A PROJECT REPORT

on

"VOICE BASED EMAIL APP"

Submitted to

KIIT Deemed to be University

In Partial Fulfillment of the Requirement for the Award of

BACHELOR'S DEGREE IN
COMPUTER SCIENCE AND ENGINEERING

BY

AKSHAY GOURISARIA	1705110
ANIK BISWAS	1705112
ARUNAVA ROY	1705126
ASHUTOSH MALLICK	1705127
RISHIVANSH PATNAIK	1705160

UNDER THE GUIDANCE OF
PROF.KUNAL ANAND

SCHOOL OF COMPUTER ENGINEERING
KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY
BHUBANESWAR, ODISHA - 751024

May 2020

A PROJECT REPORT

on

"VOICE BASED EMAIL APP"

Submitted to

KIIT Deemed to be University

In Partial Fulfillment of the Requirement for the Award of

BACHELOR'S DEGREE IN
COMPUTER SCIENCE AND ENGINEERING

BY

AKSHAY GOURISARIA	1705110
ANIK BISWAS	1705112
ARUNAVA ROY	1705126
ASHUTOSH MALLICK	1705127
RISHIVANSH PATNAIK	1705160

UNDER THE GUIDANCE OF
PROF.KUNAL ANAND

SCHOOL OF COMPUTER ENGINEERING

KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY

BHUBANESWAR, ODISHA - 751024

May 2020

KIIT Deemed to be University

School of Computer Engineering
Bhubaneswar, ODISHA 751024

CERTIFICATE

This is certify that the project entitled

"VOICE BASED EMAIL APP"

submitted by

AKSHAY GOURISARIA	1705110
ANIK BISWAS	1705112
ARUNAVA ROY	1705126
ASHUTOSH MALLICK	1705127
RISHIVANSH PATTNAIK	1705160

is a record of bonafide work carried out by them, in the partial fulfilment of the requirement for the award of Degree of Bachelor of Engineering (Computer Science & Engineering OR Information Technology) at KIIT Deemed to be university, Bhubaneswar. This work is done during year 2019-2020, under our guidance.

Date:06/06/20

(Prof. Kunal Anand)
Project Guide

Acknowledgements

We are profoundly grateful to Prof. KUNAL ANAND for his expert guidance and continuous encouragement throughout to see that this project rights its target since its commencement to its completion….

AKSHAY GOURISARIA

ANIK BISWAS

ARUNAVA ROY

ASHUTOSH MALLICK

RISHIVANSH PATTNAIK

ABSTRACT

Internet has become one of the basic amenities for day-to-day living. Every human being is widely accessing the knowledge and information through internet. However, blind people face difficulties in accessing these text materials, also in using any service provided through internet. The advancement in computer based accessible systems like Text-to-Speech(TTS), Speech-to-Text(STT) has opened up many avenues for the visually impaired across the globe in a wide way. Audio feedback based virtual environment like, the screen readers have helped Blind people to access internet applications immensely.

We have implemented a Voice operated Mail Android application that can be used by a blind person to send E-Mails easily and efficiently with the use of voice and a single screen touch. The Mail system is based on JavaMail API and uses the in-built Android Speech recognition system for converting text into speech and speech into text. The system is completely based on interactive voice response which will make it user friendly and efficient to use.

Keywords: TTS, STT, Android, JavaMail API, IVR

Contents

List of Figures

Chapter 1

Introduction

Internet plays a vital role in today's world of communication. Today the world is running on the basis of internet. No work can be done without use of internet. Electronic mail i.e. email is the most important part in day to day life. But some of the people in today's world are not able to make proper use of this facility due to visual impairment.

Even the system that are available currently like the screen reader TTS and ASR do not provide full efficiency to the blind people so as to use the internet or E-Mail. Therefore we came up with our project as voice based email system for blinds which will help a lot to visually impaired people.

The main benefit of this system is that the requirement of keyboard and visual input is completely eliminated, the user will have to operate through voice commands given after pressing anywhere in the phone screen.

1.1 **VOICE RECOGNITION**

1.1.1 SPEECH TO TEXT

The main feature of this application is to provide ease-of-use to the user by eliminating the need of physical and visual constraints by converting speech into relevant text. When user gives input in form of voice , that voice is received by the sensor or microphone and translated into text by implementing the speech recognition functionality of Android OS . In this process , all the characters, numbers and symbols are translated.

1.1.2 TEXT TO SPEECH

Another feature of this application is to re-translate the text into equivalent voice format. This process is known as computer speech recognition or automatic speech recognition. The text is processed and converted into speech with the help of the TTS functionality of Android OS and is pronounced through the speaker . This talk back functionality is essential in making the application truly voice operated.

1.2 MAIL SENDING

The E-Mail communication feature enables the user to connect to the World and make use of the Internet facility. The application obtains the credentials of the user and stores them for future use. The mail is sent by implementing the JavaMail API which allows the Android user to make use of the Simple Mail Transfer Protocol (SMTP) to send mail to another user. The application provides audible confirmation if the mail has been successfully sent.

1.3 VOICE COMMANDS AND MULTI-LANGUAGE SUPPORT

The TTS functionality has been used to provide an ease of navigation through the application by the use of voice commands. Navigating through the necessary fields of E-mail is done by voice commands.

The application is made to support multiple languages apart from English so as to enhance the ease of operation and usability.

Chapter 2

Literature Survey

2.1 EXISTING SYSTEM

There are a total number of 4.1 billion email accounts created until 2014 and an estimated 5.1 billion accounts in 2018, and around 5.9 billion accounts by the end of 2019. This makes emails the most used form of communication.

There are different kinds of email systems which provide various facilities. But these are helpful for a limited range of users. A group of blind people are not able to use these features. These systems are only able to convert voice to text format whereas text to voice is not available.

A screen reader is computer program that enables a blind computer user to know what's on the screen through speech. One of the extent is available screen readers, reads the contents in sequential order. Hence user is able to comprehend if the contents on page is in basic HTML format. Most existing systems are uses advanced web pages which provides user friendliness to the normal user visually. Besides while reading them by using screen readers it is complicated to understand for visually challenged people.

Many of the mail services in today's world are of no use to visually impaired people because they do not provide audio feedback.

2.2 PROPOSED SYSTEM

The most important aspect that has been kept in mind while developing the proposed system is accessibility. An android application is perfectly accessible to our intended audience and has an unprecedented advantage over computers in terms of usability.

