REAL TIME COMMUNICATION SYSTEM POWERED BY AI FOR SPECIALLY ABLED

PROJECT BASED LEARNING

Submitted by

A.R. AARTHI (910619104001)

A. ARUNA (910619104007)

G. HARINI (910619104024)

G. HARSHA VARDHINI (910619104028)

in partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING

K.L.N. COLLEGE OF ENGINEERING, POTTAPALAYAM
ANNA UNIVERSITY: CHENNAI 600 025.

NOVEMBER 2022

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report **“Real-Time Communication System Powered By AI For Specially Abled”** is the bonafide work of “AARTHI A R (910619104001)”, “ARUNA A (910619104007)”, “HARINI G (910619104024)” and “HARSHAVARDHINI G (910619104028)” who carried out the project under my supervision.

SIGNATURE

USHARANI K

ASSISTANT PROFESSOR-2

Computer Science and Engineering

K.L.N College of Engineering

Pottapalayam,

Sivagangai-630 612.

SIGNATURE

PREM KUMAR P

ASSOCIATE PROFESSOR

Computer Science and Engineering

K.L.N College of Engineering

Pottapalayam,

Sivagangai-630 612.

Submitted for the Project viva-voice conducted on DATE

MENTOR

EVALUATOR

ABSTRACT

Communication plays a significant role in making the world a better place. Communication creates bonding and relations among the people, whether persona, social, or political views. Most people communicate efficiently without any issues, but many cannot be due to disability. They cannot hear or speak, which makes Earth a problematic place to live for them. Even simple basic tasks become difficult for them. Disability is an emotive human condition. It limits the individual to a certain level of performance. Being deaf and dumb pushes the subject to oblivion, highly introverted. In a world of inequality, this society needs empowerment. Harnessing technology to improve their welfare is necessary. In a tech era, no one should be limited due to his or her inability. The application of technology should create a platform or a world of equality despite the natural state of humans. On the other hand, technology is the most innovative thing on Earth for every time the clock ticks, researchers, software engineers, programmers, and information technology specialists are always coming up with bright ideas to provide convenience to everyone. This paper shows how artificial intelligence is being used to help people who are unable to do what most people do in their everyday lives. Aligned with communication, D-talk is a system that allows people who are unable to talk and hear be fully understood and for them to learn their language easier and for the people that would interact and communicate with them. This system provides detailed hand gestures that show the interpretation at the bottom so that everyone can understand them. This research allows the readers to learn the system and what it can do to people who are struggling with what they are not capable of and will provide the technical terms on how the system works.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABBREVIATIONS	EXPANSION
CNN	Convolutional Neural Network
VCS	Voice Conversion System
HGR	Hand Gesture Recognition
ANN	Artificial Neural Network

1.INTRODUCTION

1. INTRODUCTION

1.1 Project Overview

In our society, we have people with disabilities. The technology is developing day by day, but no significant developments are undertaken for the betterment of these people. 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.

1.2 Purpose

The project aims 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. We are making use of a convolution neural network to create a model that is trained on different hand gestures. An app is built which uses this model. This app enables deaf and dumb people to convey their information using signs which get converted to human-understandable language and speech is given as output.

2.LITERATURE SURVEY

2. LITERATURE SURVEY

2.1 Existing problem

There are handicapped people in our society. Although technology is constantly evolving, little is being done to improve the lives of these people. It has always been difficult to communicate with someone who is deaf-mute. It is challenging for mute persons to communicate with hearing people. Because hand sign language is not taught to the general public. It might be quite challenging for them to communicate at times of crisis. In circumstances where other forms of communication, like speech, are not possible, the human hand has remained a common choice for information transmission. To have a proper conversation between a normal person and an impaired person in any language, a Voice Conversion System with Hand Gesture Recognition and Translation will come in handy.

2.2 References

[1] Design of communication Interpreter for Deaf and Dumb person was published by Pallavi Verma (Electrical and Electronics Department, Amity University, Greater Noida, Uttar Pradesh, India), Shimi S. L (Assistant Professor, NITTTR, Chandigarh, India), Richa Priyadarshini(Electrical and Electronics Department, Amity University, Greater Noida, Uttar Pradesh, India).

[2] International Journal of Science and Research (IJSR) • Jan 2013Development of full duplex intelligent communication system for deaf and dumb people was published in the year January 2017DOI:10.1109/CONFLUENCE.2017.7943247

[3] At 7th International Conference on Cloud Computing, Data Science & Engineering - Confluence (Confluence) by Surbhi Rathi Department of Information Technology, Yeshwantrao Chavan College of Engineering Nagpur, India and Ujwalla Gawande, Department of Information Technology Yeshwantrao Chavan College of Engineering Nagpur, India.

[4] A Review Paper on Sign Language Recognition for The Deaf and Dumb published by R Rumana(B.E Graduate(IV year), Department of Computer Science and Engineering, SCSVMV, Kanchipuram) , Reddygari Sandhya Rani(B.E Graduate(IV year), Department of Computer Science and Engineering, SCSVMV, Kanchipuram)

2.3 Problem Statement Definition

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. Communication plays a significant role in making the world better place. It creates a bonding and relations among the people. Only specially abled people are taught sign language and the common person is unaware its working causing a communication gap. Under the emergency situations, it is even more difficult for specially abled people to get help. Nonemergency normal environments can also be hard for them to navigate needing special assistance. The project aims 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 using the convolution neural network. An app is built which enables the deaf and dumb people to convey their information using signs which is converted to human understandable language and output is given as speech.

3. IDEATION & PROPOSED SOLUTION

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

Empathy Map Canvas is a tool to understand customers, provided by XPLANE Company. Using this tool, you can easily obtain customer characteristics. Of course, its uniqueness lies in the fact that it operates beyond demographic characteristics such as living area or income level. Empathy Map Canvas provides a better understanding of the environment, behaviour, concerns, and desires of the customer. This map makes it easier for us to identify potential customers.

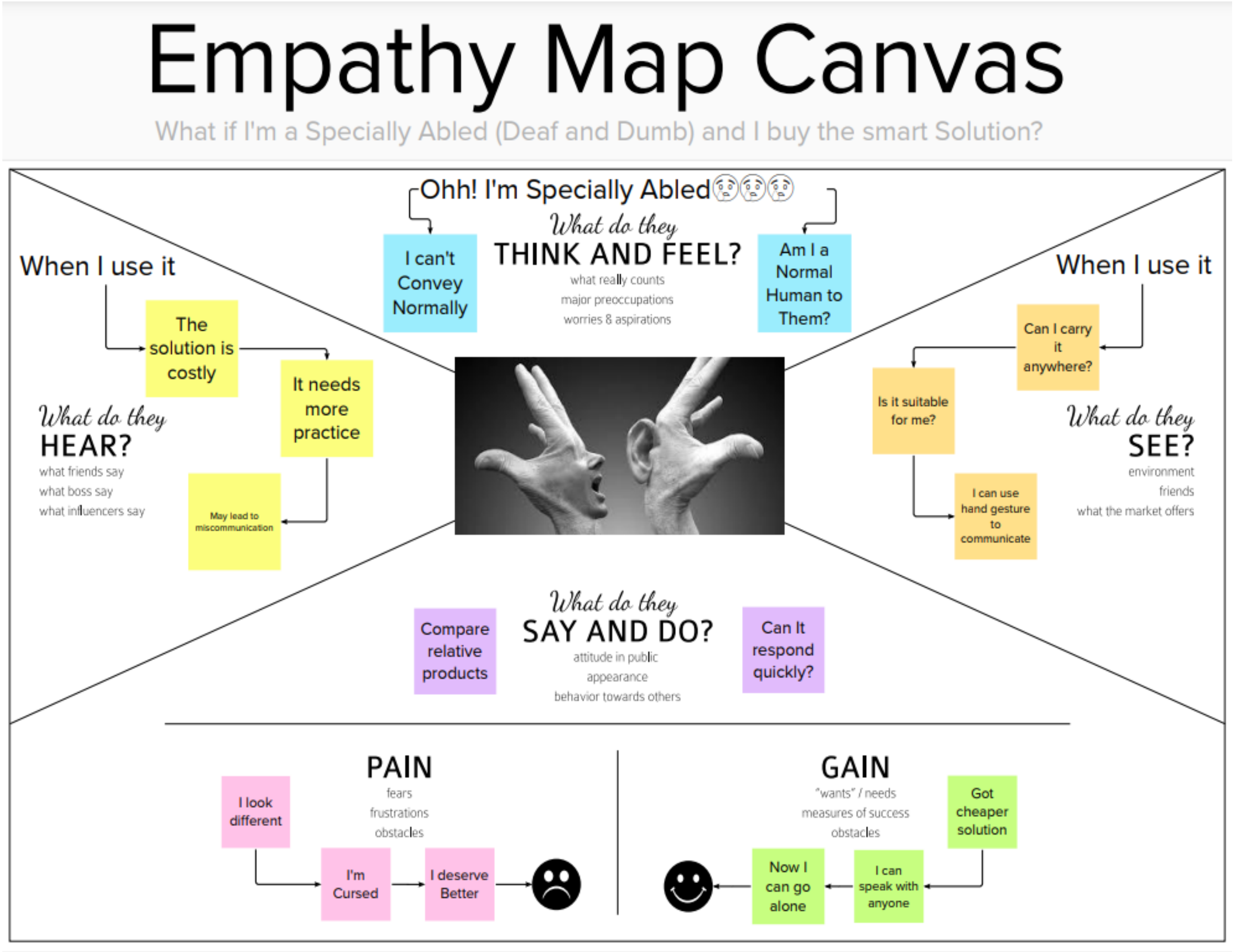
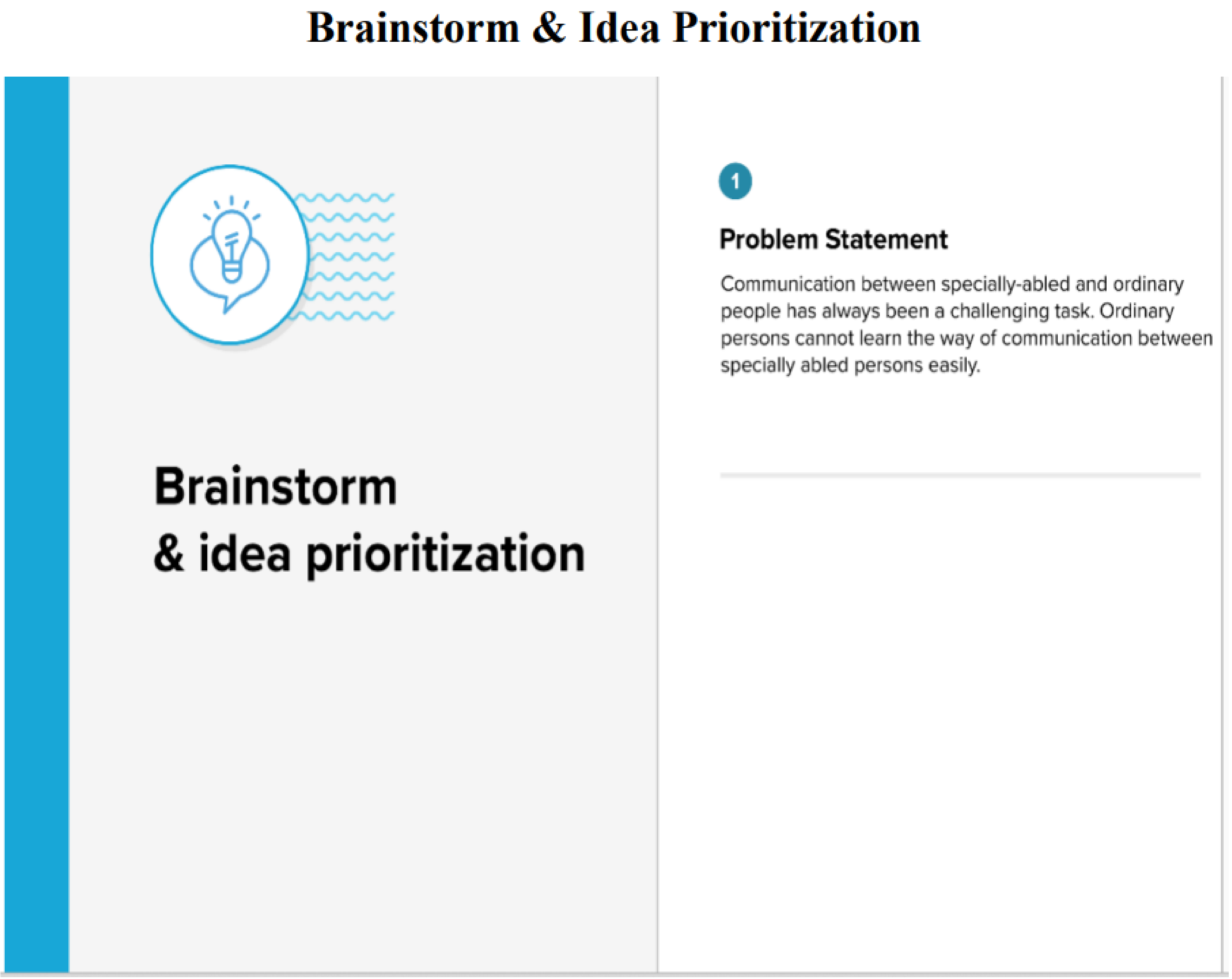


Figure no:3.1.1 Empathy Map Canvas

3.2 Ideation & Brainstorming

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. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich number of creative solutions.



Brainstorm & Idea Prioritization

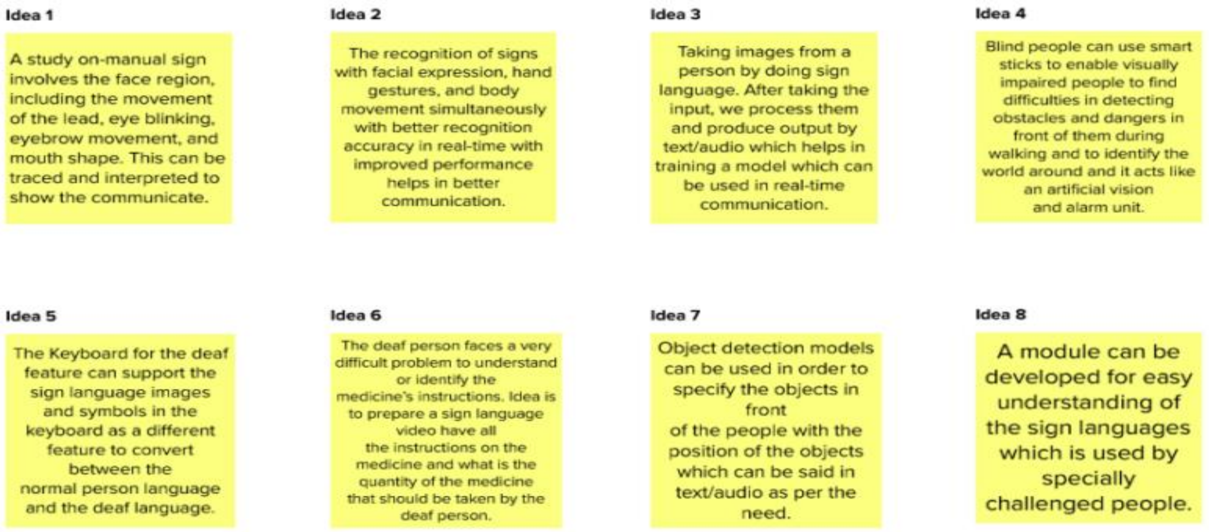
Brainstorm
& idea prioritization

1

Problem Statement

Communication between specially-abled and ordinary people has always been a challenging task. Ordinary persons cannot learn the way of communication between specially abled persons easily.

Figure no:3.2.1 Ideation



Idea 1

A study on-manual sign involves the face region, including the movement of the lead, eye blinking, eyebrow movement, and mouth shape. This can be traced and interpreted to show the communicate.

