Professional Readiness for

Innovation, Employability and Entrepreneurship

Executed By

IBM

REAL-TIME COMMUNICATION SYSTEM POWERED BY AI FOR SPECIALLY ABLED

A PROJECT REPORT

Submitted by

SRIVIVEKNATH S R

SURYA V

VIJAY A

ZAFAR SADIK H D

Of

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

Thandalam – 602105

CHAPTER 1
INTRODUCTION

1.1 Project Overview

The main purpose of this exploration is to enhance the communication of the impaired community. The proposed system is enhanced interpersonal-mortal commerce for people with special requirements, especially those with physical and communication disabilities. The proposed model comprises an automated real-time sign language recognition system, designed and enforced with the ubiquitous and affordable conception in mind to suit the depressed. The prototype presented encapsulates an automated sign language recognition system for covering the differently challenged person. Convolutional neural networks (CNN) are used to achieve this purpose. The model is tested with various cases and achieved satisfactory results. The enhanced real-time sign language recognition system, is an assistive tool to ameliorate the quality of life for the impaired by aiding them anytime and anywhere when demanded. They can do their own tasks more singly without constantly being covered physically or accompanied by their caretakers, preceptors, or indeed parents.

1.2 Purpose

Communication is a social process of swapping information from one reality to another in verbal and nonverbal forms. It comes naturally as a raw skill bedded in utmost people at birth and we acquire the ways of communication through cognitive literacy. It defines our actuality and it's an important instrument that connects people together. It plays an important part in the dissemination of information and sharing of knowledge, especially in the academic arena. Communication skills might come as a natural capability in the maturity of people. Still, there are some people foisted with some form of physical blights which affect their capability to communicate. Many assistive tools, formally nominated as Indispensable and Augmentative Communication(AAC), have been developed and employed to help people with disabled communication chops.

CHAPTER 2
LITERATURE SURVEY

2.1 Existing problem

One of the most precious gifts of nature to the mortal race is the capability to express itself by responding to the events that occur in its terrain. Every normal person sees, hears, and also reacts to the situations by expressing himself. But there are some lower lucky bones who are deprived of this precious gift. similar people, especially deaf and mute, calculate on some kind of gesture language to communicate their passions to others. The deaf, dumb and the eyeless follow analogous problems when it comes to the use of computers. In the period of advanced technologies, where computers, laptops and other processor- grounded bias are an integral part of everyday life, sweats must be made to make the disabilities in life more independent. Our thing is to design a mortal computer interface system that can directly identify the language of the deaf and dumb. With the use of image processing and artificial intelligence, Numerous ways and algorithms have been developed in this area. Each character speech recognition system is trained to fete the characters and convert them into the needed pattern. The proposed system aims to give speech speechless, a real- time character language is captured as a series of images, and it's reused and also converted into speech and textbook

2.2 References

[1] Machine Learning Techniques for Indian Sign Language Recognition Kusumika Krori Dutta, Sunny Arokia Swamy Bellary et al 2017 International Conference on Current Trends in Computer, Electrical, Electronics and Communication (ICCTCEEC-2017) trained the system with Principal Component Analysis (PCA) and Artificial Neural Network (ANN) algorithm in MATLAB. The hand gestures are acquired from different sized and complexion hands and made an image database for having ease image of size 768 by 1024. The images are pre-processed and resized to 256 by 256. Dimensionality reduction technique is applied for faster classification and it plays a vital role for feature extraction and dimension reduction has been done in one step using Principal Component Analysis (PCA).

[2] K-Nearest Correlated Neighbor Classification for Indian Sign Language Gesture Recognition using Feature Fusion Bhumika Gupta, Pushkar Shukla, Ankush Mittal et al 2016 2016 International Conference on Computer Communication and Informatics (ICCCI -2016), Jan. 07 – 09, 2016, Coimbatore, INDIA proposed that recognition of gesture of Indian Sign Languages using

static images where a test image is first categorized into a single or double handed gesture followed by its classification using a fusion of SIFT and HOG descriptors via K-Nearest Correlated Neighbours. Sub categorizing the gestured alphabets into single-handed or doubled-handed considerably lowered the cost of computation and increased the efficiency of the algorithm due to the reason that there were lesser number of classes in a subgroup compared to the complete set of characters in the Indian Sign Language.

[3] Moment Based Sign Language Recognition For Indian Languages Umang Patel, Aarti G. Ambekar et al 2017 Third International Conference on Computing, Communication, Control And Automation (ICCUBEA) has proposed that double handed sign language is used for sign recognition. The method which is introduced in this paper will use a web camera for capturing gestures & then it will pre-process images of gesture in MATLAB. After having the processed image, next step is feature extraction & followed by classifier, recognized gestures are displayed as Hindi & English text & played as Hindi & English audio. Different classification algorithms are used such as K Nearest Neighbour (KNN) classifier and Probabilistic Neural Network (PNN) classifier. Using classifier, each group is a form of an individual pattern. English text can be easily displayed using MATLAB inbuilt function. The function is called as system.Speech.Synthesis.SpeechSynthesizer which is used for English speech

[4] Two Hand Indian Sign Language dataset for benchmarking classification models of Machine Learning Leela Surya Teja Mangamuri, Lakshay Jain , Abhishek Sharmay et al 2019 2nd International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT) proposed that Skin filtering techniques are proposed for the extraction of hand from the background. Skin coloured pixels are separated from the non- skin coloured pixels by setting up a threshold. Adaptive Boost algorithm used by Fang is very popular for extracting two hands from the background. In the first stage Hand Segmentation and Filtering is done on the whole training and test set images. The hand is segmented from the background by extracting the skin coloured pixels of the image from the non-skin coloured pixels by setting a threshold limit. In the second stage, HOG features are extracted from the hand segmented images of training and test set. HOG feature bin is the magnitude of gradients to a histogram according to its orientation

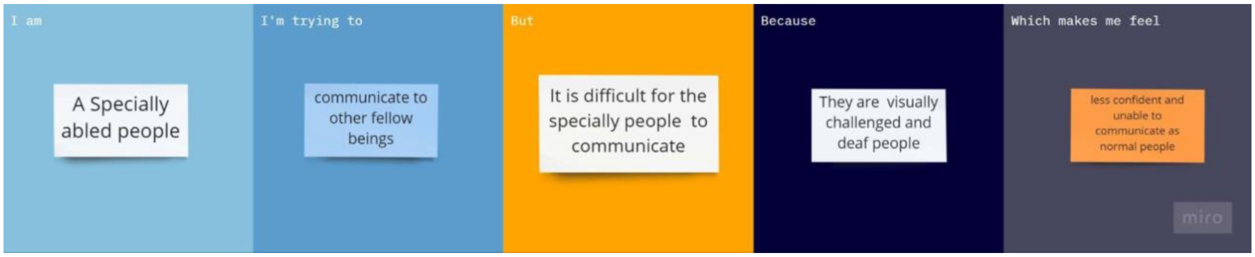
[5] Double Handed Indian Sign Language to Speech and Text. Kusurnika Krori Dutta , Satheesh Kumar Raju , Anil Kumar G , Sunny Arokia Swarny et al 2015 Third International Conference on Image Information Processing The proposed system provides voice to the deaf and mute people and promising them an independent life without any help of human translator. 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. The corresponding output is obtained after extracting Shi-Thomasi good features. Five Images of each alphabet are taken and Min Eigen Value algorithm is applied as shown in figure 3, and interesting points are extracted.

[6] Comprehensive SVM based Indian Sign Language Recognition K. Revanth, N. Sri Madhava Raja et al 2019 Proceeding of International Conference on Systems Computation Automation and Networking proposed that different machine learning algorithms have been applied and SVM (Support Vector Machines) 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 Naïve Bayes. 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 machine learning model.

[7] Indian Sign Language Animation Generation System Sandeep Kaur, Maninder Singh et al 2015 1st International Conference on Next Generation Computing Technologies (NGCT-2015) Dehradun, India, 4-5 September 2015 proposed that Indian Sign Language animation generation system is a based system in which HamNoSys corresponding to words are generated based on ISL. Then SiGML corresponding to HamNoSys is generated that is given as input to JA SIGML URL APP to check the accuracy of these notations. Database is also maintained which stores SiGML file corresponding to Punjabi words. This system has been tested on 100 words. In this system HamNoSys for 100 words are generated according to ISL (Indian Sign language).

2.3 Problem statement definition

In our society, we have people with disabilities. Technology is developing day by day but no significant developments are undertaken for the betterment of these people. Communication between a 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 in hand sign language. In emergency times conveying their message is very difficult.



