("[INFO] starting video stream...")
        vs = cv2.VideoCapture(0)
        #writer = None
        (W, H) = (None, None)

# loop over frames from the video file stream
        while True:
            # read the next frame from the file
            (grabbed, frame) = vs.read()

            # if the frame was not grabbed, then we have reached the end
            # of the stream
            if not grabbed:
                break

            # if the frame dimensions are empty, grab them
            if W is None or H is None:
                (H, W) = frame.shape[:2]

            # clone the output frame, then convert it from BGR to RGB
            # ordering and resize the frame to a fixed 64x64
            output = frame.copy()
            #print("apple")
            img = resize(frame,(64,64,1))
            img = np.expand_dims(img,axis=0)
            if(np.max(img)>1):
                img = img/255.0
            result = np.argmax(model.predict(img), axis=-1)
            index=['A', 'B','C','D','E','F','G','H','I']
            result=str(index[result[0]])
            #print(result)
            #result=result.tolist()

            cv2.putText(output, "It indicates: {}".format(result), (10, 120),
cv2.FONT_HERSHEY_PLAIN,
                        2, (0,255,255), 1)
            #converts text to speech and plays the audio
            speech = gTTS(text = result, lang = 'en', slow = False)
            #speech=gTTS(text)
            print(type(speech))
            speech.save("text.mp3")
            os.system("start text.mp3")

            cv2.imshow("Output", output)
            key = cv2.waitKey(1) & 0xFF

              # if the `q` key was pressed, break from the loop
            if key == ord("q"):
              break

          # release the file pointers
          print("[INFO] cleaning up...")
          vs.release()
          cv2.destroyAllWindows()
          return render_template("upload.html")

   if __name__ == '__main__':
      app.run(host='0.0.0.0', port=8000, debug=False)
```

Camera.py

```
 import cv2
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image

class Video(object):
    def __init__(self):
        self.video = cv2.VideoCapture(0)
        self.roi_start = (50, 150)
        self.roi_end = (250, 350)
        self.model = load_model('HGR.h5') # Execute Local Trained Model
        #self.model = load_model('IBM_Communication_Model.h5') # Execute
IBM Trained Model
        self.index=['A','B','C','D','E','F','G','H','I']
        self.y = None
    def __del__(self):
        k = cv2.waitKey(1)

        self.video.release()
    def get_frame(self):
        ret,frame = self.video.read()
        frame = cv2.resize(frame, (640, 480))
        copy = frame.copy()
        copy = copy[150:150+200,50:50+200]
        # Prediction Start
        cv2.imwrite('image.jpg',copy)
        copy_img = image.load_img('image.jpg', target_size=(64,64))
        x = image.img_to_array(copy_img)
        x = np.expand_dims(x, axis=0)
        pred = np.argmax(self.model.predict(x), axis=1)
        self.y = pred[0]
        cv2.putText(frame,'The Predicted Alphabet is:
'+str(self.index[self.y]),(100,50),cv2.FONT_HERSHEY_SIMPLEX,1,(0,0,0),3)
        ret,jpg = cv2.imencode('.jpg', frame)
        return jpg.tobytes()
```

main.py

```
import cv2

video = cv2.VideoCapture(0)

while True:
    ret, frame = video.read()
    cv2.imshow("Frame", frame)
    k = cv2.waitKey(1)
    if k == ord('q'):
        break

video.release()
cv2.destroyAllWindows()
```

HTML:

Home.html

```
<html>
<script>

</script>

<style>
.header { position: relative;
     top:0;
     margin:0px;
     z-index: 1;
     left: 0px;
     right: 0px;
     position: fixed;
     background-color: #FCAD98 ;
     color: white;
     box-shadow: 0px 8px 2px grey;
     overflow: hidden;
     padding-left:20px;
     font-family: 'Josefin Sans';
     font-size: 2vw;
     width: 100%;
     height:8%;
     text-align: center;
   }
   .topnav {
  overflow: hidden;
  background-color: #FCAD98;
}