Idea 2

The recognition of signs with facial expression, hand gestures, and body movement simultaneously with better recognition accuracy in real-time with improved performance helps in better communication.

Idea 3

Taking images from a person by doing sign language. After taking the input, we process them and produce output by text/audio which helps in training a model which can be used in real-time communication.

Idea 4

Blind people can use smart sticks to enable visually impaired people to find difficulties in detecting obstacles and dangers in front of them during walking and to identify the world around and it acts like an artificial vision and alarm unit.

Idea 5

The Keyboard for the deaf feature can support the sign language images and symbols in the keyboard as a different feature to convert between the normal person language and the deaf language.

Idea 6

The deaf person faces a very difficult problem to understand or identify the medicine's instructions. Idea is to prepare a sign language video have all the instructions on the medicine and what is the quantity of the medicine that should be taken by the deaf person.

Idea 7

Object detection models can be used in order to specify the objects in front of the people with the position of the objects which can be said in text/audio as per the need.

Idea 8

A module can be developed for easy understanding of the sign languages which is used by specially challenged people.

Figure no:3.2.2 Brainstorming

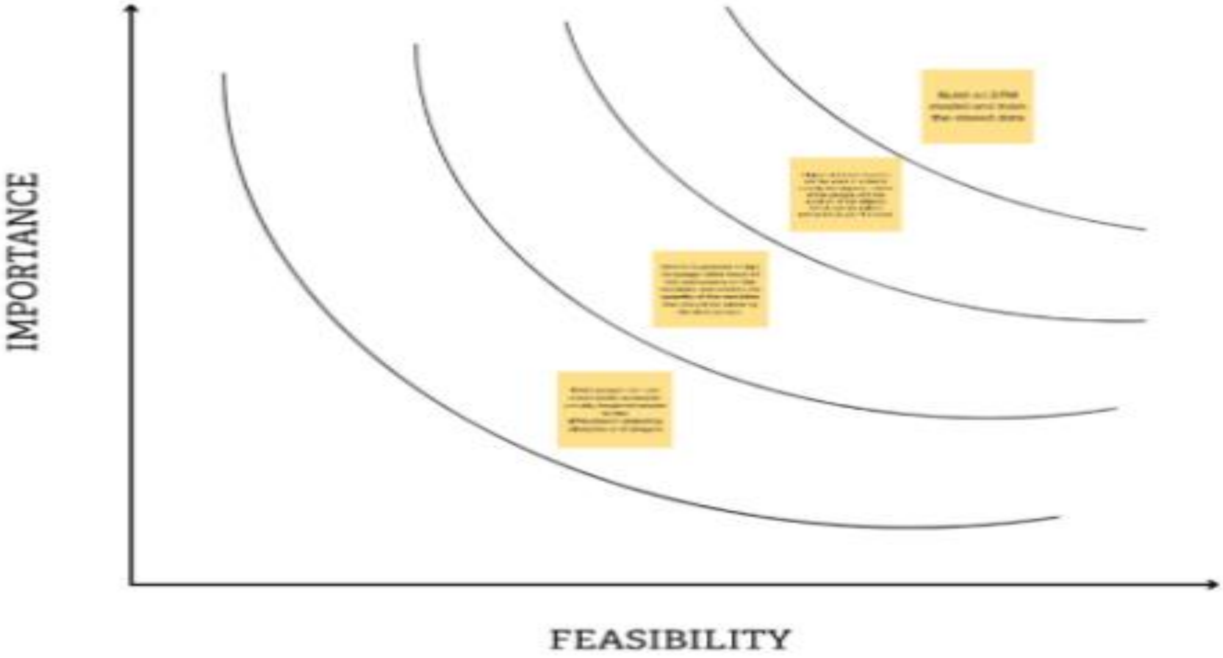
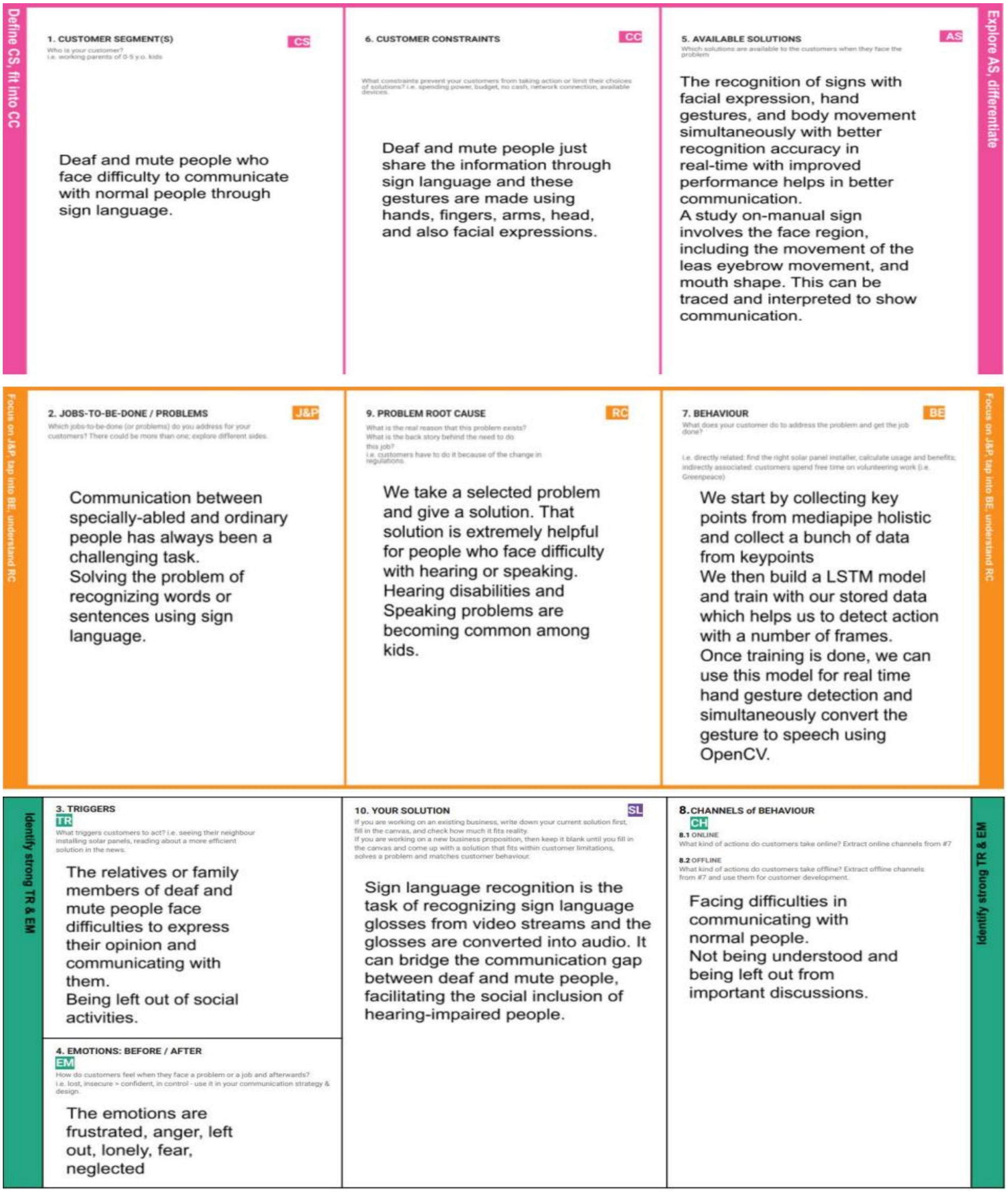


Figure no:3.2.3 Idea Prioritization

3.3 Proposed Solution

S.No.	Parameter	Description
1	Problem Statement (Problem to be solved) Problem Statement (Problem to be solved)	To model a system for aiding deaf and dumb people and help them to communicate in real-time.
2	Idea / Solution description	We start by collecting key points from media pipe holistic and collect a bunch of data from key points We then build a LSTM model and train with our stored data which helps us to detect action with several frames. Once training is done, we can use this model for real time hand gesture detection and simultaneously convert the gesture to speech using OpenCV.
3	Novelty / Uniqueness	We will be using the latest and trending wearable technology which makes it possible to access (Web Application) easily anywhere and everywhere by the disabled person which makes the communication possible by both specially abled and normal people. We will be using the most recent convolution neural network architecture to improve the efficiency of the trained model
4	Social Impact / Customer Satisfaction	Helps to bridge the gaps in communication with hearing and speaking impaired people.
5	Business Model (Revenue Model)	The implemented product will be marketed as a Retailer model, in which the product will be assigned an initial base price and will be updated once we bring new features to it
6	Scalability of the Solution	Bootstrapping the company at first through the founder’s funds, but eventually through reinvesting the profit from servicing customers.

3.4 Problem Solution fit


Define CS, fit into CC
Explore AS, differentiate
Focus on J&P, tap into BE, understand RC
Focus on J&P, tap into BE, understand RC
Identify strong TR & EM
Identify strong TR & EM

1. CUSTOMER SEGMENT(S) CS

Who is your customer?
i.e. working parents of 0-5 y.o. kids

Deaf and mute people who face difficulty to communicate with normal people through sign language.

6. CUSTOMER CONSTRAINTS CC

What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.

Deaf and mute people just share the information through sign language and these gestures are made using hands, fingers, arms, head, and also facial expressions.

5. AVAILABLE SOLUTIONS AS

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

The recognition of signs with facial expression, hand gestures, and body movement simultaneously with better recognition accuracy in real-time with improved performance helps in better communication.
A study on-manual sign involves the face region, including the movement of the leas eyebrow movement, and mouth shape. This can be traced and interpreted to show communication.

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.

Communication between specially-abled and ordinary people has always been a challenging task.
Solving the problem of recognizing words or sentences using 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?
i.e. customers have to do it because of the change in regulations.

We take a selected problem and give a solution. That solution is extremely helpful for people who face difficulty with hearing or speaking.
Hearing disabilities and Speaking problems are becoming common among kids.

7. BEHAVIOUR BE

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

i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

We start by collecting key points from mediapipe holistic and collect a bunch of data from keypoints
We then build a LSTM model and train with our stored data which helps us to detect action with a number of frames.
Once training is done, we can use this model for real time hand gesture detection and simultaneously convert the gesture to speech using OpenCV.

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.

The relatives or family members of deaf and mute people face difficulties to express their opinion and communicating with them.
Being left out of social activities.

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.

The emotions are frustrated, anger, left out, lonely, fear, neglected

10. YOUR SOLUTION SL

If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality.
If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.

Sign language recognition is the task of recognizing sign language glosses from video streams and the glosses are converted into audio. It can bridge the communication gap between deaf and mute people, facilitating the social inclusion of hearing-impaired people.

8. CHANNELS of BEHAVIOUR
CH

8.1 ONLINE
What kind of actions do customers take online? Extract online channels from #7

8.2 OFFLINE
What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.

Facing difficulties in communicating with normal people.
Not being understood and being left out from important discussions.

Figure no: 3.4.1 Problem Solution fit

4.REQUIREMENT ANALYSIS

4.REQUIREMENT ANALYSIS

4.1 Functional requirement

Following are the functional requirements of the proposed solution.

FR No.	Functional requirement (Epic)	Sub Requirement (Story/Sub-Task)
FR-1	User communication	User Must Know the Sign Language
FR-2	User communication	The user Has to communicate in Front of the Camera

4.2Non-Functional requirements

FR No	Non-Functional Requirement	Description
NFR-1	Usability	The camera captures all expressions including facial expressions and hand gestures which can be easily used by all age groups.
NFR-2	Reliability	The system is very liable, it can last for long amounts of time if well maintained.
NFR-3	Performance	The cost-effective nature of the system makes it extremely liable and thus, efficient
NFR-4	Availability	The solution fits all the sign languages when we train the model for all the sign languages. So, it is used by all the countries with different languages.
NFR-5	Scalability	The system gives output rapidly. It also predicts quickly when it gets so many inputs at a time. It predicts different types of sign language at a time.

5.PROJECT DESIGN

5.PROJECT DESIGN

5.1 Data Flow Diagram

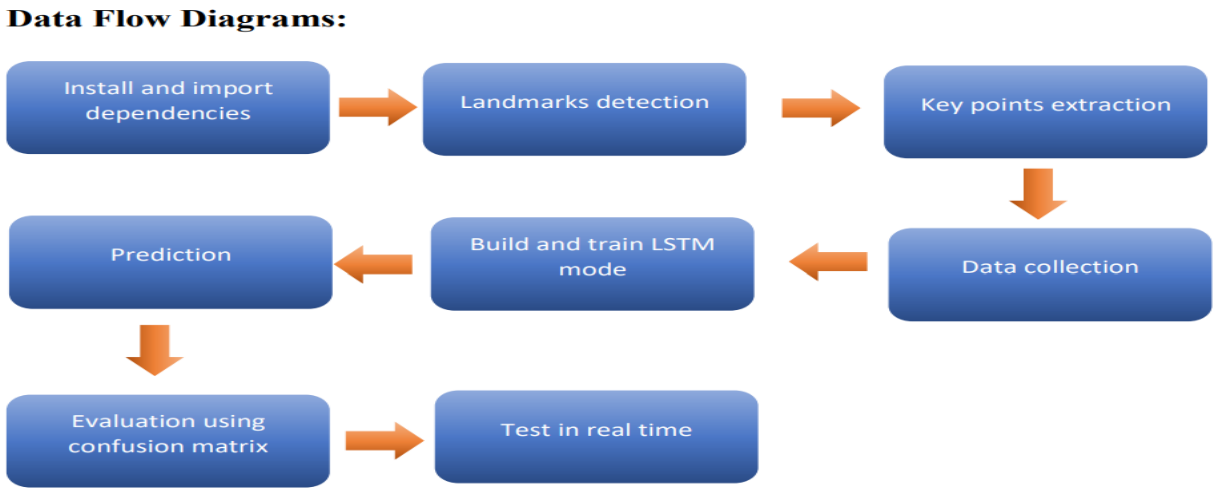


Figure no:5.1.1 Data Flow Diagram

Flow:

- We start by collecting key points from media-pipe holistic and collect a bunch of data from key-points
- Save data in the form of numpy arrays.
- We then build a LSTM model and train with our stored data
- The number of epochs for the model is determined by us, if we increase the number of epochs the accuracy increases but time taken to run the model also increases and overfitting of model can happen, for gesture recognition.
- Once training is done, we can use this model for real time hand gesture detection and simultaneously convert the gesture to speech using OpenCV.

5.2 Solution & Technical Architecture

Solution Architecture: Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions.

- Technology used are Python, Jupyter ,Numpy ,OpenCV ,Keras , Tensorflow , GTTS.
- Methodology - We start by collecting key points from media pipe holistic and collect a bunch of data from key points i.e., our hands, on our body and on our face and save data in the form of numpy arrays. We can vary the number of sequences according to our need, but each sequence will have 30 frames.

- We then build a LSTM model and train with our stored data which helps us to detect action with several frames.
- The number of epochs for the model is determined by us, if we increase the number of epochs the accuracy increases but time taken to run the model also increases and overfitting of model can happen, for gesture recognition.
- Once training is done, we can use this model for real time hand gesture detection and simultaneously convert the gesture to speech or text using OpenCV.

