

Course Design B Report

Title of the Project: Login Registration System



Major :	Software Engineering
Class:	3rd Year 2019 Batch
Name:	HOSEN ARAFAT
ID No :	199076003

School of Computer Science and Technology

Date: Sunday, May 15, 2022

Hudong road 59, Ma'anshan, Anhui, China, 243002

CONTENTS

I. Environment configuration.....	8
1. Download and install Android studio (The whole process needs network).....	8
1.1. Open the official website of Android studio, Click DOWNLOAD ANDROID STUDIO to download. (https://developer.android.google.cn/studio/)	8
1.2. Confirm Download.....	9
1.3. Run the installation package (in administrator mode).....	9
1.4. Always click 'next' to install.....	10
1.5. Choose Components “Android” and “Android Virtual Device” Then Click Next.....	11
1.6. Set Configuration Settings Install Locations. Then Click Next.....	12
1.7. Choose Start Menu Folder of the Android Studio shortcuts. Then Click Next.....	13
1.8. Android Studio being Installing. Then Click Next.....	14
1.9. Installation Complete. Setup was Successfully completed. Then Click Next.....	15
1.10. Select standard mode to install, download the components (about 10 minutes), and keep clicking next.....	16
1.11. Select Start Android studio. Click finish.....	17
2. Create a project.....	18
2.1. Open Android studio and click create new project to create the project.....	18
2.2. select Phone and Tablet. Then select basic activity and click next.....	18
2.3. Fill in the project name, save location, select language Java, minimum SDK API 19: Android 4.4 (kitkat), and click Finish.....	19

2.4. The project has been created. You need to wait for a while to install gradle (about 10 minutes). At this time, you can convert the structure type of the directory to ‘project’.....	19
II .Development design.....	20
1. Sing up interface layout	20
1.1. Add background gradient and background color.....	20
1.2. Add the top gradient in front and put into the drawable folder: app->src->main->res->drawable->top_gradient.xml.....	22
1.3. Now add my own picture in the front . put the picture into the drawable folder. Add a statement to ImageView code android:src="@drawable/login" /> Open the layout file in the following: app->src->main->res->drawable->login.png->activity_main_xml.....	24
1.4. Now add to “CODING WITH ARAFAT” and “SING UP TO LEARN QUICKLY” text below in the picture. as shown in the figure below.....	25
1.5. Add “User name” input box in the layout. as shown in the figure below.....	27
1.6. Add “Email ID” input box in the layout. as shown in the figure below.....	28
1.7. Add “Phone No” input box in the layout. as shown in the figure below.....	30
1.8. Add “Password” input box in the layout. as shown in the figure below.....	31
1.9. Add “Sing up” button in the layout . Sing up text, text color, and button color.....	33
1.10. Add “I am already a member” text below in the sing up button. as shown in the figure below.....	35
2. Sing in interface layout	36
2.1. Add background gradient and background color. as shown in the figure below.....	36
2.2. Add the top gradient in front and put into the drawable folder. as shown in the figure below.....	38
2.3. Now add my own picture in the front . put the picture into the drawable folder. Add a statement to ImageView code android:src="@drawable/login" />.	
as shown in the figure below.....	40

2.4. Now add to “WELCOME BACK” and “SING IN TO LEARN QUICKLY” text below in the picture. as shown in the figure below.....	42
2.5. Add “Email ID” input box in the layout. as shown in the figure below.....	43
2.6. Add “Password” input box in the layout. as shown in the figure below.....	45
2.7. Add “Sing In” button in the layout . Sing up text, text color, and button color.....	46
2.8. Add “New Here? Sing up instead” text below in the sing In button. as shown in the figure below.....	48
 3.Implement the login process.....	50
3.1. Open the folder app > SRC > main > java > com > example > CODE WITH ARAFAT Singup From, delete all the files in the folder, and right-click to create MainActivity.java File, start writing login Logic. as shown in the figure below.....	50
3.2. Write the framework code of the program, and fill in the layout name of the login interface in the setcontentview function (default is activity _main). as shown in the figure below.....	51
3.3. Write the response function (listening function) corresponding to the login button and the function to obtain the information in the input box. as shown in the figure below.....	52
3.4. At this time, click the icon  in the upper right	

corner to run the program, and the virtual machine will pop up. as shown in the figure below.....	54
3.5. After clicking the button on the virtual machine, the console will display the contents of the input box. as shown in the figure below.....	55
4.Pack into.Apk file and run on your phone.....	56
4.1. Open the build.gradle File, which can modify the version number, name and other information of APK. as shown in the figure below.....	56
4.2. Click 'Build->Generate Signed APK'.....	56
4.3. Select APK after opening.....	57
4.4. Click the Button' create new...'	58
4.5. Create a key file with suffix.JKS.....	59
4.6. Fill in other information.....	61
4.7. Click next, select Build Variants,Signature Version and then click finish.....	62
4.8. Click app > debug to find the. APK file and copy it to the phone.....	63
4.9. Download and install the program on your mobile.....	64
4.10. My first app is finished!.....	66
III. building Mysql database environment.....	67
1. Complete the basic user authentication.....	67
1.1. New login success page.....	67
1.2. Modify the login Activity_welcome.xml page, and modify the code to display "Welcome by ARAFAT (199076003)" . as shown in the figure below.....	68
1.3.In and MainActivity.java Create in the same folder welcomeactivity.java File, and write code,	

guide the activity_welcome.xml Page written. as shown in the figure below.....	72
1.4.stay MainActivity.java When I write user name, valid email id, valid phone no, and password must contain at least 5 to 20 characters. Then click sing up button. Show the “successful!” in the below. When i click Sing up button again show the “user already exists”. judge whether the user name= Yafei, email id=yafei2023@gmail.com, phone no= 24487654567, and password= Yafei23Ahut . as shown in the figure below.....	73
1.5. Test with virtual machine, when the Email id and password are Email id= yafei2023@gmail.com and password= Yafei23Ahut. Then click the login button, then immediately jump to the activity welcome page	75
IV. Configure Login and registration to database using PHP and MySQL :	76
1. Create database.....	76
1.1.Download and install XAMPP (The whole process needs network).....	76
1.1.1. Open the website of XAMPP , Click DOWNLOAD XAMPP to download. (https://www.apachefriends.org/download.html).....	76
1.1.2. Run the installation package (in administrator mode).....	77
1.1.3. Always click ‘next’ to install.....	77
1.1.4. Select all Components “ XAMPP ” Then Click Next.....	78
1.1.5. Choose Installation folder of the XAMPP . Then Click Next.....	79
1.1.6. Set Language of the XAMPP . Then Click Next.....	80
1.1.7. Setup is now Ready to being installing XAMPP of my computer. Then Click Next.....	81
1.1.8. Installing XAMPP , Unpacking files.....	82

1.1.9. Now finish the XAMPP installing. Click finish.....	83
1.1.10. Now open XAMPP control panel.....	83
1.1.11. Start Apache.....	84
1.1.12. Start MySQL.....	84
1.1.13. Click Admin then go to “phpMyAdmin”	85
1.1.14. Now Create a Database Folder “singupdb” show query box. and create table users (name, email, mobile, password). as shown in the figure below.....	86
2. Add insert function.....	88
2.1.Create a loginphp folder inside of the htdocs folder. And create loginin.php and signup.php inside of this folder. And write code for loginin.php and signup.php.....	88
3.connect to database using IP address :	89
3.1. Now connect the Android app with database using my local server, I mean by using our ip address we can connect out android app with Signin.php and signup.php to database.....	90
4. Testing.....	90
4.1. Login interface.....	90
4.2. Registration interface.....	91
V. curriculum requirements.....	92
1.Project Name: login registration system:	92
Coedes:	93
login.php::	93
signup.php::	93
MainActivity.java::	94
activity_main.xml::	98
LoginActivity.java::	102
activity_login.xml::	105

activity_welcome.xml: 108

AndroidManifest.xml: 110

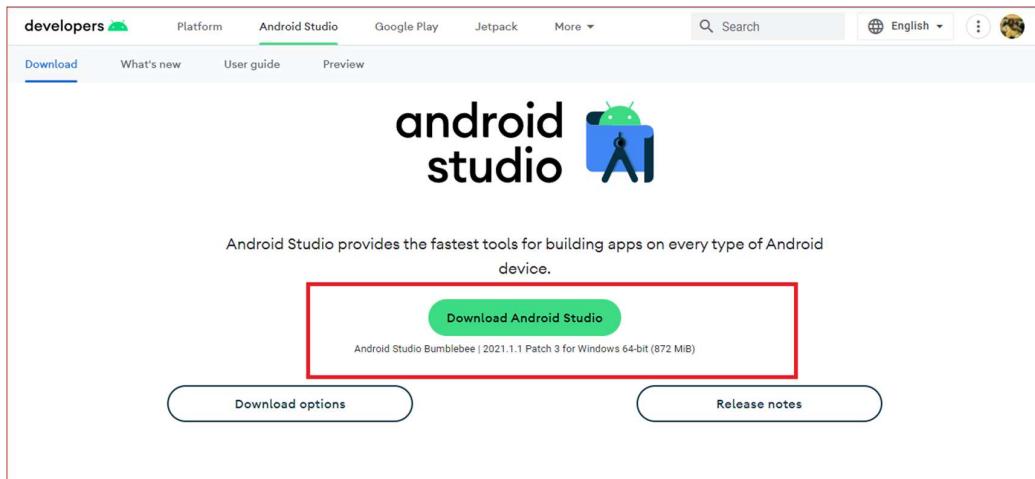
I .Environment configuration

1.Download and install Android studio (The whole process needs network)

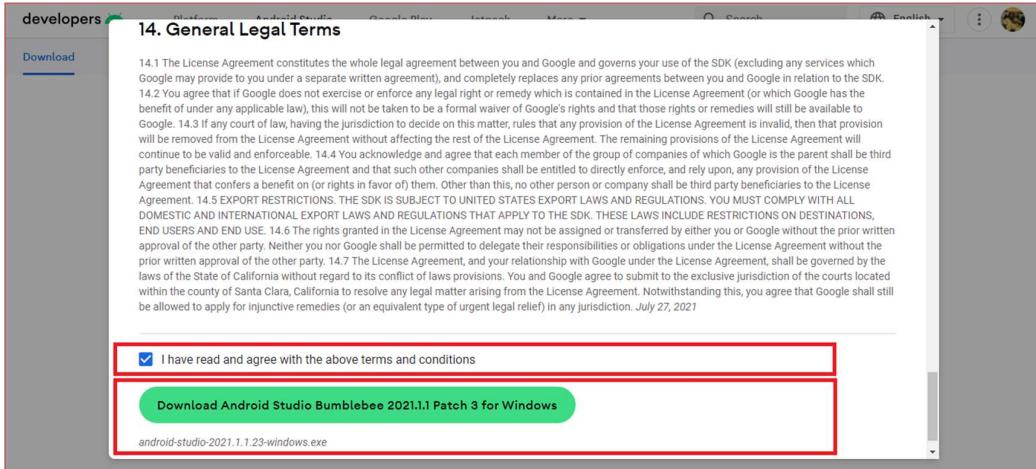
1.1. Open the official website of Android studio, Click DOWNLOAD

ANDROID STUDIO to download.

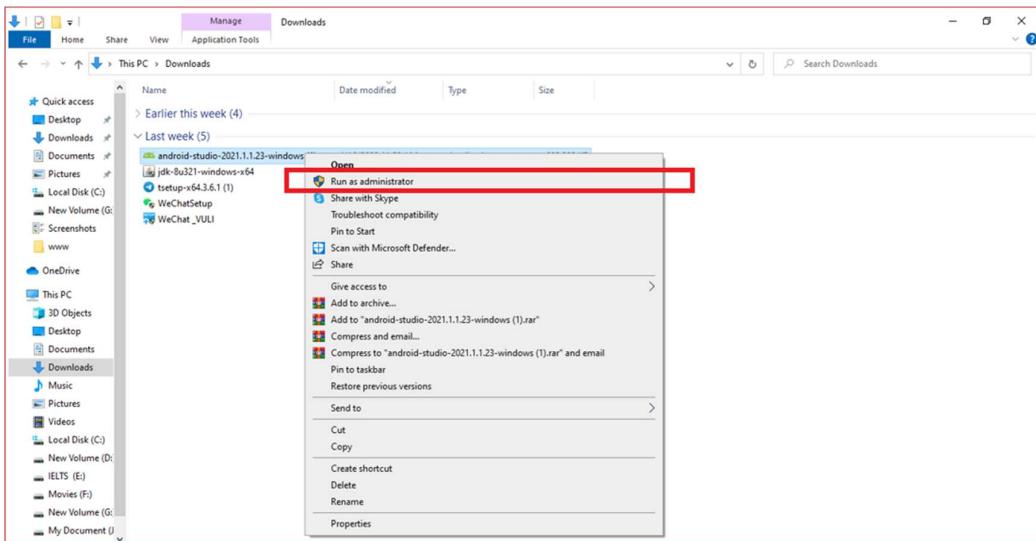
(<https://developer.android.google.cn/studio/>)



