**1.INTRODUCTION**

**1.1 Project Overview**

Today, artificial intelligence tools are extensively used to solve the problems in communication systems. The reason behind this is twofold. On one hand, existing communication systems are dramatically big and complex, and the traditional mathematical tools sometimes cannot serve them well. For example, the available tools of graph theory cannot address the dynamic characteristics of the Internet or social networks. On the other hand, the artificial intelligence algorithms are developed significantly due to the availability of big datasets [1]. It is natural that AI methods are employed to solve problems where model based methods are awkward [2]. Moreover, researchers have realized that AI applications need the strong support of the underneath layers due to the distributed nature of big data sets [3]. In other words, the marriage between AI and communication is a natural choice.

**1.2 Purpose**

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.

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**

**2.1 Existing Problem**

Current communication systems have become extraordinarily large and complex in this big data age. As a result, the traditional mathematical tools are exhaustively serving the demands from unprecedented applications. We mainly depend on artificial intelligence to address the problems. However, AI algorithms are usually black-box based, lack interpretability, and therefore cannot be used in critical and expensive cases. We are now in the middle of many unknowns. In this article, we share our belief that mathematical model guided AI is the solution for the challenges we are facing. We also present our preliminary study on a few key parts, hoping it is helpful for energetic readers to further explore the promising uncharted fields.

**2.2 Reference**

M. Jordan and T. Mitchell, "Machine learning: trends, perspectives, and prospects," Science, vol. 349, no. 6245, pp. 255–260, 2015.

Y. Sun et al., "Application of machine learning in wireless networks: key techniques and open issues," IEEE Commun. Surveys and Tutorials, in press, 2019.

S. Yu et al., "Networking for Big Data: a survey," IEEE Commun. Surveys and Tutorials, vol. 19, no. 1, pp. 531-549, 2017.

M. Hutson, "Has artificial intelligence become alchemy?," Science, vol. 360, no. 6388, 2018.

G. Box, "Science and Statistics," J. of the American Statistical Association, vol. 71, no. 356, pp. 791-799, 1976.

P. V. Mieghem, Graph Spectra for Complex Networks. Cambridge University Press, 2011.

A. Shwartz, A. Weiss, and R. Vanderbei, Large Deviations for Performance Analysis. Taylor and Francis Group, 2018.

**2.3 Problem Statement Definition**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.NO** | **TITLE** | **AUTHORS** | **DESCRIPTION** |
| 1. | Two Hand Indian Sign Language dataset for benchmarking classification models of Machine Learning | Leela Surya Teja Mangamuri, Lakshay Jain Abhishek Sharmay | This datasetwas benchmarked on six different classification modelsof machine learning by changing the parameters.Classification modelsare evaluated based on the HOG features extracted from the skin filtered image. An overall accuracy of 91.72% was achieved comprising of all machine learning models |
| 2. | Double Handed Indian Sign Language to Speech and Text | Kusurnika KroriDutta , Satheesh Kumar Raju, Anil Kumar G , Sunny ArokiaSwarny | The system is trained with double handed sign language by using a minimum eigenvalue algorithm. Here Logitech web camera is used for image acquisition and processing is performed in MATLAB. |

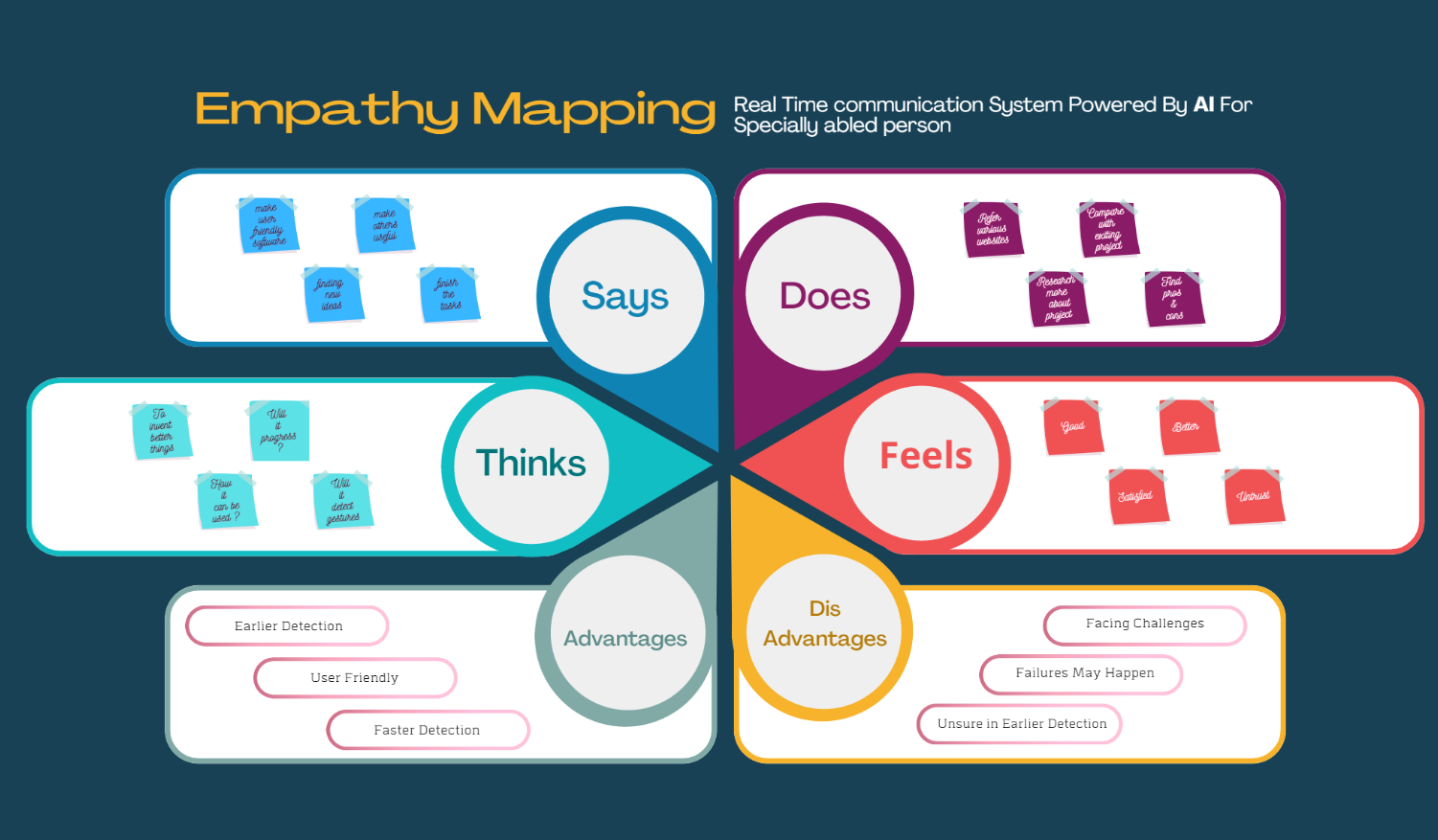
|  |  |  |  |
| --- | --- | --- | --- |
| 3. | Innovative studyof an AI voice  based smart deviceto assist deafpeople | Dhaya Sindhu  Battina | Development of an  artificial intelligence voice-based smart device that include the Flex sensors, LCD module, microcontroller, SD card memory, hearing phones, etc.improves the quality of life without the assistance of some artificial means.. |
| 4. | Communication systemfor deaf  and dumbpeople | Shraddha R.  Ghorpade, Prof. Surendra K. Waghmare | The human hand  comprises of numerous associated parts and joints,making it a complex object for input. The majority of the sign make utilization of both the hands together. Using webcam, capture the image of the hand to be tested.And convert the captured RGB image intoHSV image and then into binary image. The edges are detected using canny edge detection. |
| 5. | K-Nearest Correlated Neighbor  Classification for Indian Sign Language GestureRecognition using Feature Fusion | Bhumika Gupta,  Pushkar Shukla,Ankush Mittal | INDIA proposed that  recognition of gesture of Indian Sign Languages using static images where a test imageis first categorized into a single or double handed gesture followed by its classification using a fusion of SIFT and HOG descriptors via K- NearestCorrelated Neighbours |
| 6. | Machine Learning Techniques forIndian Sign Language Recognition | Kusumika Krori Dutta,  Sunny Arokia Swamy Bellary | Over the years, communication has  played a vital role in exchange of information and feelings in one's life. Sign language is the only medium through which specially abled people can connect to rest of theworld through different hand gestures. With the advances in machine |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | learning techniques, Hand  gesture recognition (HGR) became a very important research topic. |
| 7. | Comprehensive SVM based Indian  Sign Language Recognition | K. Revanth, N. Sri  Madhava Raja | Different machine  learning algorithms have been applied and SVM has achieved good result and comparison of different algorithm has been taken place. The classifiers used in this workflow comparison are Support Vector Machines, K – Nearest Neighbour, Logistic Regression and NaïveBayes. The selected parameter for Observations are accuracy, precision, fl score and recall. They are calculated with the inbuilt SK learn metric tool that is especially designed to calculate values for the machinelearning model. |
| 8. | Indian Sign Language Animation  Generation System | Sandeep Kaur,  Maninder Singh | This paper describes a  system which generates HamNoSys corresponding to 100 words. These Notations are generated according to the Indian Sign Language. This system covers all the simple words to generate HamNoSys. This system has been tested on 100 words and results of  the systemare very encouraging. |

