**PROJECT REPORT**

**REAL-TIME RIVER QUALITY MONITORING AND CONTROL SYSTEM**

**TEAM ID:** PNT2022TMID21004

**TEAM MEMBERS**

**SHIVARAMAN G 412419106079**

**ARAVINDAN S 412419106007**

**MOHAMMED ZAID J 412419106041**

**ARUNHACHALAM VR 412419106008**

**SATHESH KANNAN A 412419106075**

**INDEX**

**1.INTRODUCTION**

**1.1 Abstract**

**1.2 Objective**

**2.LITERATURE SURVEY**

**2.1 Problem Statement Definition**

**3.IDEATION & PROPOSED SOLUTION**

**3.1 Empathy Map Canvas**

**3.2 Ideation & Brainstorming**

**3.3 Proposed Solution**

**3.4 Problem Solution fit**

**4.REQUIREMENT ANALYSIS**

**4.1 Functional requirement**

**4.2 Non-Functional requirements**

**5.PROJECT DESIGN**

**5.1 Data Flow Diagrams**

**5.2 Solution & Technical Architecture**

**5.3 Customer Journey**

**6.PROJECT PLANNING & SCHEDULING**

**6.1 Sprint Planning & Estimation**

**6.2 Sprint Delivery Schedule**

**6.3 Reports from JIRA**

**7.CODING & SOLUTIONING**

**7.1 Feature 1**

**7.2 Feature 2**

**8.TESTING**

**8.1 Test Cases**

**8.2 User Acceptance Testing**

**9.RESULTS**

**9.1 Performance table**

**10.ADVANTAGES & DISADVANTAGES**

**11.CONCLUSION**

**12.FUTURE SCOPE**

**13.APPENDIX**

**13.1 Source Code**

**13.2 GitHub & Project Demo Link**

**1.INTRODUCTION**

**1.1 Abstract:**

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.

**1.2 Objective:**

Communication between deaf and dumb and a normal person has always been a difficult problem. We have used convolution neural network to create a model that is trained with different hand gestures. An is integrated with this. This app enables deaf and dumb people to convey their information using signs/ gestures which get converted to human-recognizable language and voice is given as output.

**2. LITERATURE SURVEY**

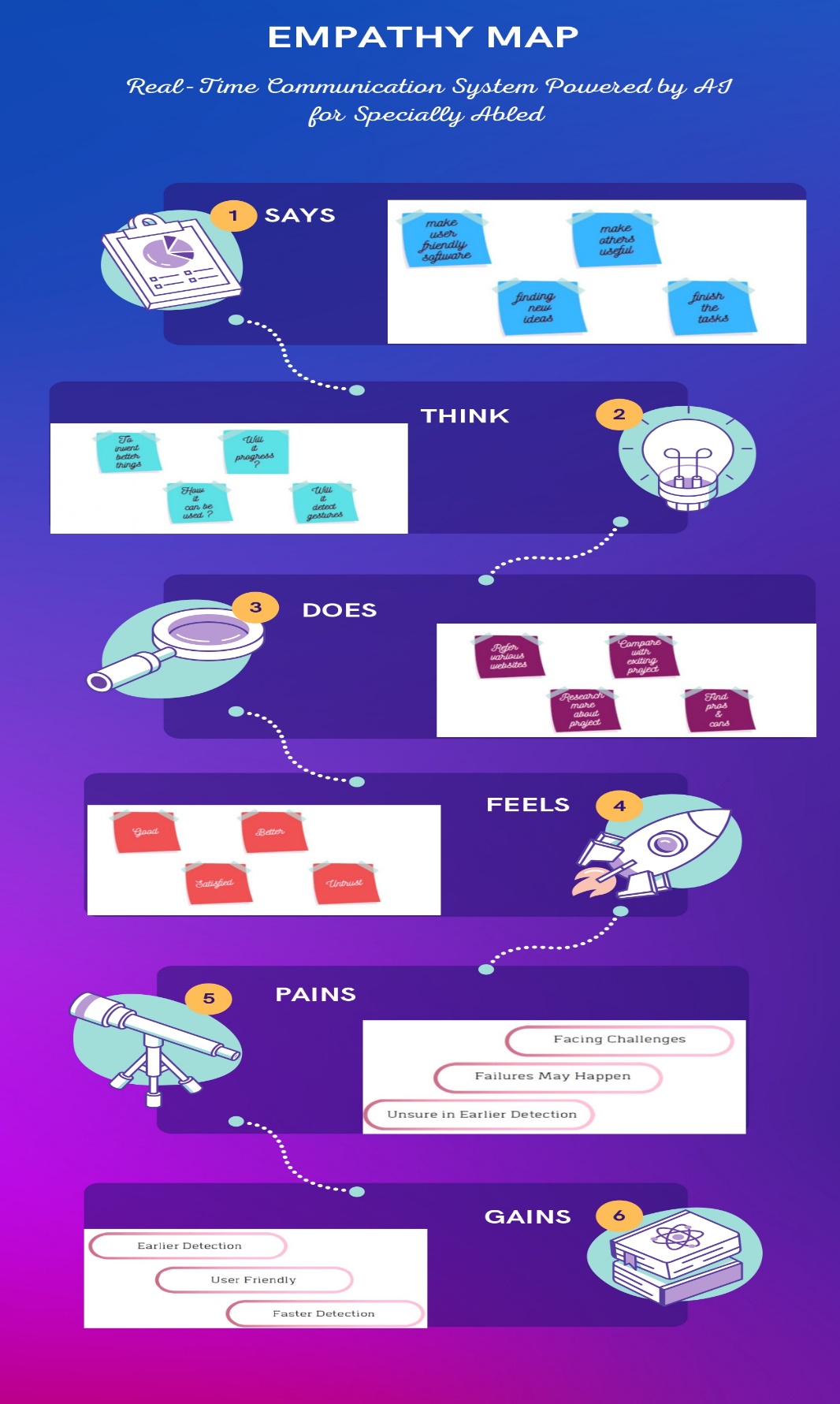
|  |  |  |  |
| --- | --- | --- | --- |
| **S.NO** | **TITLE** | **AUTHORS** | **DESCRIPTION** |
| 1. | Communication system for deaf  and dumb people | Shraddha R.  Ghorpade, Prof. Surendra K. Waghmare | The human hand  comprises of numerous associated parts and joints, making it a complex object for 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 into HSV image and then into binary image. The edges are detected using canny edge detection. |
| 2. | Double Handed Indian Sign Language to Speech and Text | Kusurnika Krori Dutta , Satheesh Kumar Raju , Anil Kumar G , Sunny Arokia Swarny | 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. | 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 system are very encouraging. |
| 4. | 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. |
| 5. | Two Hand Indian Sign Language dataset for benchmarking classification models of Machine Learning | Leela Surya Teja Mangamuri, Lakshay Jain Abhishek Sharmay | This dataset was benchmarked on six different classification models of machine learning by changing the parameters.Classification models are 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 |

