

BLIND ASSISTANT APP

AHMAD RAZA

REG. NO. CIIT/SP18-BCS-004/VHR

ABDUL QADIR KHAN

REG. NO. CIIT/SP18-BCS-046/VHR



**DEPARTMENT OF COMPUTER SCIENCES
COMSATS UNIVERSITY ISLAMABAD, VEHARI CAMPUS
VEHARI – PAKISTAN**

SESSION 2018-2022

Blind Assistant App

Undertaken By:

AHMAD RAZA

REG. NO. CIIT/SP18-BCS-004/VHR

ABDUL QADIR KHAN

REG. NO. CIIT/SP18-BCS-046/VHR

Supervised By:

UZAIR ISHTIAQ



A DISSERTATION SUBMITTED AS A PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF BACHELORS IN COMPUTER SCIENCE /
SOFTWARE ENGINEERING

DEPARTMENT OF COMPUTER SCIENCES
COMSATS UNIVESITY ISLAMABAD, VEHARI CAMPUS
VEHARI – PAKISTAN

SESSION 2018-2022

CERTIFICATE OF APPROVAL

It is to certify that the final year project of BS(CS) “Blind Assistant App” was developed by **AHMAD RAZA (CIIT/SP18-BCS/004/VHR)** and **ABDUL QADIR KHAN (CIIT/SP18-BCS/046/VHR)** under the supervision of “UZAIR ISHTIAQ” and that in his opinion; it is fully adequate, in scope and quality for the degree of Bachelors of Science in Computer Sciences.

Supervisor

External Examiner

Head of Department
(Department of Computer Science)

DEDICATION

We feel great pleasure in expressing my heartiest gratitude to my family and our honorable **HOD Dr. Ali Shahid, DOO office staff** and **Supervisor Mr. UZAIR ISHTIAQ** (Lecturer, Department of Computer Science COMSATS University Islamabad, Vehari (Campus) for kind behavior, valuable suggestions, worth and keen supervision, scholarly criticisms, sympathetic attitude towards completion of this thesis. His kind behavior and attitude during period of project work is unforgettable. We express our sincere thanks to all respectable teachers and faculty members of Computer Science Department of COMSATS University Islamabad, Vehari (Campus). I feel great happiness in expressing my thanks to my family for their love and support. Prayers of my family are a treasure for my life. So, I submit my earnest thanks again to all of them for their encouragement and moral support.

ACKNOWLEDGEMENT

All praise to Almighty Allah alone, the Omnipotent, the most compassionate. His prophet “**MUHAMMAD**” (peace be upon him), the most perfect and exalted among and of ever born on the surface of the earth, who is forever torch of guidance and knowledge for humanity as a whole We feel great pleasure in expressing my heartiest gratitude to my family and our honorable **HOD Dr. Ali Shahid, DOO office staff** and **Supervisor Mr. UZAIR ISHTIAQ** (Lecturer, Department of Computer Science COMSATS University Islamabad, Vehari (Campus) for kind behavior, valuable suggestions, worth and keen supervision, scholarly criticisms, sympathetic attitude towards completion of this thesis. His kind behavior and attitude during period of project work is unforgettable. We express our sincere thanks to all respectable teachers and faculty members of Computer Science Department of COMSATS University Islamabad, Vehari (Campus). I feel great happiness in expressing my thanks to my family for their love and support. Prayers of my family are a treasure for my life. This dissertation report is a proof of sincerity of those who helped us during this project work. So, I submit my earnest thanks again to all of them for their encouragement and moral support. Finally, we pray for health happiness and prosperity of all the participants.

PROJECT BRIEF

PROJECT NAME	BLIND ASSIDTANT APP
ORGANIZATION NAME	COMSATS UNIVERSITY ISLAMABAD
OBJECTIVE	FOR HELP OF BLIND PEOPLE
UNDERTAKEN BY	AHMAD RAZA & ABDUL QADIR KHAN
SUPERVISED BY	UZAIR ISHTIAQ LECTURER COMPUTER SCIENCE COMSATS UIVERSITY ISLAMABD (VEHARI)
STARTED ON	01 APRIL, 2021
COMPLETED ON	30 NOVEMBER, 2021
COMPUTER USED	HP CORE I5 6 TH GEN
SOURCE LANGUAGE	DART PROGRAMING 2.14.3 VERSION
OPERATING SYSTEM	WINDOWS 10 PRO
TOOLS USED	ANDROID STUDIO

ABSTRACT

Today's technology helping people in every field of life, but visually impaired peoples are not getting benefit of technology in their life as like many people are using smartphones by seeing the feature. But Blind people are not able to see the screen and use these features. Like if they want to make a call to someone from contact list but they are blind so they can't do this as normal people. If they are received a message from someone, how could they know who send this message and what is the message. So, we are developing a system that can help visually impaired people to control their smartphone by their voice. All action performed by the user the system speaks out and help the use to know his current position.

Contents

CHAPTER 1: INTRODUCTION.....	12
1.1 System Introduction	12
1.2 Background of the System	12
1.3 Objectives of the System.....	12
1.4 Significance of the System.....	13
CHAPTER 2: REQUIREMENT SPECIFICATIONS	14
2.1 Product Scope	14
2.2 Product Description	14
2.2.1 Product Perspective.....	14
2.2.2 Product Functionality	15
2.2.3 Users and Characteristics	15
2.2.4 Operating Environment.....	15
2.3 Specific Requirements	15
2.3.1 Functional Requirements	15
2.3.2 Behavioral Requirements	17
2.3.3 External Interface Requirements.....	18
2.4 Non-functional Requirements	21
2.4.1 Performance Requirements	21
2.4.2 Safety and Security Requirements	21
2.4.3 Software Quality Attributes	21
CHAPTER 3: DESIGN SPECIFICATIONS	23
3.1 System Design	23
3.2 Logical Design.....	24
3.3 System Architecture.....	24
3.4 System Interaction and Use Cases	25
CHAPTER 4: DEVELOPMENT AND TOOLS	26
4.1 Introduction	26
4.2 Development	26
4.2.1 Tools and Technologies Used.....	26
4.3 System Implementation.....	27
4.4 Data and Information flow in the system	27
4.5 User Interfaces:.....	28
4.6 Additional Modules and API Provided	28
CHAPTER 5: QUALITY ASSURANCE.....	29
5.1 Introduction	29