.topnav-right a {
  float: left;
  color: black;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
  font-size: 18px;
}

.topnav-right a:hover {
  background-color: #FCAD98;
  color: black;
}

.topnav-right a.active {
  background-color: #FCAD98;
  color: white;
}

.topnav-right {
  float: right;
  padding-right:100px;
}

body {
background-image: -webkit-linear-gradient(90deg, skyblue 0%, steelblue
100%);
  background-image: url("");
    background-size: cover;
  background-attachment: fixed;
  background-size: 100% 100%;
  background-color: ;
  background-repeat: no-repeat;
  background-size:cover;
  background-position: 0px 0px;
  }
  .button {
  background-color: #091425;
  border: none;
  color: white;
  padding: 15px 32px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  font-size: 12px;
  border-radius: 16px;
}
```

```
.button:hover {
  box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0 rgba(0,0,0,0.19);
}
form {border: 3px solid #f1f1f1; margin-left:400px;margin-right:400px;}

input[type=text], input[type=password] {
  width: 100%;
  padding: 12px 20px;
  display: inline-block;
  margin-bottom:18px;
  border: 1px solid #ccc;
  box-sizing: border-box;
}

button {
  background-color: #091425;
  color: white;
  padding: 14px 20px;
  margin-bottom:10px;
  border: none;
  cursor: pointer;
  width: 17%;
  border-radius:4px;
  font-family:Montserrat;
}

button:hover {
  opacity: 0.8;
}

.cancelbtn {
  width: auto;
  padding: 10px 18px;
  background-color: #f44336;
}

.imgcontainer {
  text-align: center;
  margin: 24px 0 12px 0;
}
```

```css
img.avatar {
  width: 30%;
  border-radius: 50%;
}

.container {
  padding: 16px;
}

span.psw {
  float: right;
  padding-top: 16px;
}

/* Change styles for span and cancel button on extra small screens */
@media screen and (max-width: 300px) {
  span.psw {
     display: block;
     float: none;
  }
  .cancelbtn {
     width: 100%;
  }
}

.home{
  margin:80px;

  width: 84%;
  height: 500px;
  padding-top:10px;
  padding-left: 30px;

}
.login{
  margin:80px;
  box-sizing: content-box;
  width: 84%;
  height: 420px;
  padding: 30px;
  border: 10px solid blue;
}
.left,.right{
 box-sizing: content-box;
 height: 400px;
 margin:20px;
 border: 10px solid blue;
}

.mySlides {display: none;}
img {vertical-align: middle;}

/* Slideshow container */
.slideshow-container {
  max-width: 1000px;
  position: relative;
  margin: auto;
}

/* Caption text */
.text {
  color: #f2f2f2;
  font-size: 15px;
  padding: 8px 12px;
  position: absolute;
  bottom: 8px;
  width: 100%;
  text-align: center;
}
/* The dots/bullets/indicators */
.dot {
  height: 15px;
  width: 15px;
  margin: 0 2px;
  background-color: #bbb;
  border-radius: 50%;
  display: inline-block;
  transition: background-color 0.6s ease;
}
```

```
.active {
  background-color: #FCAD98;
}

/* Fading animation */
.fade {
  -webkit-animation-name: fade;
  -webkit-animation-duration: 1.5s;
  animation-name: fade;
  animation-duration: 1.5s;
}

@-webkit-keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

@keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

/* On smaller screens, decrease text size */
@media only screen and (max-width: 300px) {
  .text {font-size: 11px}
}


@import
url('https://fonts.googleapis.com/css2?family=Poppins&display=swap');

* {
  box-sizing: border-box;
}

body {
  min-height: 100vh;
  margin: 0;
  color: #fff;
  font-family: 'Poppins',sans-serif;
  display: flex;
  align-items: center;
  justify-content: center;
  background-color: #f5f5f5;
}

