Secure Login application which include securing Activities and all secure coding practices in Android.

-By Saurav Chauhan
(sc07596@gmail.com)

- ✓ Register a user by providing user profile such as mobile no, email id, organization and other user relevant information
- ✓ Fetch the user credentials from a server to authenticate the user with username and password before providing access to the user
- ✓ Use SSL for communication between the client and server
- ✓ On successful login, go to the next screen. On failing login, alert user by toast
- ✓ If the user enters wrong credentials for three times, then terminate the applicationNext screen should display user profile
- ✓ No other application in the device should be able to start the internal activities except the login screen (use exported=fasle for all internal activities)
- ✓ Only use explicit intents to call activities within the application
- ✓ Sanitize all data inputedby the user before using it in the application

Literature survey:

Everyone loves a beautiful login screen, and since it's usually the very first impression people have about your app it's super important to get it right.
This article shows you how to create beautiful login and signup screens the right way using Material design spec with the assistance of Google's new design support library. The design support library implements a growing subset of the Material spec, and also includes a bunch of sexy UI widgets that can be used to give your Android apps that polished feel.
On the design and layout side of things, the focus here is on balancing the screen elements in a way that's pleasing to the eye. To add the finishing touches we will also be styling the top status bar on newer devices , and utilising floating labels from the design library (implemented via the TextInputLayout tag).

CODING:

MainActivity.java.

```
package com.example.sairamkrishna.myapplication;

import android.app.Activity;
import android.graphics.Color;
import android.os.Bundle;
import android.view.View;

import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;


public class MainActivity extends Activity  {
   Button b1,b2;
   EditText ed1,ed2;

   TextView tx1;
   int counter = 3;

   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);

      b1 = (Button)findViewById(R.id.button);
      ed1 = (EditText)findViewById(R.id.editText);
      ed2 = (EditText)findViewById(R.id.editText2);

      b2 = (Button)findViewById(R.id.button2);
      tx1 = (TextView)findViewById(R.id.textView3);
      tx1.setVisibility(View.GONE);

      b1.setOnClickListener(new View.OnClickListener() {
         @Override
         public void onClick(View v) {
            if(ed1.getText().toString().equals("admin") &&
               ed2.getText().toString().equals("admin")) {
```

```
                Toast.makeText(getApplicationContext(),
                   "Redirecting...",Toast.LENGTH_SHORT).show();
             }else{
                Toast.makeText(getApplicationContext(), "Wrong
                   Credentials",Toast.LENGTH_SHORT).show();

                tx1.setVisibility(View.VISIBLE);
                tx1.setBackgroundColor(Color.RED);
                counter--;
                tx1.setText(Integer.toString(counter));

                if (counter == 0) {
                   b1.setEnabled(false);
                }
             }
          }
       });

       b2.setOnClickListener(new View.OnClickListener() {
          @Override
          public void onClick(View v) {
             finish();
          }
       });
    }
}
```

Activity_Main.Xml

```
<?xml version = "1.0" encoding = "utf-8"?>
<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:paddingLeft=
"@dimen/activity_horizontal_margin"
   android:paddingRight = "@dimen/activity_horizontal_margin"
   android:paddingTop = "@dimen/activity_vertical_margin"
   android:paddingBottom = "@dimen/activity_vertical_margin" tools:context = ".MainActivity">

   <TextView android:text = "Login" android:layout_width="wrap_content"
      android:layout_height = "wrap_content"
      android:id = "@+id/textview"
      android:textSize = "35dp"
      android:layout_alignParentTop = "true"
      android:layout_centerHorizontal = "true" />

   <TextView
      android:layout_width = "wrap_content"
      android:layout_height = "wrap_content"
      android:text = "Tutorials point"
      android:id = "@+id/textView"
      android:layout_below = "@+id/textview"
      android:layout_centerHorizontal = "true"
      android:textColor = "#ff7aff24"
      android:textSize = "35dp" />

   <EditText
      android:layout_width = "wrap_content"
      android:layout_height = "wrap_content"
      android:id = "@+id/editText"
      android:hint = "Enter Name"
      android:focusable = "true"
      android:textColorHighlight = "#ff7eff15"
      android:textColorHint = "#ffff25e6"
      android:layout_marginTop = "46dp"
```

```
      android:layout_below = "@+id/imageView"
      android:layout_alignParentLeft = "true"
      android:layout_alignParentStart = "true"
      android:layout_alignParentRight = "true"
      android:layout_alignParentEnd = "true" />
<EditText
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:inputType="textPassword"
      android:ems="10"
      android:id="@+id/editText2"
      android:layout_below="@+id/editText"
      android:layout_alignParentLeft="true"
      android:layout_alignParentStart="true"
      android:layout_alignRight="@+id/editText"
      android:layout_alignEnd="@+id/editText"
      android:textColorHint="#ffff299f"
      android:hint="Password" />

   <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Attempts Left:"
      android:id="@+id/textView2"
      android:layout_below="@+id/editText2"
      android:layout_alignParentLeft="true"
      android:layout_alignParentStart="true"
      android:textSize="25dp" />

   <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="New Text"
      android:id="@+id/textView3"
      android:layout_alignTop="@+id/textView2"
      android:layout_alignParentRight="true"
      android:layout_alignParentEnd="true"
      android:layout_alignBottom="@+id/textView2"
      android:layout_toEndOf="@+id/textview"
      android:textSize="25dp"
      android:layout_toRightOf="@+id/textview" />

   <Button
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="login"
      android:id="@+id/button"
      android:layout_alignParentBottom="true"
      android:layout_toLeftOf="@+id/textview"
      android:layout_toStartOf="@+id/textview" />

   <Button
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Cancel"
      android:id="@+id/button2"
      android:layout_alignParentBottom="true"
      android:layout_toRightOf="@+id/textview"
      android:layout_toEndOf="@+id/textview" />

</RelativeLayout>
```