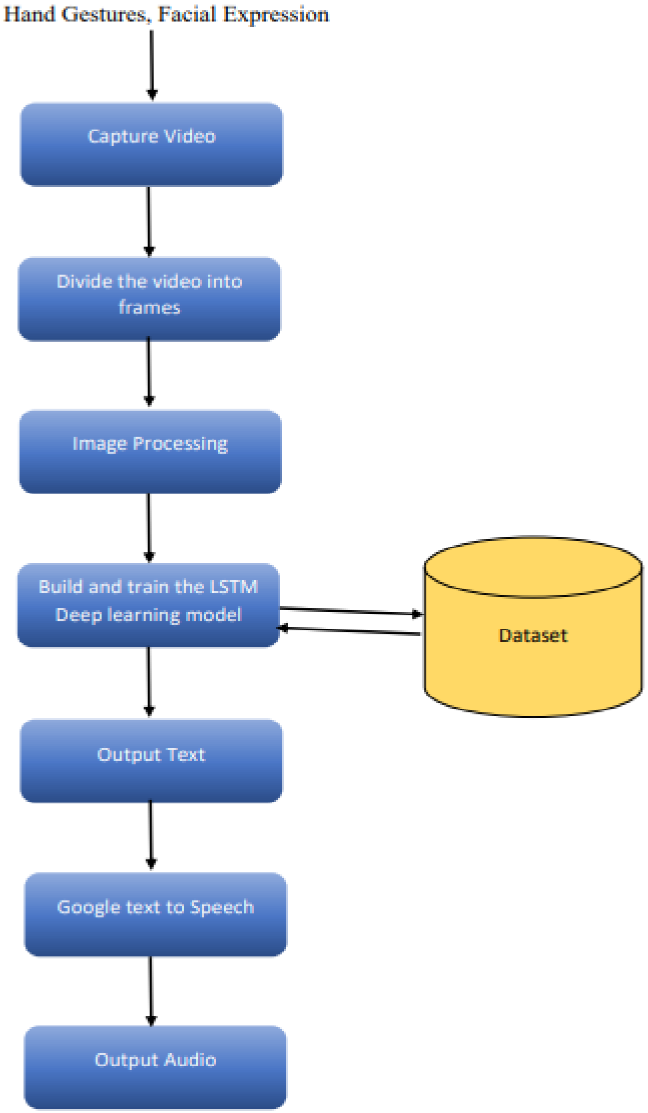

Hand Gestures, Facial Expression
Capture Video
Divide the video into frames
Image Processing
Build and train the LSTM Deep learning model
Dataset
Output Text
Google text to Speech
Output Audio

Figure no:5.2.1 Solution Architecture

Technical Architecture: The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2.

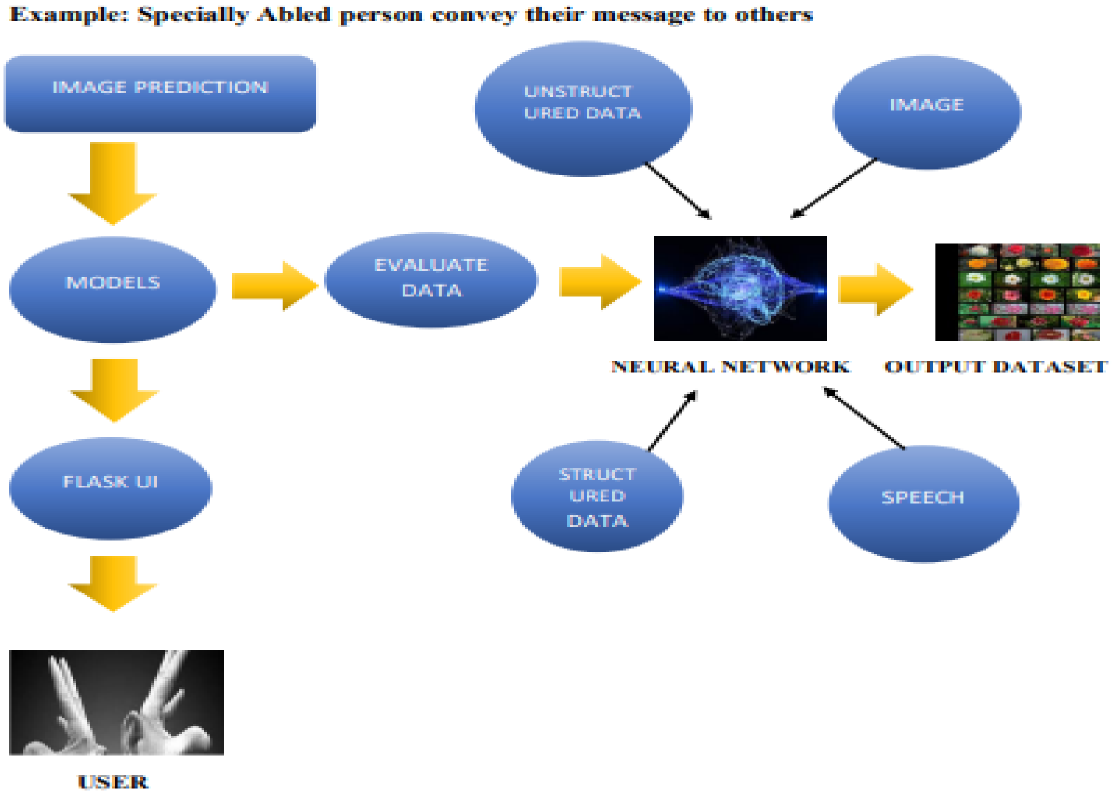


Figure no:5.2.2 Technical Architecture

TABLE-1: COMPONENTS & TECHNOLOGIES

S.No	Component	Description	Technology
1	User	Communication barriers of deaf or hearing-impaired people with other communities, contributing significantly to their social inclusion	AI technology
2	Flash UI	Flash's user interface components let you interact with the users that use your site and gather information.	Using the cloud, it can be executed
3	Models	Support Vector Machine (SVM) is subsequently applied to classify our gesture image dataset.	Machine Learning

4	Image Prediction	Gesture can be completely observable and viewing a gesture from another perspective makes the prediction.	ANN, CNN
5	Image	Image processing is used to make the image into signs by the neural network	ANN, CNN, Open CV
6	Speech	Speech translates the voice into image and sensitive neural play.	AI and machine learning methods like deep learning and neural networks
7	Evaluate date	Aims to estimate the generalization accuracy of a model on future (unseen/out-of-sample) data.	
8	Unstructured data	Unstructured data is a conglomeration of many varied types of data that are stored in their native formats	Natural Language Processing (NLP)
9	Structured data	Typically categorized as quantitative data — is highly organized and easily decipherable by machine learning algorithms	Machine language and artificial intelligence tools
10	Neural network	The same convolutional neural network architecture was used for both, the top view and the bottom view models, the only difference is the number of output units	AI technology
11	Dataset	First prototype of this system it was used a dataset of 24 static signs from the Panamanian Manual Alphabet.	AI technology

TABLE-2: Application Characteristics

S.No	Characteristics	Description	Technology
1	Open-Source Frameworks	Robots and other tools provide homebased care and other assistance, allowing people with disabilities to live independently	Artificial Intelligence like robots and software systems
2	Security Implementations	Set the inclusion and exclusion criteria, Report the results in the survey	Artificial Intelligence
3	Scalable Architecture	The improvement in the specially abled persons interaction with the environments	Artificial Intelligence
4	Availability	Technology solutions that mimic humans and use logic from playing chess to solving equations and Machine learning is one of the technologies	Artificial Intelligence
5	Performance	Enables people with disabilities to step into a world where their difficulties are understood and considered.	Artificial Intelligence

5.3 User Stories

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

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Priority	Priority
Developer	Model Building	USN-1	Collect Dataset	High	Sprint-1
		USN-2	Collecting Key points using Media Pipe Holistic	High	Sprint-1
		USN-3	Training a Model Using LSTM from key Points	High	Sprint-2
		USN-4	Convert text to Speech using google Api	Medium	Sprint-2
		USN-5	Model is integrated in flask app	High	Sprint-3
Communication	Communication	USN-1	Communicating in Front of camera	High	Sprint-1
		USN-2	Speech and text are delivered by web interface	High	Sprint-1

6.PROJECT PLANNING

&

SCHEDULING

6.PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Dataset Collection	USN-1	Collect Dataset for building model.	9	High	Harshavardhini Harini Aruna
Sprint-1	Image Pre-processing	USN-2	Perform Pre-processing techniques on the dataset.	8	Medium	Harshavardhini
Sprint-2	Model Building	USN-3	Import the required libraries, add the necessary layers and compile the model.	10	High	Harshavardhini Aarthi Aruna Harini
Sprint-2		USN-4	Training the image classification model using CNN.	7	Medium	Harshavardhini Aarthi Aruna Harini
Sprint-3	Training and Testing the Model	USN-5	Training the model and testing the model’s performance.	9	High	Harshavardhini Aarthi Aruna Harini
Sprint-4	Application Development	USN-6	Converting the input gesture image into English Alphabets.	8	Medium	Aarthi Aruna

6.2 Sprint Delivery Schedule

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	17	6 Days	24 Oct 2022	29 Oct 2022	17	29 Oct 2022
Sprint-2	17	6 Days	31 Oct 2022	05 Nov 2022	17	05 Nov 2022
Sprint-3	9	6 Days	07 Nov 2022	12 Nov 2022	9	12 Nov 2022
Sprint-4	8	6 Days	14 Nov 2022	19 Nov 2022	8	19 Nov 2022

Velocity:

$$AV = \frac{sprint\ duration}{velocity}$$

$$AV = \frac{6}{10} = 0.6$$

Figure no:6.2.1 Sprint Delivery Plan

Burndown Chart:

A burndown 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.

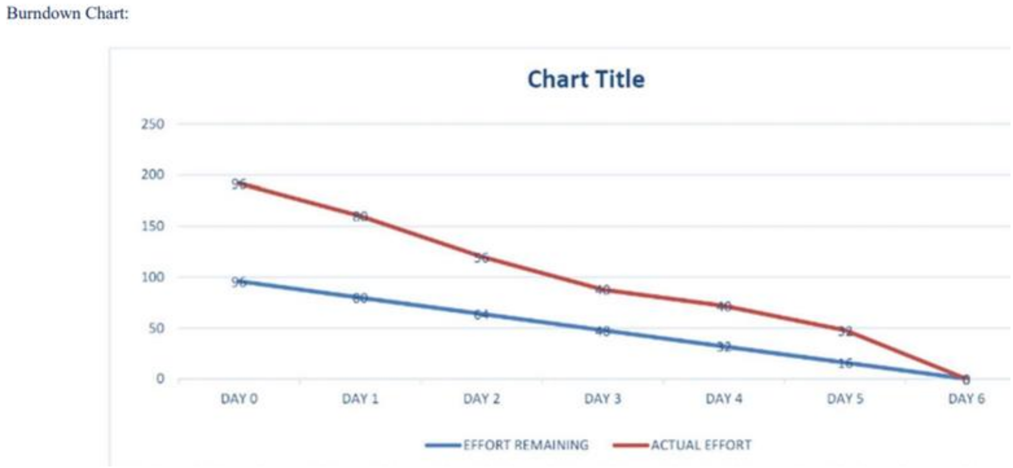
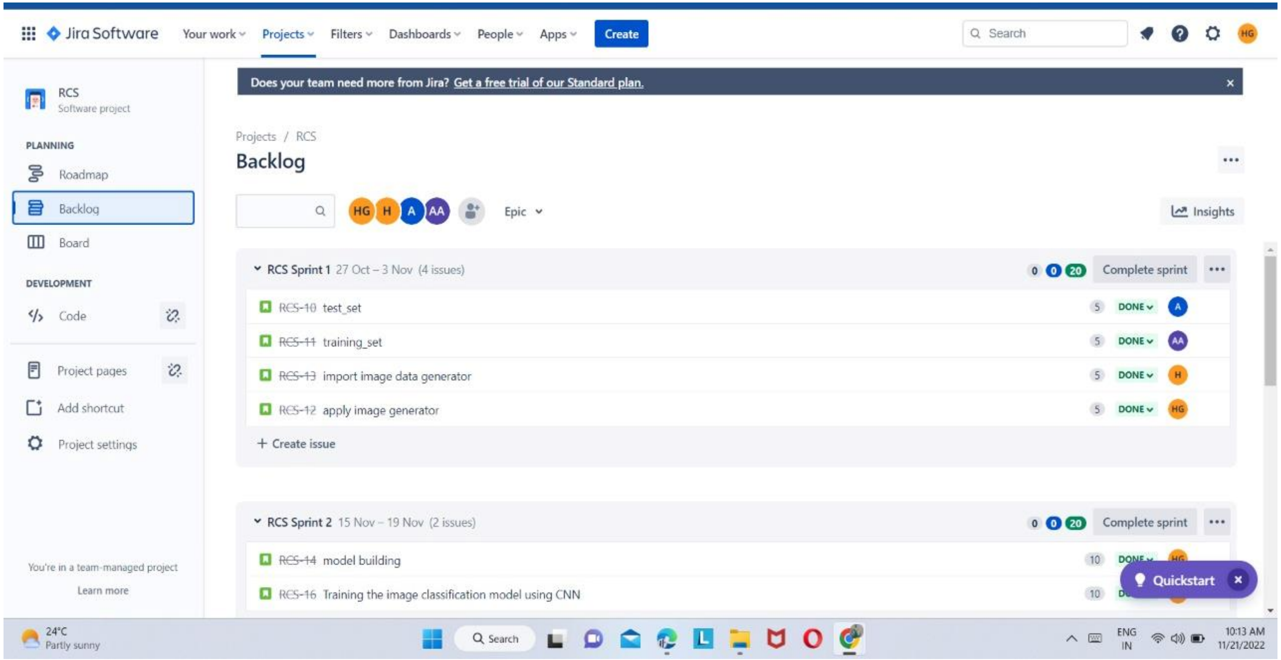


Figure no:6.2.2 Burndown Chart

6.3 Reports from JIRA



Does your team need more from Jira? Get a free trial of our Standard plan.

Projects / RCS

Backlog

RCS Sprint 1 27 Oct – 3 Nov (4 issues) 0 0 20 Complete sprint

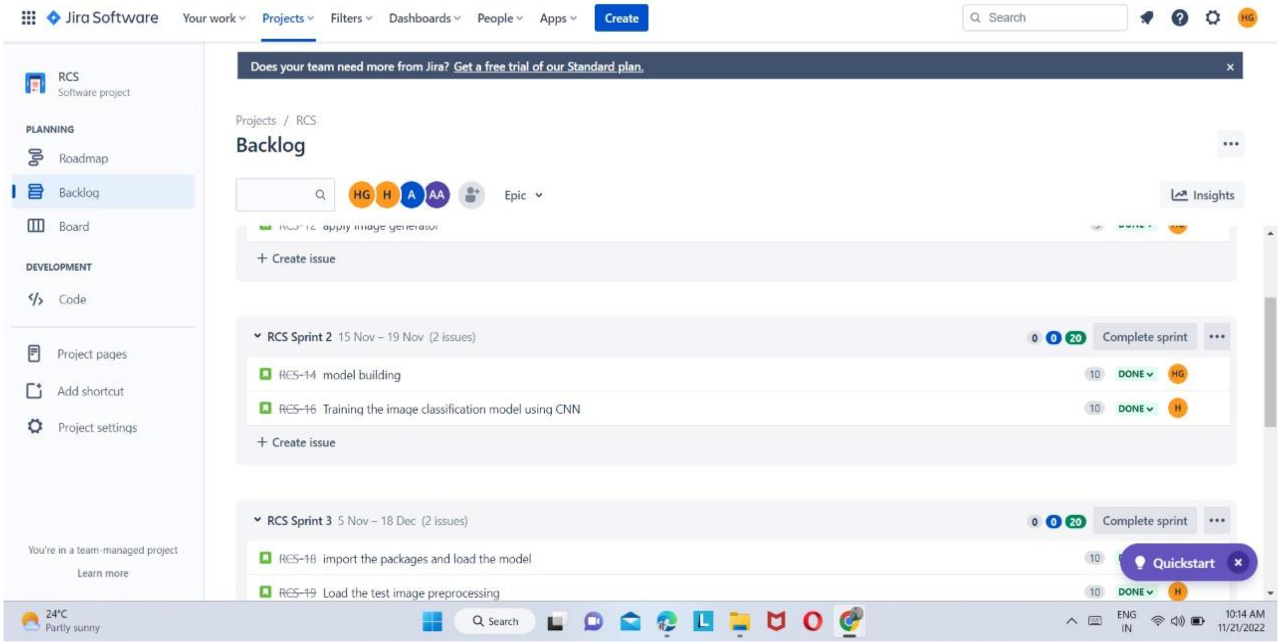
~~RCS-10~~	test_set	5	DONE	A
~~RCS-11~~	training_set	5	DONE	AA
~~RCS-13~~	import image data generator	5	DONE	H
~~RCS-12~~	apply image generator	5	DONE	HG

+ Create issue

RCS Sprint 2 15 Nov – 19 Nov (2 issues) 0 0 20 Complete sprint

~~RCS-14~~	model building	10	DONE	HG
RCS-16	Training the image classification model using CNN	10	D	

Figure no:6.3.1 JIRA Report Sprint 1



Jira Software   Your work   Projects   Filters   Dashboards   People   Apps   Create   Search

Does your team need more from Jira? Get a free trial of our Standard plan.