.container {
  max-width: 1376px;
  margin: auto;
  padding: 2rem 1.5rem;
}

.cards {
  display: flex;
  flex-wrap: wrap;
  align-items: center;
  justify-content: center;
}

.card {
  cursor: pointer;
  background-color: transparent;
  height: 300px;
  perspective: 1000px;
  margin: 1rem;
  align-items: center;
  justify-content: center;
}

.card h3 {
  border-bottom: 1px #fff solid;
  padding-bottom: 10px;
  margin-bottom: 10px;
  text-align: center;
  font-size: 1.6rem;
  word-spacing: 3px;
}

.card p{
  opacity: 0.75;
  font-size: 0.8rem;
  line-height: 1.4;
}

.card img {
  width: 360px;
  height: 300px;
  object-fit: cover;
  border-radius: 3px;
}

.card-inner {
  position: relative;
  width: 360px;
  height: 100%;
  transition: transform 0.9s;
  transform-style: preserve-3d;
}

.card:hover .card-inner {
  transform: rotateY(180deg);
}

.card-front,
.card-back {
  position: absolute;
  width: 360px;
  height: 100%;
  -webkit-backface-visibility: hidden;
  backface-visibility: hidden;
}

.card-back {
  background-color: #222;
  color: #fff;
  padding: 1.5rem;
  transform: rotateY(180deg);
}
.text-block {
  position: absolute;
  bottom: 20px;
  right: 20px;
  background-color: black;
  color: white;

  padding-left: 20px;
  padding-right: 20px;
}
p
{
color:black;
font-style:italic;
font-size:30px;
}
</style>

<body style="background-
image:url({{url_for('static',filename='images/bck3.png')}});background-
position: center;background-repeat: no-repeat;
  background-size: cover;">
<div class="header">
<div style="width:50%;float:left;font-size:2vw;text-align:left;color:black;
padding-top:1%;padding-left:5%;">Real Time Communication System for
Deaf & Dumb</div>
  <div class="topnav-right"style="padding-top:0.5%;">

    <a class="active" href="/home">Home</a>
    <a href="/upload">Open Web Cam</a>
  </div>
</div>

<div class="container">
    <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. 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. 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.</p>
</div>

</body>
<html>
```

Upload.html

```
<html lang="en">

<head>
   <title>Conversation Engine</title>
   <link
href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css"
rel="stylesheet">
<style>
.header { position: relative;
      top:0;
      margin:0px;
      z-index: 1;
      left: 0px;
      right: 0px;
      position: fixed;
      background-color: #F36262 ;
      color: white;
      box-shadow: 0px 8px 4px grey;
      overflow: hidden;
      padding-left:20px;
      font-family: 'Josefin Sans';
      font-size: 2vw;
      width: 100%;
      height:8%;
      text-align: center;
   }
   .topnav {
  overflow: hidden;
  background-color: #FCAD98;
```

```
}

.topnav-right a {
  float: left;
  color: black;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
  font-size: 18px;
}

.topnav-right a:hover {
  background-color: #FCAD98;
  color: black;
}

.topnav-right a.active {
  background-color: #FCAD98;
  color: white;
}

.topnav-right {
  float: right;
  padding-right:100px;
}

body {

  background-color: ;
  background-repeat: no-repeat;
  background-size:cover;
  background-image:
url("https://i.pinimg.com/originals/b2/1d/c6/b21dc69346915015bc4e19bd5
02f401b.gif");
    background-size: cover;
  background-position: 0px 0px;
  }
  .button {
  background-color: #091425;
  border: none;
  color: white;
  padding: 15px 32px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  font-size: 12px;
  border-radius: 16px;
}
.button:hover {
  box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0 rgba(0,0,0,0.19);
}
form {border: 3px solid #f1f1f1; margin-left:400px;margin-right:400px;}

