

## ICP – 8

**ICP GROUP:** 38

**ICP 8**

**Name:** Anil Kumar Reddy

**Email:** [anggp@umsystem.edu](mailto:anggp@umsystem.edu)

**ICP8 Repository:** <https://github.com/UMKC-APL-WebMobileProgramming/ICP8-AnilkumarreddyNandikonda>

**ICP8 source code link:** <https://github.com/UMKC-APL-WebMobileProgramming/ICP8-AnilkumarreddyNandikonda/tree/main/Source>

### **My Partner**

**Partner name:** Abhinay Yadav

**Partner Email:** [ayr6y@umsystem.edu](mailto:ayr6y@umsystem.edu)

**Partner Repository:** <https://github.com/UMKC-APL-WebMobileProgramming/ICP8-YAbhinay>

**Source Code Link:** <https://github.com/UMKC-APL-WebMobileProgramming/ICP8-YAbhinay/tree/main/Source/ICP%208/app/src/main>

### **ICP8 video:**

<https://umsystem.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=5e86c32d-ac24-4b38-a3d9-adcc007a5cbd>

### **Lesson Overview:**

Android is a mobile platform/OS that uses a modified version of the Linux kernel. It was initially developed by Android Inc., a firm later purchased by Google. It allows developers to write managed code in the Java language, controlling the device via Google-developed Java libraries. Today, we will learn how to create, build, run, and debug an Android Studio program.

### **Programming elements:**

Basics of Android (Activity)

**In Class Programming (ICP):** Simple Login: Create a basic login application to understand the starter code given for text view, button view, and make a simple login application with the following requirements.

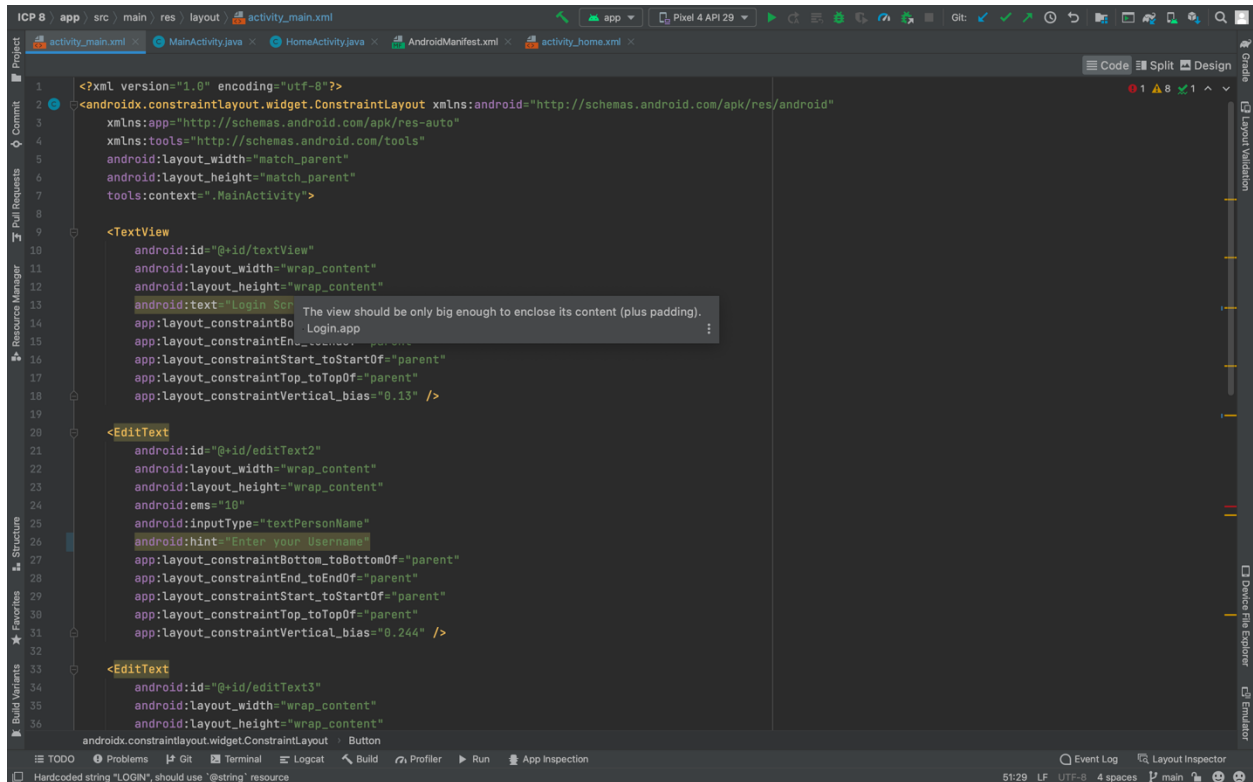
- a. The main activity should look as below (make suitable changes for the login app!).
- b. Create the 'Login' button on the login screen
- c. When clicked on the 'Login' button, the screen should navigate to the welcome screen if the login credentials are valid. Otherwise, the app should display an appropriate message to the user.
- d. Create the 'LOGOUT' button on the welcome screen
- e. When clicked on the 'LOGOUT' button, the screen should navigate to the login screen