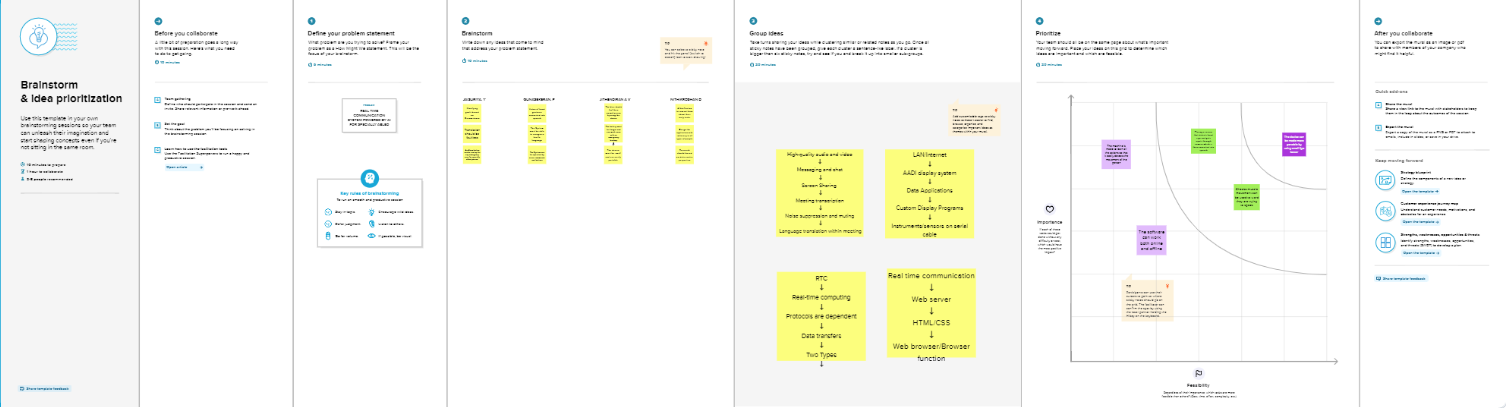
|  |  |  |  |
| --- | --- | --- | --- |
| 9. | Educational Statusof Differently  Abled Persons and Developed Policies in India | Chiranjit  Majumder | Education is utmost  significant to lift up the socio-economic status of PCPs. But education of disabled persons has not received adequate intentness and resources that it requires. Physically Challenged Persons (PCPs), few who are enrolled in schools are not given equal opportunity for middle secondary and highereducation levels.  Many Disabled persons are educated but they do not get any work for earningin our society. |
| 10. | Moment Based Sign Language Recognition For Indian Languages | Umang Patel, Aarti  G. Ambekar | Processed image, next step is feature extraction  & followed by classifier, recognized gestures are displayed as Hindi & English text & played as Hindi & English audio. |

**3.IDEATION & PROPOSED SOLUTION**

**3.1 Empathy Map Canvas**



**3.2 Ideation & Brainstorming**



**3.3 Proposed Solution**

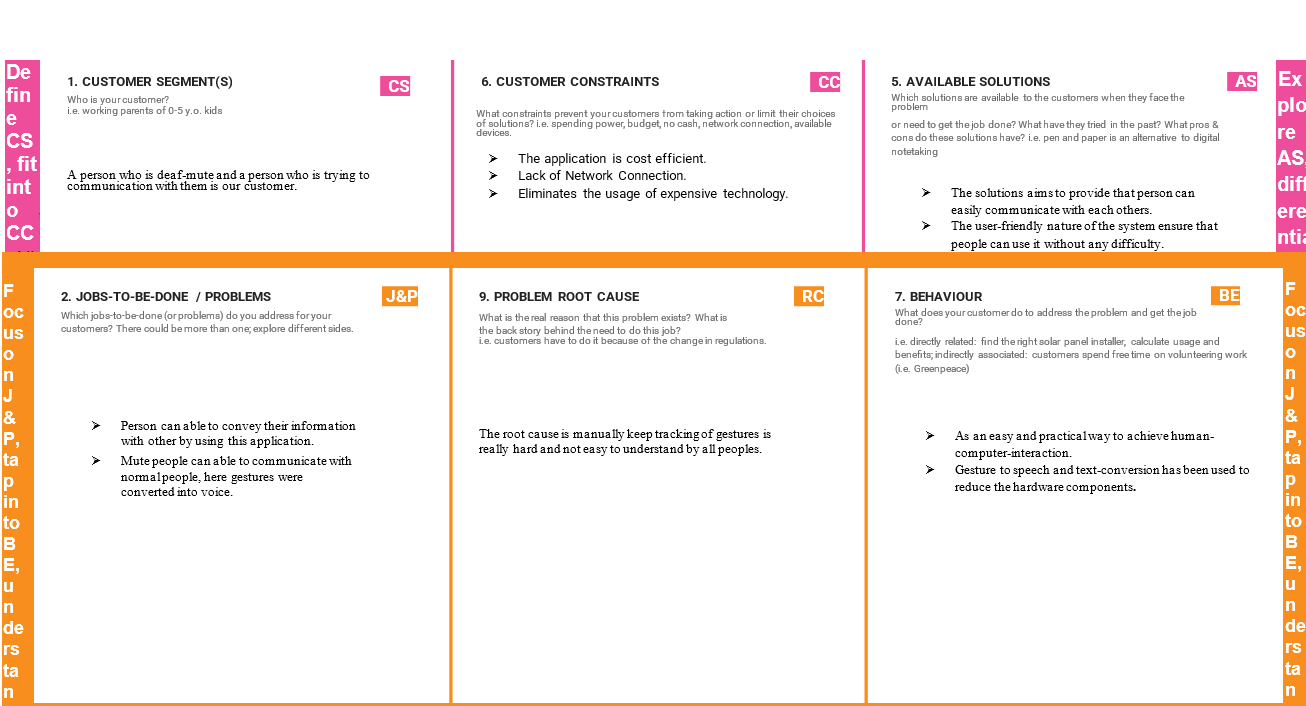
|  |  |  |  |
| --- | --- | --- | --- |
| **S.NO** | **TITLE** | **AUTHORS** | **DESCRIPTION** |
| 1. | Two Hand Indian Sign Language dataset for benchmarking classification models of Machine Learning | Leela Surya Teja Mangamuri, Lakshay Jain Abhishek Sharmay | This datasetwas benchmarked on six different classification modelsof machine learning by changing the parameters.Classification modelsare evaluated based on the HOG features extracted from the skin filtered image. An overall accuracy of 91.72% was achieved comprising of all machine learning models |
| 2. | Double Handed Indian Sign Language to Speech and Text | Kusurnika KroriDutta , Satheesh Kumar Raju, Anil Kumar G , Sunny ArokiaSwarny | The system is trained with double handed sign language by using a minimum eigenvalue algorithm. Here Logitech web camera is used for image acquisition and processing is performed in MATLAB. |