1.2. Confirm Download.



1.3. Run the installation package (in administrator mode).

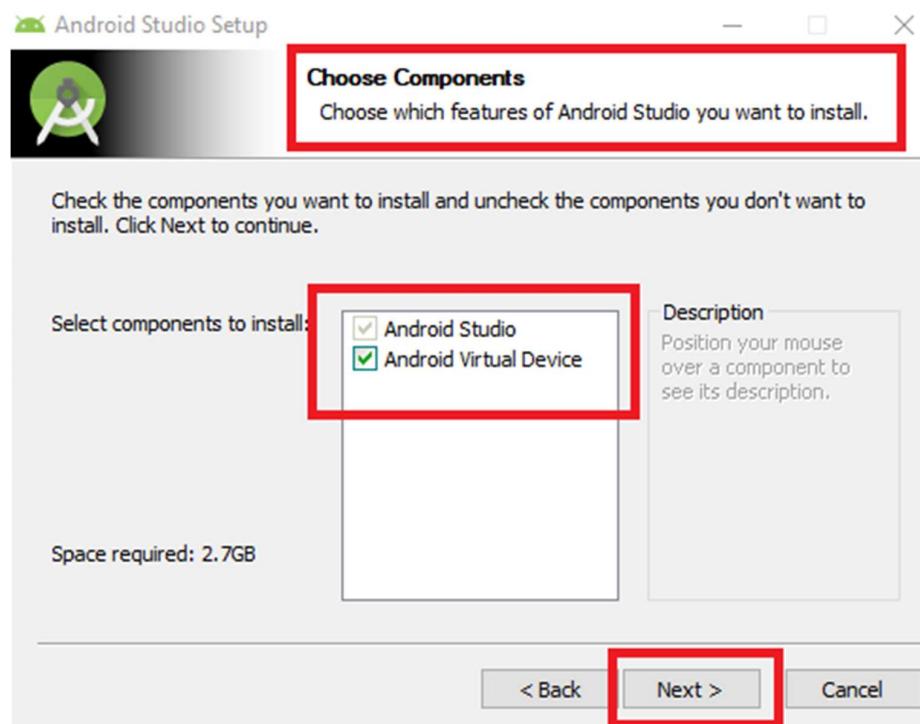


1.4. Always click 'next' to install.

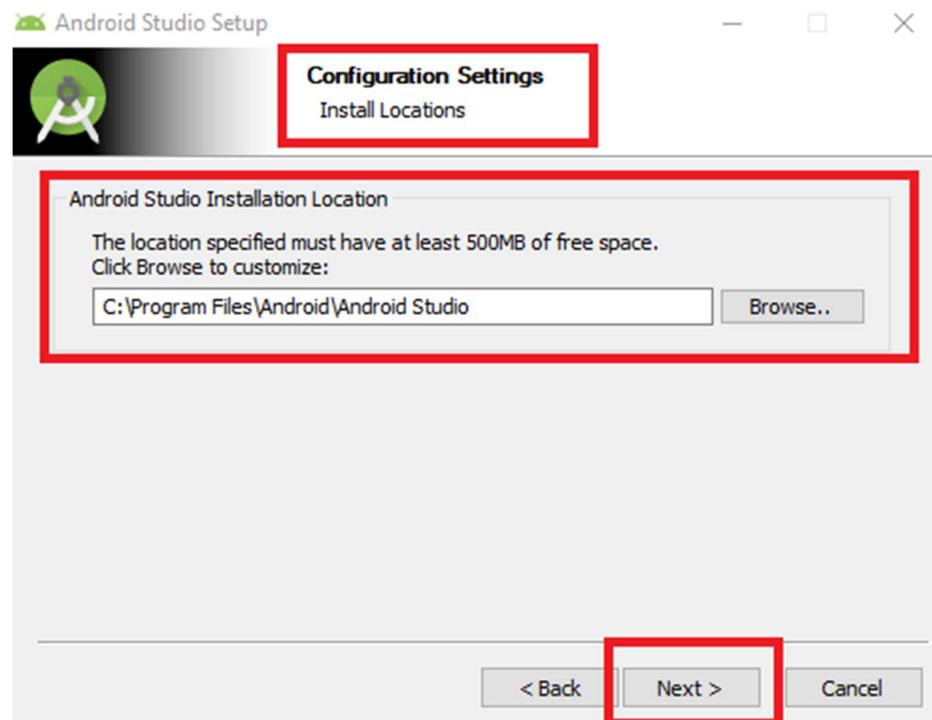


1.5. Choose Components “Android” and “Android Virtual Device” Then

Click Next.

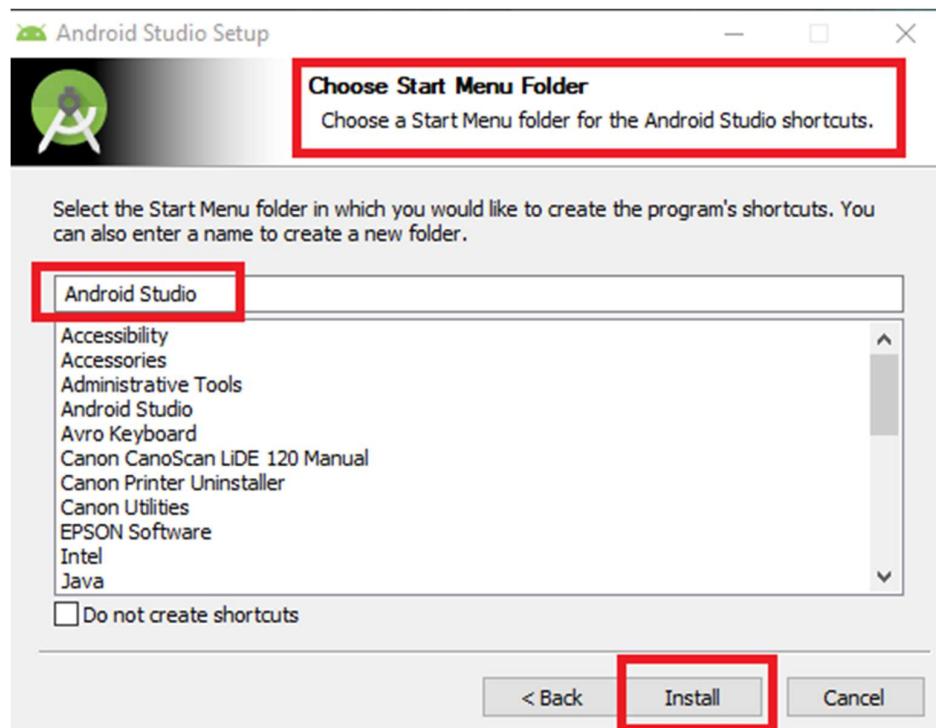


1.6. Set Configuration Settings Install Locations. Then Click Next.

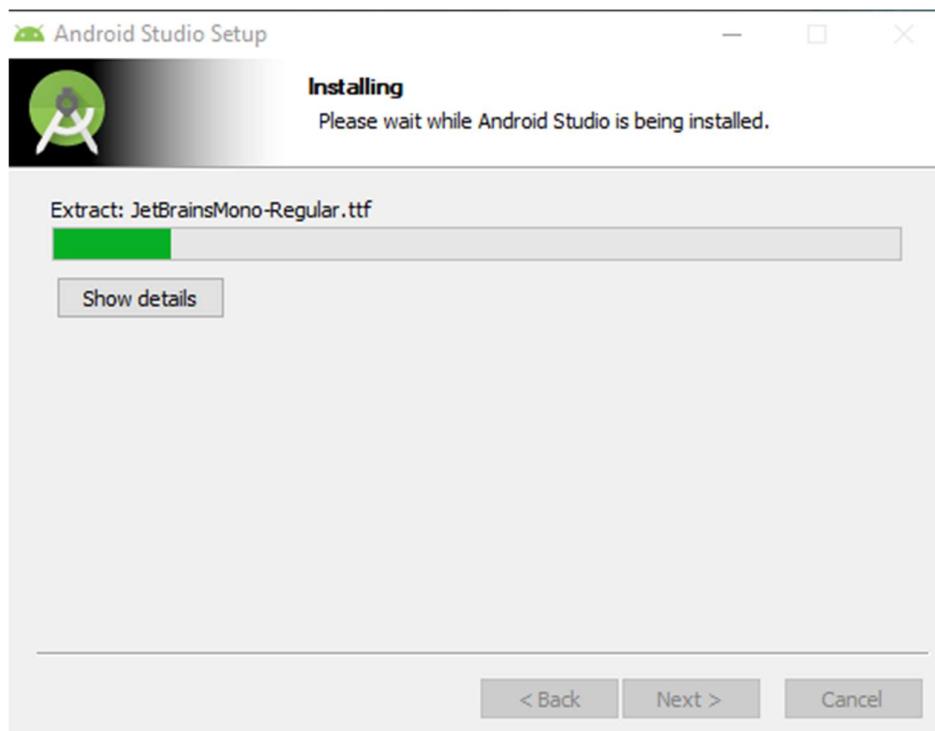


1.7. Choose Start Menu Folder of the Android Studio shortcuts. Then

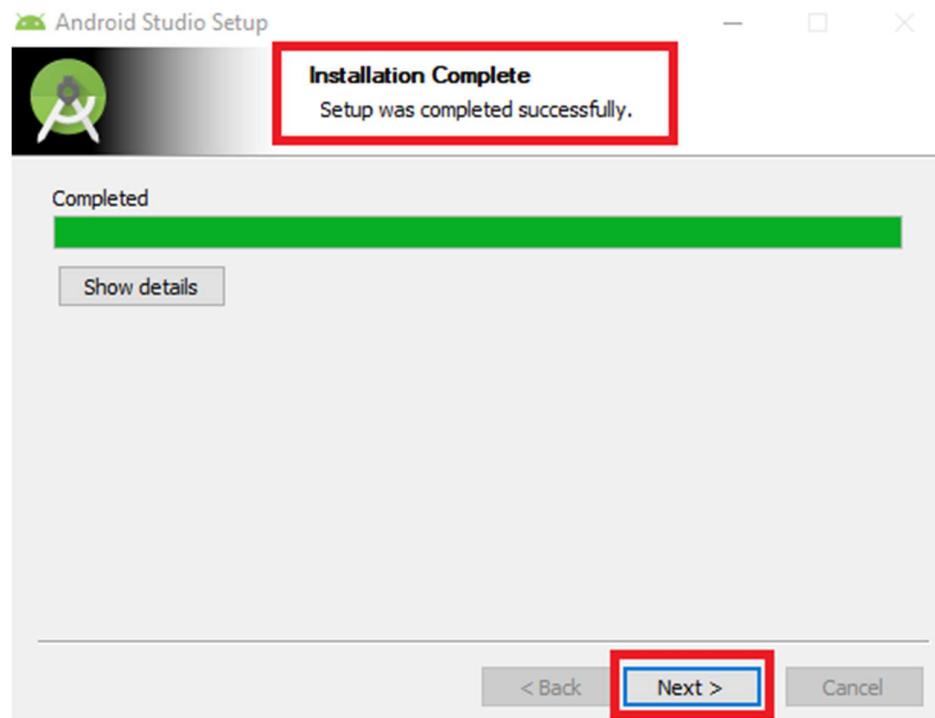
Click Next.



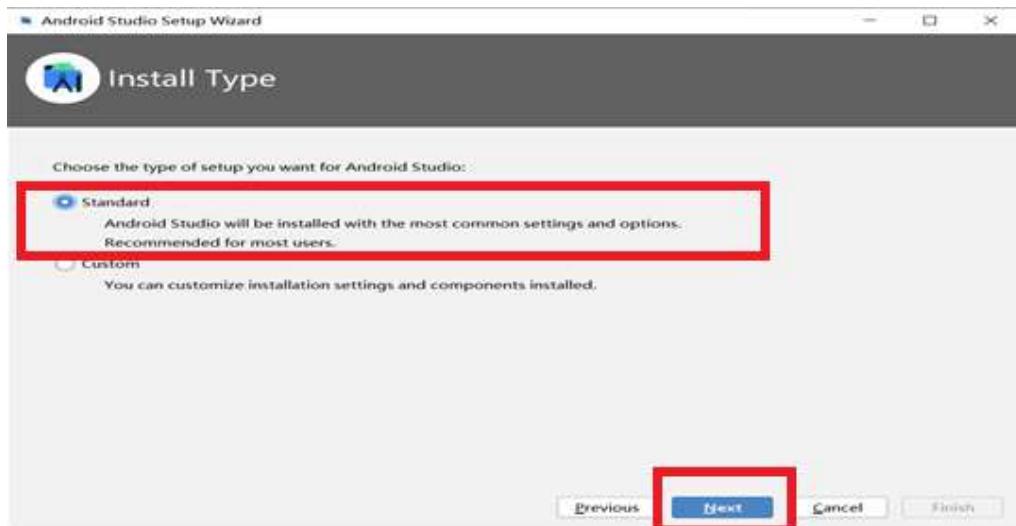
1.8. Android Studio being Installing. Then Click Next.