**2.1 Problem Statement:**

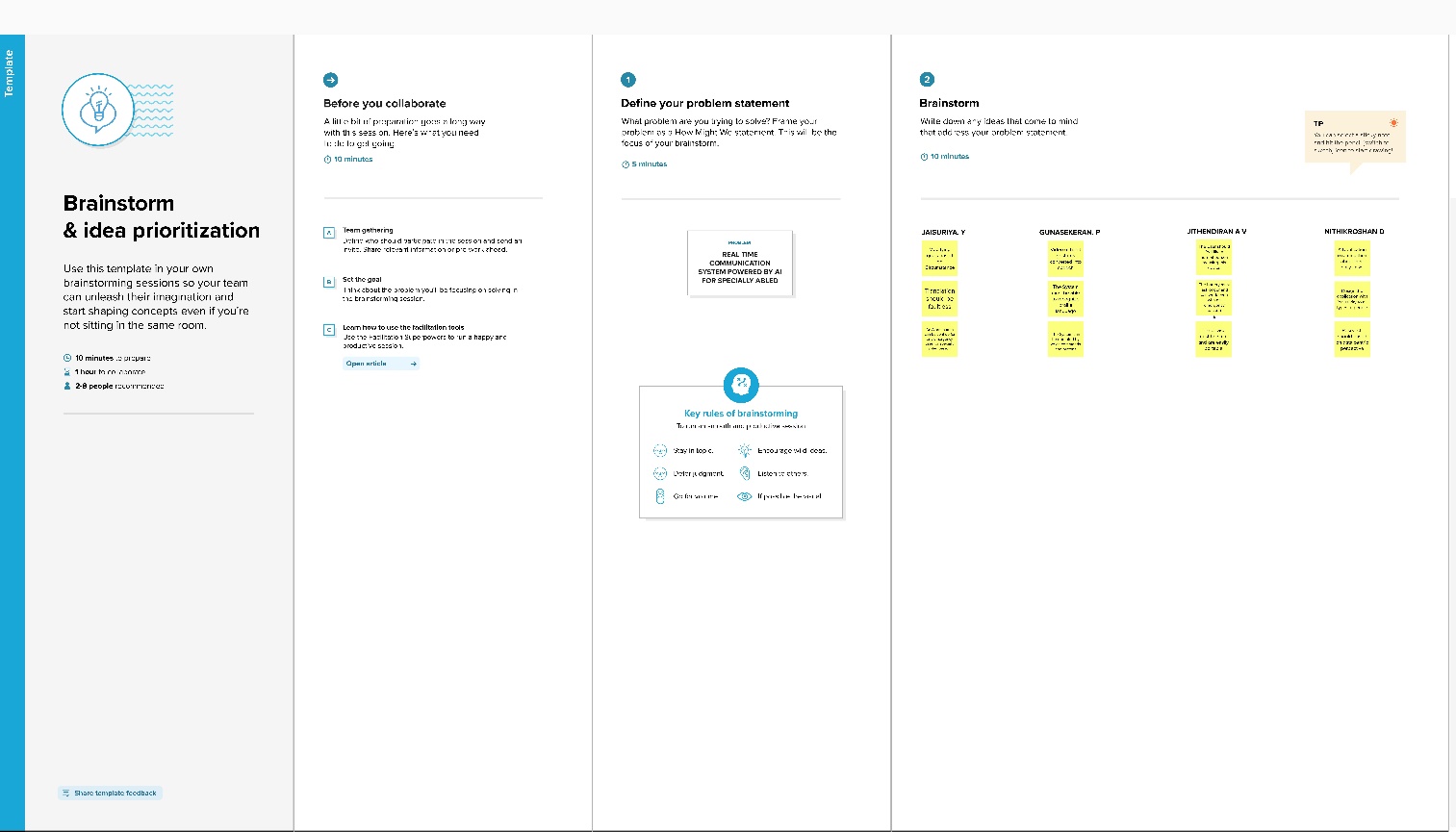
To develop a solution that converts the gestures into a human hearing voice in a respective language to convey a message to ordinary people, and also it converts our voice into recognizable sign language for the deaf and dumb people.

