

TEXT TONE APPLICATION

BY

PAARNIKA.R (16UCA025)

RESHMA.A (16UCA031)

SHEHATAJ BANU.B (16UCA039)

**PROJECT WORK SUBMITTED IN PARTIAL FULLFILLMENT OF THE
REQUIREMENTS FOR THE**

**DEGREE OF
BACHELOR OF COMPUTER APPLICATIONS**

DEPARTMENT OF COMPUTER SCIENCE

**AVINASHILINGAM INSTITUTE FOR HOME SCIENCE AND
HIGHER EDUCATION FOR WOMEN**

COIMBATORE - 43

APRIL - 2019

TEXT TONE APPLICATION

BY

PAARNIKA.R (16UCA025)

RESHMA.A (16UCA031)

SHEHATAJ BANU.B (16UCA039)

A Project Report Submitted to

**Avinashilingam Institute for Home Science and Higher Education for Women
Coimbatore-43**

In partial fulfillment of requirements for the award of

Bachelor's Degree in Computer Applications

APRIL – 2019

Signature of the Guide

Signature of the Head of the Department

Signature of the Internal Examiner

Signature of the External Examiner

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

We would like to express our sincere thanks to **God Almighty**, for his constant love and grace that he showered upon us.

We would like to express our deep sense of reverential gratitude and sincere thanks to **Shri. Dr. P. R. Krishnakumar, Chancellor**, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, for his support and encouragement during the course of our study.

We owe our great deal of gratitude to **Dr. Premavathy Vijayan M.Sc., M.Ed., Dip.Spl. Edn., M.Phil., Ph.D., Vice Chancellor**, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, for extending all resources that facilitated the conduct of the present study.

We express our gratitude to **Dr. S. Kowsalya, M.Sc., M.Phil., Ph.D., Registrar**, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, for providing all facilities necessary for the study.

We are also thankful to **Dr. K. Udaya Chandrika, M.Sc., M.Phil., Ph.D., Dean, School of Physical Sciences & Computational Sciences**, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, for granting the facility required.

We wish to place our deep sense of gratitude to **Dr. V. Radha, M.Sc., M.Phil., Ph.D., Professor and Head, Department of Computer Science**, for providing all the facilities required to complete the project.

We express our honorable thanks to our project coordinator **Dr. S. Anitha, M.Sc., M.Phil., Ph.D., Assistant Professor, Department of Computer Science**, for her kind valuable advice and knowledgeable suggestions which helped us to complete our project successfully.

We heartily thank our esteemed project guide **Dr. S. Visalakshi, MCA., M.Phil., Ph.D. Assistant Professor, Department of Computer Science**, for imparting the tremendous assistance and well-timed support for triumph of our project.

We would like to express our sincere gratitude to all the staff members of the Department of Computer Science, for their constant encouragement and for the opportunity to do our project in this esteemed university. Last yet importantly, we would like to thank our parents, family members, friends and all well-wishers for their kind inspiration, blessings and encouragement during the completion of the course of project.

SYNOPSIS

SYNOPSIS

The project entitled as “**TEXT TONE APPLICATION**” is developed as an **Android** application. Android mobile operating system developed by Google. Text Tone Android Application allows the user to input spoken information (English) and use desired text message. In this app voice-to-text and text-to-voice interaction between the user and the server will be processed. **Some of the existing app are SMS & Call reader, SMS talker. In existing app, there is no voice to text message and text to voice message are not implemented so far according to our review. In the proposed application,** Text Tone application will always run in the background. **If the user is busy and can't find time to read the message, instead if they said okay to hear the message, then by clicking our app will read the received message loudly.** Along with converting text to voice, this application also takes voice and work accordingly. Whenever the user responds then it will read out messages loudly according to user preference. System requirement to develop this app are, Front end Android Studio (ECLIPSE) and Back end as MYSQL.

TABLE OF CONTENTS

TABLE OF CONTENTS

S.NO	CONTENTS	PAGE NO
1.	INTRODUCTION	
	1.1 Problem Definition	1
	1.2 Overview of the project	1
2.	SYSTEM CONFIGURATION	
	2.1 Hardware Specification	2
	2.2 Software Specification	2
	2.3 About the Software	2
3.	SYSTEM STUDY AND ANALYSIS	
	3.1 Existing System	4
	3.2. proposed System	4
4.	SYSTEM DESIGN	
	4.1 Input Design	5
	4.2 Output Design	5
	4.3 Database Design	5
	4.4 Table Design	5
	4.5 Entity Relationship Diagram	6

	4.6 Data Flow Diagram	6
5.	SYSTEM DEVELOPMENT	
	5.1 Modules	7
	5.2 Module Description	7
	5.2.1 Registration	
	5.2.2 Speech to Text Module	
	5.2.3 Text to Speech Module	
6.	SYSTEM TESTING AND IMPLEMENTATION	
	6.1 Testing	8
	6.1.1 Unit Testing	
	6.1.2 Integration Testing	
	6.1.3 Validation Testing	
	6.2 System Implementation	9
7.	CONCLUTION	10
8.	SCOPE FOR FUTURE ENHANCEMENTS	11
9.	BIBLIOGRAPHY	12
10.	APPENDIX	
	10.1 ER Diagram	13
	10.2 Data Flow Diagram	14
	10.3 Table Design	15

10.4 Screen Shots	16
10.5 Sample Code	19

1.INTRODUCTION

1.1. Problem Definition:

Whenever the user is busy, it is difficult for them to read and respond to existing message. It is also time consuming as the user always have to check their mobile. In existing app, there is no conversion from voice to text message and text to voice message according to our study.