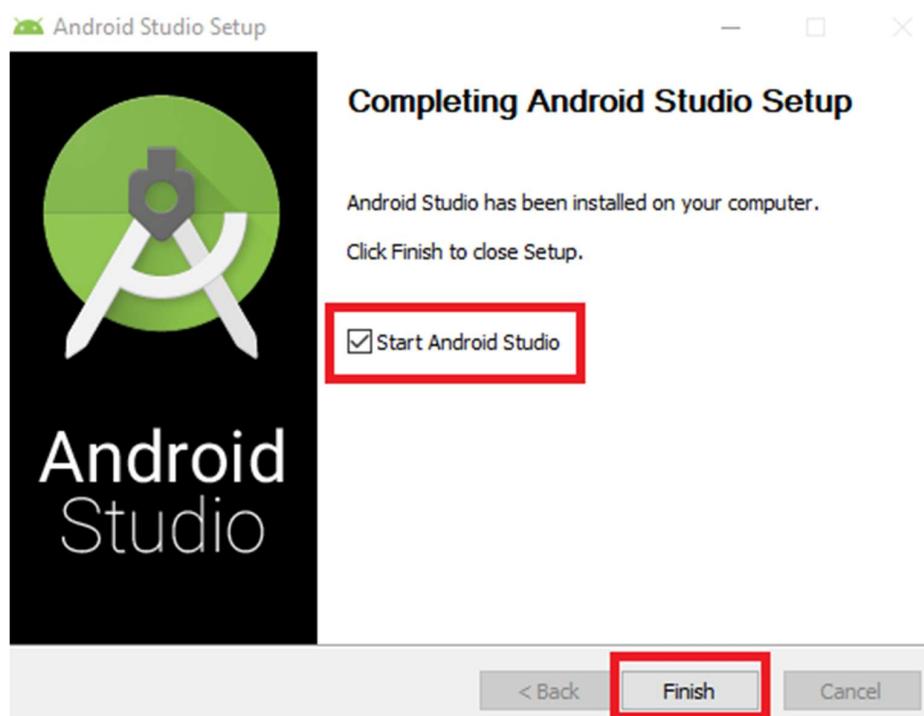
1.9. Installation Complete. Setup was Successfully completed. Then Click Next.



1.10. Select standard mode to install, download the components (about 10 minutes), and keep clicking next.

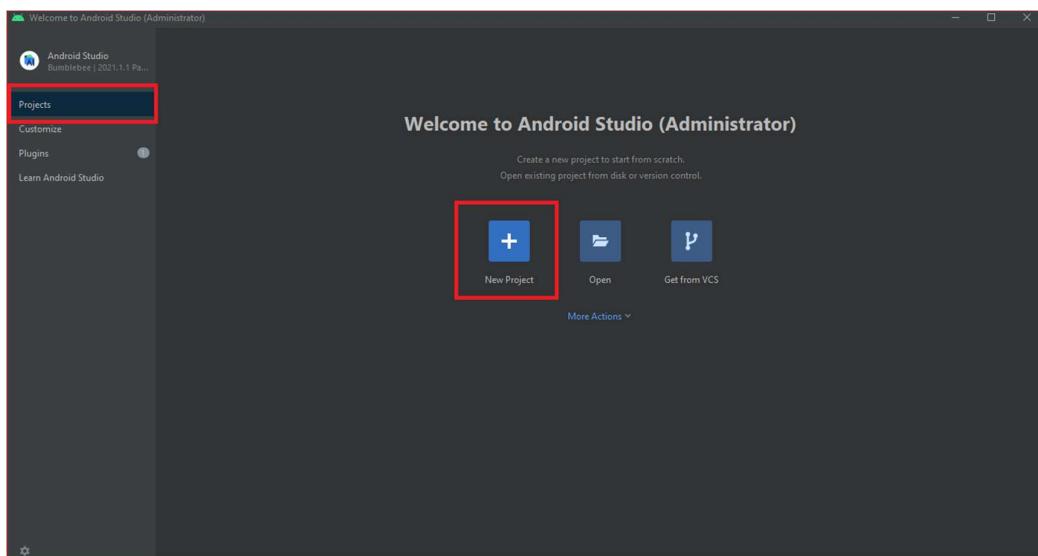


1.11. Select Start Android studio. Click finish.

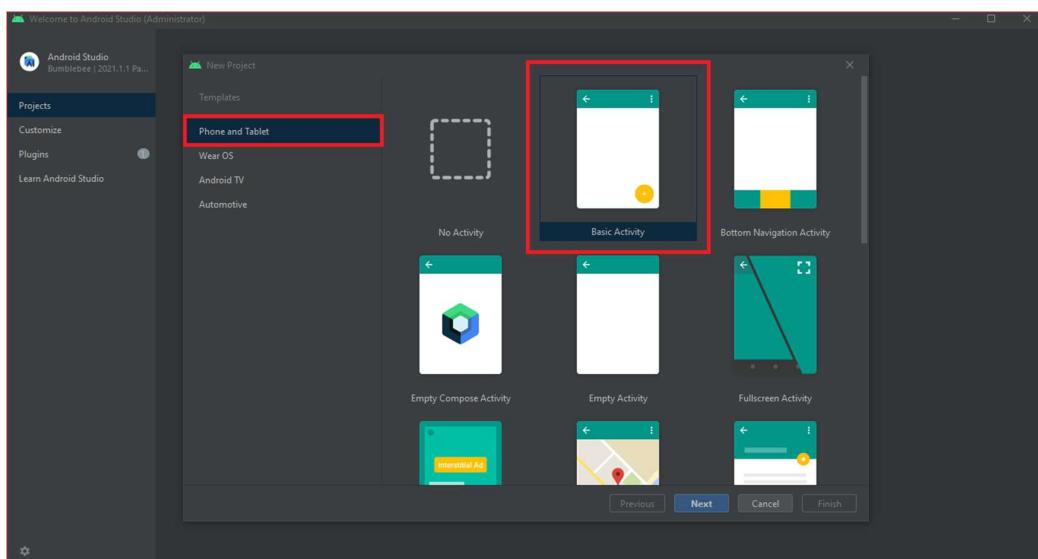


2. Create a project

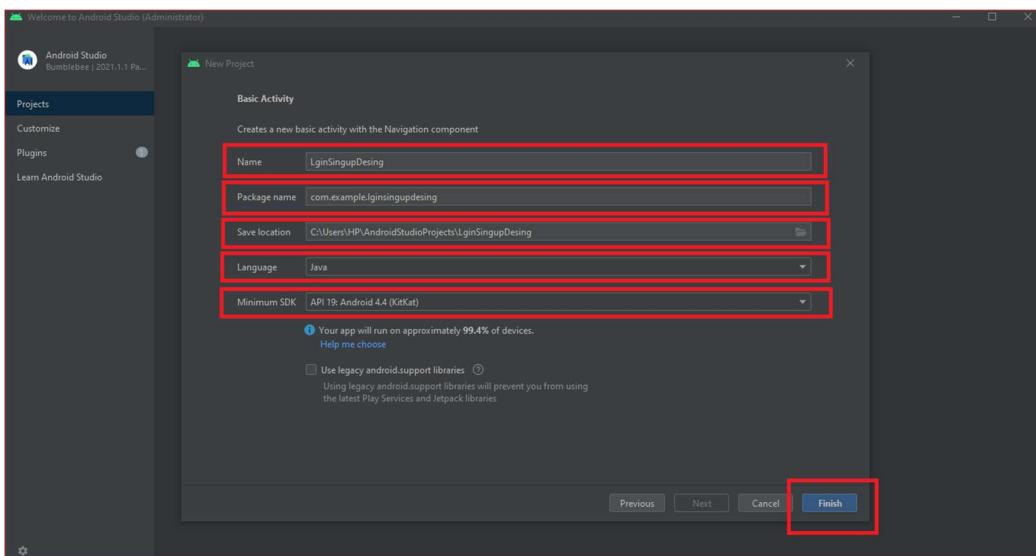
2.1. Open Android studio and click create new project to create the project.



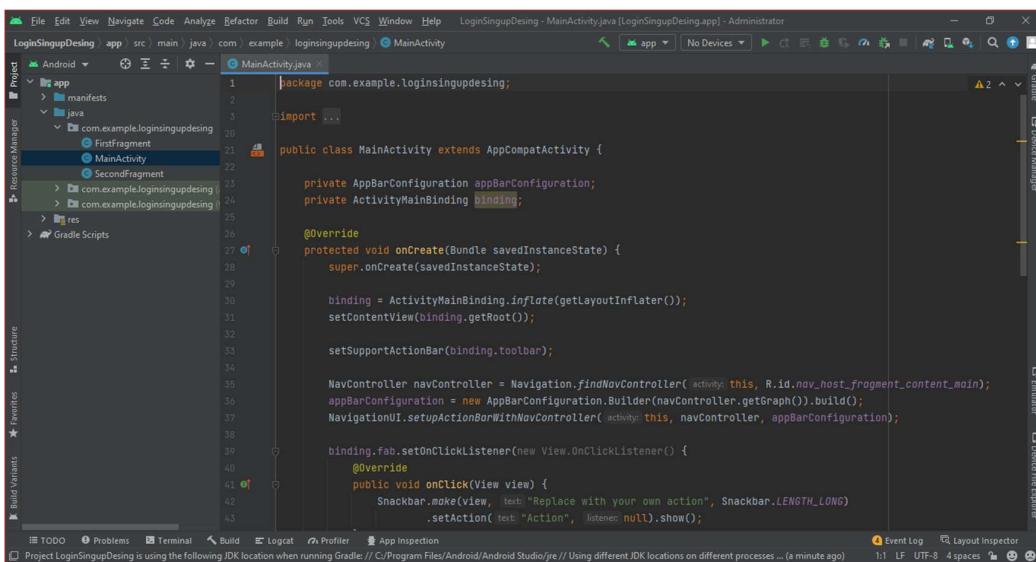
2.2. select Phone and Tablet. Then select basic activity and click next.



2.3. Fill in the project name, save location, select language Java, minimum SDK API 19: Android 4.4 (kitkat), and click Finish.



2.4. The project has been created. You need to wait for a while to install gradle (about 10 minutes). At this time, you can convert the structure type of the directory to 'project'.



II. Development design

1. Sing up interface layout

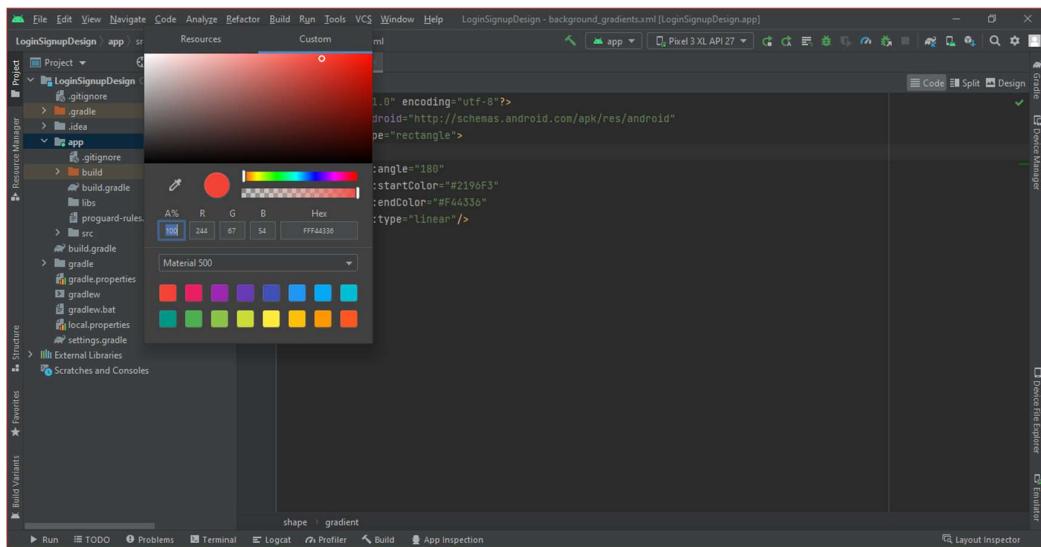
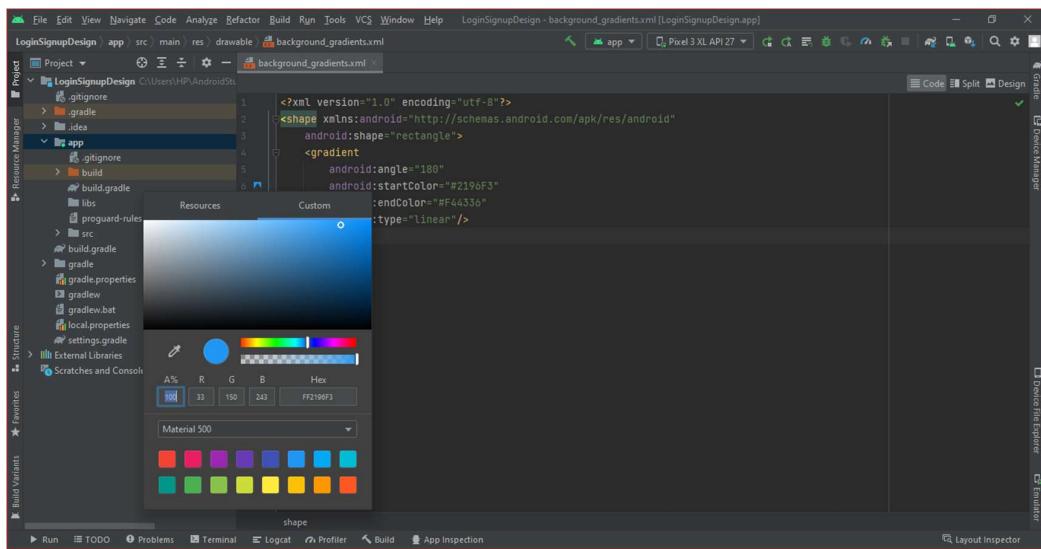
1.1. Add background gradient and background color.

