

ICP 8

Name: Aqduş Charolia Email: aacy5x@mail.umkc.edu

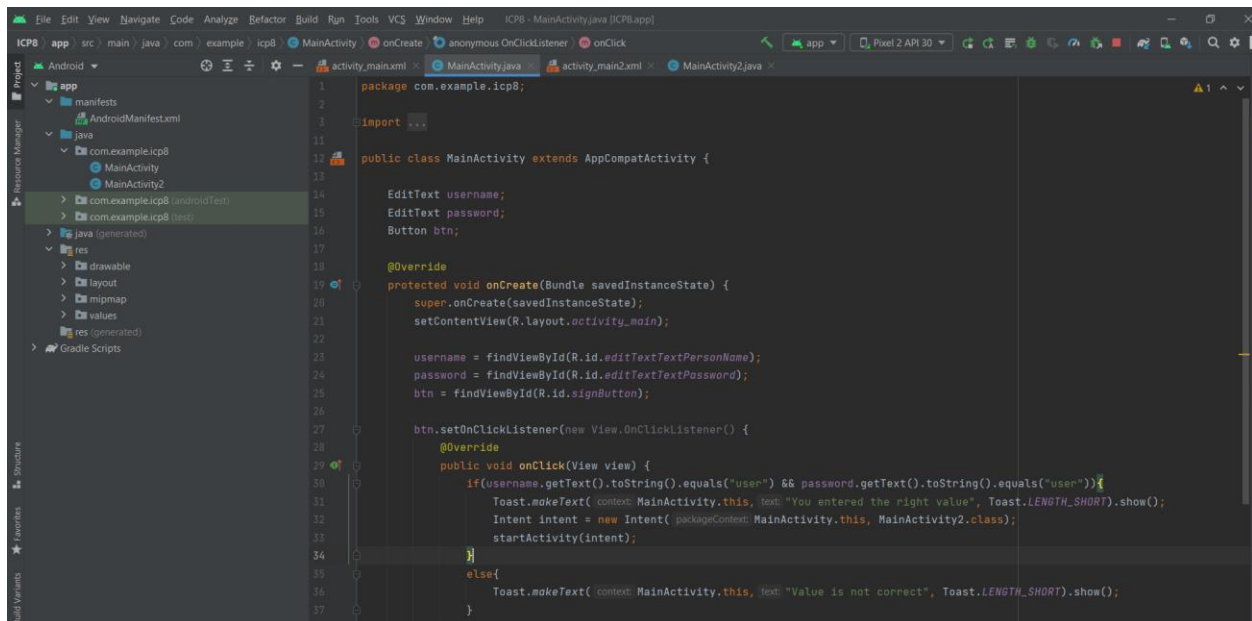
Github: https://github.com/Aqdusc/WebDevCourse/tree/main/Mobile_Development/ICP8

Name: Affan Charolia Email: aacbb8@mail.umkc.edu

Github: https://github.com/Affancharolia/WebDevCourse/tree/main/Mobile_Development/ICP8

Aim: The goal of this ICP is to create a mobile application, which has the login page and when the user enters login and password as user, they get to the welcome page and when they click on the logout button, they get back to the log in page.

Explanation:



```
1 package com.example.icp8;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 public class MainActivity extends AppCompatActivity {
6
7     EditText username;
8     EditText password;
9     Button btn;
10
11     @Override
12     protected void onCreate(Bundle savedInstanceState) {
13         super.onCreate(savedInstanceState);
14         setContentView(R.layout.activity_main);
15
16         username = findViewById(R.id.editTextTextPersonName);
17         password = findViewById(R.id.editTextTextPassword);
18         btn = findViewById(R.id.signButton);
19
20         btn.setOnClickListener(new View.OnClickListener() {
21             @Override
22             public void onClick(View view) {
23                 if(username.getText().toString().equals("user") && password.getText().toString().equals("user")){
24                     Toast.makeText(context, MainActivity.this, "You entered the right value", Toast.LENGTH_SHORT).show();
25                     Intent intent = new Intent(context, MainActivity2.class);
26                     startActivity(intent);
27                 }
28                 else{
29                     Toast.makeText(context, MainActivity.this, "Value is not correct", Toast.LENGTH_SHORT).show();
30                 }
31             }
32         });
33     }
34 }
```

