PROJECT REPORT

TEAM ID	**PNT2022TMID50812**
PROJECT NAME	**REAL-TIME COMMUNICATION SYATEM POWERED BY AI FOR SPECIALLY ABLED**
TEAM LEADER	**SHUNMUGA PRIYA K R (9532106034)**
TEAM MEMBER 1	**AJITHA K (953219106003)**
TEAM MEMBER 2	**ISHWARYA K (953219106012)**
TEAM MEMBER 3	**SANTHANA SWATHI A (953219106029)**

1. INTRODUCTION

1.1.PROJECT OVERVIEW

The project developed is a system that converts hand gestures of a Deaf-Mute individual into its respective ASL (American Sign Language) alphabets for a normal individual for communication. The main customer for our project are: People who want to communicate with deaf-mute individual who desire to communicate with others, and deaf-mute individual who desire to communicate with others. This project tries to solve the communication during the time of emergencies. The project is developed on Python

Platform using CNN (Convolutional Neural Network) model from Tensor Flow package.

1.2. PURPOSE

Everybody cannot afford to have a human translators of sign language, they may not be available all the time and they are quite expensive. People who engage in conversation with deaf-mute individual will find it hard and tedious. Deaf-mute individual may lose a lot of opportunities because they cannot speak or express their thoughts verbally in situations like an interview. This project aims to overcome the said challenges.

2. LITERATURE SURVEY

1. TITLE: Sign Language Recognition System for people with disabilities using AI

AUTHORS: Dalia Nashat1, Abeer Shoker1, Fowzyah Al-Swat2 and Reem AlEbailan2

YEAR: 2014

DESCRIPTION :
One of the most popular handicaps is the deaf and dumb type, which prevent person from listening and talking. The number of deaf and dumb in the world continuously increasing and they are introverted closed society. Therefore, Deaf-Dumb people do not have normal opportunities for learning. Uneducated Deaf-Dumb people face serious problem in communication with normal people in their society. It is notable, however, that most available application focus only on learning or recognition of sign language. In this paper, we introduce an integrated android application to blend uneducated Deaf-Dumb people within society, and help them to communicate with normal people. The introduced application proposes an easy translator in keyboard form that can translate any word from sign language to Arabic or English language and vice versa. This application also contains most daily words for teaching deaf and dumb kids in attractive

way (colours, pictures, animations, quiz …etc). Moreover, it introduces some games that help them to communicate and entertain.

2. TITLE : Artificial Intelligence and Communicatin Technology Accessibility

AUTHORS : Majzoob Kamal Aldein Omer, Mohmed Sirelkhtem Adrees, Osama E. Sheta

YEAR : 2015

DESCRIPTION:
The study aims to apply the strategy to help deaf students and dumb in academic achievement by using mobile learning technology application. This sample of the students have a high potential for the use of mobile applications and has a capacity of great learning via mobile. Smart mobile phones have the ability to create a good educational content of images, shapes, graphics and illustrations appropriate signs to the Deaf and Dumb students and the production of educational content suitable for individual differences in education between them and meets their needs mental and their interests that are different from ordinary students in Education. The paper focuses on the educational content of the component images, graphs and illustrations appropriate signs to the Deaf and Dumb students because it is not easy to understand by a normal listener on the opposite and to make things worse. Infact the technology is used to achieve the interaction between deaf and dump children with others.

3.TITLE : Artificial Intelligence Enabled Virtual Sixth Sense Application For the Disabled

AUTHOR : Muhammed Usman Tariq

YEAR : 2020

DESCRIPTION :
The main purpose of this research is to enhance the communication of the disabled community. The authors of this chapter propose an enhanced interpersonal-human interaction for people with special needs, especially those with physical and communication disabilities. The proposed model comprises of automated real time behaviour monitoring, designed and implemented with the ubiquitous and affordable concept in mind to suit the underprivileged. In this chapter, the authors present the prototype which encapsulates an automated facial expression recognition system for monitoring the disabled, equipped with a feature to send Short Messaging System (SMS) for notification purposes. The authors adapted the Viola-Jones face detection algorithm at the face detection stage and implemented template matching technique for the expression classification and recognition stage. They tested their model with a few users and achieved satisfactory results. The enhanced real time behaviour monitoring system is an assistive tool to improve the quality of life for the disabled by assisting them anytime and anywhere when needed. They can do their own tasks more independently without constantly being monitored physically or accompanied by their care takers, teachers, or even parents. The rest of this chapter is organized as follows. The background of the facial expression recognition system is reviewed in Section 2. Section 3 is the description and explanations of the conceptual model of facial expression recognition. Evaluation of the proposed system is in Section 4. Results and findings on the testing are laid out in Section 5, and the final section concludes the chapter.

4.TITLE : Artificial Intelligence Enhances Accessibility for people with disabilities

AUTHORS : Bayan Mohammed Saleh1, Reem Ibrahim Al-Beshr2,
Muhammad Usman Tariq3

YEAR : 2020

DESCRIPTION:
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 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 also 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.