I am

A Specially abled people

I'm trying to

communicate to other fellow beings

But

It is difficult for the specially people to communicate

Because

They are visually challenged and deaf people

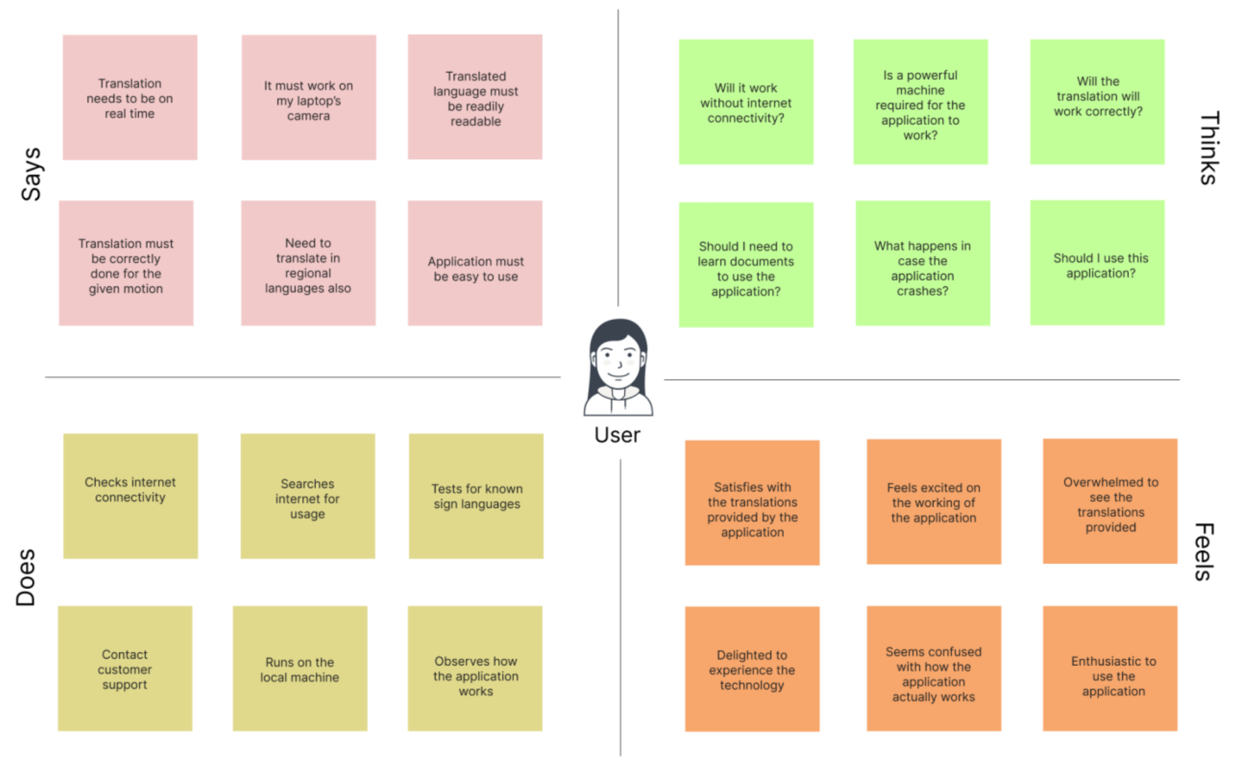
Which makes me feel

less confident and unable to communicate as normal people

CHAPTER 3
IDEATION & PROPOSED SOLUTIONS

3.1 Empathy Map Canvas

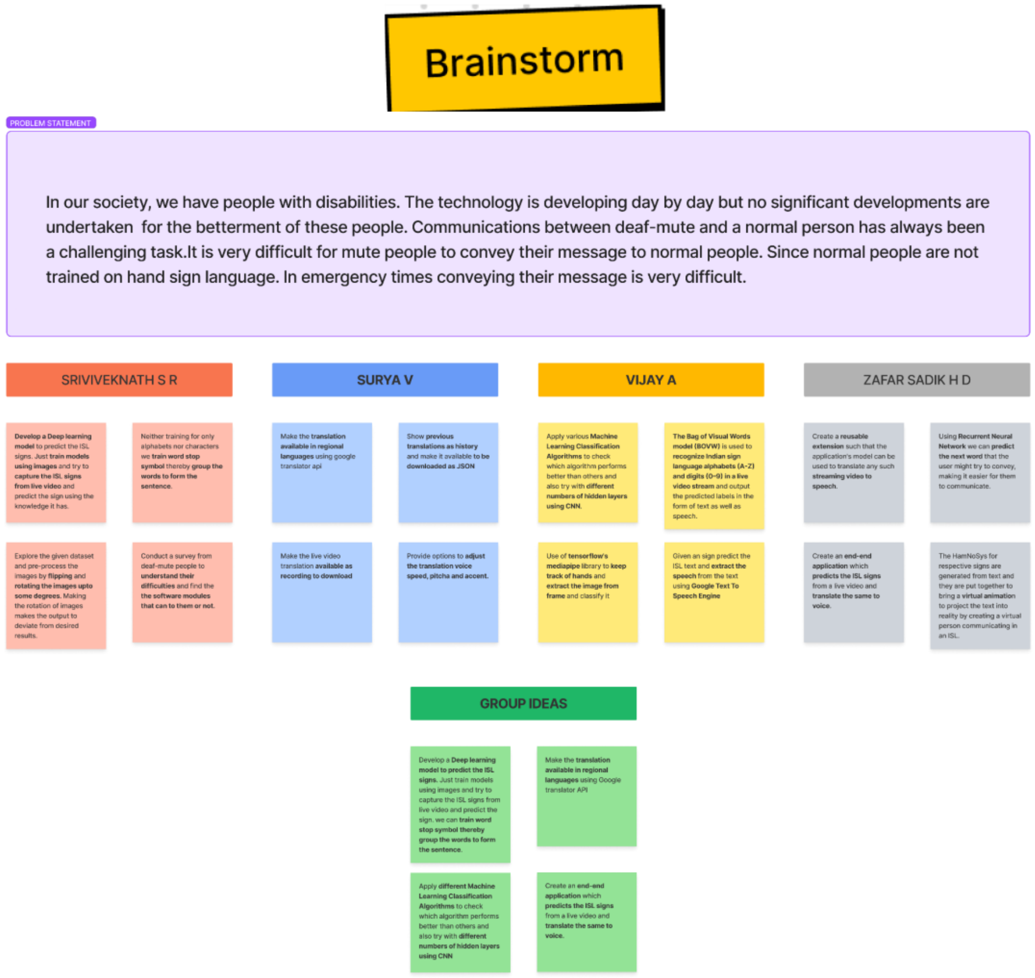
An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviors and attitudes. It is a useful tool to help teams better understand their users. Empathy mapping is a simple workshop activity that can be done with stakeholders, marketing and sales, product development, or creative teams to build empathy for end users. For teams involved in the design and engineering of products, services, or experiences, an empathy mapping session is a great exercise for groups to "get inside the heads" of users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.


Says
Translation needs to be on real time
It must work on my laptop's camera
Translated language must be readily readable
Translation must be correctly done for the given motion
Need to translate in regional languages also
Application must be easy to use
Thinks
Will it work without internet connectivity?
Is a powerful machine required for the application to work?
Will the translation will work correctly?
Should I need to learn documents to use the application?
What happens in case the application crashes?
Should I use this application?
User
Does
Checks internet connectivity
Searches internet for usage
Tests for known sign languages
Contact customer support
Runs on the local machine
Observes how the application works
Feels
Satisfies with the translations provided by the application
Feels excited on the working of the application
Overwhelmed to see the translations provided
Delighted to experience the technology
Seems confused with how the application actually works
Enthusiastic to use the application

3.2 Ideation & Brainstorming

Ideation essentially refers to the whole creative process of coming up with and communicating new ideas. Ideation is innovative thinking, typically aimed at solving a problem or providing a more efficient means of doing or accomplishing something. It encompasses thinking up new ideas, developing existing ideas, and figuring out means or methods for putting new ideas into practice.

Brainstorming is method design teams use to generate ideas to solve clearly defined design problems. In controlled conditions and a free-thinking environment, teams approach a problem by such means as “How Might We” questions. They produce a vast array of ideas and draw links between them to find potential solutions.



Brainstorm

PROBLEM STATEMENT

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.

SRIVIVEKNATH S R

Develop a Deep learning model to predict the ISL signs. Just train models using images and try to capture the ISL signs from live video and predict the sign using the knowledge it has.

Neither training for only alphabets nor characters we train word stop symbol thereby group the words to form the sentence.

Explore the given dataset and pre-process the images by flipping and rotating the images upto some degrees. Making the rotation of images makes the output to deviate from desired results.

Conduct a survey from deaf-mute people to understand their difficulties and find the the software modules that can to them or not.

SURYA V

Make the translation available in regional languages using google translator api

Show previous translations as history and make it available to be downloaded as JSON

Make the live video translation available as recording to download

Provide options to adjust the translation voice speed, pitcha and accent.

VIJAY A

Apply various Machine Learning Classification Algorithms to check which algorithm performs better than others and also try with different numbers of hidden layers using CNN.

The Bag of Visual Words model (BOVW) is used to recognize Indian sign language alphabets (A-Z) and digits (0-9) in a live video stream and output the predicted labels in the form of text as well as speech.

Use of tensorflow's mediapipe library to keep track of hands and extract the image from frame and classify it

Given an sign predict the ISL text and extract the speech from the text using Google Text To Speech Engine

ZAFAR SADIK H D

Create a reusable extension such that the application's model can be used to translate any such streaming video to speech.

Using Recurrent Neural Network we can predict the next word that the user might try to convey, making it easier for them to communicate.

Create an end-end application which predicts the ISL signs from a live video and translate the same to voice.

The HamNoSys for respective signs are generated from text and they are put together to bring a virtual animation to project the text into reality by creating a virtual person communicating in an ISL.

GROUP IDEAS

Develop a Deep learning model to predict the ISL signs. Just train models using images and try to capture the ISL signs from live video and predict the sign. we can train word stop symbol thereby group the words to form the sentence.

Make the translation available in regional languages using Google translator API

Apply different Machine Learning Classification Algorithms to check which algorithm performs better than others and also try with different numbers of hidden layers using CNN

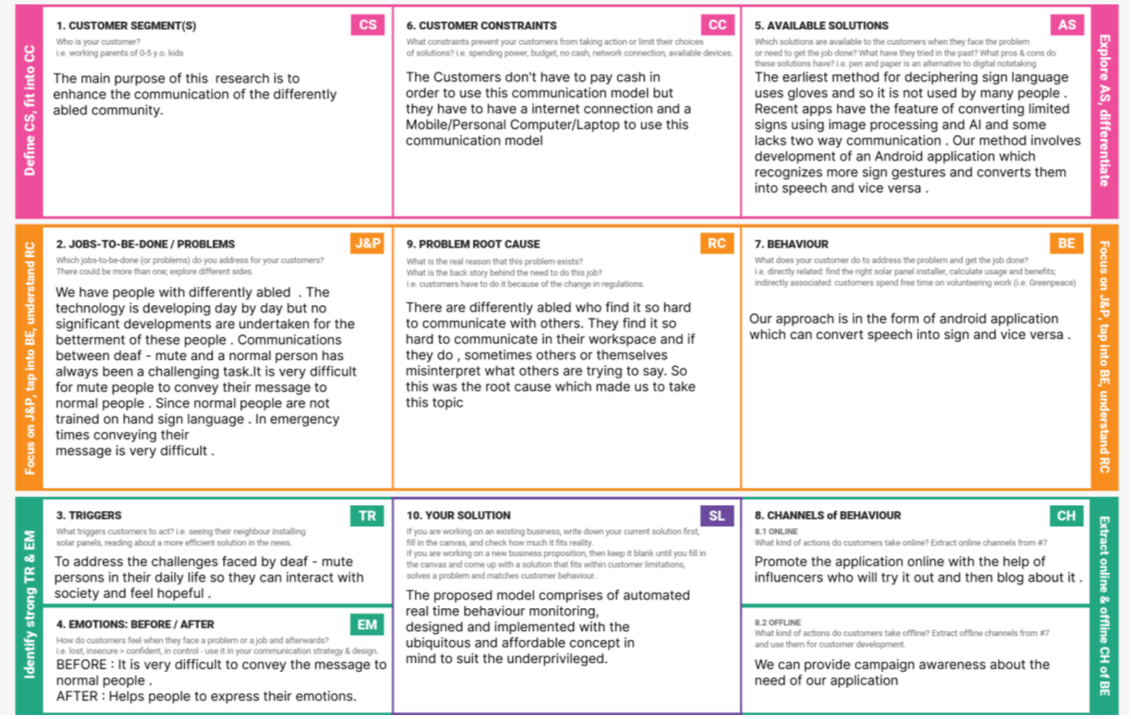
Create an end-end application which predicts the ISL signs from a live video and translate the same to voice.

3.3 Proposed Solution

Sign Language is a visual means of communicating using gestures, facial expressions, and body language with specially-abled. Since normal people are not trained in sign language, in times of emergency conveying their message is very difficult. Hence, there is a need for a system that recognizes different signs and empowers them in communicating with normal people. The idea is to create an end-end application that predicts the ISL signs from a live video and translates the same to voice such that conversing is at ease. We use a stopping symbol to group the letters into a word and generate a sentence and the resulting prediction is converted into speech to convey the information in a convenient manner. Communication is achieved without the help of additional human intervention.

No additional hardware support is needed to use the application. Improve their career opportunities in the industry. Can provide instant results to users. We can associate the application with organizations to provide support for the specially abled. Creating an association with other medical applications to utilize our product in their app. Can be used to sell to a news agency as a replacement for human translation.Since this is a web app people can access the application from any device(Mobile, Desktop, laptop, etc.), and used by everyone across the world. As it is hosted in IBM Cloud, it could be scaled up and down as per demand.

3.4 Problem Solution Fit



Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

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

The main purpose of this research is to enhance the communication of the differently abled community.

6. CUSTOMER CONSTRAINTS CC

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

The Customers don't have to pay cash in order to use this communication model but they have to have a internet connection and a Mobile/Personal Computer/Laptop to use this communication model

5. AVAILABLE SOLUTIONS AS

Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking

The earliest method for deciphering sign language uses gloves and so it is not used by many people . Recent apps have the feature of converting limited signs using image processing and AI and some lacks two way communication . Our method involves development of an Android application which recognizes more sign gestures and converts them into speech and vice versa .