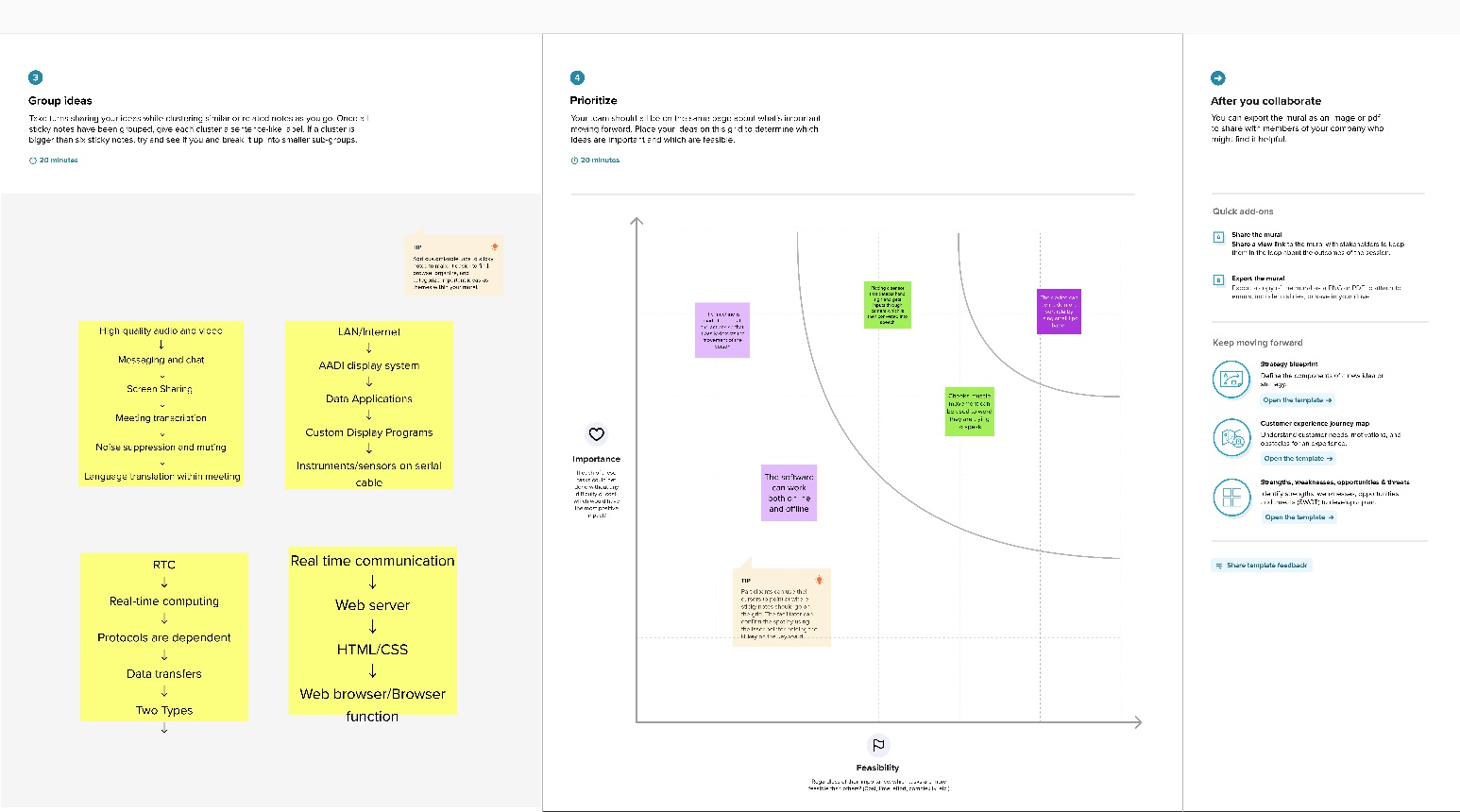
**3.IDEATION & PROPOSED SOLUTION**

**3.1 Empathy Map Canvas:**



**3.2 Ideation & Brainstorming:**





**3.3 Proposed Solution:**

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Parameter** | **Description** |
|  | Problem Statement (Problem to be solved) | * To develop a solution that converts the gestures into a human hearing voice in a respective language to convey a message to ordinary people, and also it converts our voice into recognizable sign language for the deaf and dumb people. |
|  | Idea / Solution description | * Communication between deaf and dumb and a normal person has always been a difficult problem. * We have used convolution neural network to create a model that is trained with different hand gestures. * An is integrated with this .This app enables deaf and dumb people to convey their information using signs/ gestures which get converted to human-recognizable language and voice is given as output. |
| * x | Novelty / Uniqueness | * Much cost is not involved. * It is unique because speech can be converted to text is easy but gestures to voice is something unique. |
|  | Social Impact / Customer Satisfaction | * It act as a great solution for many deaf and dumb people . * Even a specially abled person can act like a normal person. |
|  | Business Model (Revenue Model) | .   * As it is used by many differently abled persons it can increase revenue drastically. |
|  | Scalability of the Solution | * Mainly made available for many specially abled people. * Code can be updated as per requirement in future. |

**3.4 PROBLEM SOLUTION:**



**4.REQUIREMENT ANALYSIS**

## 4.1 *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 Form Registration through Gmail  Registration through LinkedIN |
| FR-2 | User Confirmation | Confirmation via Email Confirmation via OTP |
| FR-3 | system | Desktop with high resolution camera |
| FR-4 | External interface | Ethernet, Wi-Fi, USB to provide internet facility to access the resources with real time communication. |
| FR-5 | Authorization Levels | There are two levels of authorization namely standard  access level and advanced access level. |
| FR-6 | Reporting | If any issues found in the application, automatically it  will be notified to the developer. |

## 4.2 Non-*Functional Requirements:*

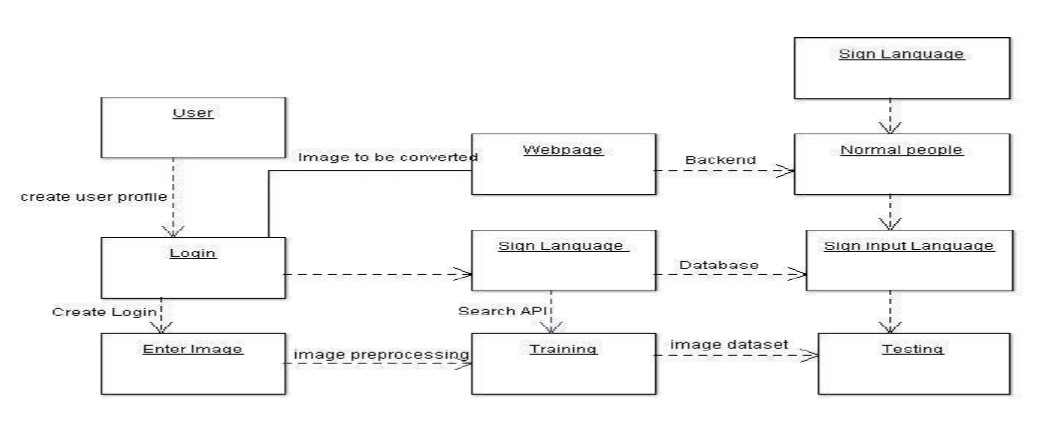
Following are the non-functional requirements of the proposed solution.