5.2 Traceability Matrix 29

5.3 Test Plan..... 29

List of Figures

Figure 1. 1: Product Prospective	14
Figure 2. 1: System level use case diagram.....	17
Figure 2. 2: Home screenshot	18
Figure 2. 3: Call screenshot	19
Figure 2. 4: Send message screenshot.....	20
Figure 2. 5: Received SMS screenshot	20
Figure 2. 6: Saved notes screenshot	20
Figure 2. 7: Write note screenshot	20
Figure 2. 8: Battery screenshot	21
Figure 3. 1: Package diagram	23
Figure 3. 2: Class diagram	24
Figure 3. 3: ERD diagram.....	24
Figure 3. 4: Sequence diagram.....	25
Figure 4. 1: Data flow diagram.....	27
Figure 5. 1: Traceability matrix table screenshot.....	29

List of Tables

Table 5. 1: Test case for Application start up.....	30
Table 5. 2: Test case for open Message Module	30
Table 5. 3: Test case for open Call Module	30
Table 5. 4: Test case for open Notes Module	31
Table 5. 5: Test case for open Battery Module	31
Table 5. 6: Test case for make a Call	31
Table 5. 7: Test case for read new message	32
Table 5. 8: Test case for open write Message.....	32
Table 5. 9: Test case for send new Message.....	32
Table 5. 10: Test case for Save Notes	33
Table 5. 11: Test case for check battery level & speak	33

CHAPTER 1: INTRODUCTION

1.1 System Introduction

The World Health Organization (WHO) estimates that 253 million people live with visual impairment. 217 million of those have moderate to severe vision impairment and 37 million are blind. This is an innovative System for visually impaired people and acts as a voice assistant for them. This system is used to help the visually impaired to have access to the most important features of the phone enhancing the quality of the system making use of different custom layouts and using speech to text. This system makes a blind to user self-dependent. This system allows visually impaired to people to get connected with world easily.

1.2 Background of the System

Many researchers have contributed to this field. Various combinations of existing technologies have been used. Braille systems, screen magnifiers, etc. went through some developments but later faced technical issues. The fusion of several sensors is one of the techniques used for obstacle detection, where combination of visual sensors, sonar and inertial measurement unit are used to detect the presence of an obstacle and give audio as well as tactile feedback to user. Another system was to implement OCR using Raspberry Pi sensor for automatic recognition of the environmental messages and by utilization of text to speech.

Android phone-controlled voice gesture and touchscreen operated wheelchair where voice and gesture are recognized through android. Developers also created a universal voice control on android which is used to launch android application via voice commands.

1.3 Objectives of the System

The System will have

- Easy to use mobile phones for blind people
- Messaging feature
- Call log feature
- Notes making feature

- Battery status feature
- Text To Speech
- Speech To Text
- Convenience

1.4 **Significance of the System**

The proposed system is to build a customized application for visually impaired people. This application acts as a voice assistant. This application is used to help the visually impaired to access most important features of the phone using text to speech and speech to text. The System will have custom messaging feature, call log feature, notes making feature and battery status feature in it. This system will speak out all the actions performed by user. This system in all is a voice assistant for whatever action the user has performed through a custom app. The custom app having these features will allow visually impaired users to do their basic things using without any other help.

CHAPTER 2: REQUIREMENT SPECIFICATIONS

2.1 Product Scope

The scope of our project is to designing a complete environment to provide a safe and user-friendly environment for blind people. The main aim of the project is to provide an easy-to-use mobile phone for blind people.

The user of this system is educated. This system is for facilitate and helping the visually impaired people. The System in all is a voice assistant for whatever action the user has performed though a custom app while taking the data from the default application. Blind people are wanting to connect the world but they are visually impaired and cannot use mobile phones. We remove this hurdle and provide facilities to those people.

2.2 Product Description

2.2.1 Product Perspective

This system provides facilities to blind people. We are trying to provide easiness virtually impaired people. A blind user can use this system by voice and get output in speech. User can do calls, making notes, manage notes, and get battery status.

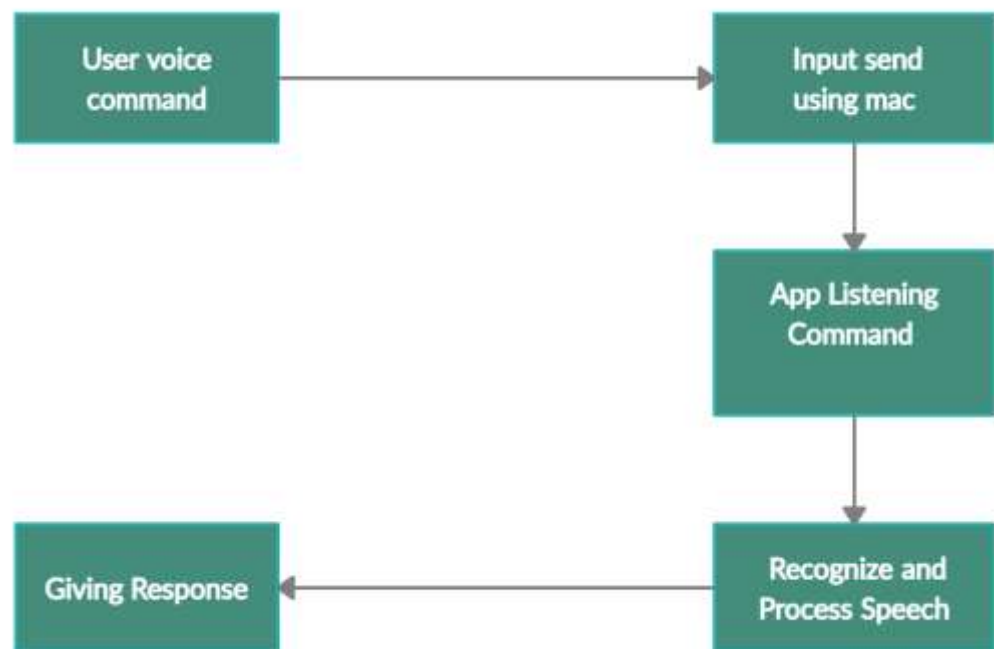


Figure 1. 1: Product Prospective

2.2.2 Product Functionality

The major functionalities of the product:

- Call log
- Messages
- Notes
- Battery status
- Text to Speech
- Speech to Text

2.2.3 Users and Characteristics

The user of this application is blind. The application has four modules. Visually impaired user uses this app with speech and listening. User can do call, messages, making notes, and check battery status. These all functionalities perform with user speech and app output also in speech form. Basically, app and user will communicate with each other.

2.2.4 Operating Environment

Android:

- Version: 8 and above
- Processor: core 2.0 and above
- RAM: 2GB or more
- Memory: 8 GB and more

IOS:

- Version: 11 and above
- Processor: core 2.0 and above

2.3 Specific Requirements

2.3.1 Functional Requirements

Basic functional requirements are given in section 2.2.2.

This app has four module Call, Messages, Notes and Battery.

- In Call module user will make call to his contacts.
 1. When the user clicks on the screen, the app will start listening to it
 2. The user will say the name of the person they want to call. Speak as “Call to X

person Name”

3. App search the user’s name and dialed call.
- In Message module user will send or receive message.
 1. When user open message module, app will start read the new message.
 2. User tap on screen. App will start to listen.
 3. When the user clicks on the screen, the app will start listening to it.
 4. The user will say the name of the person they want to send message. Speak as “message to *X person Name*”.
 5. App search the user from contact list. If person found from contact list send message page will open otherwise app speak “*X person* not found”.
 6. On send message page user clicks on the screen, the app will start listening to it. User speak his message that he wants to send.
 7. On long press app start read the written message.
 8. On long press and swap up message will send to the person.
 9. On double tap written message will be clear.
 - In notes module user will save his notes and listen his saved notes.
 1. From home screen user say “Open Notes”. Notes module will open.
 2. The first screen will show the user its saved notes.
 3. User double click on any note’s app will speak this note.
 4. When the user clicks on the screen, the app will start listening to it.
 5. User say “save a note”. Write message page will open.
 6. On write note page user clicks on the screen, the app will start listening to it. User speak his notes that he wants to save.
 7. On long press app start read the written note.
 8. On long press and swap up message will save the note.
 9. On double tap written note will be clear.
 - In Battery module user will check his phone battery level and state.
 1. On Battery Page user clicks on the screen, the app will speak battery level and state (charging or not charging).

2.3.2 Behavioral Requirements

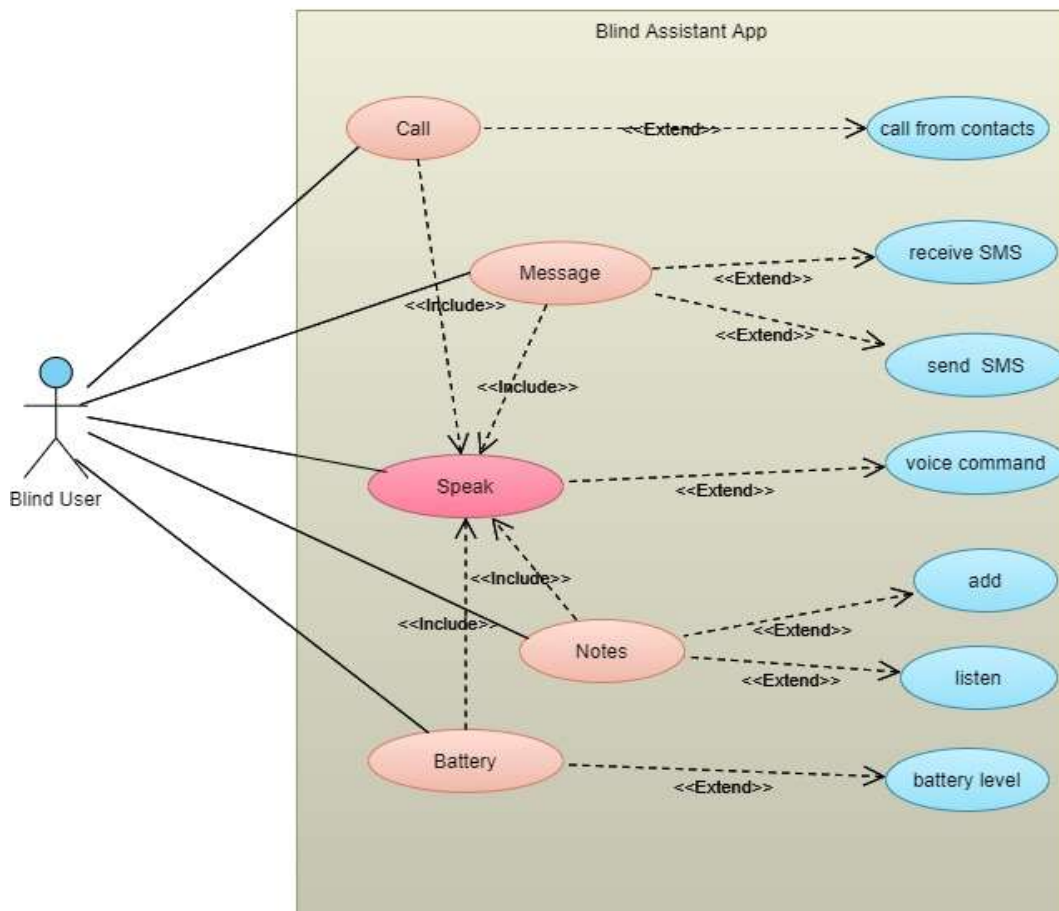


Figure 2. 1: System level use case diagram

1	Speech To Text (STT) OR Text to Speech (TTS):	In "Blind Assistant App" everything based on user voice input For every command user will speak and perform action. On the other end app will also response I voice which is called text to speech.
2	Call Handling:	In call handling user will be able to call any one he wants. He can select a contact from phonebook of mobile phone by calling the name by which he saved the contact. He can also dial a new number by voice.
3	SMS:	In SMS function he can listen any SMS delivery time, person name who send this and the content of SMS. As well he can write a SMS by voice and send.

4	Notes:	In this module User can add notes to keep things in mind. These notes will also be written by speech to text. He can also be able to delete or listen these notes when needed.
5	Battery:	In this module user will able to know his battery percentage and status of charging or discharging by voice.

Table 2. 1: Description of major use-cases

2.3.3 External Interface Requirements

2.3.3.1 User Interface

1. Home

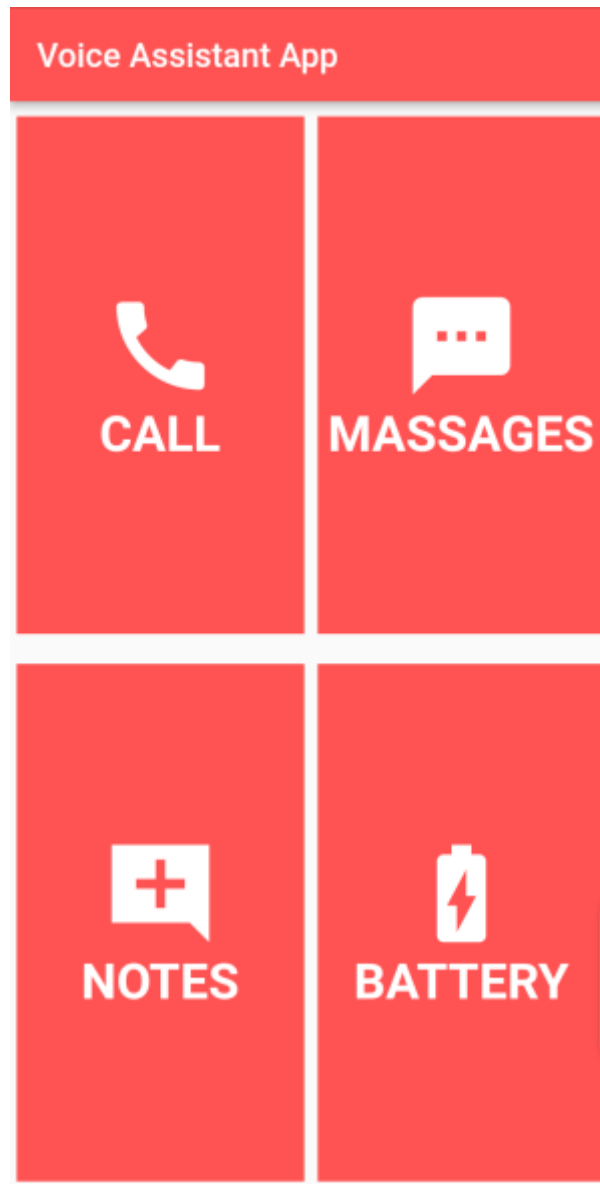


Figure 2. 2: Home screenshot

- Call
- Messages
- Notes
- Battery

This screen has only one event that is tap on screen.

On Tap:

On Single Tap app will start to listen the user if user speak “Open Call” then call module open and other will be open like this.

2. Call

When we double click on call button call logs will be open and we will call a contact name like “call to Ahmad” then if contact found then calling will be start. But if you want to dial a new number the go to dialer and speak the number and make a call.

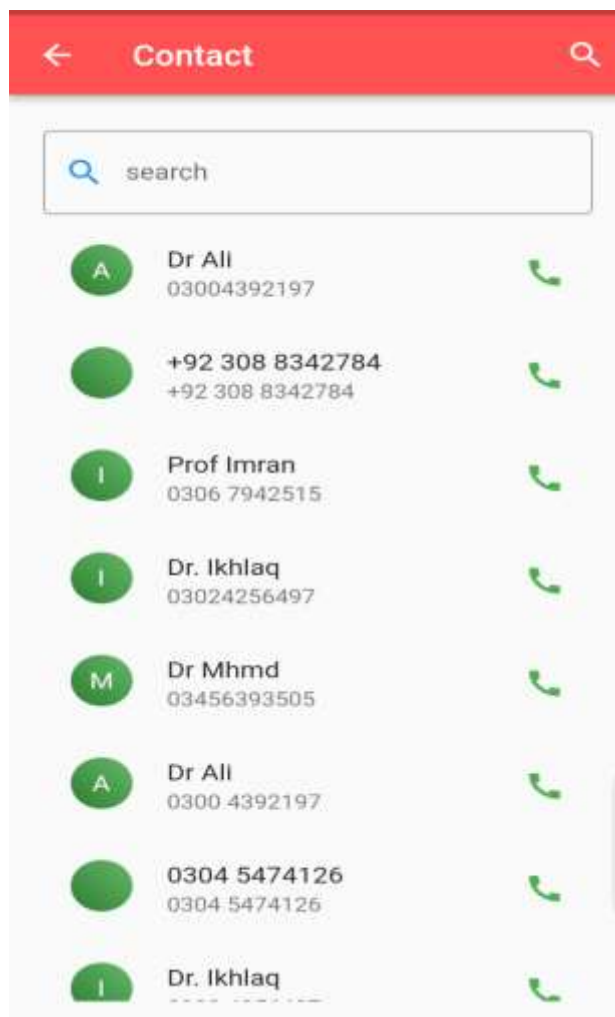


Figure 2. 3: Call screenshot

3. Message

This is SMS module where we can listen every SMS from inbox and can also send a new SMS.



Figure 2. 4: Send message screenshot



Figure 2. 5: Received SMS screenshot

4. Notes

Here user can write note to remember thing. He will speak which he wants to write and Speech to Text will convert it into text it will be saved in Firebase.



Figure 2. 6: Saved notes screenshot



Figure 2. 7: Write note screenshot

5. Battery

Here user can check battery percentage and its state whether it is charging or discharging. Touch anywhere on the screen it will speak.

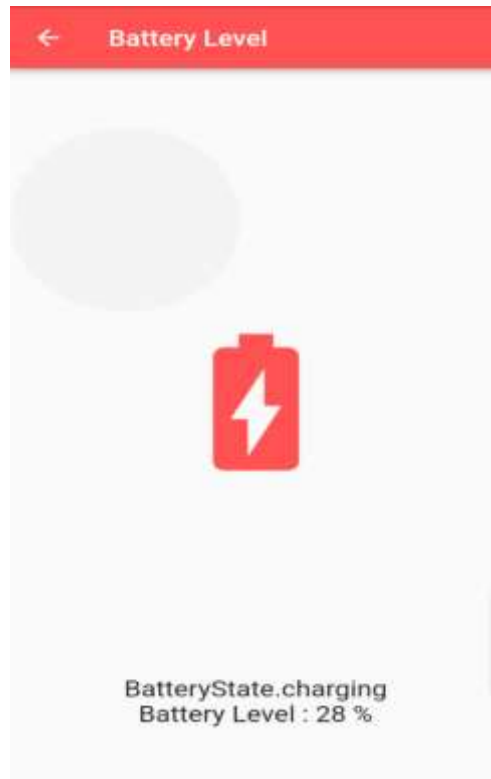


Figure 2. 8: Battery screenshot

2.4 Non-functional Requirements

2.4.1 Performance Requirements

Size of this app is kept responsive, so it does not crash on any device. The system must perform the intended functions and operations without experiencing failures. And compatible for all android mobile versions. It doesn't take much time for any operation. it is a quick responsive app.

2.4.2 Safety and Security Requirements

This App is completely secure as it doesn't contain any personnel data of user and also no need of Signup or Signing. The database is required for only Notes storage. But if any personnel information is to be stored then don't worry there will be no external access to the database; only Developers can Access.

2.4.3 Software Quality Attributes

Software Quality Attributes are features that facilitate the measurement of performance of a

software product by Software Testing professionals, and include attributes such as availability, correctness, maintainability, useability. Here the purposed system fulfills all these attributes by performing best with best availability, correctness, maintainability, useability. where the system performs in well manners in every android mobile in good environment.

CHAPTER 3: DESIGN SPECIFICATIONS

3.1 System Design

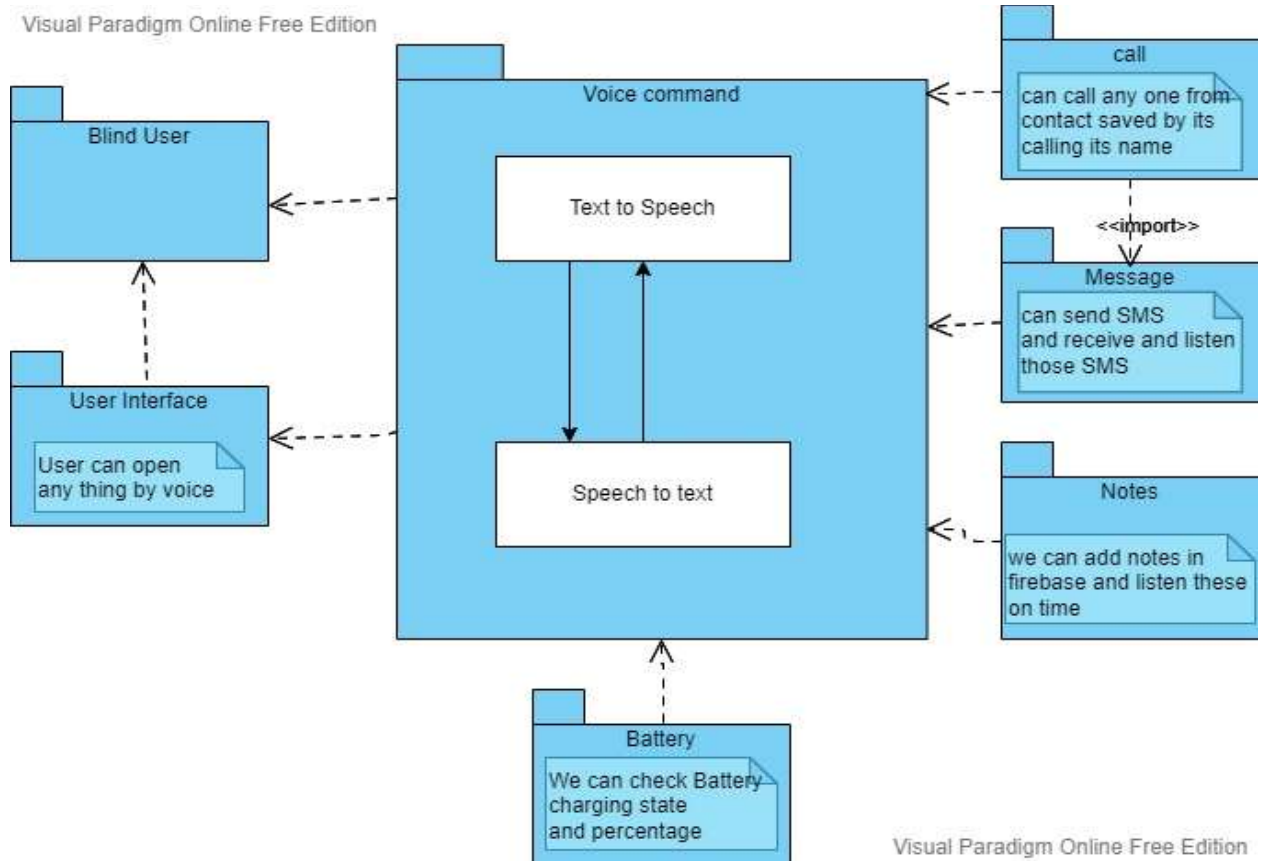


Figure 3. 1: Package diagram

3.2 Logical Design

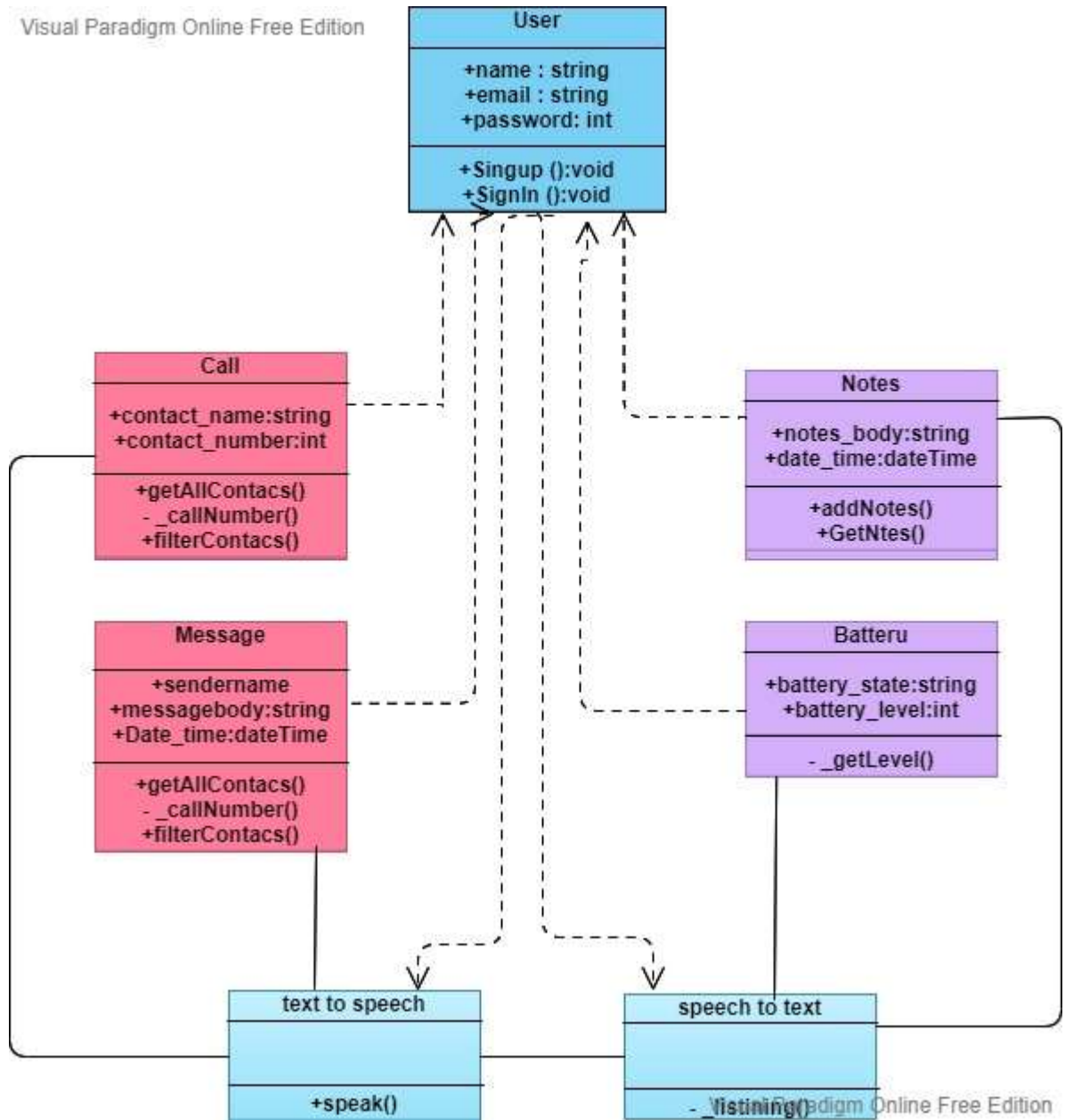


Figure 3. 2: Class diagram

3.3 System Architecture



Figure 3. 3: ERD diagram

3.4 System Interaction and Use Cases

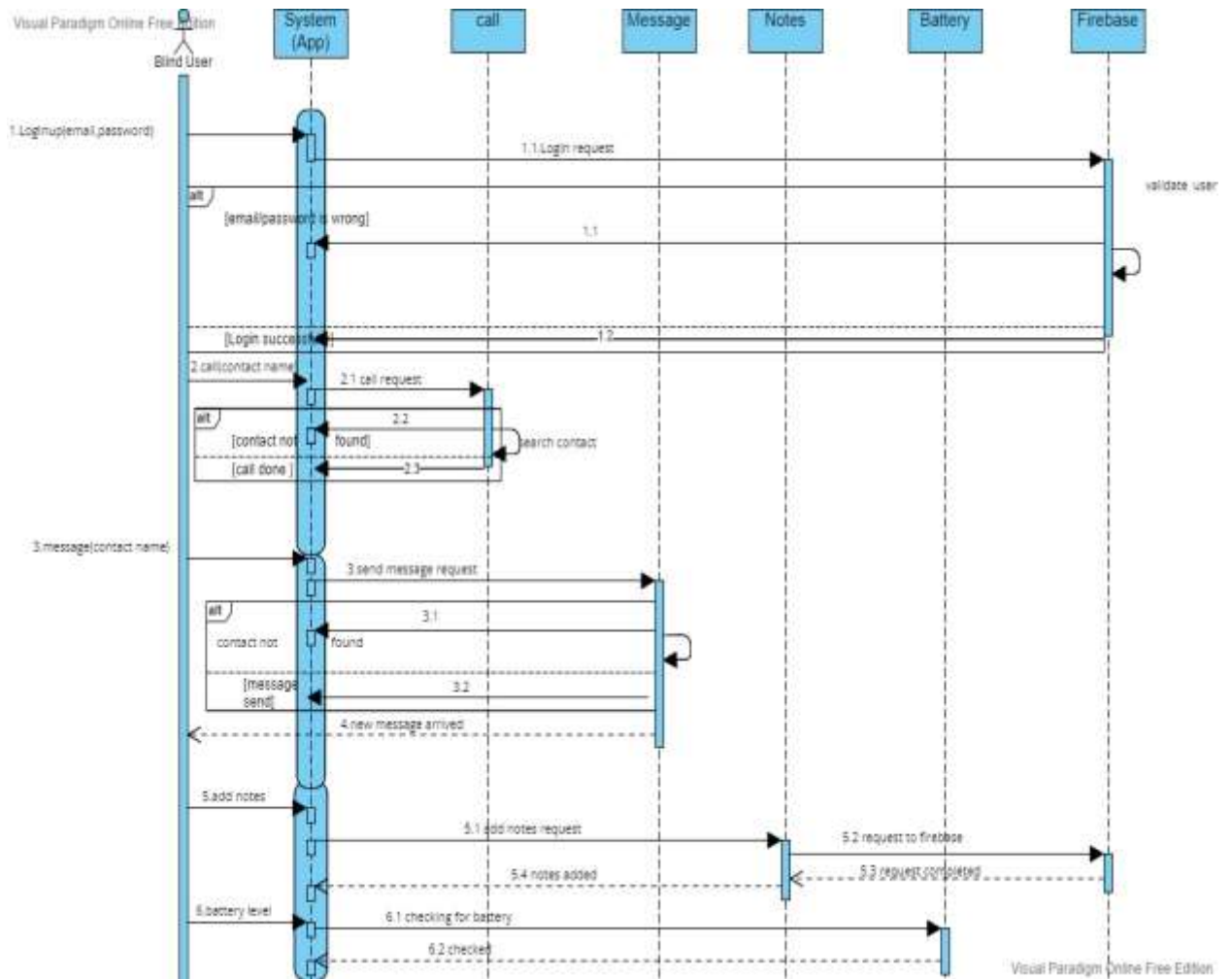


Figure 3. 4: Sequence diagram

CHAPTER 4: DEVELOPMENT AND TOOLS

4.1 Introduction

Here we will discuss how we implemented blind assistant app. Basically google is controlling the world now in each filed, and in 2017 google introduce a UI Framework for building Mobile Apps that is flutter. Flutter is very interesting and it provide many flavours of one project like Android, IOS, web and desktop application. We use Dart programming language to develop our app.

4.2 Development

Hardware Used:

- Laptop Hp Core i5 6th Generation, RAM: 8 GB, CPU Processor: Intel(R) Core (TM) i5-6300U CPU @ 2.40GHz, 2.50GHz
- Android Mobile Phones

Operating System:

- Windows 10 Pro

4.2.1 Tools and Technologies Used

Programming Language:

- Dart (2.14.3)

Tools:

- Flutter UI Framework (2.5.2)
- IDE Android studio (2020.3.1)
- MS word (2019)
- MS Power Point (2019)
- MS Visio (2019)
- Online-visual Editor
- Pencil
- Lucid chart online UML diagram tool
- creately online UML diagram tool
- Microsoft Edge search Engine