RCS
Software project

PLANNING
Roadmap
Backlog
Board

DEVELOPMENT
Code

Project pages
Add shortcut
Project settings

You're in a team-managed project
Learn more

Projects / RCS

Backlog

HG H A AA   Epic   Insights

~~RCS-12~~ apply image generator

+ Create issue

RCS Sprint 2 15 Nov – 19 Nov (2 issues)   0 0 20   Complete sprint

~~RCS-14~~ model building	10	DONE	HG
~~RCS-16~~ Training the image classification model using CNN	10	DONE	H

+ Create issue

RCS Sprint 3 5 Nov – 18 Dec (2 issues)   0 0 20   Complete sprint

~~RCS-18~~ import the packages and load the model	10		
~~RCS-19~~ Load the test image preprocessing	10	DONE	H

Quickstart

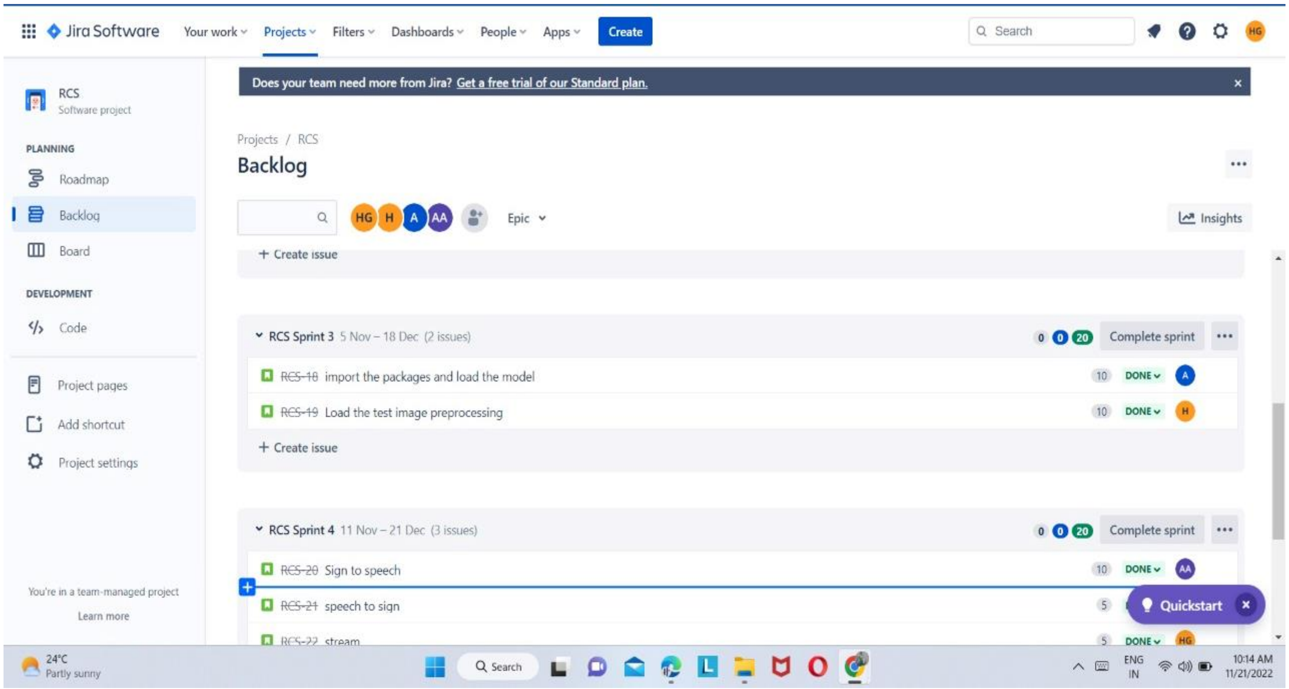
24°C
Partly sunny

Search

ENG
IN

10:14 AM
11/21/2022

Figure no:6.3.2 JIRA Report Sprint 2



Jira Software   Your work   Projects   Filters   Dashboards   People   Apps   Create   Search

Does your team need more from Jira? Get a free trial of our Standard plan.

RCS
Software project

PLANNING
Roadmap
Backlog
Board

DEVELOPMENT
Code

Project pages
Add shortcut
Project settings

You're in a team-managed project
Learn more

Projects / RCS

Backlog

HG H A AA   Epic   Insights

+ Create issue

RCS Sprint 3 5 Nov – 18 Dec (2 issues)   0 0 20   Complete sprint

~~RCS-18~~ import the packages and load the model	10	DONE	A
~~RCS-19~~ Load the test image preprocessing	10	DONE	H

+ Create issue

RCS Sprint 4 11 Nov – 21 Dec (3 issues)   0 0 20   Complete sprint

~~RCS-20~~ Sign to speech	10	DONE	AA
~~RCS-21~~ speech to sign	5		
~~RCS-22~~ stream	5	DONE	HG

Quickstart

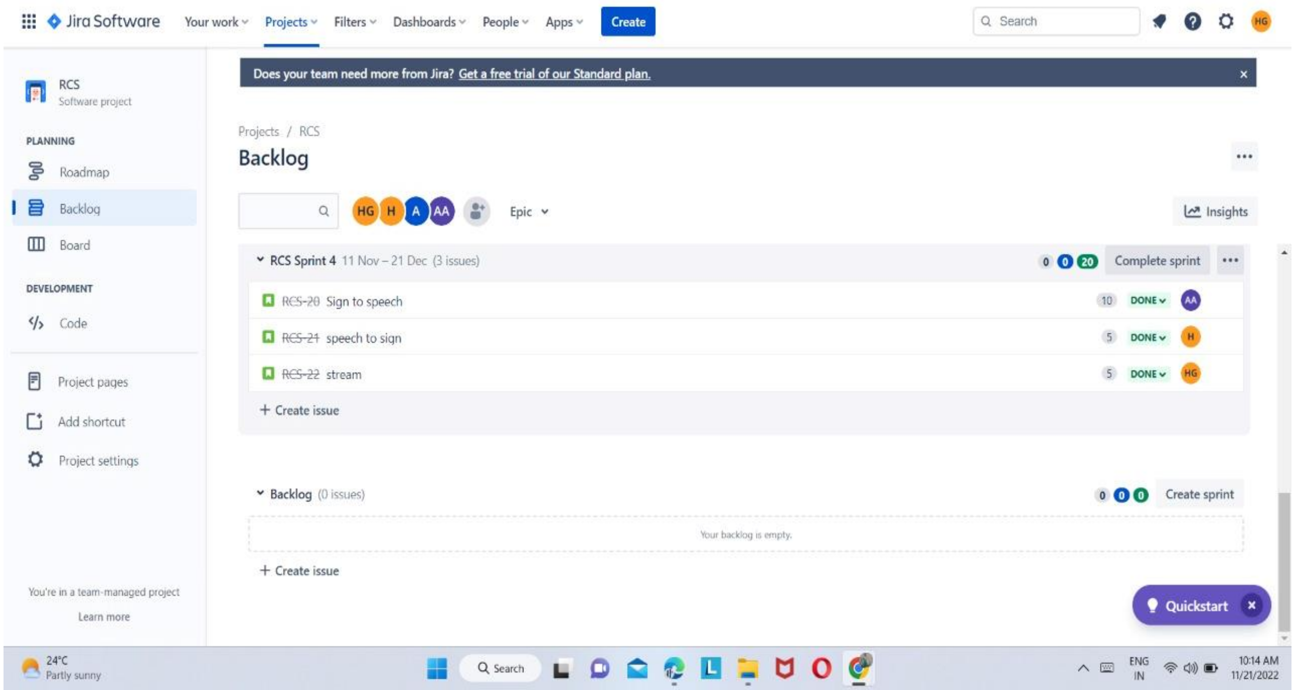
24°C
Partly sunny

Search

ENG
IN

10:14 AM
11/21/2022

Figure no:6.3.3 JIRA Report Sprint 3



Does your team need more from Jira? Get a free trial of our Standard plan.

Projects / RCS

Backlog

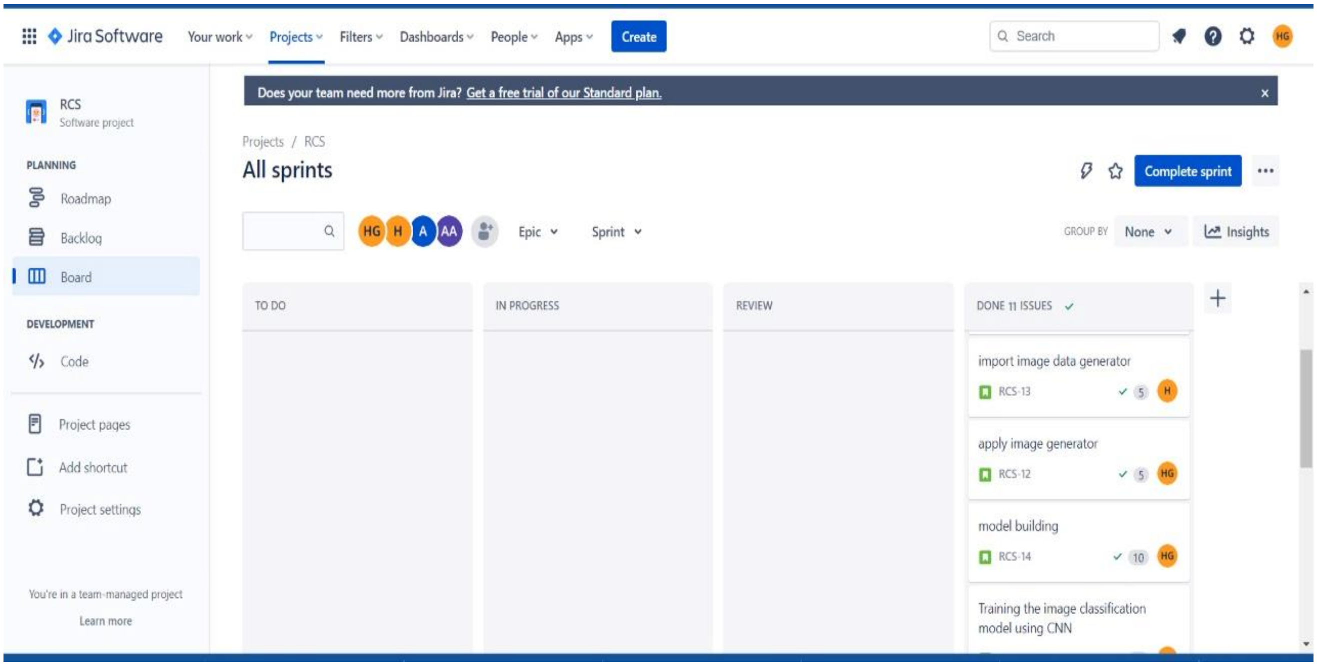
RCS Sprint 4 11 Nov – 21 Dec (3 issues) 0 0 20

~~RCS-20~~	Sign to speech	10	DONE	AA
~~RCS-21~~	speech to sign	5	DONE	H
~~RCS-22~~	stream	5	DONE	HG

Backlog (0 issues) 0 0 0

Your backlog is empty.

Figure no:6.3.4 JIRA Report Sprint 4



Does your team need more from Jira? Get a free trial of our Standard plan.

Projects / RCS

All sprints

TO DO	IN PROGRESS	REVIEW	DONE 11 ISSUES
			import image data generator RCS-13 5 H
			apply image generator RCS-12 5 HG
			model building RCS-14 10 HG
			Training the image classification model using CNN

Figure no:6.3.5 JIRA Report

7.CODING & SOLUTIONING

7.CODING & SOLUTIONING

7.1 Libraries to be installed

- pip install flask
- pip install opencv-python
- pip install numpy
- pip install keras
- pip install tensorflow
- pip install SpeechRecognition
- pip install moviepy
- pip install scikit-image
- pip install gTTS
- pip install Pillow
- pip install imutils
- pip install playsound

7.2 Feature 1

Real time sign to speech

People who are unable to talk typically use sign language to communicate. Most people find it extremely challenging to communicate with silent persons since they are unable to grasp Universal Sign Language (unless they have studied it). The core of this project is a tool that allows silent people and others communicate with one another. Our technology makes use of a CNN- built model that can recognise sign languages in real time. It quickly recognises the sign and provides feel-free cover for those with special needs.

```
img = cv2.imread('static/image.jpg') # read a image img=resize(img,(64,64,1)) # resize a image
img=image.img_to_array(img) # Convert image into array values
img=np.expand_dims(img,axis=0)
if(np.max(img)>1):
        img=img/255.0
prediction=model.predict(img)# predict the vaule using trained model

prediction=np.argmax(prediction, axis=1) pred=vals[prediction[0]] # return predicted letter
```

7.3 Feature 2

Real time speech to sign

We can use JavaScript to recognise voice thanks to the Web Speech API. JavaScript makes it very simple to recognise speech in a browser and then extract the text from the speech for use as user input. The voice is translated into text using the Speech Recognition object, which is subsequently shown on the screen as signs. This can be accomplished by our technology in real time. Any language that the user is trying to speak in can be recognised by it. But only the Chrome browser is supported for this API. The live example below will function if you are viewing this example in another browser.

```
r=sr.Recognizer() #listen a auido file=sr.AudioFile("Speech/audio.mp3")
    with file as source:
        audio_data = r.record(source)#record a source audio
    text = r.recognize_google(audio_data, language='en-IN', show_all=True)text=text['alternative']
    text=text[0]
    text=text['transcript']# predict a text using source audio
```