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Non-Functional Requirement** | **Description** |
| NFR-1 | **Usability** | To convey a message to normal people, as well as convert speech into understandable sign language  for the deaf and dumb people. |
| NFR-2 | **Security** | To convey a message to normal people, as well as convert speech into understandable sign language  for the deaf and dumb people. |
| NFR-3 | **Reliability** | Provides insight into potential issues for desktop  applications on managed devices. |
| NFR-4 | **Performance** | The time for converting signs into speech should be faster for the real time communication. |
| NFR-5 | **Availability** | The time for converting signs into speech should be  faster for the real time communication. |
| NFR-6 | **Scalability** | 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. |

**5.PROJECT DESIGN**

**5.1 Data Flow Diagrams:**

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

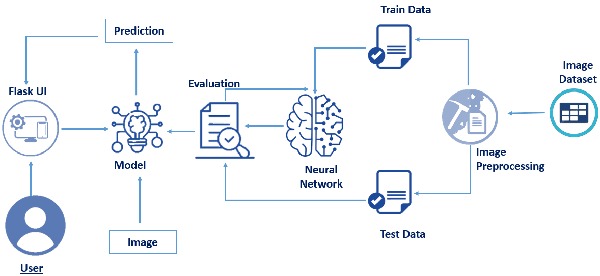


# 5.2 SOLUTION AND TECHNICAL ARCHITECTURE

# Summary

This code pattern explains how to build an Artificial Intelligence based Real Time Communication System for specially abled.

**Technology Architecture:**

****

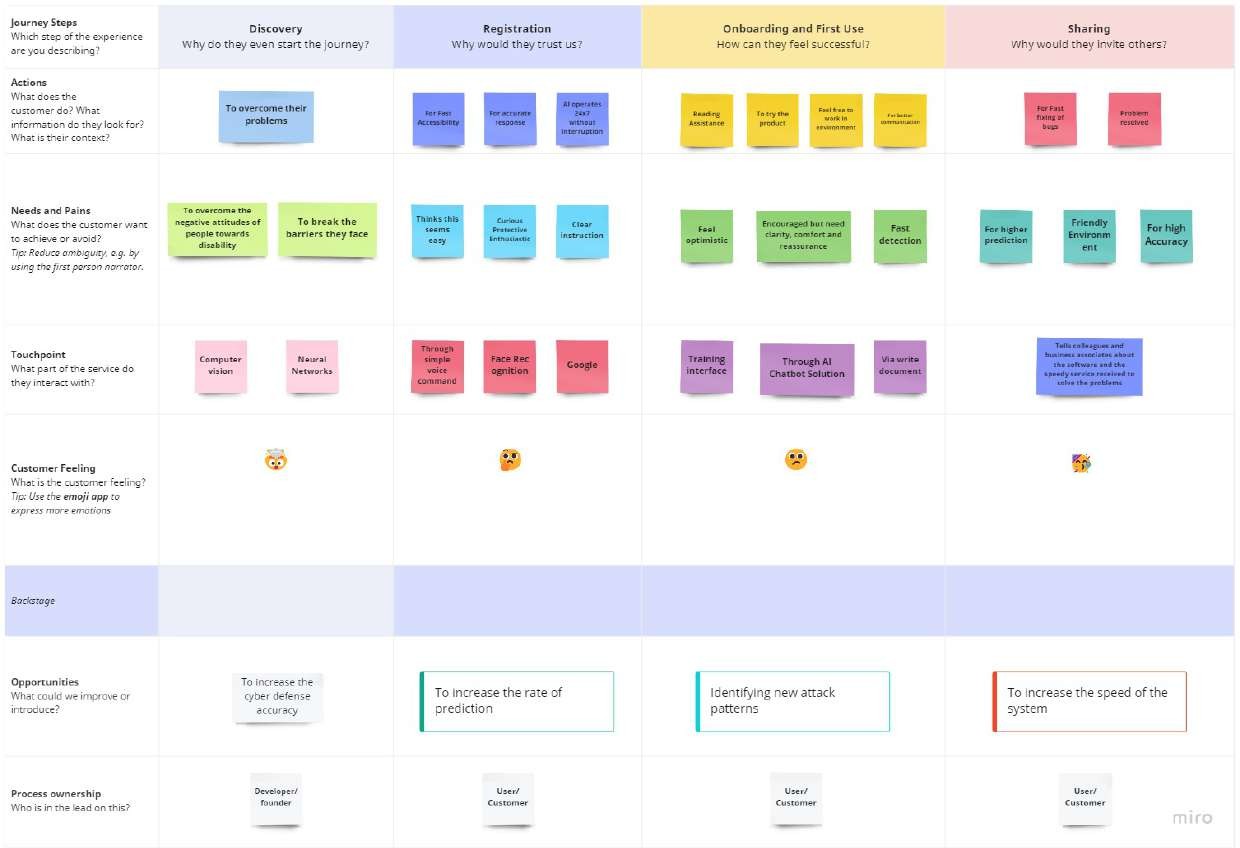
**Components & Technologies:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No.** | **Components** | **Description** | **Technology** |
| 1. | User Interface | The user interface is the point of human computer interaction and communication in device. | Python flask, HTML, CSS/JavaScript. |
| 2. | Flash UI | Flash's user interface components let you interact with the users that use your site and gather information. | Using the cloud, it can be executed. |
| 3. | Models | Support Vector Machine (SVM) is subsequently applied to classify our gesture image dataset. | Machine Learning. |
| 4. | Image | Image processing is used to extract signs from the image using neural network. | ANN, CNN, Open CV. |
| 5. | Evaluate data | Aims to estimate the generalization accuracy of a model on future (unseen/out-of-sample) data. | NLP. |
| 6. | Unstructured data | Unstructured data is a conglomeration of many varied types of data that are stored in their native formats. | Natural Language Processing (NLP). |
| 7. | Structured data | Typically categorized as quantitative data is highly organized and easily decipherable by machine learning algorithms. | Machine language and artificial intelligence tools. |
| 8. | File Storage | File storage requirements to store the trained model in order to use it whenever it is needed. | IBM Block Storage or Cloud object. |
| 9. | ML service | Provides a full range of tools and services so that you can build, train, and deploy Machine Learning models. | Python, IBM Watson. |
| 10. | IBM Cloud | IBM Watson Studio empowers data scientists, developers and analysts to build, run and manage AI models, and optimize decisions anywhere on IBM Cloud Pak for Data | IBM Cloud and Watson Studio service |
| 11. | Dataset | First prototype of this system used a dataset of 24 static signs from the Panamanian Manual Alphabet. | AI technology. |