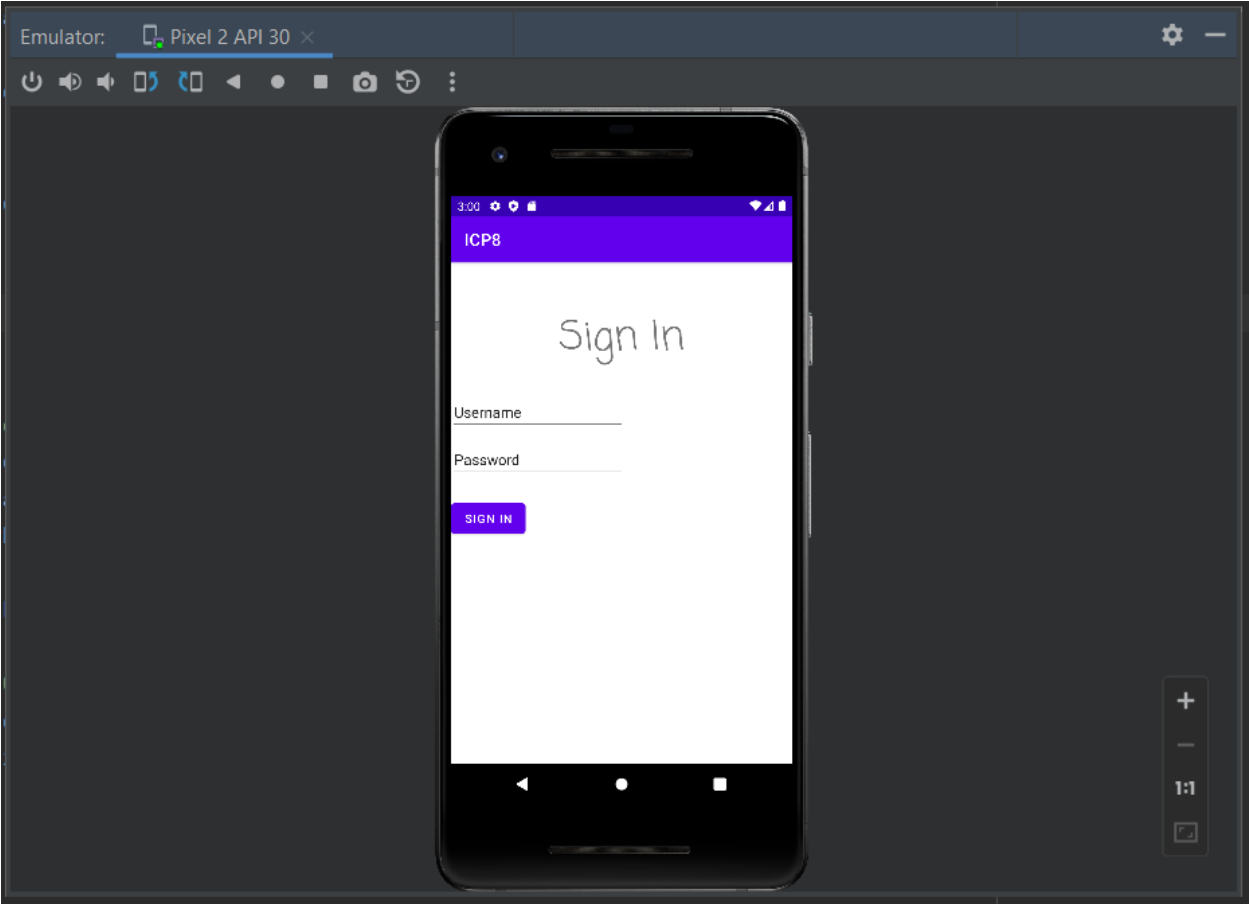
We create the MainActivity.java file to create a view. We create variables and link them to the fields using the id they are being assigned. When the fields are entered and submitted, the strings are matched with the string user and when matched, the toast function displays the title “you are entered the right way”. We then go to the next view using the intent function or if the value is incorrect, we display “value is not correct”.