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.

We have people with differently abled . 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 .

9. PROBLEM ROOT CAUSE RC

What is the real reason that this problem exists?
What is the back story behind the need to do this job?
i.e. customers have to do it because of the change in regulations.

There are differently abled who find it so hard to communicate with others. They find it so hard to communicate in their workspace and if they do , sometimes others or themselves misinterpret what others are trying to say. So this was the root cause which made us to take this topic

7. BEHAVIOUR BE

What does your customer do to address the problem and get the job done?
i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

Our approach is in the form of android application which can convert speech into sign and vice versa .

Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

3. TRIGGERS TR

What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.

To address the challenges faced by deaf - mute persons in their daily life so they can interact with society and feel hopeful .

4. EMOTIONS: BEFORE / AFTER EM

How do customers feel when they face a problem or a job and afterwards?
i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

BEFORE : It is very difficult to convey the message to normal people .
AFTER : Helps people to express their emotions.

10. YOUR SOLUTION SL

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

The proposed model comprises of automated real time behaviour monitoring, designed and implemented with the ubiquitous and affordable concept in mind to suit the underprivileged.

8. CHANNELS of BEHAVIOUR CH

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

Promote the application online with the help of influencers who will try it out and then blog about it .

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

We can provide campaign awareness about the need of our application

Extract online & offline CH of BE

CHAPTER 4
REQUIREMENTS ANALYSIS

4.1 Functional Requirements

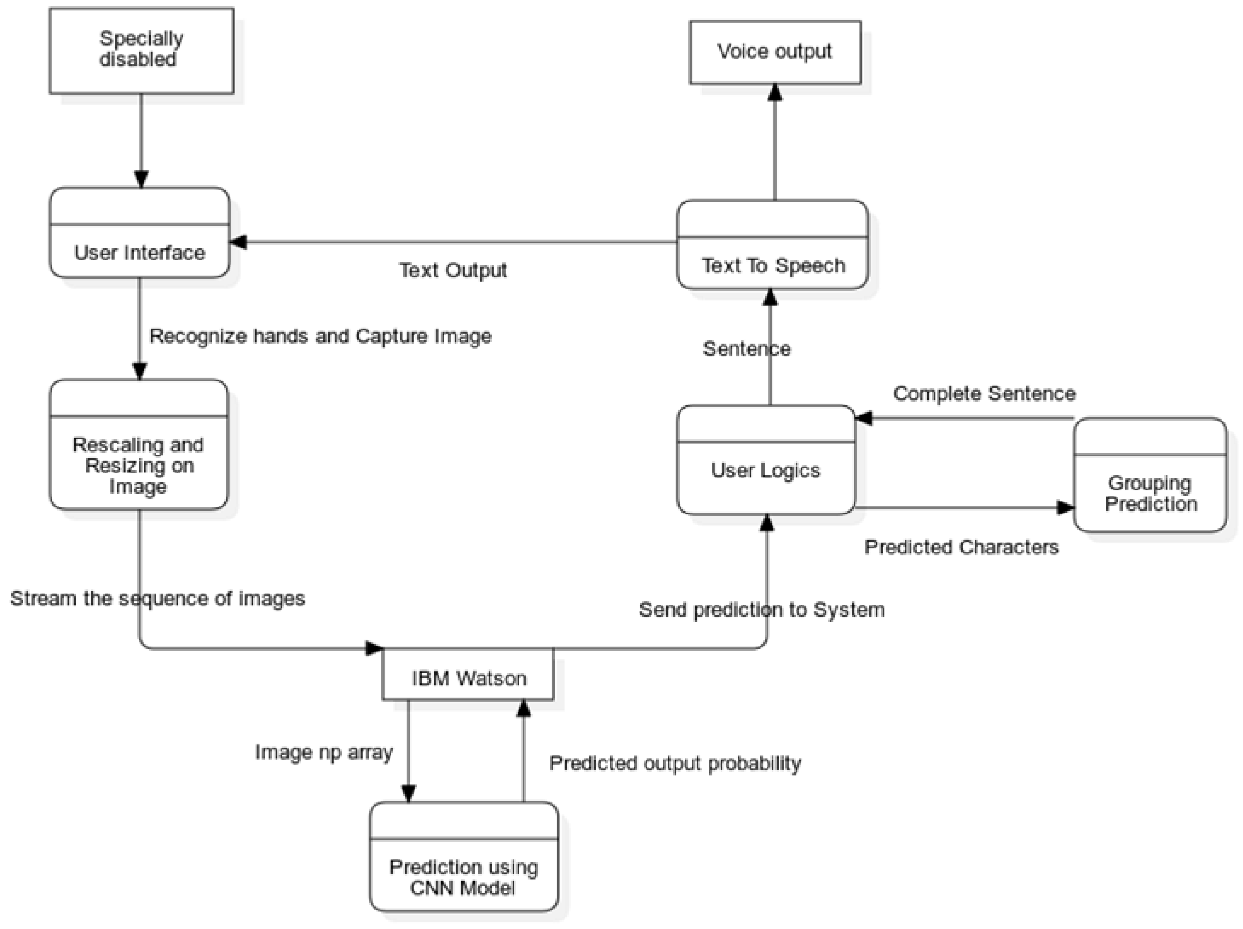
1. The user will start the video from the camera.
2. Users will be suitable to register different signs for further recognition using cameras.
3. When a user starts recognition exertion and gives colorful hand gestures in front of the camera, the sign will be detected
4. The speech will be produced to advertise detected signs.
5. The Video streaming is made available in anytime and anywhere with a bandwidth of at least 0.5Mbps

4.2 Non -functional Requirements

1. To make the application readily available for everyone by providing the same as an affordable solution for the problem.
2. To develop a tool which will help deaf people in communication.
3. The Signs can should be converted to Speech so that there is a two- way communication
4. To be able to run the solution in commodity hardware.
5. To ensure that the application satisfies the required user-friendly constraints.

CHAPTER 5
PROJECT DESIGN

5.1 Data flow diagram


Specially disabled
Voice output
User Interface
Text Output
Text To Speech
Recognize hands and Capture Image
Sentence
Complete Sentence
Rescaling and Resizing on Image
User Logics
Grouping Prediction
Predicted Characters
Stream the sequence of images
Send prediction to System
IBM Watson
Image np array
Predicted output probability
Prediction using CNN Model

5.2 Solution & Technical Architecture

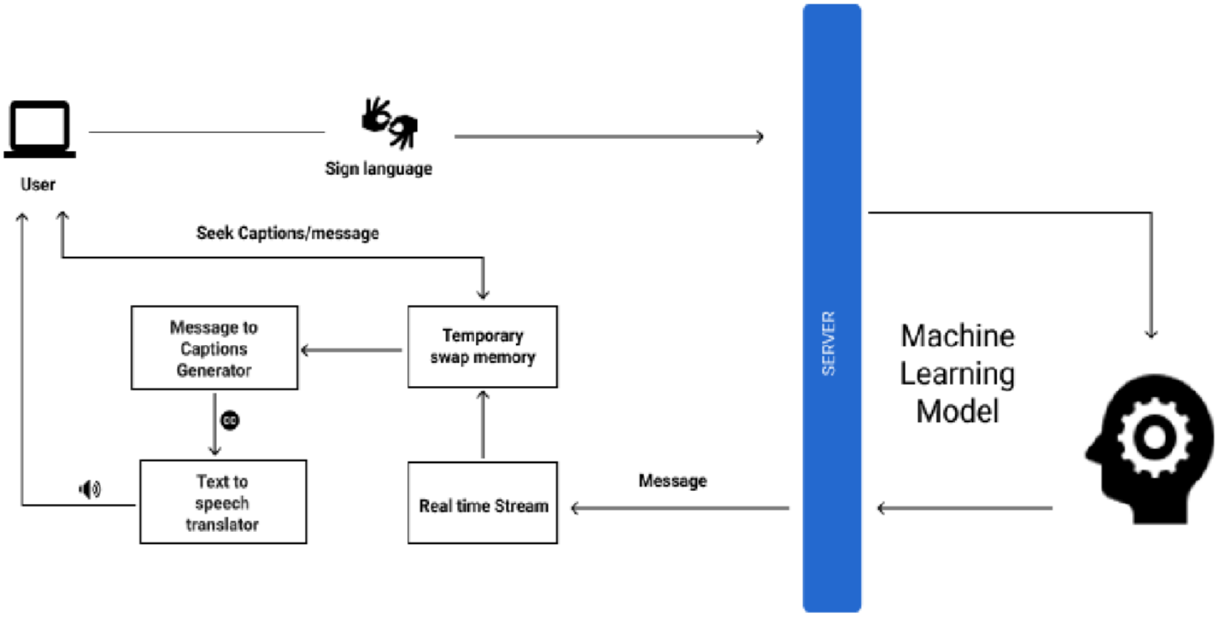


Fig. Solution architecture

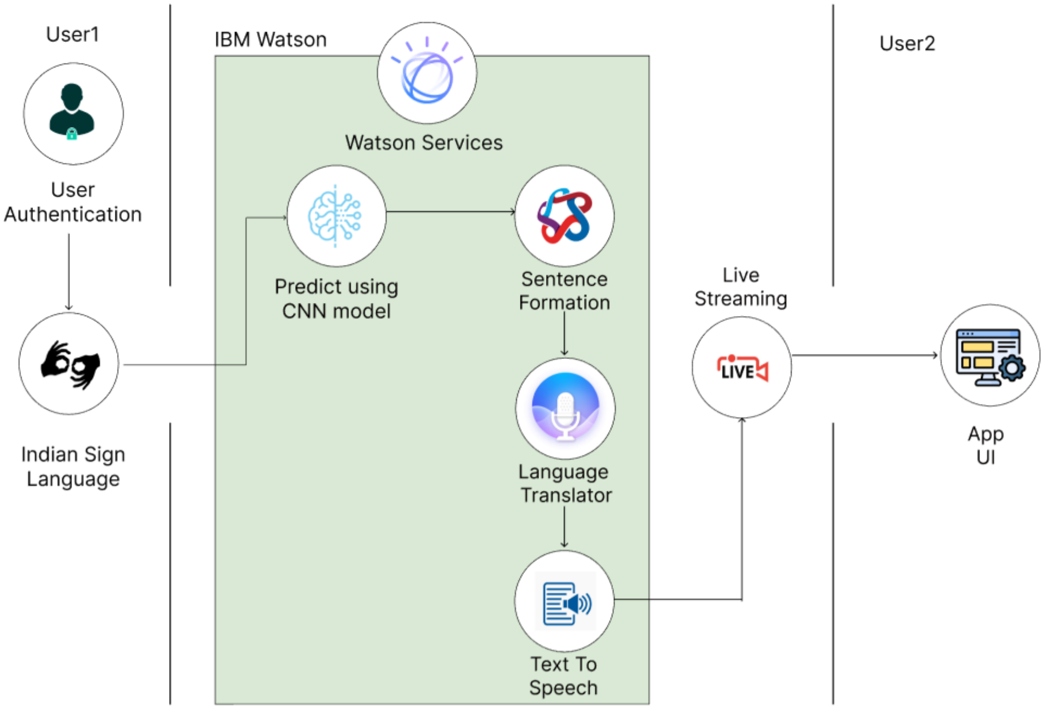


Fig. Technical architecture

5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Google		Medium	Sprint-1
	Login	USN-4	As a user, I can log into the application by entering email & password		High	Sprint-1
		USN-5	As a user, I can log into the application by Gmail		Medium	Sprint 2
		USN-6	As a user, I can stream and predict the sign language		High	Sprint 3