The image shows two screenshots of the Android Studio interface. The top screenshot displays the `activity_main.xml` layout file in the XML editor. The XML code defines a `LinearLayout` with a gradient background:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:background="@drawable/screen_background"
    android:orientation="vertical"
    tools:context=".MainActivity"
    android:id="@+id/register.activity">
</LinearLayout>
```

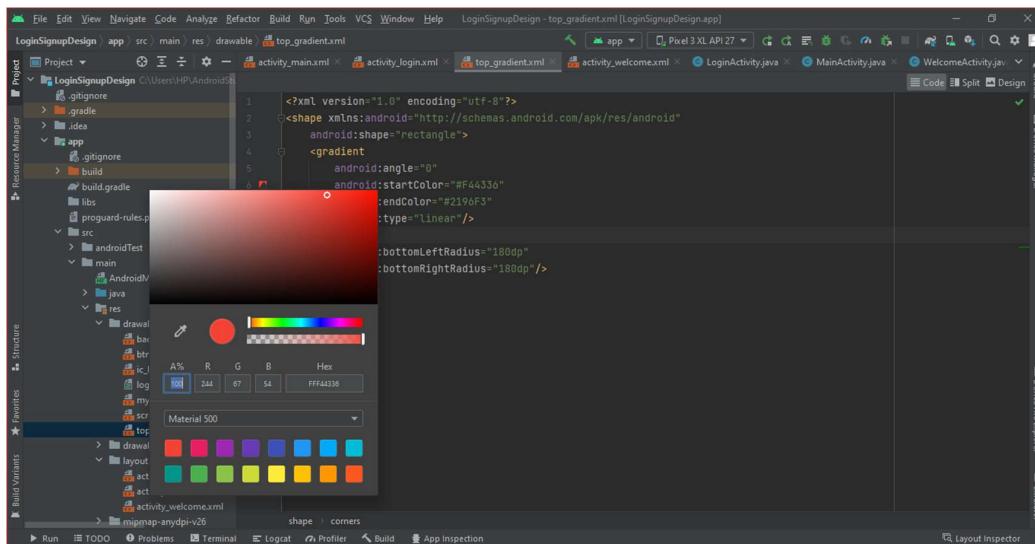
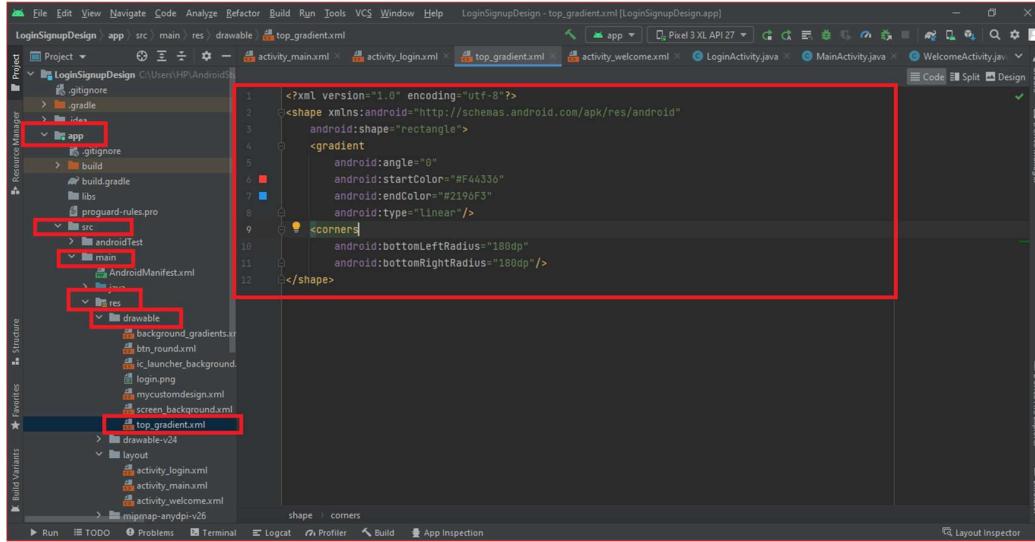
The bottom screenshot shows the `background_gradients.xml` drawable resource file in the XML editor. A red box highlights the `gradient` element, which specifies a linear gradient from blue to red:

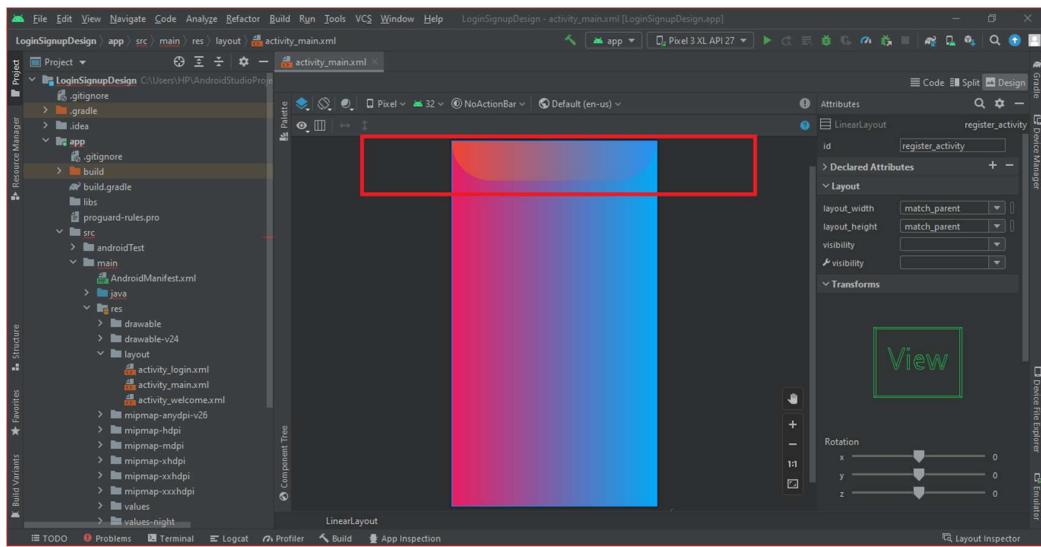
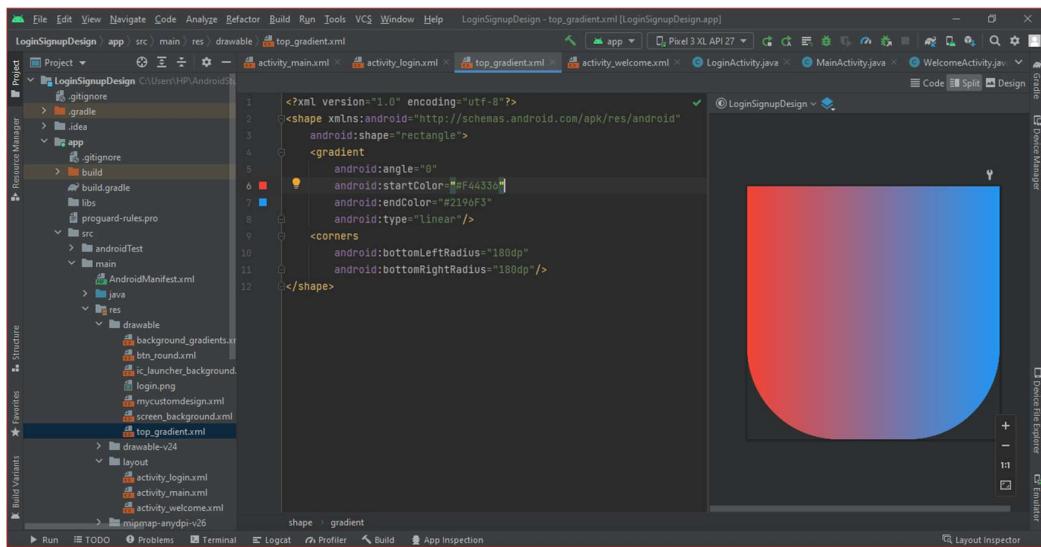
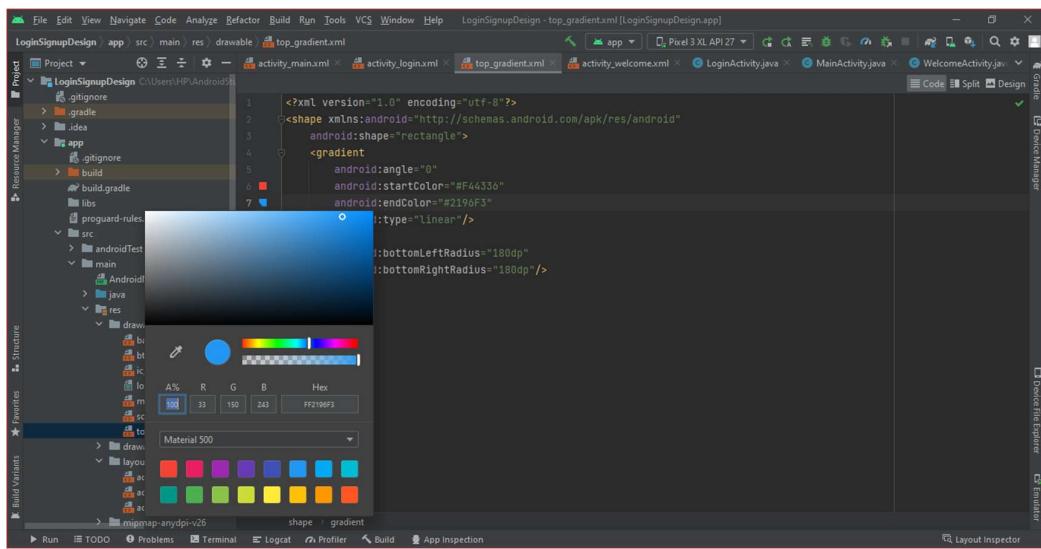
```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="rectangle">
    <gradient
        android:angle="180"
        android:startColor="#2196F3"
        android:endColor="#F44336"
        android:type="linear"/>
</shape>
```



1.2. Add the top gradient in front and put into the drawable folder:

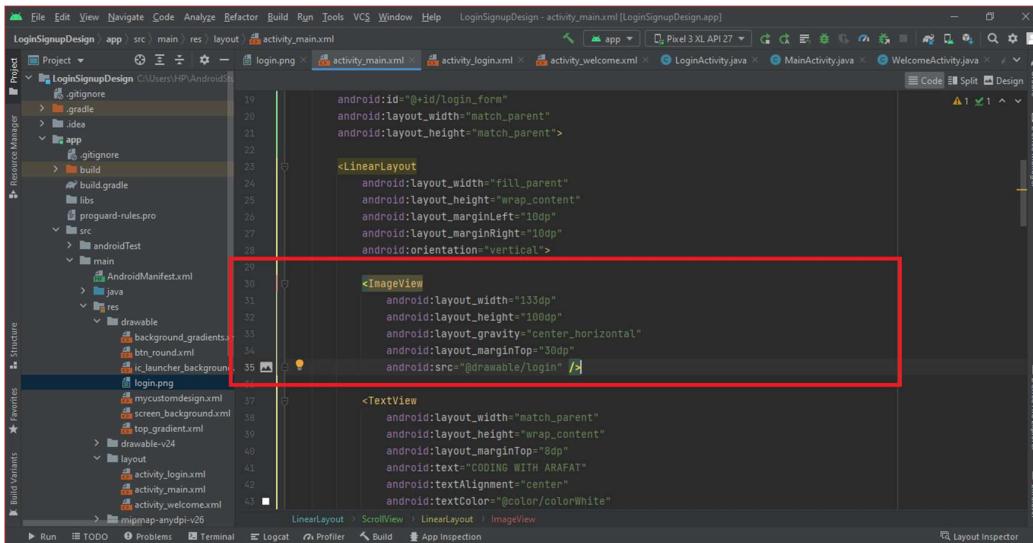
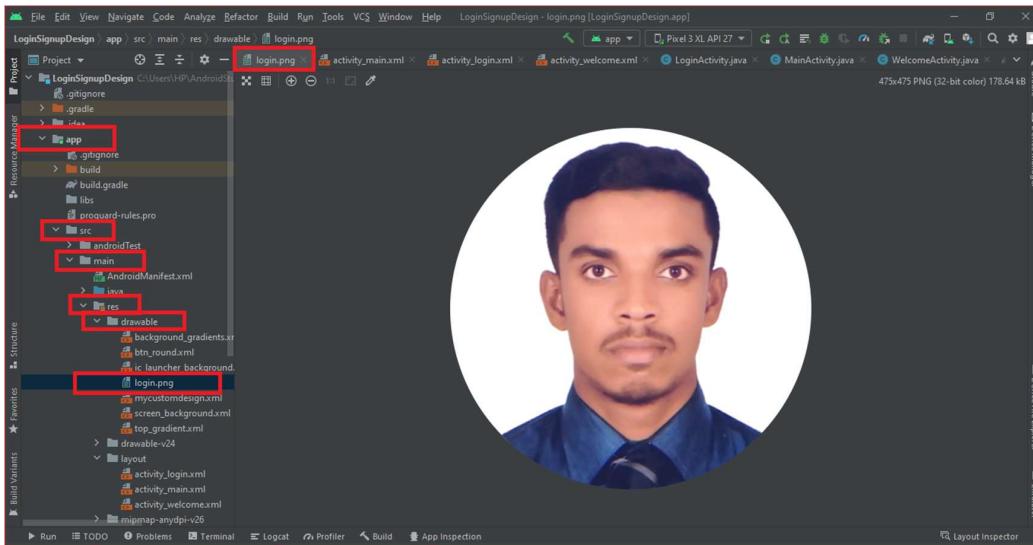
app->src->main->res->drawable->top_gradient.xml

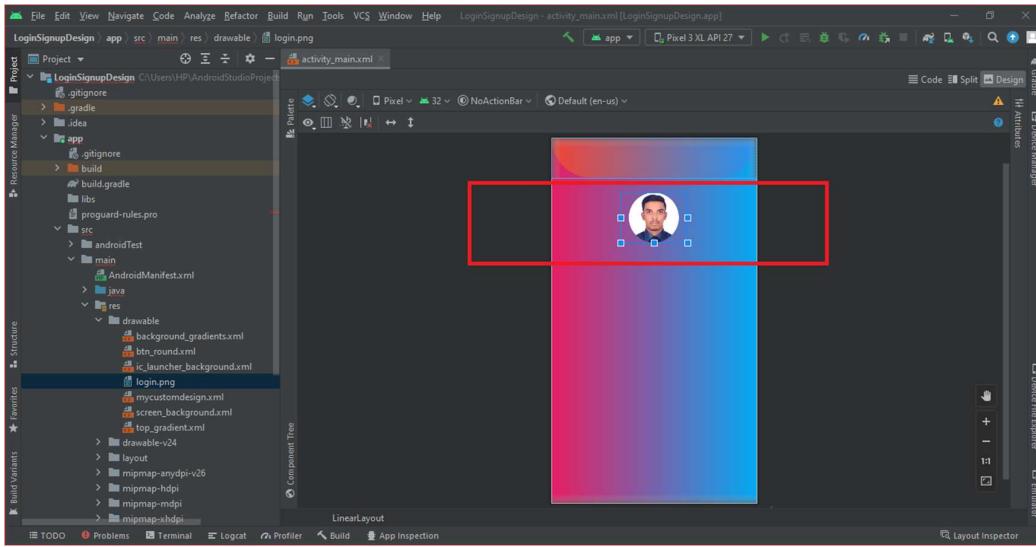




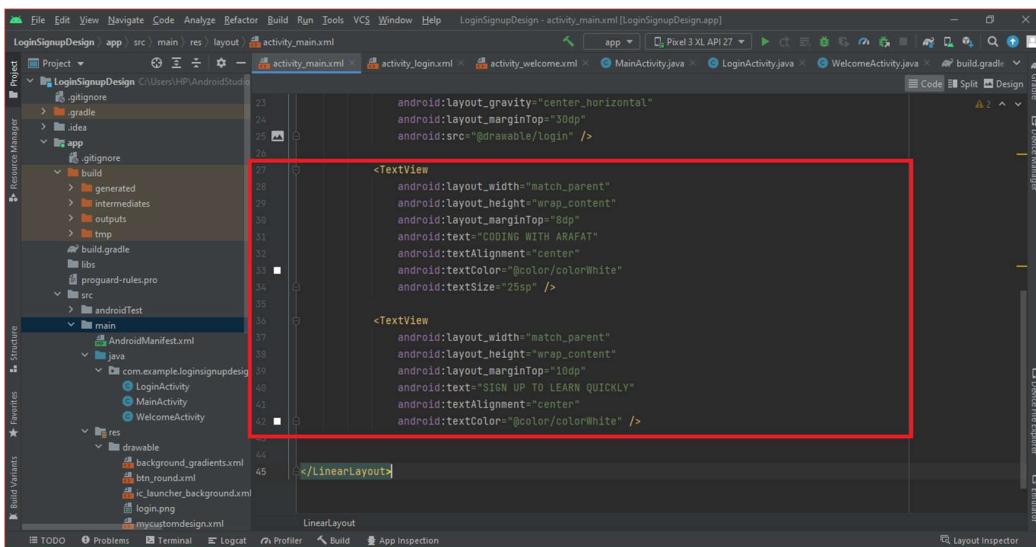
1.3. Now add my own picture in the front . put the picture into the drawable folder. Add a statement to ImageView code
`android:src="@drawable/login" />` Open the layout file in the following:

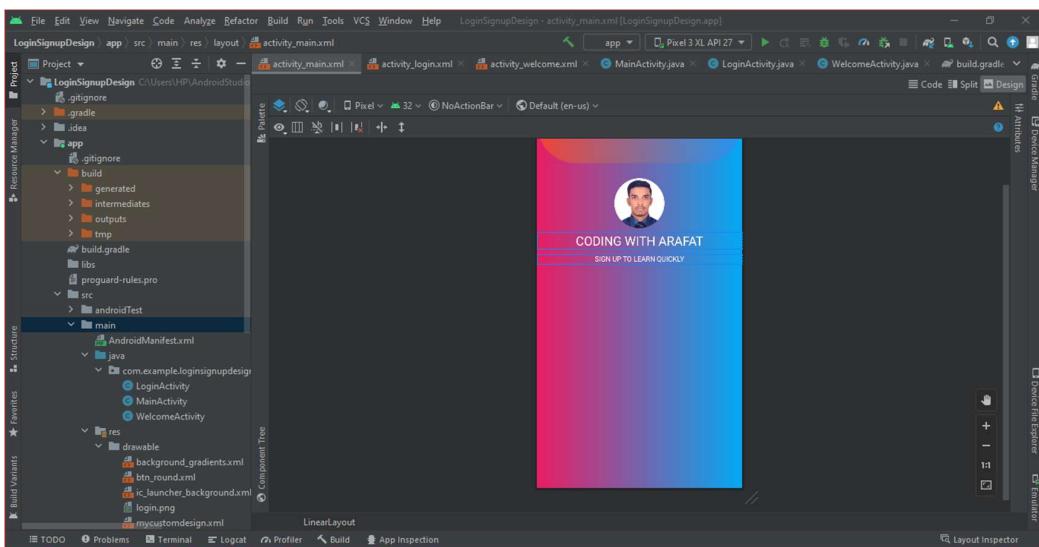
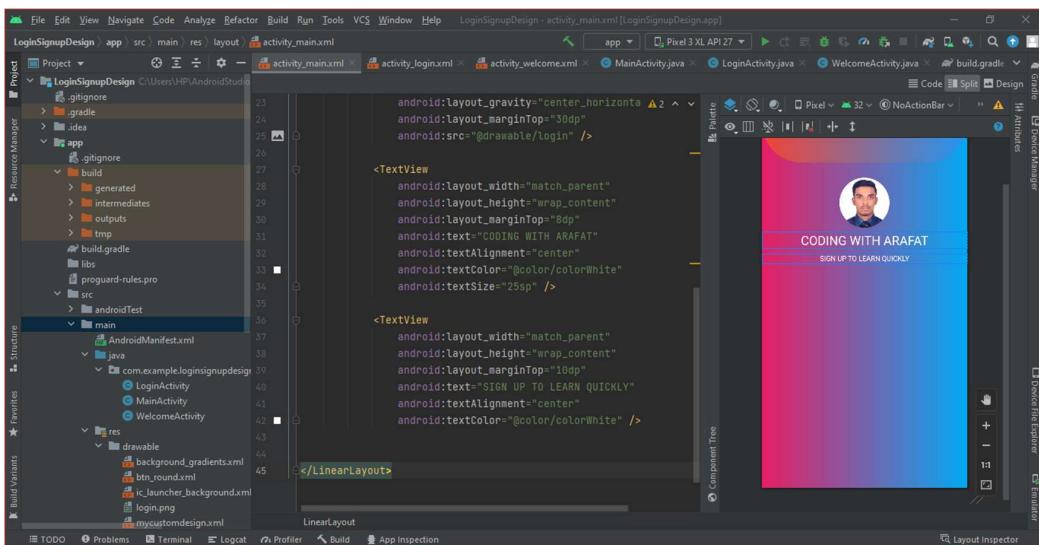
`app->src->main->res->drawable->login.png->activity_main.xml`





1.4. Now add to “CODING WITH ARAFAT” and “SING UP TO LEARN
QUICKLY” text below in the picture. as shown in the figure below.





1.5. Add “User name” input box in the layout. as shown in the figure below.

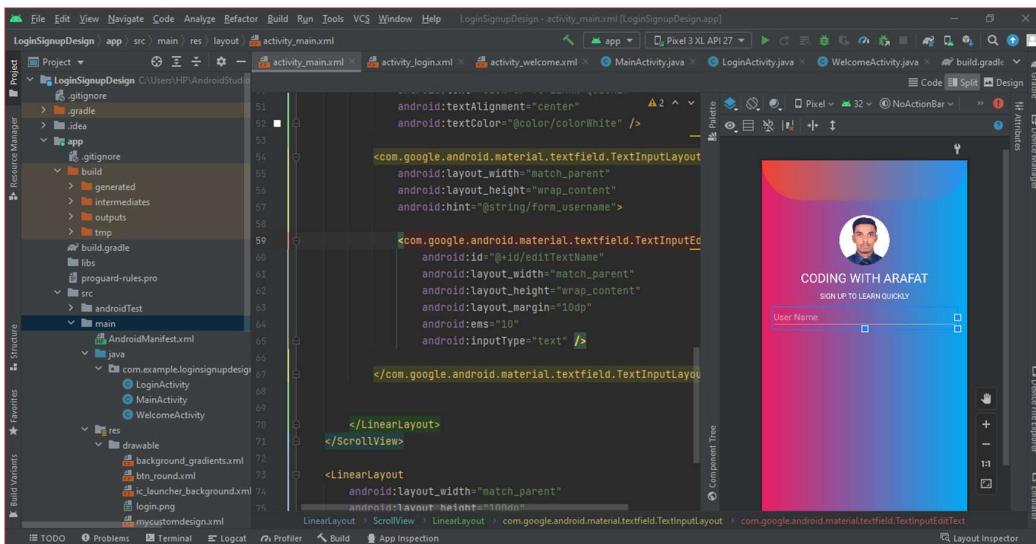
```