5. TITLE : A Face based Real Time Communication for speech disabled people

AUTHORS : Aditya Sharma 1, Aditya Vats 2 , Shiv Shankar Dash 3 and
Surinder Kaur

YEAR : 2020

DESCRIPTION :
The sixth sense is a multi-platform app for aiding the people in need that is people who are handicapped in the form of lack of speech (dumb), lack of hearing (deaf), lack of sight (blind), lack of judicial power to differentiate between objects (visual agnosia) and people suffering from autism (characterized by great difficulty in communicating and

forming relationships with other people and in using language and abstract concepts). Our current implementation of the product is on two platforms, namely, mobile and a web app. The mobile app even works for object detection cases in offline mode. What we want to achieve using this is to make a better world for the people suffering from disabilities as well as an educational end for people with cognitive disabilities using our app. The current implementation deals with object recognition and text to speech and a speech to text converter. The speech to text converter and text to speech converter utilized the Web Speech API (Application Program Interface) for the website and text to speech and speech to text library for the mobile platform. The object recognition wouldn't fetch enough use out of a website. Hence, it has been implemented on the mobile app utilizing the Firebase ML toolkit and different pre-trained models, which are both available offline as well as online.

3. STATE STATEMENT DEFINITION

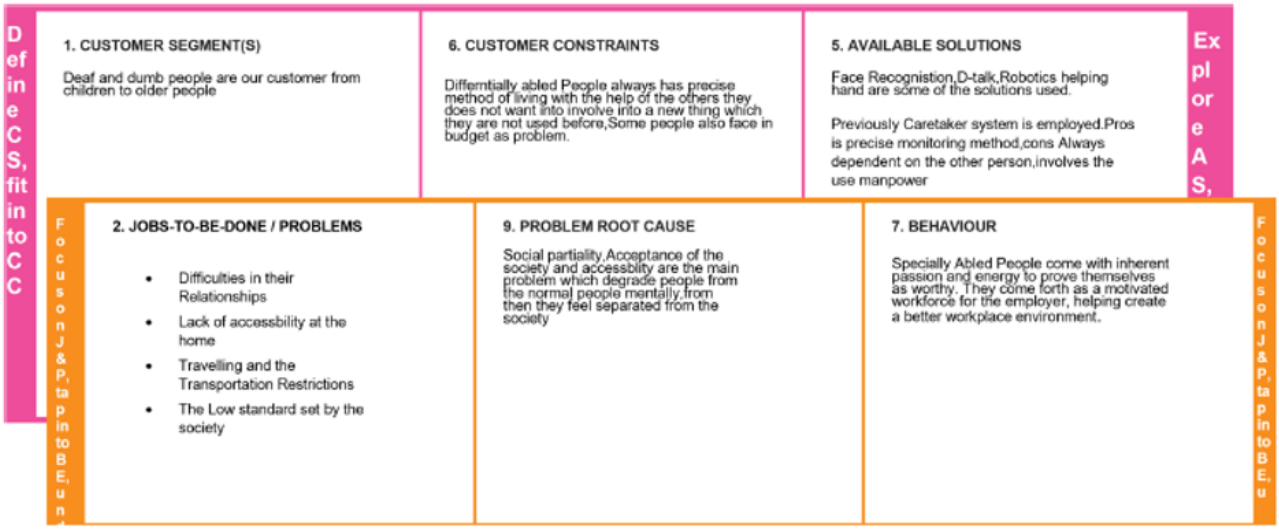
Mr. Muthu is a specially abled person (i.e., deaf and dumb) who uses sign language for communication. He use sign language to speak with other specially abled person but he can't communicate to normal people who can't understand sign language. He faces so many problem by doing so. He wants a system to make communication easier between him and the normal people who can't use sign language.

What type of users does the system focus most?	Specially abled.
Who does the problem affect?	Both specially abled and normal people.
What is the cause of the problem?	Does not know the sign language or its meaning.
How does the problem affect?	It decreases the communication between specially abled and normal person.
When this problem affects the most?	The problem affects when the communication takes place.
Does this problem can be solved? How?	This can be solved using Artificial Intelligence and Computer Vision.

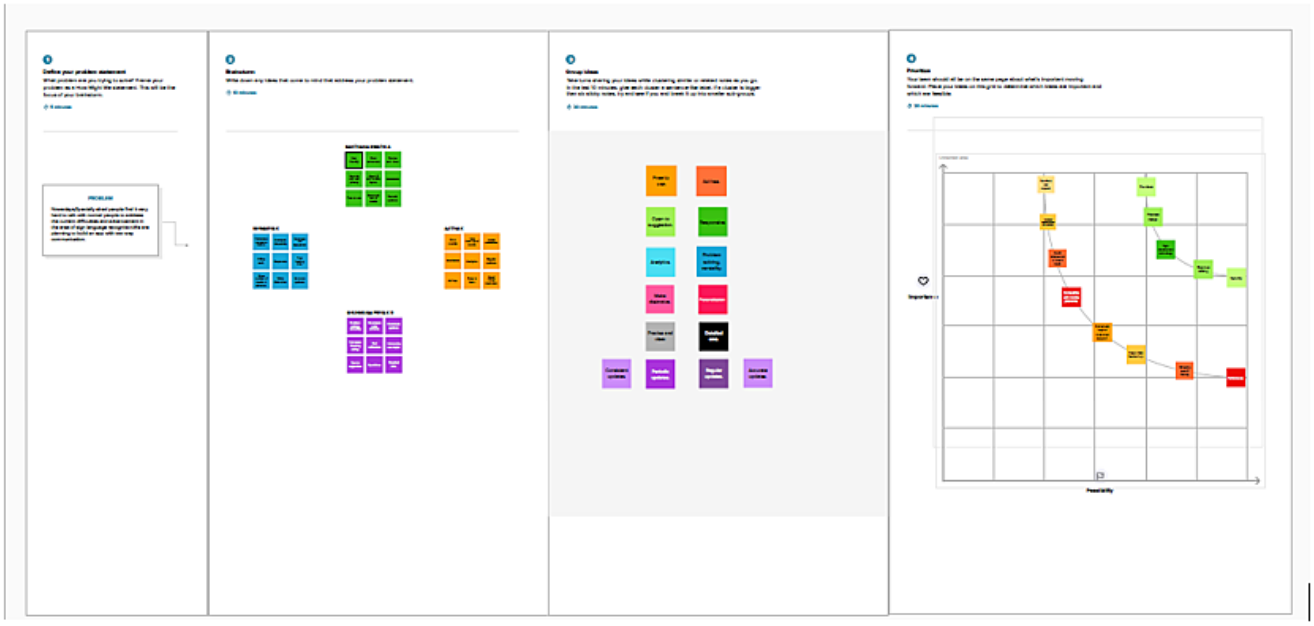
Can solving this problem increases the communication?	Yes communication will be increased.