		USN-7	As a user, I can see the sign language into a stream of words.		Medium	Sprint 3
		USN-8	As a user, I can be able to see the resulted sentence into any language		Medium	Sprint 3
		USN-9	As a user, I can translate my sign language into speech		High	Sprint 3
		USN-10	As a user, I can translate my sign language into any speech language		High	Sprint 3
		USN-11	As a user, I can get feedback from the predicted result		High	Sprint 4
	Feedback	USN-12	As a user, I can invite my friends and family to use the application.		Medium	Sprint-1

CHAPTER 6
PROJECT PLANNING AND SCHEDULE

6.1 Sprint Planning 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.	2	High	Surya V
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	Vijay A
Sprint-1		USN-3	As a user, I can register for the application through Gmail	2	Medium	Surya V
Sprint-2	Login	USN-4	As a user, I can log into the application by entering email & password	1	High	Zafar Sadik H D
Sprint 2		USN-5	As a user, I can log in to the application by Gmail	1	Medium	Vijay A

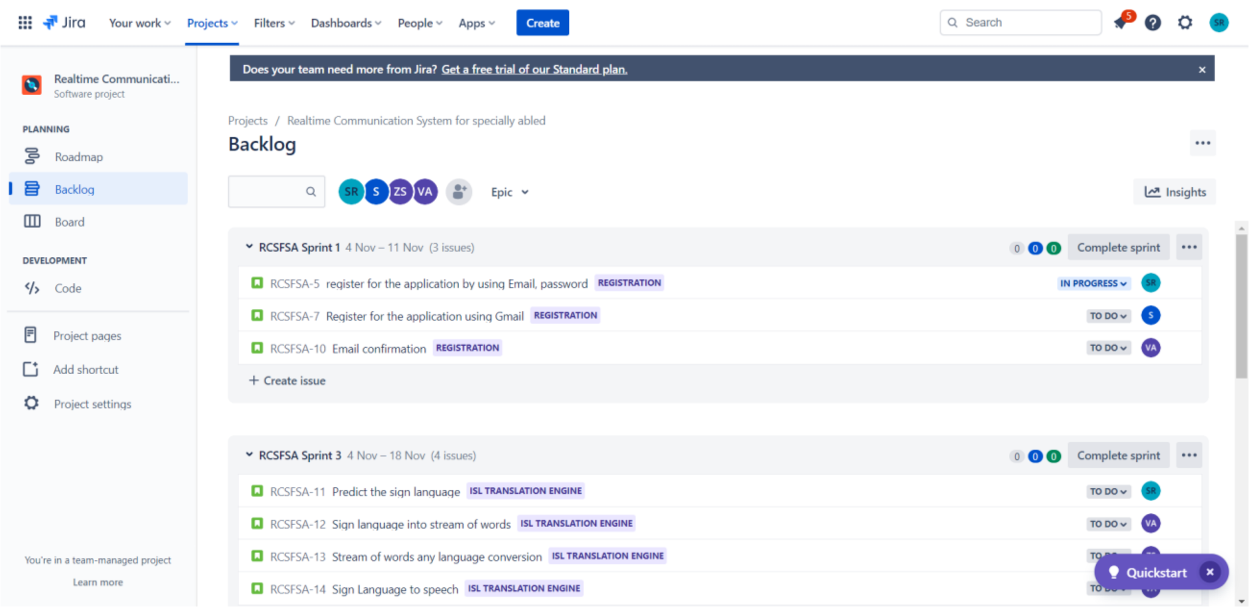
Sprint 3	Dashboard	USN-6	As a user, I can stream and predict the sign language	3	High	Srivivek nath S R
Sprint 3		USN-7	As a user, I can see the sign language into a stream of words	3	Medium	Srivivek nath S R
Sprint 3		USN-8	As a user, I can translate my sign language into speech	3	High	Vijay A
Sprint 4	Integration	USN-9	As a user, I can get the application as a whole	3	High	Zafar Sadik H D

6.2 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	5	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	2	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	9	12 Nov 2022

Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	3	19 Nov 2022

6.3 REPORTS FROM JIRA



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

Projects / Realtime Communication System for specially abled

Backlog

SR S ZS VA

RCSFSA Sprint 1 4 Nov – 11 Nov (3 issues) 0 0 0

- RCSFSA-5 register for the application by using Email, password REGISTRATION IN PROGRESS SR
- RCSFSA-7 Register for the application using Gmail REGISTRATION TO DO S
- RCSFSA-10 Email confirmation REGISTRATION TO DO VA

+ Create issue

RCSFSA Sprint 3 4 Nov – 18 Nov (4 issues) 0 0 0

- RCSFSA-11 Predict the sign language ISL TRANSLATION ENGINE TO DO SR
- RCSFSA-12 Sign language into stream of words ISL TRANSLATION ENGINE TO DO VA
- RCSFSA-13 Stream of words any language conversion ISL TRANSLATION ENGINE
- RCSFSA-14 Sign Language to speech ISL TRANSLATION ENGINE

CHAPTER 7
CODING & SOLUTIONING

7.1 Login

The login feature in the application is responsible for authenticating the user in the application. The user is authenticated through either their email or google account. Once after successful registration of the user’s email, the entry is recorded in the database and the user can use their credentials to login into the application.

```
 const onSuccess = (res) => {
    console.log("LOGIN SUCCESS! Current user: ",res.profileObj);
    return navigate("/dashboard", {state:{authProvider:'google',response :res.profileObj}})
  }
  const onFailure = (res) => {
    console.log("LOGIN FAILED! res : ",res);
  }
  const {signIn} = useGoogleLogin({
    onSuccess,
    onFailure,
    clientId,
    isSignedIn:true,
    accessType:'offline',
  })
  const [isShown,setShown] = useState(false);
  const handleCLick = evennt => {
    setShown(current => !current)
  };
 const handleChange = (prop) => (event) => {
    setValues({ ...values, [prop]: event.target.value });
  };
  const handleClickShowPassword = () => {
    setValues({values,
      showPassword: !values.showPassword,
    });
  };
  const handleMouseDownPassword = (event) => {
    event.preventDefault();
  };
   const handleSubmit = (event) => {
    event.preventDefault();
    var form = document.getElementsByName("loginForm")[0]
    const data = new FormData(event.currentTarget);
    console.log({
      email: data.get('email'),
      password: data.get('password'),
    });
    const result = {
      "email" : data.get('email'),
      "password" : data.get('password')
    }
    console.log(JSON.stringify(result));
    postDetails(JSON.stringify(result),form,event)
  };
```

7.2 Forgot and reset password

This feature is used in case the user has forgotten their credentials. An One time password (OTP) is sent to the registered email of the user and the user is prompted to enter the same in the application. Once the OTP is entered by the user, it is verified and then the user is allowed to reset their password in case the correct OTP is entered.

```
async function postDetails(data,call){
        const response = await fetch("/api/"+call,{
            method: 'POST',
            headers: {
                'Content-Type' : 'application/json'
            },
            body : data
        }).then(response => response.text()).then(response => {
          console.log(response)
          const obj = JSON.parse(response)
          const key = Object.keys(obj)
          if(key == "success"){
            const successMessage = obj["success"];
            message = successMessage;
            console.log("Success" + response["success"])
            if(call=="validOTP"){
                navigate("/ChangePassword",{state:{body:forWardEmail}})
            }
          }
          if(key == "failure"){
            const failureMessage = obj["failure"];
            message = failureMessage;
            console.log("Message: " +message)
            console.log("Failure: " + obj["failure"])
          }
        })
      }
```

7.3 Prediction using machine learning

The key feature of the application is the machine learning module that is used to predict the sign language that is captured through the camera of the user’s device. A Convolutional Neural Network (CNN) model is developed that takes input of the image captured and predicts the meaning of the sign language.

Preprocessing :

```
change_background_mp = mp.solutions.selfie_segmentation
change_bg_segment = change_background_mp.SelfieSegmentation()

def load_images_from_folder(folder):
    images = []
    for filename in os.listdir(folder):
        if filename.endswith(".jpg"):
            img = cv2.imread(os.path.join(folder, filename))
            img = cv2.cvtColor(img , cv2.COLOR_BGR2RGB)
            img = background_removal_mask(img)
            # img = cv2.cvtColor(img , cv2.COLOR_RGB2GRAY)
            # img = edge_detection(img)

            img = cv2.resize(img , (128,128))
            #img = img > 0.3
            if img is not None:
                images.append(img)
    return images

def lables_from_folder():
    label = []
    i = 0
    for folder in folders:
        for filename in os.listdir(folder):
            if filename.endswith(".jpg"):
                label.append(i)
        i = i+1
    return label

def background_removal_mask(image):
    RGB_sample_img = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
    result = change_bg_segment.process(RGB_sample_img)
    return result.segmentation_mask

#split dataset
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test =
train_test_split(X,y,test_size=0.30,shuffle=True,random_state=42)

#model*
model3 = Sequential()
model3.add(tf.keras.layers.Conv2D(64 ,(3,3) ,activation='relu',input_shape=(128,128,1)))
model3.add(tf.keras.layers.MaxPooling2D((2,2), strides=2))
model3.add(tf.keras.layers.Conv2D(32, (3, 3), activation="relu"))
model3.add(tf.keras.layers.MaxPooling2D((2, 2), strides=2))
model3.add(tf.keras.layers.Conv2D(16, (3, 3), activation="relu"))
model3.add(tf.keras.layers.MaxPooling2D((2, 2), strides=2))
model3.add(tf.keras.layers.Flatten())
model3.add(tf.keras.layers.Dense(8, activation="relu"))
model3.add(tf.keras.layers.Dense(35,activation="softmax"))
model3.compile(optimizer='adam',
              loss="categorical_crossentropy",
             metrics=['accuracy'])
datagen = ImageDataGenerator(zoom=0.2,width_shift_range=0.2, height_shift_range=0.2,
validation_split=0.25)
datagen.fit(X_train.reshape(29400, 128, 128, 1))
callbacks3=[EarlyStopping(patience=3,verbose=1,monitor='val_accuracy',mode='max'),tf.keras.
callbacks.ModelCheckpoint(filepath=checkpoint_path, save_weights_only=True,verbose=1)]
history3 = model3.fit(datagen.flow(X_train.reshape(29400, 128, 128, 1),y_train,batch_size=32,
                       shuffle=True,
                       sample_weight=None,
                       seed=100,
                       save_to_dir=None,
                      subset='training'),
                     epochs = 50,
                     validation_data = datagen.flow(X_train.reshape(29400, 128, 128, 1),
y_train,batch_size=8, subset='validation'),
                     callbacks=callbacks3)

#save model
model3.save('molde_sign_predict')

#load model
model = load_model('/content/drive/MyDrive/NTP/model_sign_predict')
```

7.4 Text to speech

The feature that is responsible for converting the text output that is predicted by the model to voice that is spoke through the speaker of the user's device

```
var msg = new SpeechSynthesisUtterance('Welcome to windows');
window.speechSynthesis.speak(msg);
```