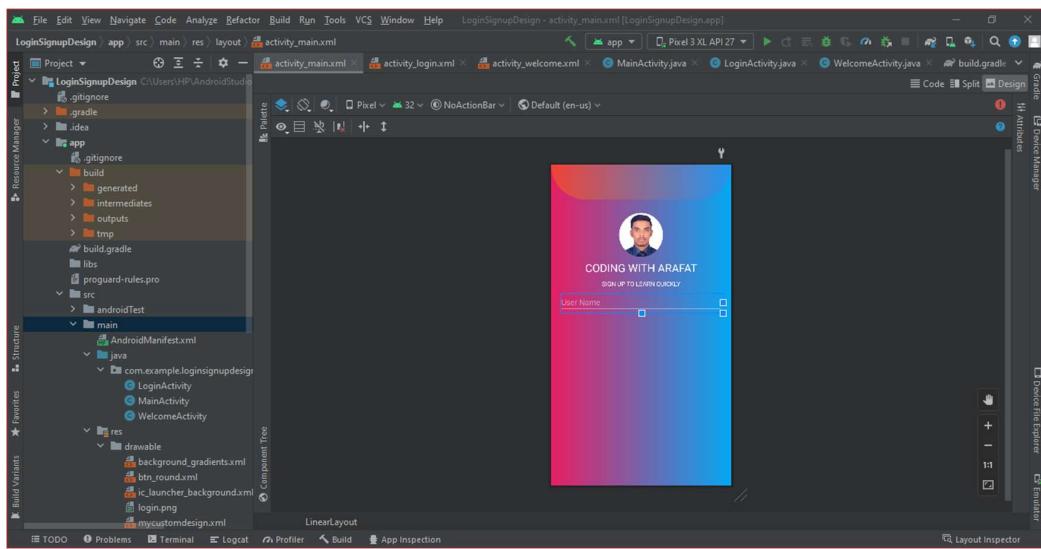
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:text="SIGN UP TO LEARN QUICKLY"
    android:textAlignment="center"
    android:textColor="@color/colorWhite" />

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="@string/form_username">

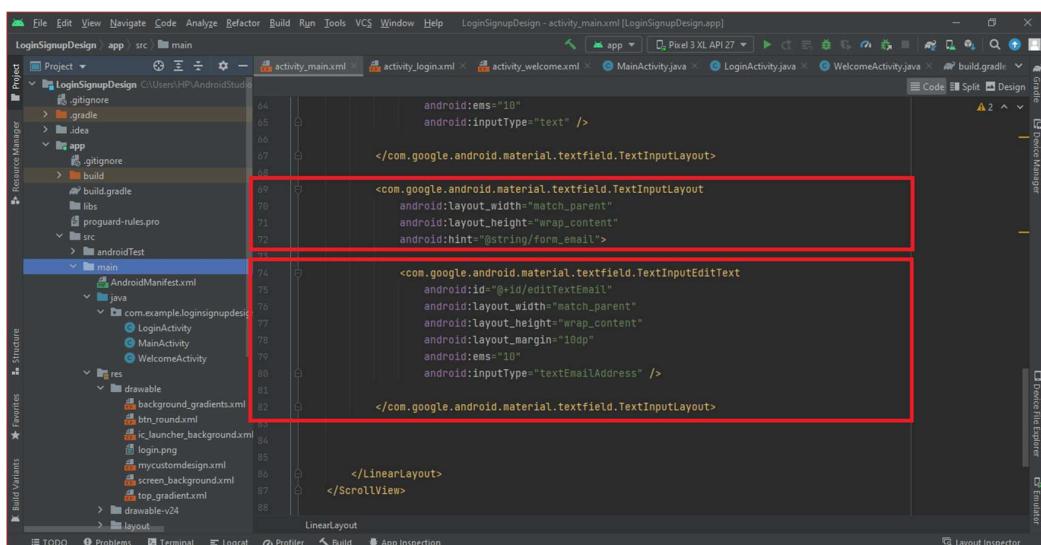
    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/editTextName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="10dp"
        android:ems="10"
        android:inputType="text" />
</com.google.android.material.textfield.TextInputLayout>

