VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELGAVI, KARNATAKA -590 018

A Mini-Project Report

on

"SECURE NOTEBOOK"

Submitted in partial fulfillment for the *"Mobile Application Development Laboratory"* with Mini-Project (18CSL68) course of Sixth Semester of Bachelor of Engineering in Computer Science & Engineering during the academic year 2021-22.

By

Mohammed Roushan	**4MH19CS126**
Prajwal Y P	**4MH19CS071**

Under the Guidance of

Dr.Wahida Banu

Associate Professor

Department of CS&E

MIT Mysore

MAHARAJA INSTITUTE OF TECHNOLOGY MYSORE

Belawadi, S.R. Patna Taluk, Mandya Dist-571477

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Accredited By: NBA NATIONAL BOARD OF ACCREDITATION, NAAC B++

2021-22

MAHARAJA INSTITUTE OF TECHNOLOGY MYSORE
Belawadi, S.R. Patna Taluk, Mandya Dist-571477

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

Certified that the mini-project work entitled "**Secure Notebook**" is a bonafide work carried out by **Mohammed Roushan**(4MH19CS126) & **Prajwal Y P** (4MH19CS071) for the **Mobile Application Development Laboratory with Mini_Project** (18CSMP68) of Sixth Semester in Computer Science & Engineering under Visvesvaraya Technological University, Belgavi during the academic year 2021-22.

It is certified that all corrections/suggestions indicated for Internal Assignment have been incorporated in the report. The report has been approved as it satisfies the course requirements.

Signature of Guide

Dr.Wahida Banu
Associate Professor
Dept. of CS&E
MIT Mysore

Signature of the HOD

Dr. Shivamurthy R C
Professor & HOD
Dept. of CS&E
MIT Mysore

External viva

Name of the Examiners	**Signature with date**
1)	
2)	

ACKNOWLEDGEMENT

It is the time to acknowledge all those who have extended their guidance, inspiration, and their wholehearted co-operation all along our project work.

We are also grateful to **Dr. B G Naresh Kumar**, principal, MIT Mysore and **Dr. Shivamurthy R C**, HOD, CS&E, MIT Mysore for having provided us academic environment which nurtured our practical skills contributing to the success of our project.

We wish to place a deep sense of gratitude to all Teaching and Non-Teaching staffs of Computer Science and Engineering Department for whole-hearted guidance and constant support without which Endeavour would not have been possible.

Our gratitude will not be complete without thanking our parents and our friends, who have been a constant source of support and aspirations

Mohammed Roushan

Prajwal Y P

ABSTRACT

In this project, we have created a “Secure Notebook” with a purpose of providing information security is concerned with the identification of an individual development and implementation of tools, techniques, policies, standards, procedures and guidelines to ensure the confidentiality, integrity and availability of the contents. The main purpose of enabling information security is protecting information and information systems from unauthorized access, use, disclosure, disruption, modification, or destruction in order to provide integrity, confidentiality, and availability.We have created a database in order to store the information provided by the users or the clients such that retrieval process is easier for the authorized users.

CONTENTS

CHAPTER 1

INTRODUCTION

1.1 AIM OF THE PROJECT

The primary aim of this project is to provide secure note-book for people, which only they can access and manipulate. This security will help to keep people privacy intact without getting leaked. Also to help them listen to the text that they want to read, enabling them to efficiently finish the Task of Reading. Hence we aim to help customers have their reading and writing task to complete efficiently with our friendly user-interface.

1.2 OVERVIEW OF THE PROJECT

The main objective of this application is to help the user to edit text in all basic yet efficient ways by providing them the super-easy, dynamic as well as friendly user-interactive environment. User can easily make use of this application from anywhere and at any time.Also our main objective is to provide them with security feature with free of cost, so their UI will now have privacy for them.

1.3 OUTCOME OF THE PROJECT

The free application "Secure Notebook" is very friendly, it has a beautiful and simple interface. It is the best choice for a pocket notebook that is always a handy tool, from which user can save, edit or listen to what ever text they write. This is also useful as a Remainder Note App too, they can make use of this to save their password , and some other private credentials as this application provide privacy feature.

CHAPTER 2

DESIGN AND IMPLEMENTATION

2.1 ALGORITHM

Step1: Start

Step2: if user is new to to this app, perform registration process by keeping the scroll to the left side

Step3: If user is already registered or registered now, perform logging process by keeping the scroll to right side

Step4: In registration process aka Sign up process is launched when switch is on the left.

Step5: In this process, fill all the necessary blocks and then click SignUp button, be sure to remember your password before you continue.

Step6: Whereas in Signin process, the switch will be on the right. Similarly fill the blocks appropriately and click SignIn button.

Step7: If all the information provided in SignIn process is valid, it will display successful message and redirect to text editor with text intact to their credentials.