7.5 Database Schema

A database schema defines how data is organized within a relational database; this is inclusive of logical constraints similar as table names, fields, data types, and the connections between these realities. Schemas generally use visual representations to communicate the armature of the database, getting the foundation for an association's data operation discipline. This process of database schema design is also known as data modeling.

```
def create_table():
    sql='''create table signLanguage(id int, first_name varchar(50),
     last_name varchar(50), birthDate varchar(50), email varchar(50), password varchar(100),
authProvider varchar(50) )'''
    stmt = ibm_db.prepare(IBM_DB_CONN,sql)
    ibm_db.execute(stmt)
```

CHAPTER 8
TESTING

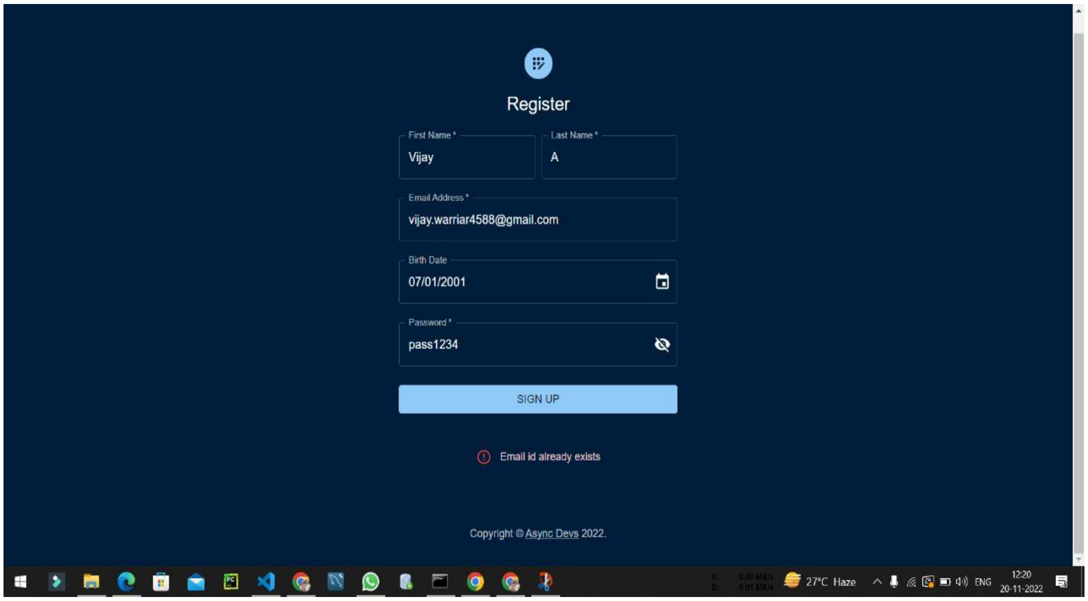
8.1 Test cases

A. Case 1 :-

a. Description : This flow tests whether/not the user is allowed to use the application without proper authentication.

b. Test : Accessing the webpage with wrong credentials

c. Screenshots :-



Register

First Name * Vijay

Last Name * A

Email Address * vijay.warriar4588@gmail.com

Birth Date 07/01/2001

Password * pass1234

Email id already exists

Copyright © Async Devs 2022.

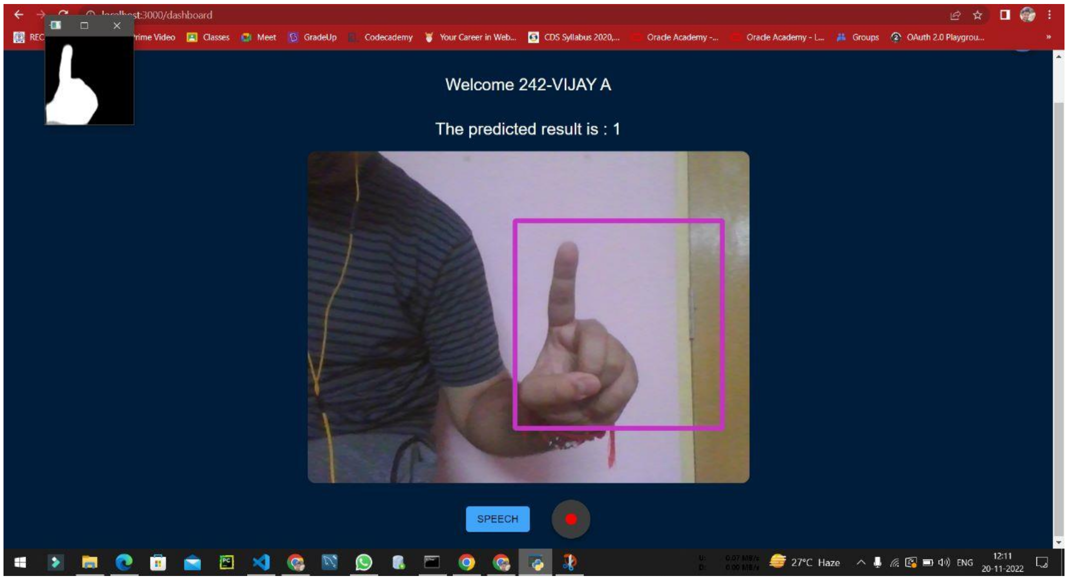
d. Result : The user is not allowed to access the webpage

B. Case 2 :-

a. Description : To test the speed and effectiveness of the program

b. Test : Imposing various actions all at a time

c. Screenshots :-



Welcome 242-VIJAY A

The predicted result is : 1

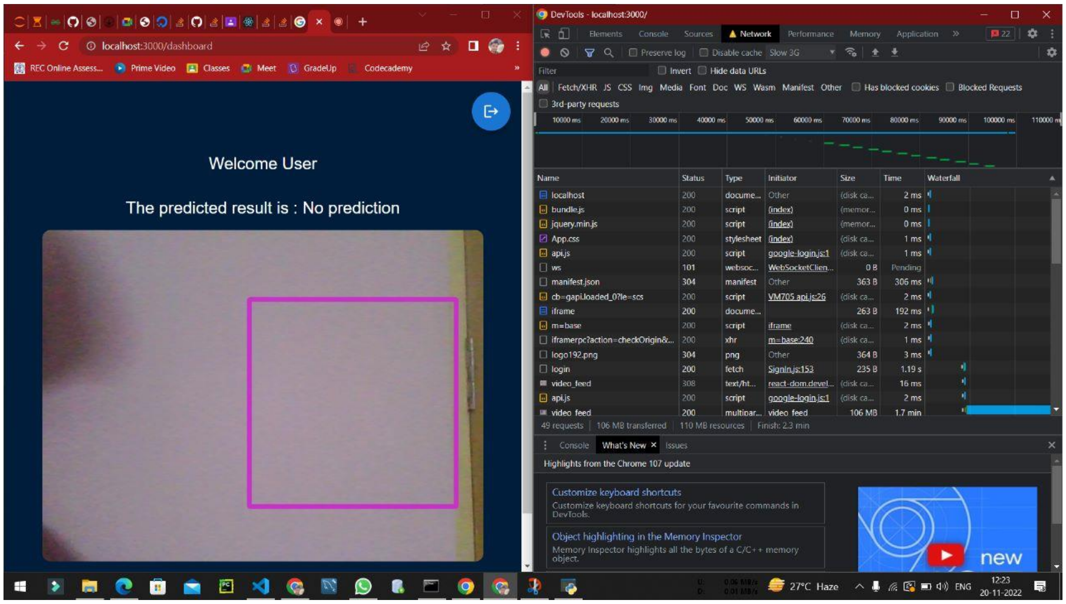
d. Result : The webpage is balanced and all the actions are performed successfully

C. Case 3 :-

a. Description : To test the application satisfies the acceptable performance under low network bandwidth program

b. Test : Throttling the network bandwidth of the user

c. Screenshots :-



Welcome User

The predicted result is : No prediction

DevTools - localhost:3000/

Slow 3G

Name	Status	Type	Initiator	Size	Time
localhost	200	docume...	Other	(disk ca...	2 ms
bundle.js	200	script	(index)	(memor...	0 ms
jquery.min.js	200	script	(index)	(memor...	0 ms
App.css	200	stylesheet	(index)	(disk ca...	1 ms
api.js	200	script	google-login.js:1	(disk ca...	1 ms
ws	101	websoc...	WebSocketClien...	0 B	Pending
manifest.json	304	manifest	Other	363 B	306 ms
cb=gapi.loaded_0?le=scs	200	script	VM705 api.js:26	(disk ca...	2 ms
iframe	200	docume...		263 B	192 ms
m=base	200	script	iframe	(disk ca...	2 ms
iframerpc?action=checkOrigin&...	200	xhr	m=base:240	(disk ca...	1 ms
logo192.png	304	png	Other	364 B	3 ms
login	200	fetch	SignIn.js:153	235 B	1.19 s
video_feed	308	text/ht...	react-dom.devel...	(disk ca...	16 ms
api.js	200	script	google-login.js:1	(disk ca...	2 ms
video_feed	200	multipar...	video_feed	106 MB	1.7 min

49 requests | 106 MB transferred | 110 MB resources | Finish: 2.3 min

Highlights from the Chrome 107 update

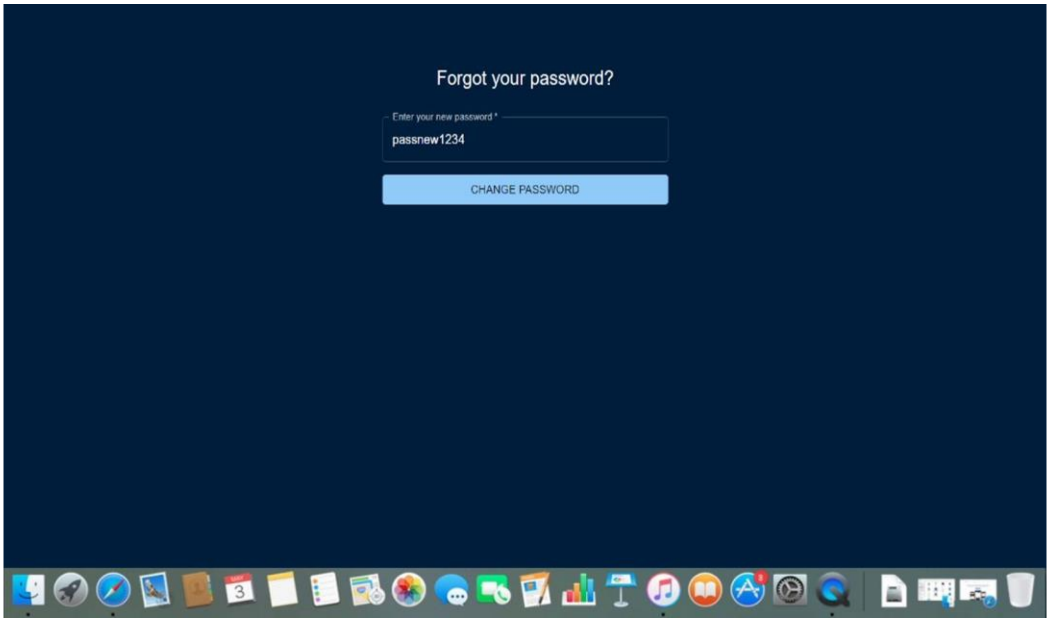
Customize keyboard shortcuts
Customize keyboard shortcuts for your favourite commands in DevTools.