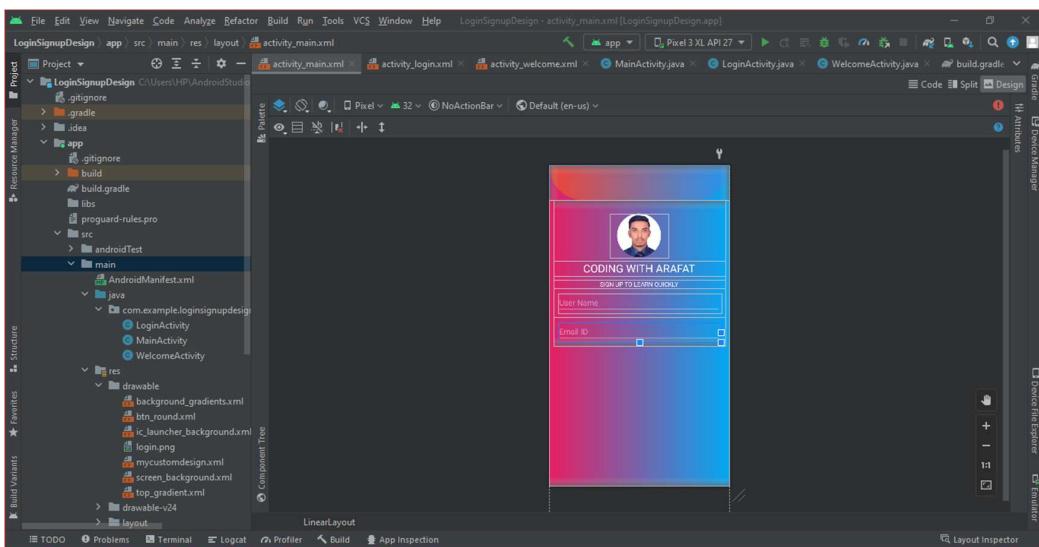
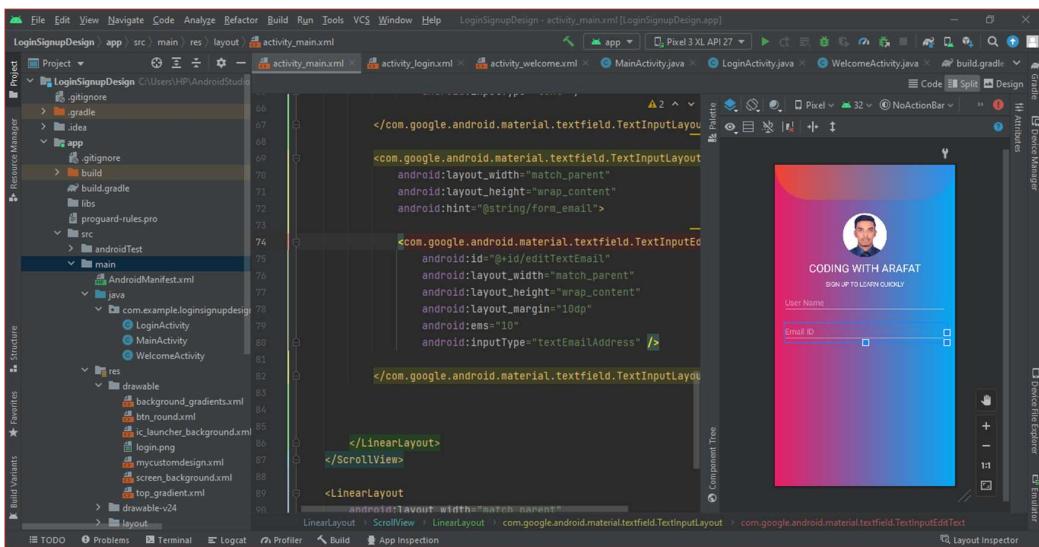
```



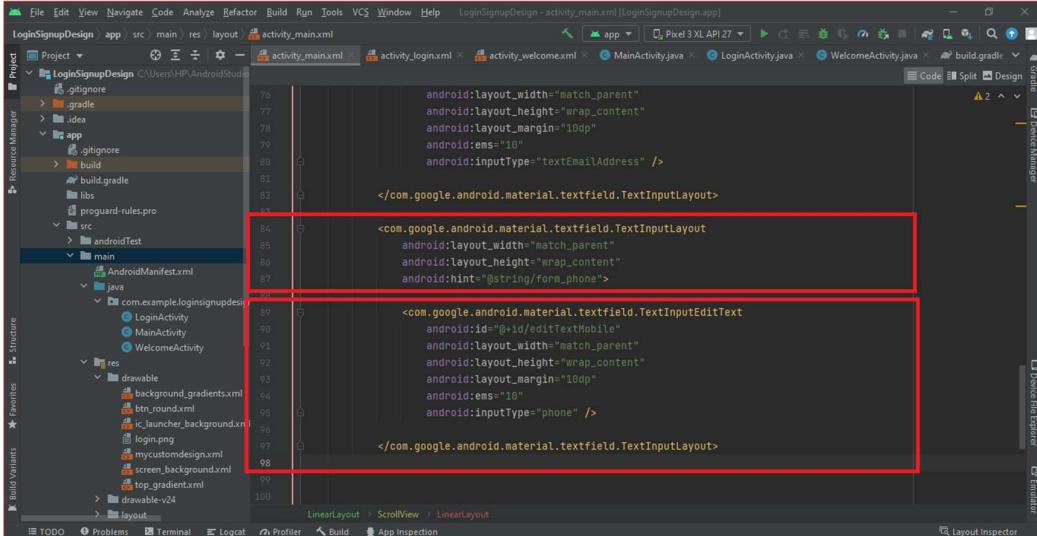


1.6. Add “Email ID” input box in the layout. as shown in the figure below.





1.7. Add “Phone No” input box in the layout. as shown in the figure below.



```

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:ems="10"
    android:inputType="textEmailAddress" />

</com.google.android.material.textfield.TextInputLayout>

```



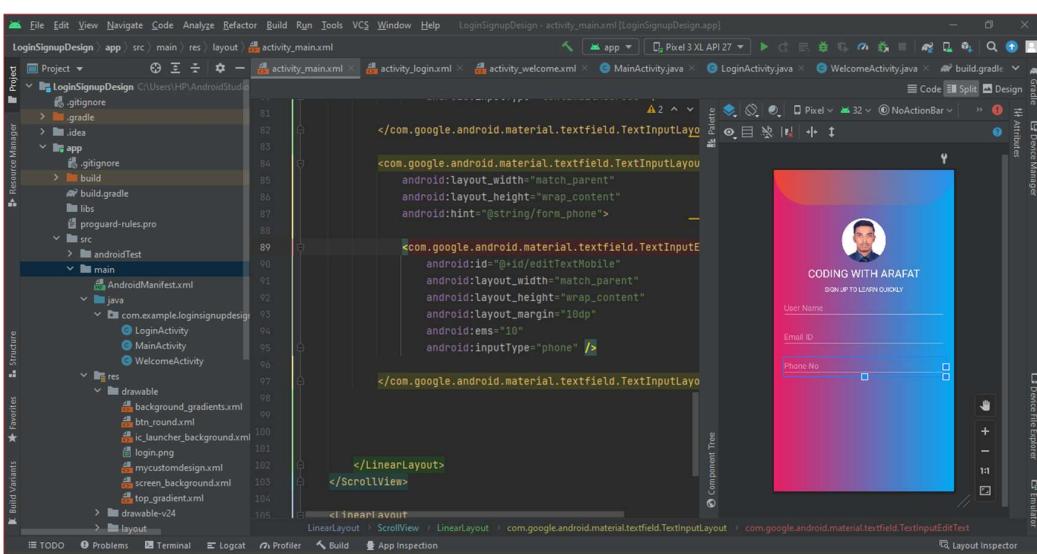
```

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="@string/form_phone">
```



```

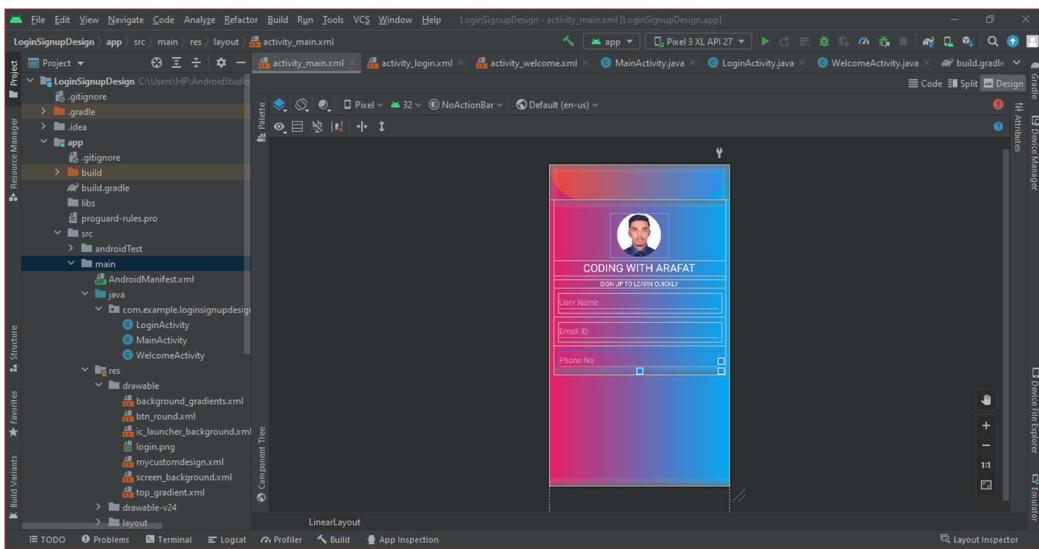
<com.google.android.material.textfield.TextInputEditText
    android:id="@+id/editTextMobile"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:ems="10"
    android:inputType="phone" />
```

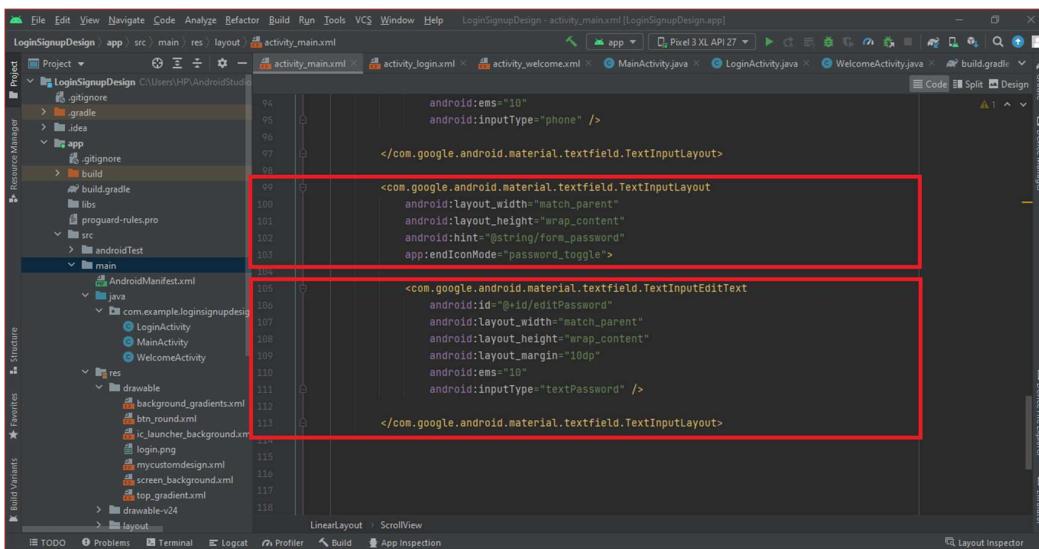
```

