Project Report

Team ID	PNT2022TMID33196
Project Name	Real-Time Communication System Powered by AI for Specially Abled
Team Lead	HARIHARAN P(Reg no:922119106030)
Team Member 1	BALAJI G(Reg no:922119106015)
Team Member 2	DHEENU GOWTHAM S(Reg no:922119106020)
Team Member 3	DOMNIC VISHAL J(Reg no:922119106025)
Team Member 4	GAJENDRAN B(Reg no:922119106027)

1.INTRODUCTION

1.1PROJECT OVERVIEW

The project developed is a system that converts hand gesturesof 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 thecommunication during the time of emergencies. The project is developed on Python Platform using CNN (Convolutional Neural Network) model from TensorFlowpackage.

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 situationslike an interview. This projectaims to overcome the said challenges.

2.LITERATURE SURVEY

2.1.EXISTING PROBLEMS

● Existing system(or) frameworks has too many false positives. The system predicts the gestures inaccurately.
● Real Time recognition of gestures into text/speech and text/speech into gestures is not available.

2.2.REFERENCES

i. Saed Mian Qaisar, Sarah Niyazi, Abdulhamit Subasi, “Efficient Isolated Speech to Sign Conversion Based on the Adaptive Rate Processing”; Procedia Computer Science,Vol. 163, PP. 35–40, 2019.

ii. T. Bohra, S. Sompura, K. Parekh and P. Raut, "Real-Time Two Way Communication System for Speech and Hearing ImpairedUsing Computer Visionand Deep Learning" International Conference on Smart Systems and InventiveTechnology (ICSSIT), pp. 734-739, 2019.

iii. Ma, Jiyong,Wen Gao, JiangqinWu, and Chunli Wang. "Acontinuous Chinese Sign Language recognition system." In Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580),pp. 428-433. IEEE, 2000.

iv. Vogler, C., and D. Handshapes Metaxas. "Movements: Multiple-Channel American Sign Language Recognition." Gesture-Based Communication in Human-Computer Interaction. Lecture Notes in Computer Science: 247-258.

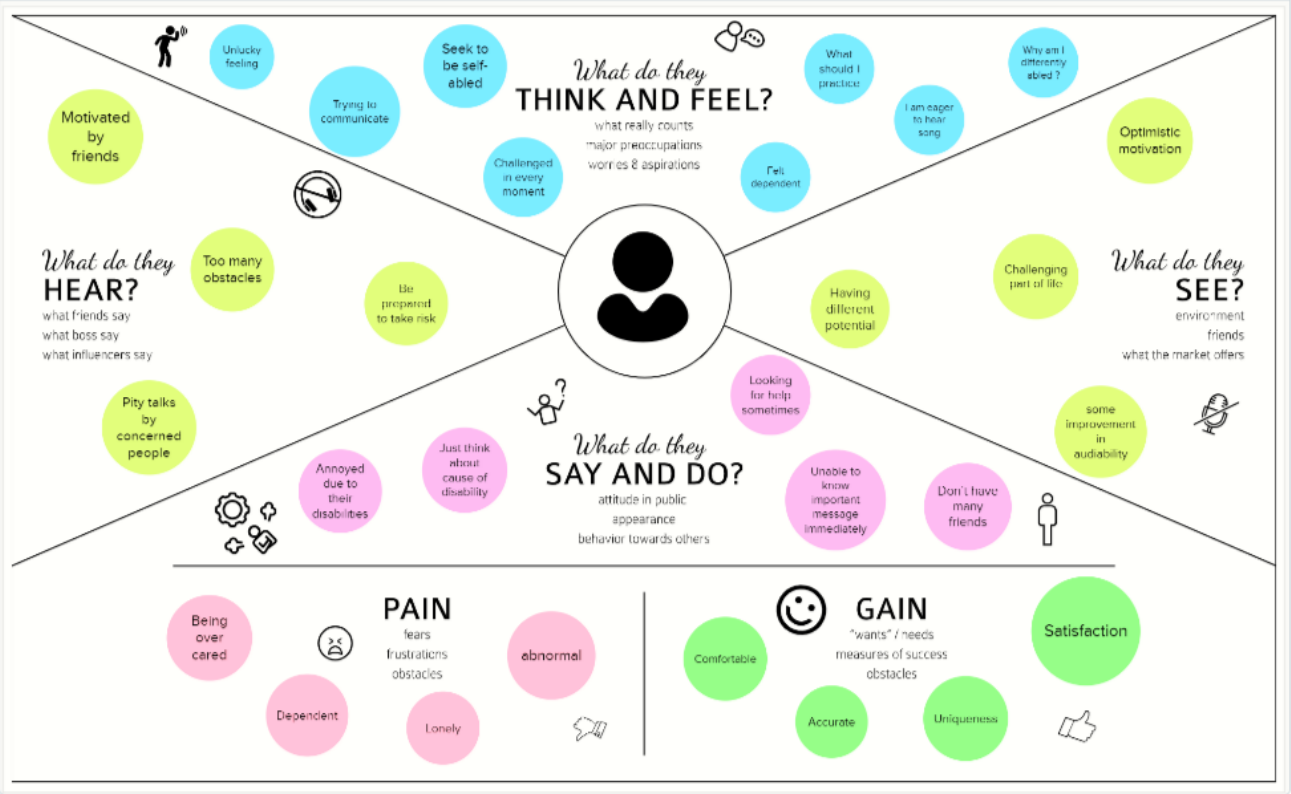
v. Pavlovic, V, Sharma, R., &Huang T., “Visual Interpretation of Hand Gesturesfor Human-ComputerInteraction (HCI): A Review”,IEEE TOPAMI, VOL. 19, NO. 7, 1999.