**Application Characteristics:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.no** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | Helps you implement best practices for data automation, model tracking, performance monitoring, and model retraining. | TensorFlow. |
| 2. | Security Implementations | It operates the largest national network of professional monitoring centres and offers a six-month, money-back guarantee to customers. | ADT type of coding. |
| 3. | Scalable Architecture | Three-tier architecture is a well-establishedsoftware application architecture that organizes applications into three  logical and physical computing tiers: the data tier, the presentation tier and the application tier. | 3 – Tier Architecture. |
| 4. | Availability | The system will be made ubiquitous so that it is available  everywhere. | Web Application. |
| 5. | Performance | The model will be fine-tuned to strike a balance between  accuracy vs performance. | Optimization of code and trained model. |

**5.3** **Customer Journey:**



**6.PROJECT PLANNING AND SCHEDULING**

**6.1 SPRINT PLANNING & SCHEDULING:**

|  |  |  |
| --- | --- | --- |
| **TITLE** | **DESCRIPTION** | **DATE** |
| **Literature Survey & Information Gathering** | Literature survey on the selected project is done by gathering information about related details on technical papers and web browsing. | 06 OCTOBER 2022 |
| **Empathy Map** | Prepared Empathy Map Canvas to combine thoughts and pains, gains of the project with all team members . | 08 OCTOBER 2022 |
| **Ideation** | Brainstorming session is conducted with all team members to list out all the ideas and prioritise the top 3 ideas. | 09 OCTOBER 2022 |
| **Proposed Solution** | Prepared the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc. | 28 OCTOBER 2022 |
| **Problem Solution Fit** | Prepared problem - solution fit document. | 30 OCTOBER 2022 |

**6.2 SPRINT DELIVERY SCHEDULE**

**Product Backlog, Sprint Schedule, and 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. | 3 | High | SHIVARAMAN G |
| Sprint – 1 | Authentication | USN – 2 | As a user, I will receive OTP to confirm details. | 2 | High | ARAVINDAN S |
| Sprint – 1 | Registration | USN – 3 | As a user, I will receive confirmation email once I have registered for the application. | 1 | Low | MOHAMMED ZAID J |
| Sprint – 1 | Login | USN – 4 | As a user, I can log into the application by entering email & password. | 2 | High | ARUNHACHALAM VR |
| Sprint – 2 | Dashboard | USN – 5 | As a user, I must have one place to explore all available features. | 3 | High | SHIVARAMAN G |
| Sprint – 2 | Login | USN – 6 | As a user, If I forget my password, I must get an auto-generated password to reset my password. | 2 | Medium | SHIVARAMAN G |
| Sprint – 3 | Help | USN – 7 | As a user, I must be able to reach out to the Support Team to get my issues resolved. | 1 | Low | SATHESH KANNAN A |
| Sprint – 3 | Management | USN – 8 | As a user, I can access the site using mobile/ desktop. | 3 | High | ARAVINDAN S |
| Sprint – 4 | System | USN – 9 | As a user, I must have access to previous usage history. | 2 | Medium | ARUNHACHALAM  VR |
| Sprint – 4 | System | USN – 10 | As a user, I can have audio output as well as text output. | 3 | High | MOHAMMED ZAID J |

**Project Tracker, Velocity & Burndown Charts**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **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 | 8 | 6 Days | 24 October, 2022 | 29 October, 2022 | 8 | 01 November, 2022 |
| Sprint – 2 | 5 | 6 Days | 31 October, 2022 | 05 November, 2022 | 5 |  |
| Sprint – 3 | 4 | 6 Days | 07 November, 2022 | 12 November, 2022 | 4 |  |
| Sprint – 4 | 5 | 6 Days | 14 November, 2022 | 19 November, 2022 | 5 |  |