Steps to follow:

1. Creating a new Android application
2. Selecting the SDK, type of device to deploy the application  
Device Type: Phone and Tablet and API: Latest one
3. Select blank activity

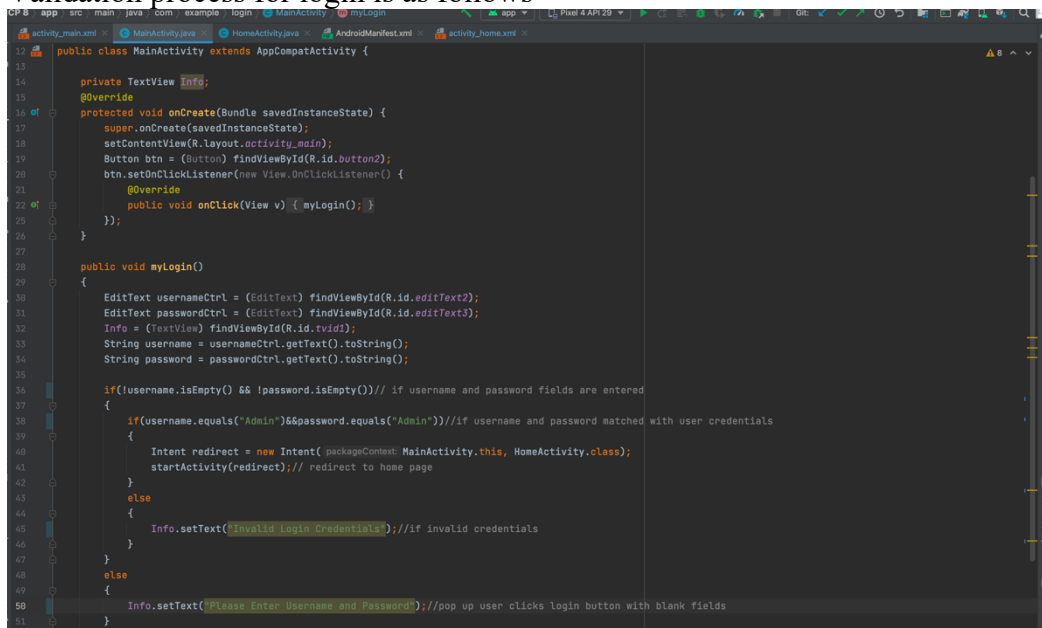
To make a basic login application, you'll need two text fields: one for the username and one for the password. It should also include the button activity, which should authenticate the username and password before redirecting to the main home page.