input[type=text], input[type=password] {
  width: 100%;
  padding: 12px 20px;
  display: inline-block;
  margin-bottom:18px;
  border: 1px solid #ccc;
  box-sizing: border-box;
}

button {
  background-color: #091425;
  color: white;
  padding: 14px 20px;
  margin-bottom:10px;
  border: none;
  cursor: pointer;
  width: 17%;
  border-radius:4px;
  font-family:Montserrat;
}

button:hover {
  opacity: 0.8;
}

.cancelbtn {
  width: auto;
  padding: 10px 18px;
  background-color: #f44336;
}

.imgcontainer {
  text-align: center;
  margin: 24px 0 12px 0;
}

img.avatar {
  width: 30%;
  border-radius: 50%;
}

.container {
  padding: 16px;
}

span.psw {
  float: right;
  padding-top: 16px;
}

/* Change styles for span and cancel button on extra small screens */
@media screen and (max-width: 300px) {
  span.psw {
     display: block;
     float: none;
  }
  .cancelbtn {
     width: 100%;
  }
}

.home{
  margin:80px;

  width: 84%;
  height: 500px;
  padding-top:10px;
```

```css
  padding-left: 30px;

}
.login{
  margin:80px;
  box-sizing: content-box;
  width: 84%;
  height: 420px;
  padding: 30px;
  border: 10px solid blue;
}
.left,.right{
 box-sizing: content-box;
 height: 400px;
 margin:20px;
 border: 10px solid blue;
}

.mySlides {display: none;}
img {vertical-align: middle;}

/* Slideshow container */
.slideshow-container {
  max-width: 1000px;
  position: relative;
  margin: auto;
}

/* Caption text */
.text {
  color: #f2f2f2;
  font-size: 15px;
  padding: 8px 12px;
  position: absolute;
  bottom: 8px;
  width: 100%;
  text-align: center;
}
/* The dots/bullets/indicators */
.dot {
  height: 15px;
  width: 15px;
  margin: 0 2px;
  background-color: #bbb;
  border-radius: 50%;
  display: inline-block;
  transition: background-color 0.6s ease;
}

.active {
  background-color: #FCAD98;
}

/* Fading animation */
.fade {
  -webkit-animation-name: fade;
  -webkit-animation-duration: 1.5s;
  animation-name: fade;
  animation-duration: 1.5s;
}

@-webkit-keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

@keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

/* On smaller screens, decrease text size */
@media only screen and (max-width: 300px) {
  .text {font-size: 11px}
}

.bar
{
margin: 0px;
padding:20px;
```

```
background-color:white;
opacity:0.6;
color:black;
font-family:'Roboto',sans-serif;
font-style: italic;
border-radius:20px;
font-size:25px;
}
a
{
color:grey;
float:right;
text-decoration:none;
font-style:normal;
padding-right:20px;
}
a:hover{
background-color:black;
color:white;
border-radius:15px;0
font-size:30px;
padding-left:10px;
}

p
{
color:black;
font-style:italic;
font-size:30px;
}
</style>
</head>

<body style="background-
image:url({{url_for('static',filename='images/bck3.png')}});background-
position: center;background-repeat: no-repeat;
  background-size: cover;">

<div class="header">
```

```
<div style="width:70%;float:left;font-size:2vw;text-align:left;color:black;
padding-top:1%;padding-left:5%;">Real Time Communication System for
Deaf & Dumb - PNT2022TMID26243</div>
 <div class="topnav-right"style="padding-top:0.5%;">

   <a  href="/home">Home</a>
   <a class="active" href="/upload">Open Web Cam</a>
 </div>
</div>

</body>
```

GitHub Link:

https://github.com/IBM-EPBL/IBM-Project-12053-1659368179

Project Demo Video Link:

https://drive.google.com/drive/folders/1JhpSYNRQTtAqGfe4vaA4GDKg7UVhmYM8?usp=share_link