res/values/string.xml

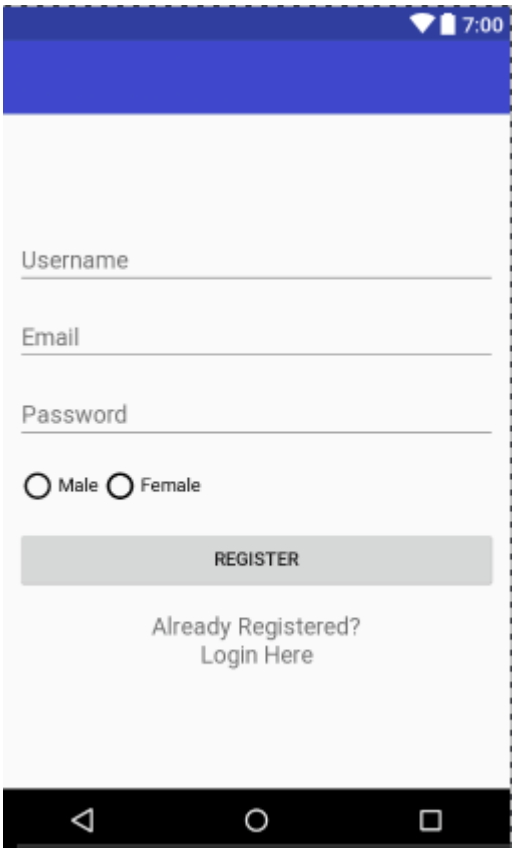
```
<resources>
   <string name="app_name">My Application</string>
</resources>
```

AndroidManifest.xml

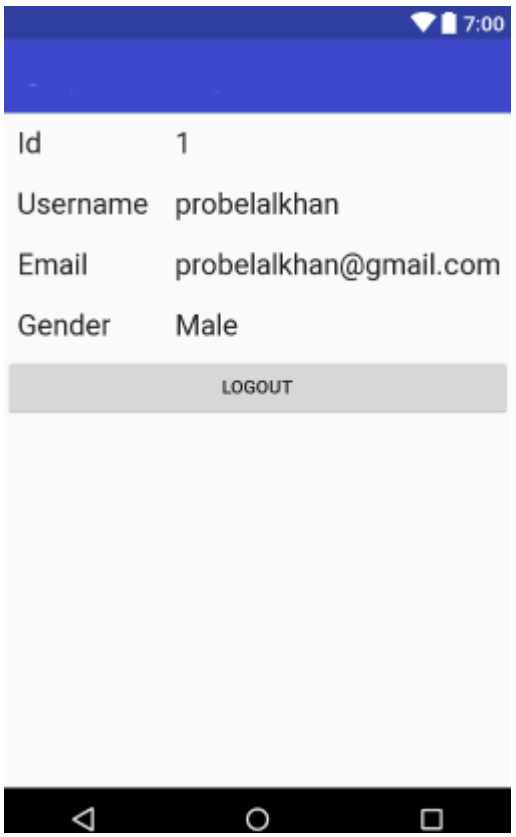
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
   package="com.example.sairamkrishna.myapplication" >

   <application
      android:allowBackup="true"
      android:icon="@mipmap/ic_launcher"
      android:label="@string/app_name"
      android:theme="@style/AppTheme" >

      <activity
         android:name=".MainActivity"
         android:label="@string/app_name" >

         <intent-filter>
            <action android:name="android.intent.action.MAIN" />
            <category android:name="android.intent.category.LAUNCHER" />
         </intent-filter>