4. IDEATION & PROPOSED SOLUTION

4.1. EMPATHY MAP



4.2. IDEATION & BRAINSTORMING



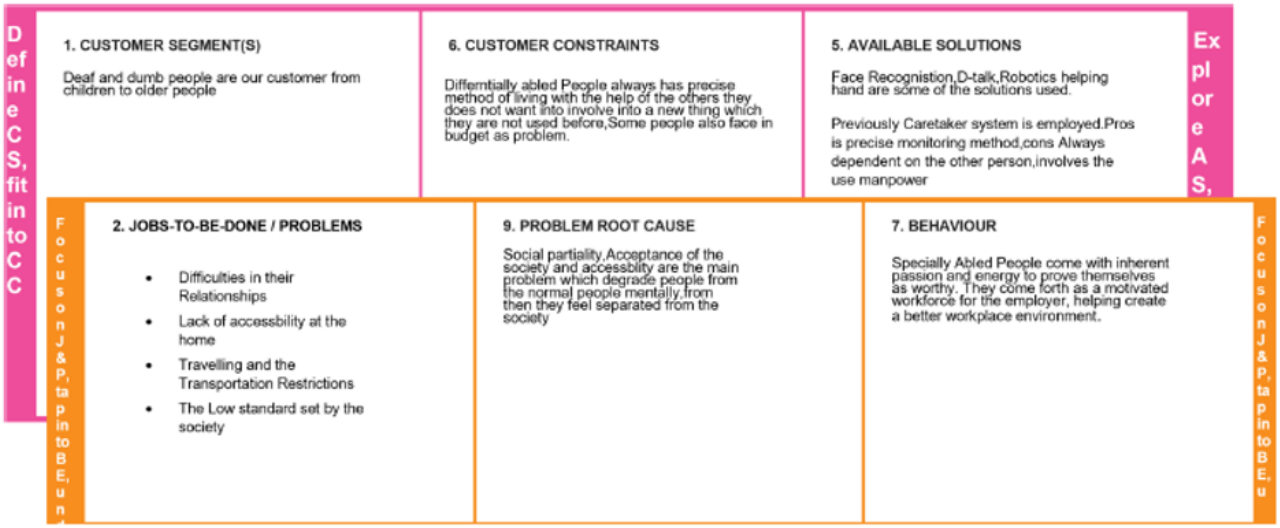
4.3. PROPOSED SOLUTION

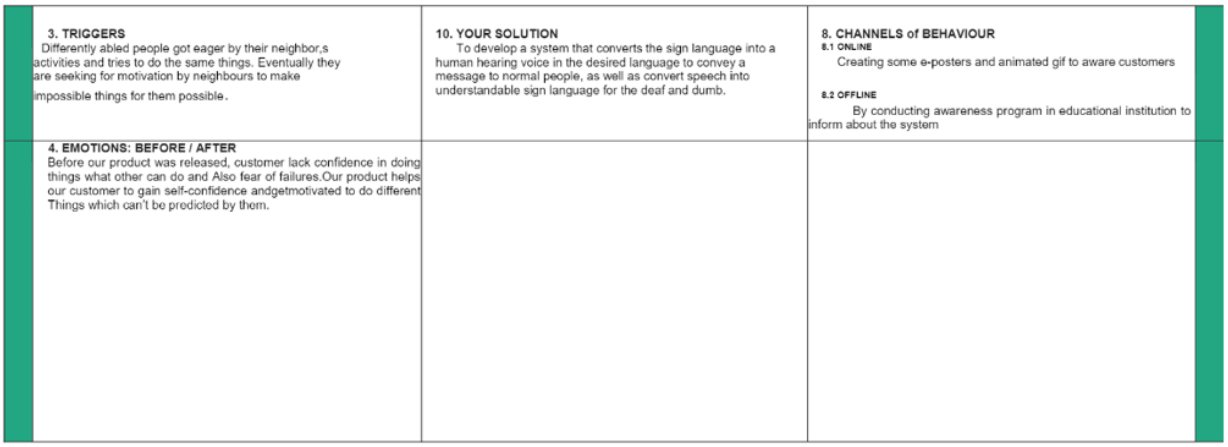
S.No.	Parameter	Description

1.	Problem Statement (Problem to be solved)	Communication between deaf-mute and normal person has always been challenging task. Since normal people are not trained on hand sign language .
2.	Idea / Solution description	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.
3.	Novelty / Uniqueness	Development of an app which enables deaf and dumb people to convey their information using signs.
4.	Social Impact / Customer Satisfaction	This system enables better understanding between normal person and deaf people thus it make sociably user friendly between both of them.
5.	Business Model (Revenue Model)	Targeted to person who are deaf and mute people .In India about 84 milllion people are deaf and mute according to Otology survey, this model helps for them, this inturn has steady revenue.

6.	Scalability of the Solution	This is highly scalable model because it can be used in gestrue recognistion, it is hands on model and it can be used in many applications.

4.4. PROBLEM SOLUTION FIT





5. REQUIREMENT ANALYSIS

5.1.Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Gmail .
FR-2	User verification	Verification email is sent to user whenever registered for first time.

FR-3	User Confirmation	Confirmation via Email Confirmation via OTP .
FR-4	Authentication	Device shutdown in case of cyber attack .
FR-5	Legal Requirements	Medical certificate is required.

5.2.Non-Functional Requirements:

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

FR No.	**Non-Functional Requirement**	**Description**
NFR-1	Portability	The transferability of device from one place to another is easy.
NFR-2	Security	Protect the sensitive information through password.

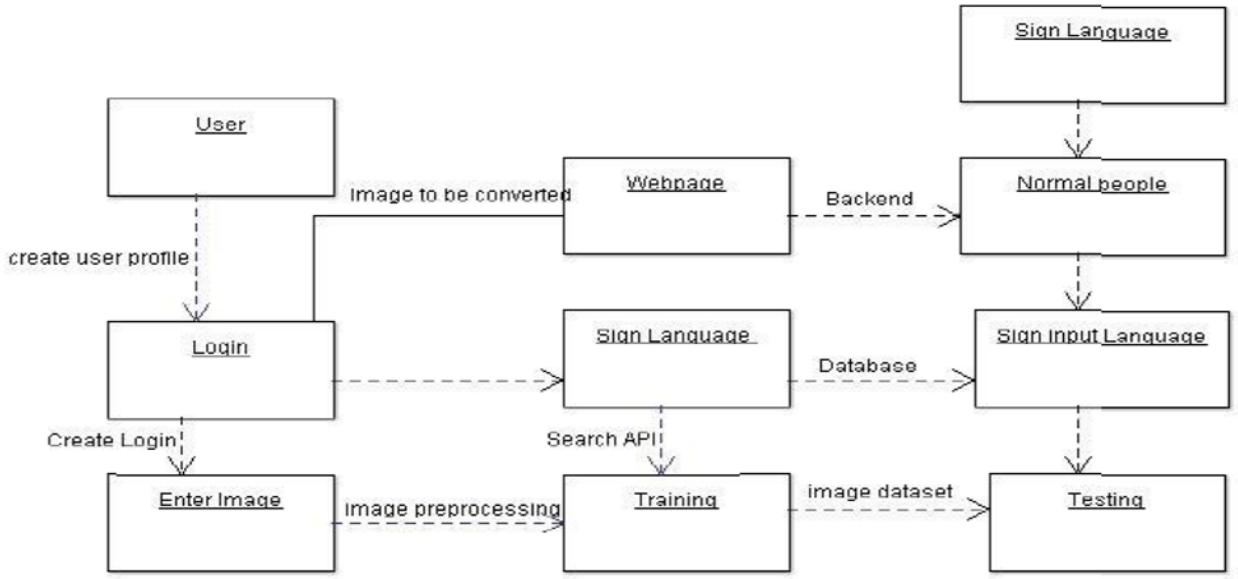
NFR-3	Reliability	This device has technology whose function is highly reliable.
NFR-4	Performance	Response time should be quick.
NFR-5	Availability	User access the device all hour a day .
NFR-6	Scalability	In standard network condition the device should convert information within second.

6. PROJECT DESIGN

6.1. 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,

Data Flow Diagram:



6.2. SOLUTION & TECHNICAL ARCHITECTURE

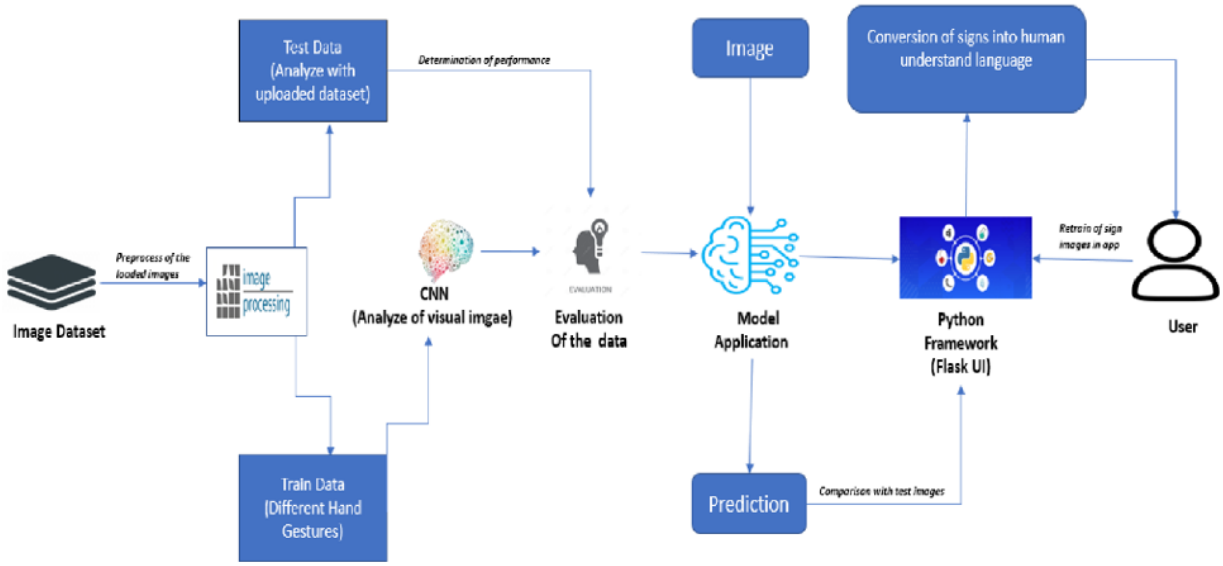


Fig: Solution Architecture

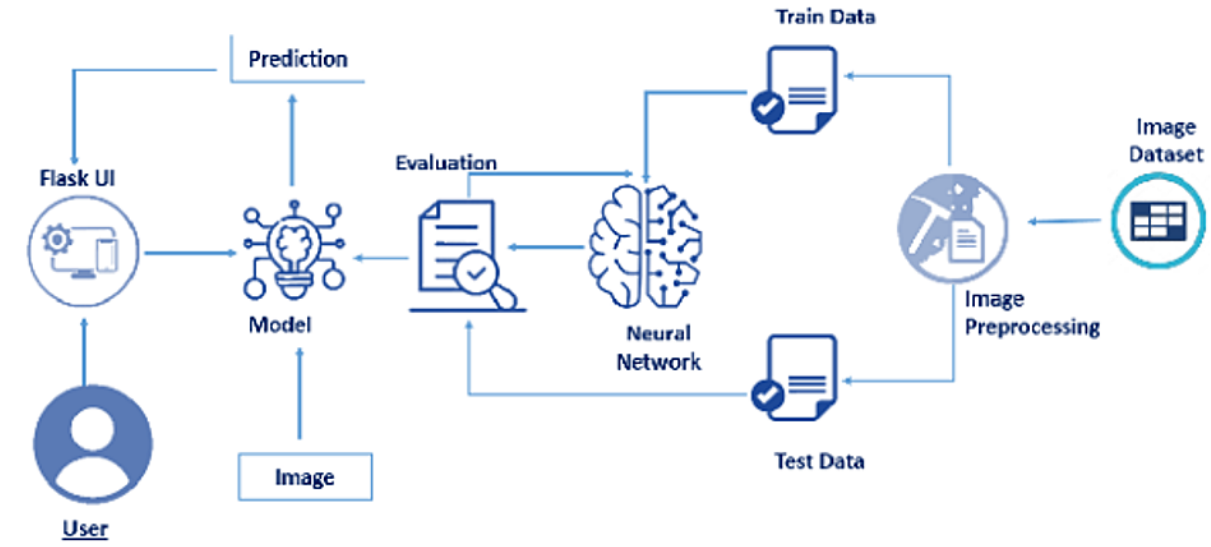


Fig: Technical Architecture

5.3. USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release

Customer (Low vision)	Registration	USN-1	As a user, who has trouble reading due to low vision, I want to be able to make the text larger on the screen so that I can read it.	I can access my account / dashboard	High	Sprint-1
Customer (Color blindness)		USN-2	As a user, who is color blind ,I want to have access to information conveyed in color so that, I do not miss anything and I understand the content.	I can receive confirmation email & click confirm	High	Sprint-1

Customer (Impaired user)		USN-3	As a user, who is hearing-impaired, I want a transcript of the spoken audio so that I can have access to all informati on provided in audio clips.	I can register & access the dashboa rd with Facebook Login Low Sprin t-2	Low	Sprint-2

6. PROJECT PLANNING& SCHEDULING

6.1. SPRINT PLANNING& ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story/Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my Password.	2	High	AJITHA

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

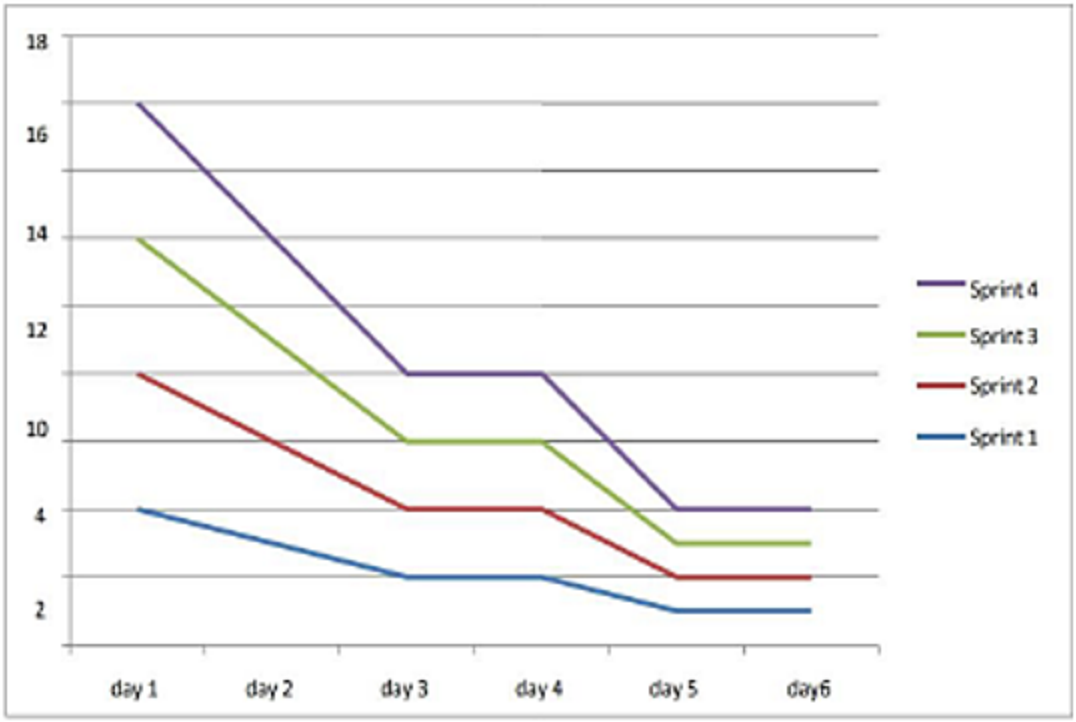
Sprint-4	Privacy	USN-8	The developed application should be secure for the users.	2	High	SHUNMUGA PRIYA

6.2. SPRINT DELIVERY SCHEDULE

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

- **Burndown Chart**

- **Velocity Chart**



7. CODING & SOLUTIONING

7.1IMAGE PREPROCESSING

1. **Importing Image Data Generator Library And Configure It**

```
[1]  from keras.preprocessing.image import ImageDataGenerator
     train_datagen=ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom_range=0.2,horizontal_flip=True)
     test_datagen=ImageDataGenerator(rescale=1./255)
```

- Image pre-processing includes zooming, shearing, flipping to increase the robustness of the model after it is built. Keras package is used for pre-processing images.
- Importing Image Data Generator Library to create an instance for which include shearing, rescale, zooming, etc to make the model robust with different types of images.

2. **Applying Image Data Generator Functionality To Train And Test Set**

```
[2] x_train = train_datagen.flow_from_directory(r'/content/drive/MyDrive/conversation engine for deaf and dumb/Dataset/training_set',target_size=(64,64),batch_size=
    Found 15770 images belonging to 9 classes.

[3] x_test = test_datagen.flow_from_directory(r'/content/drive/MyDrive/conversation engine for deaf and dumb/Dataset/test_set',target_size=(64,64),batch_size=300,cla
    Found 2250 images belonging to 9 classes.
```

7.2. MODEL BUILDING

3. **Importing The Required Model Building Libraries**

```
[4] from keras.models import Sequential
    from keras.layers import Dense
    from keras.layers import Convolution2D
    from keras.layers import MaxPooling2D
    from keras.layers import Dropout
    from keras.layers import Flatten
```

4. **Initializing The Model**

```
[5] model = Sequential()
```

5. **Adding The Convolution Layer**

```
[6] model.add(Convolution2D(32,(3,3),input_shape=(64,64,1), activation='relu'))
    #no. of feature detectors, size of feature detector, image size, activation function
```

6. **Adding The Pooling Layer**

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

7. **Adding The Flatten Layer**

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

8. **Compiling The Model**

```
[11] model.add(Dense(units=9,  activation = 'softmax'))
```

9. **Fit And Saving the Model**

```
model.fit_generator(x_train,steps_per_epoch=24,epochs=10,validation_data = x_test, validation_steps= 40)
#steps_per_epoch = no. of train images//batch size
```

```
/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 u
  """Entry point for launching an IPython kernel.
Epoch 1/10
24/24 [==============================] - ETA: 0s - loss: 1.0471 - accuracy: 0.7008  WARNING:tensorflow:Your input ran out of data; interrupting training. Make sur
24/24 [==============================] - 2079s 87s/step - loss: 1.0471 - accuracy: 0.7008 - val_loss: 0.3149 - val_accuracy: 0.9133
Epoch 2/10
24/24 [==============================] - 798s 33s/step - loss: 0.2237 - accuracy: 0.9373
Epoch 3/10
24/24 [==============================] - 468s 20s/step - loss: 0.1230 - accuracy: 0.9685
Epoch 4/10
24/24 [==============================] - 238s 10s/step - loss: 0.0752 - accuracy: 0.9812
Epoch 5/10
24/24 [==============================] - 131s 6s/step - loss: 0.0514 - accuracy: 0.9869
Epoch 6/10
24/24 [==============================] - 78s 3s/step - loss: 0.0402 - accuracy: 0.9899
Epoch 7/10
24/24 [==============================] - 46s 2s/step - loss: 0.0307 - accuracy: 0.9922
Epoch 8/10
24/24 [==============================] - 33s 1s/step - loss: 0.0211 - accuracy: 0.9959
Epoch 9/10
24/24 [==============================] - 27s 1s/step - loss: 0.0184 - accuracy: 0.9960
Epoch 10/10
24/24 [==============================] - 25s 1s/step - loss: 0.0160 - accuracy: 0.9967
<keras.callbacks.History at 0x7faa24ea3bd0>
```

7.3. TESTING THE MODEL

1. **Importing The Packages and Loading the Saved Model**

```
]  model.save('aslpng1.h5')
```

```
]  from keras.models import load_model
   import numpy as np
   import cv2
```

```
]  model=load_model('aslpng1.h5')
```

2. **Loading the Test Image, Pre-Processing it And Prediction**

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

```
]  frame=cv2.imread(r'/content/drive/MyDrive/conversation engine for deaf and dumb/Dataset/test_set/A/1.png')
   data = detect(frame)
```

```
   1/1 [==============================] - 0s 101ms/step
   [[9.8879457e-01 4.7520574e-09 3.4719829e-08 2.1185616e-07 1.1017337e-02
     1.3071556e-06 4.0822721e-05 2.1473575e-08 1.4569308e-04]]
   [0]
```

- The output [6] in the above image represents the index value in

the array [‘A’,’B’,’C’,’D’,’E’,’F’,’G’,’H’,’I’].

- Thus, the predicted alphabetis G.

7.4. FLASK APPLICATION

1. **Loading the required packages**

```
import numpy as np
import cv2
import os
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
from tensorflow.keras.backend import set_session
from flask import Flask, render_template,Response
import tensorflow as tf
from gtts import gTTS
global graph
global writer
from skimage.transform import resize
```

Initializing graph, loading the model, initializing the flask app and loading the video.

Graph element is required to work with Tensor Flow. So, graph element is created explicitly.

```
graph = tf.get_default_graph()
model = load_model('signlanguage1.h5')
vals = ['A','B','C','D','E','F','G','H','I']
app = Flask(__name__)
print("[INFO] accessing video stream...")
camera = cv2.VideoCapture(1)
camera.set(cv2.CAP_PROP_FRAME_WIDTH, 1280)
camera.set(cv2.CAP_PROP_FRAME_HEIGHT, 720)
pred=""
```

Configuring the home page

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

2. Pre-processing the frames captured from the camera

```
def detect(frame):
    global pred
    global graph
    img = resize(frame,(64,64,1))
    x = image.img_to_array(img)
    x = np.expand_dims(x,axis=0)
    with graph.as_default():
        predictions = model.predict_classes(x)
    print(predictions)
    pred=vals[predictions[0]]
    print(pred)
```

3.Video Feed call from the HTML PAGE

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

if __name__=='__main__':
    app.run()
```

7.5. HTML PAGE

```
<!doctype html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0, shrink-to-fit=no">
    <title>VHearuU</title>
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css">
    <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.12.0/css/all.css">
    <link rel="stylesheet" href="static/css/Banner-Heading-Image.css">
    <link rel="stylesheet" href="static/css/Navbar-Centered-Brand.css">
    <link rel="stylesheet" href="static/css/styles.css">
</head>
<body>

<nav class="navbar navbar-light navbar-expand-md py-3" style="background: #429691;">
        <div class="container">
            <div></div><a class="navbar-brand d-flex align-items-center" href="#"><h4 style="color: #ffffff; font-style: oblique; text-align: center;font-family: Arial
                    System Powered By AI&amp;nbsp;For Specially Abled - Team ID: PNT2022TMID47801</strong></h4></a>
            <div></div>
        </div>
    </nav>
    <section>
        <div class="d-flex flex-column justify-content-center align-items-center">
            <div class="d-flex flex-column justify-content-center align-items-center" id="div-video-feed"
                style="width: 800px;height: 600px;margin: 10px;min-height: 480px;min-width: 640px;border-radius: 10px;border: 5px groove #000000 ;">
                <img src="{{ url_for('video_feed') }}" style="width: 100%;height: 100%;color: rgb(255,255,255);text-align: center;font-size: 20px;"
                    alt="Camera Access Not Provided!">
            </div>
        </div>
    </section>
<section>
        <div class="container">
            <div class="accordion text-white" role="tablist" id="accordion-1">

                <div class="accordion-item" style="font-style: oblique; background: #429691;">
                    <h2 class="accordion-header" role="tab"><button class="accordion-button collapsed"
```

HTML page to display the processed video on the screen, so that the person can show signs which can be detected.

8. TESTING

8.1. TEST CASES

Test Case ID	Test Scenario	Steps to Execute	Expected Result	Actual Result
1.	Verify if user is able to provide camera access.	1.Enter URL and click go. 2. Give camera access.	Camera is On.	Working as expected.
2.	Verify if user is able to get the desirable prediction for the gesture.	1.Enter URL and click go. 2. Give camera access. 3. Make gesture in front of camera.	Alphabet is predicted for the gesture.	Working as Expected.

8.2. USER ACCEPTANCE TESTING

1. **Defect Analysis**

Resolution	Severity 1	Severity 1	Severity 1	Severity 1	Subtotal
By Design	0	12	1	1	14
External	5	0	0	0	5
Fixed	11	3	2	2	18
Skipped	0	0	2	0	2
Won’t Fix	4	0	0	0	4
Totals	20	15	5	3	43

2. **Test Case Analysis**

Section	Total Cases	Not Tested	Fail	Pass
Client Application	5	1	0	5
Security	2	0	0	2
Exception Reporting	2	0	0	2
Final Report Output	9	0	0	9

• The project developed was tested by an end user and the application converts the gestures to its respective alphabet accurately.

9. RESULTS

9.1. PERFORMANCE METRICS

a. Model Summary

```
In [40]: 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
_________________________________________________________________
```

b. Confusion Matrix and Classification Report

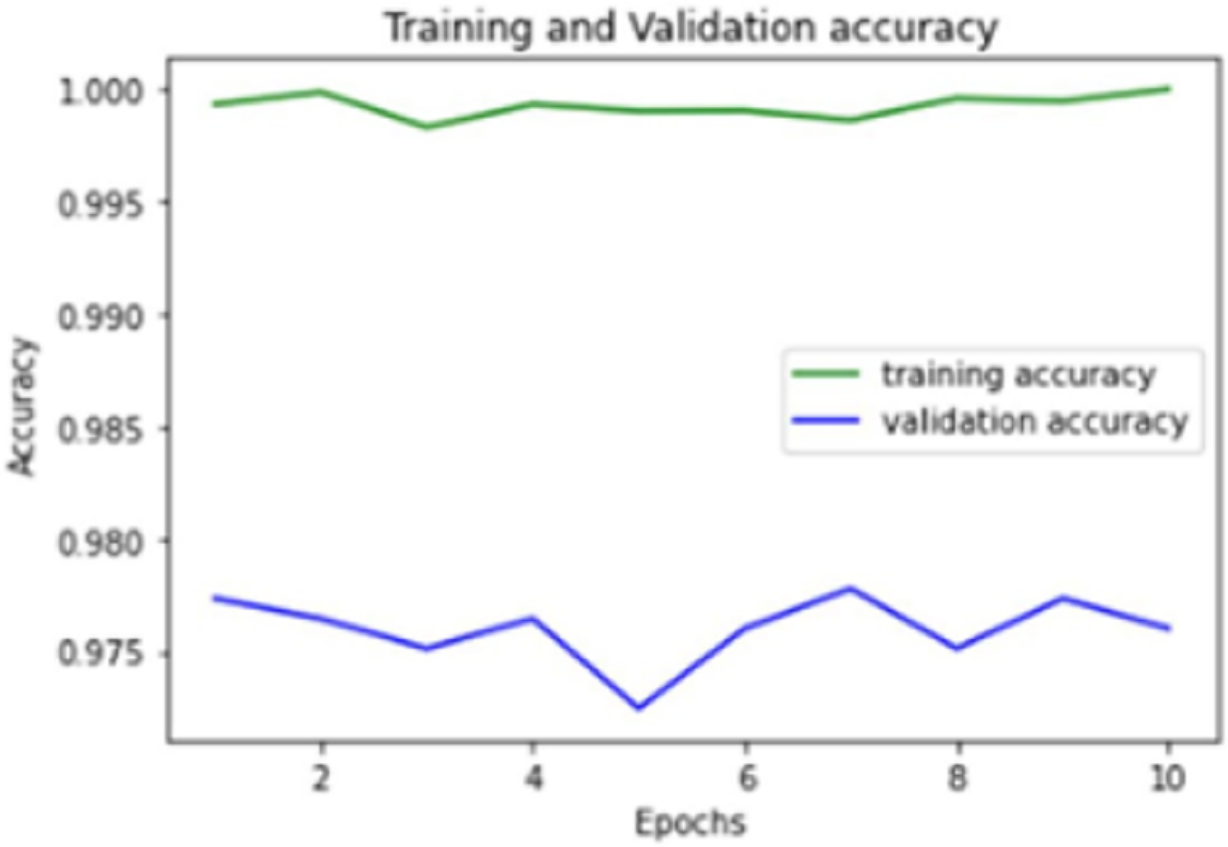
```
[[38 31 33 26 29 22 31 19 21]
 [31 28 25 27 26 26 33 26 28]
 [22 18 28 34 30 36 33 21 28]
 [32 21 23 34 30 24 42 22 22]
 [29 23 29 18 25 30 32 30 34]
 [20 29 27 26 32 25 32 22 37]
 [27 30 26 32 21 31 33 26 24]
 [26 41 25 26 24 26 30 25 27]
 [25 29 33 28 33 30 29 14 29]]

Classification Report
              precision    recall  f1-score   support

           A       0.15      0.15      0.15       250
           B       0.11      0.11      0.11       250
           C       0.11      0.11      0.11       250
           D       0.14      0.14      0.14       250
           E       0.10      0.10      0.10       250
           F       0.10      0.10      0.10       250
           G       0.11      0.13      0.12       250
           H       0.12      0.10      0.11       250
           I       0.12      0.12      0.12       250

    accuracy                           0.12      2250
   macro avg       0.12      0.12      0.12      2250
weighted avg       0.12      0.12      0.12      2250
```

Accuracy



10. ADVANTAGES & DISADVANTAGES

3. **Advantages**

- The application is conveniently simple for the end user.
- The user interface is not complex.

a. **Disadvantages**

- The dataset in limited. The alphabets only range from ‘A’ to ‘J’.
- As of now, only static gesture are converted.

11. CONCLUSION

The main objective of this project is to develop gesture recognition so that the deaf can communicate with normal individuals. One of the crucial tasks is the extraction of features, and various gestures should yield various, effectively distinguishable characteristics. To identify the character from the gesture images, we used a trained dataset for the CNN algorithm. These features combined with a labelled data enable accurate realtime ASL alphabet recognition. Our analysis found that accuracy is influenced by a variety of elements, including the camera, dataset, and approach. The accuracy drastically declines in low light and noisy backgrounds.

12. FUTURE SCOPE

The proposed system can be translated into multiple languages, enhancing its dependability and effectiveness. In the near future, it might only be accessible through mobile devices, making the system more convenient and portable. This system is unable to detect gestures made with both hands. Therefore, detecting gestures done with both hands could be a future project.

13. APPENDIX

13.1. SOURCE CODE

- Source Code is available in the GitHub link provided in Section 13.2.

13.2. GITHUB & PROJECTDEMO LINK

- GitHub: https://github.com/IBM-EPBL/IBM-Project-3324-1658550408/tree/main/Project%20Development%20Phase

- Project Demo Link: https://drive.google.com/file/d/17sOQbMVWoanP9eHs93b6HLBqqcOSrzDG/view?usp=drivesdk