ACT

(ANYONE CAN TALK)

By

Muhammad Usman Baig

Mufaddal Sadiq

Asad Ali

Submitted to Faculty of Department of Computer Software Engineering National University of Sciences and Technology, Islamabad in partial fulfilment for the requirements of a B.E Degree in Computer Software Engineering, July 2018

In the name of Allah, the Most Beneficent, the Most Merciful

ABSTRACT

Anyone Can Talk

Sign language is a natural way for communication between normal and the deaf and dumb people, but often they find difficulty in communicating with normal people as they don't understand their sign language.

Therefore, there always exists a language barrier. To overcome this barrier, we propose a device which can convert their hand gestures into voice which a normal person can understand. It will also help out the deaf and dumb people to understand what a normal person is saying by watching their sign language on a screen.

This device consists of a Wireless Glove, consisting of flex sensors and accelerometer. These sensors sense the movement of hands and fingers.

This system consists of a two way speech synthesizer software which converts these movements of hand into real time speech output and a display will give the text for the corresponding gesture.

While on the other hand it will convert the speech of the healthy person into sign language so that the patient can understand.

CERTIFICATE FOR CORRECTNESS AND APPROVAL

It is certified that work contained in the thesis – Anyone Can Talk (ACT) carried out by Muhammad Usman Baig, Asad Ali and MufaddalSadiq under supervision of Prof. Ayesha Naseer for partial fulfilment of Degree of Bachelor of Software Engineering is correct and approved.

Approved By

Prof. Ayesha Naseer

Department of CSE, MCS

Dated:

____

DECLARATION

No portion of the work presented in this dissertation has been submitted in support of another award or qualification either at this institution or elsewhere.

DEDICATION

To our parents, without whose support and cooperation, a work of this magnitude would not have been possible. To our supervisor, Prof. Ayesha Naseer who has given us great support and valuable suggestions throughout the implementation process.

ACKNOWLEDGEMENTS

There is no success without the will of ALLAH Almighty. We are grateful to ALLAH, who has given us guidance, strength and enabled us to accomplish this task. Whatever we have achieved, we owe it to Him, in totality. We are also grateful to our parents and family and well-wishers for their admirable support and their critical reviews. We would like to thank our supervisor Prof. Ayesha Naseer, for her continuous guidance and motivation throughout the course of our project. Without their help, we would have not been able to accomplish anything.

Table of Contents

List of Figures

Chapter 1: Introduction

1.1 Overview

Sign language is a natural way for communication between normal and the deaf and dumb people, but often they find difficulty in communicating with normal people as they don't understand their sign language.

Therefore, there always exists a language barrier. To overcome this barrier, we propose a device which can convert their hand gestures into voice which a normal person can understand. It will also help out the deaf and dumb people to understand what a normal person is saying by watching their sign language on a screen.

This device consists of a Wireless Glove, consisting of flex sensors and accelerometer. These sensors sense the movement of hands and fingers.

This system consists of a two way speech synthesizer software which converts these movements of hand into real time speech output and a display will give the text for the corresponding gesture.

While on the other hand it will convert the speech of the healthy person into sign language so that the patient can understand.

1.2 Problem Statement

This Application is aimed to provide a platform for the end user to edit images on an Android device.The purpose of this app is to give the end user easy access to various image extraction and image merging techniques with the help of an easy to use graphic user interface and minimal learning effort. Using this app the end user can create amazing breathtaking pictures which can then be saved to local storage or can then be shared among friends using the social media.

1.3 Approach

We are working with Image Extraction from a selected Image and Merging of the extracted piece of selected image on the base Image. It will be our algorithm that will be able to merge (clone) these 2 images seamless and effective manner .We aim to develop an algorithm that can merge 2 images in such a way that the output image is as realistic as possible.

1.4 Scope

The document only covers the requirement specifications for ACT. This application will be used by deaf and dumb people to communicate with healthy people in a normal fashion. The device will recognize and generate all the English alphabets and basic words .

For	Final Year Project and MCS Computer Software Engineering Department
What	To provide a device for converting sign language into speech and speech back into sign language , along with the following features: • Sensory gloves • Android App • Desktop app
The	Sensory gloves
Is	A wireless glove
That	Provides: • Conversion of motion signals to electric signals

1.5 Objectives

This system will be useful for dumb and deaf people to communicate with blind and normal people. This system will convert the sign language into voice and text which is easily understandable by blind and normal people. The sign language is translated into some text form, to facilitate the deaf people as well.

This system will work for both platforms i.e. PCs and Android to make it portable for people

This text will be displayed on LCD on PC and on mobile screens while using android app. In order to improve and facilitate the more gesture recognition, motion processing unit like wireless gloves can be installed which comprises of accelerometer as well and with the help sensor fusion technique, we can accommodate a number of other gestures as well for better and efficient communication.

1.6 Deliverables

Table 1

Sr.	Tasks	Deliverables
1	Literature Review	Literature Survey
2	Requirements Gathering	SRS Document
3	Application Design	Design Document (SDS)
4	Implementation	Implementation on computer with a live test to show the accuracy and ability of the project
5	Testing	Evaluation plan and test document
6	Deployment	Complete application along with necessary documentation

1.7Purpose of the document:

The purpose of this project is to enable the deaf and dumb to communicate with normal people without any communication barriers . So we propose a device which can convert their hand gestures into voice which a normal person can understand.

Chapter 2: Literature Review

There were a few projects that were based on the idea of ACT following is a detailed description of projects previously carried out in this context.

- Audio to audio translation from one language to another language for android and PC.
- Audio to text translation for android and PC.
- Text to text translation from one language to another for android and PC.

- Text to audio translation for android and PC.

Image ProcessingSolution :

Here the images of hand gestures are gone through following steps:

- Gray Scaling of image.
- Logical transformation of image.
- Edge Detection.

Sensory Gloves (Sign Speak):

- Here the gestures are recognized using gloves with sensors.
- Sensors gave different values for different gestures.
- These values are then translated into audio signals.

Hidden Markov Models:

- Markov models were used in image processing to give real time sign language recognition.
- It uses AI to guess the next gesture depending upon the previous gesture Hence it provides a faster solution to sign language recognition.

Color Based Tracking :

- It is a type of image processing in which the software is programmed to track only the skin color.
- Uses Algorithm to find the position of hands.
- It recognizes the gestures based on skin color.

Conclusion

Image Processing restricts the movements in front of the camera, any gesture outside the frame didn't count.Other existing solutions only provides translation of sign language into audio, they didn't translate audio into sign language.

So, we proposed a solution, where a deaf person can also know what a healthy person is saying, by translating audio into Sign LanguageAppearance changes could for instance also deal with the sharpness of objects of interest, thus allowing the user to make apparent changes of focus.

Chapter 3: Software Requirement Specification

3.1Introduction

The purpose of this part is to describe the project titled “Anyone Can Talk”. This part contains the functional and non-functional requirements of the project. It contains the guidelines for developers and examiners of the project.

The purpose of this project is to enable the deaf and dumb to communicate with normal people without any communication barriers . So we propose a device which can convert their hand gestures into voice which a normal person can understand.

3.2 Overall Description

3.2.1 Product Perspective

This product would enable the deaf and dumb to communicate with normal people without any communication barriers by converting their hand gestures into voice which a normal person can understand.

3.2.2 Product Features

The main features of the application are:

- The Sensory glove is connected with the application with the help of Bluetooth.
- Database of some words is maintained at back end of the application.
- When any particular hand gestures are made signals are generated with the help of sensors mounted on gloves.
- These particulars signals are sent to application for processing with the help of Bluetooth connectivity.
- These received signals or values of any particular gesture are now analyzed with pre esisting words in database.
- After finding its match with any word of database,it is converted into speech and easily hearable to healthy or normal persons.

3.2.3 User Classes and Characteristics

Following are the user classes and their brief description.

Summary Of User Classes:

The following section describes the types of users of the Speech Synthesizer Software Application.

Deaf And Dumb User:

The Deaf and Dumb User is the one who actually wants to communicate with the healthy or normal person with the help of hand glove mounted on his hand and this speech synthesizer software application.

Normal Person User:

The Normal or Healthy Person User is the one who would communicate with the deaf and dumb person without any communication barrier with this application.

3.2.4 Operating Environment

Hardware

The Speech Synthesizer Application operates with the following hardware:

- **SENSORY GLOVE:** A sensory glove is wore by a deaf and dumb person and all types of sensors (Flex Sensors and Accelerometer) are mounted on this glove.
- **FLEX SENSORS**: There are 13 flex sensors used on gloves. All types of hands and fingers bending and rotations are detected by these sensors and signal is sent to application with Bluetooth signals and corresponding speech is generated.
- **ACCELEROMETER:** Accelerometer sensor is mounted on glove which is used to detect the movements of hands. Any particular movement of hand is recorded and signal is sent to application with Bluetooth signals and further processing and conversion to speech is done.

Software

- Windows: 7, 8, 8.1, 10.
- IDE: NetbeansAnd Eclipse.
- Programming language: JAVA and Android

3.2.5 Design and Implementation Constraints

- Initially database of fifty words is maintained.

3.2.6 User Documentation

Following are the guides for the Speech Synthesizer Software Application:

- Technical usage manuals will be provided giving step to step guidance for using the software application.

3.3 External Interface Requirements

3.3.1 User Interfaces:

Both the android and desktop applications consist of two modules for user's help:

1. **Sign to Speech module**.

- The sign to speech module of desktop and android application consists of an interface which represent the data received from the sensory gloves into a text area which contains a button to start the speech to sign module.

2. **Speech to Sign module.**

- In speech to sign module of the desktop and android application we will display both the text and gestures symbols on the screen as the healthy person speaks.

3.3.2 Software Interfaces:

- To record audio data from the mic by the healthy person into the software.
- To send the audio to the server using a speech to text API to convert the spoken words into the written text.
- To convert the written text into gestures using a defined database containing words.
- To display these words along with gestures onto a screen.
- To get the signals from the gloves using Arduino with sensors by the deaf and dumb person.
- To convert these sign signals into words.

  To convert these words into audio data using a text to speech API..

3.3.3 Hardware Interfaces:

- Main hardware interface is the sensory glove which wirelessly communicate to the application.

- Speech data is collected through a mic and later sent to the speech to text API for further processing.

3.3.4 Communications Interfaces:

For communication between sensory gloves and android or desktop application we have to use Arduino Lily pad along with Bluetooth connection between application and gloves.