Object highlighting in the Memory Inspector
Memory Inspector highlights all the bytes of a C/C++ memory object.

d. Result : The webpage loads successfully with acceptable latency

D. Case 4 :-

a. Description : To test that the application performs well despite the underlying Operating systems and architectures
b. Test : Using the application in various devices with different Operating systems and architectures
c. Screenshots :-



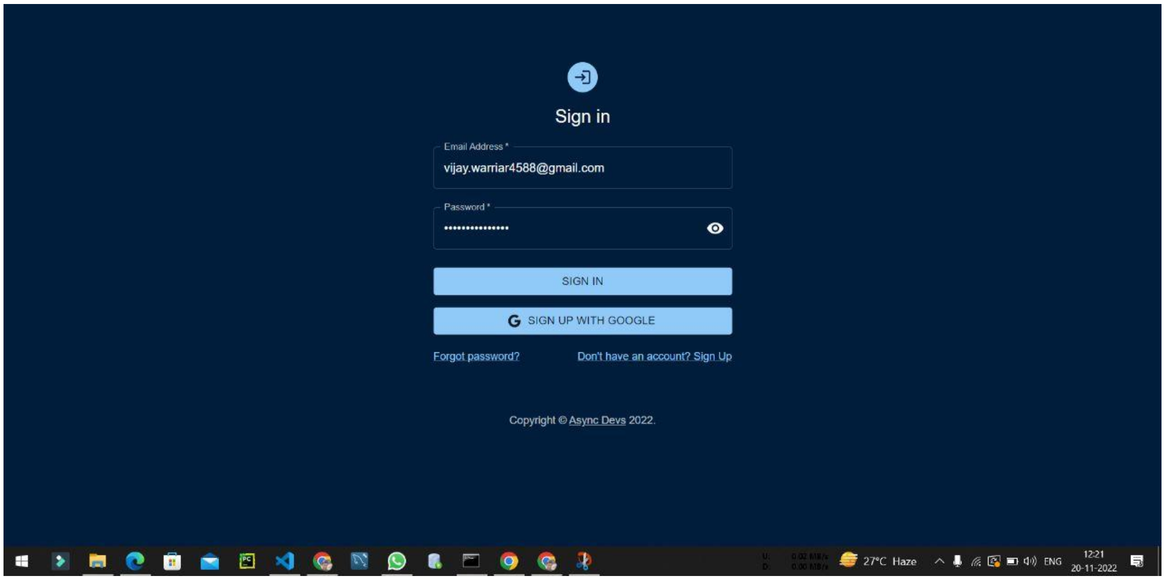
Forgot your password?

Enter your new password * passnew1234

d. Result : Despite the underlying architecture, the application satisfies the functional and nonfunctional requirements of the system.

8.2 User acceptance testing

A. Description : To test the application in the applicant's environment
B. Test : The application is tested on a real time basis in the system with commodity hardware configuration and acceptable network bandwidth of the targeted audience
C. Screenshots :-



Email Address * vijay.warriar4588@gmail.com

Password * •••••••••••••••

D. Result : The user feels positive about the application and the ease of use is comfortable

CHAPTER 9
RESULTS

9.1 Performance Metrics

A. Security:-

   a. The application does not leak any user sensitive data in the logs/consoles or prompts.
   b. SQL injection is prevented by validating the inputs from the user on both client as well as server side.

B. Reliability:-

   a. The system is load tested and the performance of the system under a considerable amount of deployment load is found to be satisfying
   b. The average rate of failure of the application is found to be negligible

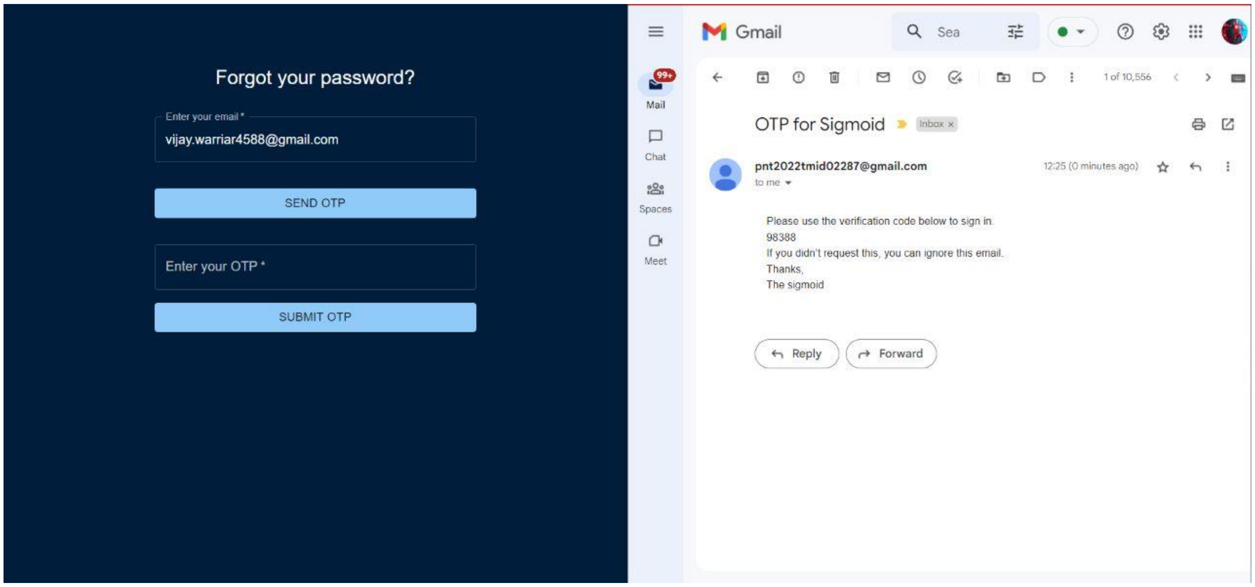
C. Performance:-

   a. The system is stress tested and the performance of the system is found to be acceptable under poor hardware configurations.
   b. Under poor network configurations, the application provided acceptable performance

D. Maintainability:-

   a. The code complexity of the system is found to be balanced.
   b. Proper comments to describe the functionality of the system is provided
   c. For future updates of the system, the code maintenance is not a show stopper

E. Screen Shots:-



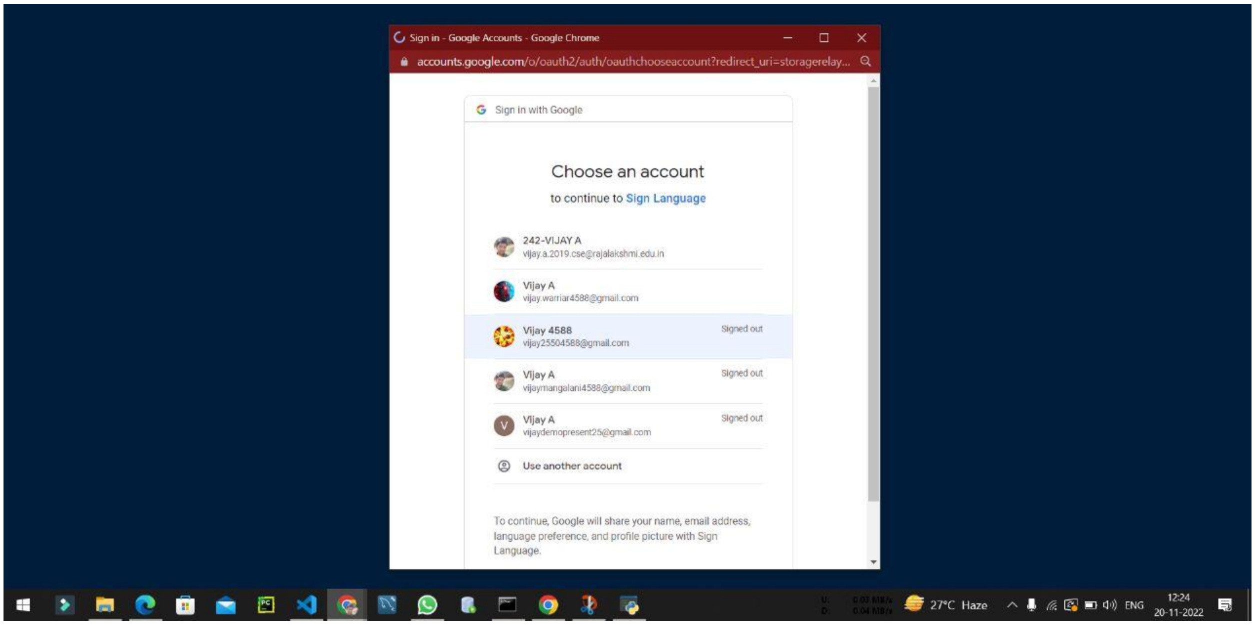
Forgot your password?

Enter your email *: vijay.warriar4588@gmail.com

OTP for Sigmoid

pnt2022tmid02287@gmail.com
to me
12:25 (0 minutes ago)

Please use the verification code below to sign in.
98388
If you didn't request this, you can ignore this email.
Thanks,
The sigmoid



Sign in - Google Accounts - Google Chrome

accounts.google.com/o/oauth2/auth/oauthchooseaccount?redirect_uri=storagerelay...

Sign in with Google

Choose an account

to continue to Sign Language

242-VIJAY A
vijay.a.2019.cse@rajalakshmi.edu.in

Vijay A
vijay.warriar4588@gmail.com

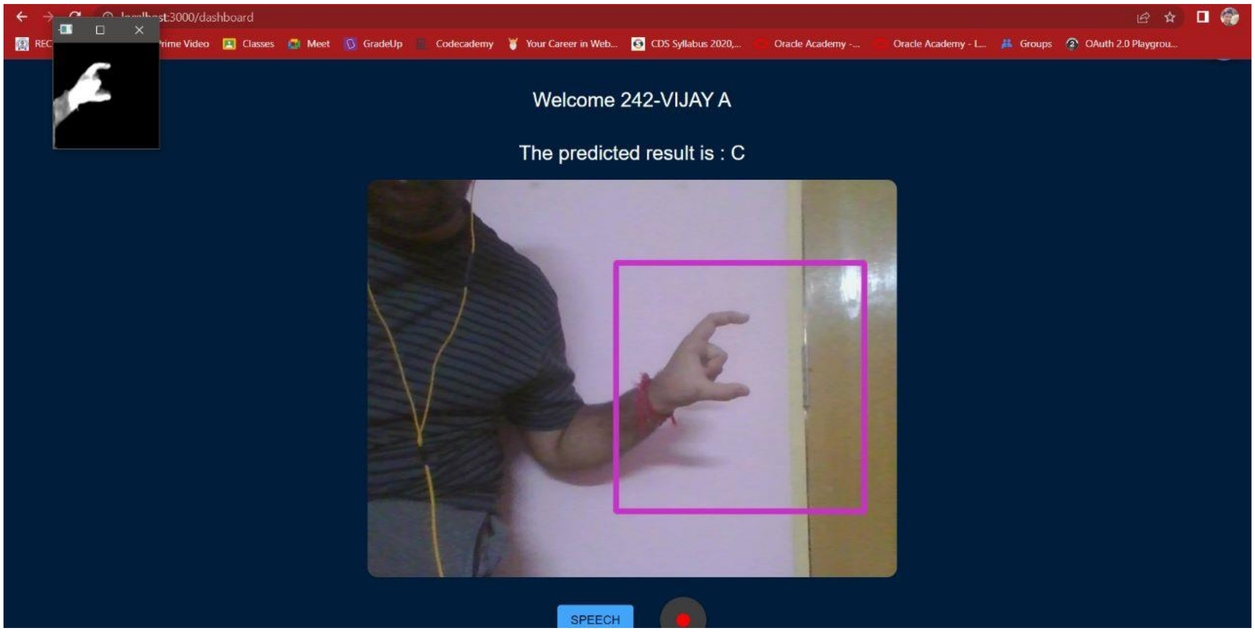
Vijay 4588 — Signed out
vijay25504588@gmail.com