The login page layout file is as follows to meet the above requirement:



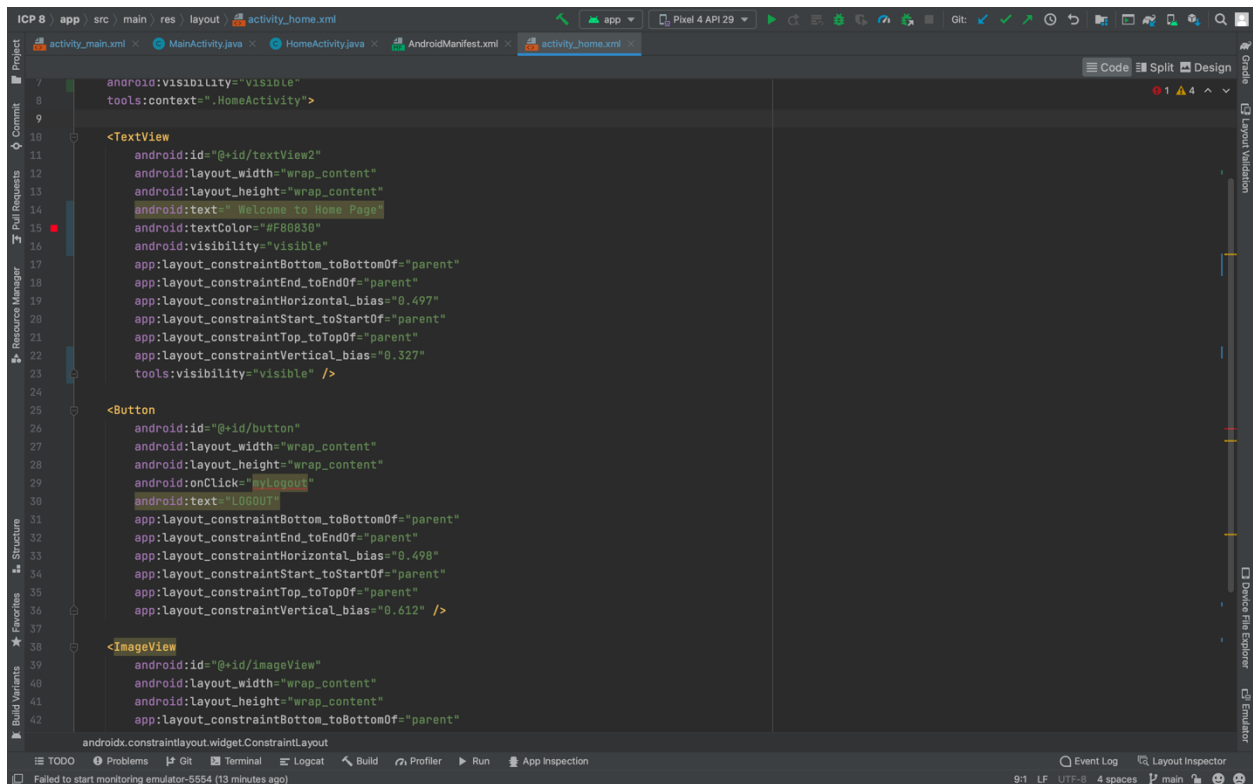
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Login Screen"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.13" />
    <EditText
        android:id="@+id/editText2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        android:hint="Enter your Username"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.244" />
    <EditText
        android:id="@+id/editText3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Validation process for login is as follows