3.4 System Diagram

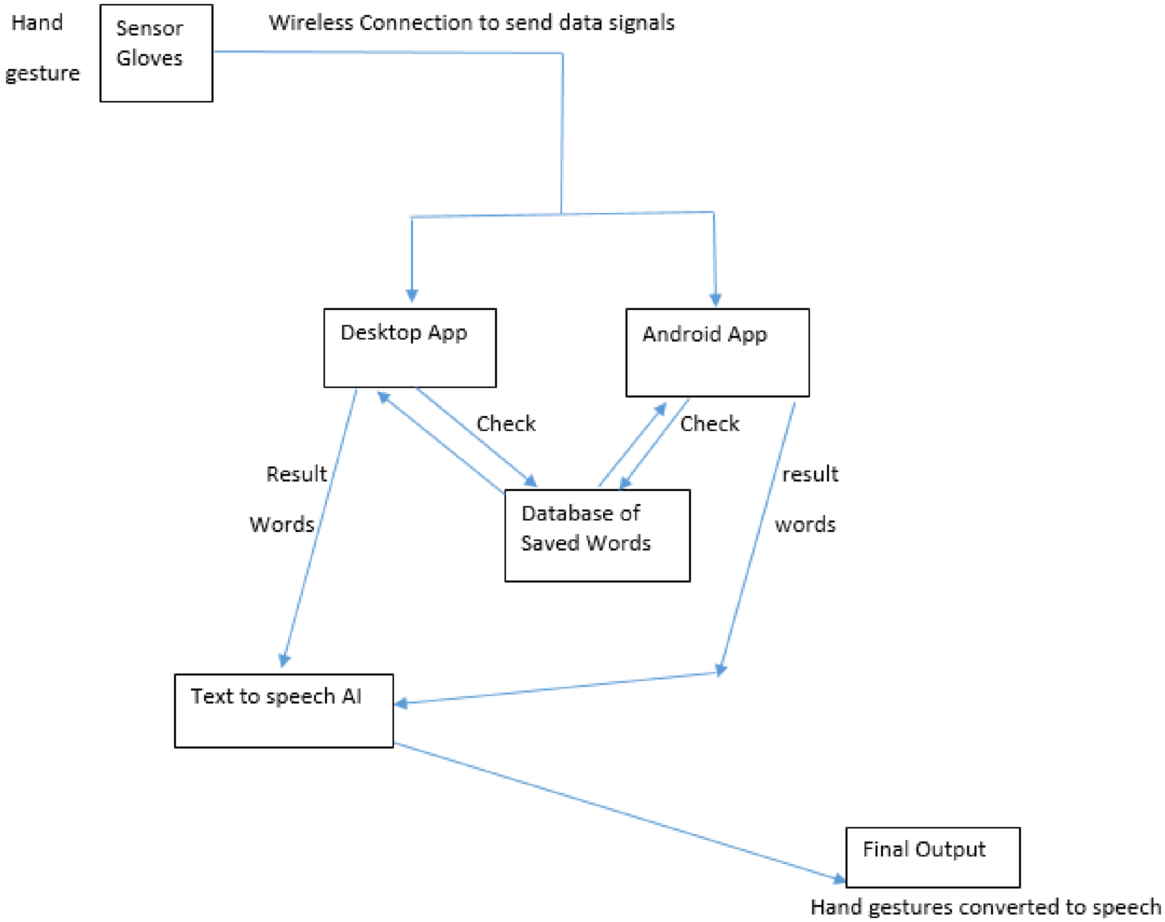


Figure 1 Signs To Speech

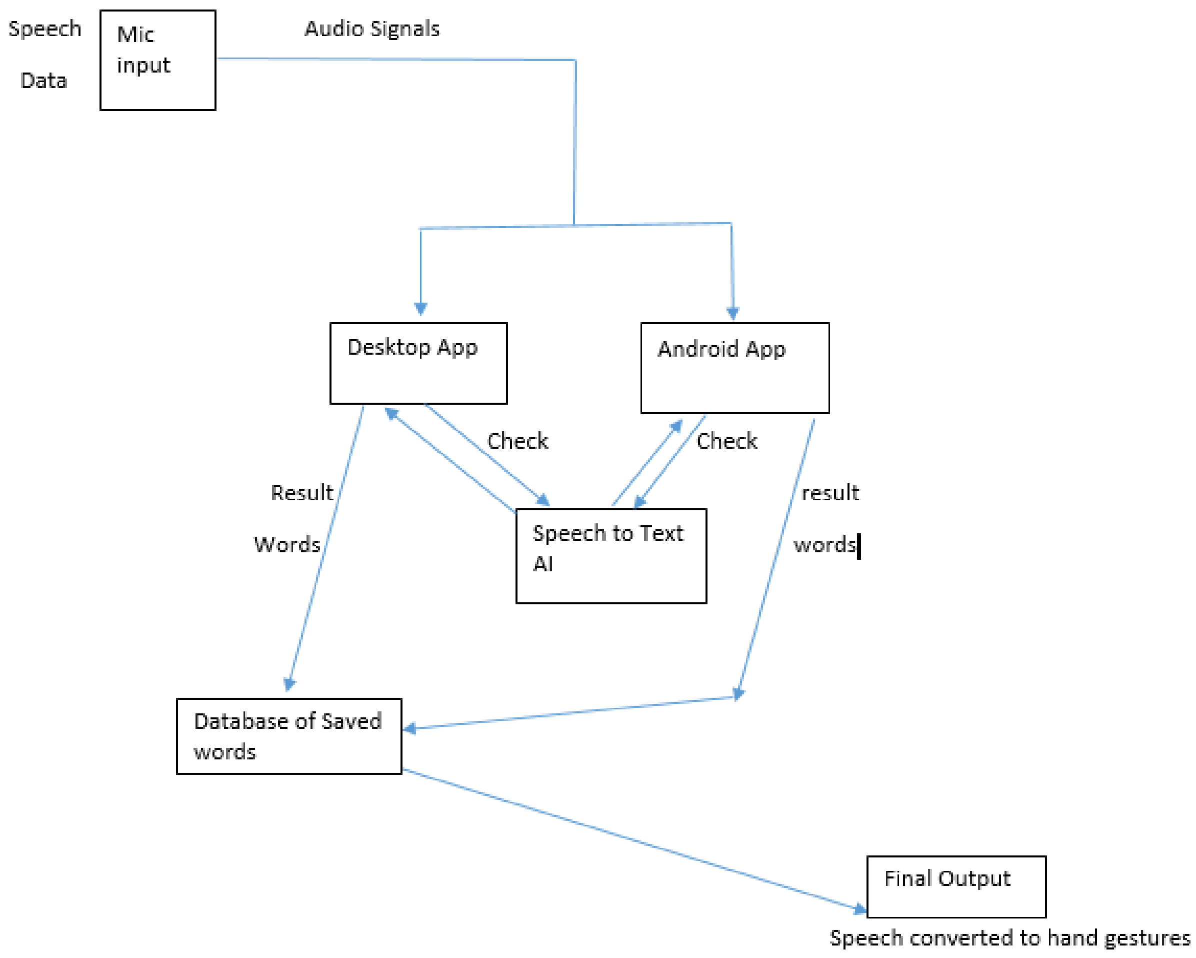


Figure 2 : Speech To Signs

3.5 Other Nonfunctional Requirements

3.5.1 Usability Requirements:

- On Desktop , the application will be an .exe file that can be used on any platform i-e it's not platform dependent. The platform can be any version of windows and operating system etc.
- It will be a user interface application allowing the user to select, browse and upload data for communication.
- User manuals will be provided in order to guide the user to make him/her familiar with the application.

3.5.2 Security Requirements:

- The data sent to the text to speech and speech to text api will be end to end encrypted and will not be stored on any external server.
- The application .exe file will run as administration requiring a username of administration and his/her password.
- It will be ensured that the data thus communicated will be protected against spoofing and sniffing.
- The ip addresses of the mobiles and desktop pcs using the API should not be shared with unauthoritized individuals.
- The android app will access microphone , Bluetooth and other phone utilites through users permission.

3.5.3 Safety Requirements:

- The gloves will be thoroughly sealed to prevent the user from experiencing any electrical shocks.
- Prevent using gloves under extreme weather conditions .
- This application is a fast and responsive program but however if large amount of data is being sent then there are chances that the application crashes but backup will be there.

3.5.4 Performance Requirements:

- The application is a fast program and will perform communication of data in minimum possible time
- If the process crash then it will recover in a very small amount of time.
- The quality of the data being transferred will be retained as much as possible.
- The bandwidth of the medium will be utilized in an efficient way in a minimum possible time.
- The application involves a statistical report of data communicating making it easier to monitor the system performance requirements.

3.5.5 ReliabilityRequirements:

- This application is a fast and responsive program but however if large amount of data is being sent then there are chances that the application crashes but backup will be there.
- If the process crash then it will recover in a very small amount of time.

3.5.6 SupportabilityRequirements:

- The desktop and android applications will constantly be updated to new versions after a continuous cycle of receiving feedback from the customers.

- The database of words and gestures will constantly be updated.

3.5.7 AvailabilityRequirements:

- The application can be run anytime on desktop and on pc , however it requires internet connection for the speech to text and text to speech module to work.
- The database for word and gestures will be available for 24/7 as it will be stored on the device itself.

3.5.8 Maintainability Requirements:

- The application will be updated and new features will be added to it as the time passes.

Chapter 4: Design and Development

4.1 Introduction:

Sign language is a natural way for communication between normal and the deaf and dumb people, but often they find difficulty in communicating with normal people as they don’t understand their sign language.

Therefore, there always exists a language barrier. To overcome this barrier, we propose a device which can convert their hand gestures into voice which a normal person can understand. It will also help out the deaf and dumb people to understand what a normal person is saying by watching their sign language on a screen.

This device consists of a Wireless Glove, consisting of flex sensors and accelerometer. These sensors sense the movement of hands and fingers.

This system consists of a two way speech synthesizer software which converts these movements of hand into real time speech output and a display will give the text for the corresponding gesture.

While on the other hand it will convert the speech of the healthy person into sign language so that the patient can understand.

4.1.1 Purpose:

The purpose of this project is to enable the deaf and dumb to communicate with normal people without any communication barriers . So we propose a device which can convert their hand gestures into voice which a normal person can understand.

4.2 Scope of the Development Project

The document only covers the requirement specifications for ACT. This application will be used by deaf and dumb people to communicate with healthy people in a normal fashion. The device will recognize and generate all the English alphabets and basic words .

For	Final Year Project and MCS Computer Software Engineering Department
What	To provide a device for converting sign language into speech and speech back into sign language , along with the following features: • Sensory gloves • Android App • Desktop app
The	Sensory gloves
Is	A wireless glove
That	Provides: • Conversion of motion signals to electric signals

4.3 SystemArchitectureDescription

Detailed description of system architecture and design pattern which this system is going to use is discussed later in the document in section 5 ‘Design Decisions and Tradeoffs’. This Section gives overview of application, its higher and lower levels details and user interfaces.