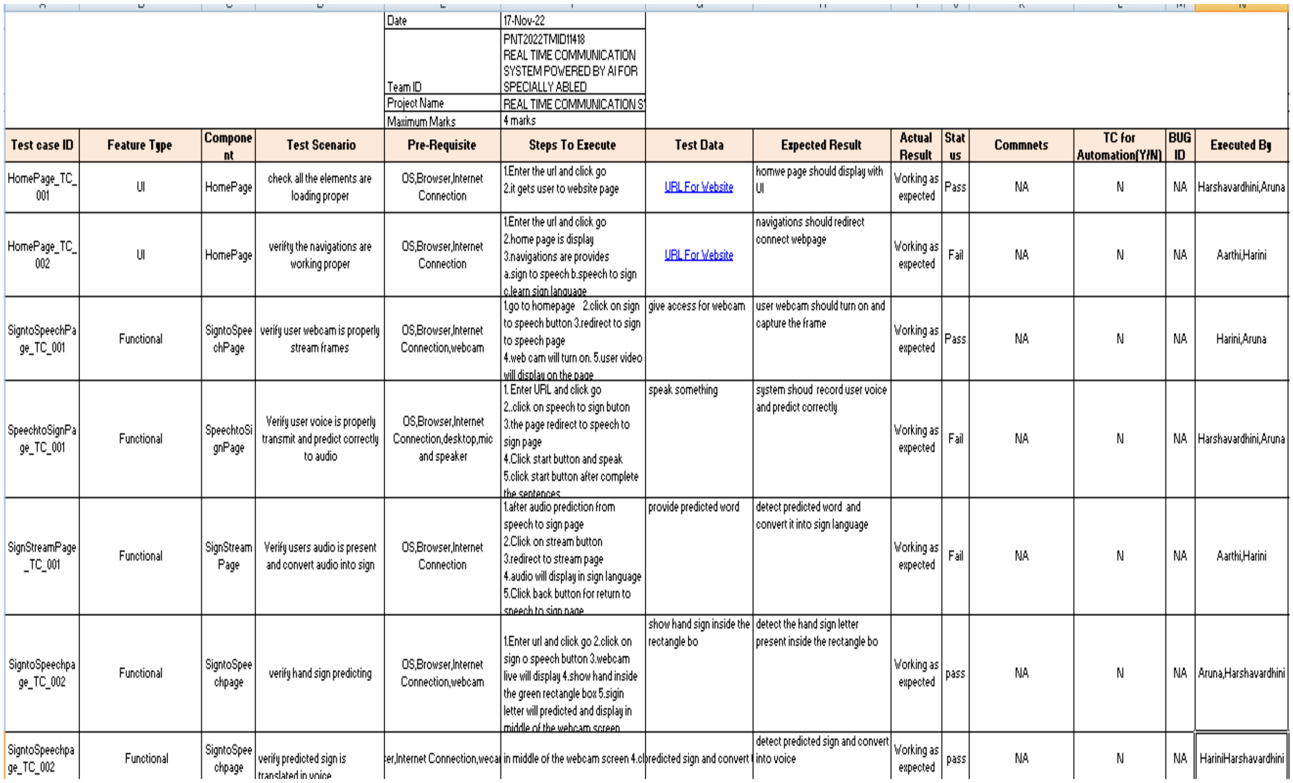
8.TESTING

8.TESTING

8.1 Test Cases

- Verify if user can see the options when user clicks the URL
- Verify if the UI elements are getting displayed properly
- Verify if the user is getting redirected to the sign to speech page
- Verify if the application can convert the sign to speech
- Verify if the user can exit the sign to speech page
- Verify if the user is getting redirected to the speech to sign page
- Verify if the UI elements are being displayed
- Verify if the application can convert speech to text on clicking voice to text button.
- Verify if the user can exit the speech to sign page.

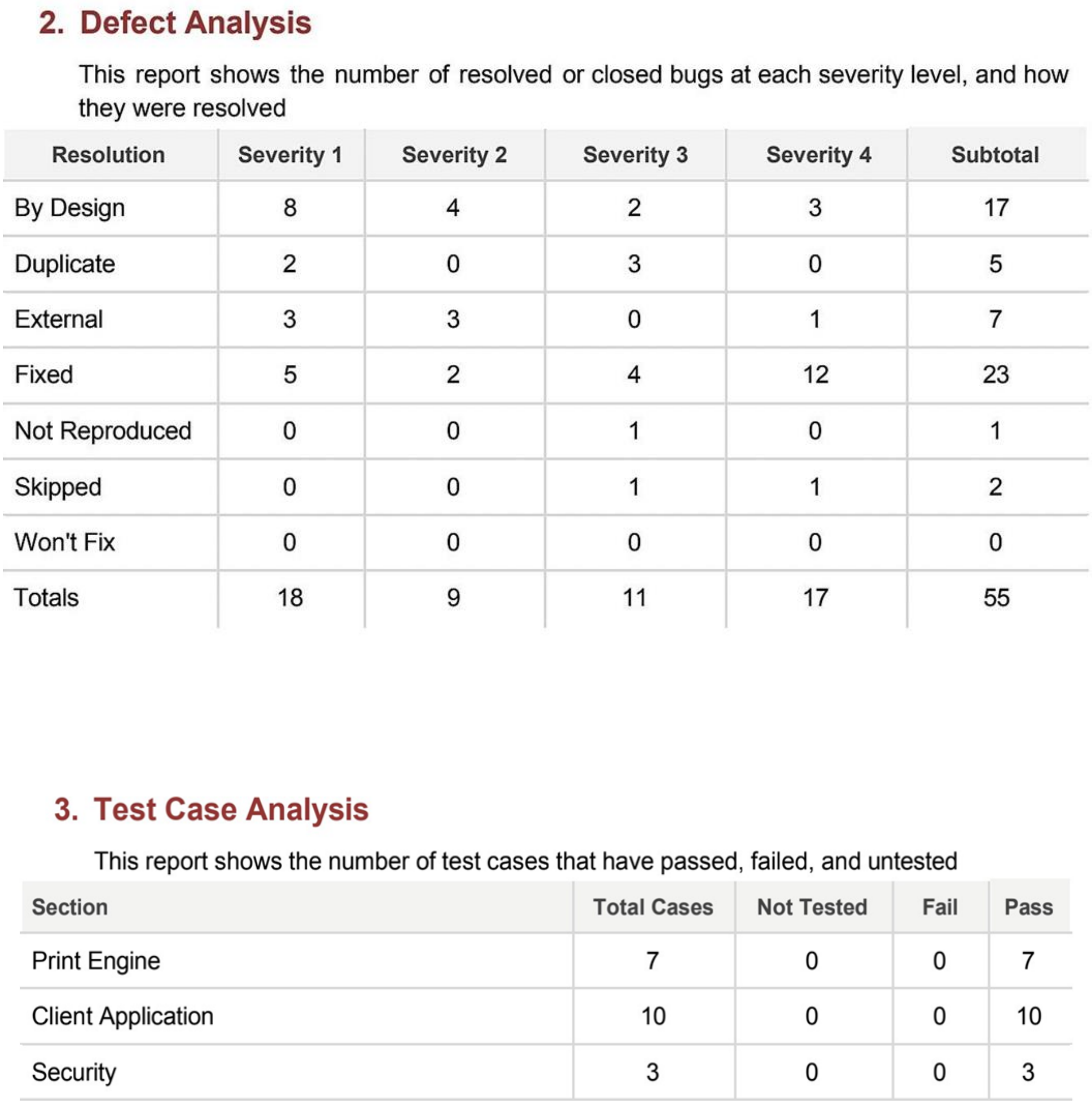
8.2 User Acceptance Testing



Date	17-Nov-22
Team ID	PNT2022TMID11418 REAL TIME COMMUNICATION SYSTEM POWERED BY AI FOR SPECIALLY ABLED
Project Name	REAL TIME COMMUNICATION S
Maximum Marks	4 marks

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation(Y/N)	BUG ID	Executed By
HomePage_TC_001	UI	HomePage	check all the elements are loading proper	OS,Browser,Internet Connection	1.Enter the url and click go 2.it gets user to website page	URL For Website	homwe page should display with UI	Working as expected	Pass	NA	N	NA	Harshavardhini,Aruna
HomePage_TC_002	UI	HomePage	verifty the navigations are working proper	OS,Browser,Internet Connection	1.Enter the url and click go 2.home page is display 3.navigations are provides a.sign to speech b.speech to sign c.learn sign language	URL For Website	navigations should redirect connect webpage	Working as expected	Fail	NA	N	NA	Aarthi,Harini
SigntoSpeechPage_TC_001	Functional	SigntoSpeechPage	verify user webcam is properly stream frames	OS,Browser,Internet Connection,webcam	1.go to homepage 2.click on sign to speech button 3.redirect to sign to speech page 4.web cam will turn on. 5.user video will display on the page	give access for webcam	user webcam should turn on and capture the frame	Working as expected	Pass	NA	N	NA	Harini,Aruna
SpeechtoSignPage_TC_001	Functional	SpeechtoSignPage	Verify user voice is properly transmit and predict correctly to audio	OS,Browser,Internet Connection,desktop,mic and speaker	1. Enter URL and click go 2..click on speech to sign buton 3.the page redirect to speech to sign page 4.Click start button and speak 5.click start button after complete the sentences	speak something	system shoud record user voice and predict correctly	Working as expected	Fail	NA	N	NA	Harshavardhini,Aruna
SignStreamPage_TC_001	Functional	SignStreamPage	Verify users audio is present and convert audio into sign	OS,Browser,Internet Connection	1.after audio prediction from speech to sign page 2.Click on stream button 3.redirect to stream page 4.audio will display in sign language 5.Click back button for return to speech to sign page	provide predicted word	detect predicted word and convert it into sign language	Working as expected	Fail	NA	N	NA	Aarthi,Harini
SigntoSpeechpage_TC_002	Functional	SigntoSpeechpage	verify hand sign predicting	OS,Browser,Internet Connection,webcam	1.Enter url and click go 2.click on sign o speech button 3.webcam live will display 4.show hand inside the green rectangle box 5.sigin letter will predicted and display in middle of the webcam screen	show hand sign inside the rectangle bo	detect the hand sign letter present inside the rectangle bo	Working as expected	pass	NA	N	NA	Aruna,Harshavardhini
SigntoSpeechpage_TC_002	Functional	SigntoSpeechpage	verify predicted sign is translated in voice	er,Internet Connection,weca	in middle of the webcam screen 4.cl	predicted sign and convert	detect predicted sign and convert into voice	Working as expected	pass	NA	N	NA	HariniHarshavardhini

Figure no:8.2.1 User Acceptance Testing



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	8	4	2	3	17
Duplicate	2	0	3	0	5
External	3	3	0	1	7
Fixed	5	2	4	12	23
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	0	0	0	0
Totals	18	9	11	17	55

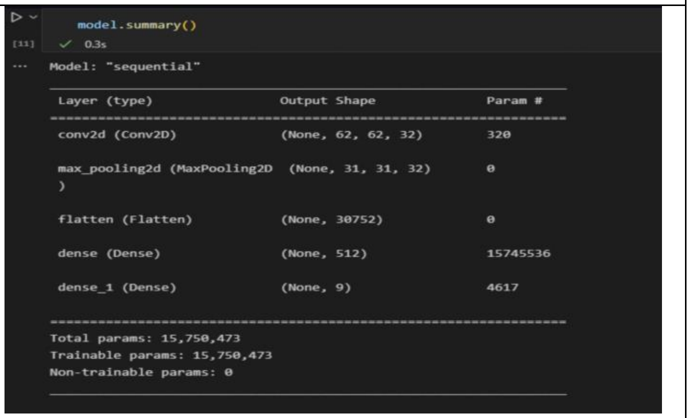
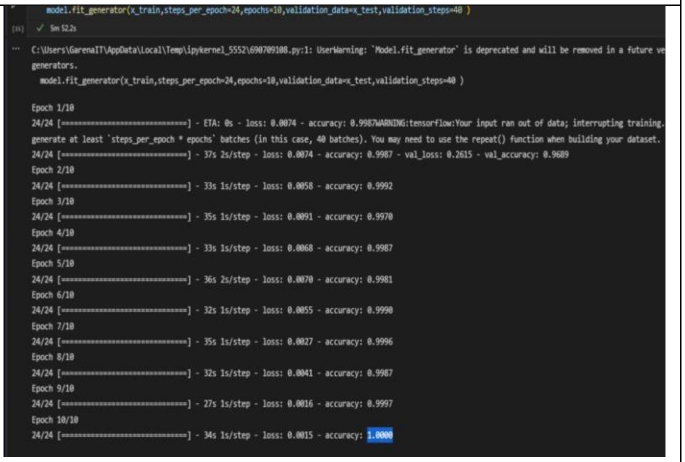
3. Test Case Analysis

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

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	10	0	0	10
Security	3	0	0	3

Figure no:8.2.2 Test Case Analysis

8.3 Performance Testing

S.No	Parameter	Values	Screenshot
1	Model Summary	Total params: 15,750,473 Trainable params: 15,750,473 Non-trainable params: 0	 **Figure no:8.3.1 Model Summary**
2	Accuracy	Training Accuracy Validation Accuracy - 0.9689	 **Figure no:8.3.2 Training Accuracy Validation**

```
  model.summary()
[11]  ✓ 0.3s

... 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
    _________________________________________________________________
```

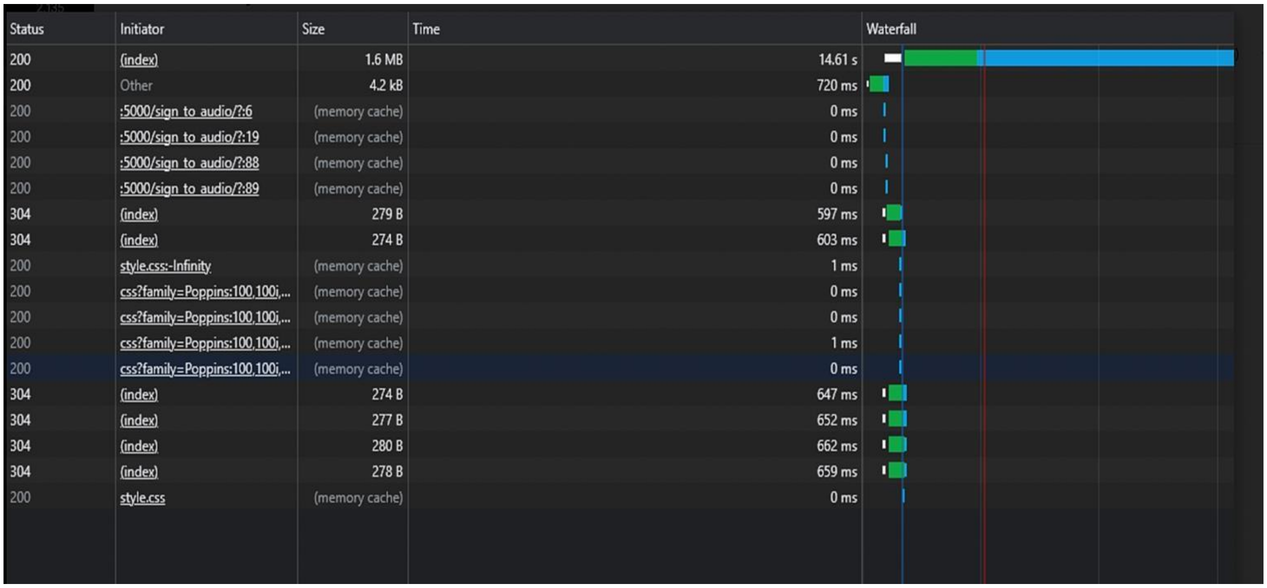
```
  model.fit_generator(x_train,steps_per_epoch=24,epochs=10,validation_data=x_test,validation_steps=40 )
[16]  ✓ 5m 52.2s

... C:\Users\GarenaIT\AppData\Local\Temp\ipykernel_5552\690709108.py:1: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future ve
    generators.
      model.fit_generator(x_train,steps_per_epoch=24,epochs=10,validation_data=x_test,validation_steps=40 )

    Epoch 1/10
    24/24 [==============================] - ETA: 0s - loss: 0.0074 - accuracy: 0.9987WARNING:tensorflow:Your input ran out of data; interrupting training.
    generate at least `steps_per_epoch * epochs` batches (in this case, 40 batches). You may need to use the repeat() function when building your dataset.
    24/24 [==============================] - 37s 2s/step - loss: 0.0074 - accuracy: 0.9987 - val_loss: 0.2615 - val_accuracy: 0.9689
    Epoch 2/10
    24/24 [==============================] - 33s 1s/step - loss: 0.0058 - accuracy: 0.9992
    Epoch 3/10
    24/24 [==============================] - 35s 1s/step - loss: 0.0091 - accuracy: 0.9970
    Epoch 4/10
    24/24 [==============================] - 33s 1s/step - loss: 0.0068 - accuracy: 0.9987
    Epoch 5/10
    24/24 [==============================] - 36s 2s/step - loss: 0.0070 - accuracy: 0.9981
    Epoch 6/10
    24/24 [==============================] - 32s 1s/step - loss: 0.0055 - accuracy: 0.9990
    Epoch 7/10
    24/24 [==============================] - 35s 1s/step - loss: 0.0027 - accuracy: 0.9996
    Epoch 8/10
    24/24 [==============================] - 32s 1s/step - loss: 0.0041 - accuracy: 0.9987
    Epoch 9/10
    24/24 [==============================] - 27s 1s/step - loss: 0.0016 - accuracy: 0.9997
    Epoch 10/10
    24/24 [==============================] - 34s 1s/step - loss: 0.0015 - accuracy: 1.0000
```