**Velocity:**

Velocity

Average velocity = -----------------------------

Sprint Duration

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

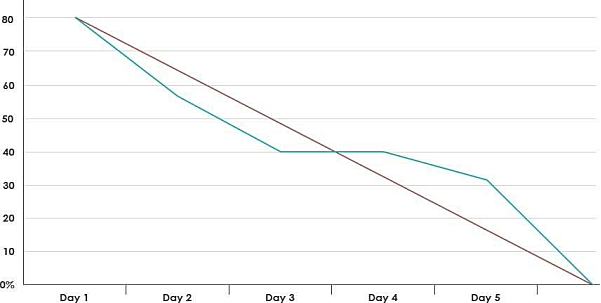
1. Sprint-1: AV = 8/6 = 1.34

2. Sprint-1: AV = 5/6 = 0.834

3. Sprint-1: AV = 4/6 = 0.67

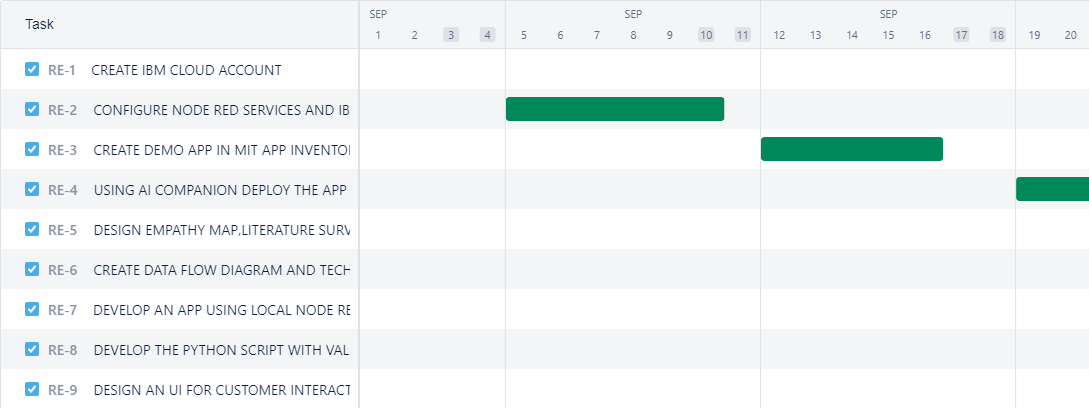
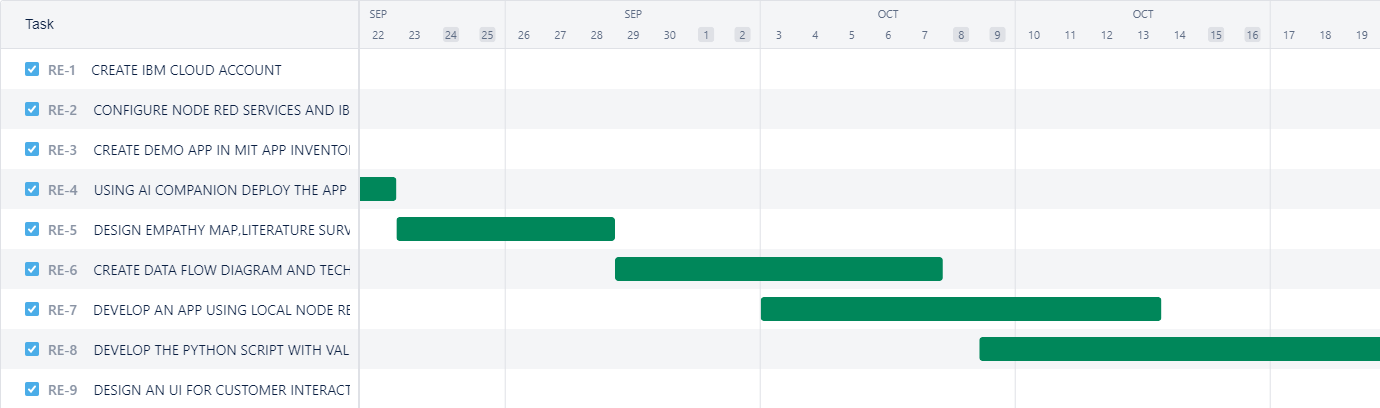
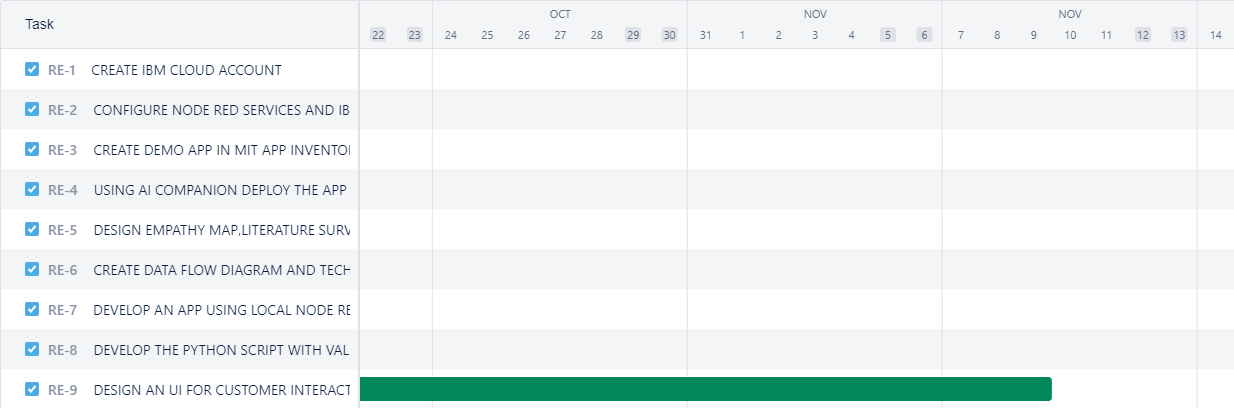
4. Sprint-1: AV = 5/6 = 0.834

**Burndown Chart:**

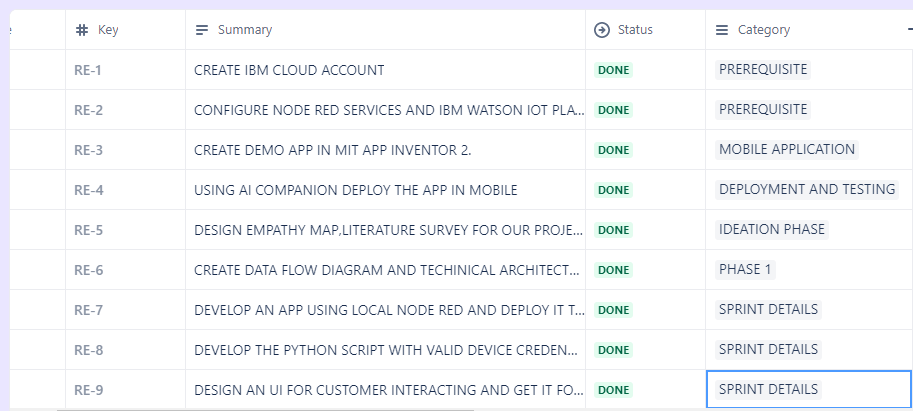


**6.3 REPORT FROM JIRA**

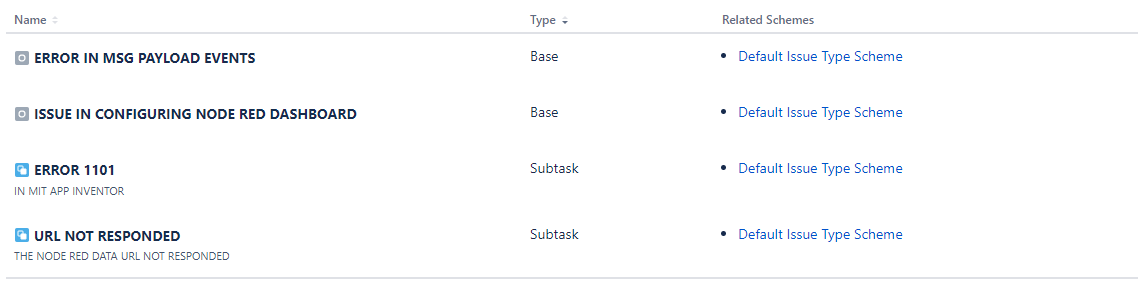
**TIMELINE CREATED USING JIRA SOFTWARE**



**LISTS IN JIRA:**



**ISSUES :**



**7.CODING AND SOLUTIONING**

**7.1 NODE RED SERVICE ASSOCIATED WITH IBM CLOUD:**

The proposed system is designed to develop a real-time sign language detector using a TensorFlow object detection API and train it through transfer learning for the created dataset. For data acquisition, images are captured by a webcam using Python and OpenCV following the procedure described under Section 3.

Following the data acquisition, a labeled map is created which is a representation of all the objects within the model, i.e., it contains the label of each sign (alphabet) along with their id. The label map contains 26 labels, each one representing an alphabet. Each label has been assigned a unique id ranging from 1 to 26. This will be used as a reference to look up the class name. TF records of the training data and the testing data are then created using generate\_tfrecord which is used to train the TensorFlow object detection

API. TF record is the binary storage format of TensorFlow. Binary files usage for storage of the data significantly impacts the performance of the import pipeline consequently, the training time of the model. It takes less space on a disk, copies fast, and can efficiently be read from the disk.