Database:

- Google Firebase Database

4.3 System Implementation

Algorithm:

- Speech to text Algorithm
- Text to speech Algorithm

User Interface:

- We implemented our app user interface with Google UI Toolkit Flutter

User Side:

- This app for Blind User, who can use it with their voice
- He will make call send SMS etc. by voice.
- User just install app and after registration process he can utilise app services.
- Now special admin we control app the developer is admin side.
- Developers maintain app and know about the user.

4.4 Data and Information flow in the system

The Data flow of our app is very simple user use his voice to control the app and perform any action. When User speech his voice will be converted into text and app will understand this text and action accordingly. If app need to provide any information to user text to speech algorithm will convert that message into voice which is understandable by the user. Following diagram shows Data flow of our system.

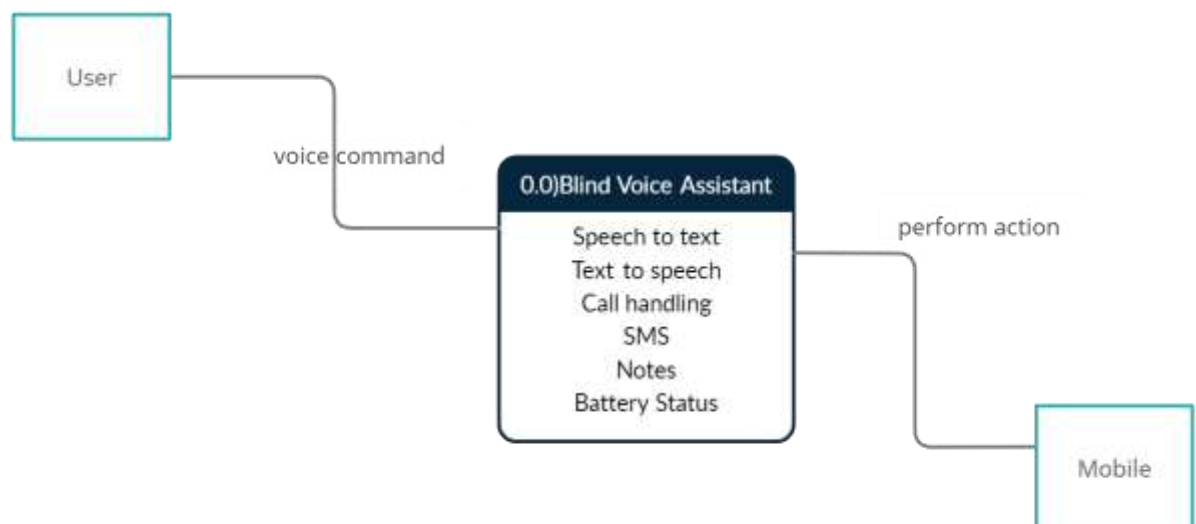


Figure 4. 1: Data flow diagram

4.5 User Interfaces:

App Interface is user friendly. But according to **HCI** (Human Computer Interaction) Point of view this app will not **affect** user eye sight or colour problem. But when user needs any thing like **needs to** open mic he just **taps** on screen and mic will be ready to listen him.