We have proposed a system which provides access to use of emails through both facilities of voice to text and text to voice conversation along multiple languages.

When using this system the phone will be prompting the user to speak specific keywords and the respective content so as to access and set the various fields of email before sending. The system will respond back in speech when given a specific command.

Chapter 3

Software Requirements Specification

3.1 FUNCTIONAL REQUIREMENTS:

3.1.1

R.1

In Login page, user is asked for email details.

User enters the details.

Goes into main page.

3.1.2

R.2

- When user speaks a particular keyword followed by the mail address,

  User's mail address is set.

- When user speaks a particular keyword followed by the password,

  User's password is set.

- When user speaks a particular keyboard followed by the contents of the subject,

  Subject is set.

- When user speaks a particular keyword followed by the receiver's mail address,

  Receiver's mail address is set.

- When user speaks a particular keyword followed by the contents of the message,
  The message is set.

3.1.3
R.3
When user speaks the keyword for sending the mail,

3.1.4
R.4
When user specifies the language,
The language of input and talkback is changed accordingly.

3.1.5
R.5
When user commands for talkback of a particular field,
The respective field is spoken out.

3.3 NON-FUNCTIONAL REQUIREMENTS

- Permission for less secure apps:

The permission to allow less secure apps to access the mail server through mail id and password should be enabled from the user's GMail Account.

- Correctness:

The speech recognition system should be accurate and give desirable outputs.

- Multilingual Support:

The system should support multiple languages, as portability is a prime goal of this project. English and Hindi should be at least be supported.

- Performance Requirement:

The system shall take no less than 2 seconds to process a typing input and no less than 1 second to process a talkback request.

- Usability Requirement:

Users must be familiar with the usage of android applications. Visually impaired users should be familiar with the usage of android phone through voice command

Chapter 4

Requirement Analysis

4.1 EXTERNAL INTERFACE REQUIREMENTS:

The mail component has to interface with the JavaMail API v1.6.2 which is open-sourced by Oracle.

The API will be used to access the Google mail server for verification of user details and sending of mail.

4.2 HARDWARE REQUIREMENTS

Processor :- 1GHz 32 bit or faster

RAM :- 1GB or more

Hard disk Space :- 512 MB or more

4.3 SOFTWARE REQUIREMENTS

Target OS- Android v4.2 or higher

Operating Environment- Windows 7/8/10

Android Studio.

4.4 CONSTRAINTS

1. App must execute on Android OS version 4.2 or above
2. Source code must be in Java
3. Internet- Internet connection required