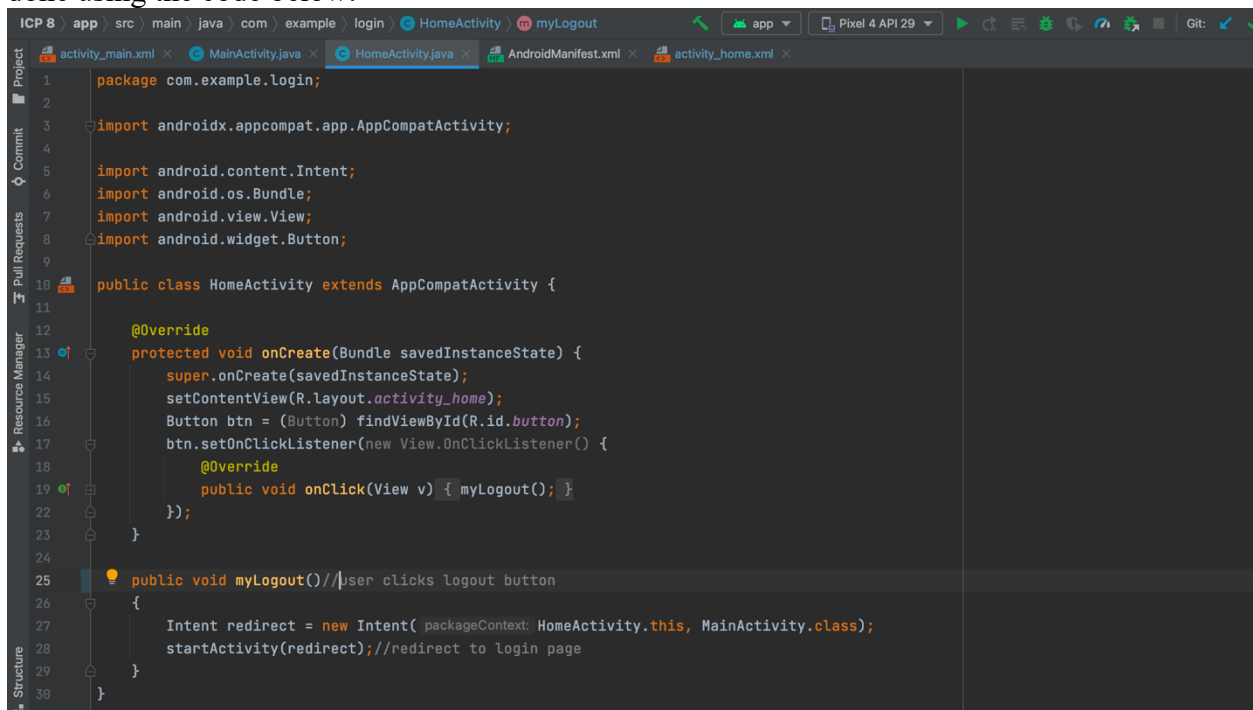
```
public class MainActivity extends AppCompatActivity {
    private TextView info;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Button btn = (Button) findViewById(R.id.button2);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) { myLogin(); }
        });
    }
    public void myLogin() {
        EditText usernameCtrl = (EditText) findViewById(R.id.editText2);
        EditText passwordCtrl = (EditText) findViewById(R.id.editText3);
        Info = (TextView) findViewById(R.id.tvId3);
        String username = usernameCtrl.getText().toString();
        String password = passwordCtrl.getText().toString();
        if((username.isEmpty() && !password.isEmpty()) || if username and password fields are entered
        {
            if(username.equals("Admin") && password.equals("Admin")) { //if username and password matched with user credentials
                Intent redirect = new Intent( packageContext MainActivity.this, HomeActivity.class);
                startActivity(redirect); // redirect to home page
            }
            else
            {
                Info.setText("Invalid Login Credentials"); //if invalid credentials
            }
        }
        else
        {
            Info.setText("Please Enter Username and Password"); //pop up user clicks login button with blank fields
        }
    }
}
```

After the user has been validated, the main page displays the following text and a logout button:



```
7      android:visibility="visible"
8      tools:context=".HomeActivity">
9
10     <TextView
11         android:id="@+id/textView2"
12         android:layout_width="wrap_content"
13         android:layout_height="wrap_content"
14         android:text="Welcome to Home Page"
15         android:textColor="#F8B300"
16         android:visibility="visible"
17         app:layout_constraintBottom_toBottomOf="parent"
18         app:layout_constraintEnd_toEndOf="parent"
19         app:layout_constraintHorizontal_bias="0.497"
20         app:layout_constraintStart_toStartOf="parent"
21         app:layout_constraintTop_toTopOf="parent"
22         app:layout_constraintVertical_bias="0.327"
23         tools:visibility="visible" />
24
25     <Button
26         android:id="@+id/button"
27         android:layout_width="wrap_content"
28         android:layout_height="wrap_content"
29         android:onClick="myLogout"
30         android:text="Logout"
31         app:layout_constraintBottom_toBottomOf="parent"
32         app:layout_constraintEnd_toEndOf="parent"
33         app:layout_constraintHorizontal_bias="0.498"
34         app:layout_constraintStart_toStartOf="parent"
35         app:layout_constraintTop_toTopOf="parent"
36         app:layout_constraintVertical_bias="0.612" />
37
38     <ImageView
39         android:id="@+id/imageView"
40         android:layout_width="wrap_content"
41         android:layout_height="wrap_content"
42         app:layout_constraintBottom_toBottomOf="parent"
```