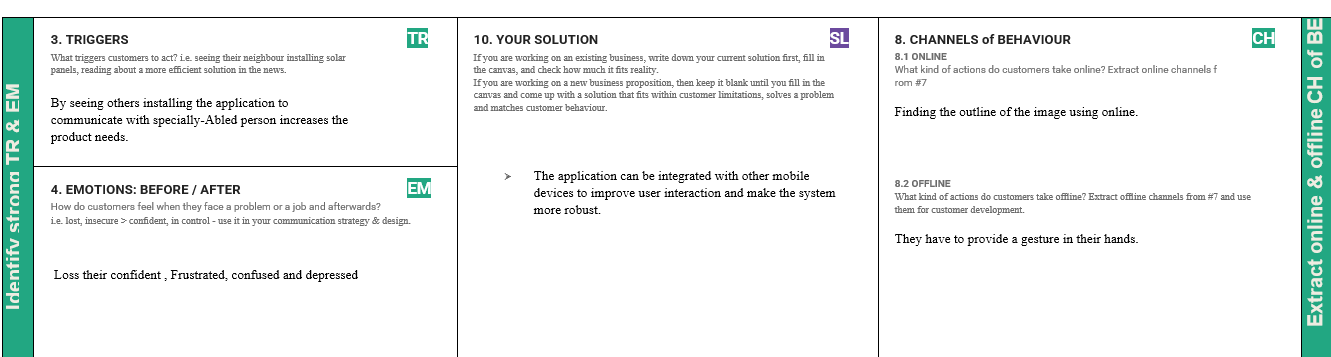
|  |  |  |  |
| --- | --- | --- | --- |
| 3. | Innovative studyof an AI voice  based smart deviceto assist deafpeople | Dhaya Sindhu  Battina | Development of an  artificial intelligence voice-based smart device that include the Flex sensors, LCD module, microcontroller, SD card memory, hearing phones, etc.improves the quality of life without the assistance of some artificial means.. |
| 4. | Communication systemfor deaf  and dumbpeople | Shraddha R.  Ghorpade, Prof. Surendra K. Waghmare | The human hand  comprises of numerous associated parts and joints,making it a complex object for input. The majority of the sign make utilization of both the hands together. Using webcam, capture the image of the hand to be tested.And convert the captured RGB image intoHSV image and then into binary image. The edges are detected using canny edge detection. |
| 5. | K-Nearest Correlated Neighbor  Classification for Indian Sign Language GestureRecognition using Feature Fusion | Bhumika Gupta,  Pushkar Shukla,Ankush Mittal | INDIA proposed that  recognition of gesture of Indian Sign Languages using static images where a test imageis first categorized into a single or double handed gesture followed by its classification using a fusion of SIFT and HOG descriptors via K- NearestCorrelated Neighbours |
| 6. | Machine Learning Techniques forIndian Sign Language Recognition | Kusumika Krori Dutta,  Sunny Arokia Swamy Bellary | Over the years, communication has  played a vital role in exchange of information and feelings in one's life. Sign language is the only medium through which specially abled people can connect to rest of theworld through different hand gestures. With the advances in machine |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | learning techniques, Hand  gesture recognition (HGR) became a very important research topic. |
| 7. | Comprehensive SVM based Indian  Sign Language Recognition | K. Revanth, N. Sri  Madhava Raja | Different machine  learning algorithms have been applied and SVM has achieved good result and comparison of different algorithm has been taken place. The classifiers used in this workflow comparison are Support Vector Machines, K – Nearest Neighbour, Logistic Regression and NaïveBayes. The selected parameter for Observations are accuracy, precision, fl score and recall. They are calculated with the inbuilt SK learn metric tool that is especially designed to calculate values for the machinelearning model. |
| 8. | Indian Sign Language Animation  Generation System | Sandeep Kaur,  Maninder Singh | This paper describes a  system which generates HamNoSys corresponding to 100 words. These Notations are generated according to the Indian Sign Language. This system covers all the simple words to generate HamNoSys. This system has been tested on 100 words and results of  the systemare very encouraging. |

|  |  |  |  |
| --- | --- | --- | --- |
| 9. | Educational Statusof Differently  Abled Persons and Developed Policies in India | Chiranjit  Majumder | Education is utmost  significant to lift up the socio-economic status of PCPs. But education of disabled persons has not received adequate intentness and resources that it requires. Physically Challenged Persons (PCPs), few who are enrolled in schools are not given equal opportunity for middle secondary and highereducation levels.  Many Disabled persons are educated but they do not get any work for earningin our society. |
| 10. | Moment Based Sign Language Recognition For Indian Languages | Umang Patel, Aarti  G. Ambekar | Processed image, next step is feature extraction  & followed by classifier, recognized gestures are displayed as Hindi & English text & played as Hindi & English audio. |

**3.4 Problem Solution Fit**





**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 Registration | Registration through Form Registration through Gmail  Registration through LinkedIN |
| FR-2 | User Confirmation | Confirmation via EmailConfirmation via OTP |
| FR-3 | system | Desktop with highresolution camera |
| FR-4 | External interface | Ethernet, Wi-Fi, USB to provide internet facility to accessthe resources with real time communication. |
| FR-5 | Authorization Levels | There are two levels of authorization namelystandard  access level andadvanced access level. |
| FR-6 | Reporting | If any issuesfound in the application, automatically it  will be notified to the developer. |