Vijay A — Signed out
vijaymangalani4588@gmail.com

Vijay A — Signed out
vijaydemopresent25@gmail.com

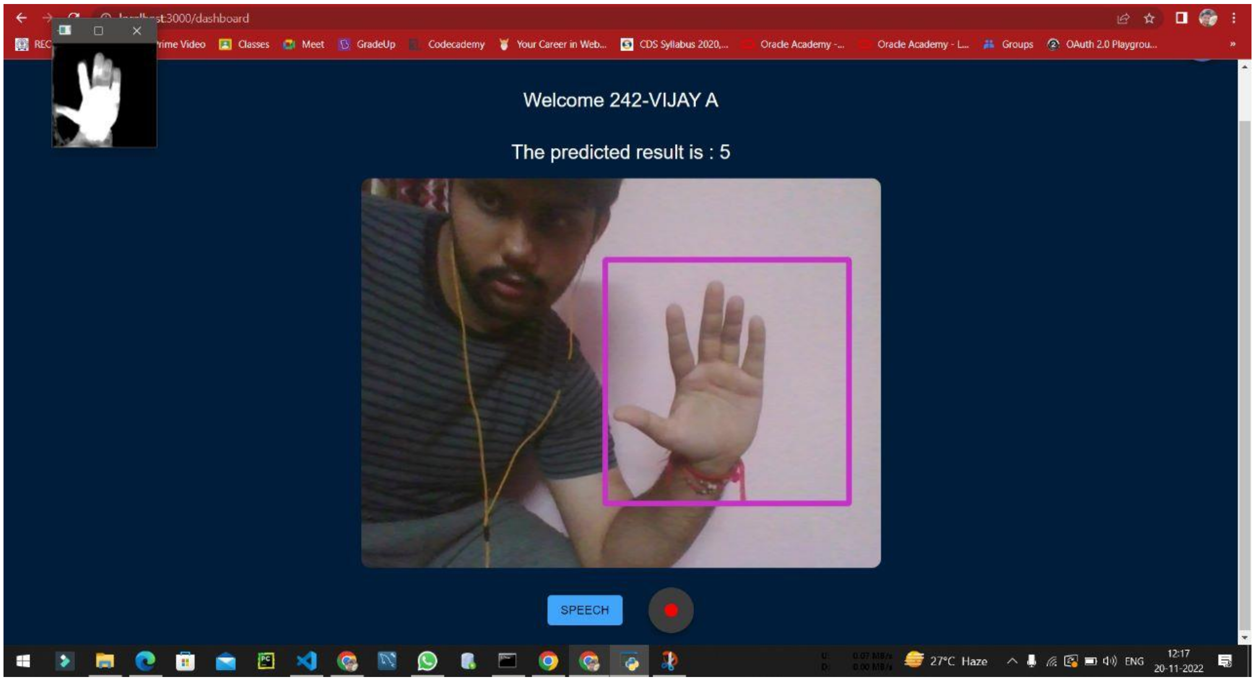
Use another account

To continue, Google will share your name, email address, language preference, and profile picture with Sign Language.



Welcome 242-VIJAY A

The predicted result is : C



Welcome 242-VIJAY A

The predicted result is : 5

CHAPTER 10
ADVANTAGES & DISADVANTAGES

10.1 Advantages

1. Converts video to speech instantaneously
2. Easily accessible using commodity hardwares.
3. Effective video streaming is available with minimum bandwidth requirements
4. Personalized experience through registration and login of user
5. Resetting password in case of user has forgotten or missed their credentials
6. User friendly such that the targeted customers can use the application effectively

10.2 Disadvantages

1. Model sometimes predicts slow, leading to increased latency
2. Server load must be effectively balanced using a proper load balancer

CHAPTER 11
CONCLUSION

The proposed communication system between Deaf and Dumb people and ordinary people are aiming for it when bridging the communication gap between two societies. Several work has been done before in this area, but the system provides a more reliable solution and serves its requirements in all aspects. The below strategies prove to be effective In terms of time and delicacy. Further advancements can be done in the perpetration of the prophet with other sign language similar as American subscribe Language, Accent recognition for different accentuations throughout Globe, recognition of feelings in sign language and language restatement.

CHAPTER 12
FUTURE SCOPE

Proposed systems Compass is related with the education of dumb peoples. Dumb people faces numerous problems when normal person couldn't understand their language. They were facing communication gaps with normal people. For communication between deaf person and an alternate person, a middleman is needed to restate the subscribed language of deaf person. But a middleman needs to know the sign language used by deaf people. But this isn't always possible since there are multiple sign languages for multiple languages. So to understand all sign languages, and gestures of deaf people by normal peoples this system is proposed. System gives an affair in the form of sound.

CHAPTER 13

APPENDIX

```
!pip install mediapipe
import pandas as pd
import numpy as np
import os
import matplotlib.pyplot as plt
import keras
from keras.utils import np_utils
import cv2
from keras.models import Sequential
from keras.layers import Dense
import tensorflow as tf
from tensorflow.keras.optimizers import RMSprop
from keras import layers
import mediapipe as mp

change_background_mp = mp.solutions.selfie_segmentation
change_bg_segment = change_background_mp.SelfieSegmentation()
sample_img = cv2.imread('/content/drive/MyDrive/NTP/HandSignTestDataset/test6.jpg')
plt.figure(figsize = [10, 10])
plt.title("Sample Image");plt.axis('off');plt.imshow(sample_img[:,:,::-1]);plt.show()
RGB_sample_img = cv2.cvtColor(sample_img, cv2.COLOR_BGR2RGB)
result = change_bg_segment.process(RGB_sample_img)
print(type(result.segmentation_mask))
plt.figure(figsize=[22,22])
plt.subplot(121);plt.imshow(sample_img[:,:,::-1]);plt.title("Original Image");plt.axis('off');
plt.subplot(122);plt.imshow(result.segmentation_mask, cmap='gray');plt.title("Probability Map");plt.axis('off');

def edge_detection(image):
   minValue = 70
   blur = cv2.GaussianBlur(image,(5,5),2)
   th3 = cv2.adaptiveThreshold(blur,255,cv2.ADAPTIVE_THRESH_GAUSSIAN_C,cv2.THRESH_BINARY_INV,11,2)
   ret, res = cv2.threshold(th3, minValue, 255, cv2.THRESH_BINARY_INV+cv2.THRESH_OTSU)
   return res
```

```
edge_sample_img = cv2.cvtColor(result.segmentation_mask, cv2.COLOR_RGB2GRAY)
plt.imshow(edge_detection(result.segmentation_mask))
binary_mask = result.segmentation_mask > 0.9
plt.figure(figsize=[22,22])
plt.subplot(121);plt.imshow(sample_img[:,:,::-1]);plt.title("Original Image");plt.axis('off');
plt.subplot(122);plt.imshow(binary_mask, cmap='gray');plt.title("Binary Mask");plt.axis('off');
def load_images_from_folder(folder):
    images = []
    for filename in os.listdir(folder):
        if filename.endswith(".jpg"):
            img = cv2.imread(os.path.join(folder, filename))
            img = cv2.cvtColor(img , cv2.COLOR_BGR2RGB)
            img = background_removal_mask(img)
            # img = cv2.cvtColor(img , cv2.COLOR_RGB2GRAY)
            # img = edge_detection(img)

            img = cv2.resize(img , (128,128))
            #img = img > 0.3
            if img is not None:
                images.append(img)
    return images
def lables_from_folder():
    label = []
    i = 0
    for folder in folders:
        for filename in os.listdir(folder):
            if filename.endswith(".jpg"):
                label.append(i)
        i = i+1
    return label

def background_removal_mask(image):
    RGB_sample_img = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
    result = change_bg_segment.process(RGB_sample_img)
    return result.segmentation_mask

def edge_detection(image):
    minValue = 70
    blur = cv2.GaussianBlur(image,(5,5),2)
    th3 =
cv2.adaptiveThreshold(blur,255,cv2.ADAPTIVE_THRESH_GAUSSIAN_C,cv2.THRESH_BIN
ARY_INV,11,2)
    ret, res = cv2.threshold(th3, minValue, 255,
cv2.THRESH_BINARY_INV+cv2.THRESH_OTSU)
    return res

root_folder = "/content/drive/MyDrive/Dataset/data"
folders = [os.path.join(root_folder, x) for x in ('1', '2',
'3','4','5','6','7','8','9','A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','
Y','Z',)]
all_images = [img for folder in folders for img in load_images_from_folder(folder)]

labels = lables_from_folder()

"""# Converting images and labels into np array"""

X = np.array(all_images)
y = np.array(labels)

print(X.shape)
print(y.shape)

X = np_utils.normalize(X)

# y = np_utils.to_categorical(y)

y = pd.get_dummies(y, columns =
[0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34
])
print(y)

# y = keras.utils.to_categorical(y, num_classes=35)

y

"""# Split the dataset into train and test set"""

from sklearn.model_selection import train_test_split
```

```
X_train, X_test, y_train, y_test =
train_test_split(X,y,test_size=0.30,shuffle=True,random_state=42)

#X_train,X_val,y_train,y_val =
train_test_split(X_train,y_train,test_size=0.25,shuffle=True,random_state=42)

alpha = [chr(c) for c in range(65,91)]
un,count = np.unique(y_train,return_counts=True)
j=1
plt.figure(figsize=(20,20))
for i in un:
    plt.subplot(7,4,j)
    plt.imshow(X_train[np.where(y_train == np.array(i))[0][0]])
    plt.axis('off')
    j=j+1

plt.imshow(X[24000],cmap="gray")
plt.show()

Char_Map = {0: '1', 1: '2', 2: '3', 3: '4', 4: '5', 5: '6', 6: '7', 7: '8', 8: '9', 9: 'A', 10: 'B', 11: 'C', 12: 'D',
            13: 'E', 14: 'F', 15: 'G', 16: 'H', 17: 'I', 18: 'J', 19: 'K', 20: 'L', 21: 'M', 22: 'N', 23: 'O', 24:
'P',
            25: 'Q', 26: 'R', 27: 'S', 28: 'T', 29: 'U', 30: 'V', 31: 'W', 32: 'X', 33: 'Y', 34: 'Z'}

import tensorflow as tf
print("Num GPUs Available: ", len(tf.config.experimental.list_physical_devices('GPU')))

physical_devices = tf.config.list_physical_devices()
print("Num GPUs:", len(physical_devices))
print(physical_devices)

X_train.shape

model3 = Sequential()

##Adding Layers to Model

model3.add(tf.keras.layers.Conv2D(64 ,(3,3) ,activation='relu',input_shape=(128,128,1)))
model3.add(tf.keras.layers.MaxPooling2D((2,2), strides=2))
model3.add(tf.keras.layers.Conv2D(32, (3, 3), activation="relu"))
```

```
model3.add(tf.keras.layers.MaxPooling2D((2, 2), strides=2))
model3.add(tf.keras.layers.Conv2D(16, (3, 3), activation="relu"))
model3.add(tf.keras.layers.MaxPooling2D((2, 2), strides=2))
model3.add(tf.keras.layers.Flatten())
model3.add(tf.keras.layers.Dense(8, activation="relu"))
model3.add(tf.keras.layers.Dense(35,activation="softmax"))


#Compiling model
model3.compile(optimizer='adam',
              loss="categorical_crossentropy",
             metrics=['accuracy'])



import os
from tensorflow.keras.callbacks import EarlyStopping
from tensorflow.keras.preprocessing.image import ImageDataGenerator

checkpoint_path = "training_1/cp.ckpt"
checkpoint_dir = os.path.dirname(checkpoint_path)

datagen = ImageDataGenerator(                              zoom_range=0.2,
                              width_shift_range=0.2,
                              height_shift_range=0.2, validation_split=0.25)
datagen.fit(X_train.reshape(29400, 128, 128, 1))


callbacks3=[EarlyStopping(patience=3,verbose=1,monitor='val_accuracy',mode='max'),tf.keras.
callbacks.ModelCheckpoint(filepath=checkpoint_path, save_weights_only=True,verbose=1)]

history3 = model3.fit(datagen.flow(X_train.reshape(29400, 128, 128, 1),y_train,batch_size=32,
                        shuffle=True,
                        sample_weight=None,
                        seed=100,
                        save_to_dir=None,
                        subset='training'),
                       epochs = 50,
                       validation_data = datagen.flow(X_train.reshape(29400, 128, 128, 1),
y_train,batch_size=8, subset='validation'),
                    callbacks=callbacks3)

X_train.shape

model3.save('molde_sign_predict')

from keras.models import load_model

model = load_model('/content/drive/MyDrive/NTP/model_sign_predict')

# y_pred_init = model.predict(X_test)

# y_pred = []

# for pred in y_pred_init:
#   loc = np.where(pred == max(pred))
#   y_pred.append(loc[0][0])
#   print(Char_Map[loc[0][0]])

score = model.evaluate(X_test, y_test, verbose = 0)


print(score*100)

test_images.shape

test_images = load_images_from_folder('/content/')

test_images = np_utils.normalize(test_images)

i=0

print(test_images[i].reshape(128,128).tolist()[5])

plt.imshow(test_images[i])
plt.show()

pred_ex = model.predict(test_images[i].reshape(1, 128, 128, 1))

from keras.models import load_model
```

```
y_pred_test = []

for pred in pred_ex:
  loc = np.where(pred == max(pred))

  print(Char_Map[loc[0][0]])

pred_ex
```