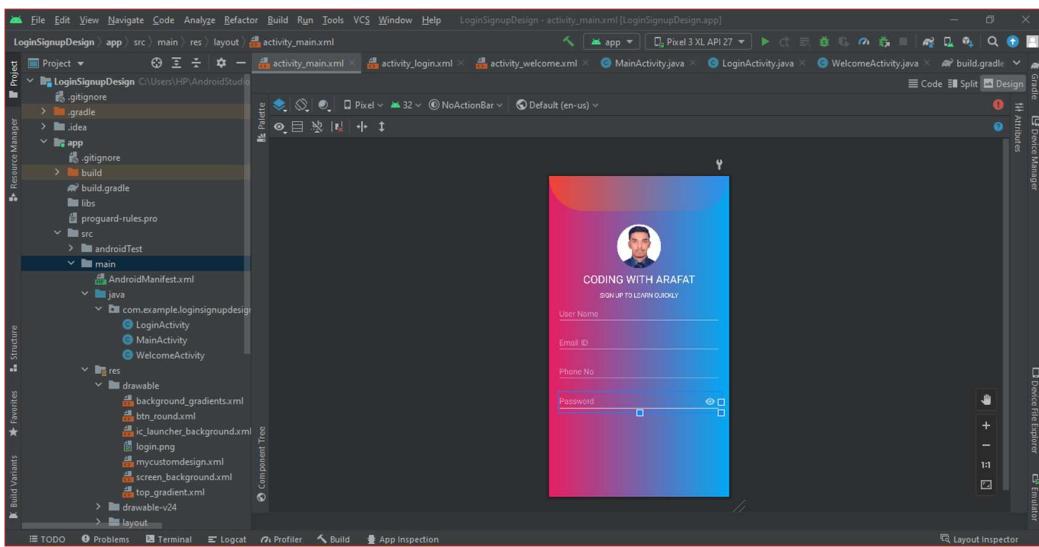
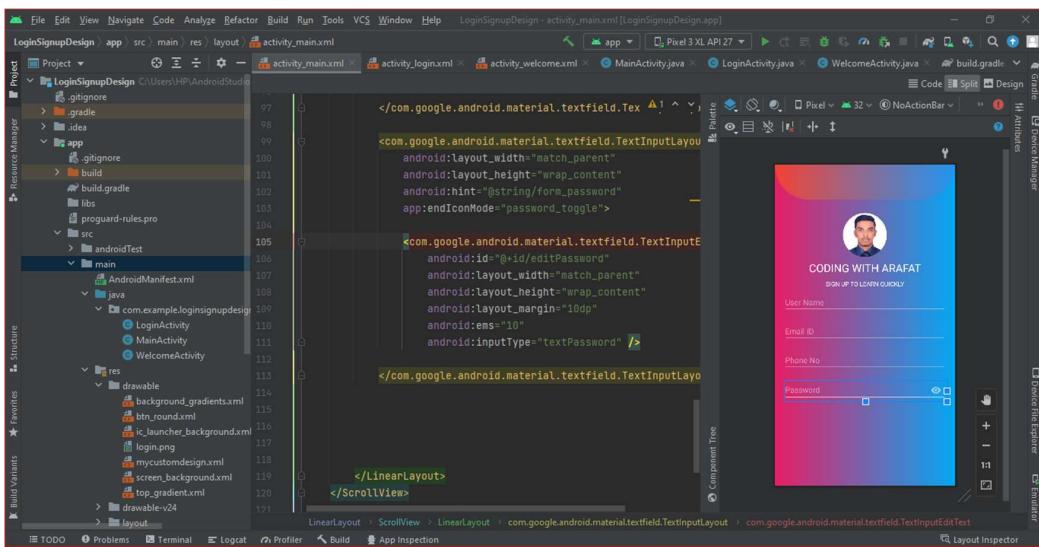
</com.google.android.material.textfield.TextInputLayout>

```

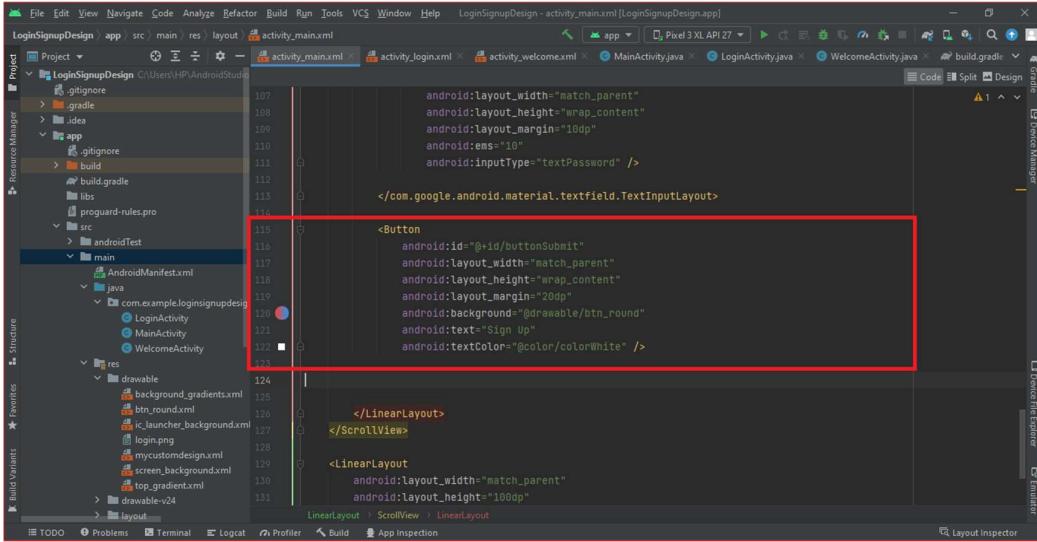


1.8. Add “Password” input box in the layout. as shown in the figure below.



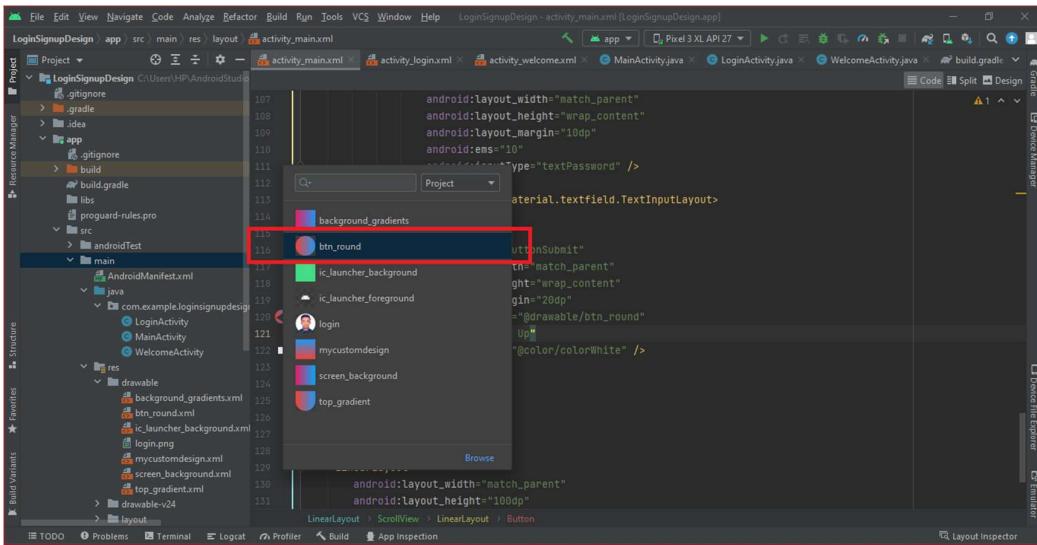


1.9. Add “Sing up” button in the layout . Sing up text, text color, and button color.



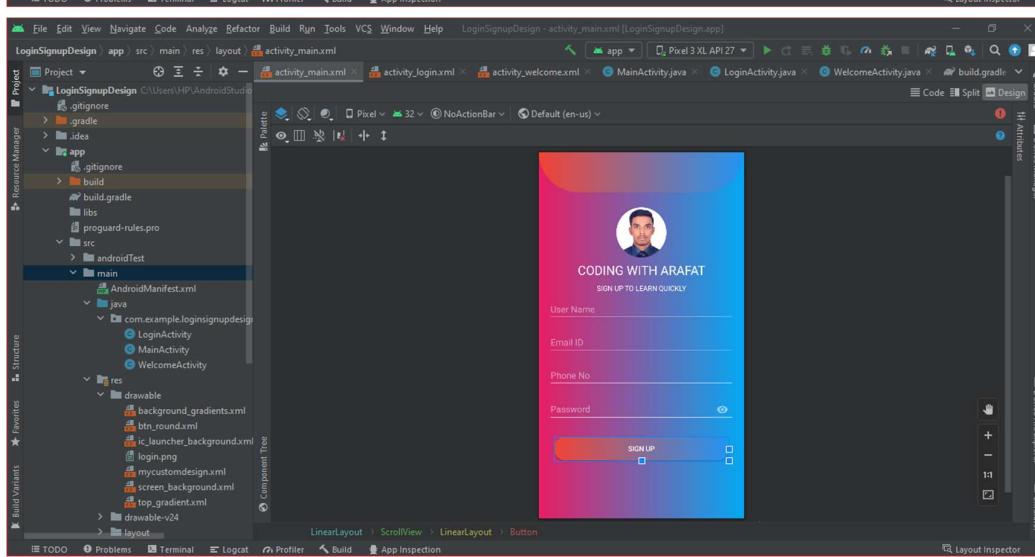
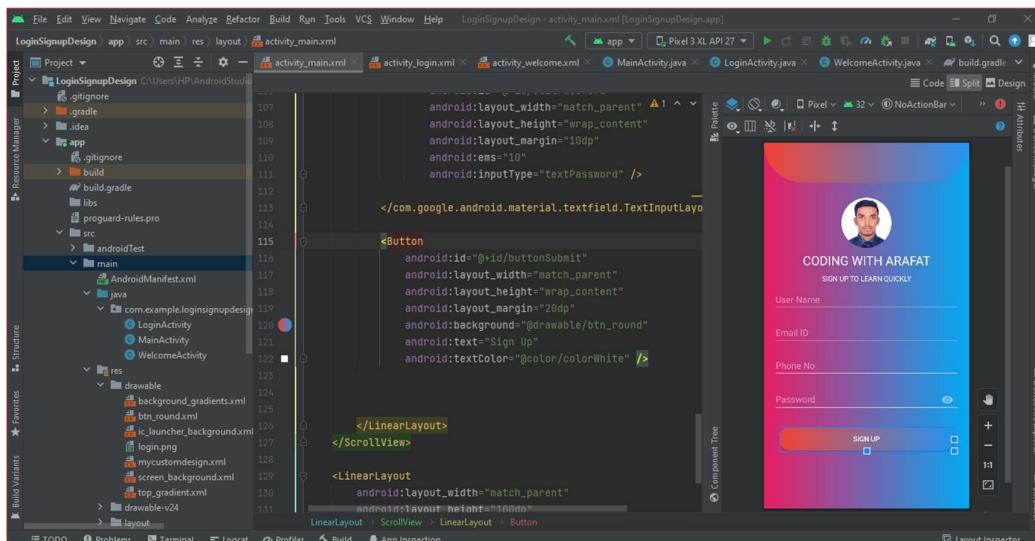
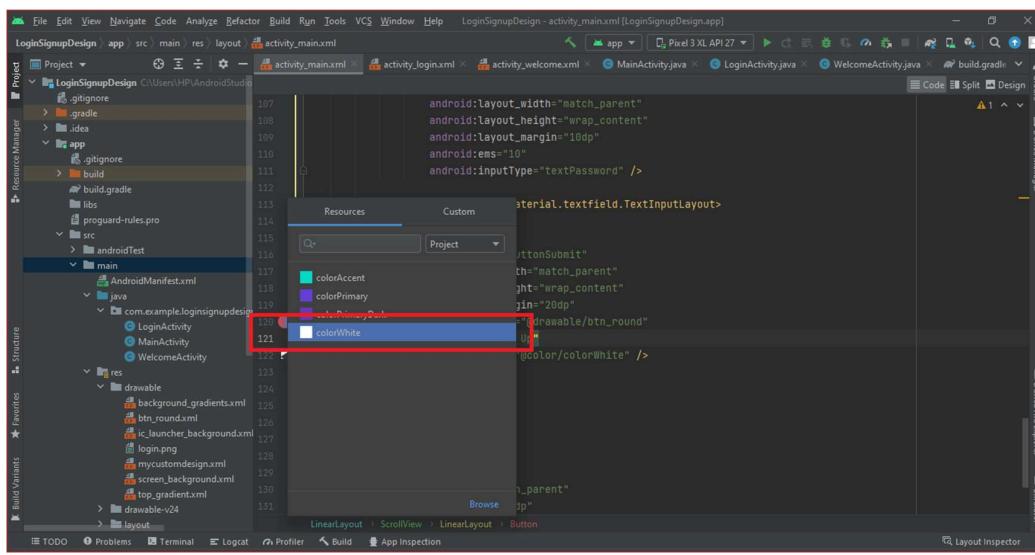
```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help LoginSignupDesign - activity_main.xml [LoginSignupDesign.app]
LoginSignupDesign app src main res layout activity_main.xml activity_login.xml activity_welcome.xml MainActivity.java LoginActivity.java WelcomeActivity.java build.gradle
Project .gradle .idea app build .gitignore build.gradle libs proguard-rules.pro
src main AndroidManifest.xml java com.example.loginsignupdesign LoginActivity MainActivity WelcomeActivity
res drawable background_gradients.xml btn_round.xml ic_launcher_background.xml login.png mycustomdesign.xml screen_background.xml top_gradient.xml
layout layout-v24 layout
activity_main.xml
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:ems="10"
    android:inputType="textPassword" />
</com.google.android.material.textfield.TextInputLayout>
<Button
    android:id="@+id/buttonSubmit"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="20dp"
    android:background="@drawable	btn_round"
    android:text="Sign Up"
    android:textColor="@color/colorWhite" />
</LinearLayout>
</ScrollView>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="100dp"
    android:layout_margin="10dp" />

```

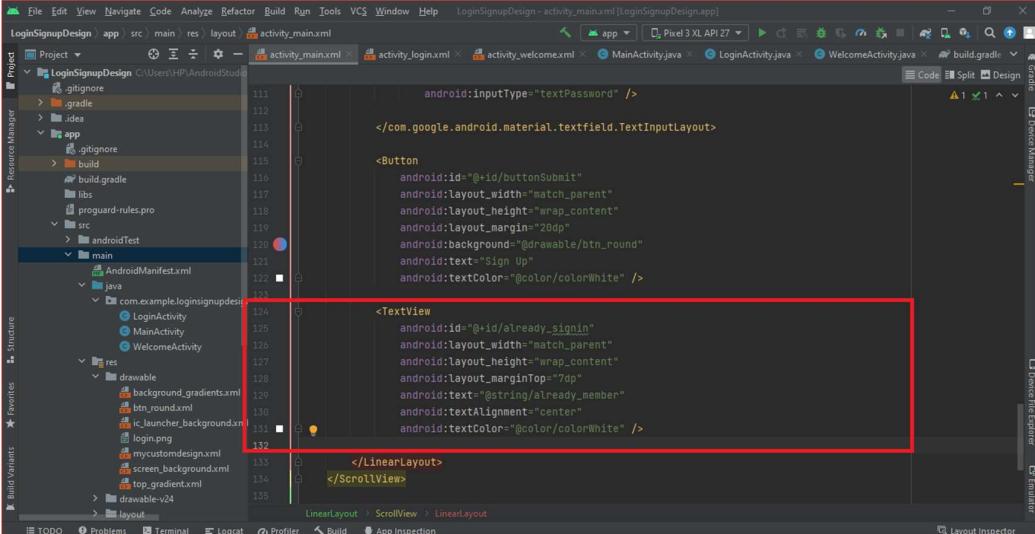