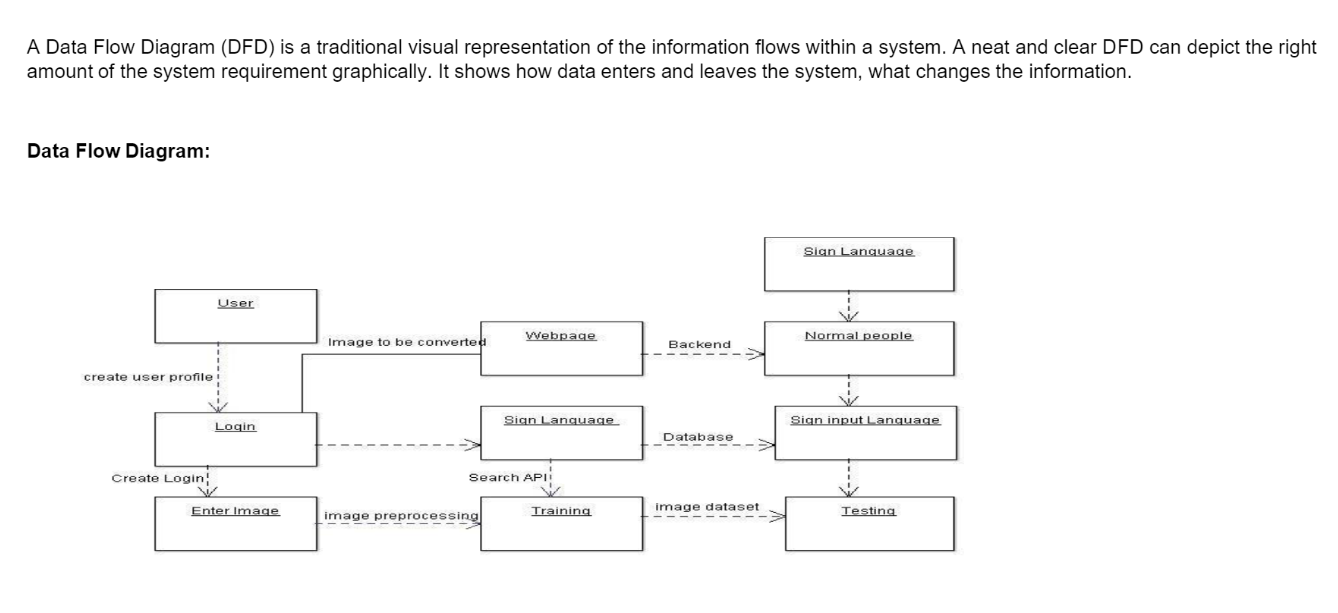
**4.2 Non- Functional requirement**

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

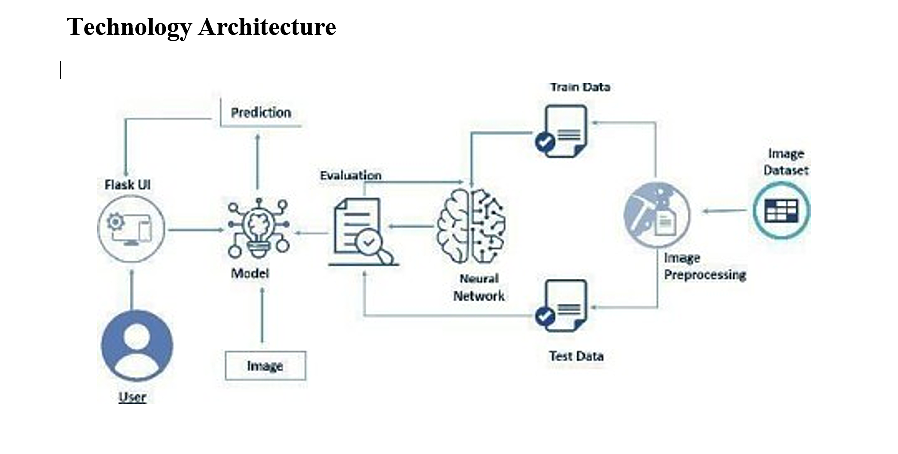
|  |  |  |
| --- | --- | --- |
| **FR No.** | **Non-Functional Requirement** | **Description** |
| NFR-1 | **Usability** | To convey a message to normal people, as well as convert speechinto understandable signlanguage  for the deaf and dumb people. |
| NFR-2 | **Security** | To convey a message to normal people, as well as convert speechinto understandable signlanguage  for the deaf and dumb people. |
| NFR-3 | **Reliability** | Provides insight into potential issuesfor desktop  applications on managed devices. |
| NFR-4 | **Performance** | The time for converting signsinto speech shouldbe faster forthe real timecommunication. |
| NFR-5 | **Availability** | The time for converting signs into speechshould be  faster for the real time communication. |
| NFR-6 | **Scalability** | This app enables deaf and dumb people to convey their information using signs whichget converted to human-understandable language and speech is  given as output. |