9.RESULTS

9.RESULTS

9.1 Performance Metrics

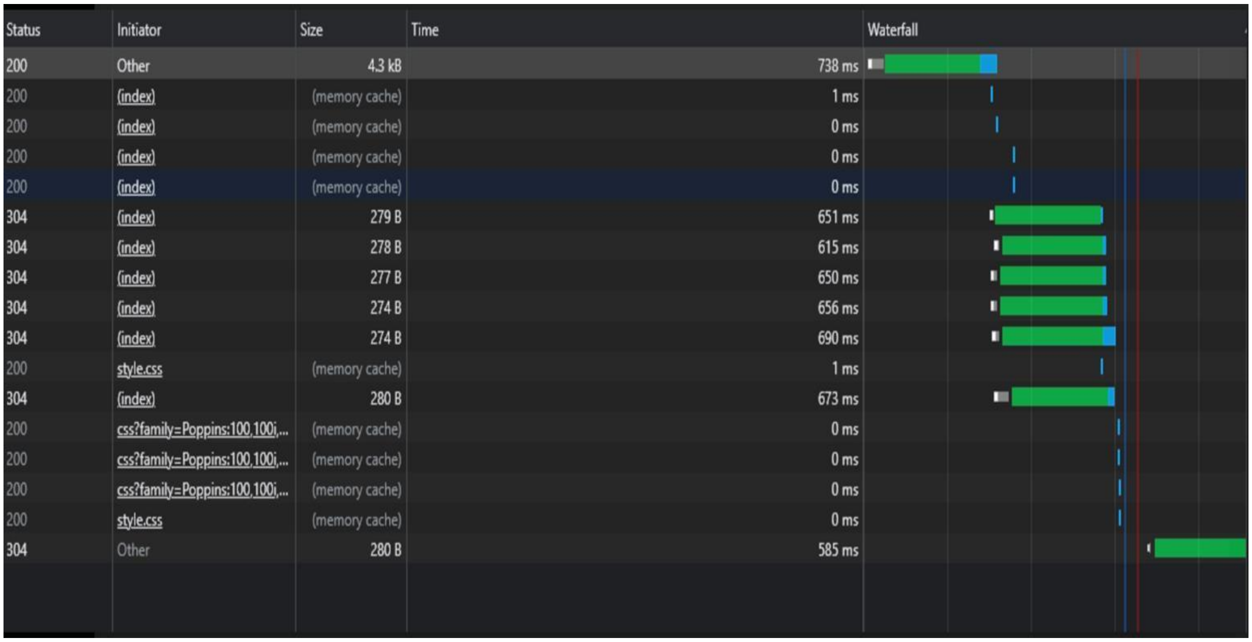
For Sign to speech



Status	Initiator	Size	Time	Waterfall
200	(index)	1.6 MB	14.61 s	
200	Other	4.2 kB	720 ms	
200	:5000/sign to audio/?:6	(memory cache)	0 ms	
200	:5000/sign to audio/?:19	(memory cache)	0 ms	
200	:5000/sign to audio/?:88	(memory cache)	0 ms	
200	:5000/sign to audio/?:89	(memory cache)	0 ms	
304	(index)	279 B	597 ms	
304	(index)	274 B	603 ms	
200	style.css:-Infinity	(memory cache)	1 ms	
200	css?family=Poppins:100,100i,...	(memory cache)	0 ms	
200	css?family=Poppins:100,100i,...	(memory cache)	0 ms	
200	css?family=Poppins:100,100i,...	(memory cache)	1 ms	
200	css?family=Poppins:100,100i,...	(memory cache)	0 ms	
304	(index)	274 B	647 ms	
304	(index)	277 B	652 ms	
304	(index)	280 B	662 ms	
304	(index)	278 B	659 ms	
200	style.css	(memory cache)	0 ms	

Figure no:9.1.1 Performance Metrics Sign to Speech

For Speech to sign



Status	Initiator	Size	Time	Waterfall
200	Other	4.3 kB	738 ms	
200	(index)	(memory cache)	1 ms	
200	(index)	(memory cache)	0 ms	
200	(index)	(memory cache)	0 ms	
200	(index)	(memory cache)	0 ms	
304	(index)	279 B	651 ms	
304	(index)	278 B	615 ms	
304	(index)	277 B	650 ms	
304	(index)	274 B	656 ms	
304	(index)	274 B	690 ms	
200	style.css	(memory cache)	1 ms	
304	(index)	280 B	673 ms	
200	css?family=Poppins:100,100i,...	(memory cache)	0 ms	
200	css?family=Poppins:100,100i,...	(memory cache)	0 ms	
200	css?family=Poppins:100,100i,...	(memory cache)	0 ms	
200	style.css	(memory cache)	0 ms	
304	Other	280 B	585 ms	

Figure no:9.1.2 Performance Metrics Speech to Sign

10. ADVANTAGES & DISADVANTAGES

10.ADVANTAGES & DISADVANTAGES

Advantages:

1. Real time sign to speech detection.
2. Model provides good accuracy.
3. Real time facial emotion detection.
4. Language Customization.
5. Real time speech to text conversion.
6. Friendly UI
7. Data privacy

Disadvantages:

1. At times the website may lag.
2. Model is not tested on a wide set of data set, having all the signs.
3. Sign language customization feature is not available.
4. User cannot take notes while using the app.
5. User cannot make calls using the app.
6. Speech recognition works only on google chrome.

11.CONCLUSION

11.CONCLUSION

The ability to express oneself requires communication. Additionally, it satisfies one's needs. Career advancement requires effective communication. By promoting mutual understanding, good communication skills can simplify your daily life and enhance your connections with others. As part of our effort, a system that converts speech into suitable sign language for the deaf and dumb has been created. In order to converse with regular people, it also transforms sign language into a human voice. A model that has been trained on several hand motions has been created using a convolutional neural network. An app is produced using this idea. This software makes it simple for deaf and dumb people to communicate by using signs that can be translated into speech and human-understandable English.

12.FUTURE SCOPE

12.FUTURE SCOPE

The following are the features that can be added in our application:

- A communication app can be built with the same set of features. The user can choose the appropriate mode (speech to sign or sign to speech) and accordingly the real time detection would take place on both the end users' application.
- The accuracy of the model shall be increased.
- Customization of languages shall be added.
- Users shall be allowed to write notes while on call.
- Customization of signs can also be added as a feature.
- Provide the device consists of gloves, flex sensors, microcontroller and 16X16 LED display, which converts the hand gestures into real time speech.

13.APPENDIX

13.APPENDIX

13.1 Source Code

Index.html

```
<div class="section full-height" >
    <div class="absolute-center">
      <div class="section">
        <div class="container">
          <div class="row">
            <div class="col-12">

<h1><span>W</span><span>e</span><span>l</span><span>c</span><span>o</span><span
>m</span><span>e</span></h1>
          <br>

          <form action="/sign_to_audio/"><button class="btn fifth">Sign To
Speech</button></form>
          <form action="/audio_to_sign/"><button class="btn fifth">Speech to
Sign</button></form>

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

    </div>
```

app.py

```
from flask import Flask, render_template,redirect,url_for,request,Response
from moviepy.editor import VideoFileClip
from moviepy.video.io.VideoFileClip import VideoFileClip
import cv2
import speech_recognition as sr
from PIL import Image
import numpy as np
from skimage.transform import resize
from gtts import gTTS
from playsound import playsound
from keras.utils import image_utils as image
from keras.models import load_model
import h5py
import pandas as pd
import tensorflow as tf
from tensorflow import keras
app=Flask(__name__)

vals=['A','B','C','D','E','F','G','H','I']

model=tf.keras.models.load_model('C:\\Users\\Harini\\aslpng1.h5')

app.secret_key = "secret key"
arr = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l',
        'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z','.']

@app.route('/')
def index():
    return render_template('index.html')
@app.route('/audio_to_sign/')
def audio_to_sign():
    return render_template('SpeechtoSign.html')

@app.route('/audio', methods=['POST'])
def audio():
    r = sr.Recognizer()

    frameSize = (281, 363)
    out = cv2.VideoWriter('./static/uploads/output_video.mp4',cv2.VideoWriter_fourcc(*'DIVX'), 1, frameSize)
    with open('D:\\IBM PROJECT CODING\\dist\\Speech\\audio.mp3', 'wb') as f:
        f.write(request.data)
```

```
  with sr.AudioFile('D:\\IBM PROJECT CODING\\dist\\Speech\\audio.mp3') as source:
        audio_data = r.record(source)
        text = r.recognize_google(audio_data, language='en-IN', show_all=True)
        print(text)

        try:
            for num, texts in enumerate(text['alternative']):
                transcript = texts['transcript'].lower()
                print(transcript)

                break

        except:
            transcript = " Sorry!!!! Voice not Detected "

    for i in range(len(transcript)):
        if transcript[i] in arr:



            ImageAddress = 'L/'+transcript[i]+'.png'
            ImageItself = Image.open(ImageAddress)
            ImageNumpyFormat = np.asarray(ImageItself)
            img = cv2.imread(ImageAddress)
            out.write(img)

    out.release()
    #videoFileClip=VideoFileClip("./static/uploads/output_video.mp4")
    videoFileClip.write_gif("./static/uploads/output_video.gif")
    #videoFileClip.write_gif("./static/uploads/output_video1.gif")


    return str(transcript)

@app.route('/scrn', methods=['POST'])
def upload_video():
    r=sr.Recognizer()
    file=sr.AudioFile("D:\\IBM PROJECT CODING\\dist\\Speech\\audio.mp3")
    with file as source:
        audio_data = r.record(source)
    text = r.recognize_google(audio_data, language='en-IN', show_all=True)
    text=text['alternative']
    text=text[0]
    text=text['transcript']
    return render_template('stream.html', filename='output_video.mp4',text=text)
```

```
@app.route('/display/<filename>')
def display_video(filename):

    return redirect(url_for('static', filename='uploads/' + filename), code=301)

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

def gen():
    string = " "
    count = 90
    video = cv2.VideoCapture(0)
    while (video.isOpened()):
        ret, frame = video.read()
        gray=cv2.cvtColor(frame,cv2.COLOR_BGR2GRAY)
        gray = cv2.threshold(gray, 0, 255,cv2.THRESH_BINARY_INV | cv2.THRESH_OTSU)[1]
        color_dict=(0,255,0)
        cv2.rectangle(frame,(24,24),(250 , 250),color_dict,2)
        copy=gray.copy()
        copy = copy[24:250,24:250]
        count = count + 1
        cv2.imwrite('D:\IBM PROJECT CODING\dist\static\image.png',copy)
        img = cv2.imread('D:\IBM PROJECT CODING\dist\static\image.png')
        img=resize(img,(64,64,1))
        img=image.img_to_array(img)
        img=np.expand_dims(img,axis=0)
        if(np.max(img)>1):
            img=img/255.0
        prediction=model.predict(img)
        prediction=np.argmax(prediction, axis=1)
        pred=vals[prediction[0]]
        print(pred)
        if(count == 200):
            count = 99
            prev= vals[prediction[0]]
            if(len(prev) == 0):
                string = string + "_"
                myobj = gTTS(text=string, lang="en", slow=False)
                myobj.save("D:\IBM PROJECT CODING\dist\Speech\sign.mp3")

            else:
                string = string + prev
                myobj = gTTS(text=string, lang="en", slow=False)
                myobj.save("D:\IBM PROJECT CODING\dist\Speech\sign.mp3")
        cv2.putText(frame, pred, (24, 14),cv2.FONT_HERSHEY_SIMPLEX,0.8,(255,255,255),2)
        cv2.putText(frame, string, (275, 50),cv2.FONT_HERSHEY_SIMPLEX,0.8,(200,200,200),2)
        if not ret:
            break
        else:
            ret,buffer=cv2.imencode('.jpg',frame)
            frame=buffer.tobytes()
        yield (b'--frame\r\n'
                    b'Content-Type: image/jpeg\r\n\r\n' + frame + b'\r\n\r\n')

@app.route('/video_feed')
def video_feed():
     return Response(gen(),
                    mimetype='multipart/x-mixed-replace; boundary=frame')

@app.route('/redirect')
def delet():
    video = cv2.VideoCapture(0)
    video.release()
    return render_template("index.html")

@app.route('/play')
def play():
    playsound('D:\IBM PROJECT CODING\dist\Speech\sign.mp3', True)
    return render_template("SigntoSpeech.html")

@app.route('/alp')
def alp():

    return render_template("Alphabet.html")

if __name__ == "__main__":
    app.run(debug=True)
```

Speech to Sign.html

```
<!DOCTYPE html>
<html lang="en" >
<head>
  <meta charset="UTF-8">
  <title>Communication System for Specially abled People</title>
  <link rel='stylesheet' href='https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.1.3/css/bootstrap.min.css'><link rel="stylesheet" href="../static/style.css">
  <script src="/static/recorder.js"></script>
  <script src="/static/audiodisplay.js"></script>
  <script src="/static/main.js"></script>
  <script src="/static/recorderWorker.js"></script>
  <script src="../static/script.js"></script>
</head>
<body>
<!-- partial:index.partial.html -->
<body class="hero-anime">

    <div class="navigation-wrap bg-light start-header start-style">
        <div class="container">
            <div class="row">
                <div class="col-12">
                    <nav class="navbar navbar-expand-md navbar-light">

                        <a class="navbar-brand" href="/" ><img src="https://user-images.githubusercontent.com/115576572/196229016-e13b9825-dc49-4dcc-b76f-ced72557d9f5.png" alt=""></a>

                        <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
                            <span class="navbar-toggler-icon"></span>
                        </button>

                        <div class="collapse navbar-collapse" id="navbarSupportedContent">
                            <ul class="navbar-nav ml-auto py-4 py-md-0">
                                <li class="nav-item pl-4 pl-md-0 ml-0 ml-md-4 active">
                                    <a class="nav-link dropdown-toggle" data-toggle="dropdown" href="/" role="button" aria-haspopup="true" aria-expanded="false">Home</a>
                                    <div class="dropdown-menu">

                                        <a class="dropdown-item" href="/sign_to_audio/">Sign To Speech</a>

                                    </div>
                                </li>
```

```
<li class="nav-item pl-4 pl-md-0 ml-0 ml-md-4 active">
                                <a class="nav-link dropdown-toggle" data-toggle="dropdown" href="#" role="button" aria-
haspopup="true" aria-expanded="false">Contributors</a>
                                <div class="dropdown-menu">
                                    <a class="dropdown-item" href="#">Indirakumar</a>
                                    <a class="dropdown-item" href="#" >Richard</a>
                                    <a class="dropdown-item" href="#">Rahul Fernandez</a>
                                    <a class="dropdown-item" href="#">Srinivasan</a></div> </li>
</ul>
                        </div>  </nav>  </div></div></div> </div>
    <div class="section full-height">
        <div class="absolute-center">
            <div class="section">
                <div class="container">
                    <div class="row">
                        <div class="col-12">
                        <button id="start" class="btn btn-success" onclick="startRecording()" disabled>Start</button>
                        <button id="stop" class="btn btn-danger" onclick="stopRecording()" disabled>Stop</button>
                        </div>
                        <div class="column2" >
                            <h1 style="padding-left: 60px;"><span>Audio Prediction</span> </h1>
                            <p class="glow" id="output" style="font-size: 20px;"></p>
                            <form action="/scrn" method="post" enctype="multipart/form-data">
                              <input type="submit" id="stream" value="Stream" class="btn btn-info stream" disabled>
                           </form><a href="/">
                                <button class="btn btn-danger btn-lg" >Back</button>
                            </a></div>
                        <div class="note">
                            <b>Note:</b>  Please wait a second for the audio prediction; if it is a statisfactable output, then click
"Stream." Thank you
                        </div></div </div> </div>
          </div></div>
    <div class="my-5 py-5">
    </div>
 <!-- Link to page
================================================ -->
</body>
<!-- partial -->
  <script src='https://cdnjs.cloudflare.com/ajax/libs/jquery/3.3.1/jquery.min.js'></script>
<script src='https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js'></script>
<script  src="../static/script.js"></script>
</body>
</html>
```