2.3.PROBLEM STATEMENTDEFINITION

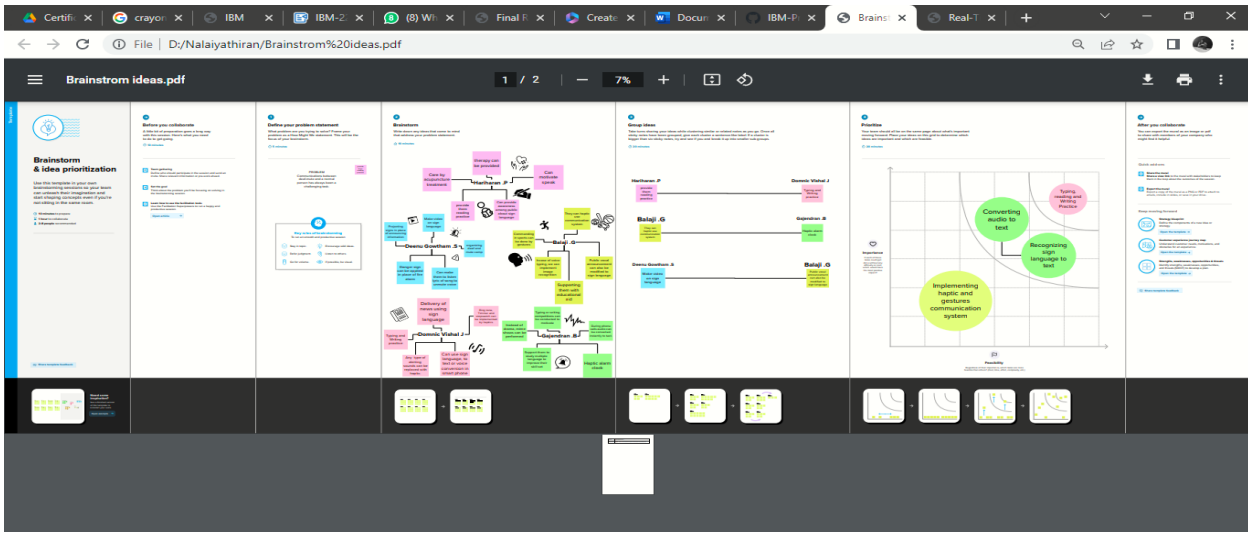
The study of human-computer interaction has shown a great deal of interest in hand gesture recognition. In many areas of human-computer interaction, including virtual reality, gaming, automobile system control, and robotic control, quick and precise hand gesture recognition is crucial. As more sensors are added, there are numerous different ways to categorise hand motions. Since gesture identification is a problem of image classification and 2D CNNs are effective in image classification, we have chosen to employ a convolutional neural network for this task. A system that converts the sign language into the respective ASL (American Sign Language)alphabet to convey a message to normal people is developed in this project.

3.IDEATION & PROPOSED SOLUTION

3.1.EMPATHY MAP CANVAS



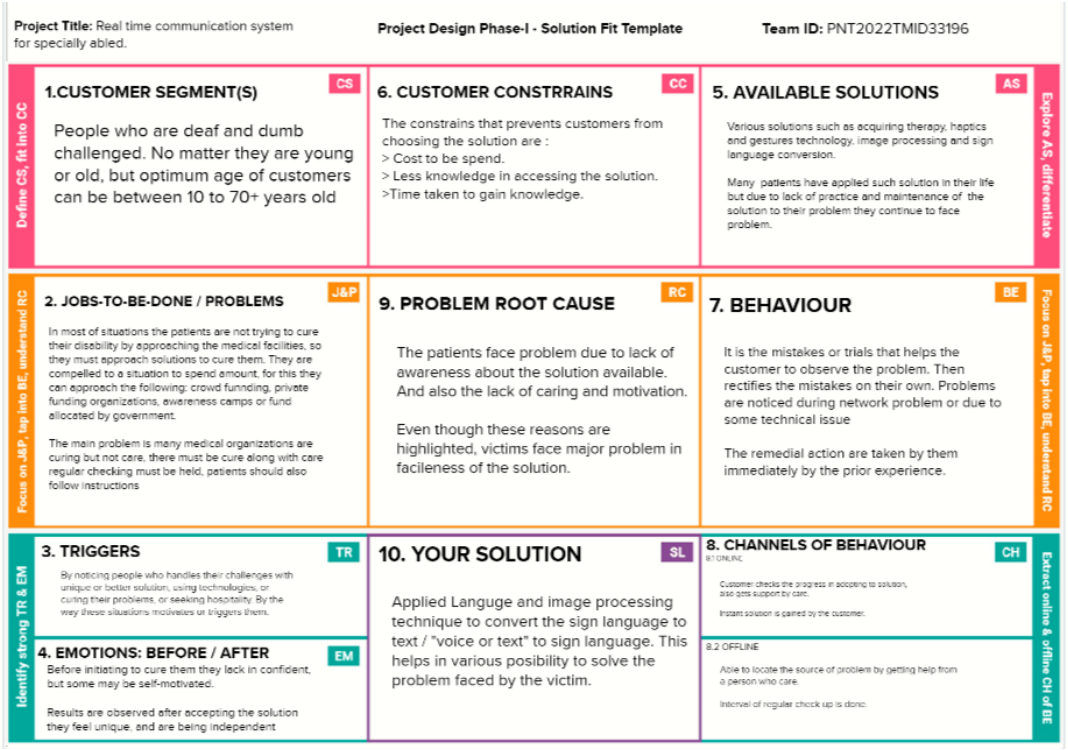
3.2.IDEATION & BRAINSTORMING



3.3.PROPOSED SOLUTION

S.NO	Parameters	Descriptions
1	Problem Statement (Problem to be solved)	To solve their communication trouble with AI.
2	Idea / Solution description	AI technology can empower people living with limited physical mobility. AI for Accessibility program uses the potential of Artificial Intelligence to develop solutions to many physical and cognitive challenges disabled individuals face at work and in daily life to promote social inclusion for them.
3	Novelty / Uniqueness	•Converts the sign language into a human hearing voice . •Convert speech into understandable sign language for the deaf and dumb. •Emergency button.
4	Social Impact / Customer Satisfaction	The customer can convey their information without any trouble.
5	Business Model (Revenue Model)	Selling this product to specially abled perosons as well as normal people.To access more features must pay(Freemium model).
6	Scalability of the Solution	AI based real time communication system To empower specially abled to chase their dreams without any communication trouble.

3.4.PROBLEM SOLUTION FIT



4.Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No	Functional Requirement (Epic)	Sub Requirement (Story / SubTask)
1	User Registration	To get required informations from cutomer for registration process
2	User Confirmation	Confirm the details with Email and OTP verifications.
3	Permission access	Permit the required functions like GPS,Micro phone and Camera.
4	Information provide priority based.	From the frequent information provide the particular information first.
5	Feedback	To get feedback from

		customer via product or software.
6	Troubleshoot	To solve problems in product with customer feedback.