**5.PROJECT DESIGN**

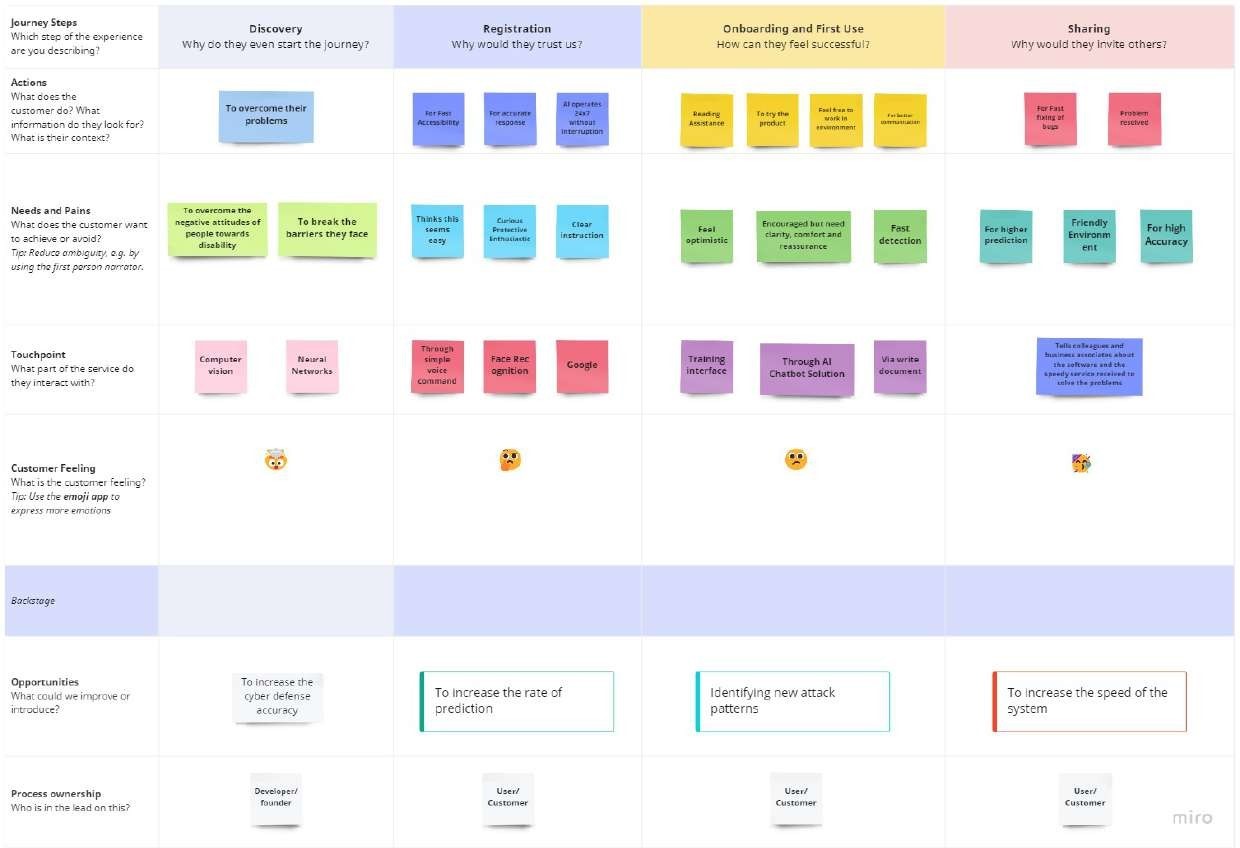
**5.1 Data Flow Diagrams**



**5.2 Solution & Technical Architecture**

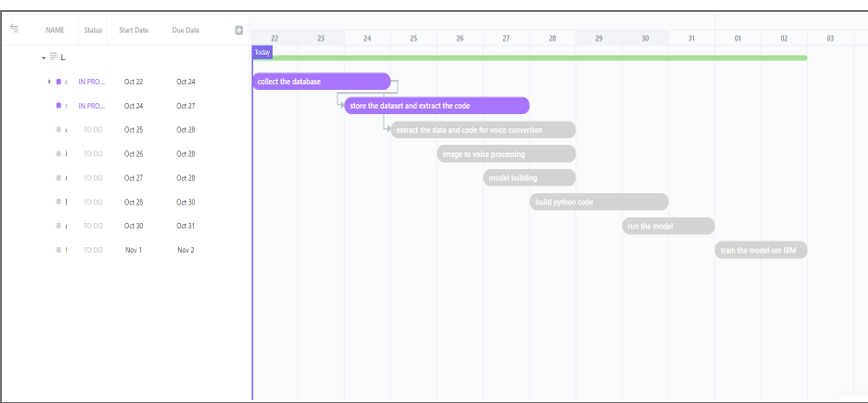


**5.3 User Stories**



**6. PROJECT PLANNING & SCHEDULING**

**6.1 Sprint Planning & Estimation**



**6.2 Sprint Delivery Schedule**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sprint** | **Functional**  **Requirement (Epic)** | **User Story**  **Number** | **User Story / Task** | **Story**  **Points** | **Priority** | **Team Members** |
| Sprint-1 | Data Collection |  | Dataset is collected on the basisof various handsigns and curated according to the problem statement. | 4 | High | Keerthana, Jeya Carolin Agnes,Sounthariyaa, Sneka Nayaki |
| Sprint-1 | Data Preprocessing |  | The dataset is preprocessed in order to check noisy data and other inconsistencies before executing it to thealgorithm. | 6 | Low | Keerthana, Sounthariyaa, |
| Sprint-2 | Model Building |  | Model is built according to the image features in sucha way that the modelidentifies the features of the sign image and learns in order to give correct output. | 8 | Medium | Keerthana, Jeya Carolin Agnes |
| Sprint-2 | Model Training |  | Data is fed into the model and the model is trained in orderto find the optimal weightsthat give help in predicting the correct output. | 8 | High | Keerthana, Sneka Nayaki |
| Sprint-2 | Model Testing |  | Model is testedin such a way that the collection data or images are trained frame by frame according to the user requirements. | 6 | High | Keerthana, Sneka Nayaki |
| Sprint-3 | User Registration | USN-1 | As a user, I need to register and create a login using my credentials. Oncecreated , I can logininto the application and access the site using mobile/Desktop. | 8 | Medium | Keerthana, Jeya Carolin Agnes |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sprint** | **Functional**  **Requirement (Epic)** | **User Story**  **Number** | **User Story / Task** | **Story**  **Points** | **Priority** | **Team Members** |
| Sprint-3 | Sign Capture/ Text input | USN-2 | I can see an option to start capturing the video usingmy camera, so that the signs are converted into text or audio | 8 | High | Keerthana, Sneka Nayaki |
| USN-3 | As a user, there’s an option to convert thetext or audiothat i provideinto signs | 4 | Low | Keerthana, Sounthariyaa, |
| Sprint-4 | Converted Message/Sign | USN-4 | The signsi capture usingmy camera areconverted to englishtext or audio | 8 | Medium | Keerthana, Jeya Carolin Agnes |
| Sprint-4 | Application Testing |  | The application is then tested under various circumstances. | 8 | High | Keerthana, Sneka Nayaki |

**Project Tracker,Velocity & BurndownChart: (4 Marks)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sprint** | **Total Story Points** | **Duration** | **Sprint Start Date** | **Sprint End Date (Planned)** | **Story Points Completed (as on Planned End Date)** | **Sprint Release Date (Actual)** |
| Sprint-1 | 20 | 6 Days | 24 Oct 2022 | 29 Oct 2022 | 10 | 02 Nov 2022 |
| Sprint-2 | 20 | 6 Days | 31 Oct 2022 | 05 Nov 2022 | 22 |  |
| Sprint-3 | 20 | 6 Days | 07 Nov 2022 | 12 Nov 2022 | 20 |  |
| Sprint-4 | 20 | 6 Days | 14 Nov 2022 | 19 Nov 2022 | 16 |  |

**Velocity:**

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

|  |  |
| --- | --- |
|  |  |
|  |  |

**Burndown Chart:**

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile [software development](https://www.visual-paradigm.com/scrum/what-is-agile-software-development/) methodologies such as [Scrum](https://www.visual-paradigm.com/scrum/scrum-in-3-minutes/). However, burn down chartscan be applied to any project containing measurable progress over time.

https://www.visual-paradigm.com/scrum/scrum-burndown-chart/ https://www.atlassian.com/agile/tutorials/burndown-charts

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

**7.1 Feature 1**

<!DOCTYPE html>

<html lang="en">

<head>

<!--Sets up stylesheet and script file linking. Adds favicon and site title as well-->

<meta charset="UTF-8">

<link rel="shortcut icon" type="image/png" href="Images/asl\_logo.ico" />

<title>Sign Translator | Sufiyaan</title>

<link href="https://fonts.googleapis.com/css?family=Google+Sans:100,300,400,500" rel="stylesheet">

<link rel="stylesheet" href="CSS/style.css">

<link rel="stylesheet" href="CSS/animate.css">

<script src="dist/build.js"></script>

</head>

<body>

<!--Welcome Screen-->

<div id="welcomeContainer" class="animated">

<div id="welcomeScreen">

<h1>Welcome to Sign Language Translator</h1>

<h3 class="animated fadeInUp delay-2s">Artificial Intellegence</h3>

</div>

<button id="proceedButton">Proceed</button>

</div>

<!--Title bar explains the stage of the program

(eg: instructions for Training, Prediction, and Video Chat)-->

<div id="titleBar">

<h1 id="stage">Train Gestures</h1>

<h3 id="steps">Train about 30 samples of your Start Gesture and 30 for your idle, Stop Gesture.</h3>