      </activity>

   </application>
</manifest>
```

SCREENSHOTS:

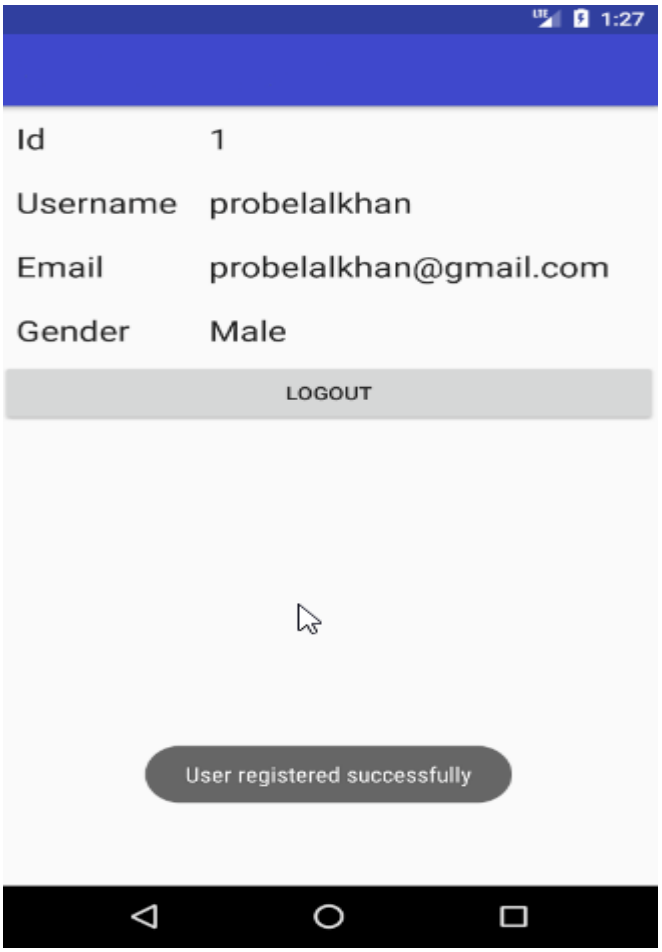
Login window:



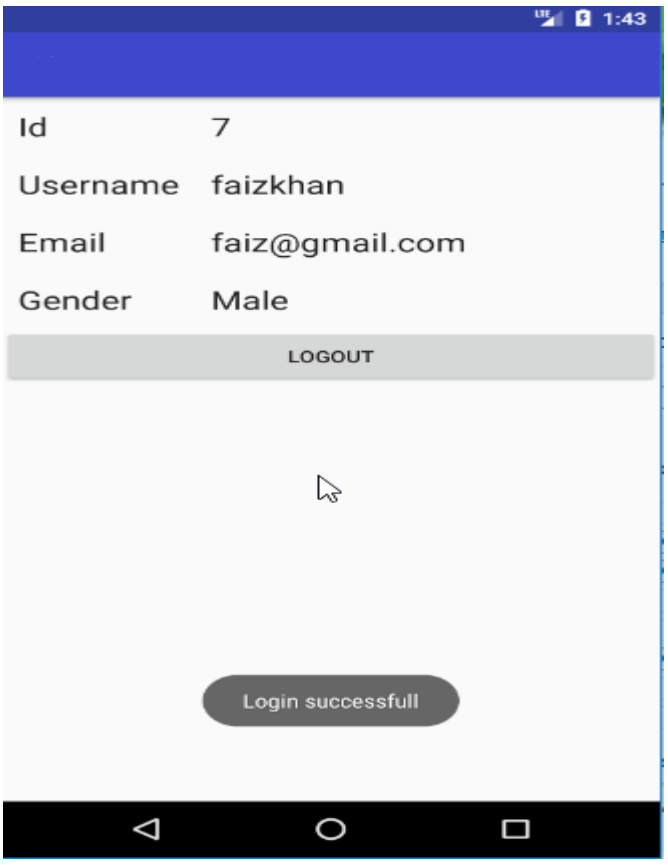
Profile Screen:



User Registration:



Successful Login:



CONCLUSION:

Security is a very important and difficult issue for every area. Everyone has a different kind of idea about “security'', to which extent security is needed and what type of risk is affordable. Secure architecture for any organization depends on the concept of security for that organization. Once that has been defined, it’s very significant to get their feedback, analyze it and to know about improvement that can be incorporated and it's important to know and understand the reasons of anything wrong happened in the system. The risk that should be acceptable should be defined by the organization. The organization should also know the steps that have been followed to decrease the organization's exposure to different kind of risks. Safety or security measures are everybody's business. It is possible to achieve it with only everyone's support. An intelligent policy or steady practices will be achievable with everyone’s cooperation. .

Security is fundamentally about protecting possessions or resources, or it can be assets. Resources may be objects, items, any hardware resource or any company’s database. It may be, otherwise, less significant, such as a company’s status. Security relies on elements such as authentication, authorization, auditing, confidentiality and integrity. The operating system is at the heart of the computer system. Security is very crucial part of the operating system. Different approaches have been proposed, but nothing is going to be perfect. Security should be maintained in spite of the common violations as theft, hacking, etc. Secure logon to the operating system avoids un-authenticated users from using the system and making unwanted changes in the system. Not only secure logon but also securing data and information from the system is very crucial. For that, in implementing work of the dissertation, Access Control Based models are implemented and analyzed. As windows operating system is not open source, models are simulated on ERP and cloud system which are having commercial importance.