The open-source framework, TensorFlow object detection API makes it easy to develop, train and deploy an object detection model. They have their framework called the TensorFlow detection model zoo which offers various models for detection that have been pre-trained on the COCO 2017 dataset. The pre-trained TensorFlow model that is being used is SSD MobileNet v2 320x320. The SSD MobileNet v2 Object detection model is combined with the FPN-lite feature extractor, shared box predictor, and focal loss with training images scaled to 320x320. Pipeline configuration, i.e., the configuration of the pre-trained model is set up and then updated for transfer learning to train it by the created dataset. For configuration, dependencies like TensorFlow, config\_util, pipeline\_pb2, and text\_format have been imported. The major update that has been done is to change the number of classes which is initially 90 to 26, the number of

signs (alphabets) that the model will be trained on. After setting up and updating the configuration, the model was trained in 10000 steps. The hyper-parameter used during the training was to set up the number of steps in which the model will be trained which was set up to 10000 steps. During the training, the model has some losses as classification loss, regularization loss, and localization loss. The localization loss is mismatched between the predicted bounding box correction and the true values.

**8.TESTING**

# 8.1 Test Case Analysis

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Section** | **Total Cases** | **Not Tested** | **Fail** | **Pass** |
| Print Engine | 15 | 0 | 0 | 15 |
| Client Application | 45 | 0 | 0 | 45 |
| Security | 1 | 0 | 0 | 1 |
| Outsource Shipping | 2 | 0 | 0 | 2 |
| Exception Reporting | 10 | 0 | 0 | 10 |
| Final Report Output | 4 | 0 | 0 | 4 |
| Version Control | 3 | 0 | 0 | 3 |

**8.2 USER ACCEPTANCE TESTING:**

# Purpose of Document

# The purpose of this document is to briefly explain the test coverage and open issues of the REAL TIME COMMUNICATION SYSTEM USING AI FOR DIFFERENTIALY ABLED project at the time of the release to User Acceptance Testing (UAT).

# Defect Analysis

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

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Resolution** | **Severity 1** | **Severity 2** | **Severity 3** | **Severity 4** | **Subtotal** |
| By Design | 9 | 5 | 4 | 3 | 21 |
| Duplicate | 2 | 0 | 2 | 0 | 4 |
| External | 3 | 4 | 1 | 2 | 10 |
| Fixed | 10 | 1 | 5 | 17 | 33 |
| Not Reproduced | 0 | 0 | 1 | 0 | 1 |
| Skipped | 0 | 0 | 1 | 2 | 3 |
| Won't Fix | 0 | 3 | 3 | 1 | 7 |
| Totals | 24 | 13 | 17 | 25 | 79 |

**9.RESULT**

**PERFORMANCE TABLE**

|  |  |  |
| --- | --- | --- |
| PARAMETER | PERFORMANCE | DESCRIPTION |
| ADMIN TESTING | 95%-100% | THE TESTING DONE BEFORE IT IS DEPLOYED AS AN APP |
| CUSTOMER SATISFACTION | 75-85% | THE SPECIALLY ABLED PERSON NEED TO BE SATISFIED WITH THE APPLICATION |
| USER INTERFACE | 65-85% | THE APP CAN USED BY ANYONE. (EASE OF ACCESS) |
| SEVER RESPONSE | 50-75% | URL - response |
| DATA VALIDATION WITH NO. OF TEST CASE | 60-80%  (15-30 TESTCASE) | VALID DATA FROM THE APP |
| ERROR | 3-5% | SOMETIMES IT MAY LAG. |

**10.ADVANTAGES AND DISADVANTAGES**

**ADVANTAGES:**

* Mainly made available for many specially abled people.
* Code can be updated as per requirement in future.
* Much cost is not involved.
* It is unique because speech can be converted to text is easy but gestures to voice is something unique.

**DISADVANTAGES:**

* Computational time in training is higher.
* Selection of an optimal threshold value is difficult.
* Sometimes it may lag due to network connectivity.

**11.CONCLUSION**

The problem of Sign Language (SL) recognition using images is still a challenge. Similarity of gestures, user’s accent, context and signs with multiple meanings lead to ambiguity. These are some reasons why previous work used limited datasets Here I used keras and Tensor flow to train the models then finally converted the hand gesture into the Text. This can easily be understood by a person by seeing it though he is a deaf and dumb person.

**12.FUTURE SCOPE**

The implementation of our model on sign languages such as Indian sign language or American sign language, helps specially abled persons to communicate through gestures. Further training with large dataset to efficiently recognize symbols. Improving the model's ability to identify expression.

**13.APPENDIX**

**13.1 SOURCE CODE:**

**TESTED AND TRAINED**

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)

x\_train = train\_datagen.flow\_from\_directory('/content/Dataset/training\_set',target\_size=(64,64),batch\_size=300,class\_mode='categorical',color\_mode="grayscale")

Found 15750 images belonging to 9 classes.

x\_test = test\_datagen.flow\_from\_directory('/content/Dataset/test\_set',target\_size=(64,64),batch\_size=300,class\_mode='categorical',color\_mode="grayscale")

Found 2250 images belonging to 9 classes.