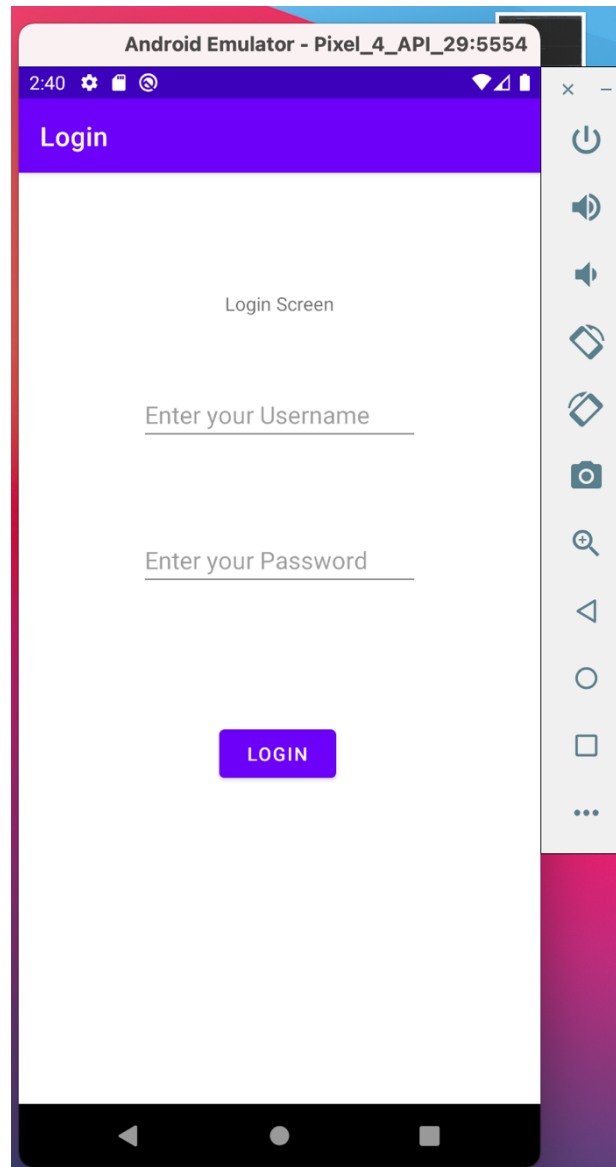
When a user clicks the logout button, the page must be redirected to the login page, which is done using the code below:

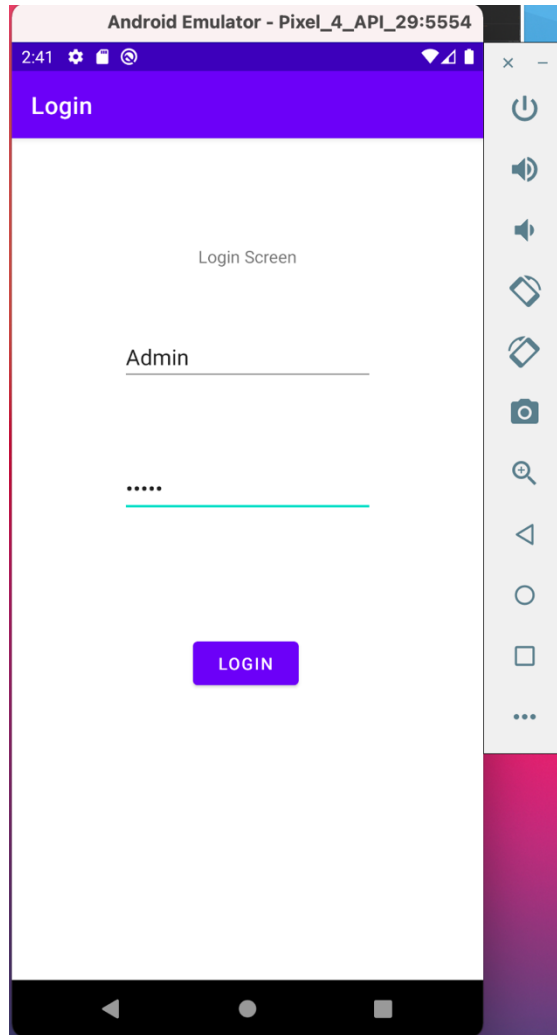


```
1 package com.example.login;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.content.Intent;
6 import android.os.Bundle;
7 import android.view.View;
8 import android.widget.Button;
9
10 public class HomeActivity extends AppCompatActivity {
11
12     @Override
13     protected void onCreate(Bundle savedInstanceState) {
14         super.onCreate(savedInstanceState);
15         setContentView(R.layout.activity_home);
16         Button btn = (Button) findViewById(R.id.button);
17         btn.setOnClickListener(new View.OnClickListener() {
18             @Override
19             public void onClick(View v) { myLogout(); }
20         });
21     }
22
23
24
25     public void myLogout()//user clicks logout button
26     {
27         Intent redirect = new Intent( packageContext: HomeActivity.this, MainActivity.class);
28         startActivity(redirect);//redirect to login page
29     }
30
31 }
```