<button id="nextButton" class="animated flash delay-3s">Next</button>

<button id="predictButton" class="animated flash slideInRight faster">Translate</button>

<button id="backButton" class="animated slideInLeft faster">Back to Training</button>

<button id="videoCallBtn" class="videoCallBtn animated slideInRight faster">Video Call</button>

</div>

<!--Status bar shows the status of translation-->

<div id="status">

<p id="status-text">Status: Not Ready</p>

</div>

<!--The Translator Window displays the video of the user, various buttons, and holds

the training and translation screens-->

<div id="translatorWindow">

<!--Initial Training Holder creates the screen where users train Start and Stop Gestures-->

<div id="initialTrainingHolder">

<img src="" alt="checkmark" id="checkmark\_startButton" class="checkMark">

<button id="startButton" class="trainButton ">Start Gesture</button>

<button id="clear\_startButton" class="clearButton">Clear</button>

<h3 id="counter\_startButton" class="counter"></h3>

<button id="stopButton" class="trainButton ">Stop Gesture</button>

<button id="clear\_stopButton" class="clearButton">Clear</button>

<h3 id="counter\_stopButton" class="counter"></h3>

<img src="" alt="checkmark" id="checkmark\_stopButton" class="checkMark">

</div>

<!--Video Holder displays the video feed from the user for training and translation.

It also displays the Video Call feed once the user decides to do that-->

<div id="videoHolder" class="videoContainerTrain">

<video id="video" class="videoTrain" src='' muted autoplay playsinline></video>

<iframe src="https://tokbox.com/embed/embed/ot-embed.js?embedId=f37957b6-0f91-4fc5-90ce-f818cc85b5bf&room=DEFAULT\_ROOM&iframe=true"

width=650 height=370 allow="microphone; camera" id="videoCall"></iframe>

</div>

<!--Training Holder displays the screen where users add and train new gestures-->

<div id="trainingHolder">

<h5 id="add-gesture">Add Gesture</h5>

<img src="Images/plus\_sign.svg" alt="Plus Sign" id="plus\_sign" class="plus\_sign animated">

<form id="add-word" autocomplete="off">

<input type="text" id="new-word" placeholder="New Gesture Title">

<input type="submit" id="submit-word" value="Add Word &#9658;">

</form>

<button id="doneRetrain" class="doneRetrain">Done Retraining</button>

<div id="trainingCommands"></div>

</div>

<!--Translation Holder is where the Gesture Card and Text that was translated is displayed.-->

<div id="translationHolder">

<div id="translatedCard">

</div>

<h3 id="translationText">Start Signing!</h3>

</div>

</div>

<!--Trained Card Holder contains all the trained gestures' cards.

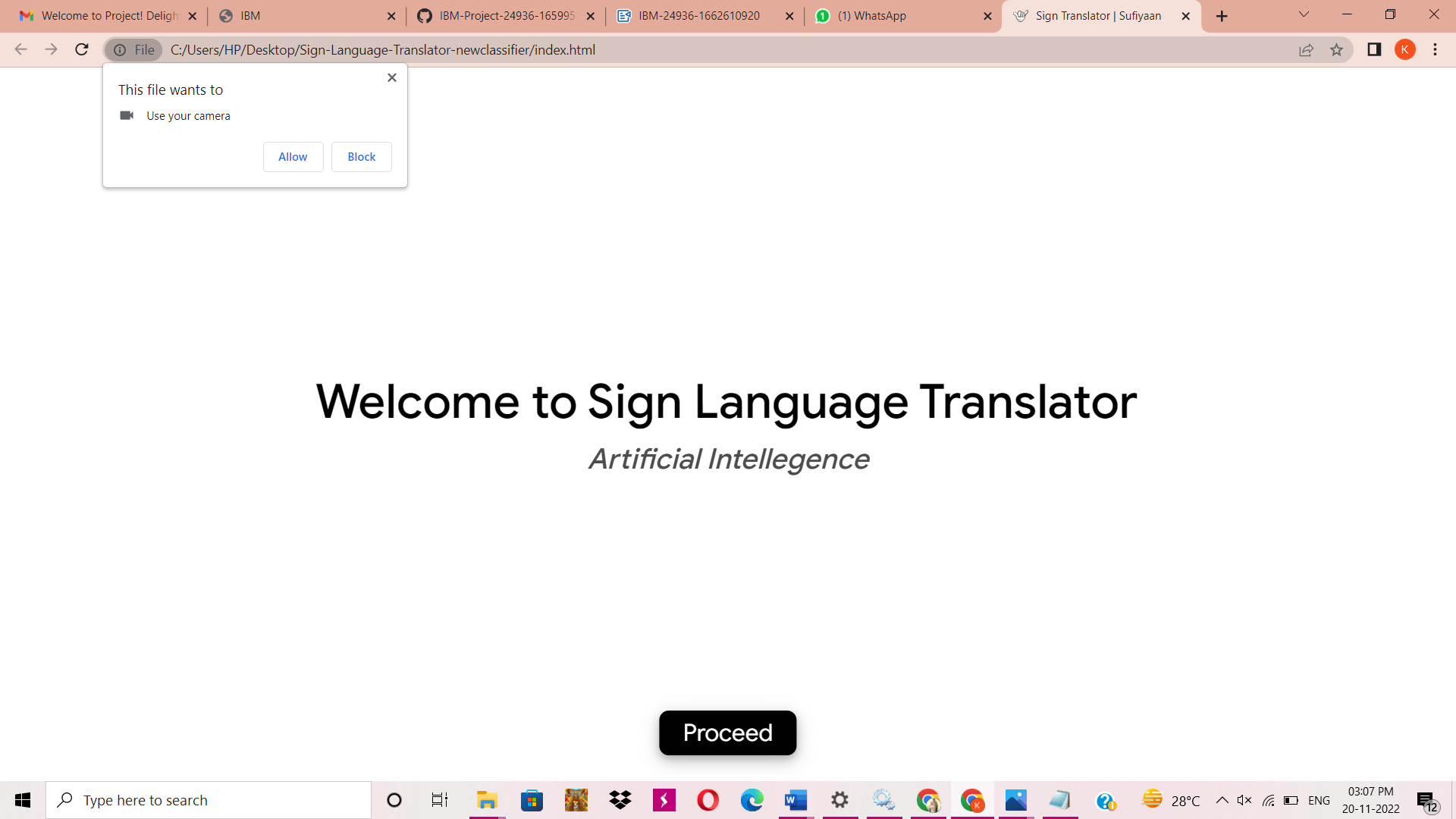
It is outside the Training and Translation Windows because it has to be displayed on both screens-->

<div id="trainedCardsHolder">

</div>

</body>

</html>



**7.2 Feature 2**

{

"name": "sign-language-translator",

"version": "0.0.1",

"license": "do what you want but try to give back",

"dependencies": {

"deeplearn": "~0.5.0",

"deeplearn-knn-image-classifier": "^0.3.0",

"pre-commit": "^1.2.2"

},

"scripts": {

"start": "budo main.js:dist/build.js --live --host localhost",

"build": "browserify main.js -o dist/build.js"

},

"pre-commit": [

"build"

],

"browserify": {

"transform": [

[

"babelify",

{

"presets": [

"es2015"

],

"plugins": [

"syntax-async-functions",

"transform-regenerator"

]

}

]