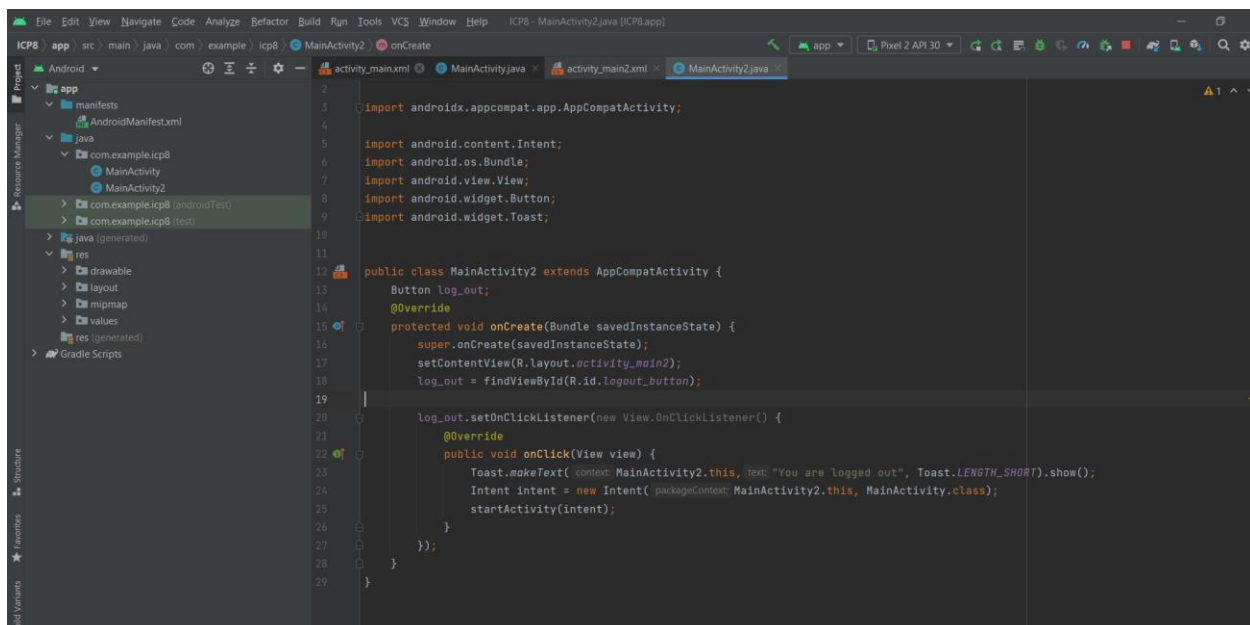
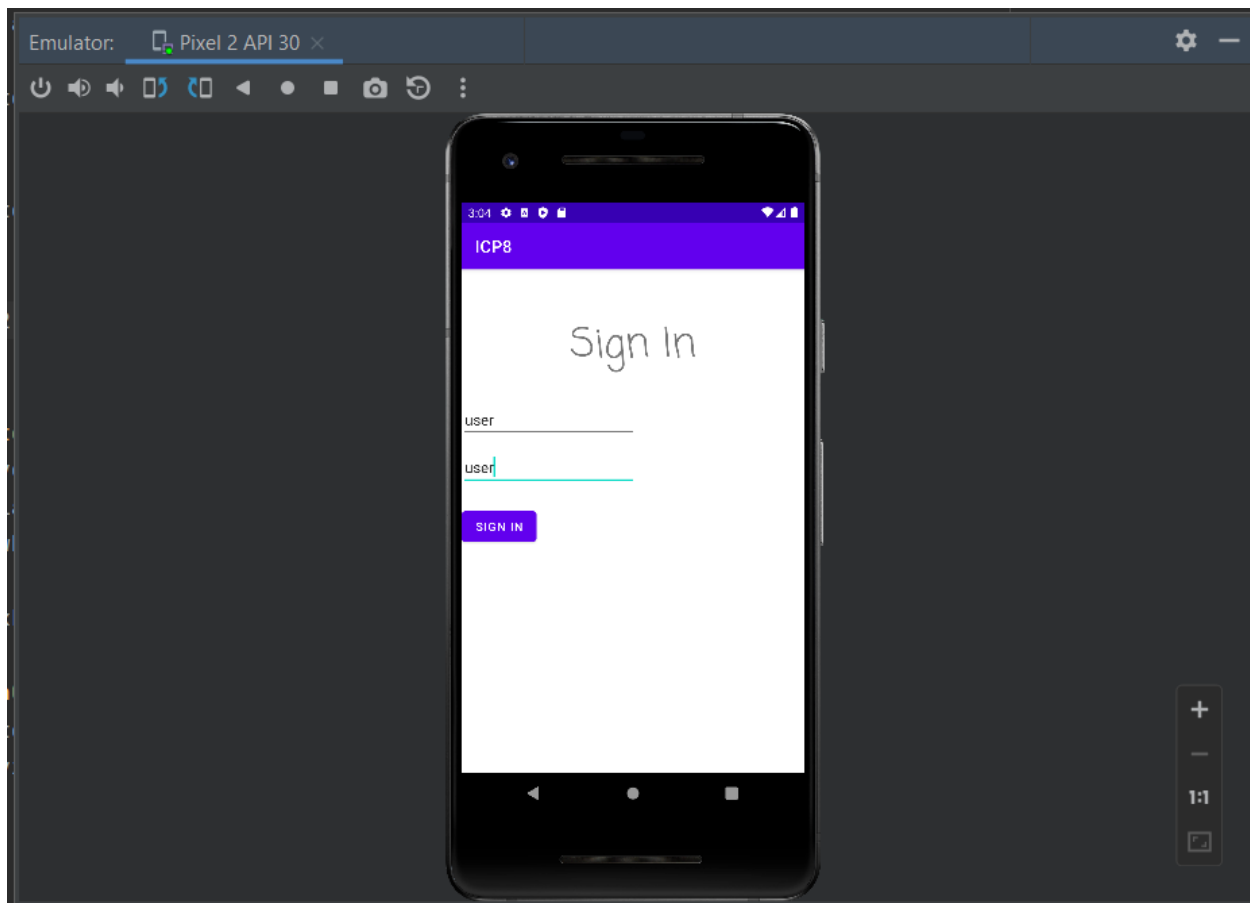
```
activity_main.xml x MainActivity.java x activity_main2.xml x MainActivity2.java x
Code Split Design
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   tools:context=".MainActivity">
8
9   <TextView
10     android:id="@+id/logo"
11     android:layout_width="wrap_content"
12     android:layout_height="wrap_content"
13     android:fontFamily="casual"
14     android:text="@string/sign_in"
15     android:textSize="48sp"
16     app:layout_constraintBottom_toBottomOf="parent"
17     app:layout_constraintLeft_toLeftOf="parent"
18     app:layout_constraintRight_toRightOf="parent"
19     app:layout_constraintTop_toTopOf="parent"
20     app:layout_constraintVertical_bias="0.103" />
21
22   <Button
23     android:id="@+id/signButton"
24     android:layout_width="wrap_content"
25     android:layout_height="wrap_content"
26     android:layout_marginTop="24dp"
27     android:text="@string/sign_in"
28     app:layout_constraintTop_toBottomOf="@id/editTextTextPassword"