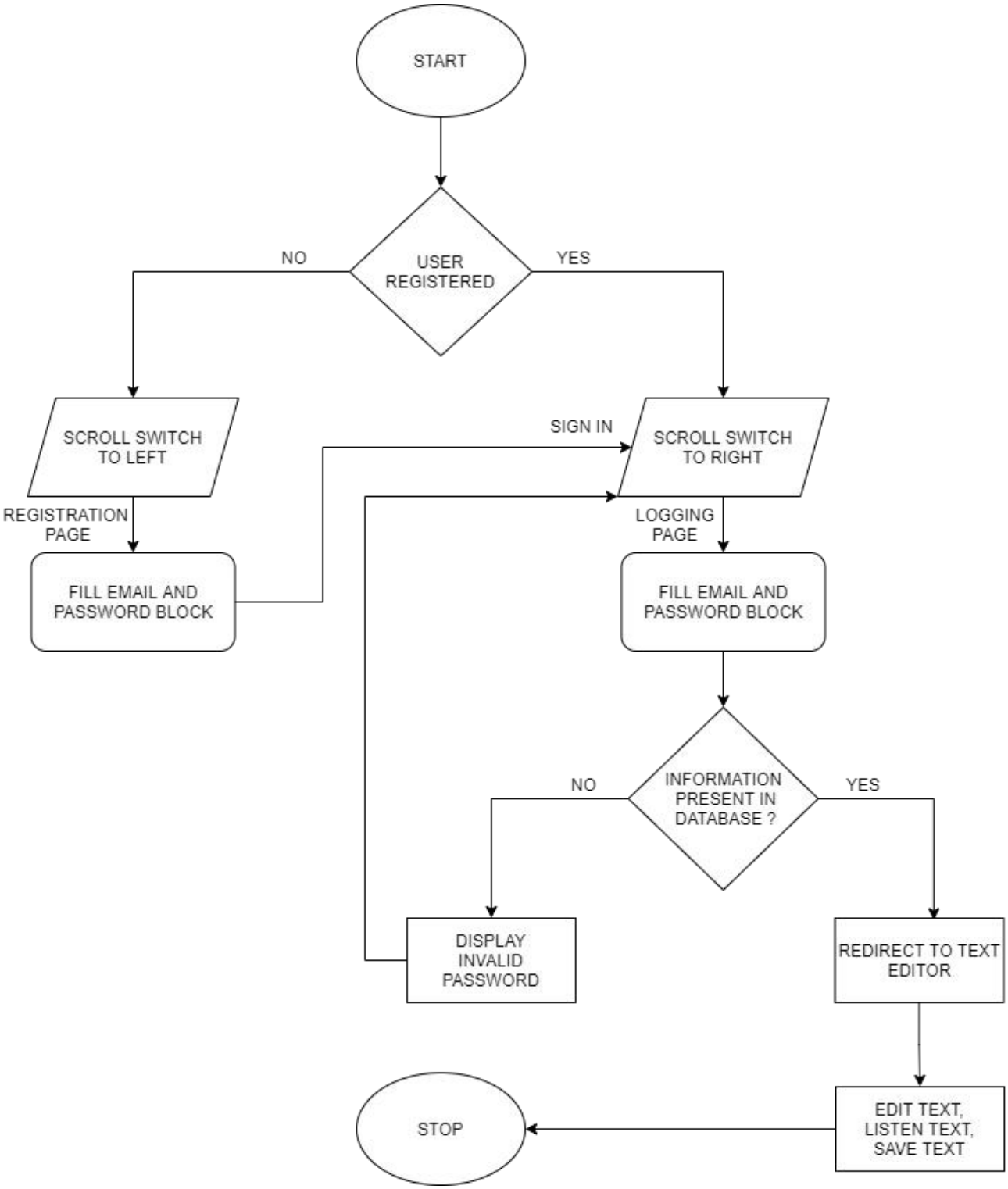
Step8: User is allowed to type the text, edit the text with Bold, Italic, Underline and also undo the format, also listen to the text that the user have typed.

Step9: User can save the text by pressing the Save button, also reset the text field with the help of Reset button.

Step10: User can then logout and login to their respective notebook aka text editor whenever and wherever it is required with full privacy.

Step11: Stop

2.2 Flow Chart



2.3 Proposed System

This project has been developed in the IDE called Android Studio. It consists of two main activity with xml files as well as java files such as MainActivity, MainActivity2., and an addition Database handler Java Class.

Both the activities have been derived the instances of the database class, in which Main activity with accept the help of DBconnection aka Database class to insert the credentials of the user who need to be registered. Also it uses the database with this class help to verify an external user credential.

If all the information in the blocks provided by user in MainActivity activity, are true and verified properly, I will display successful message and redirect to text editor present in MainActivity2 activity.

Where as the MainActivity2 activity use the Database to retrieve the previously type string by that particular user with the help of his or her queries.

In the MainActivity2 activity, the text editor will allow the user to use its special features such as enhancing the selected text only to various formats such as Bold, Italic, Underline etc… Also it will enable the user the feature to listen to the text what they have typed in the text editor with the help of Text To Speech library associated with Talk button.

In the same activity, once the user click the save button all the text which has been typed in the text editor will be transported to the database called User, specifically to the users table with the field name called Strings.

Now, User can logout anytime and end the session. Again after he/she login to their respective accounts with proper credentials, that user can be able to visualize the same text which he/she has bees saved in th text editor in their previous sessions.

By this, we can conclude that our project application enables us the most elegant features such as privacy, editing tools and text to speech tools and efficient storage too.

2.4 Source Code

(activity_main.xml)

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity2">

    <Button
        android:id="@+id/btnBold"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginEnd="306dp"
        android:layout_marginBottom="620dp"
        android:background="@drawable/btntext"
        android:onClick="bold"
        android:text="BOLD" />

    <TextView
        android:id="@+id/textView3"
        android:layout_width="192dp"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginEnd="110dp"
        android:layout_marginBottom="620dp"
        android:text="T E X T\nN O T E B O O K"
        android:textAlignment="center"
        android:textColor="#7B1FA2"
        android:textSize="20sp" />

    <Button
        android:id="@+id/btnItalic"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginEnd="18dp"
        android:layout_marginBottom="620dp"
        android:background="@drawable/btntext"
        android:onClick="italic"
        android:text="ITALIC" />

    <Button
```

```
    android:id="@+id/btnNoformat"
    android:onClick="noformat"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
    android:layout_marginEnd="15dp"
    android:layout_marginBottom="554dp"
    android:background="@drawable/btntext"
    android:text="NO FORMAT"
    android:textSize="15sp" />

<Button
    android:id="@+id/btnTalk"
    android:onClick="talk"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
    android:layout_marginEnd="164dp"
    android:layout_marginBottom="553dp"
    android:background="@drawable/btntext"
    android:text="TALK" />

<Button
    android:id="@+id/btnunderline"
    android:onClick="underline"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
    android:layout_marginEnd="303dp"
    android:layout_marginBottom="554dp"
    android:background="@drawable/btntext"
    android:text="UNDERLINE"
    android:textSize="15sp" />

<EditText
    android:id="@+id/edtTxt"
    android:layout_width="match_parent"
    android:layout_height="451dp"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
    android:layout_marginEnd="0dp"
    android:layout_marginBottom="90dp"
    android:background="@drawable/rounded_transparent"
    android:gravity="start|top"
    android:hint="Enter text here . . ."
    android:inputType="textMultiLine"
    android:paddingStart="10sp"
```

```
        android:textSize="25sp" />


    <Button
        android:id="@+id/btnSave"
        android:layout_width="wrap_content"
        android:onClick="save"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginEnd="301dp"
        android:layout_marginBottom="20dp"
        android:background="@drawable/btntext"
        android:text="SAVE" />

    <Button
        android:id="@+id/btnDownload"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginEnd="164dp"
        android:layout_marginBottom="19dp"
        android:background="@drawable/btntext"
        android:onClick="download"
        android:enabled="false"
        android:text="DOWNLOAD" />

    <Button
        android:id="@+id/btnReset"
        android:onClick="reset"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginEnd="17dp"
        android:layout_marginBottom="24dp"
        android:background="@drawable/btntext"
        android:text="RESET" />


</RelativeLayout>
```

(ii) (**MainActivity.java**)

```
package com.example.project;

import android.content.Context;
import android.content.Intent;
import android.database.Cursor;
```

```
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Switch;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

   EditText sname,lname,spass,lpass;
   Button sbtn,lbtn;
   Switch swtch;
   DBConnection db;

   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      sname=findViewById(R.id.edtSignName);
      spass=findViewById(R.id.edtSignPass);
      lname=findViewById(R.id.edtLogName);
      lpass=findViewById(R.id.edtLogPass);
      sbtn=findViewById(R.id.btnSign);
      lbtn=findViewById(R.id.btnLog);
      swtch = findViewById(R.id.switch1);
      db = new DBConnection(this);

   }

   public void sign(View view) {
      String name= sname.getText().toString();
      String pass=spass.getText().toString();

      boolean result = db.insertNew(name,pass,"");
      if(result==true){
         Toast.makeText(this, "Successfully Signed Up",
Toast.LENGTH_SHORT).show();
         sname.setText("");
         spass.setText("");
      }
      else
         Toast.makeText(this, "User Email Already Registered!!",
Toast.LENGTH_SHORT).show();
   }

   public void log(View view) {
      String name= lname.getText().toString();
      String pass=lpass.getText().toString();
```

```
        String upass="";



       try {
          Cursor cursor = db.verifyData(name);
          cursor.moveToFirst();

          do{
             upass = upass +
String.valueOf(cursor.getString(cursor.getColumnIndex("PASSWORD")));
          }while (cursor.moveToNext());

       }catch (Exception e){
          Toast.makeText(this, "Email Not Exists !!", Toast.LENGTH_SHORT).show();
          return;
       }

       if(upass.equals(pass)){
          Toast.makeText(this, "Login Successfull", Toast.LENGTH_SHORT).show();
          Intent move = new Intent(MainActivity.this,MainActivity2.class);
          move.putExtra("email",name);
          startActivity(move);
       }
       else {
          Toast.makeText(this, "Invalid Password", Toast.LENGTH_SHORT).show();
       }
    }

    public void swtch(View view) {
       if (swtch.isChecked()){
          lname.setVisibility(View.VISIBLE);
          lpass.setVisibility(View.VISIBLE);
          lbtn.setVisibility(View.VISIBLE);
          sname.setVisibility(View.INVISIBLE);
          spass.setVisibility(View.INVISIBLE);
          sbtn.setVisibility(View.INVISIBLE);
          sname.setText("");
          spass.setText("");
       }
       else{
          sname.setVisibility(View.VISIBLE);
          spass.setVisibility(View.VISIBLE);
          sbtn.setVisibility(View.VISIBLE);
          lname.setVisibility(View.INVISIBLE);
          lpass.setVisibility(View.INVISIBLE);
          lbtn.setVisibility(View.INVISIBLE);
          lname.setText("");
          lpass.setText("");
       }
```

```
    }
}
```

(iii) (**MainActivity2.java**)

```
package com.example.project;

import android.database.Cursor;
import android.graphics.Typeface;
import android.os.Bundle;
import android.speech.tts.TextToSpeech;
import android.text.Spannable;
import android.text.SpannableStringBuilder;
import android.text.style.StyleSpan;
import android.text.style.UnderlineSpan;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import java.util.Locale;

public class MainActivity2 extends AppCompatActivity {

    EditText edt;
    String initial="";
    TextToSpeech tts;
    DBConnection db;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);
        edt=findViewById(R.id.edtTxt);
        tts = new TextToSpeech(getApplicationContext(), new
TextToSpeech.OnInitListener() {
            @Override
            public void onInit(int i) {
                if(i==tts.SUCCESS){
                    tts.setLanguage(Locale.ENGLISH);
                }
            }
        });
        String email = getIntent().getStringExtra("email");
        db = new DBConnection(this);


        try {
            Cursor cursor = db.copystring(email);
```

```
            cursor.moveToFirst();

            do{
                initial = initial +
String.valueOf(cursor.getString(cursor.getColumnIndex("STRING")));
            }while (cursor.moveToNext());
        }catch (Exception e){
        }

        edt.setText(initial);
    }

    public void bold(View view) {
        Spannable spannable =new SpannableStringBuilder(edt.getText());
        spannable.setSpan(new StyleSpan(Typeface.BOLD),
                edt.getSelectionStart(),
                edt.getSelectionEnd(),
                0);
        edt.setText(spannable);
        return;
    }

    public void underline(View view) {
        Spannable spannable =new SpannableStringBuilder(edt.getText());
        spannable.setSpan(new UnderlineSpan(),
                edt.getSelectionStart(),
                edt.getSelectionEnd(),
                0);
        edt.setText(spannable);
        return;
    }

    public void italic(View view) {
        Spannable spannable =new SpannableStringBuilder(edt.getText());
        spannable.setSpan(new StyleSpan(Typeface.ITALIC),
                edt.getSelectionStart(),
                edt.getSelectionEnd(),
                0);
        edt.setText(spannable);
        return;
    }

    public void talk(View view) {
        String txt=edt.getText().toString();
        tts.speak(txt,TextToSpeech.QUEUE_FLUSH,null);

    }

    public void noformat(View view) {
        edt.setText(edt.getText().toString());
```

```
        return;
    }

    public void save(View view) {
        String email = getIntent().getStringExtra("email");
        String string = edt.getText().toString();
        boolean result = db.insertString(email,string);
        if(result==true){
            Toast.makeText(this, "Successfully Saved", Toast.LENGTH_SHORT).show();
        }
        else
            Toast.makeText(this, "Not Saved", Toast.LENGTH_SHORT).show();
    }

    public void download(View view) {
    }

    public void reset(View view) {
        edt.setText("");
    }
}
```

(iv) **(DB Connections.java)**

```
package com.example.project;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class DBConnection extends SQLiteOpenHelper {
    public DBConnection(Context context) {
        super(context, "UserDB", null, 1);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL("CREATE TABLE USERS(EMAIL TEXT PRIMARY
KEY,PASSWORD TEXT,STRING TEXT);");
    }

    @Override
    public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {

    }

    public  boolean insertNew(String email,String password,String string){
        SQLiteDatabase db = this.getWritableDatabase();
```

```
        ContentValues contentValues = new ContentValues();
        contentValues.put("EMAIL",email);
        contentValues.put("PASSWORD",password);
        contentValues.put("STRING",string);
        long result=db.insert("USERS",null,contentValues);

        if(result==-1)
            return  false;
        else
            return true;
    }

    public Cursor verifyData(String email){
        SQLiteDatabase db = this.getReadableDatabase();
        Cursor cursor = db.rawQuery("SELECT * FROM USERS WHERE EMAIL =
'"+email+"'",null);
        return  cursor;
    }

    public boolean insertString(String email,String string){
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put("STRING",string);

        long result = db.update("USERS",contentValues,"EMAIL = ?",new
String[]{email});

        if(result==-1)
            return  false;
        else
            return true;
    }

    public  Cursor copystring(String email){
        SQLiteDatabase db = this.getReadableDatabase();
        Cursor cursor = db.rawQuery("SELECT * FROM USERS WHERE EMAIL =
'"+email+"'",null);
        return  cursor;
    }

}
```

(v) **(main_activity2.xml)**

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
```

```
tools:context=".MainActivity2">

<Button
    android:id="@+id/btnBold"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
    android:layout_marginEnd="306dp"
    android:layout_marginBottom="620dp"
    android:background="@drawable/btntext"
    android:onClick="bold"
    android:text="BOLD" />

<TextView
    android:id="@+id/textView3"
    android:layout_width="192dp"
    android:layout_height="wrap_content"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
    android:layout_marginEnd="110dp"
    android:layout_marginBottom="620dp"
    android:text="T E X T\nN O T E B O O K"
    android:textAlignment="center"
    android:textColor="#7B1FA2"
    android:textSize="20sp" />

<Button
    android:id="@+id/btnItalic"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
    android:layout_marginEnd="18dp"
    android:layout_marginBottom="620dp"
    android:background="@drawable/btntext"
    android:onClick="italic"
    android:text="ITALIC" />

<Button
    android:id="@+id/btnNoformat"
    android:onClick="noformat"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
    android:layout_marginEnd="15dp"
    android:layout_marginBottom="554dp"
    android:background="@drawable/btntext"
    android:text="NO FORMAT"
```

```
    android:textSize="15sp" />

<Button
    android:id="@+id/btnTalk"
    android:onClick="talk"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
    android:layout_marginEnd="164dp"
    android:layout_marginBottom="553dp"
    android:background="@drawable/btntext"
    android:text="TALK" />

<Button
    android:id="@+id/btnunderline"
    android:onClick="underline"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
    android:layout_marginEnd="303dp"
    android:layout_marginBottom="554dp"
    android:background="@drawable/btntext"
    android:text="UNDERLINE"
    android:textSize="15sp" />

<EditText
    android:id="@+id/edtTxt"
    android:layout_width="match_parent"
    android:layout_height="451dp"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
    android:layout_marginEnd="0dp"
    android:layout_marginBottom="90dp"
    android:background="@drawable/rounded_transparent"
    android:gravity="start|top"
    android:hint="Enter text here . . ."
    android:inputType="textMultiLine"
    android:paddingStart="10sp"
    android:textSize="25sp" />


<Button
    android:id="@+id/btnSave"
    android:layout_width="wrap_content"
    android:onClick="save"
    android:layout_height="wrap_content"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
```

```
        android:layout_marginEnd="301dp"
        android:layout_marginBottom="20dp"
        android:background="@drawable/btntext"
        android:text="SAVE" />

    <Button
        android:id="@+id/btnDownload"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginEnd="164dp"
        android:layout_marginBottom="19dp"
        android:background="@drawable/btntext"
        android:onClick="download"
        android:enabled="false"
        android:text="DOWNLOAD" />

    <Button
        android:id="@+id/btnReset"
        android:onClick="reset"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginEnd="17dp"
        android:layout_marginBottom="24dp"
        android:background="@drawable/btntext"
        android:text="RESET" />


</RelativeLayout>
```