]

},

"devDependencies": {

"babel-core": "^6.26.0",

"babel-loader": "^7.1.2",

"babel-plugin-syntax-async-functions": "^6.13.0",

"babel-plugin-transform-regenerator": "^6.26.0",

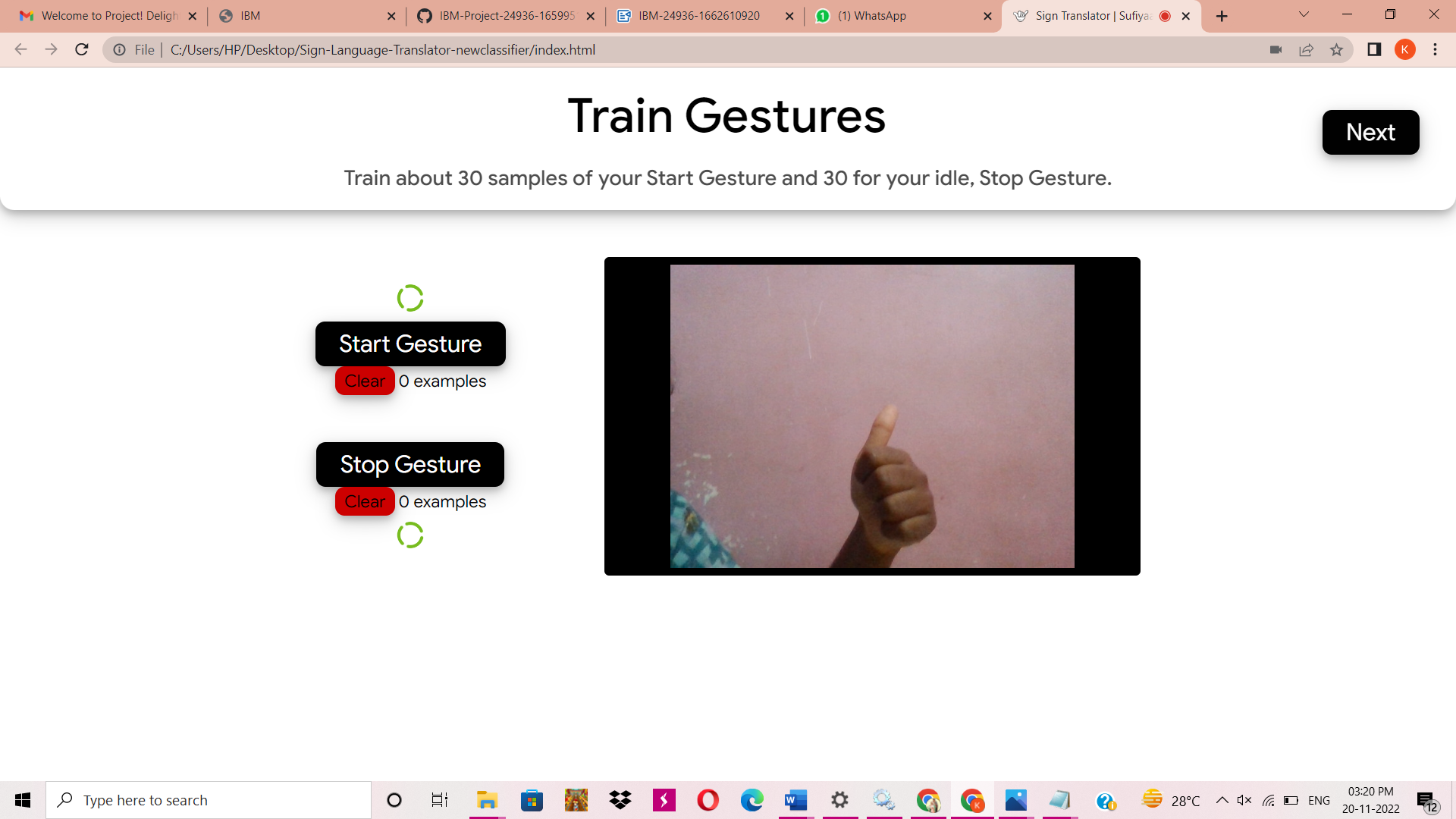
"babel-preset-es2015": "^6.24.1",

"babelify": "^8.0.0",

"budo": "^10.0.4"

}

}



**8.TESTING**

**8.1 Test Cases**

# **PROJECT** : Real-Time Communication system poweredby AI for specially abled

# Import The Required Model Building Libraries

#import imagedatagenerator

from keras.preprocessing.image importImageDataGenerator

#training datagen

train\_datagen=ImageDataGenerator(rescale=1./255,shear\_range=0.2,zoom\_range=0.2,horizontal\_

#testing datagen

test\_datagen=ImageDataGenerator(rescale=1./255)

# IMPORTING tensorflow

import tensorflow astf import os

# IMPORTING LIBRARIES TO INITIALIZE NEURAL NETWORK LAYER

from keras.models import Sequential fromkeras.layers import Dense

from keras.layers import Convolution2D from keras.layers import MaxPooling2D from keras.layers importDropout

from keras.layers import Flatten

from tensorflow.keras.preprocessing.image importImageDataGenerator

import numpy as np

import matplotlib.pyplot as plt #to view graph in colab itself import IPython.display as display

from PIL import Imageimport pathlib

# Unzipping the dataset

!unzip '/content/conversation engine for deaf and dumb.zip' inflating: Dataset/training\_set/I/947.png

inflating: Dataset/training\_set/I/948.png inflating: Dataset/training\_set/I/949.png inflating: Dataset/training\_set/I/95.png

inflating: Dataset/training\_set/I/950.png inflating: Dataset/training\_set/I/951.png inflating: Dataset/training\_set/I/952.png inflating: Dataset/training\_set/I/953.png inflating: Dataset/training\_set/I/954.png inflating: Dataset/training\_set/I/955.png inflating: Dataset/training\_set/I/956.png inflating: Dataset/training\_set/I/957.png inflating: Dataset/training\_set/I/958.png inflating: Dataset/training\_set/I/959.png inflating: Dataset/training\_set/I/96.png inflating: Dataset/training\_set/I/960.png inflating: Dataset/training\_set/I/961.png inflating: Dataset/training\_set/I/962.png inflating: Dataset/training\_set/I/963.png inflating: Dataset/training\_set/I/964.png inflating: Dataset/training\_set/I/965.png inflating: Dataset/training\_set/I/966.png inflating: Dataset/training\_set/I/967.png inflating: Dataset/training\_set/I/968.png inflating: Dataset/training\_set/I/969.png inflating: Dataset/training\_set/I/97.png inflating: Dataset/training\_set/I/970.png inflating: Dataset/training\_set/I/971.png inflating: Dataset/training\_set/I/972.png

extracting: Dataset/training\_set/I/973.png inflating: Dataset/training\_set/I/974.png inflating: Dataset/training\_set/I/975.png inflating: Dataset/training\_set/I/976.png inflating: Dataset/training\_set/I/977.png inflating: Dataset/training\_set/I/978.png inflating: Dataset/training\_set/I/979.png inflating: Dataset/training\_set/I/98.png inflating: Dataset/training\_set/I/980.png inflating: Dataset/training\_set/I/981.png inflating: Dataset/training\_set/I/982.png

extracting: Dataset/training\_set/I/983.png inflating: Dataset/training\_set/I/984.png inflating: Dataset/training\_set/I/985.png inflating: Dataset/training\_set/I/986.png inflating: Dataset/training\_set/I/987.png inflating: Dataset/training\_set/I/988.png inflating: Dataset/training\_set/I/989.png inflating: Dataset/training\_set/I/99.png inflating: Dataset/training\_set/I/990.png inflating: Dataset/training\_set/I/991.png inflating: Dataset/training\_set/I/992.png

extracting: Dataset/training\_set/I/993.png inflating: Dataset/training\_set/I/994.png inflating: Dataset/training\_set/I/995.png extracting: Dataset/training\_set/I/996.png inflating: Dataset/training\_set/I/997.png inflating: Dataset/training\_set/I/998.png inflating: Dataset/training\_set/I/999.png

# Applying ImageDataGenerator to training set

x\_train=train\_datagen.flow\_from\_directory('/content/Dataset/training\_set',target\_size=(64,

class\_mode='categorical',color\_mode="grayscale") Found 15750 images belonging to 9 classes.

# Applying ImageDataGenerator to test set

x\_test=test\_datagen.flow\_from\_directory('/content/Dataset/test\_set',target\_size=(64,64),ba

class\_mode='categorical',color\_mode="grayscale") Found 2250 images belonging to 9 classes.

a=len(x\_train) b=len(x\_test)

# Length of training set

print(a)

79

# Length of test set

print(b)

12

# AddLayers

#create model

model=Sequential()

# AddThe Convolution Layer

model.add(Convolution2D(32,(3,3),input\_shape=(64,64,1),activation='relu'))

# AddPooling Layer

model.add(MaxPooling2D(pool\_size=(2,2)))

**9.RESULTS**

**9.1 Performance Metrics**

This project thus gives the solution for the normal person to understand Symbol language of specially abled. This project capture gesture and then save the meaning of the respective gestures.

Once a saved gesture is shown infornt of the camera ,The App displaces the meaning of that gesture.

By this every normal person would understand the meaning of that particular gesture.Thus this App paves a way for the communication between specially abled and normal person.

**10. ADVANTAGES & DISADVANTAGES**

**Advantages**

This project brings many benefits to all specially abled people regardless of whether they are deaf or struggling with their hearing. As well as helping them to communicate and fully express themselves, it also improves their social skills by increasing their confidence and self-esteem.This project is user-friendly to handle.

**Disadvantages**

Sign language requires the use of hands to make gestures. This can be a problem for people who do not have full use of their hands. Even seemingly manageable disabilities such as Parkinson's or arthritis can be a major problem for people who must communicate using sign language.

**11. CONCLUSION**

Use Dynamic Loading for Dataset: Our original dataset was quite large and is impossible to use without a

server with a lot of RAM and disk space. A possible solution is to split the file names into training, validation,

and test sets and dynamically loading images in the Dataset class. Using such a loading technique would

allow us to train the model on more samples in the dataset.

**12. FUTURE SCOPE**

1. We can develop a model for ISL word and sentence level recognition. This will require a system that can detect changes with respect to the temporal space.

2. We can develop a complete product that will help the speech and hearing impaired people, and thereby reduce the communication gap.

**13. APPENDIX**

**index.html**

<!DOCTYPE html>

<html lang="en">

<head>

<!--Sets up stylesheet and script file linking. Adds favicon and site title as well-->

<meta charset="UTF-8">

<link rel="shortcut icon" type="image/png" href="Images/asl\_logo.ico" />

<title>Sign Translator | Sufiyaan</title>

<link href="https://fonts.googleapis.com/css?family=Google+Sans:100,300,400,500" rel="stylesheet">

<link rel="stylesheet" href="CSS/style.css">

<link rel="stylesheet" href="CSS/animate.css">

<script src="dist/build.js"></script>

</head>

<body>

<!--Welcome Screen-->

<div id="welcomeContainer" class="animated">

<div id="welcomeScreen">

<h1>Welcome to Sign Language Translator</h1>

<h3 class="animated fadeInUp delay-2s">Artificial Intellegence</h3>

</div>

<button id="proceedButton">Proceed</button>

</div>

<!--Title bar explains the stage of the program

(eg: instructions for Training, Prediction, and Video Chat)-->

<div id="titleBar">

<h1 id="stage">Train Gestures</h1>

<h3 id="steps">Train about 30 samples of your Start Gesture and 30 for your idle, Stop Gesture.</h3>

<button id="nextButton" class="animated flash delay-3s">Next</button>

<button id="predictButton" class="animated flash slideInRight faster">Translate</button>

<button id="backButton" class="animated slideInLeft faster">Back to Training</button>

<button id="videoCallBtn" class="videoCallBtn animated slideInRight faster">Video Call</button>

</div>

<!--Status bar shows the status of translation-->

<div id="status">

<p id="status-text">Status: Not Ready</p>

</div>

<!--The Translator Window displays the video of the user, various buttons, and holds

the training and translation screens-->

<div id="translatorWindow">

<!--Initial Training Holder creates the screen where users train Start and Stop Gestures-->

<div id="initialTrainingHolder">

<img src="" alt="checkmark" id="checkmark\_startButton" class="checkMark">

<button id="startButton" class="trainButton ">Start Gesture</button>

<button id="clear\_startButton" class="clearButton">Clear</button>

<h3 id="counter\_startButton" class="counter"></h3>

<button id="stopButton" class="trainButton ">Stop Gesture</button>

<button id="clear\_stopButton" class="clearButton">Clear</button>

<h3 id="counter\_stopButton" class="counter"></h3>

<img src="" alt="checkmark" id="checkmark\_stopButton" class="checkMark">

</div>

<!--Video Holder displays the video feed from the user for training and translation.

It also displays the Video Call feed once the user decides to do that-->

<div id="videoHolder" class="videoContainerTrain">

<video id="video" class="videoTrain" src='' muted autoplay playsinline></video>

<iframe src="https://tokbox.com/embed/embed/ot-embed.js?embedId=f37957b6-0f91-4fc5-90ce-f818cc85b5bf&room=DEFAULT\_ROOM&iframe=true"

width=650 height=370 allow="microphone; camera" id="videoCall"></iframe>

</div>

<!--Training Holder displays the screen where users add and train new gestures-->

<div id="trainingHolder">

<h5 id="add-gesture">Add Gesture</h5>

<img src="Images/plus\_sign.svg" alt="Plus Sign" id="plus\_sign" class="plus\_sign animated">

<form id="add-word" autocomplete="off">

<input type="text" id="new-word" placeholder="New Gesture Title">

<input type="submit" id="submit-word" value="Add Word &#9658;">

</form>

<button id="doneRetrain" class="doneRetrain">Done Retraining</button>

<div id="trainingCommands"></div>

</div>

<!--Translation Holder is where the Gesture Card and Text that was translated is displayed.-->

<div id="translationHolder">

<div id="translatedCard">

</div>

<h3 id="translationText">Start Signing!</h3>

</div>

</div>

<!--Trained Card Holder contains all the trained gestures' cards.

It is outside the Training and Translation Windows because it has to be displayed on both screens-->

<div id="trainedCardsHolder">

</div>

</body>

</html>

|  |  |
| --- | --- |
|  |  |
|  |  |