```

```
ain.xml
activity_main.xml x MainActivity.java x activity_main2.xml x MainActivity2.java x
Code Split Design
29   tools:ignore="MissingConstraints"
30   tools:layout_editor_absoluteX="160dp" />
31
32   <EditText
33     android:id="@+id/editTextTextPersonName"
34     android:layout_width="wrap_content"
35     android:layout_height="wrap_content"
36     android:layout_marginTop="30dp"
37     android:ems="10"
38     android:inputType="textPersonName"
39     android:text="Username"
40     app:layout_constraintTop_toBottomOf="@id/logo"
41     tools:layout_editor_absoluteX="103dp" />
42
43   <EditText
44     android:id="@+id/editTextTextPassword"
45     android:layout_width="wrap_content"
46     android:layout_height="wrap_content"
47     android:layout_marginTop="12dp"
48     android:ems="10"
49     android:inputType="textPersonName"
50     android:text="Password"
51     app:layout_constraintTop_toBottomOf="@id/editTextTextPersonName"
52     tools:ignore="TouchTargetSizeCheck"
53     tools:layout_editor_absoluteX="97dp" />
54
55 </androidx.constraintlayout.widget.ConstraintLayout>

```





When we get to the welcome page, it displays the string and the logout button. On clicking the logout button we get to the login page again, along with the displayed “You are logged out” string.

```
activity_main.xml x MainActivity.java x activity_main2.xml x MainActivity2.java x
Code Split Design

1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   tools:context=".MainActivity2">
8
9   <TextView
10     android:id="@+id/textView1"
11     android:layout_width="210dp"
12     android:layout_height="19dp"
13     android:text="You have successfully logged in!"
14     tools:layout_editor_absoluteX="7dp"
15     tools:layout_editor_absoluteY="144dp" />
16
17   <Button
18     android:id="@+id/logout_button"
19     android:layout_width="wrap_content"
20     android:layout_height="wrap_content"
21     android:layout_marginTop="28dp"
22     android:text="Logout"
23     app:layout_constraintTop_toBottomOf="@id/textView1"
24     tools:layout_editor_absoluteX="64dp" />
25 </androidx.constraintlayout.widget.ConstraintLayout>
```