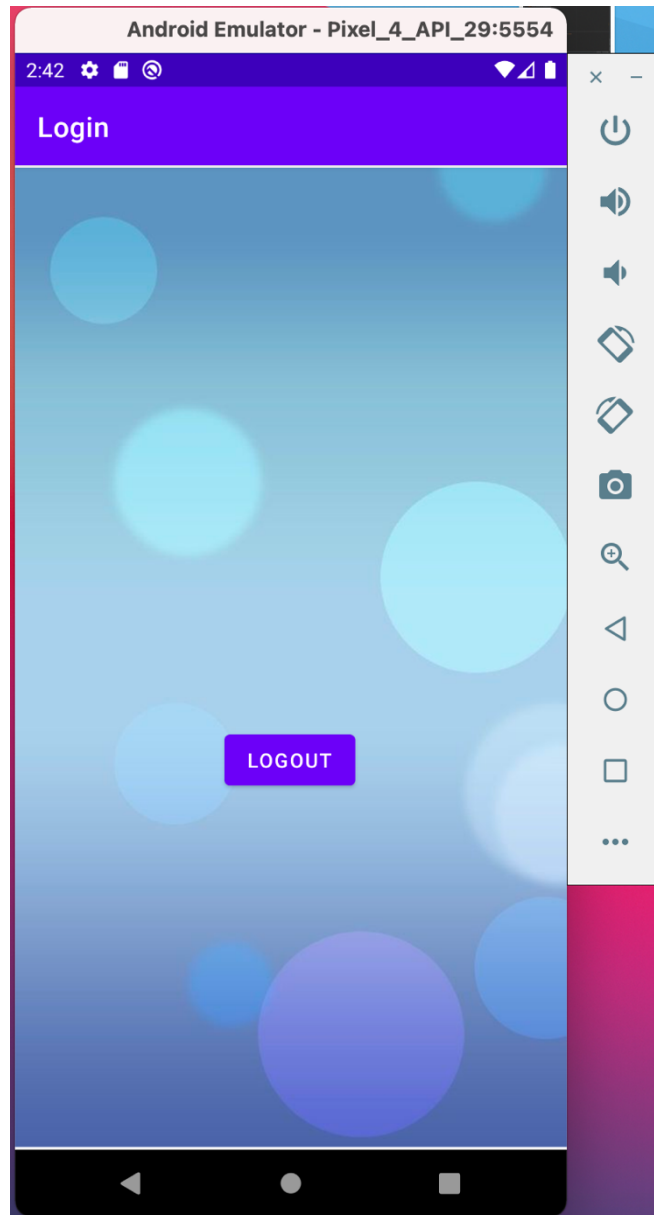
## OUTPUT:

The main page to enter the username and password





After entering the username and password it will validate and allows to welcome page.



If the username and password are not validating, a message will be shown up as follows:

