**ANDROID-OVERVIEW**

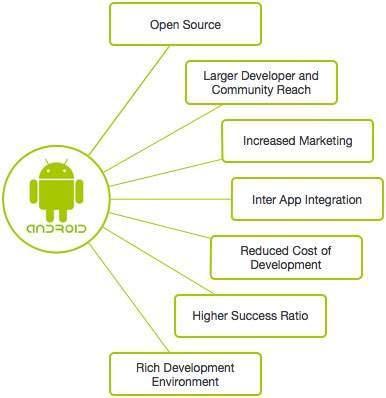
Android is an open source and Linux-based Operating System for mobile devices such as smartphones and tablet computers. Android was developed by the Open Handset Alliance, led by Google, and other companies.

Android offers a unified approach to application development for mobile devices which means developers need only develop for Android, and their applications should be able to run on different devices powered by Android.

The first beta version of the Android Software Development Kit (SDK) was released by Google in 2007 where as the first commercial version, Android 1.0, was released in September 2008.

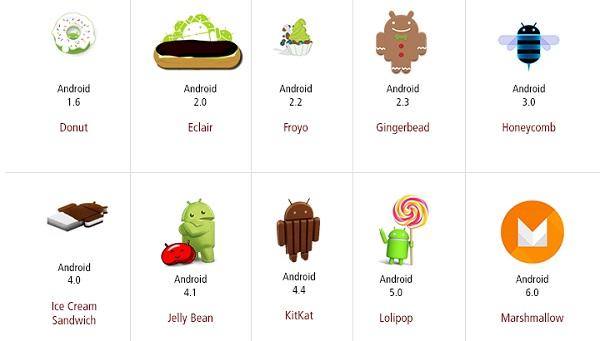
On June 27, 2012, at the Google I/O conference, Google announced the next Android version, 4.1 Jelly Bean. Jelly Bean is an incremental update, with the primary aim of improving the user interface, both in terms of functionality and performance.

The source code for Android is available under free and open source software licenses. Google publishes most of the code under the Apache License version 2.0 and the rest, Linux kernel changes, under the GNU General Public License version 2.



## History of Android

The code names of android ranges from A to N currently, such as Aestro, Blender, Cupcake, Donut, Eclair, Froyo, Gingerbread, Honeycomb, Ice Cream Sandwitch, Jelly Bean, KitKat, Lollipop and Marshmallow. Let's understand the android history in a sequence.



**Android Architecture**

Android is an operating system and is a stack of software components which is divided into five sections and four main layers that is

* Linux kernel
* Libraries
* Android runtime

### **Linux kernel:**

The android uses the powerful Linux kernel and it supports wide range of hardware drivers. The kernel is the heart of the operating system that manages input and output requests from software..

### **Libraries:**

. These libraries are used to play and record audio and video. The SQLite is a data base which is useful for storage and sharing of application data. The SSL libraries are responsible for internet security etc.

### **Android Runtime:**

The android runtime provides a key component called Dalvik Virtual Machine which is a kind of java virtual machine. It is specially designed and optimized for android.

**Android Versions**

API Level is an integer value that uniquely identifies the framework API revision offered by a version of the Android platform.

|  |  |  |  |
| --- | --- | --- | --- |
| **Platform Version** | **API Level** | **VERSION\_CODE** |  |
| Android 6.0 | 23 | MARSHMALLOW |  |
| Android 5.1 | 22 | LOLLIPOP\_MR1 |  |
| Android 5.0 | 21 | LOLLIPOP |  |
| Android 4.4W | 20 | KITKAT\_WATCH |  |
| Android 4.4 | 19 | KITKAT |  |
| Android 4.3 | 18 | JELLY\_BEAN\_MR2 |  |
| Android 4.2, 4.2.2 | 17 | JELLY\_BEAN\_MR1 |  |
| Android 4.1, 4.1.1 | 16 | JELLY\_BEAN |  |
| Android 4.0.3, 4.0.4 | 15 | ICE\_CREAM\_SANDWICH\_MR1 |  |
| Android 4.0, 4.0.1, 4.0.2 | 14 | ICE\_CREAM\_SANDWICH |  |
| Android 3.2 | 13 | HONEYCOMB\_MR2 |  |
| Android 3.1.x | 12 | HONEYCOMB\_MR1 |  |
| Android 3.0.x | 11 | HONEYCOMB |  |
| Android 2.3.4  Android 2.3.3 | 10 | GINGERBREAD\_MR1 |  |
| Android 2.3.2  Android 2.3.1  Android 2.3 | 9 | GINGERBREAD |  |
| Android 2.2.x | 8 | FROYO |  |
| Android 2.1.x | 7 | ECLAIR\_MR1 |  |
| Android 2.0.1 | 6 | ECLAIR\_0\_1 |  |
| Android 2.0 | 5 | ÉCLAIR |  |
| Android 1.6 | 4 | DONUT |  |
| Android 1.5 | 3 | CUPCAKE |  |
| Android 1.1 | 2 | BASE\_1\_1 |  |
| Android 1.0 | 1 | BASE |  |

## Android Applications

Android applications are usually developed in the Java language using the Android Software Development Kit.

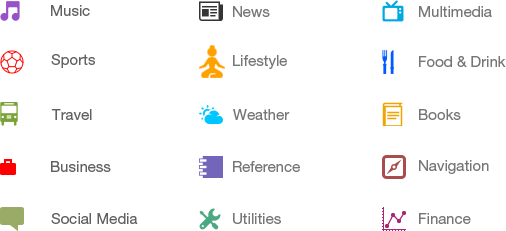
Once developed, Android applications can be packaged easily and sold out either through a store such as GooglePlay, SlideME, Opera Mobile Store, Mobango, F-droid and the Amazon Appstore.

Android powers hundreds of millions of mobile devices in more than 190 countries around the world. It's the largest installed base of any mobile platform and growing fast. Every day more than 1 million new Android devices are activated worldwide.

This tutorial has been written with an aim to teach you how to develop and package Android application. We will start from environment setup for Android application programming and then drill down to look into various aspects of Android applications.

## Categories of Android applications

There are many android applications in the market. The top categories are −



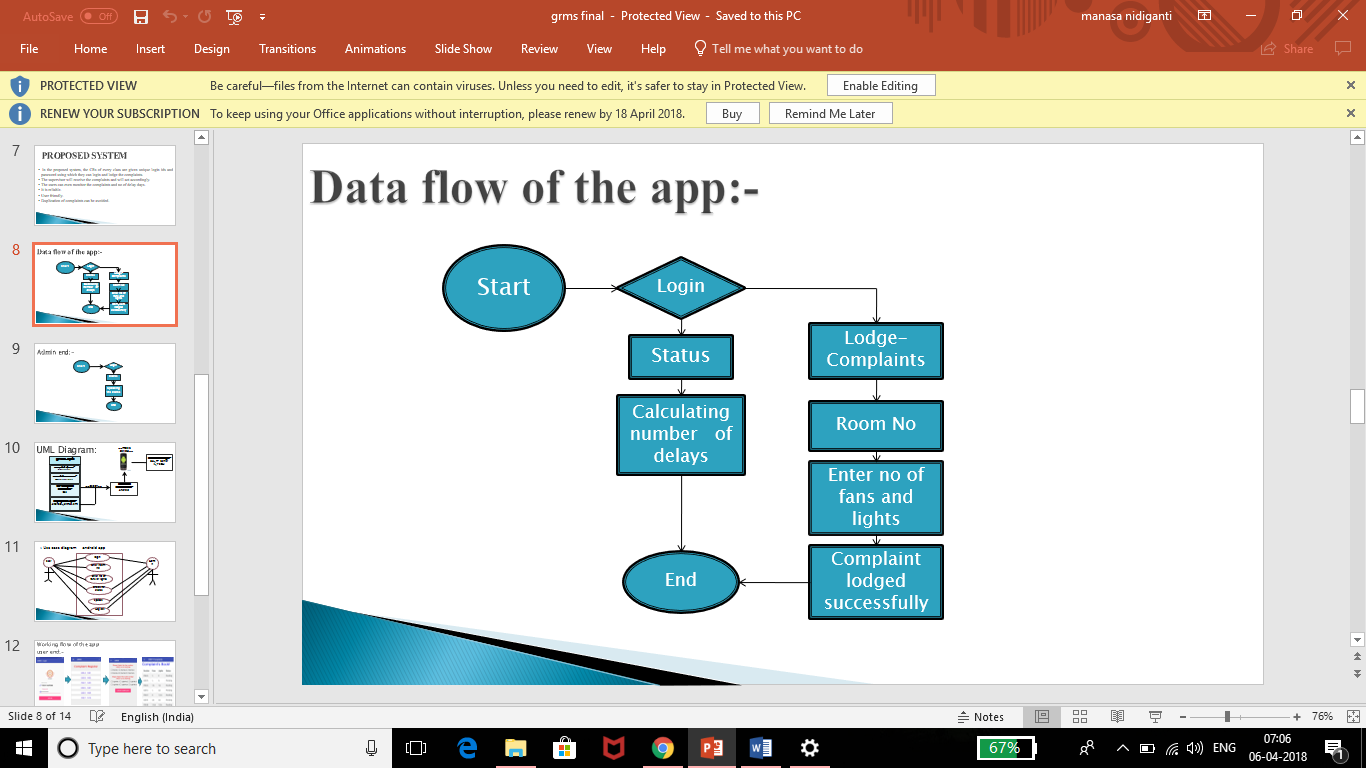
**Implementation**

Our app has been implemented to accept complaints against damaged fans and lights and to show the status of the registered complaint. The app has two sides, the user end and the admin end. The user end is for the class CRs which allows them to lodge complaints, view the status of the complaint and see the complaint’s history. Whereas, in the admin end, the admin can view the lodged complaints and can update the status of the complaint.

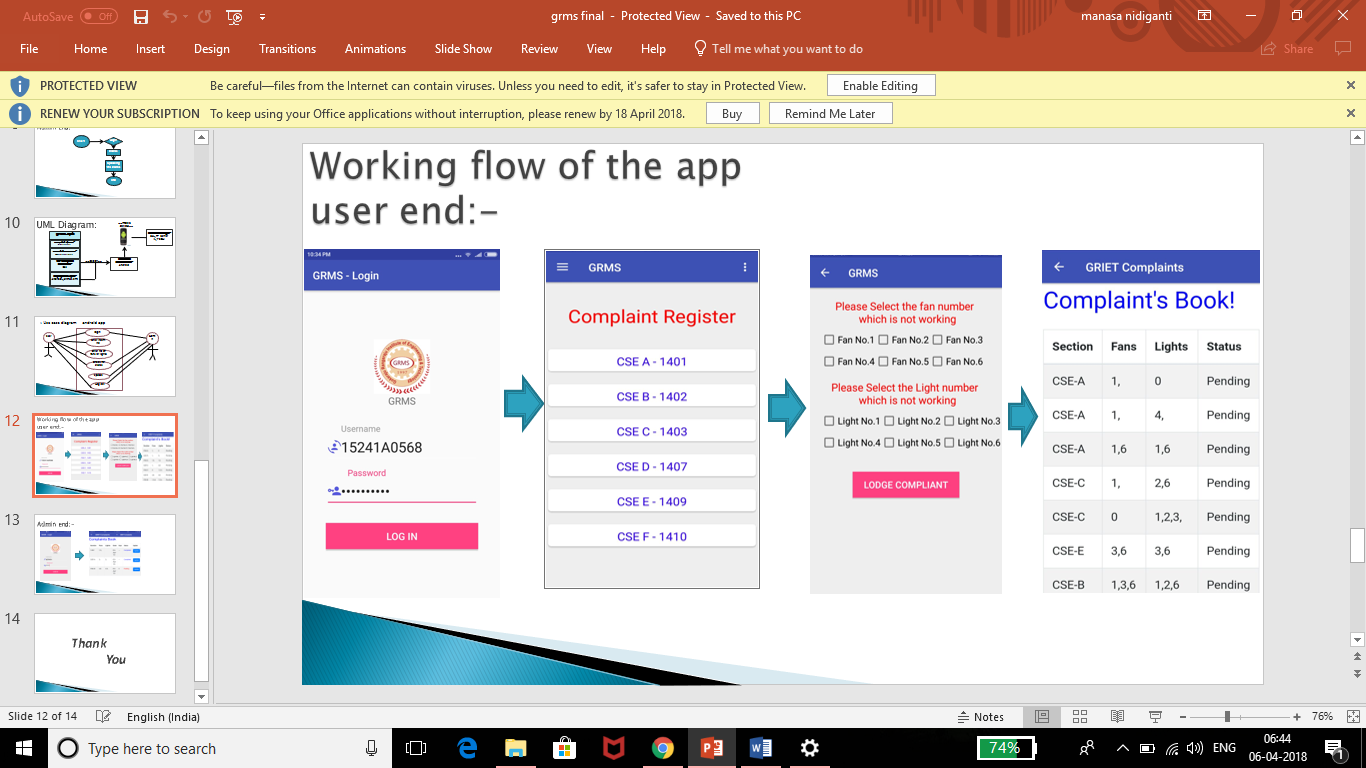
1. **User end:**

First, after opening the app, a login page appears. The user must log in to the app, with his/her respective login ID and password. If the details match with any of the CR’s details, then the app will be directed to the user side. After logging in, a page with all the room numbers is displayed. When the respective room is selected, the user is directed to a page where he will be asked to select the fan number or the light number that needs repairs. On clicking the submit button, the complaint will be lodged successfully. In the home page, there is a menu bar towards the extreme left, which has two options-complaints book and logout. Complaints book has the information about all the complaints that have been lodged till date. The status of the complaint whether it is repaired, or it is still pending, is shown for each complaint. By using the logout option, the user can log out of the app. This is the user side of the app.

**Dataflow of the App (User end):**



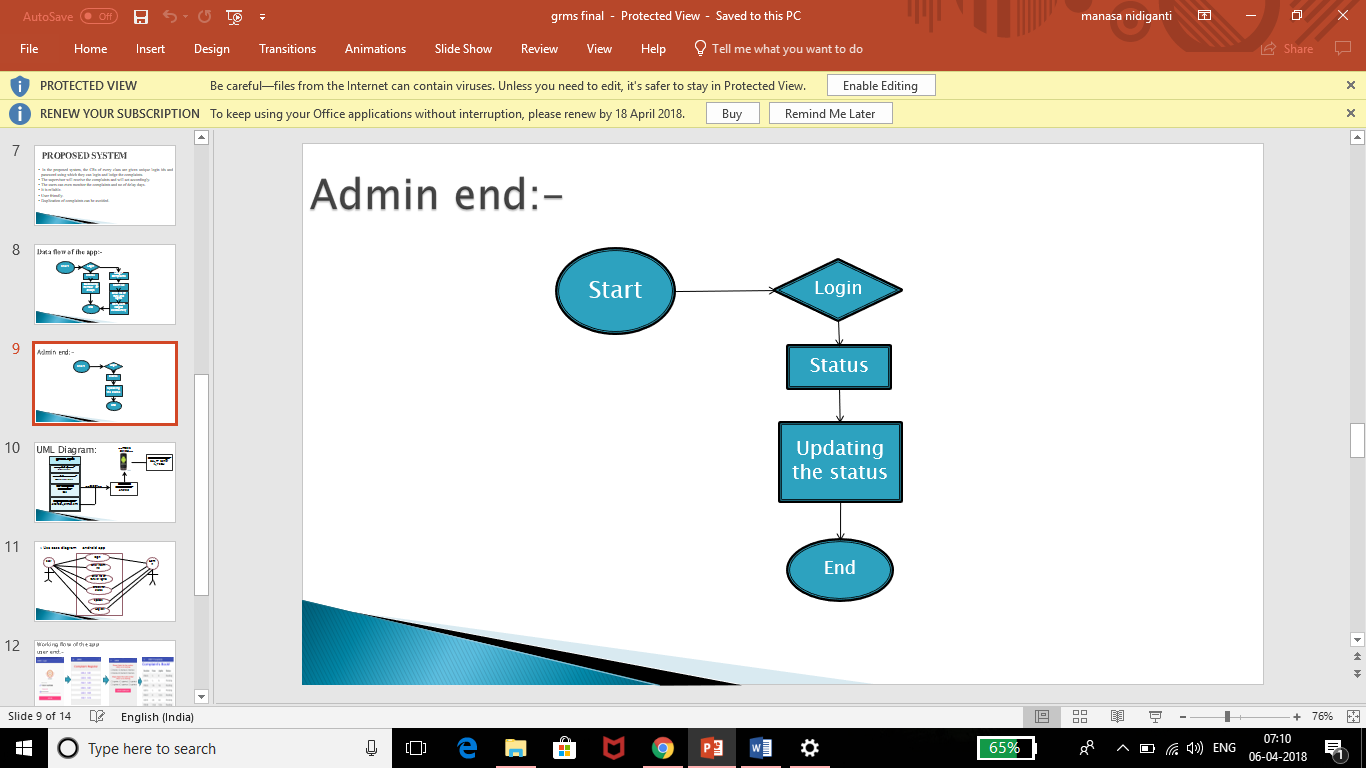
**Working Flow of the App(User end):**



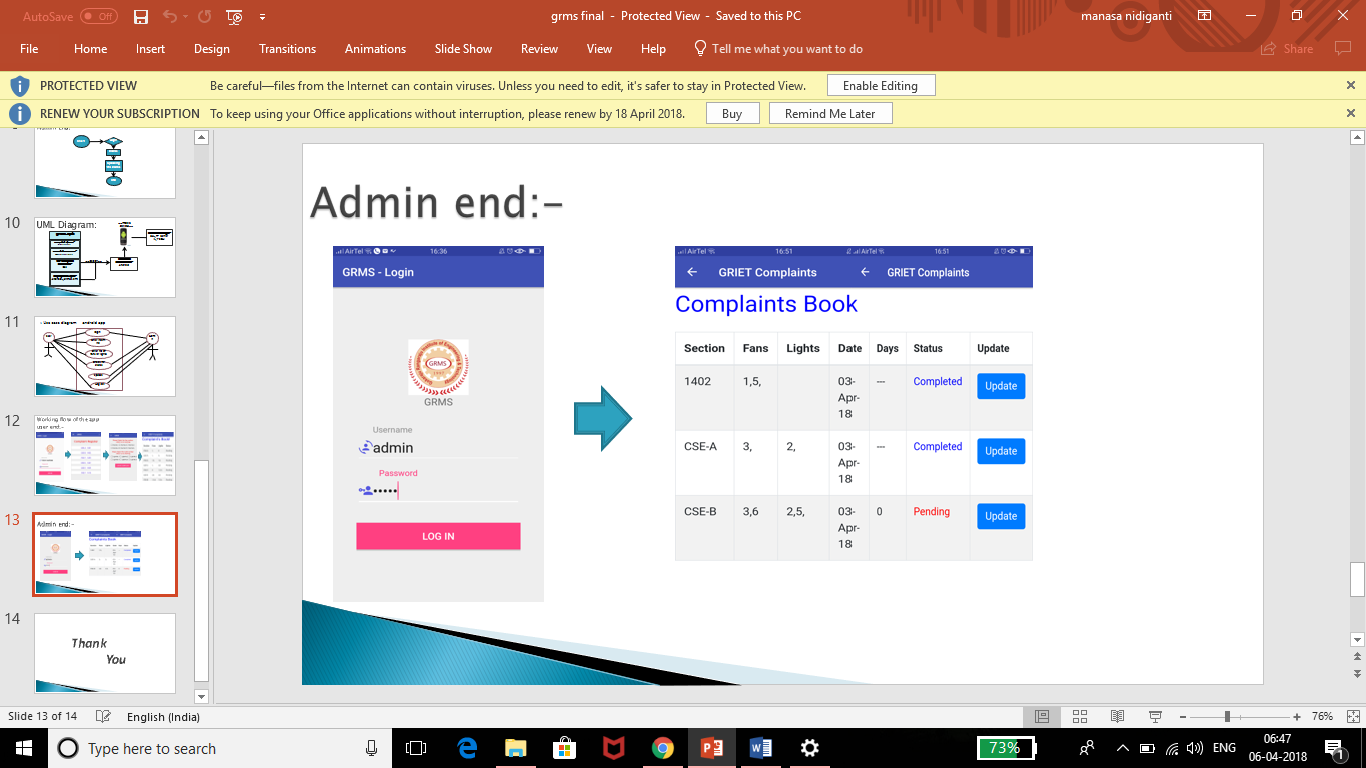
1. **Admin end:**

When the given user ID and password match with the details of the admin, then the app will be directed to the admin side. After the admin logs in, he will be directly navigated to the complaints book. Here, he can view all the complaints, old as well as new ones, that have been registered. At the end of each entry, he has an option to update the status of that entry.

**Dataflow of the App(Admin end):**



**Working Flow of the App (Admin end):**



**Backend Server**

Backend of anything is not visible to general user or administrator. It can be a piece of code or a program running on the server machine to serve the user need. The backend server that has been used for this application is XAMPP server.XAMPP stands for Cross-Platform (X), Apache (A), MariaDB (M), PHP (P) and Perl (P). XAMPP  is a free and an open source cross-platform webserver solution stack package developed by Apache Friends, consisting mainly of the Apache http server, MariaDB database or MySQL database and interpreters for scripts written in the PHP and Perl programming languages.  It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes. Everything needed to set up a web server – server application, database , and scripting language – is included in an extractable file. XAMPP is also cross-platform, which means it works equally well on Linux, Mac and Windows. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server extremely easy as well. We have used MySQL as our database in the XAMPP server.

**Frontend**

Frontend is all about what a user able to see. It may be the buttons, images, layout, input field etc. The front end of a website is the part that users interact with. Everything that you see when you’re navigating around the Internet, from fonts and colors to dropdown menus and sliders, is a combo of HTML, CSS, and JavaScript being controlled by your computer’s browser.

The programming language behind our Android app is Java. In addition, we have used XML, a markup language, to set up the layout of the app.

**UML Diagrams:-**

**Use case diagram**

Android app

**PROJECT CODE**

**Splash Activity:**

**Splash Activity java**:

**import**.android.app.Activity;  
**import** android.content.Context;  
**import** android.content.DialogInterface;  
**import** android.content.Intent;  
**import** android.net.ConnectivityManager;  
**import** android.net.NetworkInfo;  
**import** android.os.Bundle;  
**import** android.os.Handler;  
**import** android.widget.Toast;  
**import** de.mrapp.android.dialog.ProgressDialog;  
  
**public class** SplashActivity **extends** Activity {  
  
 DBHelper **dbHelper**;  
 @Override  
 **public void** onCreate(Bundle icicle) {  
 **super**.onCreate(icicle);  
 setContentView(R.layout.***activity\_splash***);  
 **dbHelper** = **new** DBHelper(**this**);  
  
  
  
 **if** (*internetConnectionCheck*(SplashActivity.**this**)) {  
**try**{  
  
 **new** Handler().postDelayed(**new** Runnable(){  
 @Override  
 **public void** run() {  
 **dbHelper**.getData();  
 **if**(Global.*au*.**Name**.equals(**""**)) {  
 startActivity(**new** Intent(getApplicationContext(),LoginActivity.**class**)); }  
  
 **else** { startActivity(**new** Intent(getApplicationContext(),LoginActivity.**class**)); }  
  
 finish();  
 }  
 }, 1000);  
  
 }**catch** (Exception e) {e.printStackTrace();}  
  
  
 } **else** {  
 dialog();  
 }  
  
  
 }  
  
  
  
  
 **public static boolean** internetConnectionCheck(Activity CurrentActivity) {  
 Boolean Connected = **false**;  
 ConnectivityManager connectivity = (ConnectivityManager) CurrentActivity.getApplicationContext()  
 .getSystemService(Context.***CONNECTIVITY\_SERVICE***);  
 **if** (connectivity != **null**) {  
 NetworkInfo[] info = connectivity.getAllNetworkInfo();  
 **if** (info != **null**) **for** (**int** i = 0; i < info.**length**; i++)  
 **if** (info[i].getState() == NetworkInfo.State.***CONNECTED***) {  
 Connected = **true**;  
 }  
 } **else** {  
 Toast.*makeText*(CurrentActivity.getApplicationContext(),  
 **"Please Check Your internet connection"**,  
 Toast.***LENGTH\_LONG***).show();  
 Connected = **false**;  
  
 }  
 **return** Connected;  
 }  
  
  
 **public void** dialog()  
 {  
 ProgressDialog.Builder dialogBuilder = **new** ProgressDialog.Builder(**this**);  
 dialogBuilder.setTitle(**"No Internet Connection!!"**);  
 dialogBuilder.setMessage(**"Please Check Ur Internet Connection"**);  
 dialogBuilder .setPositiveButton(**"OK"**, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialogInterface, **int** i) {  
 finish();  
  
 }  
 } );  
  
 dialogBuilder.setProgressBarPosition(ProgressDialog.ProgressBarPosition.***LEFT***);  
 ProgressDialog dialog = dialogBuilder.create();  
 dialog.show();  
  