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

model = Sequential()

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

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

model.add(Flatten())

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

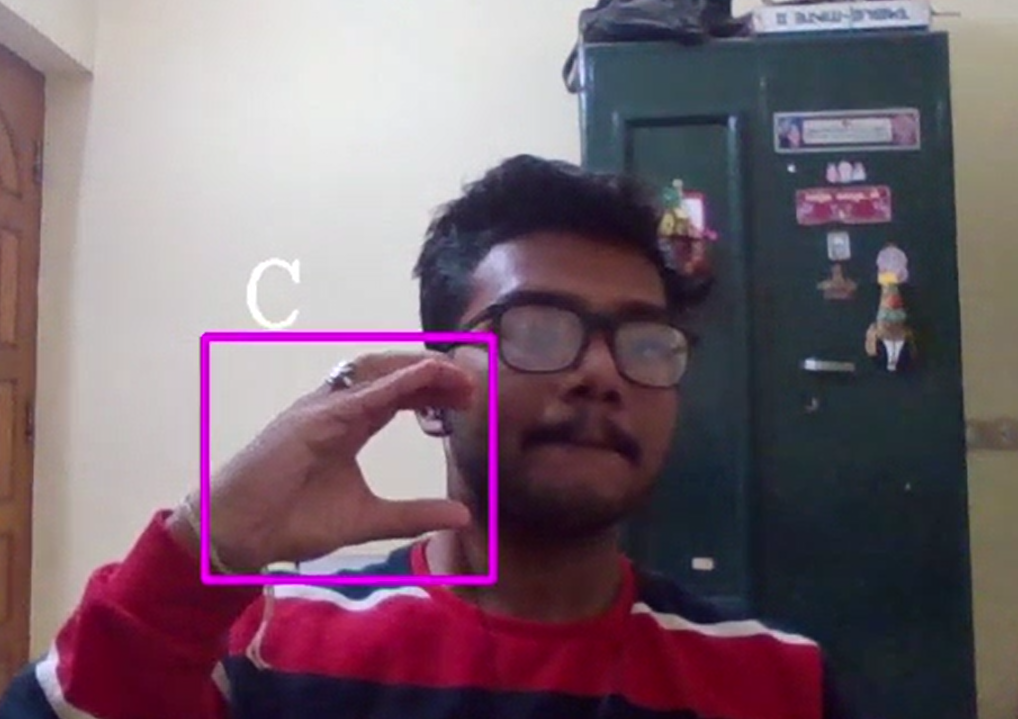
model.add(Dense(units=9, activation = 'softmax'))

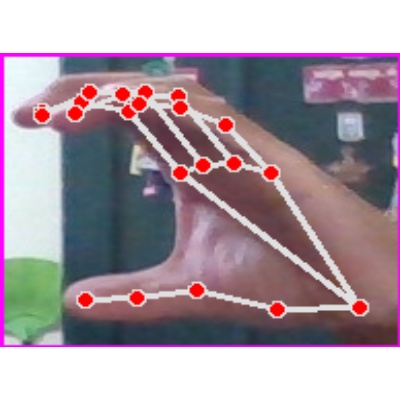
model.compile(loss='categorical\_crossentropy', optimizer = 'adam', metrics = ['accuracy'])

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

**OUTPUT**

****

****

**HTML CODE:**

<!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>Sign Language Detection</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="assets/css/Banner-Heading-Image.css">

<link rel="stylesheet" href="assets/css/Navbar-Centered-Brand.css">

<link rel="stylesheet" href="assets/css/styles.css">

</head>

<body style="background: rgb(247, 246, 244);">

<nav class="navbar navbar-light navbar-expand-md py-3" style="background: #212529;">

<div class="container">

<div></div><a class="navbar-brand d-flex align-items-center" href="#"><span

class="bs-icon-sm bs-icon-rounded bs-icon-primary d-flex justify-content-center align-items-center me-2 bs-icon"><i

class="fas fa-flask"></i></span><span style="color: rgb(255,255,255);">Real-Time Communication

System Powered By AI&nbsp;For Specially Abled</span></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: 640px;height: 480px;margin: 10px;min-height: 480px;min-width: 640px;border-radius: 10px;border: 4px dashed rgb(0, 0, 0) ;">

<img src="{{ url\_for('video\_feed') }}" style="width: 100%;height: 100%;color: rgb(0, 0, 0);text-align: center;font-size: 20px;"

alt="Camera Access Not Provided!">

</div>

</div>

<div class="d-flex flex-column justify-content-center align-items-center" style="margin-bottom: 10px;"><button

class="btn btn-info" type="button" data-bs-target="#modal-1" data-bs-toggle="modal">Quick Reference

-<strong> ASL Alphabets</strong></button></div>

</section>

<section>

<div class="container">

<div class="accordion text-white" role="tablist" id="accordion-1">

<div class="accordion-item" style="background: rgb(33,37,41);">

<h2 class="accordion-header" role="tab"><button class="accordion-button" data-bs-toggle="collapse"

data-bs-target="#accordion-1 .item-1" aria-expanded="true"

aria-controls="accordion-1 .item-1"

style="background: rgb(39,43,48);color: rgb(255,255,255);">About The Project</button></h2>

<div class="accordion-collapse collapse show item-1" role="tabpanel" data-bs-parent="#accordion-1">

<div class="accordion-body">

<p class="mb-0">Artificial Intelligence has made it possible to handle our daily activities

in new and simpler ways. With the ability to automate tasks that normally require human

intelligence, such as speech and voice recognition, visual perception, predictive text

functionality, decision-making, and a variety of other tasks, AI can assist people with

disabilities by significantly improving their ability to get around and participate in

daily activities.<br><br>Currently, Sign Recognition is available <strong>only for

alphabets A-I</strong> and not for J-Z, since J-Z alphabets also require Gesture

Recognition for them to be able to be predicted correctly to a certain degree of

accuracy.</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>

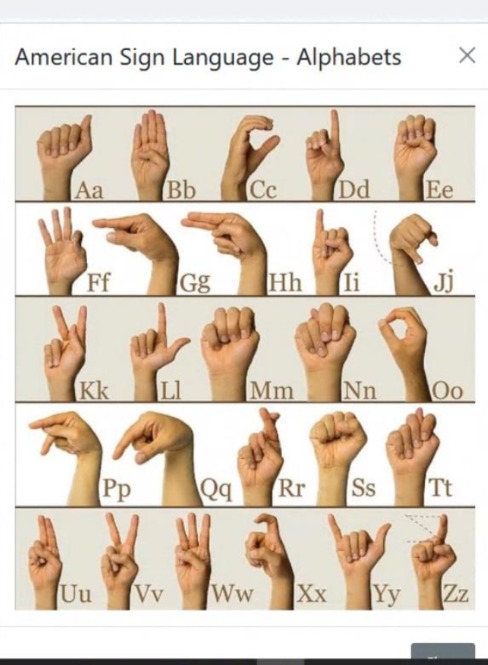
</div>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>

</body>

</html>

**OUTPUT**



**13.2 GIT-HUB LINK:**

**https://github.com/IBM-EPBL/IBM-Project-13394-1659517893**

**PROJECT DEMO LINK:**

1. **https://vimeo.com/772824247**