STREAM.HTML

```
<div class="section">
  <div class="container">
    <div class="row">
      <div class="col-12">
        <img src="{{ url_for('display_video', filename=filename) }}">
        <p>{{text}}</p>
      </div>
      <form action="/audio_to_sign/">
        <button class="btn fifth back">Back</button>
      </form>
    </div>
  </div>

</div>
```

SigntoSpeech.html

```
<!DOCTYPE html>
<html lang="en" >
<head>
  <meta charset="UTF-8">
  <title>Communication System for Specially abled People</title>
  <link rel='stylesheet' href='https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.1.3/css/bootstrap.min.css'><link rel="stylesheet" href="../static/style.css">
  <script src="/static/recorder.js"></script>
  <script src="/static/audiodisplay.js"></script>
  <script src="/static/main.js"></script>
  <script src="/static/recorderWorker.js"></script>
  <script src="../static/script.js"></script>
</head>
<body>
<!-- partial:index.partial.html -->
<body class="hero-anime">

    <div class="navigation-wrap bg-light start-header start-style">
        <div class="container">
            <div class="row">
                <div class="col-12">
                    <nav class="navbar navbar-expand-md navbar-light">

                        <a class="navbar-brand" href="/" ><img src="https://user-images.githubusercontent.com/115576572/196229016-e13b9825-dc49-4dcc-b76f-ced72557d9f5.png" alt=""></a>
                        <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
                            <span class="navbar-toggler-icon"></span>
                        </button>

                        <div class="collapse navbar-collapse" id="navbarSupportedContent">
                            <ul class="navbar-nav ml-auto py-4 py-md-0">
                                <li class="nav-item pl-4 pl-md-0 ml-0 ml-md-4 active">
                                    <a class="nav-link dropdown-toggle" data-toggle="dropdown" href="/" role="button" aria-haspopup="true" aria-expanded="false">Home</a>
                                    <div class="dropdown-menu">
                                        <a class="dropdown-item" href="/audio_to_sign/">Speech TO Sign</a>

                                    </div>
                                </li>
                                <li class="nav-item pl-4 pl-md-0 ml-0 ml-md-4 active">
                                    <a class="nav-link dropdown-toggle" data-toggle="dropdown" href="#" role="button" aria-haspopup="true" aria-expanded="false">Contributors</a>
 <div class="dropdown-menu">
                                        <a class="dropdown-item" href="#">HARINI</a>
                                        <a class="dropdown-item" href="#">AARTHI</a>
                                        <a class="dropdown-item" href="#" >ARUNA</a>
                                        <a class="dropdown-item" href="#">HARSHAVARDHINI</a>

    </div>
                                </li>

                            </ul>
                        </div>
                </nav>
                </div>
            </div>
        </div>
    </div>
    <div class="section full-height">
        <div class="absolute-center">
            <div class="section">
                <div class="container">
                    <div class="row">

                        <div class="column2" >
                            <h1 style="margin-top: 10%;">Live </h1>
                            <br> <div class="stream-video">
                                <img src="{{ url_for('video_feed') }}" width="576px" height="324px"/>
                                <br>
                <a href="/play">
                  <button class="btn btn-info btn-lg" id="play">Play & Clear</button>
                </a>
                <a href="/redirect">
                  <button class="btn btn-danger btn-lg" >Exit</button>
                </a>
              </div>
              <div>
                <b>Note:</b> According to the given dataset, the model is designed for the alphabet A–I, so it predicts only limited alphabets And press the "Play & Clear" button after the predicted letter displays on the right side of the green rectangle. Thank you
              </div>

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



    </div>
  </div>
  <div class="my-5 py-5">
  </div>

<!-- Link to page
======================================================== -->



</body>
<!-- partial -->
 <script src='https://cdnjs.cloudflare.com/ajax/libs/jquery/3.3.1/jquery.min.js'></script>
<script src='https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js'></script>
<script  src="../static/script.js"></script>
<script>
document.getElementById("play").disabled = true;
setTimeout(function(){document.getElementById("play").disabled = false;},20000);
</script>
</body>
</html>
```

Alphabet.html

```
<!DOCTYPE html>
<html lang="en" >
<head>
  <meta charset="UTF-8">
  <title>Communication System for Specially abled People</title>
  <link rel='stylesheet' href='https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.1.3/css/bootstrap.min.css'><link
rel="stylesheet" href="../static/style.css">

</head>
<body>
<!-- partial:index.partial.html -->
<body class="hero-anime">

    <div class="navigation-wrap bg-light start-header start-style">
        <div class="container">
            <div class="row">
                <div class="col-12">
                    <nav class="navbar navbar-expand-md navbar-light">

                        <a class="navbar-brand" href="/" ><img src="https://user-
images.githubusercontent.com/115576572/196229016-e13b9825-dc49-4dcc-b76f-ced72557d9f5.png" alt=""></a>

                        <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle
navigation">
                            <span class="navbar-toggler-icon"></span>
                        </button>

                        <div class="collapse navbar-collapse" id="navbarSupportedContent">
                            <ul class="navbar-nav ml-auto py-4 py-md-0">
                                <li class="nav-item pl-4 pl-md-0 ml-0 ml-md-4 active">
                                    <a class="nav-link dropdown-toggle" data-toggle="dropdown" href="/" role="button" aria-
haspopup="true" aria-expanded="false">Home</a>
                                    <div class="dropdown-menu">

                                        <a class="dropdown-item" href="https://giphy.com/signwithrobert/" target="_blank">Learn Sign
Language(GIF)</a>
                                        <a class="dropdown-item"
href="https://www.youtube.com/playlist?list=PLMN7QCuj6dfYD8DfG1rN6rEo1b1RyvgKF" target="_blank">Learn Sign
From Youtube</a>

                                        </div>
                                </li><li class="nav-item pl-4 pl-md-0 ml-0 ml-md-4 active">
```

```
<a class="nav-link dropdown-toggle" data-toggle="dropdown" href="#" role="button" aria-haspopup="true" aria-expanded="false">Contributors</a>
                                    <div class="dropdown-menu">
                                        <a class="dropdown-item" href="#">HARINI</a>
                                        <a class="dropdown-item" href="#" >ARUNA</a>
                                        <a class="dropdown-item" href="#">AARTHI</a>
                                        <a class="dropdown-item" href="#">HARSHAVARDHINI</a>

                                    </div>
                                </li>

                            </ul>
                        </div>

                    </nav>
                </div>
            </div>
        </div>
    </div>
    <div class="section full-height" >
        <div class="absolute-center">
            <div class="section">
                <div class="container">
                    <div class="row">
                        <div class="col-12">
                <h1><span>A</span><span>L</span><span>P</span><span>H</span><span>A</span><span>B</span><span>E</span><span>T</span><span>S</span></h1>
                    <br>

     <img src="https://i.pinimg.com/564x/98/2c/d1/982cd1fb0bac83d6b12c9a6acdff0879.jpg" style="text-align: left;">
                        </div>
                    </div>
                </div>
            </div> </div>
    </div>
    <div class="my-5 py-5">
    </div></body>
<!-- partial -->
  <script src='https://cdnjs.cloudflare.com/ajax/libs/jquery/3.3.1/jquery.min.js'></script>
<script src='https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js'></script>
<script  src="../static/script.js"></script>

</body>
</html>
```