1.2. Overview of the project:

The project entitled as “TEXT TONE APPLICATION” system. The main objective of the project is, whenever the user is busy during that time, he/ she can't spend more time to read and reply message manually, instead our app helps the user to read and reply messages through voice/text message. Google server is used in the conversion of text to voice and voice to text. if he/she permits to read a message by clicking the selected message, app automatically read out the message and if the user wants to reply, instead of typing, user can spell out the message, app automatically receive those voice signal message and convert it into text.

The proposed application has three modules such as Register module, Speech to Text module and Text to Speech Module. After installing the app in both end users, first the user needs to register and then login into the application using username and password. The unauthorized user cannot send or receive messages. After successfully login into the system, the user can able to send or receive messages based on their need. In speech to text module the spoken information is transformed into text and simultaneously message is sent to receiver. In text to speech module whenever a text message is received, the corresponding message is read out loudly by touching the message and if the receiver needs to reply to the message automatically that voice message will be sent to the other end. The application is developed with Android studio eclipse as frontend and MYSQL as backend. By using or application, the user can send and receive a message in voice/text.

SYSTEM CONFIGURATION

2.SYSTEM CONFIGURATION

2.1. Hardware Specification

- Android Mobile

2.2. Software Specification

Tools Required:

- Eclipse juno version 4.2
- Android sdk: Android 5.1 platform
- Android adt plug-in

Programming Language:

- Java

Development Environment

The configuration given below is the Hardware handled for the system development

- Operating system : Windows 10
- Processor : 2.21 Ghz
- Hard disk : 470GB
- Ram : 4 GB
- Monitor : Dell -VDV2TDM

2.3. About the Software

Front End:

Android (Eclipse):

Android has a large community of developers writing applications that extend the functionality of the devices. Eclipse is an Integrated Development Environment (IDE) used in computer programming and is the most widely used Java IDE. It contains a base workspace and an extensible plug-in system for customizing the environment. Developers in Eclipse write primarily in a customized version of Java.

Java (Programming Language):

Java is a programming Language originally developed by James Gosling at Sun Microsystems (which has since merged into Oracle Corporation) and released in 1995 as a core component of Sun Microsystem's Java platform. This Language derives from C and C++ but has a simpler object model and fewer low-level facilities.

Java Virtual Machine (JVM):

A JVM provides an environment in which Java bytecode can be executed, enabling such features as automated exception handling which provides "root-cause" debugging information for every software error (exception) and independent of the source code. A JVM is distributed along with a set of standard class libraries that implement the java Application Programming Interface (API). Appropriate APIs bundle together with JVM form the Java Runtime Environment (JRE).

Android (Operating System):

Apps can be downloaded from third-party sites or through online stores such as Google Play (formerly Android Market), the application store run by Google. As of February 2012, there were more than 4,50,000 apps available for android, and the estimated number of applications downloaded from the Android Market as of December 2011 exceeded 10 billion.

Backend:

MYSQL is an Oracle-backed open source Relational Database Management System (RDBMS) based on Structured Query Language (SQL). Although it can be used in a wide range of applications, including data warehousing, e-commerce, and logging applications. The most common use for MYSQL however, is for the purpose of a web database. It has also been tested to be a "fast, stable and true multi-user, multi-threaded MySQL database server".

SYSTEM STUDY AND ANALYSIS

3. SYSTEM STUDY AND ANALYSIS

3.1. Existing System

- In existing app, there is no voice to text message and text to voice message which implemented so far according to our review.

Existing applications:

- Some of the existing app are SMS & Call reader and SMS talker.

Drawbacks

- Manually typing a text, viewing notifications and reading a message will be difficult when the user is busy.
- It is also time consuming as the user read and reply to each message received in their login.

3.2. Proposed System

The Text tone application will always run in the background. If the user is busy and can't find time to read the message, instead if they click to hear the message, then automatically our app will read the received message loudly.

Benefits

- Along with converting text to voice, this application also takes voice and work accordingly.
- Whenever the user asks, it will respond and read out only those messages whatever the user prefers.

SYSTEM DESIGN

4. SYSTEM DESIGN

4.1. Input Design

The input design is to link between the information system and the user. In general, it comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data into a form. Here there are two modules which takes the input information to proceed with the task.

1. Registration
2. Speech-text

4.2. Output Design

The design of output is the most important task of any system. During output design, developers identify the type of outputs needed, and consider the necessary output controls and prototype report layout. In our proposed system the output module is:

- Text-Speech

4.3. Database Design

‘Designing’ the database is a part of the system design. A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective of database design is to make the data access easy, inexpensive and flexible to the user. In our system the database is designed to contain user login details such as (full name, username, password, re-type password, e-mail address, address, phone number) and message information.

4.4 Table Design

The table design provides the tools for representing our application in the form of diagrams.

There are two ways to represent them

- 1.Entity Relationship Diagram (ERD)
- 2.Dataflow Diagram

4.5 Entity Relationship Diagram (ERD)

An entity-relationship diagram is a graphical depiction of organizational system elements and the association among the elements. E-R diagrams will help to define the system boundaries. The entity Relationship diagram is also called as ERD or E-RD is network model that describes the stored data layouts of the system at the high level of abstraction.

- Object type, is a rectangular box and it represents a collection or a set of objects, here they are represented in the form of rectangular boxes.
- Relationships, objects are connected to one another by relationships. A relationship represents the set of connections between the objects and it is represented by a diamond box
- Associative object type indicators, is a special notation in the ERD. This represents something that functions both as an object and a relationship.

- Super type or subtype indicators, consists of object type and one or more subcategories, connected by a relationship.

4.6 Dataflow Diagram (DFD)

The Data Flow Diagram (DFD) is one of the most important tools used by system analysts. There are four components in DFD. Process, terminators, data flow and data source are the four components of DFD. Unlike detailed flowchart, Data Flow Diagram does not supply detailed description of the modules but graphically describes a system's data and how the data interact with the system.

Circles are used to represent the process, data flows are represented using a thin line with arrow, the terminators are represented using rectangle and the data source is represented as an open-ended rectangle.

SYSTEM DEVELOPMENT

5. SYSTEM DEVELOPMENT

5.1. Modules

The following are the modules of the system:

- Register Module
- Speech to Text Module
- Text to Speech Module

5.2. Module Description

Various modules are constructed to employ the user to send and receive the messages correspondingly.

5.2.1 Register module:

In this module, the register details such as (full name, username, password, re-type password, e-mail address, address, phone number) fields must be entered to get registered into the application and login process are performed. The user can able to get into the registration or login only if the database is connected with the server.

5.2.2. Speech to Text Module

After completion of the login process, the voice is connected via google and is converted into text. Then by clicking the send button the message will be sent to the corresponding receiver.

5.2.3. Text to Speech Module

In the same way whenever the user receives a message from the sender, user can touch the message so that it is automatically read out using the google server via internet.

SYSTEM TESTING AND IMPLEMENTATION

6.SYSTEM TESTING AND IMPLEMENTATION

6.1. Testing

In this application we have performed various testing such as unit testing, integration testing, validation testing.

6.1.1. Unit Testing

Unit testing focuses on the smallest unit of software. The unit test is always white box oriented and the step can be conducted in parallel for multiple modules. According to unit testing the router system is error free. In our system the syntax and logical errors have been corrected then and there.

Test Cases:

S.No.	Module Tested	Test Case Description	Expected Result	Actual Result
1)	User Registration	Fill the required field in registration form, then click the registration button.	Displays as successfully registered.	Requests us to check our credentials.
2)	User Login	Enter the username and password to login into the application.	After logging into the account, it moves to the user page.	Login failed, invalid user name or password
3)	Speech to Text	User can be able to reply to a message in voice that will be converted into text.	Jumps back to the user page.	Server couldn't recognize the voice note
4)	Text to Speech	Without touching on the required message	The message simply remains on the page.	The message is not read out loudly.

6.1.2. Integration Testing

Integration testing is the systematic technique for construction the program structure while the same time conducting tests to uncover tests associated with interfacing. The objective is to take unit test modules and build a program structure that has been dictated by the design. Integration Testing is performed for the modules like services, correction is difficult because the isolation of causes is complicated by the vast expense of the entire program once their errors are corrected new ones appear and the process continues and endless loop.

In our app the integration testing is done to check whether the registered account integrates with the login page and additionally checks whether the speech to text/text to speech gets connected to the respective user.

6.1.3. Validation Testing

Validation can be defined in many ways. But a simple definition is validation exceeds when the software functions in a manner that can be reasonably expected by the customer. The specification contains a section called validation testing approach.

Here, in this application the validation testing is performed to check whether the entered username and password gets verified with the database speech to text and text to speech function gets converted to perform the tasks.

6.2. System Implementation

System Implementation is considered to be the most crucial stage in achieving the successful system, because if it is not properly and controlled, it will become complicated for end user. Various life cycles are available to implement a system such as requirement analysis, design phase, test and verification, finally followed by the implementation phase which results in a successful project management.

To implement our Text Tone application, user must have android mobile with internet connection. It is mandatory to install our application to end users. The data such as username,

password, converted text messages are stored in the server using MYSQL, which are highly secured in the web server.

CONCLUSION

7. CONCLUSION

The project entitled as “**Text Tone Application**” will facilitate the **Android Based** application to convert the voice message into text and vice-versa thereby sending/receiving messages. The application has been developed with Android (Eclipse) as front-end and MYSQL as backend. In existing applications, there is no voice to text messages. **Some of the existing app are, SMS & Call reader and SMS talker.** To overcome these challenges, the proposed system provides the conversion of speech into text which can also be simultaneously send as a text message to the required people. If the user permits, then message will be read out loudly and if the user like to reply the message they can give message as voice, which will be converted into text. The application developed can be used by the people during traffic jams. Hence this **Text Tone Application** is implemented successfully at the end level user and are helpful for all level of users.

SCOPE FOR FUTURE ENHANCEMENT

8. SCOPE FOR FUTURE ENHANCEMENT

Further the application can be developed by constructing various sections like “family, friends, colleagues’, it also displays the notifications, additionally we could also enter the receiver name via voice input, the send and received date and time features can also be implemented here. In future it can also be developed as a normal messenger app where we no need to login every time.

BIBLIOGRAPHY

9. BIBLIOGRAPHY

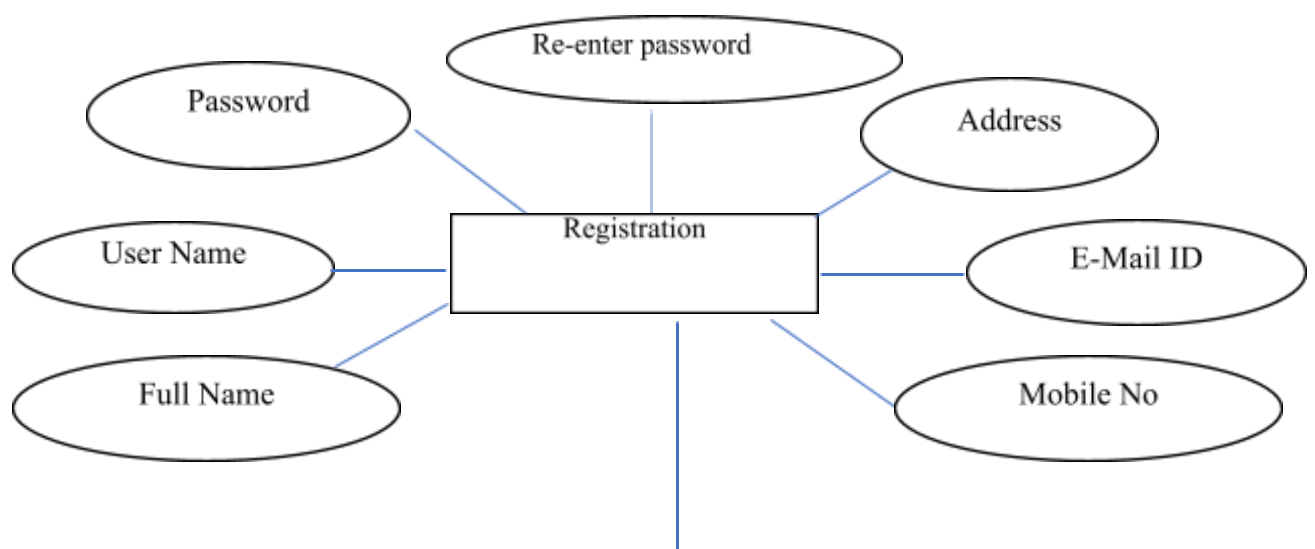
REFERENCE WEBSITES

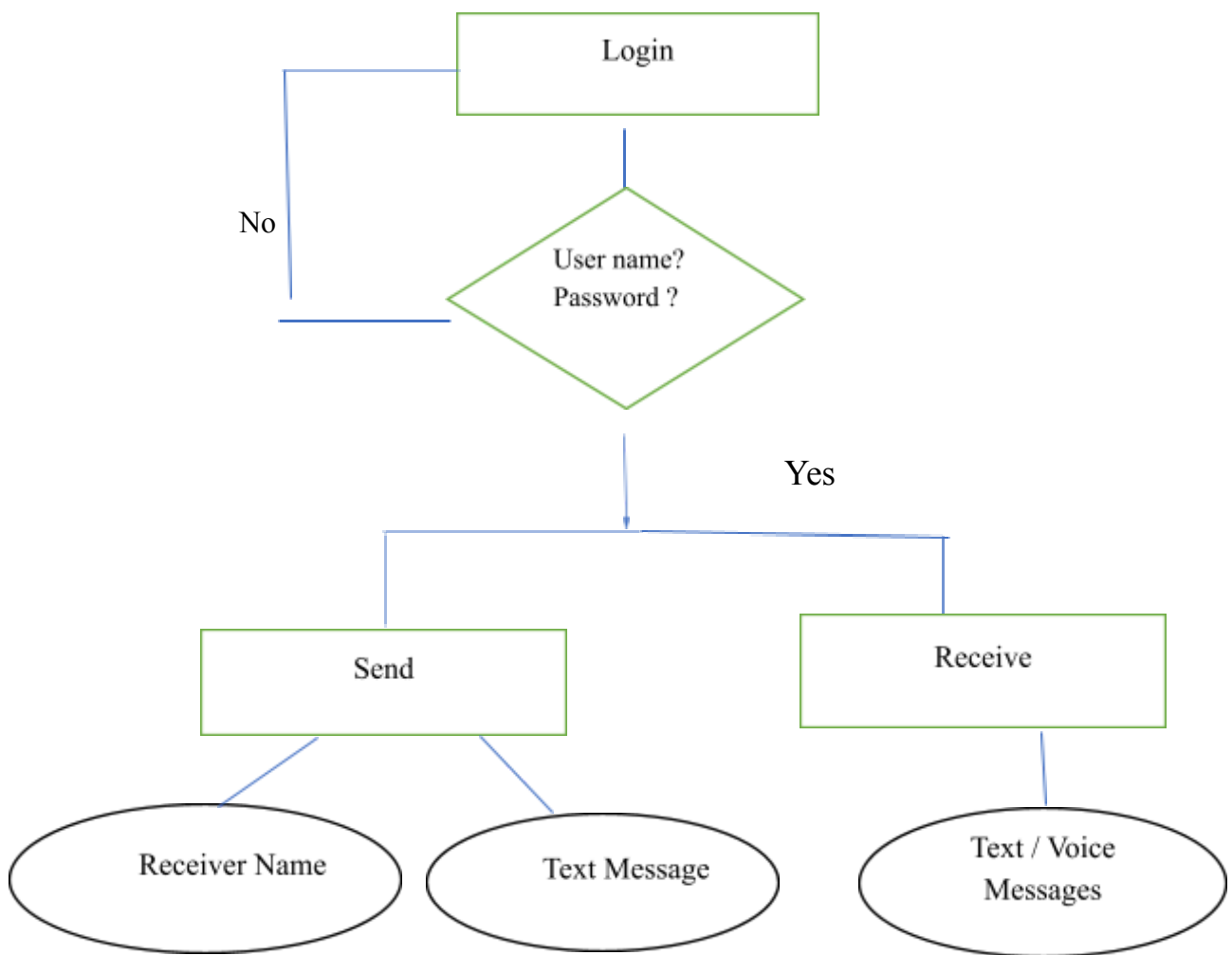
- https://www.google.com/search?q=i.+www.researchware.com&rlz=1C1JZAP_enIN824IN824&oq=i.+www.researchware.com&aqs=chrome..69i57.2324j0j7&sourceid=chrome&ie=UTF-8
- <https://www.androidhive.info/>
- https://www.google.com/search?q=i.+www.javapapers.com&rlz=1C1JZAP_enIN824IN824&oq=i.+www.javapapers.com&aqs=chrome..69i57.4789j0j4&sourceid=chrome&ie=UTF-8

REFERENCE BOOKS

1. Michel F. Mc Tear “Voice Application Development for Android”,2013.
2. Thierry Dutoit “An Introduction to Text to Speech Synthesis”,2010.

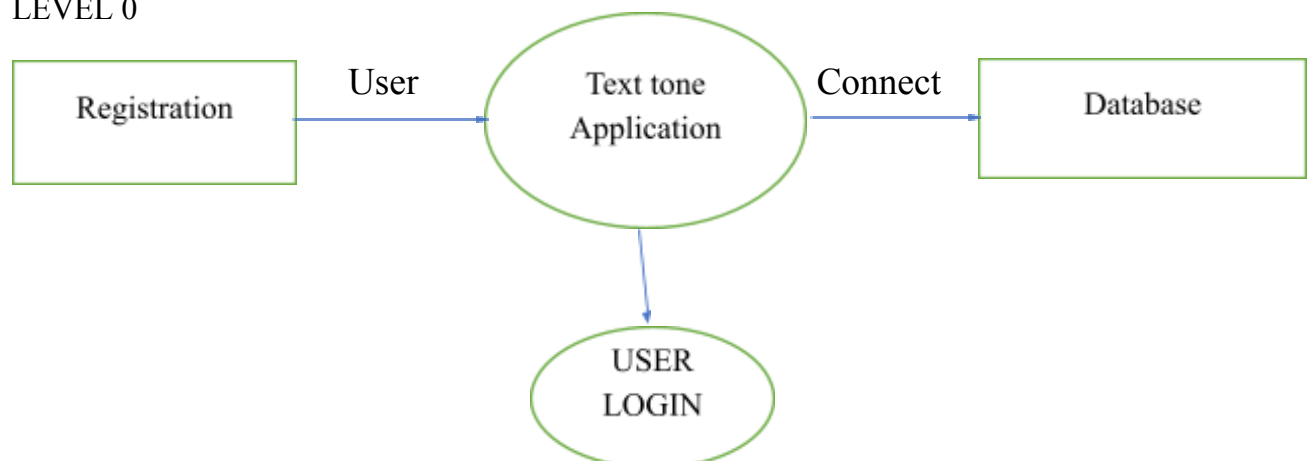
10.1 ER DIAGRAM

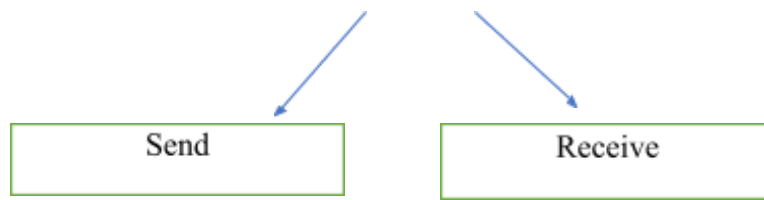




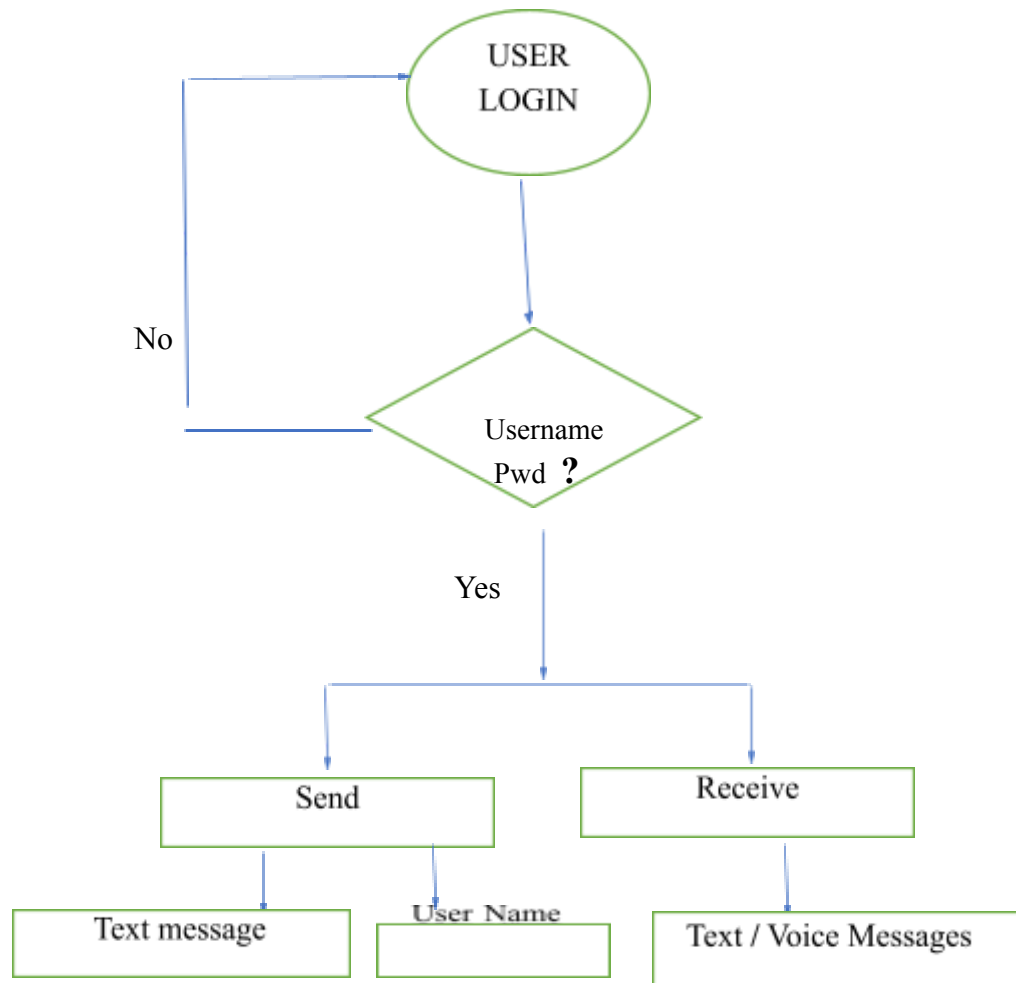
10.2 DATA FLOW DIAGRAM

LEVEL 0





Level 1



10.3 TABLE DESIGN

Table name : User Details

Primary Key : User name

Column Name	Data Type
Name	Varchar2 (50)
Username	Varchar2 (50)
Password	Varchar2 (50)
Mail id	Varchar2 (50)
Phone number	Number (10)
Address	Varchar2 (50)

Table name : Messages

Primary Key : Sender

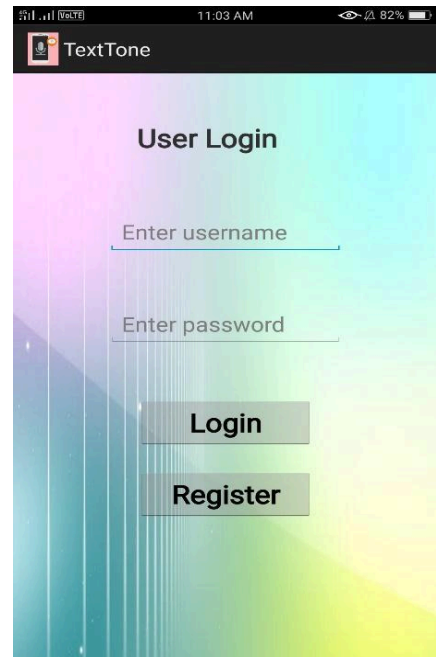
Column Name	Data Type
Sender	Varchar2 (50)
Receiver	Varchar2 (50)
Messages	Varchar2 (300)