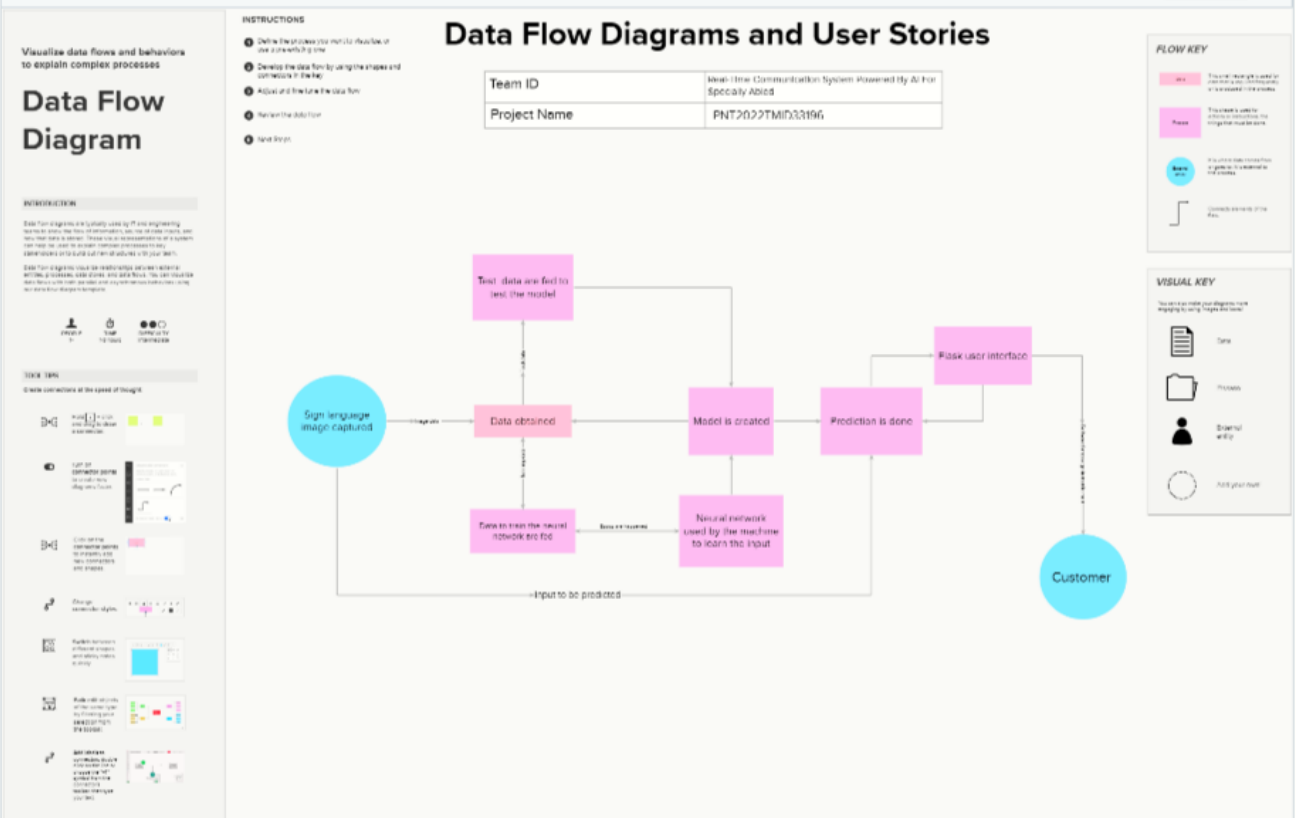
Non-functional Requirements:

Following are the non-functional requirements of the proposed solution

FR No.	Non-Functional Requirement	Description
1	Usability	Easily access sign languages during process.
2	Security	To ensure customer personal informations from malware and hackers.
3	Reliability	To perform sign language conversion without any interruptions.
4	Performance	Sign language conversion should not load more than one minute.
5	Availability	Availability describes how likely the system is accessible to a user at a given point in time.
6	Scalability	Handle many sign languages conversions without any product degradation.

5.PROJECT DESIGN

5.1:Data Flow Diagram:



User Stories:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Deaf-Mute people)	User Input	USN-1	As a user, I can input my sign-language to the system for processing.	The user can input sign language into the system	Low	Sprint-3
		USN-2	As a user, I can input sign-language images to the system for processing.	The user can input images into the system	High	Sprint-1
		USN-3	As a user, I can make sure the input is captured correctly by the system.	The system should capture the input correctly	Medium	Sprint-2
	Processing	USN-4	As a user, I can ensure that the sign language input is correctly getting translated into normal message and voice.	The user can ensure that the processing is done correctly.	Medium	Sprint-2
		USN-5	As a user, I can get acknowledgement from the system about the processing of the input.	The user should get an acknowledgement	High	Sprint-1
		USN-6	As a user, I will get feedback about the processing of the system.	The user should get feedback from the system	Low	Sprint-3
	System Output	USN-7	As a user, I can acknowledge the output of the system by ensuring messages are displayed.	The user should get an acknowledgement from the system	High	Sprint-1
		USN-8	As a user, I can get feedback about the system from its output.	The user should get feedback from the system	Medium	Sprint-2

5.2:Solution Architecture:

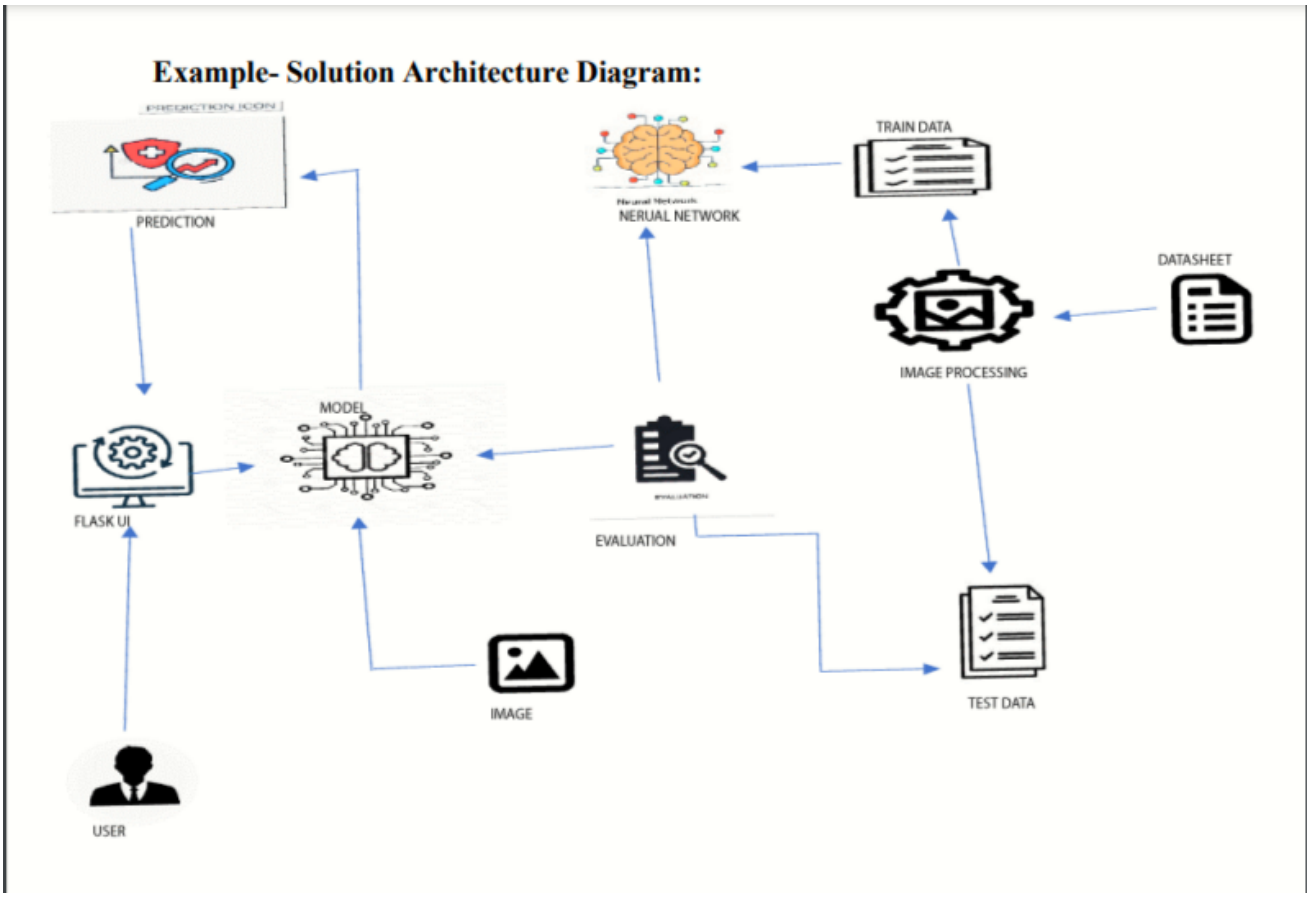
Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

• Find the best tech solution to solve existing business problems.
cv

• Describe the structure, characteristics, behaviour, and other aspects of the software to project stake holders.

• Define features, development phases, and solution requirements.

• Provide specifications according to which the solution is defined, managed, and delivered.



5.3:Technical Architecture:

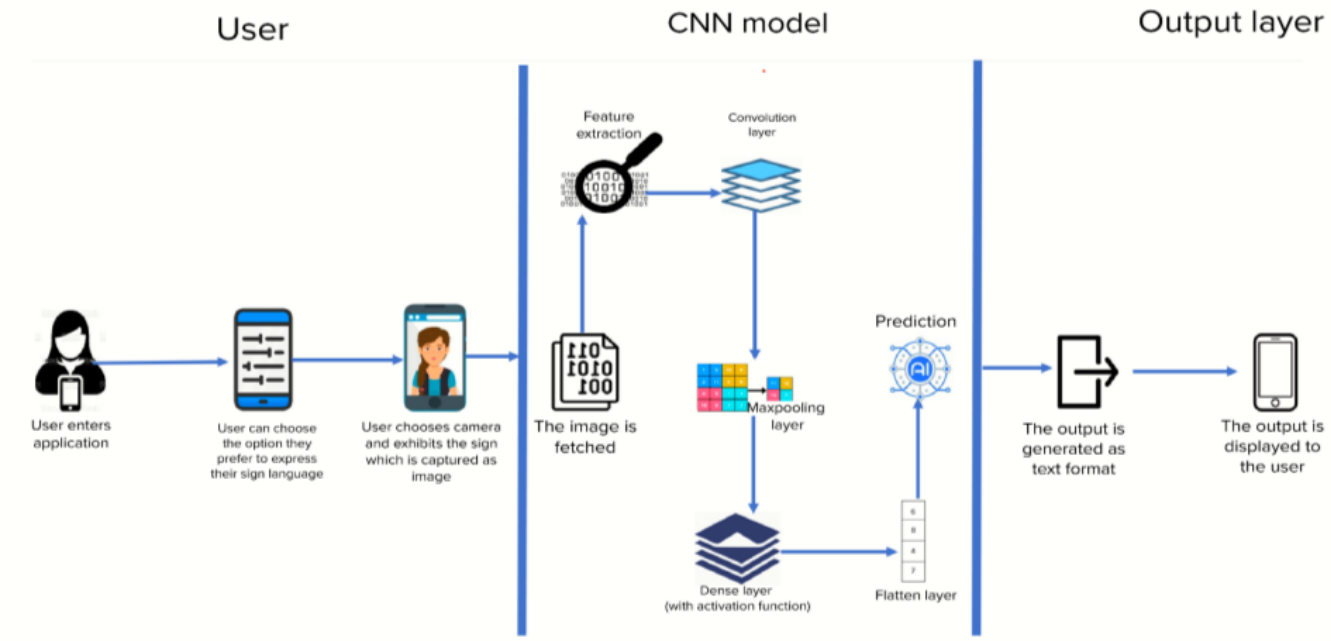


Table-1: Components & Technologies:

S. No	Component	Description	Technology
1.	User (Deaf-mute)	The deaf-mute user will benefit from the system which uses several technologies.	Cloud tech, OpenCV and AI tech like Machine Learning, Deep Learning, etc.
2.	User Interface	The user interface lets the user interact with the system which is hosted in the cloud.	Suitable UI Technology, Cloud Hosting
3.	Models	A machine learning model is used to classify our gesture image dataset.	Machine Learning
4.	Image Prediction	The image prediction is done with the help of deep learning which implements neural networks of various kinds to solve the problem.	ANN, CNN
5.	Image	Image processing is done on input image.	OpenCV
6.	Speech	The output of the system is speech (voice) to be heard for normal users.	Suitable Speech System

Table-2: Application Characteristics:

S. No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Our system implements many open-source frameworks.	AI frameworks, OpenCV, Speech System, UI system, Python Language
2.	Security Implementations	Necessary security measures will be implemented in the system.	Necessary Security Technologies
3.	Scalable Architecture	The architecture is very much scalable to accommodate any future needs.	Scalable Technologies
4.	Availability	The system will be made ubiquitous so that it is available everywhere.	Necessary Technologies
5.	Performance	The model will be fine-tuned to strike a balance between accuracy vs performance.	Optimization of code and trained model

6.PROJECT PLANNING & SCHEDULING:

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

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Data collection	USN-1	Collection of required data, login information from user	2	Low	GAJENDRAN
Sprint-1		USN-2	Image pre-processing	3	High	HARIHARAN
Sprint-2	Model building	USN-3	Import the required libraries, add the necessary layers, and compile the model	2	Low	DOMINIC VISHAL
Sprint-2		USN-4	Training the image classification model using CNN	3	High	BALAJI
Sprint-3	Training and testing	USN-5	Training the model and testing the model's performance	3	High	GAJENDRAN
Sprint-3		USN-6	Converting the input sign language images into English alphabets and save model for deployment	2	Low	DHEENU GOWTHAM
Sprint-4	Implementation and dashboard	USN-7	As a user, I can acknowledge the output of the system by ensuring messages are displayed.	2	Low	BALAJI
Sprint-4		USN-8	As a user, I can get and give feedback about the system from its output.	3	High	HARIHARAN

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

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	05	6 Days	24 Oct 2022	29 Oct 2022	05	05 Nov 2022
Sprint-2	05	6 Days	31 Oct 2022	05 Nov 2022	05	08 Nov 2022
Sprint-3	05	6 Days	07 Nov 2022	12 Nov 2022	05	12 Nov 2022
Sprint-4	05	6 Days	14 Nov 2022	19 Nov 2022	05	19 Nov 2022

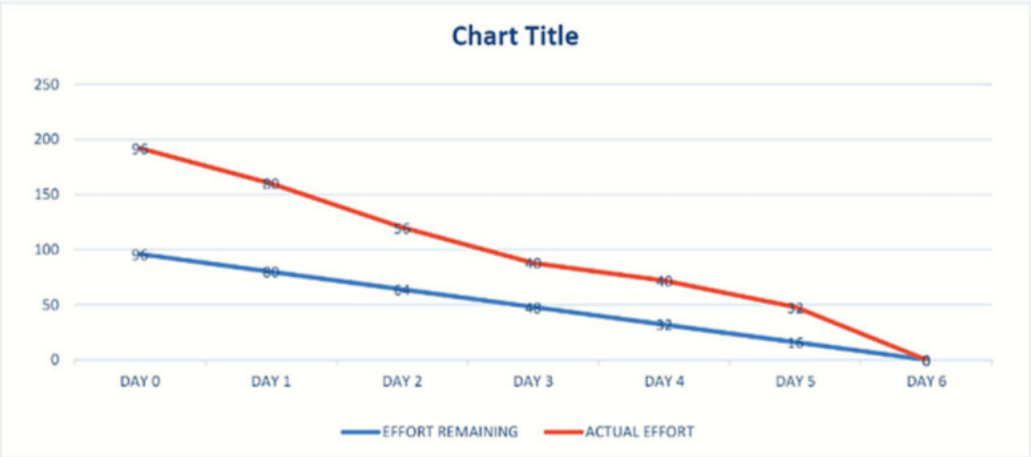
Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

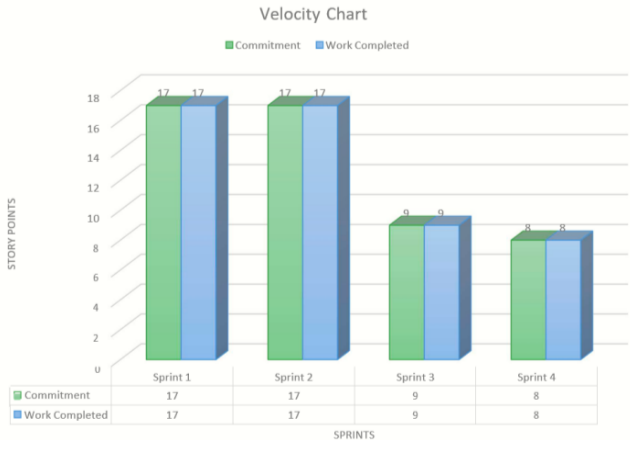
$$AV = \frac{sprint\ duration}{velocity}$$

$$\mathbf{AV = 5/10 = 0.5}$$

- Burndown Chart



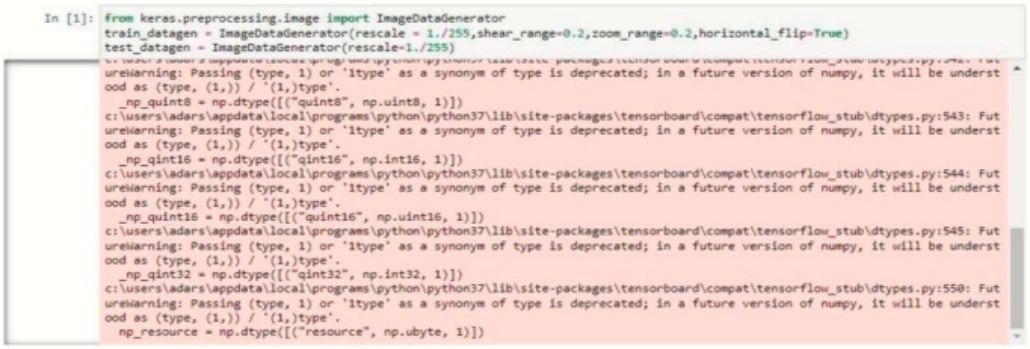
- Velocity Chart



7.1IMAGE PREPROCESSING

7.1IMAGE PREPROCESSING

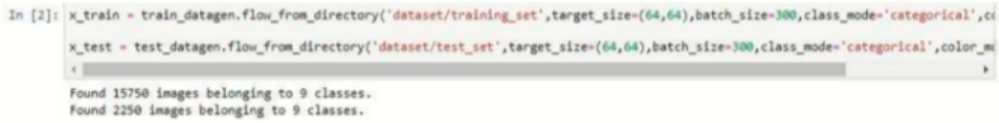
- Image pre-processing includes zooming, shearing,flipping to increasethe robustness of the model after itis built. Keras package is used for pre-processing images.
- Importing ImageDataGenerator Library to create an instancefor which include shearing, rescale,zooming,etc to make the model robust with differenttypes of images.



```
In [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)
```

```
ureWarning: Passing (type, 1) or 'itype' as a synonym of type is deprecated; in a future version of numpy, it will be underst
ood as (type, (1,)) / '(1,)type'.
  _np_quint8 = np.dtype([("quint8", np.uint8, 1)])
c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\tensorboard\compat\tensorflow_stub\dtypes.py:543: Fut
ureWarning: Passing (type, 1) or 'itype' as a synonym of type is deprecated; in a future version of numpy, it will be underst
ood as (type, (1,)) / '(1,)type'.
  _np_qint16 = np.dtype([("qint16", np.int16, 1)])
c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\tensorboard\compat\tensorflow_stub\dtypes.py:544: Fut
ureWarning: Passing (type, 1) or 'itype' as a synonym of type is deprecated; in a future version of numpy, it will be underst
ood as (type, (1,)) / '(1,)type'.
  _np_quint16 = np.dtype([("quint16", np.uint16, 1)])
c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\tensorboard\compat\tensorflow_stub\dtypes.py:545: Fut
ureWarning: Passing (type, 1) or 'itype' as a synonym of type is deprecated; in a future version of numpy, it will be underst
ood as (type, (1,)) / '(1,)type'.
  _np_qint32 = np.dtype([("qint32", np.int32, 1)])
c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\tensorboard\compat\tensorflow_stub\dtypes.py:550: Fut
ureWarning: Passing (type, 1) or 'itype' as a synonym of type is deprecated; in a future version of numpy, it will be underst
ood as (type, (1,)) / '(1,)type'.
  np_resource = np.dtype([("resource", np.ubyte, 1)])
```

- Applying ImageDataGenerator Functionality To Train And Test Set



```
In [2]: x_train = train_datagen.flow_from_directory('dataset/training_set',target_size=(64,64),batch_size=300,class_mode='categorical',c

        x_test = test_datagen.flow_from_directory('dataset/test_set',target_size=(64,64),batch_size=300,class_mode='categorical',color_m

Found 15750 images belonging to 9 classes.
Found 2250 images belonging to 9 classes.
```

7.2.MODEL BUILDING

- Importing The RequiredModel Building Libraries

```
In [3]: 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
```

- Initializing The Model

```
In [4]: model = Sequential()
```

```
WARNING:tensorflow:From c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\keras\backend\tensorflow_backen
d.py:74: The name tf.get_default_graph is deprecated. Please use tf.compat.v1.get_default_graph instead.
```

- Adding The Convolution Layer

```
In [5]: model.add(Convolution2D(32,(3,3),input_shape=(64,64,1),activation='relu'))
```

```
WARNING:tensorflow:From c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\keras\backend\tensorflow_backen
d.py:517: The name tf.placeholder is deprecated. Please use tf.compat.v1.placeholder instead.

WARNING:tensorflow:From c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\keras\backend\tensorflow_backen
d.py:4138: The name tf.random_uniform is deprecated. Please use tf.random.uniform instead.
```

- Adding The Pooling Layer

```
In [6]: model.add(MaxPooling2D(pool_size=(2,2)))
```

```
WARNING:tensorflow:From c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\keras\backend\tensorflow_backen
d.py:3976: The name tf.nn.max_pool is deprecated. Please use tf.nn.max_pool2d instead.
```

- Adding The Flatten Layer

```
In [7]: model.add(Flatten())
```

- Compiling The Model

```
In [9]: model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
```

```
WARNING:tensorflow:From c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\keras\optimizers.py:790: The na
me tf.train.Optimizer is deprecated. Please use tf.compat.v1.train.Optimizer instead.

WARNING:tensorflow:From c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\keras\backend\tensorflow_backen
d.py:3295: The name tf.log is deprecated. Please use tf.math.log instead.
```

- Fit And Saving the Model

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

         model.save('signlanguage.h5')
```

```
WARNING:tensorflow:From c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\tensorflow\python\ops\math_gra
d.py:1250: add_dispatch_support.<locals>.wrapper (from tensorflow.python.ops.array_ops) is deprecated and will be removed in a
future version.
Instructions for updating:
Use tf.where in 2.0, which has the same broadcast rule as np.where
WARNING:tensorflow:From c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\keras\backend\tensorflow_backen
d.py:986: The name tf.assign_add is deprecated. Please use tf.compat.v1.assign_add instead.

Epoch 1/10
24/24 [==============================] - 17s 720ms/step - loss: 1.3771 - acc: 0.6265 - val_loss: 0.5072 - val_acc: 0.8889
Epoch 2/10
24/24 [==============================] - 15s 635ms/step - loss: 0.3532 - acc: 0.9039 - val_loss: 0.2881 - val_acc: 0.9362
Epoch 3/10
24/24 [==============================] - 15s 633ms/step - loss: 0.1826 - acc: 0.9522 - val_loss: 0.1999 - val_acc: 0.9471
Epoch 4/10
24/24 [==============================] - 15s 645ms/step - loss: 0.1130 - acc: 0.9707 - val_loss: 0.1982 - val_acc: 0.9676
Epoch 5/10
24/24 [==============================] - 15s 630ms/step - loss: 0.0773 - acc: 0.9802 - val_loss: 0.1742 - val_acc: 0.9653
Epoch 6/10
24/24 [==============================] - 15s 641ms/step - loss: 0.0534 - acc: 0.9872 - val_loss: 0.1953 - val_acc: 0.9733
Epoch 7/10
24/24 [==============================] - 15s 633ms/step - loss: 0.0366 - acc: 0.9917 - val_loss: 0.1928 - val_acc: 0.9698
Epoch 8/10
24/24 [==============================] - 15s 637ms/step - loss: 0.0514 - acc: 0.9936 - val_loss: 0.2024 - val_acc: 0.9716
Epoch 9/10
24/24 [==============================] - 15s 638ms/step - loss: 0.0271 - acc: 0.9939 - val_loss: 0.2371 - val_acc: 0.9758
Epoch 10/10
24/24 [==============================] - 15s 634ms/step - loss: 0.0192 - acc: 0.9965 - val_loss: 0.1944 - val_acc: 0.9747
```

7.3.TESTING THE MODEL

- Importing The Packagesand Loading the Saved Model

```
In [1]: from keras.models import load_model
        from keras.models import Sequential
        import numpy as np
        import cv2
```

```
c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\tensorboard\compat\tensorflow_stub\dtypes.py:541: Fut
ureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be underst
ood as (type, (1,)) / '(1,)type'.
  _np_qint8 = np.dtype([("qint8", np.int8, 1)])
c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\tensorboard\compat\tensorflow_stub\dtypes.py:542: Fut
ureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be underst
ood as (type, (1,)) / '(1,)type'.
  _np_quint8 = np.dtype([("quint8", np.uint8, 1)])
c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\tensorboard\compat\tensorflow_stub\dtypes.py:543: Fut
ureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be underst
ood as (type, (1,)) / '(1,)type'.
  _np_qint16 = np.dtype([("qint16", np.int16, 1)])
c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\tensorboard\compat\tensorflow_stub\dtypes.py:544: Fut
ureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be underst
ood as (type, (1,)) / '(1,)type'.
  _np_quint16 = np.dtype([("quint16", np.uint16, 1)])
c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\tensorboard\compat\tensorflow_stub\dtypes.py:545: Fut
ureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be underst
ood as (type, (1,)) / '(1,)type'.
  _np_qint32 = np.dtype([("qint32", np.int32, 1)])
c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\tensorboard\compat\tensorflow_stub\dtypes.py:550: Fut
ureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be underst
ood as (type, (1,)) / '(1,)type'.
  np_resource = np.dtype([("resource", np.ubyte, 1)])
```

```
In [2]: model=load_model('signlanguage.h5')
```

```
end.py:3976: The name tf.nn.max_pool is deprecated. Please use tf.nn.max_pool2d instead.

WARNING:tensorflow:From c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\keras\backend\tensorflow_back
end.py:174: The name tf.get_default_session is deprecated. Please use tf.compat.v1.get_default_session instead.

WARNING:tensorflow:From c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\keras\backend\tensorflow_back
end.py:181: The name tf.ConfigProto is deprecated. Please use tf.compat.v1.ConfigProto instead.

WARNING:tensorflow:From c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\keras\backend\tensorflow_back
end.py:186: The name tf.Session is deprecated. Please use tf.compat.v1.Session instead.

WARNING:tensorflow:From c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\keras\optimizers.py:790: The
name tf.train.Optimizer is deprecated. Please use tf.compat.v1.train.Optimizer instead.

WARNING:tensorflow:From c:\users\adars\appdata\local\programs\python\python37\lib\site-packages\tensorflow\python\ops\math_gr
ad.py:1250: add_dispatch_support.<locals>.wrapper (from tensorflow.python.ops.array_ops) is deprecated and will be removed in
a future version.
Instructions for updating:
Use tf.where in 2.0, which has the same broadcast rule as np.where
```

- Loading the Test Image, Pre-Processing it And Prediction

```
In [19]: 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)
             predictions = model.predict_classes(img)
             print(predictions)
```

```
In [21]: frame = cv2.imread(r"dataset/test_set/G/1.png")
         data = detect(frame)

         [[1.1529493e-09 1.6801257e-12 3.0758306e-07 3.6168924e-08 2.1814937e-11
           6.9361130e-09 9.9995184e-01 4.7746969e-05 3.6307211e-09]]
         [6]
```

➤ 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

- Loading the requiredpackages

```
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 TensorFlow. 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')
```

- Pre-processingthe frames capturedfrom 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)
```

- Video Feed call from the HTML PAGE

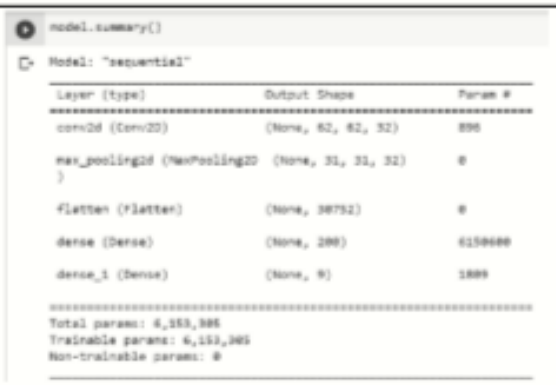
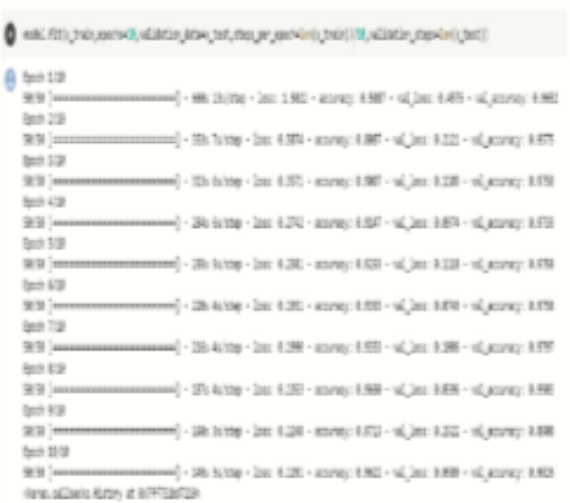
```
@app.route('/video_feed')
def video_feed():
    return Response(gen(),mimetype='multipart/x-mixed-replace; boundary=frame')

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

7.5.HTML PAGE

- HTML page to display the processedvideo on the screen, so that the person can show signs which canbe detected.

```
<!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>Hearuu</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 Black"><strong> Real-Time Communicatio
                System Powered By AI For Specially Abled - Team ID: PNT2022TMID09233</strong></h4></a>
            <div></div>
        </div>
    </nav>

            <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">


<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"
                            data-bs-toggle="collapse" data-bs-target="#accordion-1 .item-2" aria-expanded="false"
                            aria-controls="accordion-1 .item-2"
                            style="font-style: oblique; background: #c7931c;color: #ffffff;">Project: Real-Time Communication System Powered by AI for Specially Abled</button></h2>
                    <div class="accordion-collapse collapse item-2" role="tabpanel" data-bs-parent="#accordion-1">
                        <div Class="accordion-body">
                            <p class="mb-0">Team ID: PNT2022TMID09233<br><br>Hearuu Developed By,<br><br>1. <strong>Abubucker Siddique</strong> 310619104001<br>2.
                                <strong>Adarsh Srinivas</strong> 310619104003<br>3. <strong>Ashwath</strong> 310619104012<br>4. <strong>Jayasurya</strong> 310619104044
                            </p>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </section>
    <div class="modal fade" role="dialog" tabindex="-1" id="modal-1">
        <div class="modal-dialog" role="document">
            <div class="modal-content">
                <div class="modal-header">
                    <h4 class="modal-title">American Sign Language - Alphabets</h4><button type="button"
                        class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
                </div>
                <div class="modal-body"><img src="{{ url_for('static', filename='img/ASL_Alphabets.png') }}" width="100%"></div>
                <div class="modal-footer"><button class="btn btn-secondary" type="button"
                        data-bs-dismiss="modal">Close</button></div>
            </div>
        </div>
    </div>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

8.Testing:

S.No.	Parameter	Values	Screenshot
1.	Model Summary	Model - Sequential model Layers: Conv2D-(None,62,62,32) MaxPooling2D- (None,31,31,32) Flatten-(None,30752) Dense-(None,200) Dense_1 -(None,9)	
2.	Accuracy	Training Accuracy - 0.9622 Validation Accuracy -0.9826	
3	Confidence Score (Only Yolo Projects)	Class Detected - NA Confidence Score -NA	NA

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of project-Real Time Communication System Powered By AI For Specially Abled at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

Section	Total Cases	Not Tested	Fail	Pass
View Home Page	7	0	1	6
Click Reference	15	0	3	12
Image displayed	12	0	0	12
Allow camera access	11	0	2	9

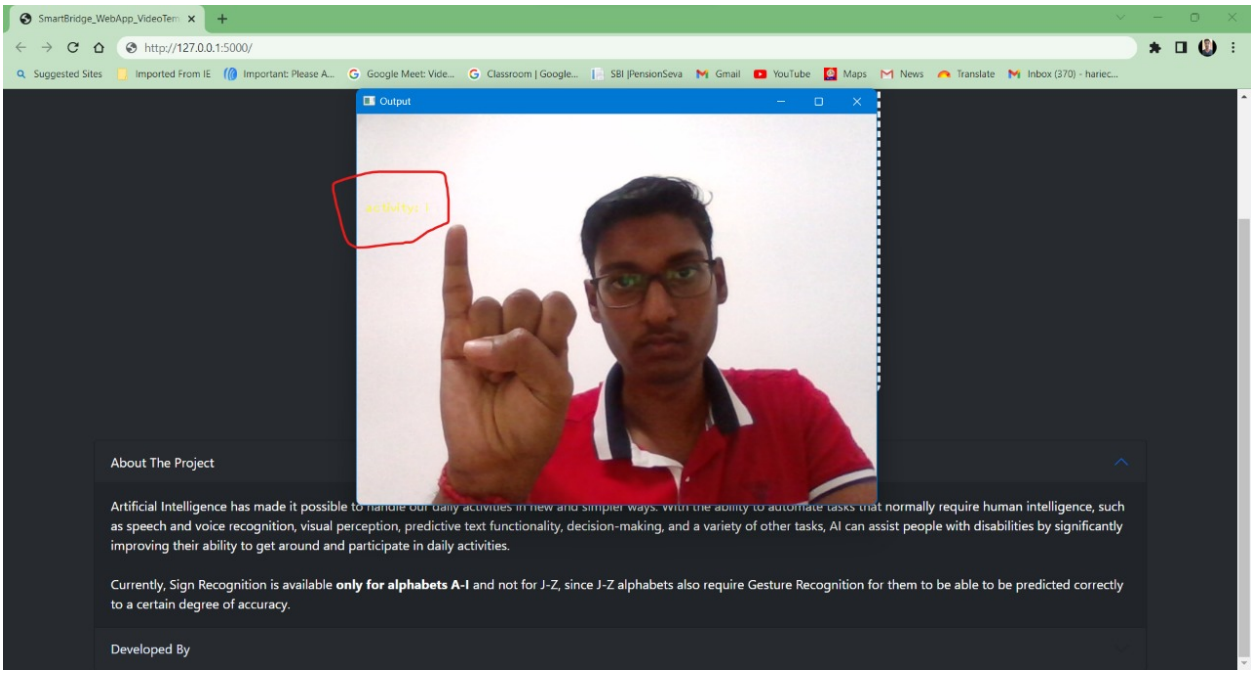
This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	0	0	0	2	2
Duplicate	1	0	0	0	1
External	0	0	1	0	1
Fixed	0	1	1	0	2
Not Reproduced	0	1	0	0	1
Skipped	0	0	0	0	0
Won't Fix	0	1	0	0	1
Totals	1	3	2	2	8

3. Test Case Analysis

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

Gesture prediction	9	0	2	7
Final Output	9	0	2	7
Version Control	5	0	2	3



9.Results

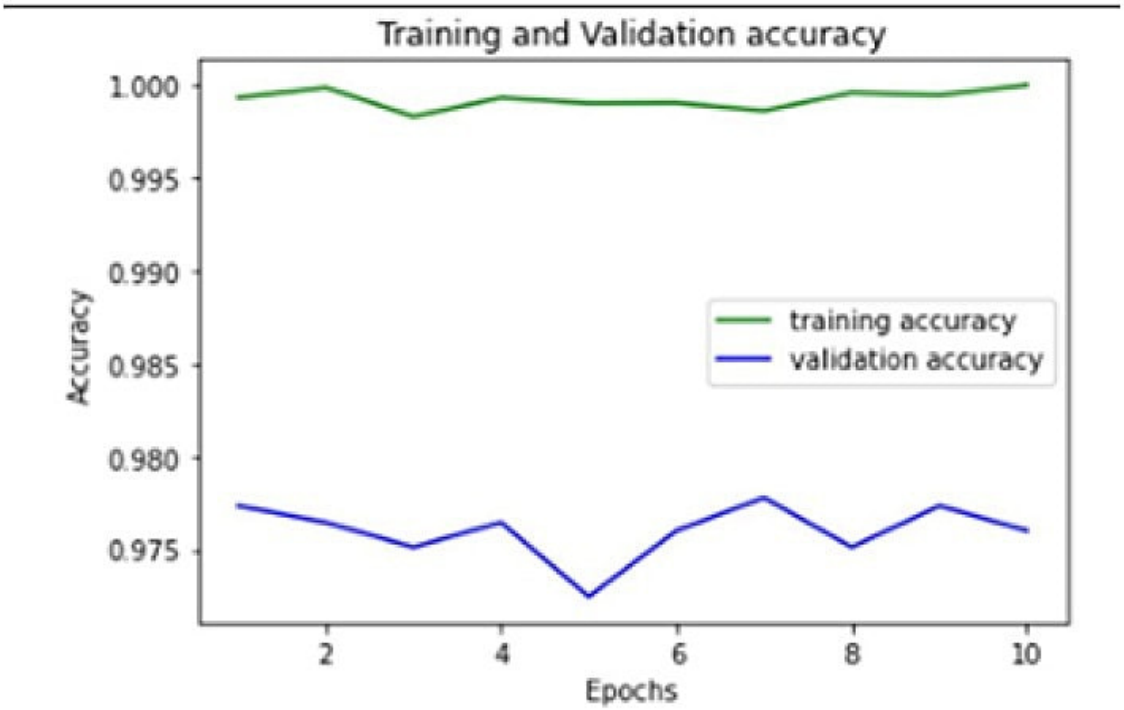
9.1:Performance Metrics:

Model summary

```
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 62, 62, 32)        896

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

 flatten (Flatten)           (None, 30752)             0

 dense (Dense)               (None, 200)               6150600

 dense_1 (Dense)             (None, 9)                 1809

=================================================================
Total params: 6,153,305
Trainable params: 6,153,305
Non-trainable params: 0
_________________________________________________________________
```

Accuracy



10.ADVANTAGES & DISADVANTAGES

- **Advantages**

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

- **Disadvantages**

- The dataset in limited. The alphabets only range from 'A' to 'J'.
- As of now, only static gestures 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 gestureimages, we used a traineddataset for the CNN algorithm. These features combinedwith 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, makingthe system more convenient and portable. This system is unable to detect gestures made with both hands. Therefore, detecting gesturesdone 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.

- GitHub: https://github.com/IBM-EPBL/IBM-Project-2282-1658469092

- Project Demo Link: https://drive.google.com/file/d/1QiXsJGflEszoXGfV_hbrIgFpV4DI92Qr/view?usp=share_link