App.py

```
from flask import Flask, request, Response
from ibm_cloud_sdk_core.authenticators import IAMAuthenticator
from ibmcloudant.cloudant_v1 import CloudantV1,Document
from ibm_cloud_sdk_core import ApiException
import cv2
import uuid
import ibm_db,ibm_db_dbi
import hashlib
import smtplib, random
from email.message import EmailMessage
import mediapipe as mp
import cv2
import mediapipe as mp
from keras.models import load_model
import numpy as np
import json
import cv2
otp = 1234
change_background_mp = mp.solutions.selfie_segmentation
change_bg_segment = change_background_mp.SelfieSegmentation()
model = load_model('model_sign_predict')
loc = None
Char_Map = {0: '1', 1: '2', 2: '3', 3: '4', 4: '5', 5: '6', 6: '7', 7: '8', 8: '9', 9: 'A', 10: 'B', 11: 'C', 12: 'D',
              13: 'E', 14: 'F', 15: 'G', 16: 'H', 17: 'I', 18: 'J', 19: 'K', 20: 'L', 21: 'M', 22: 'N', 23: 'O', 24:
'P',  25: 'Q', 26: 'R', 27: 'S', 28: 'T', 29: 'U', 30: 'V', 31: 'W', 32: 'X', 33: 'Y', 34: 'Z'}
```

```
app = Flask(__name__)
IBM_DB_CONN = ibm_db.connect("DATABASE=bludb;HOSTNAME=b1bc1829-6f45-4cd4-bef4-10cf081900bf.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32304;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=pnf42623;PWD=F581rHVxDEmhkgML;", "", "")
conn = ibm_db_dbi.Connection(IBM_DB_CONN)

def predict(img):
    prediction = model.predict(img.reshape(1, 128, 128, 1))
    global loc
    #print(prediction)
    for pred in prediction:
        loc = np.where(pred == max(pred))
    print(Char_Map[loc[0][0]])

def load_image(img):

    img = cv2.cvtColor(img , cv2.COLOR_BGR2RGB)
    img = background_removal_mask(img)
    img = cv2.resize(img , (128,128))
    #print(type(img))
    return img

def background_removal_mask(image):
    RGB_sample_img = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
    result = change_bg_segment.process(RGB_sample_img)
    return result.segmentation_mask

sampleNum = 1
class VideoCamera(object):
    def __init__(self):
        self.video = cv2.VideoCapture(0)
    def clear(self):
        print("In clear function")
        self.video.release()

    def get_frame(self):
        _, frame = self.video.read()
        # h, w, c = frame.shape
        upper_left = (300, 100)
        bottom_right = (600, 400)
        frame = cv2.flip(frame, 1)
        r = cv2.rectangle(frame, upper_left, bottom_right, (200, 50, 200), 5)
        rect_img = frame[upper_left[1] : bottom_right[1], upper_left[0] : bottom_right[0]]

        sketcher_rect = rect_img
        sketcher_rect = load_image(sketcher_rect)
        #print(type(sketcher_react))

        result = ''
        if sketcher_rect is not None:
            if cv2.waitKey(1) & 0xFF == ord('p'):
                predict(sketcher_rect)
            cv2.imshow('new_frame',sketcher_rect)



        ret, jpeg = cv2.imencode('.jpg',frame)

        return jpeg.tobytes()

ob = VideoCamera()

@app.route("/api/get_result",methods = ['GET','POST'])
def get_result():
    global loc
    print(loc)
    if loc is not None:
        return Response(Char_Map[loc[0][0]],mimetype="text")

    return Response("No precition",mimetype="text")

@app.route("/api/add", methods = ['GET','POST'])
def add():
    result = request.get_json()
    id = uuid.uuid4().int
    fname = result["firstName"]
    lname = result["lastName"]
    birthDate = result["birthDate"]
    email = result["email"]
    password = result["password"]
    authProvider = result["authProvider"]

    row = checkEmail(email)

    # fetch_stmt = f"""SELECT * FROM "PNF42623"."SIGNLANGUAGE" WHERE "EMAIL"
= '{email}';"""
    # print(fetch_stmt)
    # stmt = ibm_db.prepare(IBM_DB_CONN, fetch_stmt)
    # cur = conn.cursor()
    # cur.execute(fetch_stmt)
    # row = cur.fetchall()
    if(len(row)!=0):
        print("Email id already exists")
        return {"failure" : "Email id already exists"}
    else:
        password = getHashForPassword(password)
        sql=f"""INSERT INTO "PNF42623"."SIGNLANGUAGE"
VALUES('{fname}','{lname}','{birthDate}','{email}','{password}','{authProvider}')"""
        reg_user = ibm_db.exec_immediate(IBM_DB_CONN,sql)
        print("User Registered successfully")
        return {"success": "Details added successfully"}

def checkEmail(email):
    fetch_stmt = f"""SELECT * FROM "PNF42623"."SIGNLANGUAGE" WHERE "EMAIL" =
'{email}';"""
    print(fetch_stmt)
    # stmt = ibm_db.prepare(IBM_DB_CONN, fetch_stmt)
    cur = conn.cursor()
    cur.execute(fetch_stmt)
    row = cur.fetchall()
    return row

@app.route("/api/login", methods = ['POST'])
def login():
    result = request.get_json()
    email = result["email"]
    password = result["password"]

    row = checkEmail(email)
    print(row)
    if(len(row)!=0):
        print("Email ID validated")
        stored_pass = row[0][4]

        if(getHashForPassword(password) == stored_pass):
            print("Password success")
            print("User logged in successfully")
        else:
            print("User login failed due to invalid password")

    return {"success": "User Logged In Successfully"}

@app.route('/api/sendEmail',methods = ['POST'])
def sendEmail():
    result = request.get_json()
    sender_email = "pnt2022tmid02287@gmail.com"
    mailMessage = getMailMessage()
    print(mailMessage)
    message = generateEmail(sender_email,result["email"],mailMessage)
    return message

@app.route('/api/changePass',methods=['POST'])
def changePass():
    result = request.get_json()
    update_stmt = f"""UPDATE "PNF42623"."SIGNLANGUAGE" SET "PASSWORD" = '{result["newPass"]}' WHERE "EMAIL" = '{result["email"]}';"""
    ibm_db.exec_immediate(IBM_DB_CONN,update_stmt)
    return {"success":"Password changed successfully" }

@app.route('/api/validOTP',methods=['POST'])
def validOTP():
    result = request.get_json()
    print("Valid OTP: " + result["OTP"])
    global otp
    if(otp == int(result["OTP"])):
        return {"success":"OTP Validated successfully"}
    else:
        print(type(otp),int(result["OTP"]))
        return {"failure":"Incorrect OTP"}
```

```
def generateOTP():
    tempOTP = random.random()
    return round(tempOTP * 100000)


def generateMailContent(OTP):
    content = '''
    Please use the verification code below to sign in.
    {0}
    If you didn’t request this, you can ignore this email.
    Thanks,
    The sigmoid'''.format(OTP)
    return content

def getMailMessage():
    global otp
    otp = generateOTP()
    print("OTP in getMailMessage " + str(otp))
    content = generateMailContent(otp)
    return content

def generateEmail(sender_email, receiver_email, content):
    msg = EmailMessage()
    msg.set_content(content)
    sender_password = "quurwsaeefcwkvpa"
    msg['Subject'] = 'OTP for Sigmoid'
    msg['From'] = sender_email
    msg['To'] = receiver_email
    s = smtplib.SMTP('smtp.gmail.com', 587)
    s.starttls()
    s.login(sender_email, sender_password)
    s.send_message(msg)
    s.quit()
    return {"success":"Email sent successfully"}

def getHashForPassword(password):
    return hashlib.md5(password.encode()).hexdigest()

def isPasswordCorrect(oldPassword, newPassword):
    return oldPassword == newPassword


def create_table():
    sql='''create table signLanguage(id int, first_name varchar(50),
     last_name varchar(50), birthDate varchar(50), email varchar(50), password varchar(100),
authProvider varchar(50) )'''
    stmt = ibm_db.prepare(IBM_DB_CONN,sql)
    ibm_db.execute(stmt)
def gen(camera):
    while True:

        frame = camera.get_frame()

        if cv2.waitKey(1) & 0xFF == ord('q'):
            break

        yield (b'--frame\r\n'
               b'Content-Type: image/jpeg\r\n\r\n' + frame + b'\r\n\r\n')

        # predicted = result
        # yield {"frame_details" : frame_details,
        # "predicted" : predicted}

    camera.video.release()
    cv2.destroyAllWindows()

def releaseCamera(camera):
    camera.clear()

@app.route("/api/video_feed", methods=['GET'])
def video_feed():

    return Response(gen(ob),
                    mimetype='multipart/x-mixed-replace; boundary=frame')

@app.route("/api/release")
def release():
    return Response(releaseCamera(ob))
app.run(debug=True)
```

13.2 GitHub & Project Demo Link:

GitHub :

https://github.com/IBM-EPBL/IBM-Project-17957-1659677464

Project Demo Link: https://drive.google.com/file/d/1rofO9mGM3BFOa4PSJ-Z5D1AP0N7DJ47J/view?usp=sharing