10.4 SCREEN SHOTS

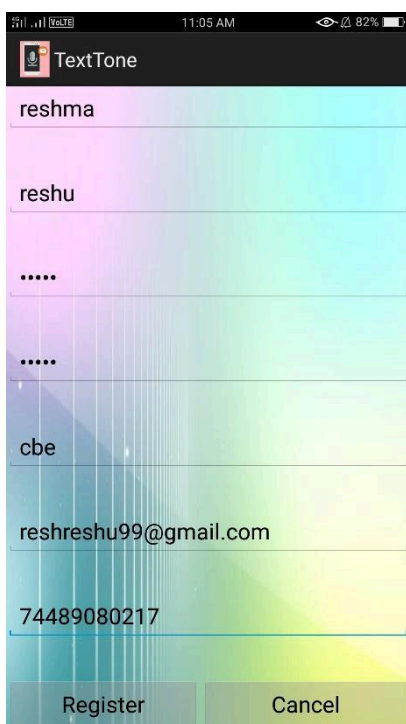
Home Page:



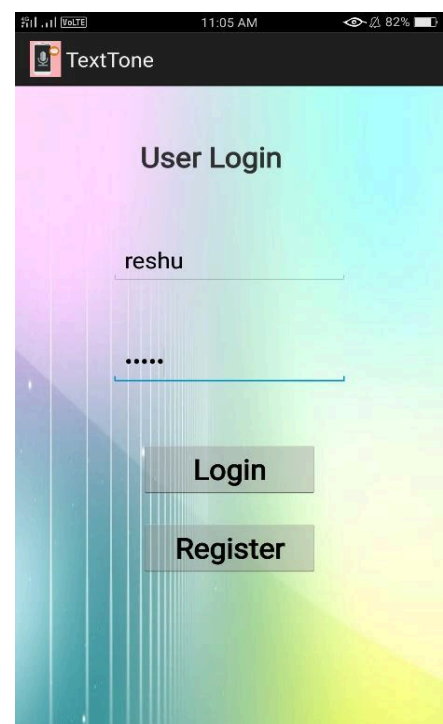
1



2

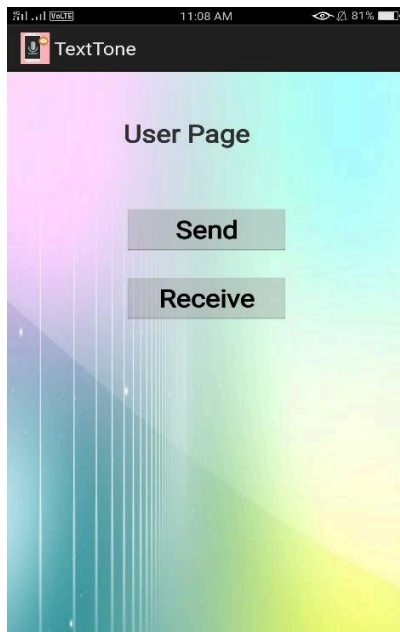


3

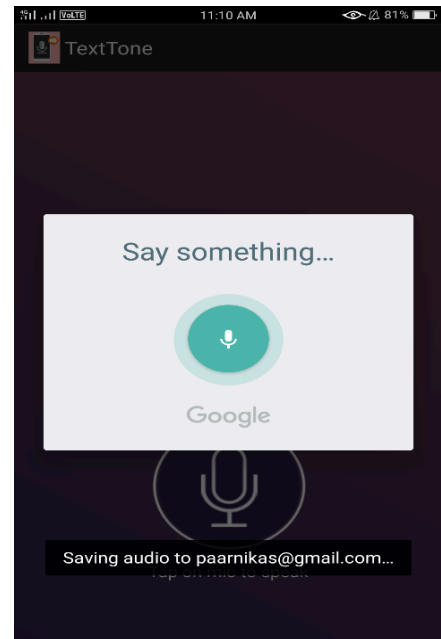


4

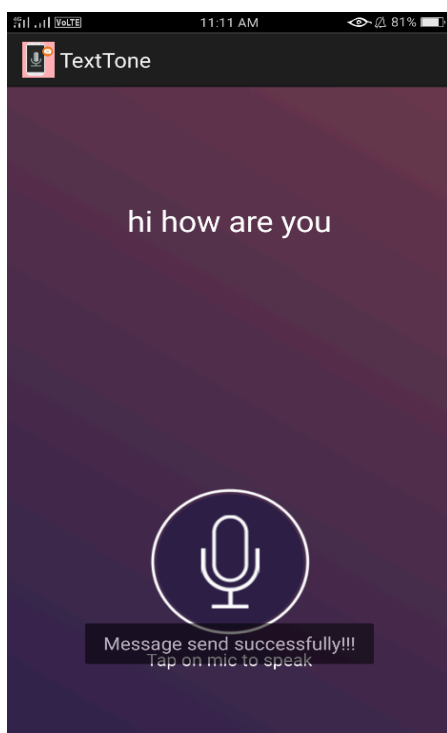
Sender Page:



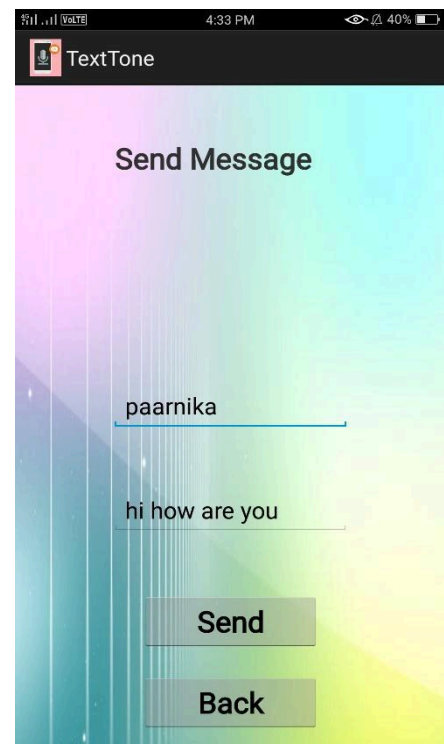
1



2

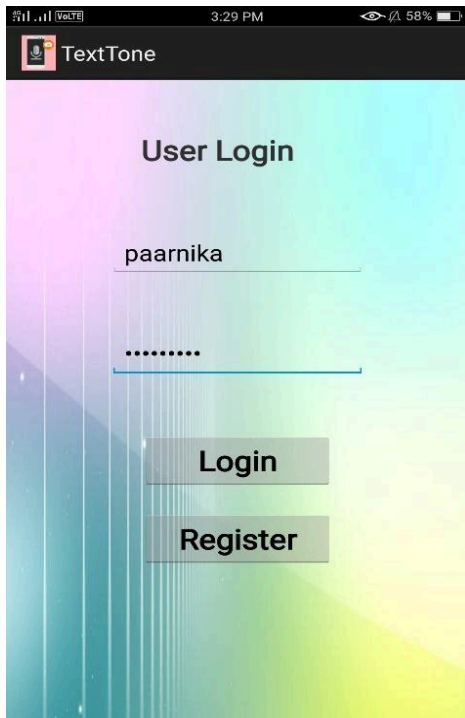


3

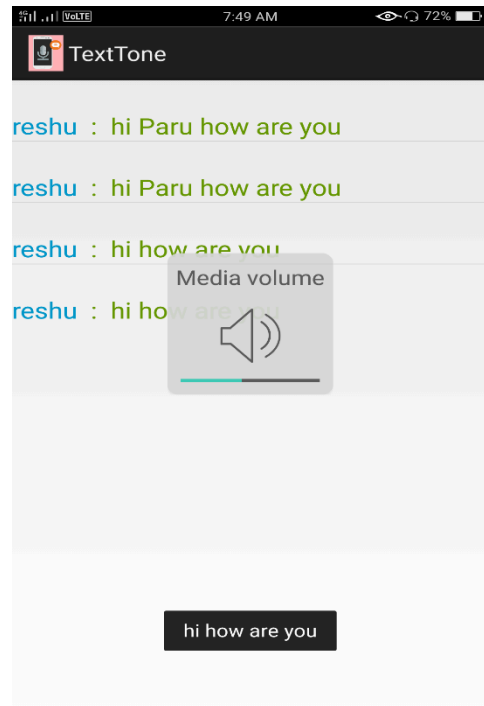


4

Receiver Page:



1



2

10.5 SAMPLE CODE

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".HomePage"
    android:background="@drawable/text">

</RelativeLayout>

<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"

    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:background="@drawable/simple">
    <LinearLayout
        android:orientation="vertical"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <TextView
            android:id="@+id/textView7"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text=""
            android:textAppearance="?android:attr/textAppearanceLarge" />

        <EditText
            android:id="@+id/registerfullname"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Full Name"
            android:ems="10" >
            <requestFocus />
        </EditText>
        <TextView
            android:id="@+id/textView2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text=""
            android:textAppearance="?android:attr/textAppearanceLarge" />

        <EditText
            android:id="@+id/registerpassword"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:hint="Password"
            android:inputType="textPassword" />
```

```

<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text=""
    android:textAppearance="?android:attr/textAppearanceLarge" />

<EditText
    android:id="@+id/registeraddress"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="Address"
    />

<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text=""
    android:textAppearance="?android:attr/textAppearanceLarge" />

<EditText
    android:id="@+id/registermobilenno"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="Mobile Number" />

<TextView
    android:id="@+id/textView6"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text=""
    android:textAppearance="?android:attr/textAppearanceLarge" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content" >
    <Button
        android:id="@+id/register_btn_reg"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="0.52"
        android:text="Register" />
    <Button
        android:id="@+id/register_btn_cancel"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="0.70"
        android:text="Cancel" />

</LinearLayout>
</LinearLayout>
</ScrollView>

```