CHAPTER 3

RESULT ANALYSIS

3.1 SNAPSHOTS

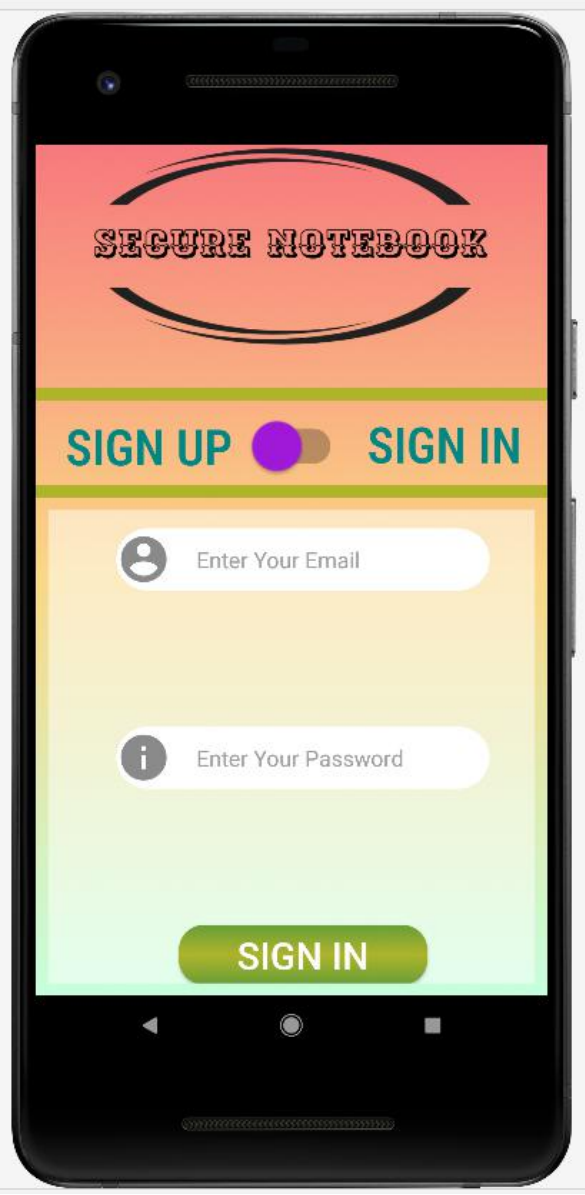


Fig. 3.1.1

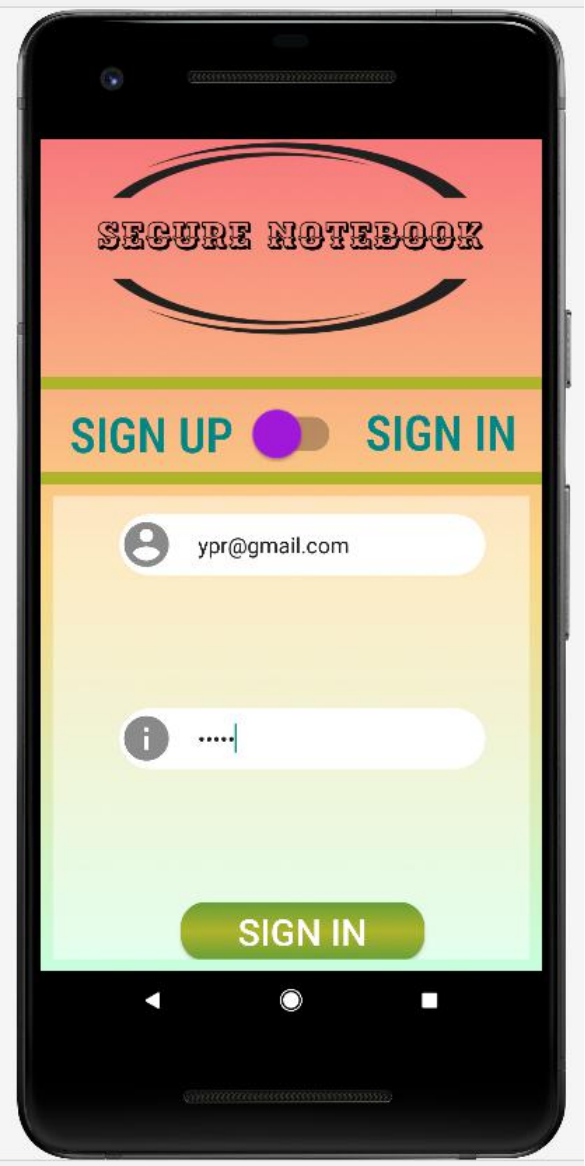


Fig. 3.1.2

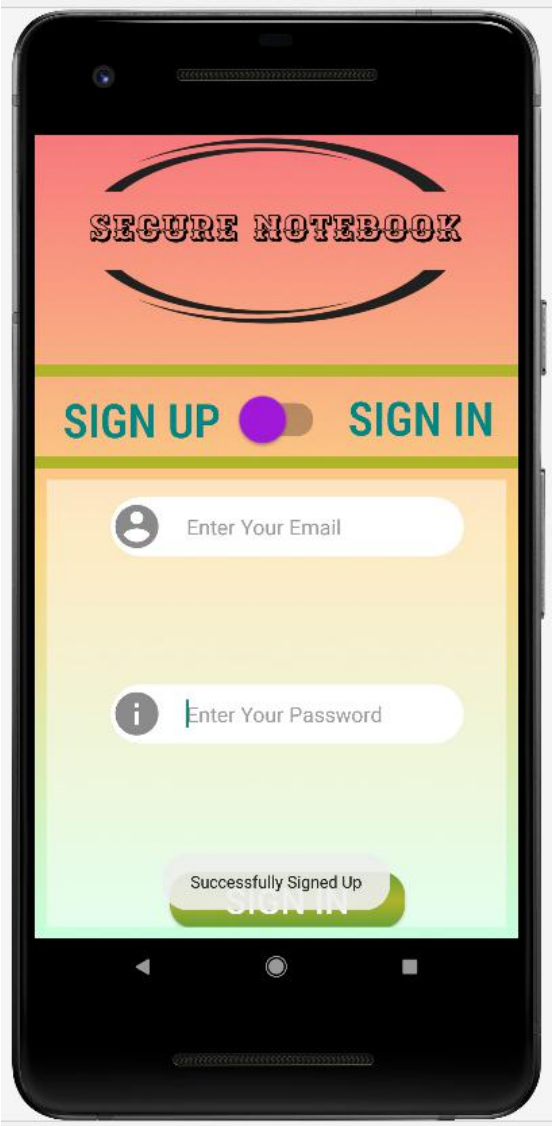


Fig. 3.1.3

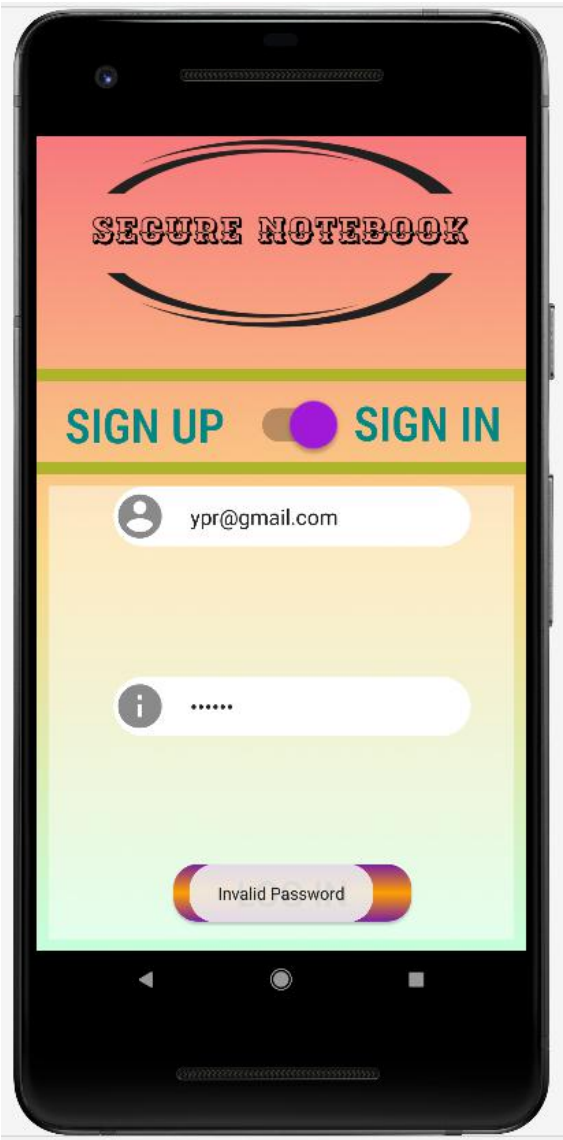


Fig. 3.1.4

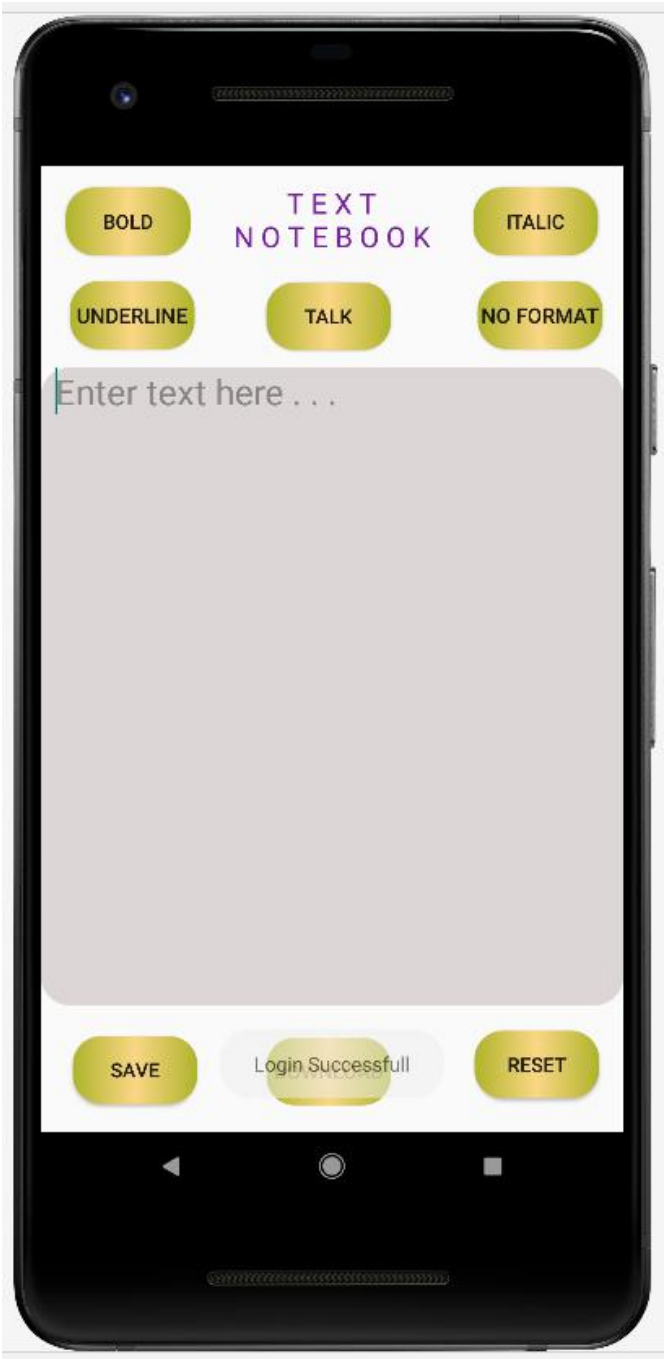


Fig. 3.1.5

Fig. 3.1.6

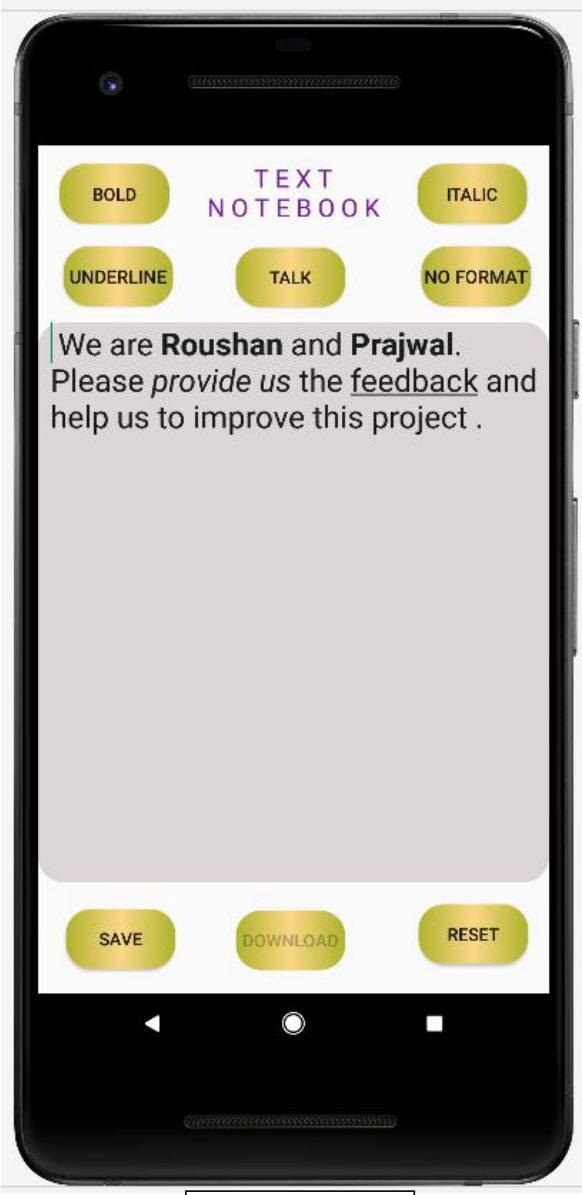


Fig. 3.1.7

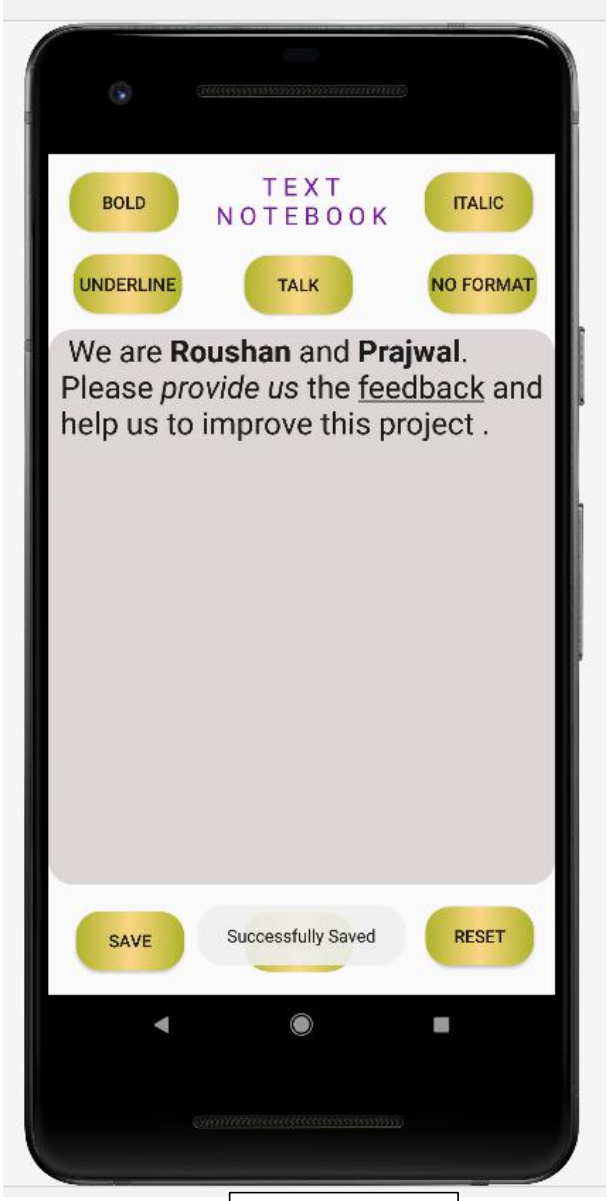


Fig. 3.1.8

3.2 Results

As per above images depicts we have successfully completed our Secire Notebook application. This application start with Fig 3.1.1 which is a string page of our app.

In Fig. 3.1.2 we can observe that the user is filling the block details. In fig 3.1.3 We can observer that the application display Successfully Signed in when user click on the signUp button.