4.3.1Document Overview:

This document shows the design and working of ACT. It starts from higher level details for a non-technical reader to understand just by seeing the diagrams to the lower level details that aid the developer to code and understand other technical details of the application. Section 2 the **System Architecture Description** gives a detailed overview of the application. Section 2.1 **Structure and Relationships** shows the higher level details system working by the means of System Block, Activity, State Transition, and Use Case diagrams. Lower level details are described using the Class, Sequence diagrams and Structure Chart. Section 2.2 describes how the application is designed to curb the tendency of **User Interface Issues** and problems during User Interaction.

In Section 3, **Detailed Description of Component** is given to show the working of modules with low level details. It shows the purpose, function, subordinates, dependencies, interfaces, resources, processing and data of the components and their relationships with each other.

Section 4 shows the **Reuse and Relationship to other Products** i.e.; information about work done in the same project before and any reuse of the same work. The section also provides a key to reuse this system for further upgrades.

Section 5, **Design Decisions and Tradeoffs** shows the architecture style and design pattern of the application, while in the Section 6 the **Pseudo Code** of the components is given in for human reading rather than machine reading.

4.3.2Work Breakdown Structure:

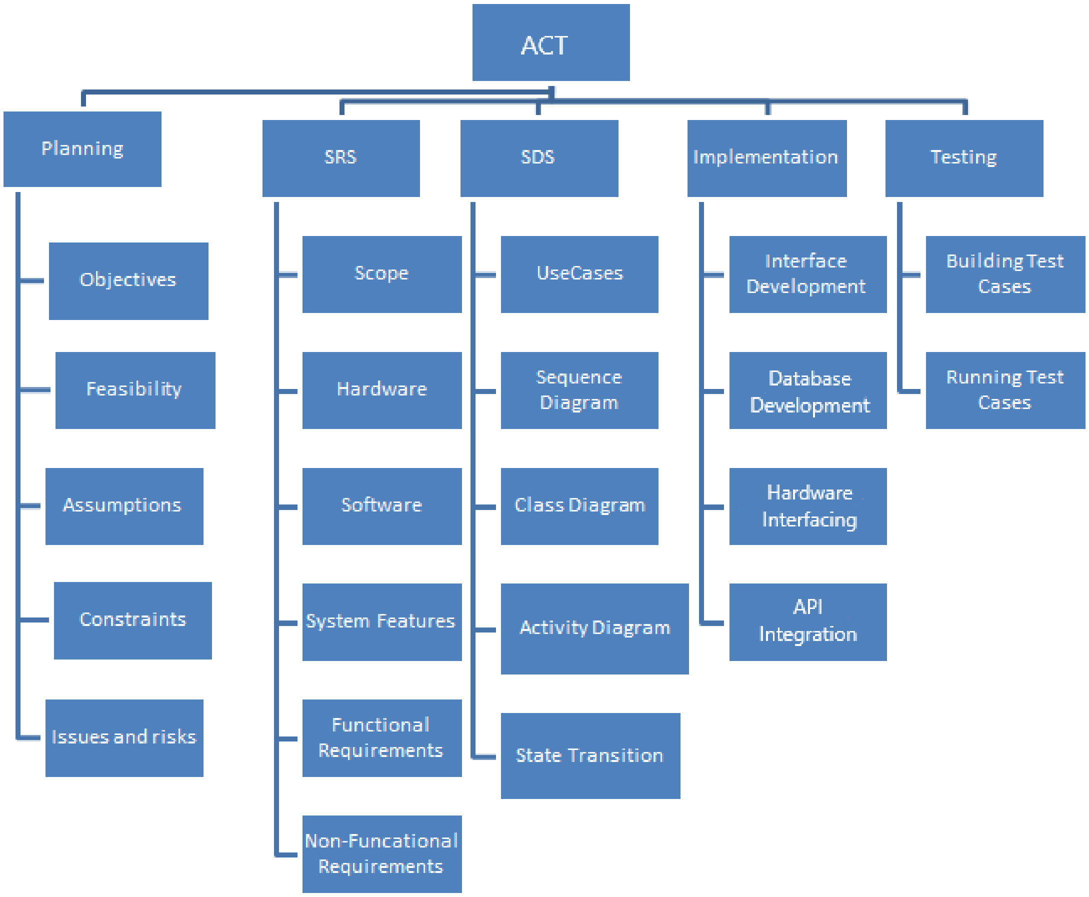


Figure 3 : Breakdown Structure

4.4 Structure and Relationships

This section covers the overall technical description of ACT. It shows the working of application in perspective of different point-of-views and also shows relationships between different components.

4.4.1 System Block Diagram

The diagram(s) show the higher level description of the application(s), generic working of the application(s) and interaction with the user.

User interacts with the Application using I/O devices such as Sensory Gloves and Microphone. Our Application then read the user inputs and provide output in the form of Gestures and Audio.

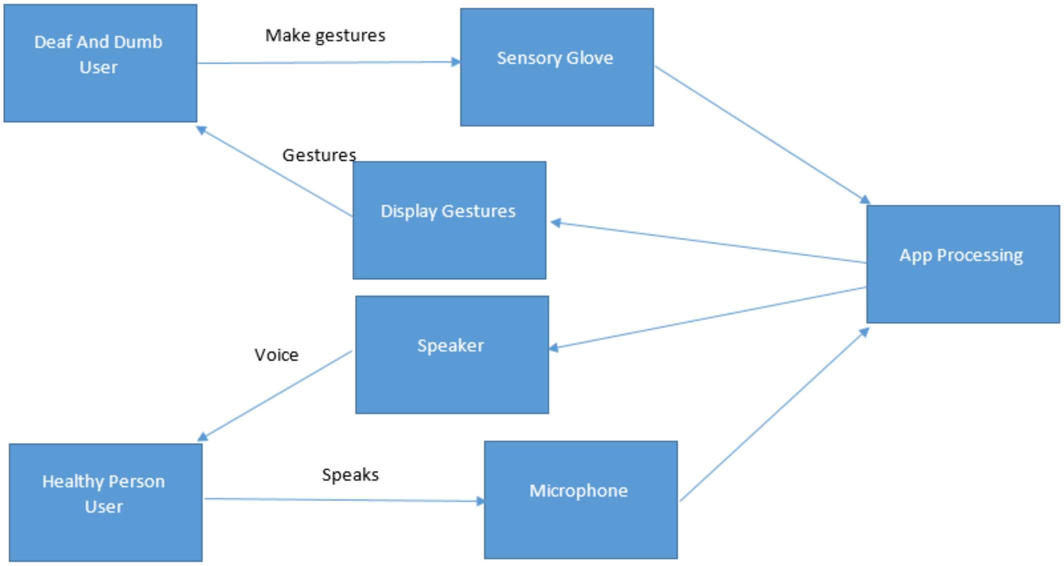


Figure 4 : Block Diagram

4.4.2 Component Diagram:

The main components are

- GUI App
- Hardwares
- User Input

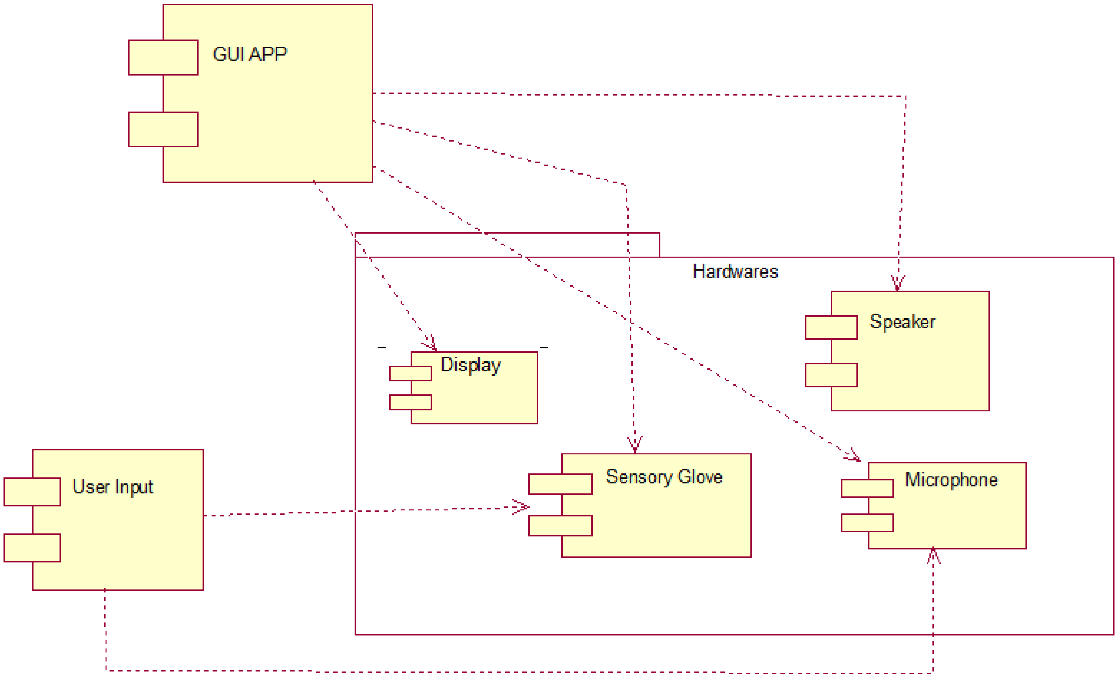


Figure 5 :Component Diagram

4.4.3 Use Case Diagram:

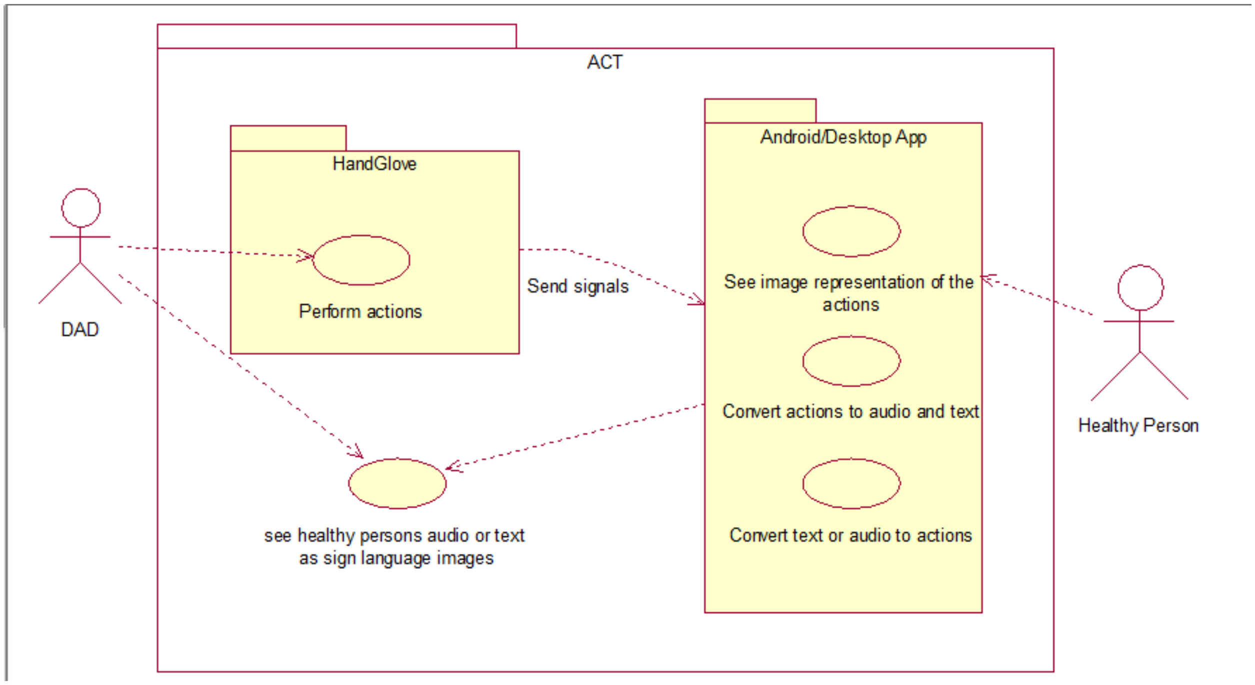

ACT
HandGlove
Perform actions
Send signals
Android/Desktop App
See image representation of the actions
Convert actions to audio and text
Convert text or audio to actions
see healthy persons audio or text as sign language images
DAD
Healthy Person

Figure 6 : Use Case Diagram

Actors:

Primary Actors:

- DAD (Deaf & dumb User)
- Healthy User

Secondary Actors:

- None

Use Cases:

- Perform Actions
- See image representations of the actions
- Convert text or audio to actions
- See healthy persons audio or text as sign language images
- Convert actions to audio or text

Use Case Description:

Use Case 1

Use Case	Perform Actions
Actors	DAD (Deaf and Dumb person)
Use Case Description	This use case will enable the deaf and dumb person to perform sign language actions using the glove
Normal Flow	The actions performed will be transmitted to the Desktop/Android app via Bluetooth signals , the app would recognize those actions , convert it to audio/text and display it to the healthy person.
Alternative Flow	App may not recognize the actions , DAD may have to perform the actions again or those actions may have to be added in the database
Pre-condition	The glove and the desktop/android app should be connected via bluetooth
Post Condition	Audio and text is displayed.
Includes	(i) Audio (Relative to sign language) (ii) Text (Relative to sign language)
Extends	N/A
Assumptions	The glove fits the hand size of DAD

Use Case 2

Use Case	See image representation of the actions
Actors	Healthy Persons
Use Case Description	This use case will allow the user to see digital images of the actions performed by DAD
Normal Flow	DAD performs actions , the app displays the same actions as digital images to

	the healthy person.
Alternative Flow	The app may fail to recognize the actions and display no or faulty images.
Pre-condition	The glove and the app are both connected via Bluetooth
Post Condition	Digital images of sign languages are displayed
Includes	Digital images of sign language actions
Extends	N/A
Assumptions	The glove fits the hand size of DAD

Use Case 3

Use Case	Convert text or audio to actions
Actors	Healthy person
Use Case Description	The use case will allow audio or text generated by healthy to be converted into sign language images for Deaf and Dumb person
Normal Flow	The audio and text are converted into relative digital images of sign language.
Alternative Flow	The app may: (i) Convert audio and text to faulty sign language images (ii) Return no images at all
Pre-condition	Mic should be enabled for receiving audio input
Post Condition	Digital images relative to audio or text input of sign language are displayed
Includes	Digital images
Extends	N/A
Assumptions	Mic is working properly

Use Case 4

Use Case	See healthy persons audio or text as sign language images
Actors	DAD
Use Case Description	Audio and text input of the healthy person would be available to DAD as digital images of sign language
Normal Flow	The audio and text are converted into relative digital images of sign language.
Alternative Flow	The app may: (i) Convert audio and text to faulty sign language images (ii) Return no images at all
Pre-condition	The database should have sign language images of relative audio and text .
Post Condition	Digital images are displayes
Includes	N/A
Extends	N/A
Assumptions	DAD can recognize digital images of sign language

Use Case 5

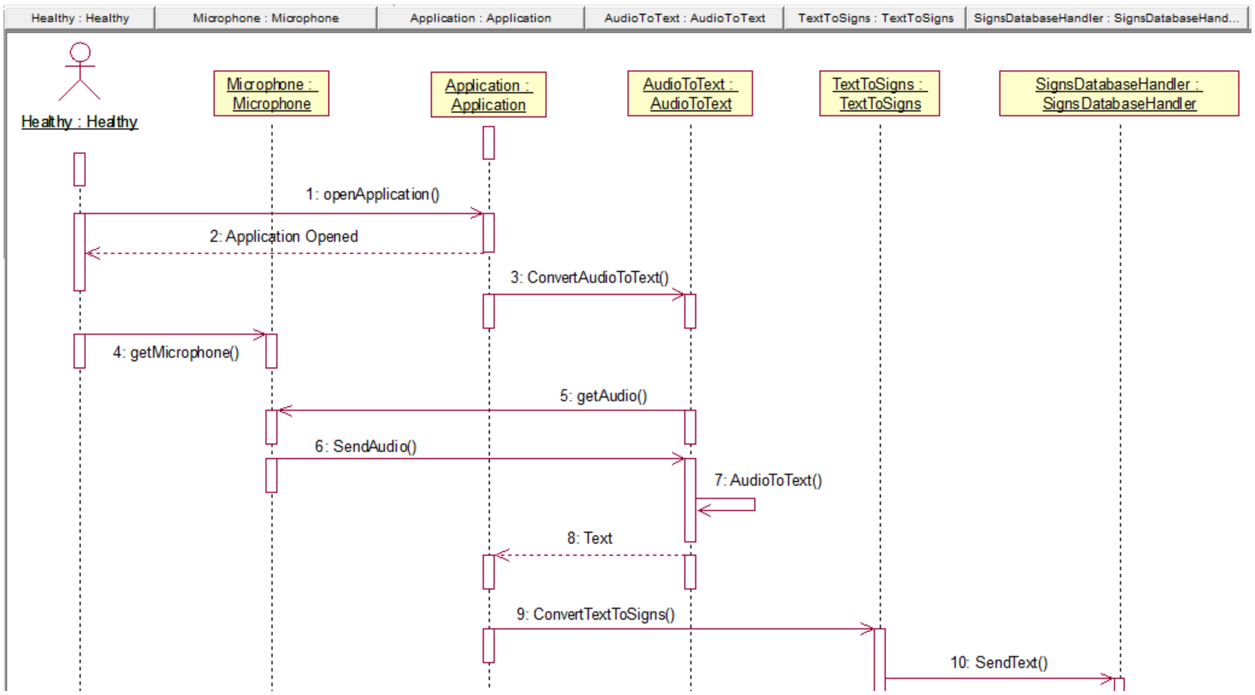
Use Case	Convert actions to audio or text
Actors	Healthy Person
Use Case Description	This use case will allow healthy person to convert sign language actions performed by DAD to audio and text
Normal Flow	The app correctly recognizes the actions performed and sign language actions performed by DAD would be converted into relative audio and text
Alternative Flow	No or faulty audio and text would be generated
Pre-condition	The glove and app are connected via Bluetooth

Post Condition	Relative audio and text is displayed
Includes	Audio and text corresponding the sign language actions performed
Extends	N/A
Assumptions	The audio and text is the same as what the deaf and dumb person intended to say

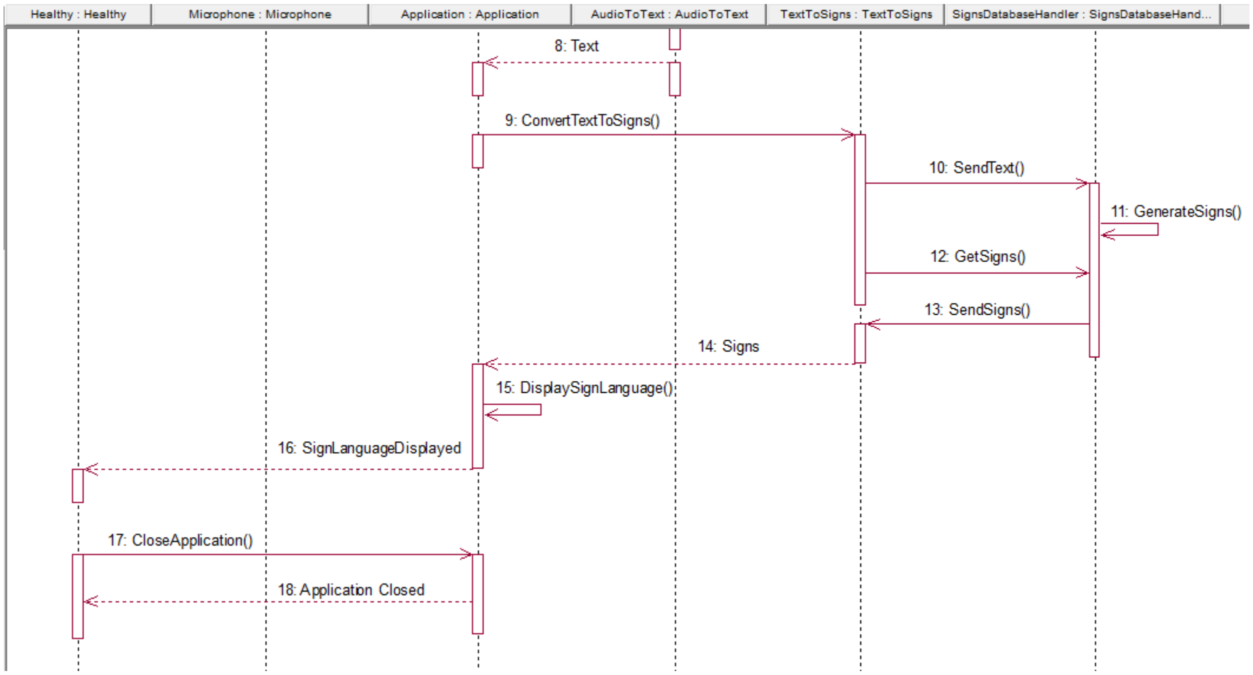
4.4.4 Sequence Diagrams:

The sequence diagram(s) of Anyone Can Talk is given below:

Sequence Diagram for Healthy User:

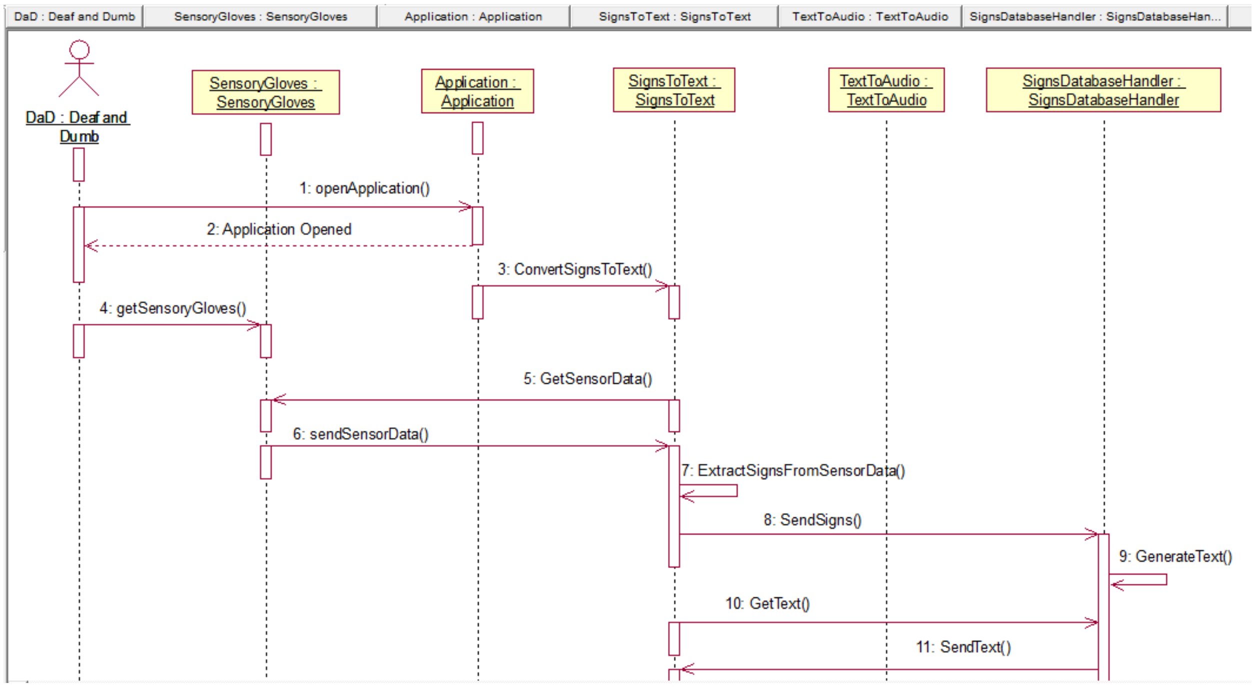


Continued Sequence Diagram for the Healthy User..



Fugure7A :Sequence diag for healthy user

Sequence Diagram for Deaf & Dumb User:


DaD : Deaf and Dumb
SensoryGloves : SensoryGloves
Application : Application
SignsToText : SignsToText
TextToAudio : TextToAudio
SignsDatabaseHandler : SignsDatabaseHan...
DaD : Deaf and Dumb
SensoryGloves : SensoryGloves
Application : Application
SignsToText : SignsToText
TextToAudio : TextToAudio
SignsDatabaseHandler : SignsDatabaseHandler
1: openApplication()
2: Application Opened
3: ConvertSignsToText()
4: getSensoryGloves()
5: GetSensorData()
6: sendSensorData()
7: ExtractSignsFromSensorData()
8: SendSigns()
9: GenerateText()
10: GetText()
11: SendText()

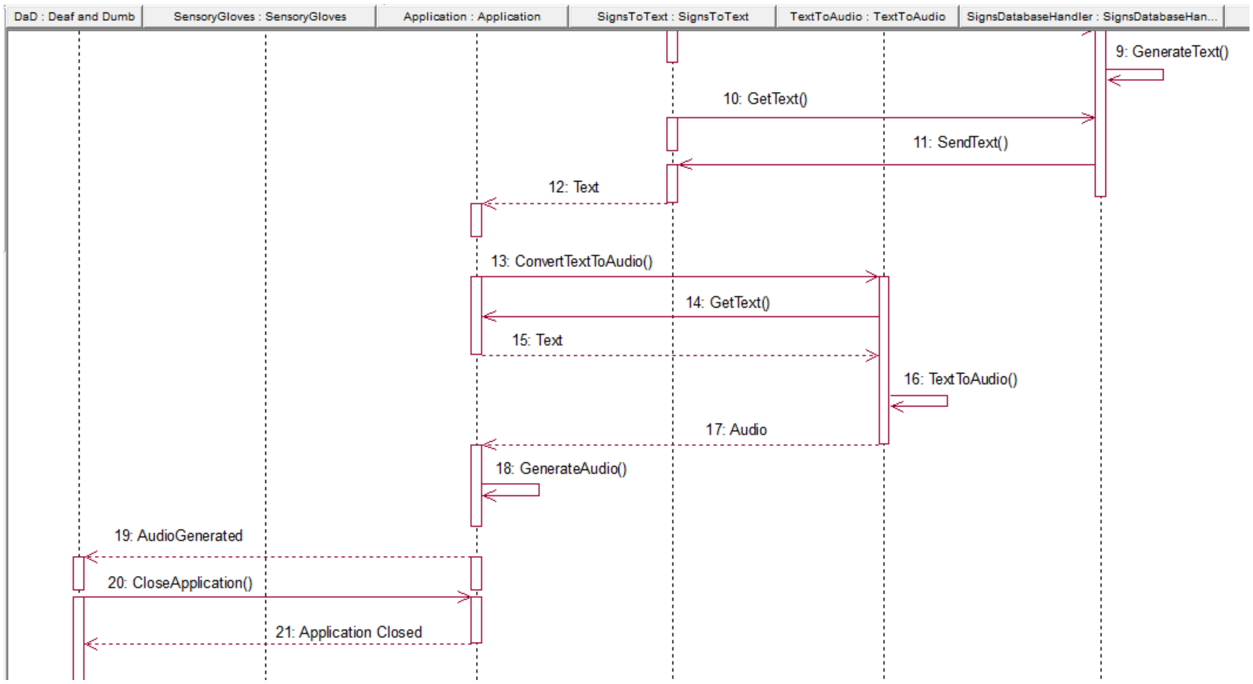


Figure 7B :Sequencediag for DAD user

4.4.5 Class Diagram:

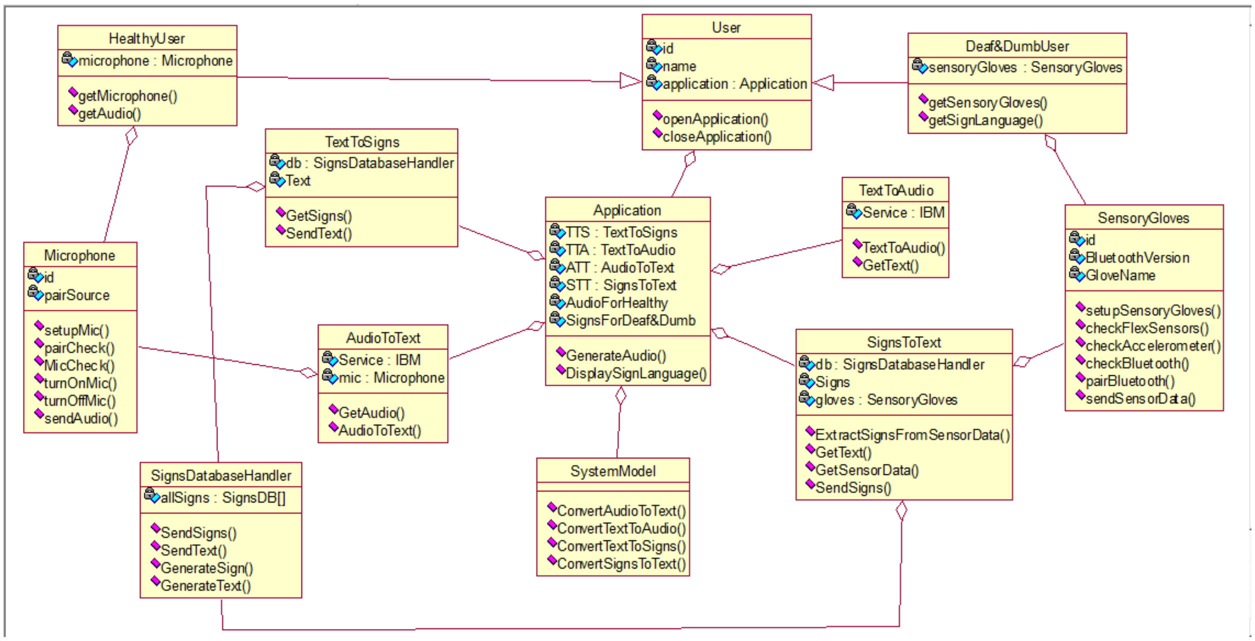


Figure8 : Class Diagram

Name	Description
User	This is the same class for all the users. It contains methods which any user can perform like opening and closing application
Deaf&DumbUser	Deaf&DumbUser extends the Main user class to add the functionality of getting sensory gloves and visual sign language on screen.
HealthyUser	HealthyUser class extends the Main user class and add the functionality of of getting microphone and converted audio of signs performed by Deaf&Dumb Person
SystemModel	This SystemModel class acts as the model for the MVC structure of our class diagram it is later on implemented by our Application. It contains all the abstract methods' definitions.
Application	This is the main Processing class of our project it handles all the operations such as conversion from audio to text, text to audio, signs to text, text to signs. It is also responsible for generating audio and displaying guestures.
SignsToText	This Class takes data from the sensory gloves and convert it to the textual data using the sign language database.
TextToSigns	This class takes text, converted from the audio captured by microphone, and convert it to the sign language for deaf and dumb people.
AudioToText	This class is used to record the audio being captured by the microphone and also convert it into textual data.
TextToAudio	TextToAudio class used the converted text from the sign language and convert it further into audio too hence making it more easier for the healthy person to understand
SensoryGloves	SensoryGloves class contains the information about flex sensors, sensor values generated, it also records the signs being generated and send it to the Signs to Text conversion class.
Microphone	Microphone class contains the data about Microphone and it is used to record the audio and send it to the Audio to text conversion class
SignsDatabaseHandler	SignsDatabaseHandler contains the full data base of sign language and is used to give the respective text of any sign language word or respective sign of any

	word.

4.4.6 Logical View (State Transition Diagram) :

The State Transition diagram is shown in the figure below:

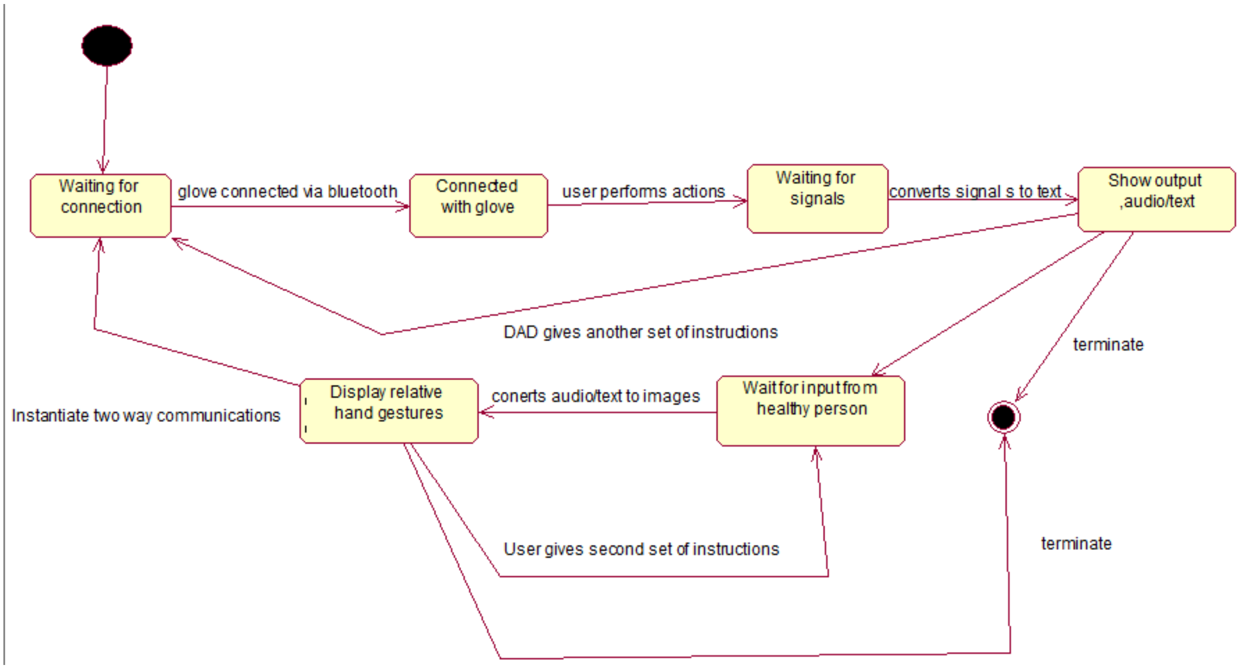

Waiting for connection
glove connected via bluetooth
Connected with glove
user performs actions
Waiting for signals
converts signal s to text
Show output ,audio/text
DAD gives another set of instructions
terminate
Instantiate two way communications
Display relative hand gestures
conerts audio/text to images
Wait for input from healthy person
User gives second set of instructions
terminate

Figure 9 : State Transition Diagram

4.4.7 Dynamic view (Activity Diagram):

In activity diagram, the dynamic view of the system is shown. All the activities are shown concurrently with their respective start and end states.

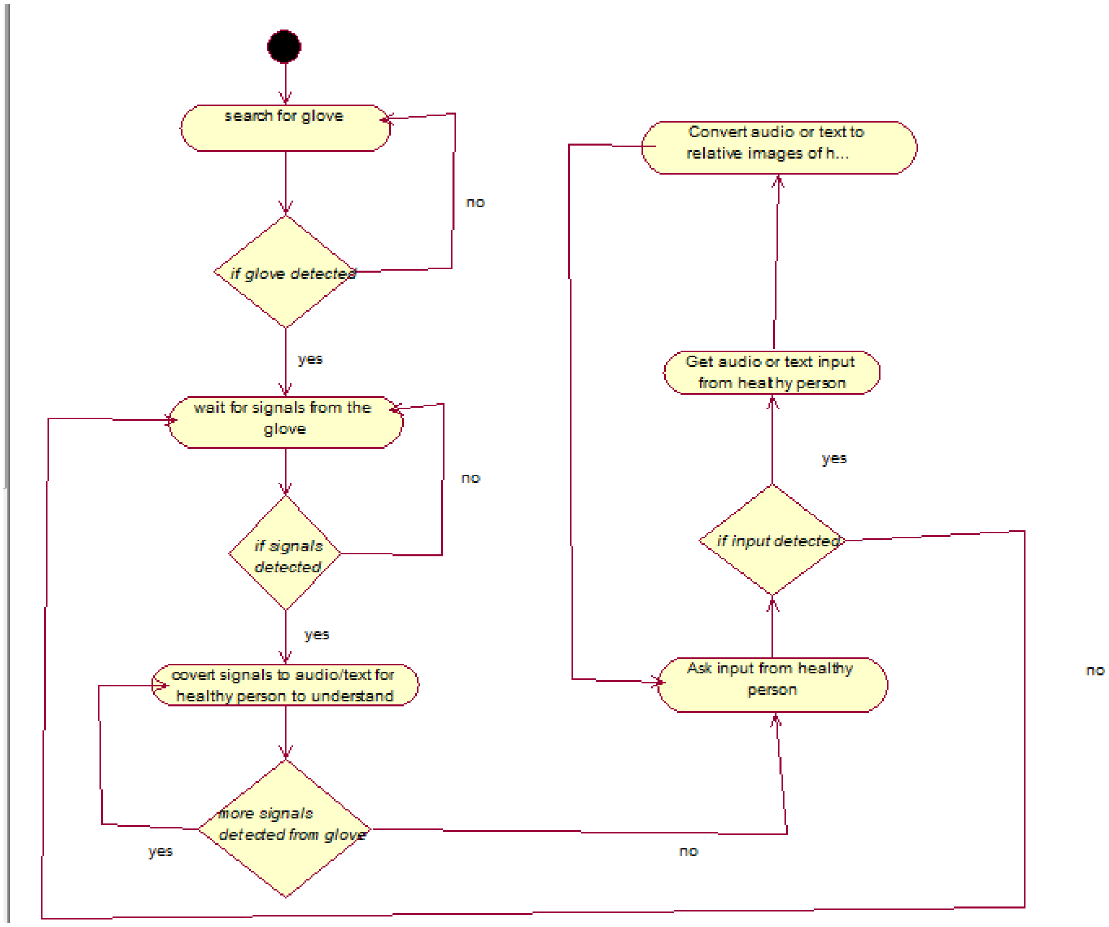

search for glove
no
if glove detected
yes
wait for signals from the glove
no
if signals detected
yes
covert signals to audio/text for healthy person to understand
more signals detected from glove
yes
no
Convert audio or text to relative images of h...
Get audio or text input from healthy person
yes
if input detected
no
Ask input from healthy person

Figure 10 :Activity Diagram of Application

4.4.8 Structure Chart:

This chart shows the breakdown of the application to its lowest manageable levels. It shows the modules and their corresponding functions which this application will implement. This chart basically shows the structure breakdown of the application starting from main modules to specific functions.

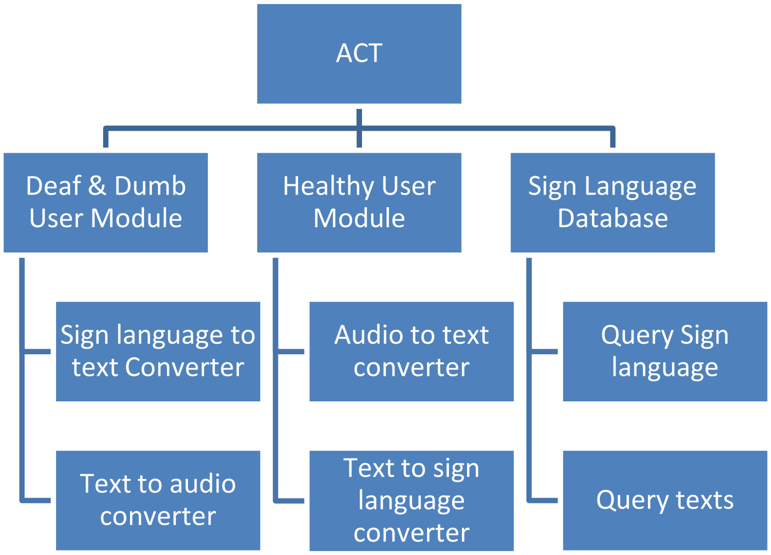
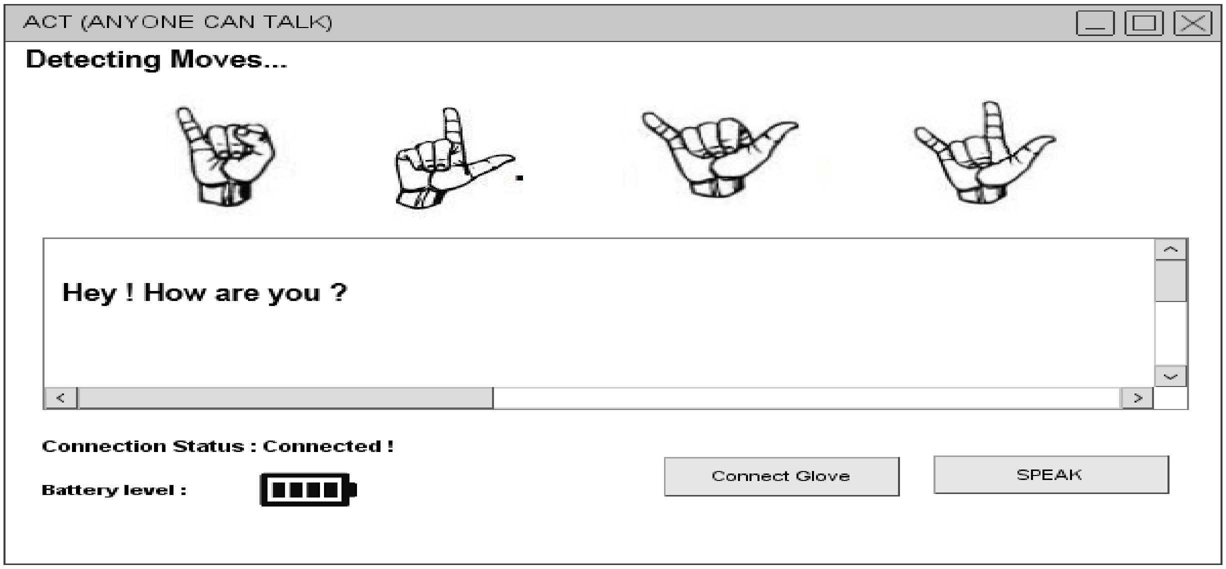


Figure 11 : Structure Chart

User Interface

The user Interface of Anyone can talk is as follows:



Detecting Moves...

Hey ! How are you ?

Connection Status : Connected !

Battery level :

Figure 12 : User Interface

4.5 Detailed Description of Components:

4.5.1 User Input

Identification	**Name**: User input **Location**: View
Type	Component
Purpose	• The purpose of this component is to receive user input. • The input voice of the healthy person is sent through microphone to application processing. • The input gestures of the deaf and dumb person are sent through sensory glove to application processing.
Function	**What the component does:** Detects input in the form of voice from healthy person and hand gestures from deaf and dumb person..
Subordinates	**Constituents of the component:** The component has no sub-components. **Functional Requirements:** **Requirement 1:** The healthy person user will be able to send his voice through microphone to application processing. **Requirement 2:** The deaf and dumb user will be able to send gestures through sensory glove to application processing.
Dependencies	**Components using this component:** The sensory glove and microphone components of the hardware component will get its input from this module. This input will be further processed by application processing component.
Interfaces	The external hardware interfaces interacting with this component will be: • Arduino Lillypad,Flex Sensors and accelerometers.
Resources	The resources used by this component are sensory glove (for hand gestures inputs) and microphone (for audio voice input).
Processing	The processing required for this component is receiving the user’s input and giving this input to the GUI application processing.

Data	Users inputs (hand gestures,voice).

4.5.2 GUI

Identification	**Name:** GUI (Graphical User Interface) **Location:** View
Type	Component
Purpose	To display the application.
Function	**What the component does:** - The complete application environment is displayed by this component.
Subordinates	**Constituents of the component:** GUI basically contains all the control of application processing. **Functional Requirements:** **Requirement 1:** The deaf and dumb user will be able to see the hand gestures on the screen.
Dependencies	GUI is dependent on input of deaf and dumb person and healthy person. The sensory glove and microphone component of the hardware components will be providing input to the GUI.
Interfaces	**External interface requirement for GUI** An output display screen which it will be using to display all the gestures created by the application processing.
Resources	It will be using Android smartphone screen to display the gestures for the deaf and dumb person.
Processing	The processing required of this component is to respond and display the gestures made for corresponding voice of the healthy person.
Data	User input (hand gestures, audio voice)

4.5.3Hardwares

Identification	**Name:**Hardwares **Location:** Model
Type	Module
Purpose	Hardwares module is the main part that contains all hardwares including sensory glove, display screen, microphone and speaker.Its purpose is to take input in the form of hand gestures and audio voice from User input component and sent further to GUI app for processing.
Function	**What the component does:** The main functions of this module are as follows: • Display hand gestures on screen • Send audio voice with the help of microphone to GUI app • Provide audio with the help of speakers to healthy person • Send hand gestures with the help of sensory glove to GUI app
Subordinates	**Constituents of the component:** The other components using this component are: • GUI • User Input **Functional requirements:** **Requirement 1:** The hand gestures of deaf and dumb person are converted into audio voice and similarly voice of healthy person is converted into hand gestures.
Dependencies	**Components using this component:** User Input Component GUI app Component
Interfaces	The external interfaces interacting with hardwares are • Android smartphone GUI app • User Input in the form of gestures and voice.

Resources	The resources used by this module are RAM Graphics memory CPU usage Speakers Microphone
Processing	The processing done by this module is that it: • Displays the gestures made by GUI app on the screen. • Processes the input from user and sends to the GUI application for converting into audio or gestures.
Data	Float values, Integer values, Strings, Arrays

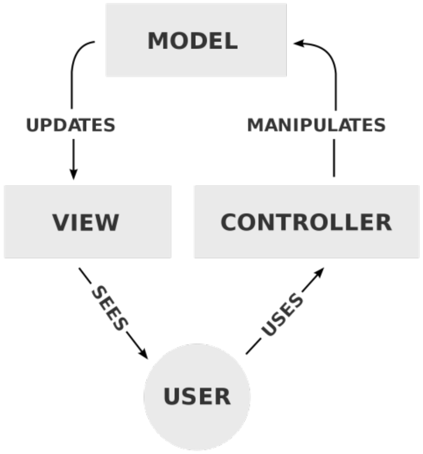
4.6 Reuse and Relationship to other products:

ACT is not based on any previous systems neither it's an extension of any other applications at any level. But it can be evolved into a bigger and more complex system with more features and functionality. Developers can also reuse some of the modules of the system. The application can also be enhanced to further include more activities such as a database can be maintained to help user keep a record of his performance throughout and see if any there is any improvement i.e. were the sessions effective? It can also be further enhanced by developing an augmented reality version of the application to make it more immersive.

4.7 Design and Tradeoffs:

ACT is an interactive application which requires multiple types of user interface. Developing such systems require thorough consideration on the design factors as it might result in complexity problem. A poorly-designed system results in a system consuming more resources with very little efficiency and a slower response time which directly affects the experience of the target user besides this, poor designs make testing and maintenance activities difficult.

MVC pattern will be used for the implementation of this application. General behavior of MVC is shown below.



4.8 Pseudo Code for Components

Healthy User :

```
Begin

OpenApplication()

GetMicrophone()

Speak

GetAudioBack()

CloseApplication()

End
```

Deaf and Dumb User :

```
Begin

OpenApplication()

GetSensoryGloves()

Perform sign language

GetSignLanguageBack()

CloseApplication()

End
```

Microphone :

```
Begin

TurnOnMic()

SetupMic()

PairCheck()

MicCheck()

SendAudio()

TurnOffMic()

End
```

Sensory Gloves :

```
Begin

SetupSensoryGloves()

CheckFlexSensors()

CheckAccelerometer()

CheckBluetooth()

PairBluetooth()

SendSensorData()

UnpairBluetooth()

End
```

Application :

```
Begin

If User is “Healthy User”:

ConvertAudioToText()

ConvertTextToSignLanguage()

DisplaySignLanguage()

If User is “Deaf & Dumb User”:

ConvertSignLanguageToText()

ConvertTextToAudio()

GenerateAudio()

End
```

AudioToText :

```
Begin
Import IBM service
GetAudio()
ConvertAudioToText()
End
```

TextToSigns :

```
Begin
Send text to sign language database
Get sign language from its database
End
```

SignsToText :

```
Begin
Send sign language to its database
Get text from sign language database
End
```

TextToAudio :

```
Begin
Import IBM service
GetText()
```

```
ConvertTextToAudio()
End
```

SignsDatabaseHandler :

```
Begin
Open Sign language database
If IncomingData = “SignLanguage”
GenerateText()
SendText() to signToTextConverter
If IncomingData = “TextualData”
GenerateSignLanguage()
SendSignLanguage() to textToSignLanguageConverter
```

Chapter 5: Testing and Evaluation

5.1 Introduction

This test plan chapter describes the appropriate strategies, process and methodologies used to plan, execute and manage testing of the ACT Android/Desktop application project. The test plan will ensure that ACT meets the customer requirements at an accredited level.

Manual Testing will be followed which includes testing a software manually, i.e., without using any automated tool or any script. In this type, the tester takes over the role of an end-user and tests the software to identify any unexpected behavior or bug. Each Unit will be tested separately and then will be integrated with other units, therefore Unit Testing and Integration testing will be followed. For each unit Black box Testing is done and for combined units Acceptance Testing is done.

The test scope includes the Testing of all functional, application performance and use cases requirements listed in the *requirement document*

Software testing, depending on the testing method employed, can be implemented at any time in the development process. However, most of the test effort occurs after the requirements have been defined and the coding process has been completed.

This document includes the plan, scope, approach and procedure of ACT test. The pass/fail criteria of the test items are also defined. The Test Plan document documents and tracks the necessary information required to effectively define the approach to be used in the testing of the product.

5.2 Test Items

Based on the ACT requirements and design description, application modules of mobile Android/Desktop Application/Desktop App and non-functional scenario will be tested. The Requirements Defined in Software Requirements Specification and the Design entities as explained in Software Design Document will be tested.

5.3 Features tested

Following Features are Tested :

1. Ability to give speech input
2. Ability to send speech input to google/ibmapi.
3. Ability to view speech input as text.
4. Ability to convert text into sign language.
5. Ability to view sign language in a grid/ slideshow layout.
6. Ability to say words , numbers and special characters in speech input.
7. Ability to perform actions on glove.
8. Ability to see text output of actions on screen.
9. Ability to convert text from gloves to sound using google/ibmapi.

5.4 Approach

Acceptance test will be executed based on this acceptance test plan. And after all test cases are executed, a test report will be summarized to show the quality of ACT. Following test approaches will be used in test execution:

1. **Unit test**. Developers are responsible for unit test as white-box testing. The implementation of each module and individual component will be verified separately.
2. **Integration test**. After the unit test is passed above the defined quality threshold, testers will execute the integration test cases. After all the modules are integrated, it’s crucial to test the product as a black-box. End-to-end scenarios will be tested to ensure the communication functionality.
3. **Regression test**.After developers fix the bug in one feature, regression test will be executed by testers to ensure that the other functions are not affected.
4. **Field test**. Firstly, untrained end users recreate one or more existing (but narrow) mass observation events in the ACT Android/Desktop Application. A number of observers will be invited to help with evaluation. After that, post event questionnaires will be used to collect quantitative usage data as well as qualitative data and further improvement will be taken into consideration.

5. **Positive and negative testing design technique**. This approach will be combined with unit test and integration test. Test cases are designed in obvious scenarios, which ensure that all functional requirements are satisfied. What’s more, different test cases will also be covered to show how the system reacts with invalid operations.

5.5 Item Pass/Fail Criteria

Details of the test cases are specified in section Test Deliverables. Following the principles outlined below, a test item would be judged as pass or fail.

1. Preconditions are met
2. Inputs are carried out as specified
3. The result works as what specified in output => Pass

- The system doesn't work or not the same as output specification => Fail

5.6 Suspension Criteria and Resumption Requirements

Any bugs found can be fixed by developers quickly and no need to start the testing process from the beginning. However, when major bugs will block the some test cases as they are interdependent and the testing has to be paused. The test will restart from the very beginning until the major error is solved.

5.7 Test Deliverables

Following are the Test Cases:

Test Case Name	Give Speech Input

Test Case No	1
Description	Testing Feature Giving speech input
Testing Technique Used	Unit Testing
Preconditions	Application should be installed in Android/Desktop Operating System . Internet Connection should be available .
Input Values	Speech input using mike
Valid Inputs	English language speech containing , words , numbers or special characters
Steps	Select the ACT Android/Desktop application installed in Android/Desktop Operating System. Open speech to sign module , click on the mike button and speak any sentence.
Expected Output	Images of signs of corresponding words in the speech .
Actual Output	Images of signs of corresponding words in the speech . In gird layout or presentation view.

Test Case Name	Select Speech to Sign button on main fragment
Test Case No	2
Description	Testing the App feature
Testing Technique Used	Unit Testing
Preconditions	The Application must be opened displaying Main Choice Menu
Input Values	Touch
Valid Inputs	Touch

Steps	First select the ACT Android/Desktop application installed in Android/Desktop Operating System,
Expected Output	Speech to sign fragment (Android) opens . Speech to sign window (Desktop) opens
Actual Output	Speech to sign fragment (Android) opens . Speech to sign window (Desktop) opens

Test Case Name	Checking number input in speech
Test Case No	3
Description	Testing if number inputs are processed appropriately by Android/Desktop App
Testing Technique Used	Unit Testing
Preconditions	User must be in the speech to sign menu and connected to the internet
Input Values	Numbers
Valid Inputs	Whole integer values from 0 onwards
Steps	Select the ACT Android/Desktop application installed in Android/Desktop Operating System. Open speech to sign module , click on the mike button and speak any number.
Expected Output	Appropriate images of numbers displayed
Actual Output	Appropriate images of numbers displayed

Test Case Name	Checking speech input with special characters
Test Case No	4
Description	Testing how the app processes special characters in speech
Testing Technique Used	Unit Testing
Preconditions	User must be in the speech to sign menu and connected to the internet
Input Values	Special Characters
Valid Inputs	Special characters such as commas , hyphens , fullstops, exclamation marks
Steps	Select the ACT Android/Desktop application installed in Android/Desktop Operating System. Open speech to sign module , click on the mike button and speak any sentence with special character.
Expected Output	Special character image should not be displayed
Actual Output	Blank or no image in place of special character

Test Case Name	Checking speech input with spaces
Test Case No	5
Description	Testing how the application processes spaces
Testing Technique Used	Unit Testing
Preconditions	User must be in the speech to sign menu and connected to the internet
Input Values	Spaces

Valid Inputs	Spaces
Steps	Select the ACT Android/Desktop application installed in Android/Desktop Operating System. Open speech to sign module , click on the mike button and speak any sentence with spaces.
Expected Output	A blank image in place of space
Actual Output	Blank Image in place of space

Test Case Name	Feature Choice Menu
Test Case No	6
Description	Testing the choice menu which appears on the screen when the cursor is moved on dots icon at top right corner of the screen.
Testing Technique Used	Unit Testing
Preconditions	The user must have drag the cursor on dots icon at top right corner of the screen.
Input Values	Move cursor at icon on main home screen.
Valid Inputs	Only Cursor movements at dots icon of main home screen.
Steps	First select the ACT Android/Desktop application installed in Android/Desktop Operating System then move the cursor on three dots icon which slides and displays horizontal and slidable menu containing speech to signs and signs to speech icons.
Expected Output	The dots icon should display the home ,speech to signs and signs to speech icons on moving cursor on it at top right corner of the

	screen.
Actual Output	The home, speech to signs and signs to speech icons are displayed on moving cursor on dot icon at top right corner of the screen.

Test Case Name	Speech to Signs Feature
Test Case No	7
Description	Testing the speech to signs menu that opens an interface where we can convert the speech into signs by speaking in the mic.
Testing Technique Used	Unit Testing
Preconditions	The user must have clicked on speech to signs menu.
Input Values	Mouse clicking
Valid Inputs	The speech to signs menu is clicked by mouse button.
Steps	First select the ACT Android/Desktop application installed in Android/Desktop Operating System. Drag the cursor on the dots icon at top right corner of the screen and then click on the speech to signs menu.
Expected Output	The speech to signs interface should be displayed where we can speak in the mic and it is converted into signs.
Actual Output	The speech to signs interface is displayed where we can speak in the mic and it is converted into signs.

Test Case Name	Start listening the speech using mic
Test Case No	8
Description	Testing Start listening the speech button when it is clicked then we can speak in the mic to get its corresponding signs which are displayed at bottom of the screen.
Testing Technique Used	Unit Testing
Preconditions	The user must have clicked on the Start Listening button on the speech to signs interface.
Input Values	Clicking Start Listening button
Valid Inputs	Select the Start Listening button and valid English words are spoken in the mic.
Steps	First select the ACT Android/Desktop application installed in Android/Desktop Operating System. Drag the cursor on the dots icon at top right corner of the screen and then click on the speech to signs menu.Now click on the Start Listening button on the screen that would enable the application to get our voice as input and convert into signs.
Expected Output	The Listening message should be displayed on the screen and it should start listening our voice and should start processing on it to convert into signs.
Actual Output	The Listening message is displayed on the screen and it started listening our voice and started processing on it to convert into signs.

Test Case Name	Generate Signs for Speech/Text
Test Case No	9
Description	Testing Generate Signs feature for the input speech or text
Testing Technique Used	Unit Testing
Preconditions	The user must have clicked on the Generate Signs button on the screen.
Input Values	Clicking Generate Signs Button
Valid Inputs	Select the Generate Signs button by clicking on it.
Steps	First select the ACT Android/Desktop application installed in Android/Desktop Operating System. Drag the cursor on the dots icon at top right corner of the screen and then click on the speech to signs menu. Now click on the Start Listening button on the screen then speak in the mic and click on the generate Sign button that would generate the corresponding signs for the speech.
Expected Output	The Signs should be displayed on the bottom of the screen.
Actual Output	The Signs are displayed on the screen for corresponding speech.

Test Case Name	Auto Generate signs
Test Case No	10

Description	Testing Auto Generate Signs feature for the input speech or text
Testing Technique Used	Unit Testing
Preconditions	The user must have clicked on the Auto Generate button.
Input Values	Clicking the Auto Generate button
Valid Inputs	Select the Auto Generate button
Steps	First select the ACT Android/Desktop application installed in Android/Desktop Operating System. Drag the cursor on the dots icon at top right corner of the screen and then click on the speech to signs menu. Now click on the Start Listening button on the screen then speak in the mic and click on the Auto Generate Signs button that would generate the corresponding signs for the speech
Expected Output	The Signs should be displayed on the bottom of the screen.
Actual Output	The Signs are displayed on the screen for corresponding speech.

Test Case Name	Clear Button on Speech to Signs interface/Screen
Test Case No	11
Description	Testing the function of Clear Button on Speech to Signs interface
Testing Technique Used	Unit Testing

Preconditions	The Speech to Signs interface must be displayed
Input Values	Clicking Clear Button
Valid Inputs	Select the Clear Button
Steps	First select the ACT Android/Desktop application installed in Android/Desktop Operating System. Drag the cursor on the dots icon at top right corner of the screen and then click on the speech to signs menu. Now click on the Start Listening button on the screen then speak in the mic. Now you can clear the text written in the Text Area by simply clicking Clear Button.
Expected Output	The Text Area should become empty and should clear all text written in it.
Actual Output	The Text Area became empty and all text is cleared which was written in it.

Test Case Name	Writing text in the Text Area and converting into Signs
Test Case No	12
Description	Testing the Writing text in the Text Area feature and converting into Signs
Testing Technique Used	Unit Testing
Preconditions	The Speech to Signs interface must be opened and cursor is pointed in the Text Area
Input Values	Clicking in the Text Area with mouse
Valid Inputs	Select the Text Area by clicking in it.

Steps	First select the ACT Android/Desktop application installed in Android/Desktop Operating System. Drag the cursor on the dots icon at top right corner of the screen and then click on the speech to signs menu. Now click on Text Area and write valid words or sentences. Then click on the Generate Signs Button to get the corresponding Signs of the Text.
Expected Output	The Text Area should be highlighted and enabled user to write some text in it and ultimately convert into signs.
Actual Output	The Text Area is highlighted and enabled user to write some text in it and then convert into signs by clicking on Generate Signs Button

Test Case Name	Clicking on Displayed Signs on Speech to Signs Interface
Test Case No	13
Description	Testing the Clicking on Displayed Signs on Speech to Signs Interface. Signs of next word are shown by clicking on the sign of current word.
Testing Technique Used	Unit Testing
Preconditions	The user must have generated the signs for any speech.
Input Values	Mouse click on signs of current words
Valid Inputs	Click on the signs of the current word being displayed on the screen.
Steps	First select the ACT Android/Desktop application installed in Android/Desktop Operating System. Drag the cursor on the dots icon at top right corner of the screen and then click on the speech to signs menu. Now click on the Start Listening button on the screen then speak in the mic and click on the generate Signs button that would generate the corresponding signs for the

	speech.Now click on the signs that are displayed at bottom of the screen.By clicking it the signs of next word are displayed.
Expected Output	The Signs of next word should be displayed on the screen by clicking on the signs of the current word
Actual Output	By clicking on the signs of the current word, the Signs of next word are displayed on the screen.

Test Case Name	Clicking Cross Icon on The Screen
Test Case No	14
Description	Testing the quitting button of the App
Testing Technique Used	Unit Testing
Preconditions	The user must have opened the ACT application.
Input Values	Mouse clicking on cross icon
Valid Inputs	Choose cross icon by clicking on it.
Steps	First select the ACT Android/Desktop application installed in Android/Desktop Operating System then click on cross icon on the top left corner of the screen.
Expected Output	The ACT Android/Desktop APP should be closed after confirmation by showing the message.

Actual Output	The ACT Android/Desktop APP is closed by clicking on cross icon after confirmation message.

5.8 Environmental Needs

Hardware :

1. Mobile with Android platform
2. Sensory glove

Software :

1. Mobile Platform: Android 3.0/3.1 or later.
2. Eclipse Oxygen 4.7.3a with IBM jar 3.0.1 file included.
3. Android Studio InteliJ 3.1

5.9 Responsibilities, Staffing and Training Needs

Responsibilities :

- Usman Baig is responsible for Acceptance Testing
- Asad Ali is responsible for Integration Testing
- MufaddalSadiq is responsible for testing each separate unit that is Unit Testing.

Skills :

- Skills needed to test ACT using QACenter

Chapter 6: Future Work

Different languages will be included including advanced vocabulary. We can also add more than 50 words for each language.By adding more words and sentences its work can be increased.Moreover we can add multiple languages also for more enhancements.

Bibliography

- https://firebase.google.com/docs/reference/rest/database/
- https://firebase.google.com/docs/cloud-messaging/http-server-ref
- http://ieeexplore.ieee.org/document/741940/?tp=&isnumber=16016&arnumber=741940&punumber=5982
- https://en.wikipedia.org/wiki/Sequence_diagram
- https://en.wikipedia.org/wiki/Block_diagram
- https://en.wikipedia.org/wiki/Internet_of_things
- https://en.wikipedia.org/wiki/Facade_pattern
- https://docs.python.org/2/howto/sockets.html

Appendix A : Glossary

Firebase: The cloud platform being used

API:	Application Programming Interface
OS:	Operating System

Appendix B: Issues/Limitations :

All possible issues have already been mentioned where required in the SRS. Any remaining ones are listed below:

1. The group shall try to match the features and NFRs as best as possible, however, like all software projects, any discrepancies are apologized for at this stage.
2. Feedback on requirements is expected from the users to help the group in improving the design and implementation of the project.