Chapter 5

System Design

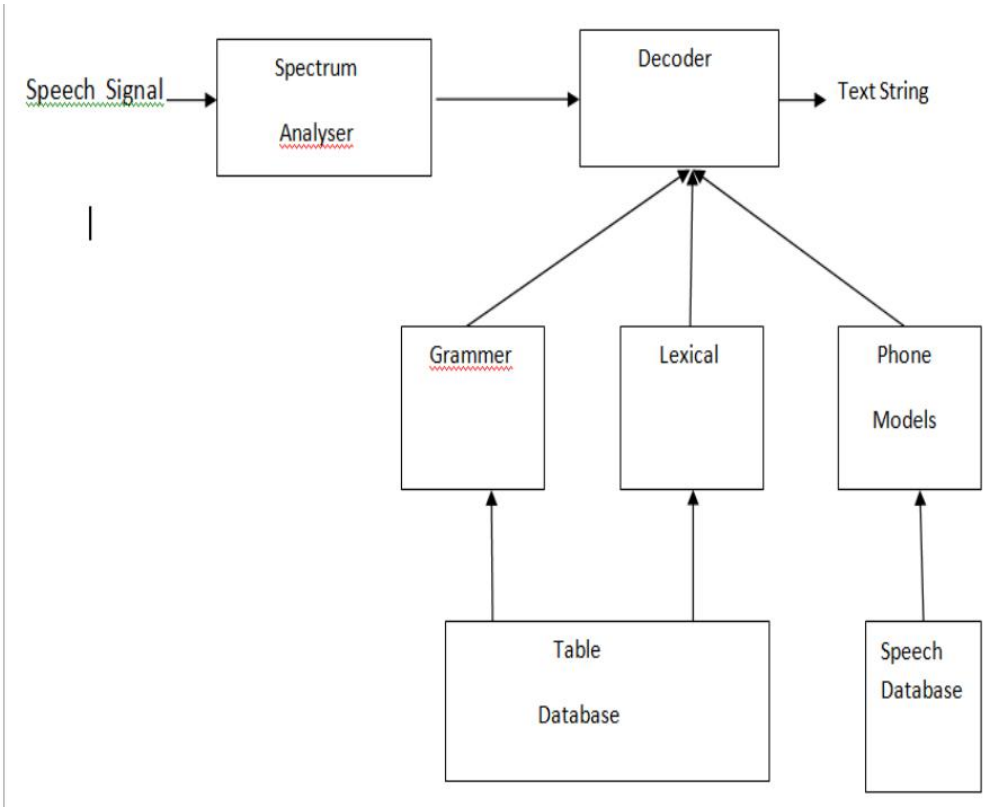


Fig 1. Speech Recognition Framework

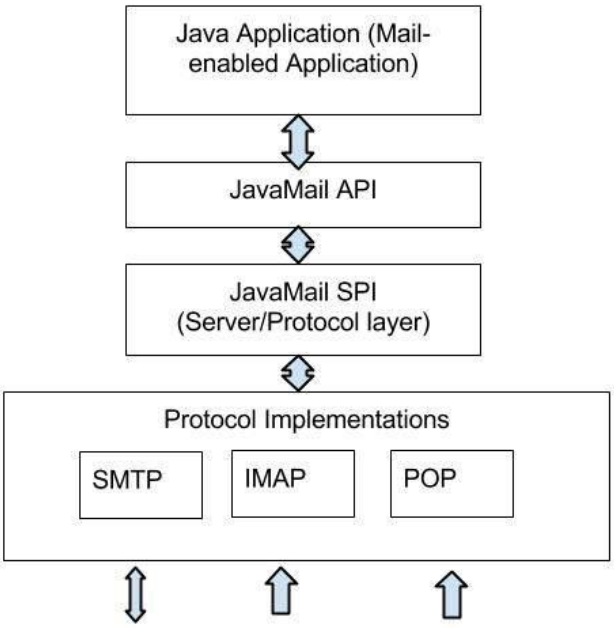


Fig 2. Workflow of JavaMail API

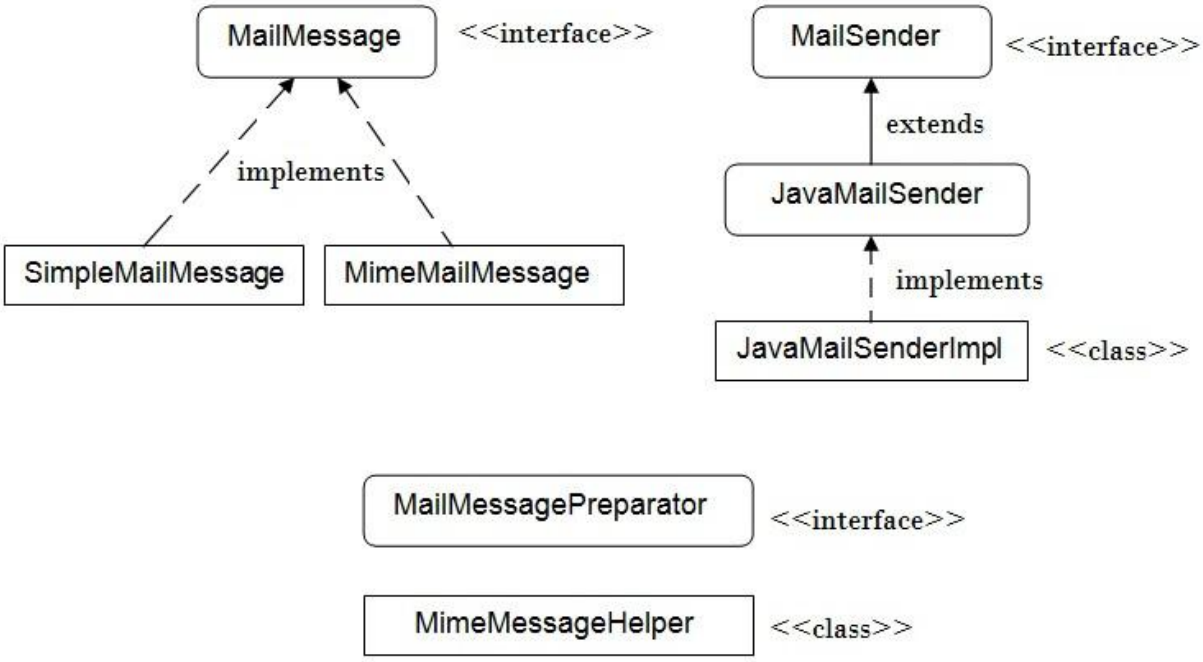