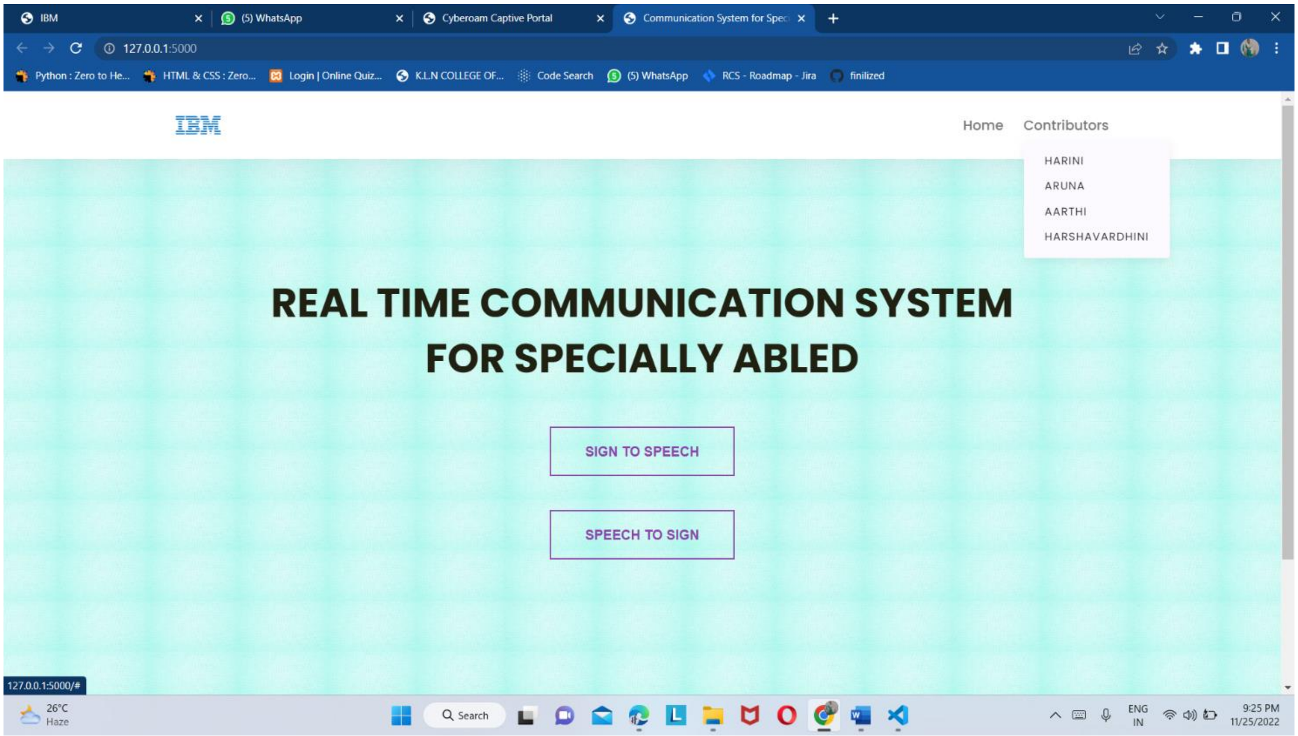


Figure no:13.1.1 Index Page-Contributors

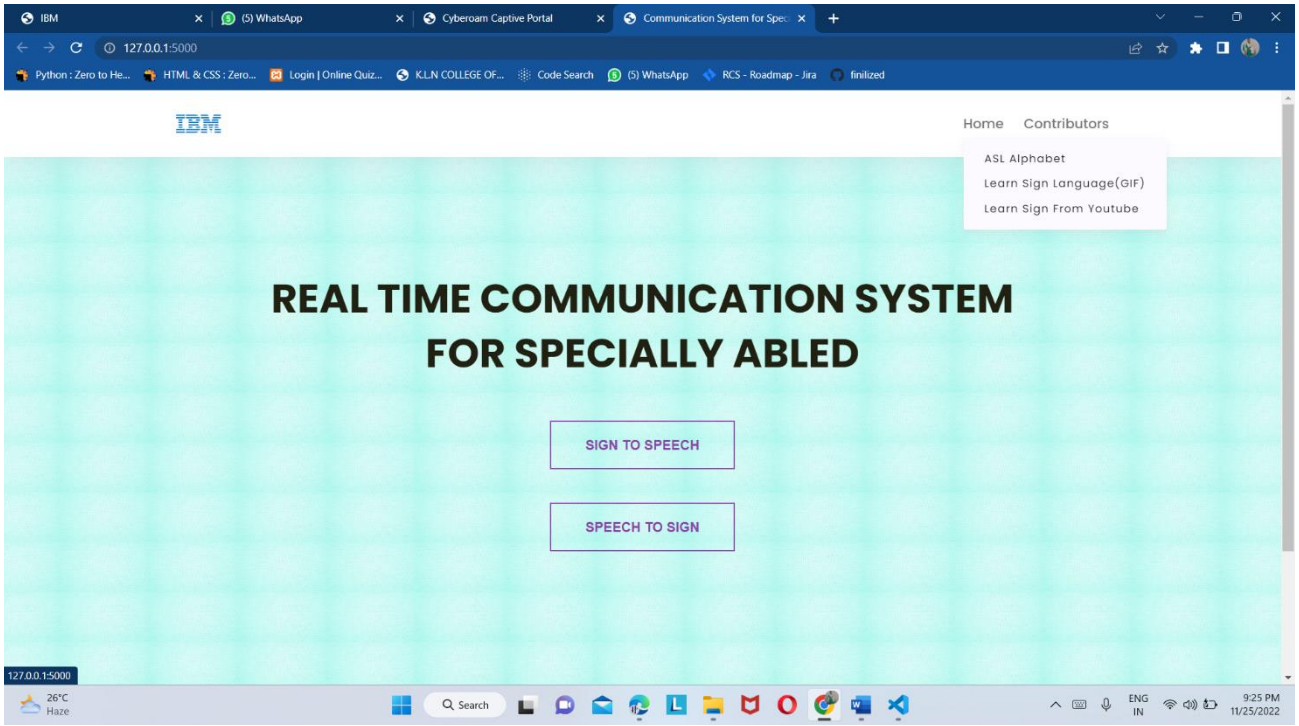


Figure no:13.1.2 Index Page

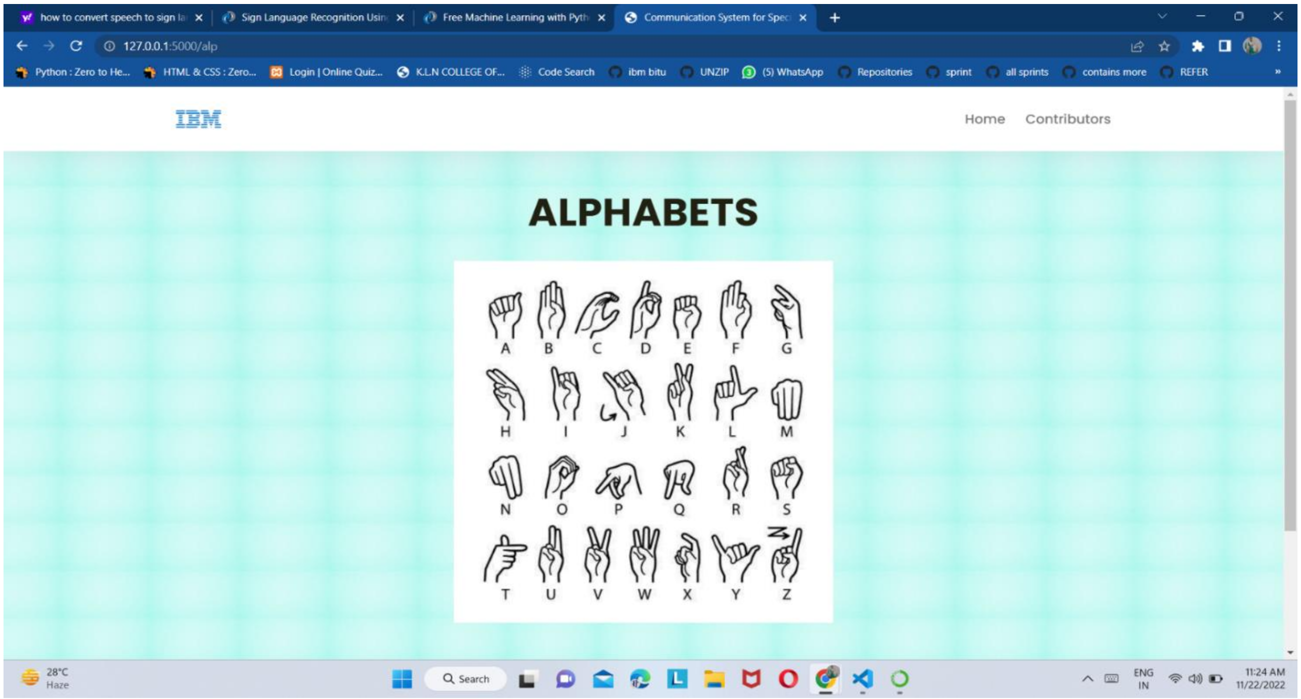
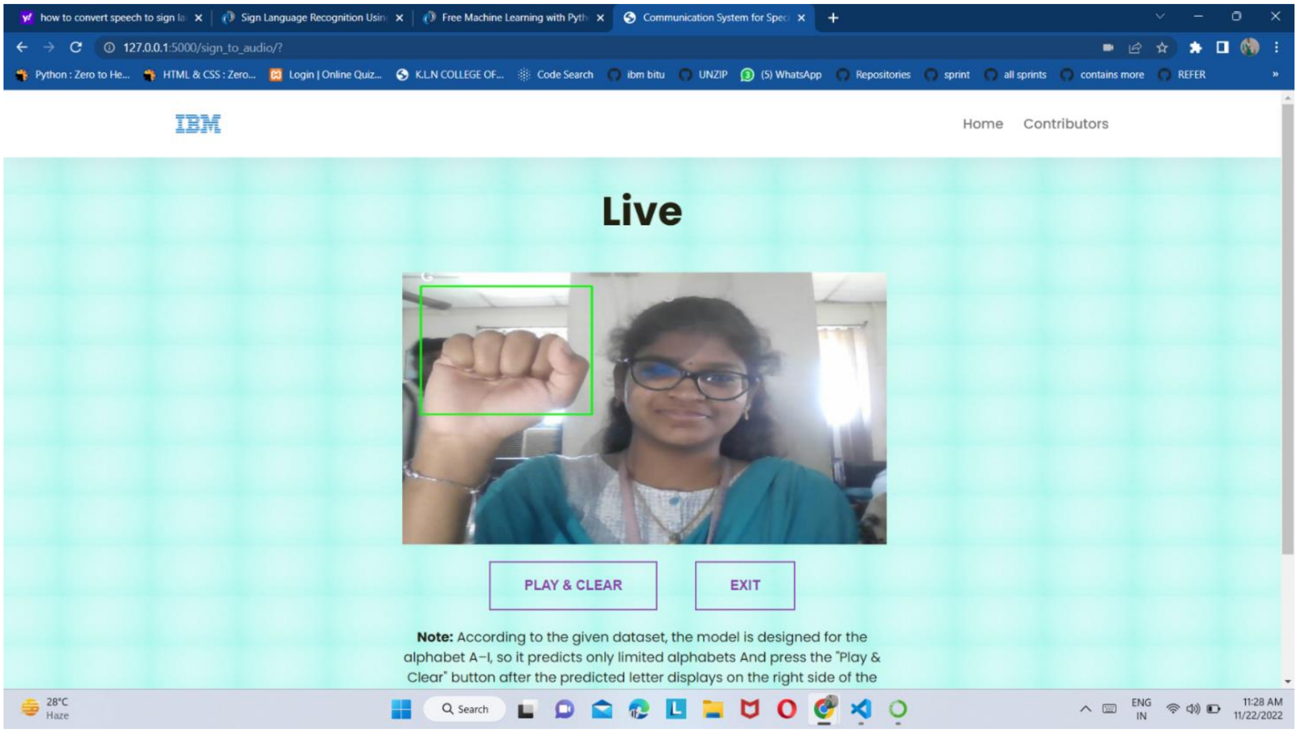
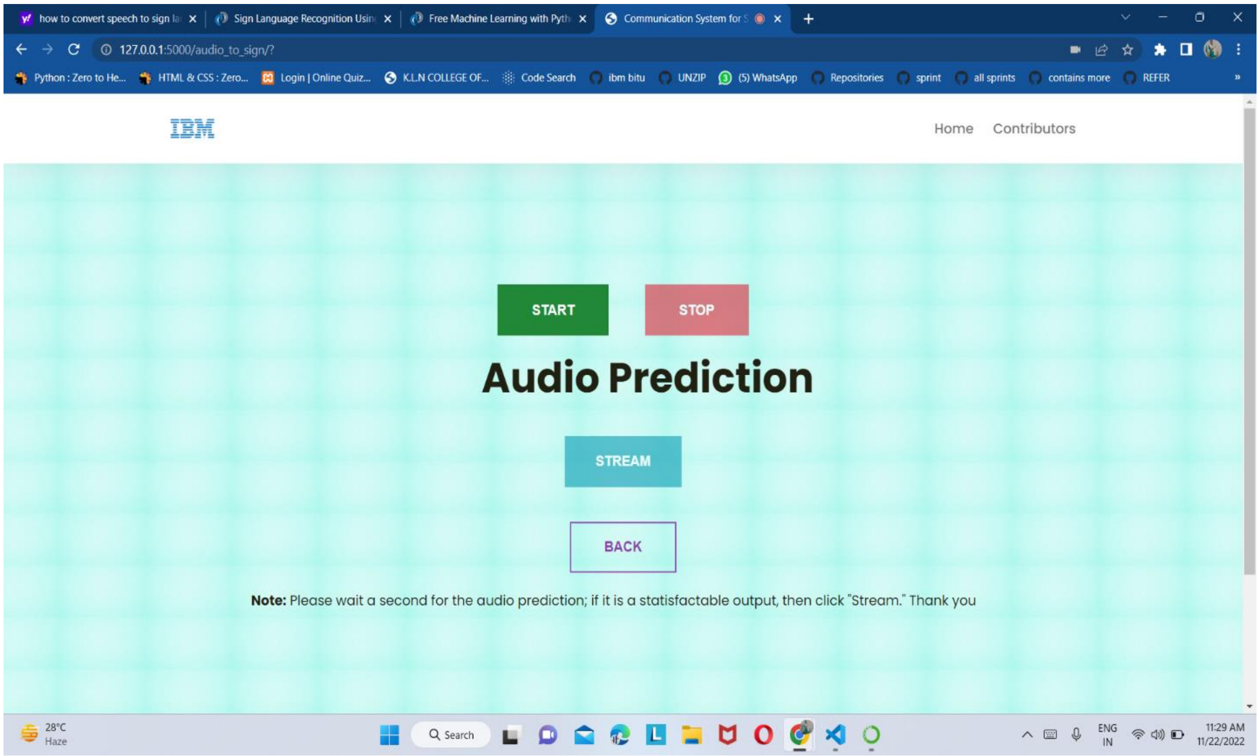


Figure no:13.1.3 Alphabets



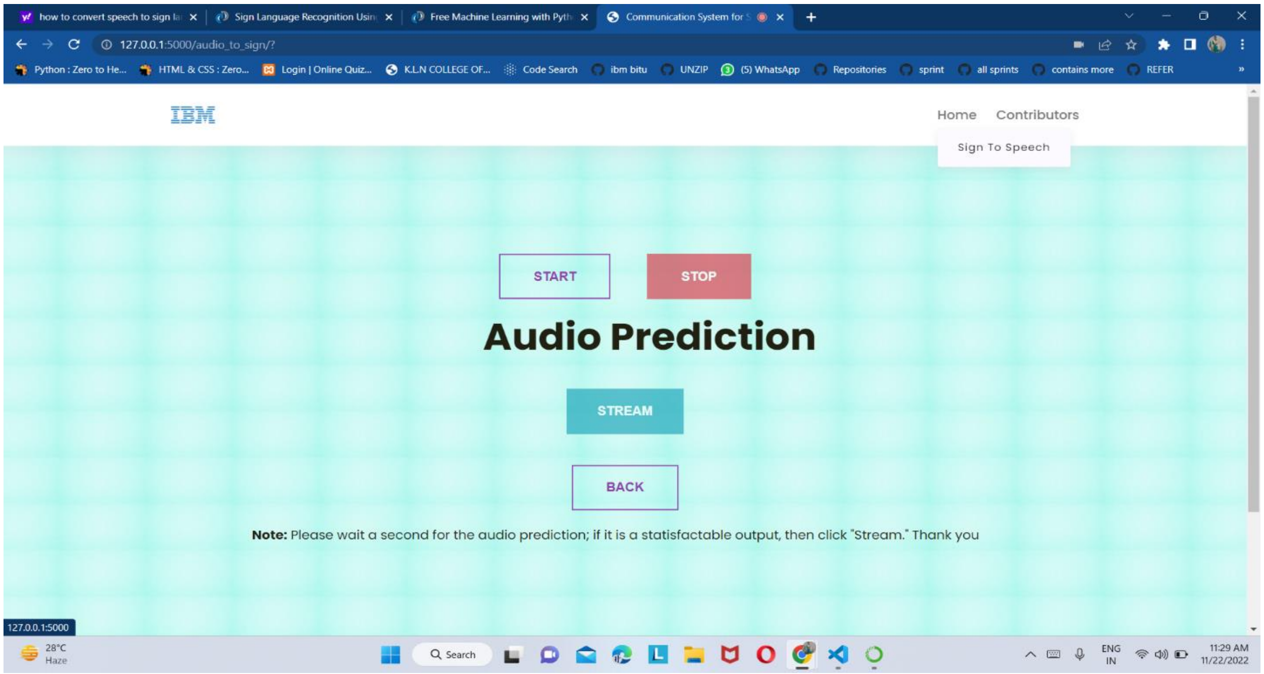
Note: According to the given dataset, the model is designed for the alphabet A–I, so it predicts only limited alphabets And press the "Play & Clear" button after the predicted letter displays on the right side of the

Figure no:13.1.4 Sign to Speech


how to convert speech to sign la
Sign Language Recognition Usin
Free Machine Learning with Pyth
Communication System for S
127.0.0.1:5000/audio_to_sign/?
IBM
Home
Contributors
START
STOP
Audio Prediction
STREAM
BACK
28°C
Haze
Search
11:29 AM
11/22/2022

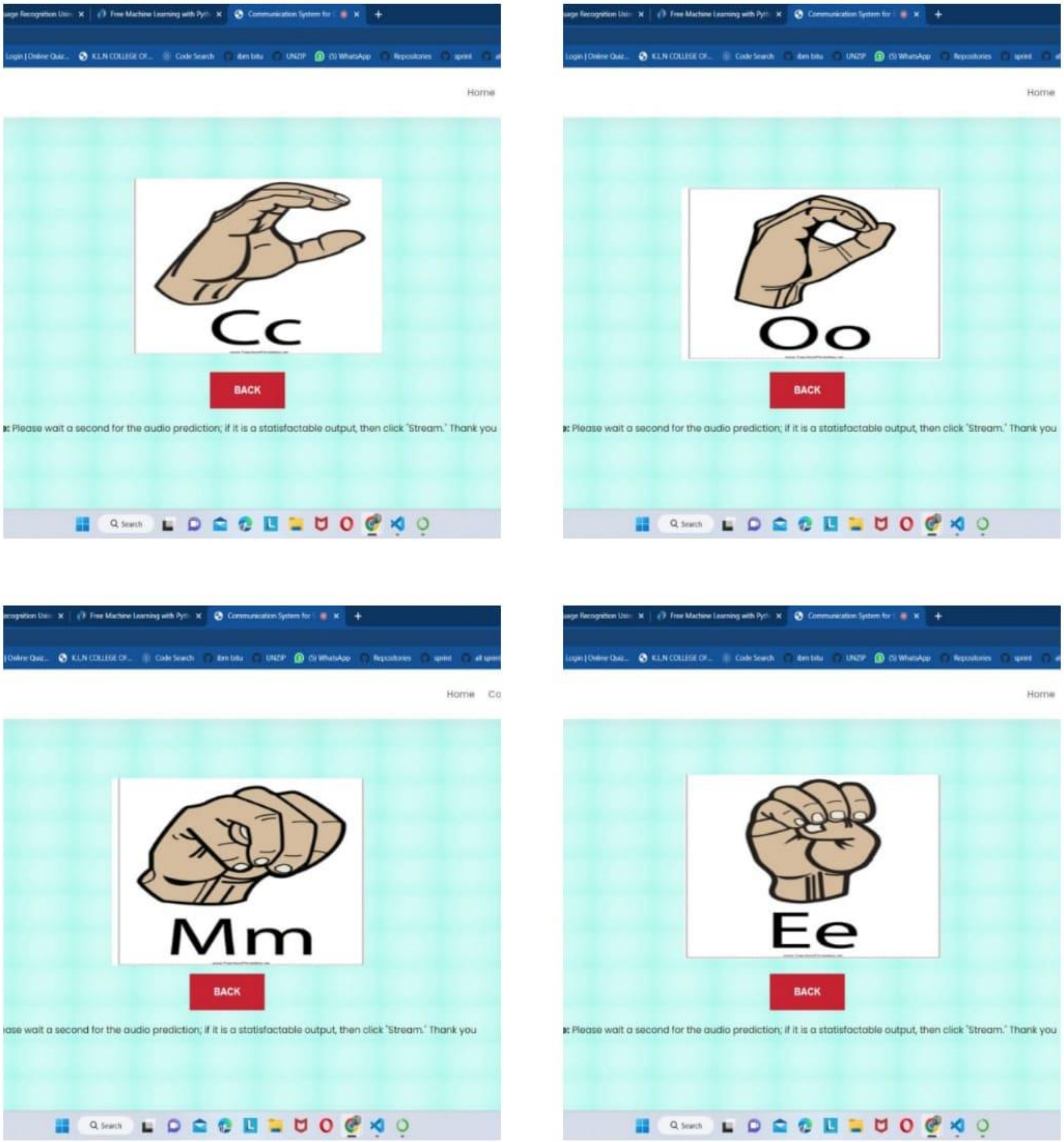
Note: Please wait a second for the audio prediction; if it is a statisfactable output, then click "Stream." Thank you

Figure no:13.1.5 Speech to Sign


how to convert speech to sign la
Sign Language Recognition Usin
Free Machine Learning with Pyth
Communication System for S
127.0.0.1:5000/audio_to_sign/?
IBM
Home
Contributors
Sign To Speech
START
STOP
Audio Prediction
STREAM
BACK
127.0.0.1:5000
28°C
Haze
Search
11:29 AM
11/22/2022

Note: Please wait a second for the audio prediction; if it is a statisfactable output, then click "Stream." Thank you

Figure no:13.1.6 Audio Prediction



: Please wait a second for the audio prediction; if it is a statisfactable output, then click 'Stream.' Thank you

: Please wait a second for the audio prediction; if it is a statisfactable output, then click 'Stream.' Thank you

ase wait a second for the audio prediction; if it is a statisfactable output, then click 'Stream.' Thank you

: Please wait a second for the audio prediction; if it is a statisfactable output, then click 'Stream.' Thank you

Figure no:13.1.7 Output

13.2 GitHub & Project Demo Link

GitHub -https://github.com/IBM-EPBL/IBM-Project-38050-1660369180

Project Demo Link -https://youtu.be/QlVAkrZ32xQ