 }  
  
}

**Activity\_splash.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"  
android:layout\_width="match\_parent"  
android:layout\_height="match\_parent"  
android:background="#ffffff"**>  
  
  
<**ImageView  
 android:id="@+id/imageView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="#00ffffff"  
 android:gravity="center"  
 android:scaleType="center"  
 android:src="@drawable/logo"** />  
  
  
</**RelativeLayout**>

**Output:**



**LOGIN ACTIVITY:**

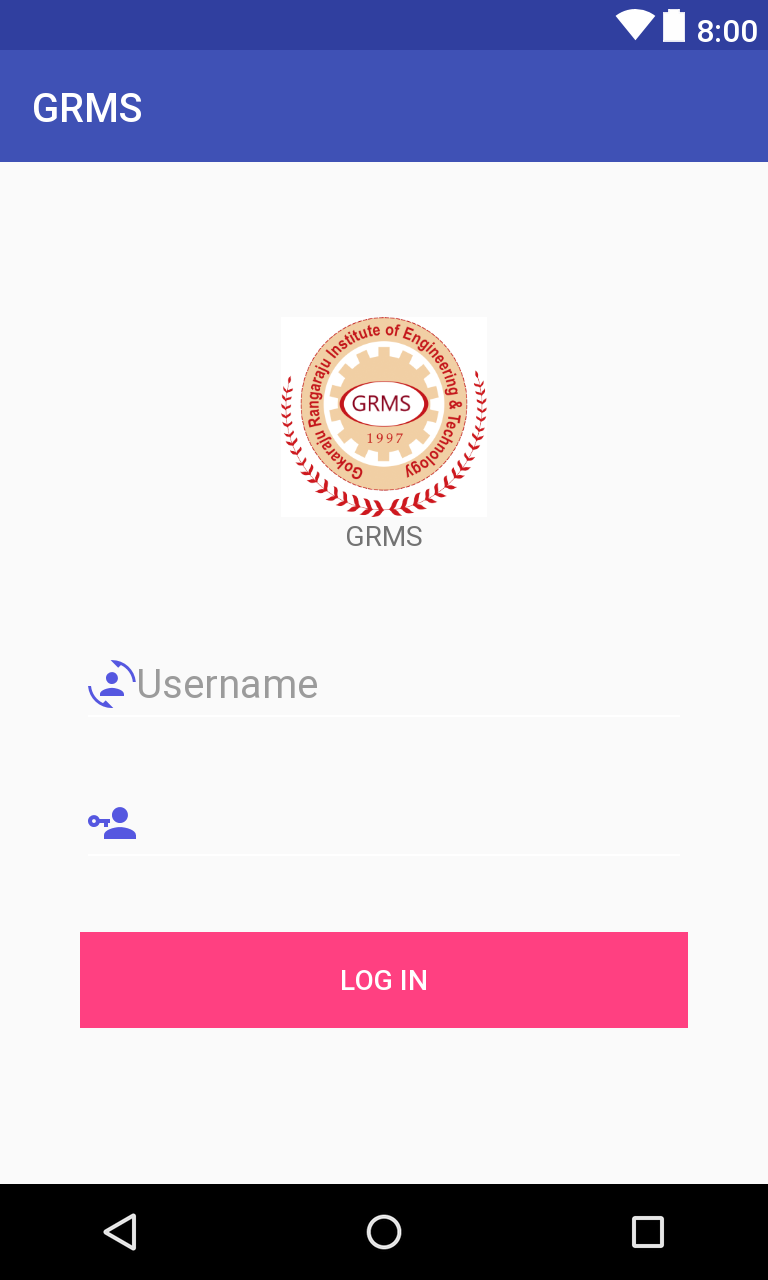
**loginActivity.java:**

**public class** LoginActivity **extends** AppCompatActivity {  
  
**private** EditText **EditTextpassword**,**EditTextnric**;  
 **private** Button **Buttonlogin**;  
 **private** ProgressDialog **pDialog**;  
  
 DBHelper **dbHelper** = **null**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_login***);  
  
 **EditTextnric** = (EditText) findViewById(R.id.***roll\_no***);  
 **EditTextpassword** = (EditText) findViewById(R.id.***password***);  
  
 **dbHelper** = **new** DBHelper(**this**);  
  
  
 }  
  
  
 **public void** login(View view){  
  
 **pDialog** = **new** ProgressDialog(LoginActivity.**this**);**pDialog**.setMessage(**"Logging In..."**);  
 **pDialog**.setCancelable(**false**);**pDialog**.show();  
RestAdapter adapter = **new** RestAdapter.Builder()  
 .setEndpoint(Global.*ROOT\_URL*) .build();MKAPI api = adapter.create(MKAPI.**class**);  
  
  
  
api.loginuser(  
  
**EditTextnric**.getText().toString(),  
 **EditTextpassword**.getText().toString(),  
  
**new** Callback<Response>() {  
 @Override  
 **public void** success(Response result, Response response) {  
BufferedReader reader = **null**;  
  
String output = **""**;  
  
 **try** {  
reader = **new** BufferedReader(**new** InputStreamReader(result.getBody().in()));  
  
output = reader.readLine();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
  
 **try** {  
 JSONObject jsonObject = **new** JSONObject(output);  
 **int** rescode = jsonObject.getInt(**"resCode"**);  
 **if**(rescode==1)  
 {  
 String fname = jsonObject.getString(**"fname"**);  
 String username = jsonObject.getString(**"username"**);  
 String section = jsonObject.getString(**"section"**);  
 **if** (**pDialog**.isShowing())  
 **pDialog**.dismiss();  
  
  
 User uobj = **new** User();  
 uobj.**Name** = fname;  
 uobj.**Username** = username;  
 uobj.**Section** = section;  
Global.*au* = uobj; **dbHelper**.insertUser(uobj);  
  
  
  
 Toast.*makeText*(LoginActivity.**this**,**"Welcome"**+**" "**+fname, Toast.***LENGTH\_LONG***).show();  
 startActivity(**new** Intent(getApplicationContext(),MainActivity.**class**));  
 finish();  
 }  
 **else if**(rescode==0)  
 {  
 **if** (**pDialog**.isShowing())  
 **pDialog**.dismiss();  
  
  
 String resMsg = jsonObject.getString(**"resMsg"**);  
 Toast.*makeText*(LoginActivity.**this**, resMsg, Toast.***LENGTH\_LONG***).show();  
 }  
  
 **else if**(rescode==3)  
 {  
 **if** (**pDialog**.isShowing())  
 **pDialog**.dismiss();  
  
  
startActivity(**new** Intent(getApplicationContext(),Complainttable.**class**));  
 finish();  
 }  
  
  
 } **catch** (JSONException e) {  
 e.printStackTrace();  
 }  
  
 }  
  
 @Override  
 **public void** failure(RetrofitError error) {  
 Toast.*makeText*(LoginActivity.**this**, error.toString(), Toast.***LENGTH\_LONG***).show();  
 }  
  
  
 }  
 );  
 }  
  
  
 **public void** signup(View v) {  
}  
  
 **public void** forgot\_password1(View v) {  
 }  
}

**Activity\_login.xml:**

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:gravity="center"  
 android:orientation="vertical"  
 android:padding="40dp"**>  
  
 <**ImageView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="100dp"  
 android:src="@drawable/logo"** />  
  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="GRMS"** />  
  
  
 <**android.support.design.widget.TextInputLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="30dp"** >  
  
 <**EditText  
 android:id="@+id/roll\_no"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:backgroundTint="#ffff"  
 android:drawableLeft="@drawable/user"  
 android:hint="Username"  
 android:inputType="textPersonName"  
 android:textSize="20sp"** />  
  
 </**android.support.design.widget.TextInputLayout**>  
  
 <**android.support.design.widget.TextInputLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="10dp"** >  
  
 <**EditText  
 android:id="@+id/password"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:backgroundTint="#ffff"  
 android:drawableLeft="@drawable/lock"  
 android:hint=" Password"  
 android:inputType="textPassword"  
 android:textSize="20sp"** />  
 </**android.support.design.widget.TextInputLayout**>  
  
<**Button  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="30dp"  
 android:background="#FF4081"  
 android:text="Log In"  
 android:textColor="#ffffff"  
 android:onClick="login"**/>  
  
 </**LinearLayout**>

**OUTPUT:**



**DATABASE CONNECTIVITY:**

**MKAPI:**

**import** retrofit.Callback;  
**import** retrofit.client.Response;  
**import** retrofit.http.Field;  
**import** retrofit.http.FormUrlEncoded;  
**import** retrofit.http.POST;  
  
  
**public interface** MKAPI {  
 @FormUrlEncoded  
 @POST(**"/login.php"**)  
 **public void** loginuser(  
 @Field(**"username"**) String username,  
 @Field(**"password"**) String password,  
 Callback<Response> callback);  
  
  
  
 @FormUrlEncoded  
 @POST(**"/signup.php"**)  
 **public void** signup(  
 @Field(**"fname"**) String fname,  
 @Field(**"phone\_number"**) String phone\_number,  
 @Field(**"emergency\_number"**) String emergency\_number,  
 @Field(**"email"**) String email,  
 @Field(**"emergency\_email"**) String emergency\_email,  
 @Field(**"password"**) String password,  
  
 Callback<Response> callback);  
  
 @FormUrlEncoded  
 @POST(**"/complaint.php"**)  
 **public void** complaint(  
 @Field(**"section"**) String section,  
 @Field(**"fans"**) String fans,  
 @Field(**"lights"**) String lights,  
 Callback<Response> callback);  
  
  
 @FormUrlEncoded  
 @POST(**"/location.php"**)  
 **public void** location(  
 @Field(**"email"**) String email,  
 @Field(**"latitude"**) String latitude,  
 @Field(**"longitude"**) String longitude,  
 Callback<Response> callback);  
  
  
}

**DBHelper.java:**

**public class** DBHelper **extends** SQLiteOpenHelper {  
  
  
  
 **private static final int *DATABASE\_VERSION*** = 1;  
 **public static final** String ***DATABASE\_NAME*** = **"gcap.db"**;  
  
  
 **public** DBHelper(Context context)  
 {  
 **super**(context, ***DATABASE\_NAME***, **null**, ***DATABASE\_VERSION***);  
 }  
  
@Override  
 **public void** onCreate(SQLiteDatabase db) {  
db.execSQL(**"CREATE TABLE IF NOT EXISTS users (Name text,Username text,Section text)"**);  
 }  
 @Override  
 **public void** onUpgrade(SQLiteDatabase db, **int** oldVersion, **int** newVersion) {  
 db.execSQL(**"DROP TABLE IF EXISTS users"**);  
 onCreate(db);  
 }  
  
 **public boolean** insertUser(User obj) {  
  
  
 SQLiteDatabase db = **this**.getWritableDatabase();  
 ContentValues contentValues = **new** ContentValues();  
 contentValues.put(**"Name"**, obj.**Name**);  
 contentValues.put(**"Username"**, obj.**Username**);  
 contentValues.put(**"Section"**, obj.**Section**);  
 db.insert(**"users"**, **null**, contentValues);  
 db.close();  
 **return true**;  
 }  
  
**public void** getData() {  
 SQLiteDatabase dbd = **this**.getReadableDatabase();  
 Cursor res = dbd.rawQuery( **"select \* from users"**, **null** );  
 **if**(res!=**null**)  
 {  
 **if**(res.moveToFirst())  
 {  
  
  
  
 Global.*au*.**Name** = res.getString(res.getColumnIndex(**"Name"**));  
 Global.*au*.**Username** = res.getString(res.getColumnIndex(**"Username"**));  
 Global.*au*.**Section** = res.getString(res.getColumnIndex(**"Section"**));  
}dbd.close();  
 }  
 }**public void** logout()  
 {  
 SQLiteDatabase db=**this**.getWritableDatabase();  
 Global.*clearData*();  
 db.close();  
  
 }  
  
  
**MAIN ACTIVITY:**

**MainActivity.java:**

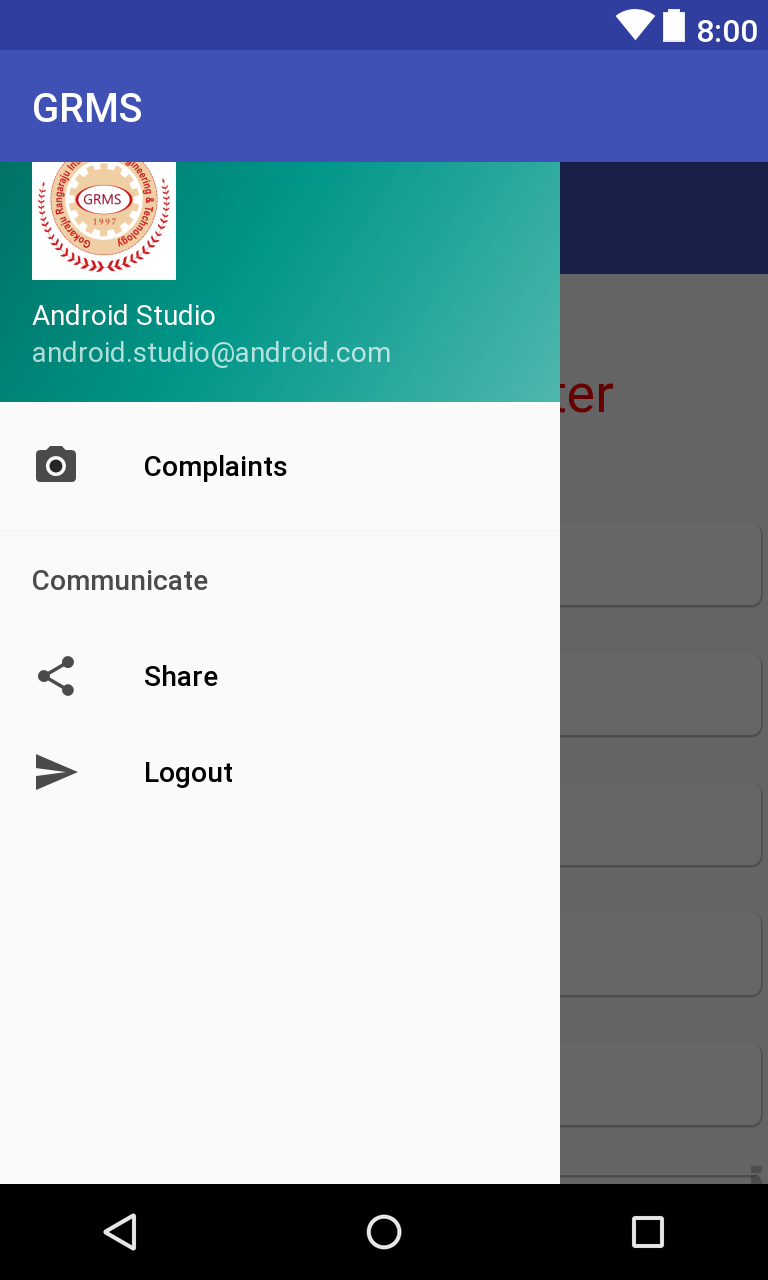
**public class** MainActivity **extends** AppCompatActivity  
 **implements** NavigationView.OnNavigationItemSelectedListener {  
  
 DBHelper **dbHelper**=**null**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 Toolbar toolbar = (Toolbar) findViewById(R.id.***toolbar***);  
 setSupportActionBar(toolbar);  
  
 Toast.*makeText*(**this**, **"Welcome"** + **" "** + Global.*au*.**Name**, Toast.***LENGTH\_SHORT***).show();  
 **dbHelper** = **new** DBHelper(**this**);  
  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 ActionBarDrawerToggle toggle = **new** ActionBarDrawerToggle(  
 **this**, drawer, toolbar, R.string.***navigation\_drawer\_open***, R.string.***navigation\_drawer\_close***);  
 drawer.addDrawerListener(toggle);  
 toggle.syncState();  
  
 NavigationView navigationView = (NavigationView) findViewById(R.id.***nav\_view***);  
 navigationView.setNavigationItemSelectedListener(**this**);  
  
  
 View header=navigationView.getHeaderView(0);  
  
 TextView nav\_Name=(TextView) header.findViewById(R.id.***nav\_name***);  
 nav\_Name.setText(Global.*au*.**Name**);  
  
 TextView nav\_Name12=(TextView) header.findViewById(R.id.***nav\_username***);  
 nav\_Name12.setText(Global.*au*.**Username**);  
  
 }  
  
 @Override  
 **public void** onBackPressed() {  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 **if** (drawer.isDrawerOpen(GravityCompat.***START***)) {  
 drawer.closeDrawer(GravityCompat.***START***);  
 } **else** {  
 **super**.onBackPressed();  
 }  
 }  
  
 @Override  
 **public boolean** onCreateOptionsMenu(Menu menu) {

getMenuInflater().inflate(R.menu.***main***, menu);  
 **return true**;  
 }  
  
 @Override  
 **public boolean** onOptionsItemSelected(MenuItem item) {  
**int** id = item.getItemId();  
  
 *//* **if** (id == R.id.***action\_settings***) {  
 **return true**;  
 }  
  
 **return super**.onOptionsItemSelected(item);  
 }  
  
 @SuppressWarnings(**"StatementWithEmptyBody"**)  
 @Override  
 **public boolean** onNavigationItemSelected(MenuItem item) {  
 **int** id = item.getItemId();  
  
 **if** (id == R.id.***nav\_camera***) {  
  
 startActivity(**new** Intent(**this**,Complainttable.**class**));  
  
 }

**else if** (id == R.id.***nav\_share***) {  
  
 } **else if** (id == R.id.***nav\_send***) {  
  
 logout();  
  
 }  
  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 drawer.closeDrawer(GravityCompat.***START***);  
 **return true**;  
 }  
  
  
  
  
 **public void** logout()  
 {  
  
 AlertDialog.Builder build = **new** AlertDialog.Builder(MainActivity.**this**);  
 build.setMessage(**"Are You Sure You Want To Logout?"**)  
 .setCancelable(**false**)  
  
 .setPositiveButton(**"Yes"**, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialogInterface, **int** i) {  
  
 **dbHelper**.logout();  
 Intent intent = **new** Intent(MainActivity.**this**, LoginActivity.**class**);  
 startActivity(intent);  
 finish();  
 Toast.*makeText*(MainActivity.**this**,**"You Have Logged Out"**,Toast.***LENGTH\_SHORT***).show();  
 }  
 } )  
  
 .setNegativeButton(**"No"**, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialogInterface, **int** i) {  
 dialogInterface.cancel();  
 Toast.*makeText*(MainActivity.**this**,**"You Have Come Back"**,Toast.***LENGTH\_SHORT***).show();  
  
 }  
 });  
 AlertDialog alert = build.create();  
 alert.setTitle(**"GRMS :)"**);  
 alert.show();  
 }  
  
  
  
  
 **public void** csea(View view){  
Global.*section*=**"CSE-A"**;  
 startActivity(**new** Intent(MainActivity.**this**,ComplaintActivity.**class**));  
 }  
  
  
  
 **public void** cseb(View view){  
 Global.*section*=**"CSE-B"**;  
 startActivity(**new** Intent(MainActivity.**this**,ComplaintActivity.**class**));  
 }  
  
 **public void** csec(View view){  
 Global.*section*=**"CSE-C"**;  
 startActivity(**new** Intent(MainActivity.**this**,ComplaintActivity.**class**));  
 }  
  
  
  
 **public void** csed(View view){  
 Global.*section*=**"CSE-D"**;  
 startActivity(**new** Intent(MainActivity.**this**,ComplaintActivity.**class**));  
 }  
  
 **public void** csee(View view){  
 Global.*section*=**"CSE-E"**;  
 startActivity(**new** Intent(MainActivity.**this**,ComplaintActivity.**class**));  
 }  
  
  
  
 **public void** csef(View view){  
 Global.*section*=**"CSE-F"**;  
 startActivity(**new** Intent(MainActivity.**this**,ComplaintActivity.**class**));  
 }  
  
  
  
  
**Activity\_main.xml:**

*<?***xml version="1.0" encoding="utf-8"***?>*<**android.support.v4.widget.DrawerLayout 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:id="@+id/drawer\_layout"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:fitsSystemWindows="true"  
 tools:openDrawer="start"**>  
  
 <**include  
 layout="@layout/app\_bar\_main"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"** />  
  
 <**android.support.design.widget.NavigationView  
 android:id="@+id/nav\_view"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_gravity="start"  
 android:fitsSystemWindows="true"  
 app:headerLayout="@layout/nav\_header\_main"  
 app:menu="@menu/activity\_main\_drawer"** />  
  
</**android.support.v4.widget.DrawerLayout**>

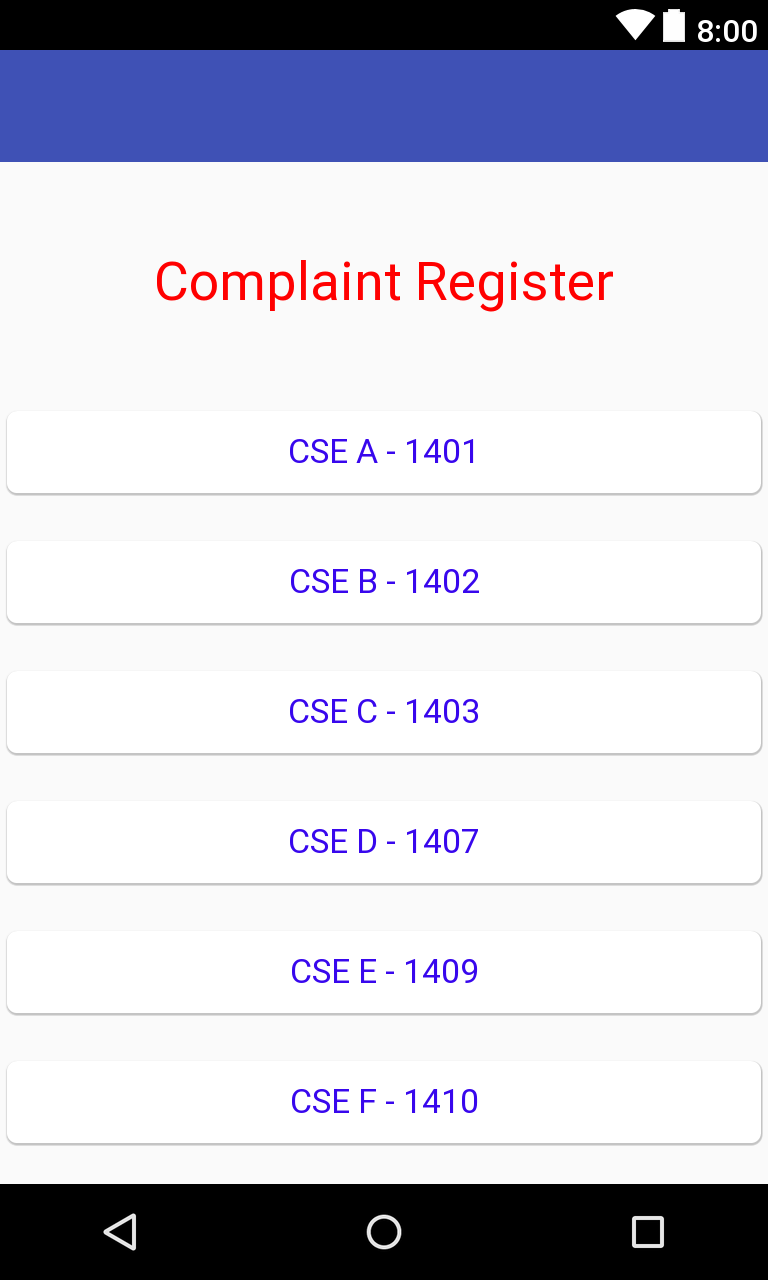
**OUTPUT:**



**Contents\_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:card\_view="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 app:layout\_behavior="@string/appbar\_scrolling\_view\_behavior"  
 tools:context=".MainActivity"  
 tools:showIn="@layout/app\_bar\_main"**>  
  
  
 <**android.support.v7.widget.CardView  
 android:id="@+id/card\_view"  
 android:layout\_width="match\_parent"  
 android:layout\_height="50dp"  
 android:layout\_marginTop="120dp"  
 android:layout\_gravity="center"  
 android:elevation="3dp"  
 card\_view:cardUseCompatPadding="true"  
 card\_view:cardElevation="2dp"  
 card\_view:cardCornerRadius="5dp"  
 android:onClick="csea"**>  
 </**android.support.v7.widget.CardView**>  
 <**android.support.v7.widget.CardView  
 android:id="@+id/card\_view1"  
 android:layout\_width="match\_parent"  
 android:layout\_height="50dp"  
 android:layout\_gravity="center"  
 android:elevation="3dp"  
 android:layout\_below="@+id/card\_view"  
 android:layout\_marginTop="15dp"  
 card\_view:cardUseCompatPadding="true"  
 card\_view:cardElevation="2dp"  
 card\_view:cardCornerRadius="5dp"  
 android:onClick="cseb"**>  
 <**TextView  
 android:id="@+id/title1"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:textAlignment="center"  
 android:textSize="17sp"  
 android:layout\_marginTop="8dp"  
 android:textColor="#3807ed"  
 android:text="CSE B - 1402"  
 android:textStyle="normal"  
 android:typeface="sans"** />  
 </**android.support.v7.widget.CardView**>  
 <**android.support.v7.widget.CardView  
 android:id="@+id/card\_view2"  
 android:layout\_width="match\_parent"  
 android:layout\_height="50dp"  
 android:layout\_gravity="center"  
 android:elevation="3dp"  
 android:layout\_below="@+id/card\_view1"  
 android:layout\_marginTop="15dp"  
 card\_view:cardUseCompatPadding="true"  
 card\_view:cardElevation="2dp"  
 card\_view:cardCornerRadius="5dp"  
 android:onClick="csec"**>  
 <**TextView  
 android:id="@+id/title2"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:textAlignment="center"  
 android:textSize="17sp"  
 android:textColor="#3807ed"  
 android:layout\_marginTop="8dp"  
 android:text="CSE C - 1403"  
 android:textStyle="normal"  
 android:typeface="sans"** />  
 </**android.support.v7.widget.CardView**>  
 <**android.support.v7.widget.CardView  
 android:id="@+id/card\_view3"  
 android:layout\_width="match\_parent"  
 android:layout\_height="50dp"  
 android:layout\_gravity="center"  
 android:elevation="3dp"  
 android:layout\_below="@+id/card\_view2"  
 android:layout\_marginTop="15dp"  
 card\_view:cardUseCompatPadding="true"  
 card\_view:cardElevation="2dp"  
 card\_view:cardCornerRadius="5dp"  
 android:onClick="csed"**>  
 <**TextView  
 android:id="@+id/title3"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:textAlignment="center"  
 android:textSize="17sp"  
 android:textColor="#3807ed"  
 android:layout\_marginTop="8dp"  
 android:text="CSE D - 1407"  
 android:textStyle="normal"  
 android:typeface="sans"** />  
 </**android.support.v7.widget.CardView**>  
 <**android.support.v7.widget.CardView  
 android:id="@+id/card\_view4"  
 android:layout\_width="match\_parent"  
 android:layout\_height="50dp"  
 android:layout\_gravity="center"  
 android:elevation="3dp"  
 android:layout\_below="@+id/card\_view3"  
 android:layout\_marginTop="15dp"  
 card\_view:cardUseCompatPadding="true"  
 card\_view:cardElevation="2dp"  
 card\_view:cardCornerRadius="5dp"  
 android:onClick="csee"**>  
<**TextView  
 android:id="@+id/title4"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:textAlignment="center"  
 android:textSize="17sp"  
 android:textColor="#3807ed"  
 android:layout\_marginTop="8dp"  
 android:text="CSE E - 1409"  
 android:textStyle="normal"  
 android:typeface="sans"** />  
 </**android.support.v7.widget.CardView**>  
 <**android.support.v7.widget.CardView  
 android:id="@+id/card\_view5"  
 android:layout\_width="match\_parent"  
 android:layout\_height="50dp"  
 android:layout\_gravity="center"  
 android:elevation="3dp"  
 android:layout\_below="@+id/card\_view4"  
 android:layout\_marginTop="15dp"  
 card\_view:cardUseCompatPadding="true"  
 card\_view:cardElevation="2dp"  
 card\_view:cardCornerRadius="5dp"  
 android:onClick="csef"**>  
 <**TextView  
 android:id="@+id/title5"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:textAlignment="center"  
 android:textSize="17sp"  
 android:textColor="#3807ed"  
 android:layout\_marginTop="8dp"  
 android:text="CSE F - 1410"  
 android:textStyle="normal"  
 android:typeface="sans"** />  
 </**android.support.v7.widget.CardView**>  
  
</**RelativeLayout**>

**OUTPUT:**



**LODGE COMPLAINTS:**

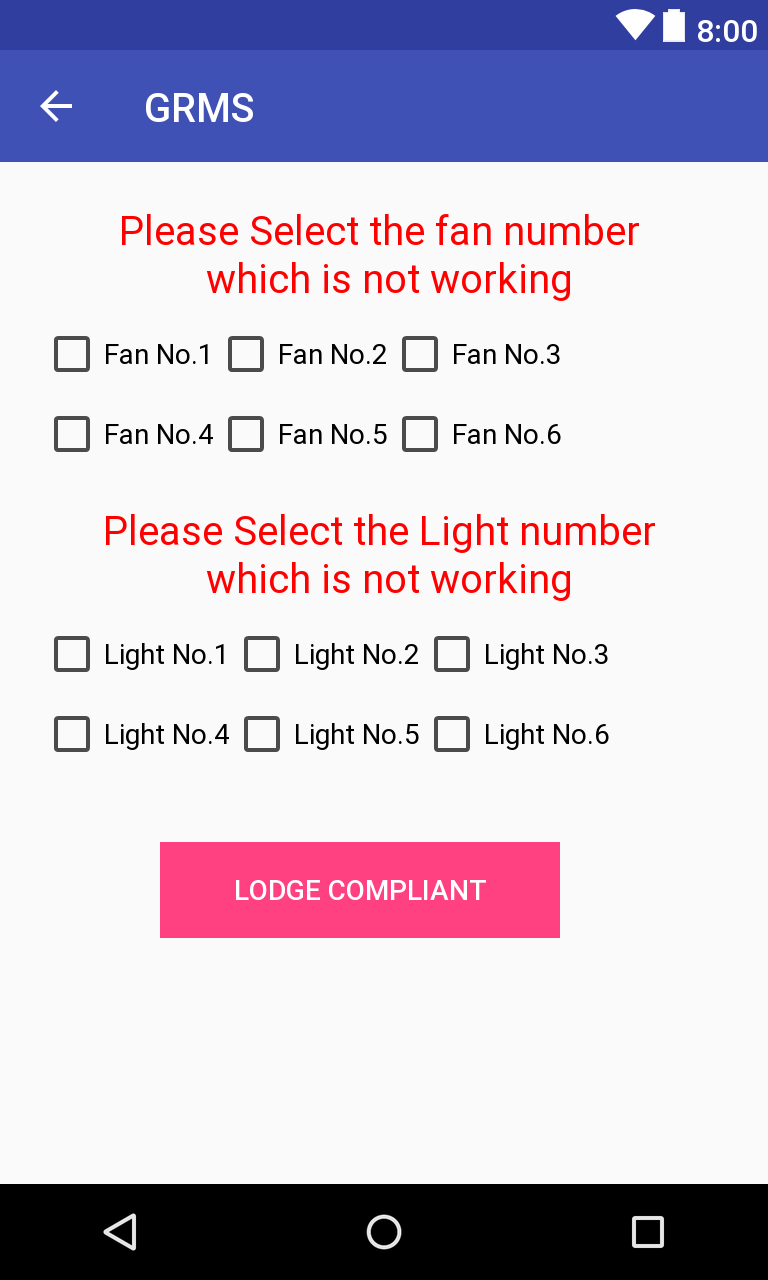
**ComplaintsActivity.java:**

**import** android.app.ProgressDialog;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.CheckBox;  
**import** android.widget.Toast;  
  
**import** org.json.JSONException;  
**import** org.json.JSONObject;  
  
**import** java.io.BufferedReader;  
**import** java.io.IOException;  
**import** java.io.InputStreamReader;  
  
**import** retrofit.Callback;  
**import** retrofit.RestAdapter;  
**import** retrofit.RetrofitError;  
**import** retrofit.client.Response;  
  
**public class** ComplaintActivity **extends** AppCompatActivity {  
  
 **public** CheckBox **c1**,**c2**,**c3**,**c4**,**c5**,**c6**,**l1**,**l2**,**l3**,**l4**,**l5**,**l6**;  
  
 **private** ProgressDialog **pDialog**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_complaint***);  
  
  
  
  
 **c1** = (CheckBox)findViewById(R.id.***c1***);  
 **c2** = (CheckBox) findViewById(R.id.***c2***);  
 **c3** = (CheckBox)findViewById(R.id.***c3***);  
 **c4** = (CheckBox)findViewById(R.id.***c4***);  
 **c5** = (CheckBox) findViewById(R.id.***c5***);  
 **c6** = (CheckBox) findViewById(R.id.***c6***);  
 **l1** = (CheckBox)findViewById(R.id.***l1***);  
 **l2** = (CheckBox) findViewById(R.id.***l2***);  
 **l3** = (CheckBox)findViewById(R.id.***l3***);  
 **l4** = (CheckBox)findViewById(R.id.***l4***);  
 **l5** = (CheckBox) findViewById(R.id.***l5***);  
 **l6** = (CheckBox) findViewById(R.id.***l6***);  
  
  
  
  
 }  
  
 **public void** lodge(View view){  
  
 **if** (**c1**.isChecked()) { Global.*test1*=**"1,"**; } **else** Global.*test1*=**""**;  
 **if** (**c2**.isChecked()) { Global.*test2*=**"2,"**; } **else** Global.*test2*=**""**;  
 **if** (**c3**.isChecked()) { Global.*test3*=**"3,"**; } **else** Global.*test3*=**""**;  
 **if** (**c4**.isChecked()) { Global.*test4*=**"4,"**; } **else** Global.*test4*=**""**;  
 **if** (**c5**.isChecked()) { Global.*test5*=**"5,"**; } **else** Global.*test5*=**""**;  
 **if** (**c6**.isChecked()) { Global.*test6*=**"6"**; } **else** Global.*test6*=**""**;  
  
 **if** (**l1**.isChecked()) { Global.*l1*=**"1,"**; } **else** Global.*l1*=**""**;  
 **if** (**l2**.isChecked()) { Global.*l2*=**"2,"**; } **else** Global.*l2*=**""**;  
 **if** (**l3**.isChecked()) { Global.*l3*=**"3,"**; } **else** Global.*l3*=**""**;  
 **if** (**l4**.isChecked()) { Global.*l4*=**"4,"**; } **else** Global.*l4*=**""**;  
 **if** (**l5**.isChecked()) { Global.*l5*=**"5,"**; } **else** Global.*l5*=**""**;  
 **if** (**l6**.isChecked()) { Global.*l6*=**"6"**; } **else** Global.*l6*=**""**;  
  
  
  
 Global.*test*=Global.*test1*+Global.*test2*+Global.*test3*+Global.*test4*+Global.*test5*+Global.*test6*;  
 Global.*l*=Global.*l1*+Global.*l2*+Global.*l3*+Global.*l4*+Global.*l5*+Global.*l6*;  
 complaint();  
  
  
 }  
  
 **public void** complaint(){  
  
 **pDialog** = **new** ProgressDialog(ComplaintActivity.**this**);**pDialog**.setMessage(**"Logging In..."**);  
 **pDialog**.setCancelable(**false**);**pDialog**.show();  
  
 RestAdapter adapter = **new** RestAdapter.Builder()  
 .setEndpoint(Global.*ROOT\_URL*)  
 .build();  
  
 MKAPI api = adapter.create(MKAPI.**class**);  
 api.complaint(  
  
 Global.*section*,  
 Global.*test*,  
 Global.*l*,  
  
 **new** Callback<Response>() {  
 @Override  
 **public void** success(Response result, Response response) {  
 BufferedReader reader = **null**;  
  
 String output = **""**;  
  
 **try** {  
  
 reader = **new** BufferedReader(**new** InputStreamReader(result.getBody().in()));  
 output = reader.readLine();  
 JSONObject jsonObject = **new** JSONObject(output);  
 **int** rescode = jsonObject.getInt(**"resCode"**);  
 **if**(rescode==1)  
 {  
 **if** (**pDialog**.isShowing())  
 **pDialog**.dismiss();  
  
  
 String resMsg = jsonObject.getString(**"resMsg"**);  
 Toast.*makeText*(ComplaintActivity.**this**, resMsg, Toast.***LENGTH\_LONG***).show();  
 }  
 **else if**(rescode==0)  
 {  
 **if** (**pDialog**.isShowing())  
 **pDialog**.dismiss();  
  
 String resMsg = jsonObject.getString(**"resMsg"**);  
 Toast.*makeText*(ComplaintActivity.**this**, resMsg, Toast.***LENGTH\_LONG***).show();  
 }  
  
  
  
  
  
 } **catch** (IOException|JSONException e) {  
 e.printStackTrace();  
 }  
  
  
 }  
  
 @Override  
 **public void** failure(RetrofitError error) {  
 Toast.*makeText*(ComplaintActivity.**this**, error.toString(), Toast.***LENGTH\_LONG***).show();  
 }  
  
  
 }  
 );  
 }  
  
  
 }

**Activity\_Complaints.xml:**

*<?***xml version="1.0" encoding="utf-8"***?>*<**android.support.constraint.ConstraintLayout 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=".ComplaintActivity"**>  
  
 <**TextView  
 android:layout\_marginTop="20dp"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:text="Please Select the fan number \n which is not working"  
 android:textColor="#f00"  
 android:gravity="center\_horizontal"  
 android:textSize="20dp"** />  
  
  
 <**RelativeLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_marginTop="70dp"  
 android:layout\_marginLeft="20dp"**>  
  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
 <**CheckBox  
 android:id="@+id/c1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Fan No.1"  
 android:layout\_marginTop="10dp"** />  
  
 <**CheckBox  
 android:id="@+id/c2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Fan No.2"  
 android:layout\_marginTop="10dp"** />  
  
 <**CheckBox  
 android:id="@+id/c3"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Fan No.3"  
 android:layout\_marginTop="10dp"** />  
  
 </**LinearLayout**>  
  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
  
 <**CheckBox  
 android:id="@+id/c4"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Fan No.4"  
 android:layout\_marginTop="50dp"** />  
  
 <**CheckBox  
 android:id="@+id/c5"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Fan No.5"  
 android:layout\_marginTop="50dp"** />  
  
 <**CheckBox  
 android:id="@+id/c6"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Fan No.6"  
 android:layout\_marginTop="50dp"** />  
  
 </**LinearLayout**>  
  
 </**RelativeLayout**>  
  
  
  
 <**TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:text="Please Select the Light number \n which is not working"  
 android:textColor="#f00"  
 android:layout\_marginTop="170dp"  
 android:gravity="center\_horizontal"  
 android:textSize="20dp"** />  
  
  
 <**RelativeLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_marginTop="220dp"  
 android:layout\_marginLeft="20dp"**>  
  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
 <**CheckBox  
 android:id="@+id/l1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Light No.1"  
 android:layout\_marginTop="10dp"** />  
  
 <**CheckBox  
 android:id="@+id/l2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Light No.2"  
 android:layout\_marginTop="10dp"** />  
  
 <**CheckBox  
 android:id="@+id/l3"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Light No.3"  
 android:layout\_marginTop="10dp"** />  
  
 </**LinearLayout**>  
  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
  
 <**CheckBox  
 android:id="@+id/l4"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Light No.4"  
 android:layout\_marginTop="50dp"** />  
  
 <**CheckBox  
 android:id="@+id/l5"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Light No.5"  
 android:layout\_marginTop="50dp"** />  
  
 <**CheckBox  
 android:id="@+id/l6"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Light No.6"  
 android:layout\_marginTop="50dp"** />  
  
 </**LinearLayout**>  
  
  
  
 <**Button  
 android:layout\_width="200dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="120dp"  
 android:background="#FF4081"  
 android:layout\_marginLeft="60dp"  
 android:text="Lodge Compliant"  
 android:textColor="#ffffff"  
 android:onClick="lodge"**/>  
  
 </**RelativeLayout**>  
</**android.support.constraint.ConstraintLayout**>

**OUTPUT:**



**CONCLUSION**

The ultimate goal of this project is to eliminate huge loss of man hours and manual intervention. The goal is to provide an effective platform to lodge the complaints and also to maintain a chain of responsibility and attend the complaints in an efficient way. The entire project has been built according to the requirements stated and can be implemented without errors. The complaints can be stored and can be monitored. The is very useful as it would reduce the stress and it is very easy to use. This process is less time consuming and is highly reliable.

This project can be scaled to add even more sophisticated features and be made more flexible. The application can be integrated by adding several other complaints regarding infrastructure etc. A communication system can be provided where the students can clarify their queries. This would enhance the standard of the application.

References:-

<https://developer.android.com/index.html> - to learn how to work with android.

<https://www.quora.com/> - provides solutions to the technical problems and queries.

<https://www.youtube.com> – used it to watch video lectures regarding creation of different pages.

<https://www.stackoverflow.com> – used it to learn the concepts of different features of android.

https://www.wikipedia.com – used to learn about backend services.