4.6 Additional Modules and API Provided

- Additional Module & API not used

Future Plan:

In Future our plan is:

- Add multiple languages that different languages user will get benefit from this app.
- We can add multiple features like email send and receive by voice.
- Send or receive message on Whatsapp by voice.

CHAPTER 5: QUALITY ASSURANCE

5.1 Introduction

We are applied different strategies on our project for quality assurance that is Unit Testing, Integration Testing, Validating Testing and System Testing. Result of these strategies are given blow. Every unit of app have different cases. We tested the different app unit according to case and the result are shown in Traceability Matrix table 5.1.

5.2 Traceability Matrix

Requirement Traceability Matrix													
	Test Case ID	TC1	TC2	TC3	TC4	TC5	TC6	TC7	TC8	TC9	TC10	TC11	Test Case for respective Requirement
Req. ID													
Req. 1		×	×	×	×	×							5
Req. 2							×						1
Req. 3								×	×	×			3
Req. 4											×		1
Req. 5												×	1

Figure 5. 1: Traceability matrix table screenshot1

5.3 Test Plan

Fist, we perform unit testing each unit of application tested according to the different cases. Basically, this app has four units Call, Message, Notes and Battery. First, each unit developed separately and then perform unit testing on it. **Second**, after unit testing, integrate all the units and perform integration testing. In this process of testing test the interface between modules and expose faults in the interaction between integrated units. **Third**, Perform Validating testing, Requirements are validated against the constructed app.

Fourth, Finally, perform System Testing test the app other system elements are tested as a whole.

Table 5. 2: Test case for Application start up

Test ID	TC-1
Test name	Application start up
Date of test	01/12/2021
Name of application	Blind Assistant
Description	Home screen will be displayed where user will speak “Back to Home” from a module.
Input	Tap on screen and speak “Back to Home”
Expected output	Home screen displayed
Actual output	Home screen displayed
Test Role (Actor)	Team Member
Test verified by	Team Member/Supervisor

Table 5. 3: Test case for open Message Module

Test ID	TC-2
Test name	Open Message Module
Date of test	01/12/2021
Name of application	Blind Assistant
Description	Open Message module when user will tap on home screen and speak “Open Message”
Input	Tap on the screen and Speak “Open Message”
Expected output	Message screen displayed
Actual output	Message screen displayed
Test Role (Actor)	Team Member
Test verified by	Team Member/Supervisor

Table 5. 4: Test case for open Call Module

Test ID	TC-3
Test name	Open Call Module
Date of test	01/12/2021
Name of application	Blind Assistant
Description	Open Call module when user will tap on home screen and speak “Open Call”
Input	Tap on the screen and Speak “Open Call”
Expected output	Contact screen displayed
Actual output	Contact screen displayed
Test Role (Actor)	Team Member
Test verified by	Team Member/Supervisor

Table 5. 5: Test case for open Notes Module

Test ID	TC-4
Test name	Open Notes Module
Date of test	01/12/2021
Name of application	Blind Assistant
Description	Open Notes module when user will tap on home screen and speak “Open Notes”
Input	Tap on the screen and Speak “Open Notes”
Expected output	Notes screen displayed
Actual output	Notes screen displayed
Test Role (Actor)	Team Member
Test verified by	Team Member/Supervisor

Table 5. 6: Test case for open Battery Module

Test ID	TC-5
Test name	Open Notes Module
Date of test	01/12/2021
Name of application	Blind Assistant
Description	Open Battery module when user will tap on home screen and speak “Open Battery”
Input	Tap on the screen and Speak “Open Battery”
Expected output	Battery screen displayed
Actual output	Battery screen displayed
Test Role (Actor)	Team Member
Test verified by	Team Member/Supervisor

Table 5. 7: Test case for make a Call

Test ID	TC-6
Test name	Make a Call
Date of test	01/12/2021
Name of application	Blind Assistant
Description	Open Message module when user will tap on screen and speak “Call to X Person”
Input	Tap on the screen and Speak “Call to X Person”
Expected output	Phone call made
Actual output	Phone call made
Test Role (Actor)	Team Member
Test verified by	Team Member/Supervisor

Table 5. 8: Test case for read new message

Test ID	TC-7
Test name	Read new Messages
Date of test	01/12/2021
Name of application	Blind Assistant
Description	Read new messages when message module open
Input	Message Screen open
Expected output	Read new messages
Actual output	Read new messages
Test Role (Actor)	Team Member
Test verified by	Team Member/Supervisor

Table 5. 9: Test case for open write Message

Test ID	TC-8
Test name	Write Message
Date of test	01/12/2021
Name of application	Blind Assistant
Description	Write message screen open when user speak “Send message to X person”
Input	Tap on the screen and Speak “Message to X Person”.
Expected output	Write Message Screen open
Actual output	Write Message Screen open
Test Role (Actor)	Team Member
Test verified by	Team Member/Supervisor

Table 5. 10: Test case for send new Message

Test ID	TC-9
Test name	Send new Messages
Date of test	01/12/2021
Name of application	Blind Assistant
Description	Write message when user tap on screen and speak message. After write long press on screen.
Input	Tap on the screen. After write message long press on screen
Expected output	Write and send message
Actual output	Write and send message
Test Role (Actor)	Team Member
Test verified by	Team Member/Supervisor

Table 5. 11: Test case for Save Notes

Test ID	TC-10
Test name	Save Notes
Date of test	01/12/2021
Name of application	Blind Assistant
Description	Tap on screen speak your notes and long press for save
Input	Tap on the screen speak notes & long press to save
Expected output	Notes saved
Actual output	Notes saved
Test Role (Actor)	Team Member
Test verified by	Team Member/Supervisor

Table 5. 12: Test case for check battery level & speak

Test ID	TC-11
Test name	Check Battery Level & Speak
Date of test	01/12/2021
Name of application	Blind Assistant
Description	In Battery module tap on screen then app will speak battery level & state
Input	Tap on the screen
Expected output	App Speak Battery Level & State
Actual output	App Speak Battery Level & State
Test Role (Actor)	Team Member
Test verified by	Team Member/Supervisor