Fig 3. Interfaces and Classes of JavaMail API

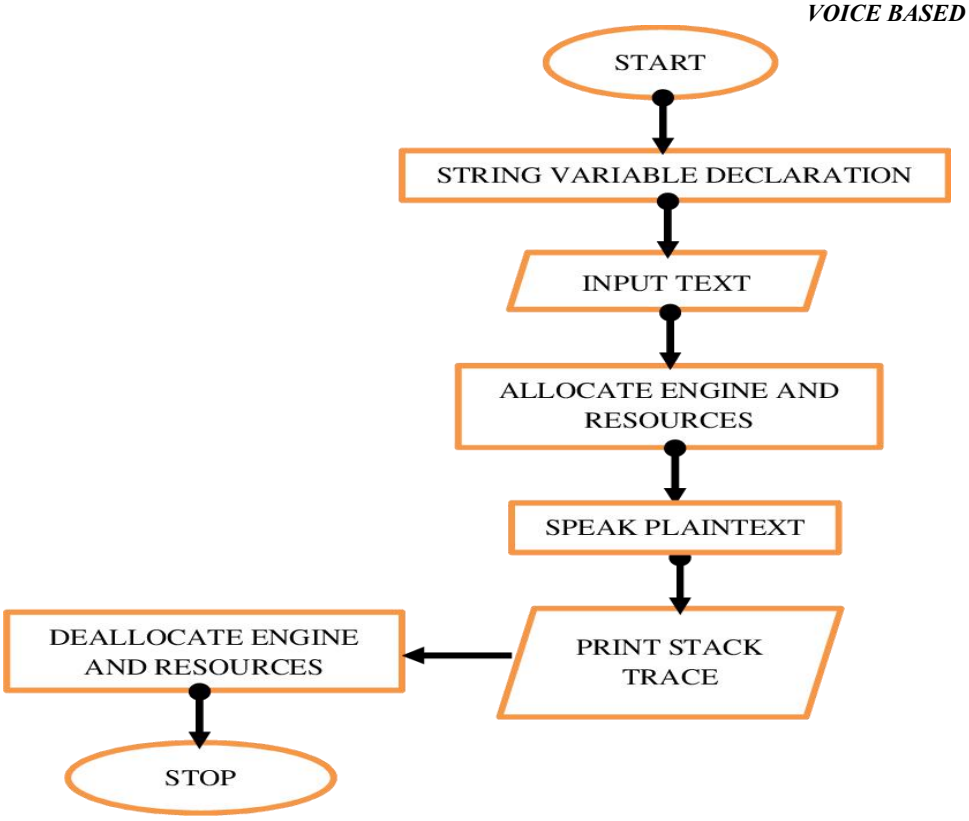


Fig 4. Workflow of Text to Speech framework

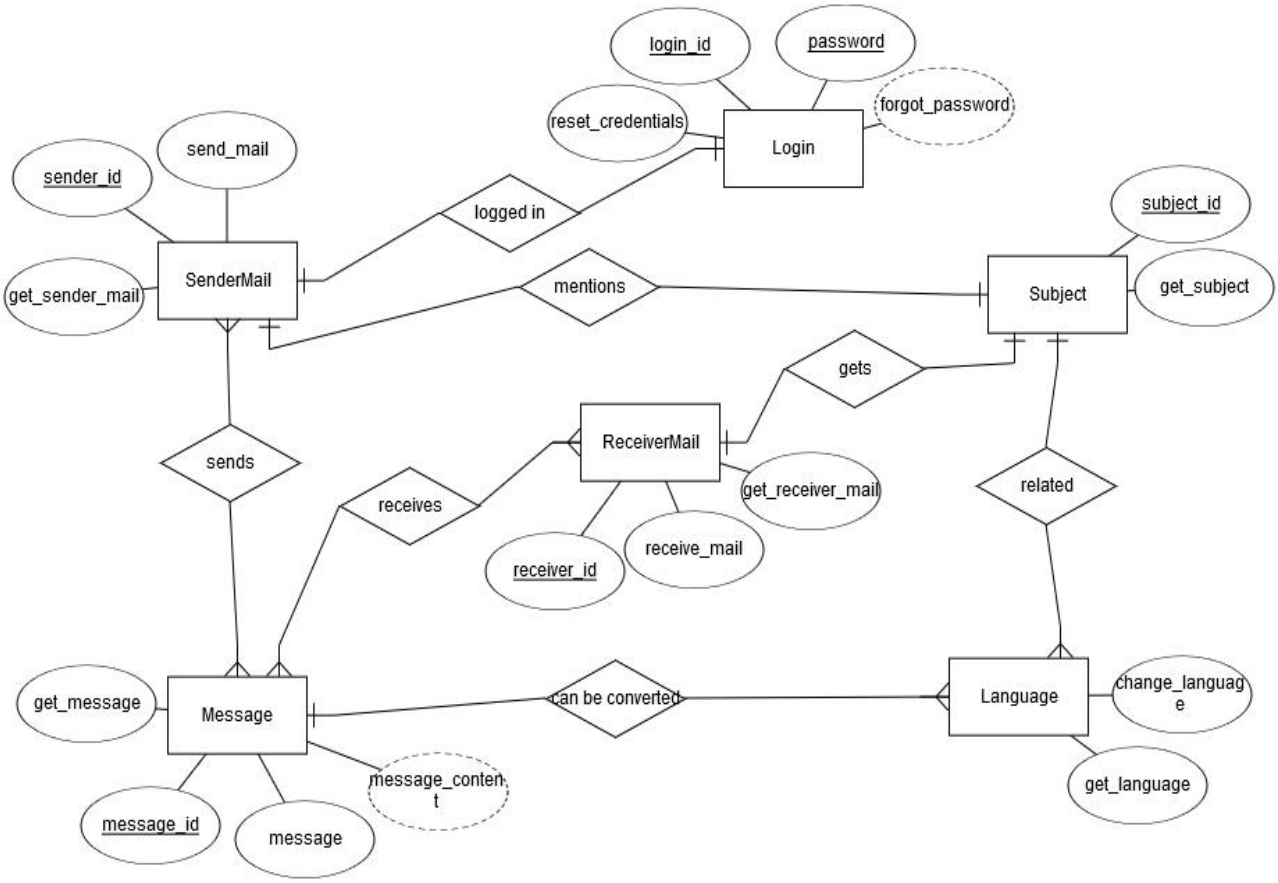
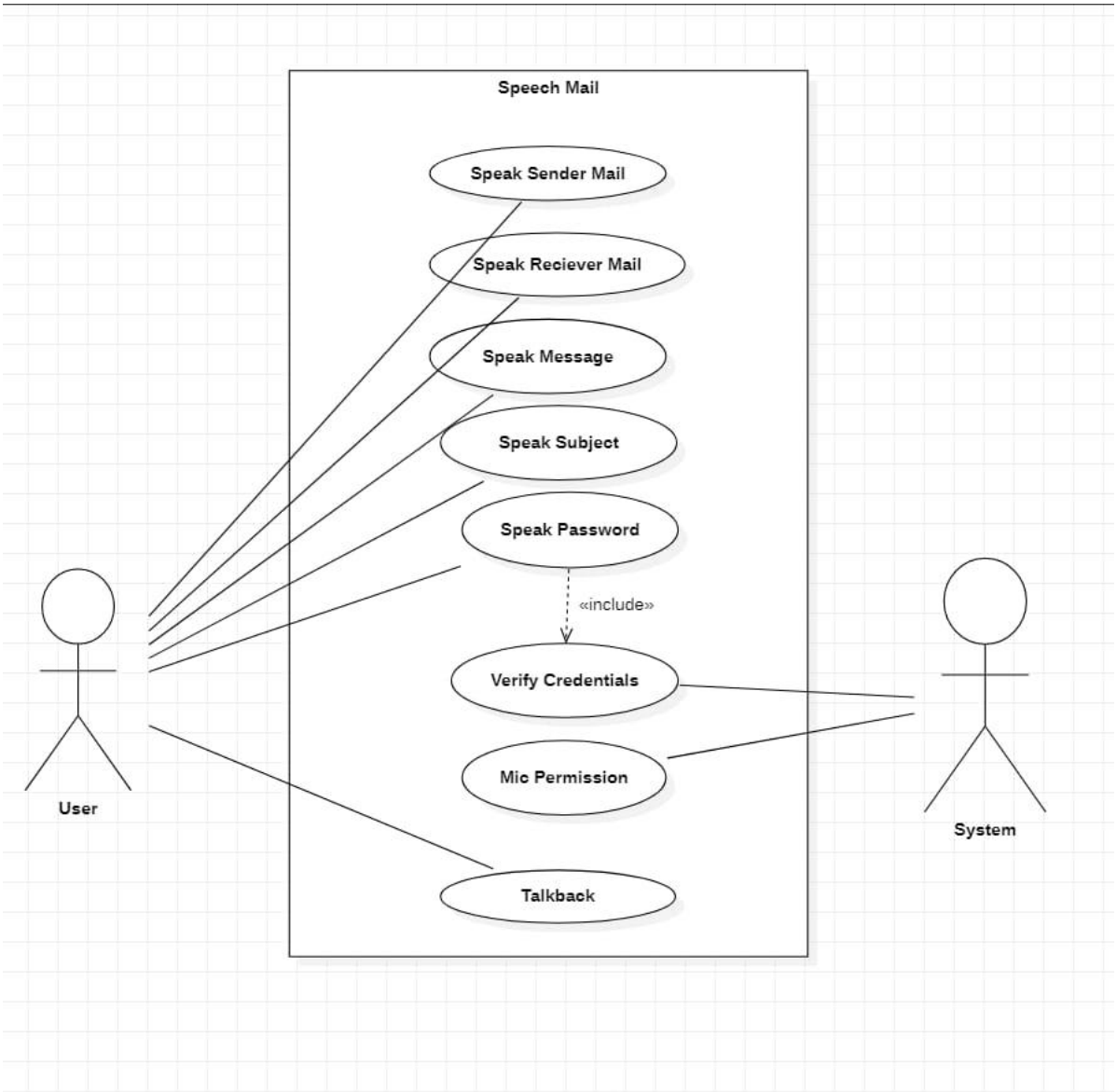


Fig 8. Entity Relationship Diagram



Fug 5. Use Case Diagram

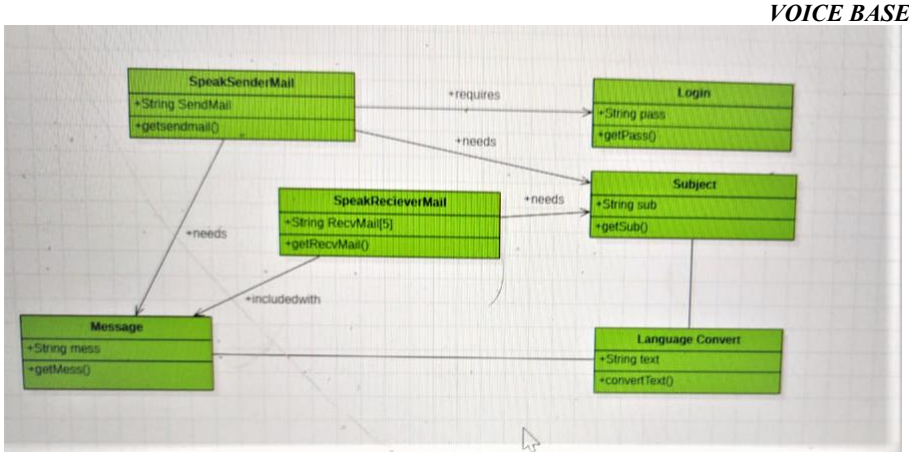


Fig 6. Class Diagram

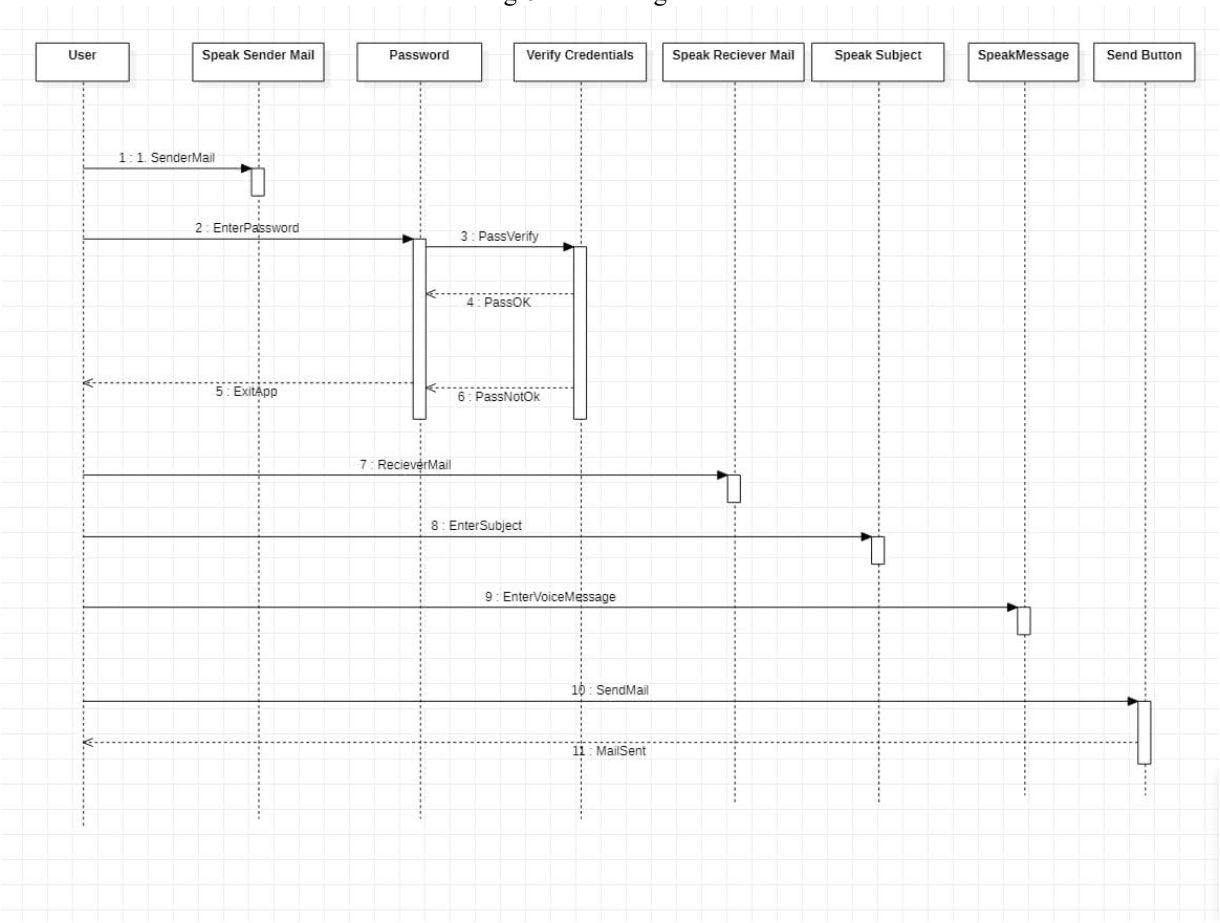


Fig 7. Sequence Diagram

Chapter 6

System Testing

6.1 Test Cases and Results

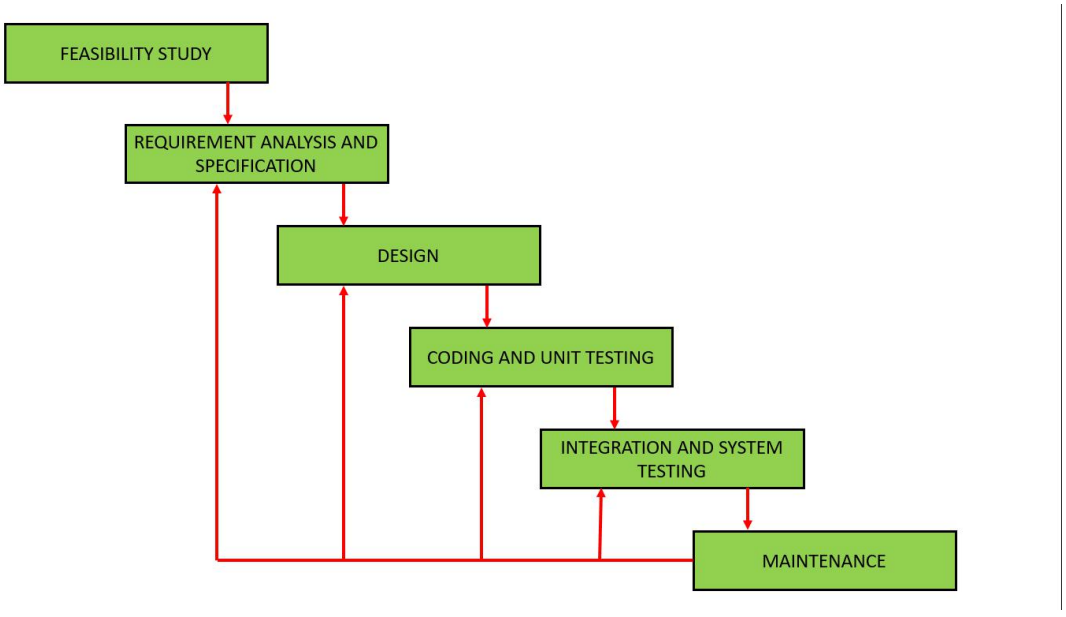
Test ID	Test Case Title	Test Condition	System Behaviour	Expected Result
TC_01	Valid EMail Credentials	The user enters a valid email address and password	System accepts the credentials for sending	System accepts the credentials for sending
TC_02	Speech to text	User speaks the keyword for a field and content	System displays the content for the field in small letters	System displays the content with proper formatting
TC_03	Text to speech	User commands for conversion of a text content to speech	System produces the respective speech through the speaker	System produces the respective speech through the speaker
TC_04	Punctuation Marks	The user speaks for a punctuation anywhere in the content	The respective punctuation is applied to the position	The respective punctuation is applied to the position
TC_05	Common Special Symbol	User commands for a special symbol to be inserted in any content	The respective special symbol is applied to the position	The respective special symbol is applied to the position
TC_06	Editing commands	User voices a command for content manipulation	System is unable to process certain commands and does not return any output	The system is able to process the commands and edit the content accordingly
TC_07	Command for sending mail	The user speaks the keyword for sending the prepared mail to the receiver's mail address	The mail is sent and system displays a message saying the mail is sent	The mail is sent and system displays a message saying the mail is sent

Chapter 7

Project Planning

In a practical software development project, the classical waterfall model is hard to use. So, Iterative waterfall model can be thought of as incorporating the necessary changes to the classical waterfall model to make it usable in practical software development projects. It is almost same as the classical waterfall model except some changes are made to increase the efficiency of the software development.

The iterative waterfall model provides feedback paths from every phase to its preceding phases, which is the main difference from the classical waterfall model.



Chapter 8

Implementation

We have implemented the following modules:

SPEECH TO TEXT:

In Android there are basically two ways in which speech recognition can be implemented. First, through the RecognizerIntent which uses the in-built Google text to speak API, second, through The SpeechRecognizer which does not have predefined usability but is very modifiable.

For our project we have used the SpeechRecognizer, which allowed us to have more modifiable capabilities as we wanted to configure our own commands.

```
final SpeechRecognizer mSpeechRecognizer = SpeechRecognizer.createSpeechRecognizer(this);

final Intent mSpeechRecognizerIntent = new Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
mSpeechRecognizerIntent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,
        RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);
mSpeechRecognizerIntent.putExtra(RecognizerIntent.EXTRA_MAX_RESULTS, value: 1);

mSpeechRecognizerIntent.putExtra(RecognizerIntent.EXTRA_LANGUAGE,
        value: "en_IN");
```

Initialization of the Speech Recognizer

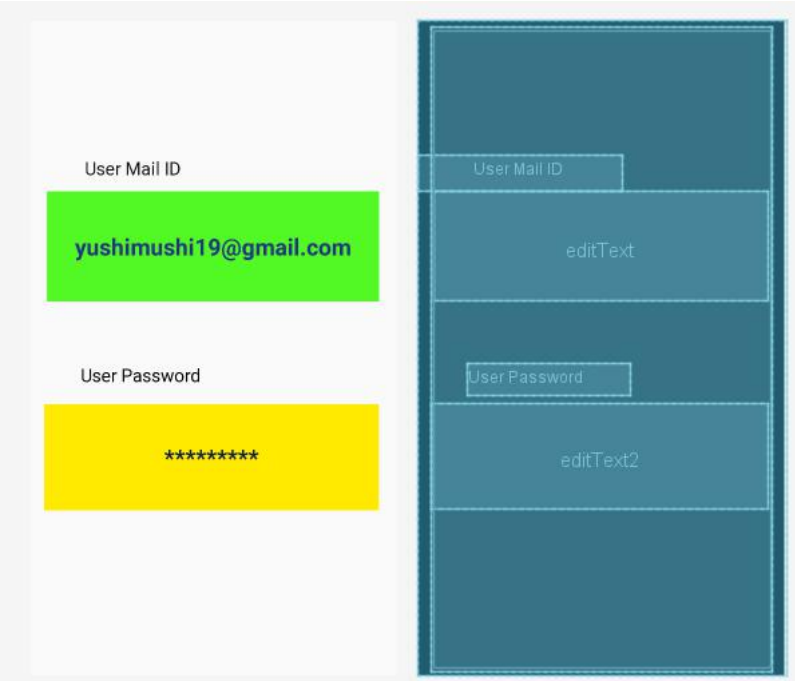
```
if(arr[0].equals("message")||arr[0].equals("massage")){
    arr[0]="";
    temp=arr[1];//.replaceAll(" ","");
    temp=temp.replaceAll( regex: "dot", replacement: ".");
    temp=temp.replaceAll( regex: "attherate", replacement: "@");
    temp=temp.replaceAll( regex: "underscore", replacement: "_");
    temp=temp.replaceAll( regex: "korma", replacement: ",");
    temp=temp.replaceAll( regex: "karma", replacement: ",");
    temp=temp.replaceAll( regex: "comma", replacement: ",");
    temp=temp.replaceAll( regex: "hyphen", replacement: "-");
    temp=temp.replaceAll( regex: "hifriend", replacement: "-");
    temp=temp.replaceAll( regex: "space", replacement: " ");
    temp=temp.replaceAll( regex: "fullstop", replacement: ".");

    finaltext=mMessage.getText().toString();
    mMessage.setText(finaltext+" "+temp);
```

Instance of the speech recognizer for one field i.e Message

LOGIN

The user must log in with an already registered account. Login module will ask user to provide username and password. Here the process goes in speech to text conversation of user. User is told to validate whether he/she entered details are correct or not. If the details are correct then the user is authorized and will enter to the main page.



Visualization of the code for the login page

COMPOSE MAIL

This is not only the most used mail function but also a very important feature of mailing systems. Without compose, one cannot mail. Since the system is for visually challenged people and keyboard operations are completely avoided composing mail is totally done on voice input . No typed input will be required, as the system totally focuses on voice input. User can record the messages by pressing anywhere on the bottom half of the screen and speaking the message.

```
try {
    //Creating MimeMessage object
    MimeMessage mm = new MimeMessage(mSession);

    //Setting sender address
    mm.setFrom(new InternetAddress(mEmail));
    //Adding receiver
    for (String toEmail : toMail) {

        mm.addRecipient(Message.RecipientType.TO,
                new InternetAddress(toEmail));
    }
    //Adding subject
    mm.setSubject(mSubject);
    //Adding message
    mm.setText(mMessage);
    //Sending email
    Transport.send(mm);
```

Calling of MimeMessage object for passing mail data

```
public void SendMail( String fromEmail, String fromPassword,List toEmailList, String emailSubject,
                      String emailBody){
    Log.i( tag: "SendMailActivity",  msg: "Send Button Clicked.");

     fromEmail = ((TextView) findViewById(R.id.MyMailID))
            .getText().toString();
     fromPassword = ((TextView) findViewById(R.id.MyPassword))
            .getText().toString();
    String toEmails = ((TextView) findViewById(R.id.ToMailID))
            .getText().toString();
    toEmailList = Arrays.asList(toEmails
            .split( regex: "\\s*,\\s*"));
    Log.i( tag: "SendMailActivity",  msg: "To List: " + toEmailList);
    emailSubject = ((TextView) findViewById(R.id.ToSubject))
            .getText().toString();
     emailBody = ((TextView) findViewById(R.id.Message))
            .getText().toString();
    new SendMailTask( activity: MainActivity.this).execute(fromEmail,
            fromPassword, toEmailList, emailSubject, emailBody);
```

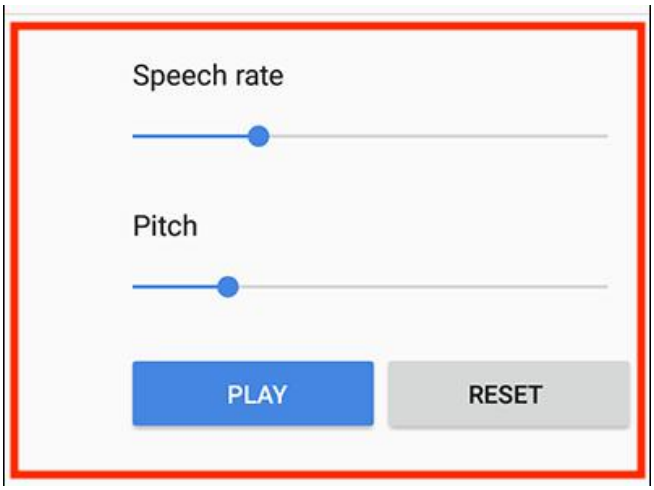
Creation of SendMail class that sets the data before final upload

TEXT TO SPEECH:

Another task of this system is to re-translate the text into equivalent voice format. This process is known as computer speech recognition or automatic speech recognition. There is a microphone which take text as input, a speech recognition software and a soundcard for pronunciation of the texts.

```
t1=new TextToSpeech(getApplicationContext(), new TextToSpeech.OnInitListener() {
    @Override
    public void onInit(int status) {
        if(status != TextToSpeech.ERROR) {
            t1.setLanguage(Locale.UK);
        }
    }
});
```

Implementation of the Text to Speech class



Adjusting of the pitch and speed of speech for better understandability

LANGUAGE CHANGE:

The system supports multiple languages for voice based operations.Keeping in mind the accessibility of the users due to implementation of local languages, we have included the option to change the language at any point of time. Both speech to text and text to speech operations are possible in multiple languages.

```
tempdisplay.setText(str);
if(str.equals("Hindi")||str.equals("hindi")){
    mSpeechRecognizerIntent.putExtra(RecognizerIntent.EXTRA_LANGUAGE,
            value: "hi_IN");
             LanguageToggle=1;
}
if(str.equals("इंग्लिश")||str.equals("इंग्लीश")||str.equals("इंगलिश")){
    mSpeechRecognizerIntent.putExtra(RecognizerIntent.EXTRA_LANGUAGE,
            value: "en_IN");
    LanguageToggle=0;

}
```

Initialization for toggle of language between hindi and english

```
if(hindiarr[0].equals("सब्जेक्ट")||hindiarr[0].equals("सबजेक्ट")||hindiarr[0].equals("सब्जैक्टट") ){
    hindiarr[0]="";

    finaltext=mSubject.getText().toString();
    mSubject.setText(finaltext+" "+hindiarr[1]);
    matches.clear();
}
```

Implementation of Language change for one context i.e Subject

Conclusion and Future Scope

10.1 Conclusion

We have proposed a system which is helpful for visually impaired people to access email services efficiently through both the facilities of voice - to - text and text- to - voice conversion along multiple languages. The system is completely based on interactive voice response which will make it user friendly and efficient to use.

The main benefit of this system is that the requirement of keyboard and visual input is completely eliminated, the user will have to operate through voice commands given after pressing anywhere in the phone screen.

10.2 Future Scope

The major drawbacks of the application can be used as the future enhancements for this project. There are two major drawbacks in this application i.e..,

- The exact voice recognition. The accuracy of the converted text can be improved.

- Image or document attachment. So in the future enhancement, we can add the image or document attachment for the sender.

- The users have to remember certain keywords for them to use the application which is sometimes not feasible. Since ours is a user-friendly oriented application, this aspect can be improved upon in the future

References

[1]. C. Kang, H. Jo and B. Kim, "A Machine-to-Machine based Intelligent Walking Assistance System for Visually Impaired Person", The Journal of KICS, vol. 36, no. 3, (2011), pg. 195-304.

[2]. S. Kumar, M. A. Qadeer and A. Aupta, "Location Based Service using Android", Internet Multimedia Service Architecture and Applications, IEEE International Conference, (2009),vol 45,pg.205-208.

[3]. H. -W. Jung, "Smartphones and future changes", The Korea Contents Association(2008), vol. 8,pg 209-298. .

[4]. Rupavathy N, Dr Carmel Mary Belinda and G.Nivedhitha, "A mobile application using IoT enabled navigation system for bus riders" International Journal of Engineering &Technology, Vol 7 (1.7) (2018) 71-74.

[5]. Shoba, G. Anusha, V. Jeevitha, R. Shanmathi. "AN Interactive Email for Visually Impaired". In International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), 2014 on Pages 5089-5092.(Volume 3, Issue 1).

VOICE BASED EMAIL APP

AKSHAY GOURISARIA
1705110

Abstract: We have implemented a Voice operated Mail Android application that can be used by a blind person to send E-Mails easily and efficiently with the use of voice and a single screen touch. The Mail system is based on JavaMail API and uses the in-built Android Speech recognition system for converting text into speech and speech into text. The system is completely based on interactive voice response which will make it user friendly and efficient to use.

Individual contribution and findings:

Implemented the Speech Recognition module in the Android App using the in-built Android Speech Recognizer. Found out the logic behind speech recognition and the way to apply Text to Speech functionality in our project. Speech recognition is the ability of a machine or program to identify words and phrases in spoken language and convert them to a machine-readable format. Rudimentary speech recognition software has a limited vocabulary of words and phrases, and it may only identify these if they are spoken very clearly.
Had provided the code for Speech to Text implementation for the app.

Individual contribution to project report preparation:

Contributed to the chapters 'Introduction' in the part of 'Text to Speech' and prepared the 'System Testing' portion.

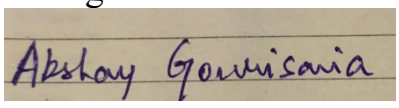
Individual contribution for project presentation and demonstration:

Prepared and demonstrated the 'Speech to Text' part of the presentation

Full Signature of Supervisor:

…….

Full signature of the student



VOICE BASED EMAIL APP

ANIK BISWAS
1705112

Abstract:

We have implemented a Voice operated Mail Android application that can be used by a blind person to send E-Mails easily and efficiently with the use of voice and a single screen touch. The Mail system is based on JavaMail API and uses the in-built Android Speech recognition system for converting text into speech and speech into text. The system is completely based on interactive voice response which will make it user friendly and efficient to use.

Individual contribution and findings:

Implemented the part for editing sentences by inserting punctuation marks. This helps in properly formatting the the text message as punctuation marks are not recognized by the speech recognizer by default.

Laid out the idea and logic for implementing multiple language functionality in the project. Also provided the code for the same. The option to change the language for writing and hearing is essential for the app to be truly user-friendly.

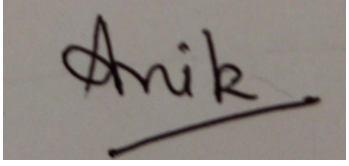
Individual contribution to project report preparation:

Contributed to the chapters of 'System Design' and 'Introduction' in the part of 'Voice commands and Multiple Language Support'

Individual contribution for project presentation and demonstration:

Done the main demonstration of the presentation. Also prepared the documentation for the same and the introduction.

Full Signature of Supervisor: Full signature of the student



…….

VOICE BASED EMAIL APP

ASHUTOSH MALLICK
1705127

Abstract:

We have implemented a Voice operated Mail Android application that can be used by a blind person to send E-Mails easily and efficiently with the use of voice and a single screen touch. The Mail system is based on JavaMail API and uses the in-built Android Speech recognition system for converting text into speech and speech into text. The system is completely based on interactive voice response which will make it user friendly and efficient to use.

Individual contribution and findings:

The mind behind the inception of the app. Provided the idea for the project topic.

Laid the idea and provided the logic for implementation of 'User Login' feature. Also provided the code for the layout of the login screen and back-end code for the same feature. Through this feature, user does not have to provide his/her credentials everytime he/she opens the app.

Created the User Interface (UI) for the whole application by the use of Intents, Views and Constraint Layout. The proper UI helps in smoothening the workflow of the user.

Individual contribution to project report preparation:

Contributed to the chapters of 'System Requirement Specification' and 'Requirement Analysis'

Individual contribution for project presentation and demonstration:

Prepared and presented the UI part of the presentation

Full Signature of Supervisor: Full signature of the student

Ashutosh Mallick

……

VOICE BASED EMAIL APP

ARUNAVA ROY
1705126

Abstract:

We have implemented a Voice operated Mail Android application that can be used by a blind person to send E-Mails easily and efficiently with the use of voice and a single screen touch. The Mail system is based on JavaMail API and uses the in-built Android Speech recognition system for converting text into speech and speech into text. The system is completely based on interactive voice response which will make it user friendly and efficient to use.

Individual contribution and findings:

Laid the idea and provided the logic for implementation of 'Text to Speech' feature of the application. By implementing this feature, we provide a seamless experience to the user for truly voice based application. Provided the code for the this feature.

Had given the logic and code for implementing operations through voice commands like delete last line, read the text, goto next line. Voice operations help in navigating through the fields and makes the app more usable to the user.

Individual contribution to project report preparation:

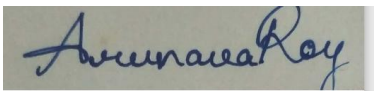
Contributed to the chapter of 'Literature Survey'. Found out about related works and 'References'.

Individual contribution for project presentation and demonstration:

Prepared and presented the Text to Speech part of the presentation

Full Signature of Supervisor: Full signature of the student

…….



VOICE BASED EMAIL APP

RISHIVANSH PATNAIK
1705160

Abstract:

We have implemented a Voice operated Mail Android application that can be used by a blind person to send E-Mails easily and efficiently with the use of voice and a single screen touch. The Mail system is based on JavaMail API and uses the in-built Android Speech recognition system for converting text into speech and speech into text. The system is completely based on interactive voice response which will make it user friendly and efficient to use.

Individual contribution and findings:

Laid the idea and provided the logic for 'Mail Sending' feature of the application. Sending the mail is the final and main goal of the appication.

Gave the idea of using JavaMail API for the purpose of sending mail. Provided the code and implemented JavaMail API to send mail through Google's SMTP port.

Provided the code for integrating the fields required for sending mail.

Individual contribution to project report preparation:

Contributed to the chapter of 'Introduction' and 'Implementation.

Individual contribution for project presentation and demonstration:

Prepared and presented the Sending Mail part of the presentation

Full Signature of Supervisor:
…….

Full signature of the student