when user scroll switch to right in fig 3.1.4 we can observe signin page. Also we can see invalid password display as user entered not valid credentials. Where as in fig 3.1.5 we can see successful message as user enter correct credentials and has been redirected to text editor page.

In fig3.1.6 we can see user has type the text in the text editor, in fig 3.1.7 user has edited the text with the help of the editing tools, in fig 3.1.8 user the finally clicked the save button after typing and we also can observe the successfully saved Message that has been pop up.

Hence we here that the actual proof that that our application is in good proper condition with special features

CHAPTER 4

CONCLUSION AND FUTURE WORK

4.1 Conclusion

We conclude that our Android Application Software “Secure Notebook” is actually secure and has tempting features for users, and is also of free of cost and no additional service charges for database usage.

We hope that this software will be of many other usages too for users in their futures, as it is flexible.

4.2 Future Enhancement

We are also planning on extending the application features by adding download options and many others in future via update, if possible in Google Android App Store.

\

CHAPTER 5

REFERENCES

- https://www.youtube.com/watch?v=iRfw3OPVJ0Q&t=3264s
- https://www.youtube.com/watch?v=PA4A9IesyCg&t=425s
- https://www.youtube.com/watch?v=jU3cen28vBU
- https://www.youtube.com/watch?v=FUGYcKVh0iU&t=612s
- https://www.youtube.com/watch?v=FUGYcKVh0iU&t=612s
- https://stackoverflow.com/questions/40508303/the-activity-must-be-exported-or-contain-an-intent-filter
- https://developer.android.com/training/basics/firstapp/starting-activity