```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help LoginSignupDesign - activity_main.xml [LoginSignupDesign.app]
LoginSignupDesign app src main res layout activity_main.xml activity_login.xml activity_welcome.xml MainActivity.java LoginActivity.java WelcomeActivity.java build.gradle
Project .gradle .idea app build .gitignore build.gradle libs proguard-rules.pro
src main AndroidManifest.xml java com.example.loginsignupdesign LoginActivity MainActivity WelcomeActivity
res drawable background_gradients.xml btn_round.xml ic_launcher_background.xml login.png mycustomdesign.xml screen_background.xml top_gradient.xml
layout layout-v24 layout
activity_main.xml
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:ems="10"
    android:inputType="textPassword" />
</com.google.android.material.textfield.TextInputLayout>
<Button
    android:id="@+id/buttonSubmit"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="20dp"
    android:background="@drawable	btn_round"
    android:text="Sign Up"
    android:textColor="@color/colorWhite" />
</LinearLayout>
</ScrollView>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="100dp"
    android:layout_margin="10dp" />

```



1.10. Add “I am already a member” text below in the sing up button. as shown in the figure below.

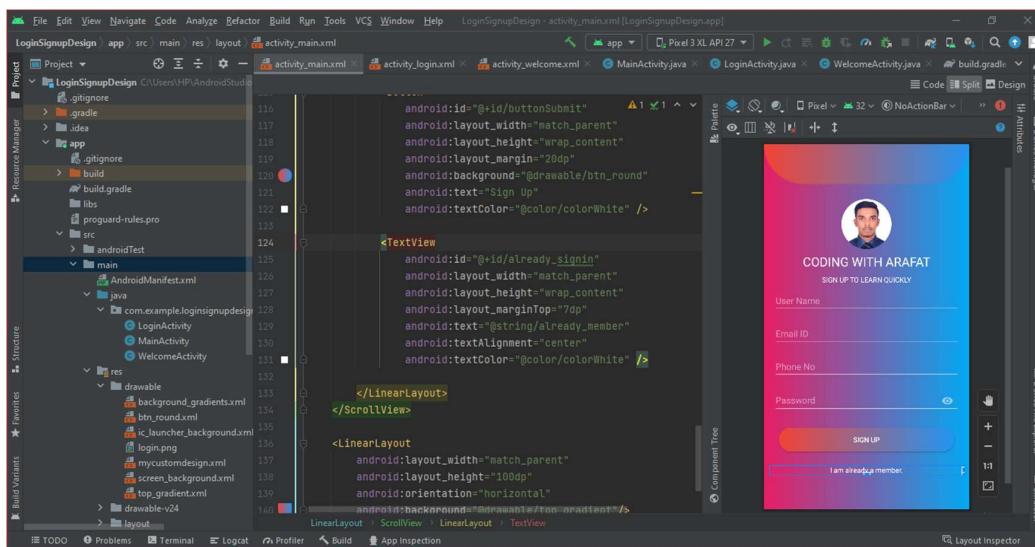


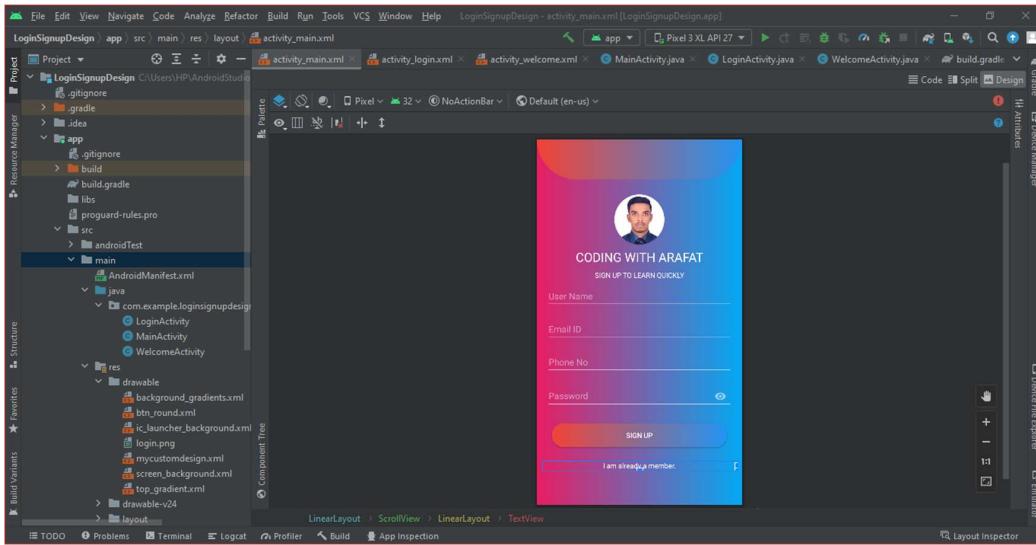
```

<Button
    android:id="@+id/buttonSubmit"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="20dp"
    android:background="@drawable/btn_round"
    android:text="Sign Up"
    android:textColor="@color/colorWhite" />

<TextView
    android:id="@+id/already_signin"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:text="@string/already_member"
    android:textAlignment="center"
    android:textColor="@color/colorWhite" />

```





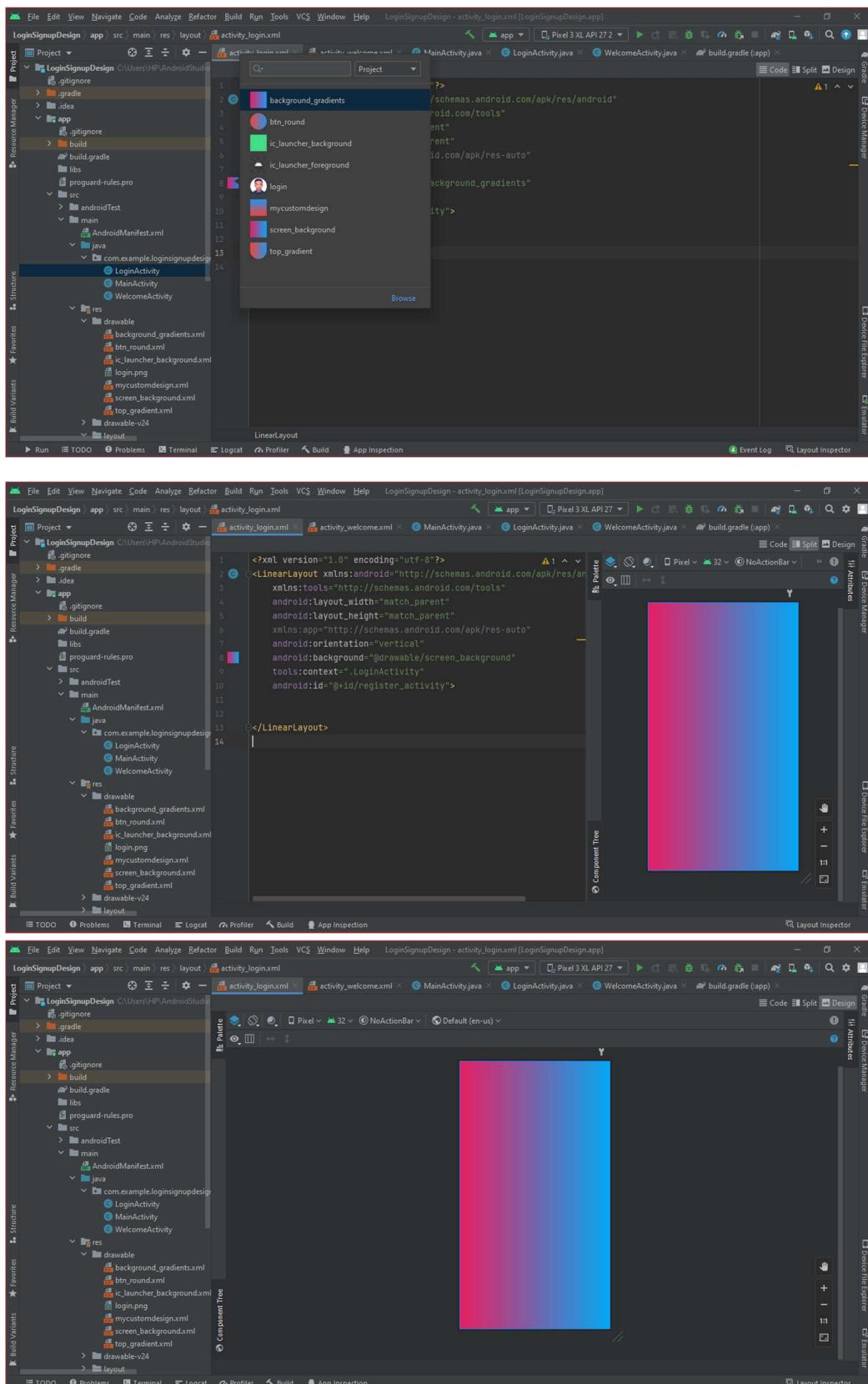
2. Sing In interface layout

2.1. Add background gradient and background color. as shown in the figure below.

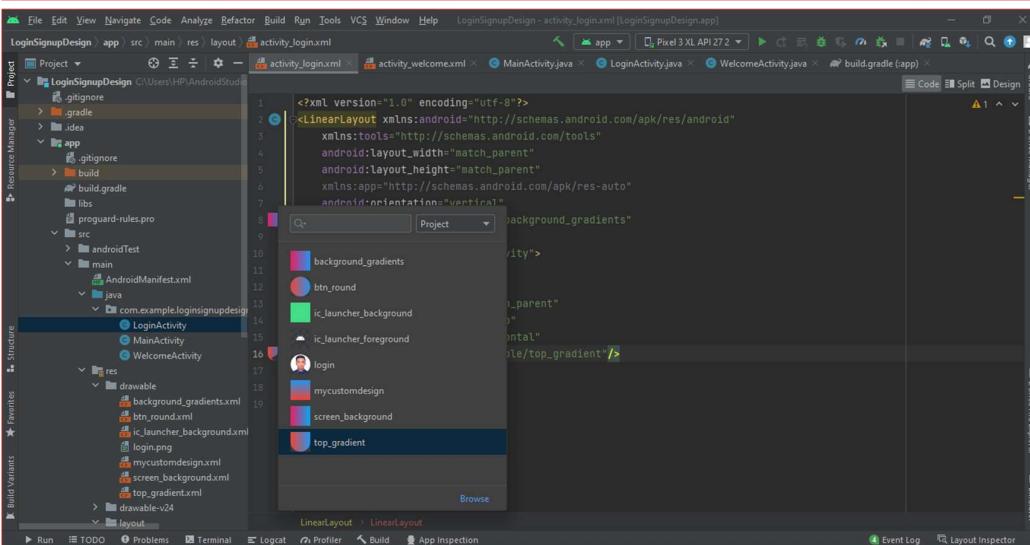
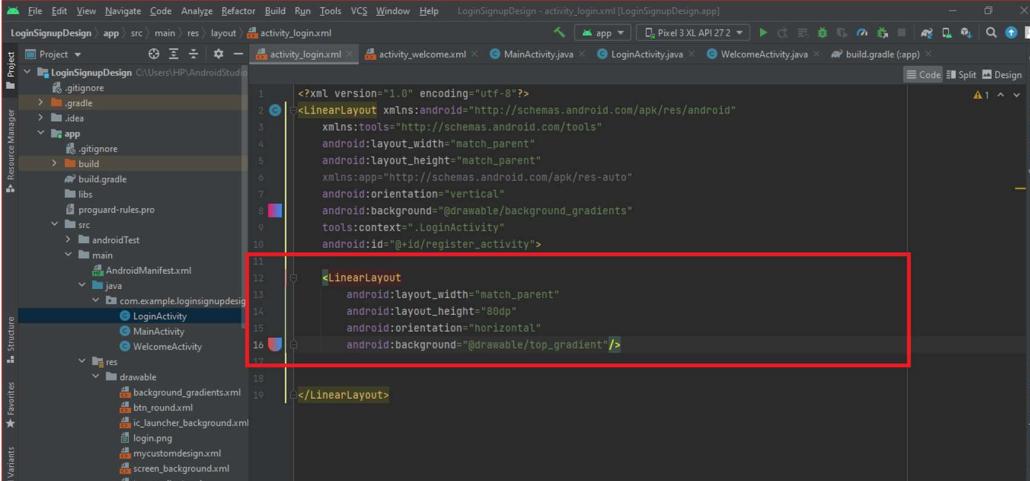
```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:orientation="vertical"
    android:background="#e64a89"
    tools:context=".LoginActivity"
    android:id="@+id/register.activity">

```



2.2. Add the top gradient in front and put into the drawable folder. as shown in the figure below.

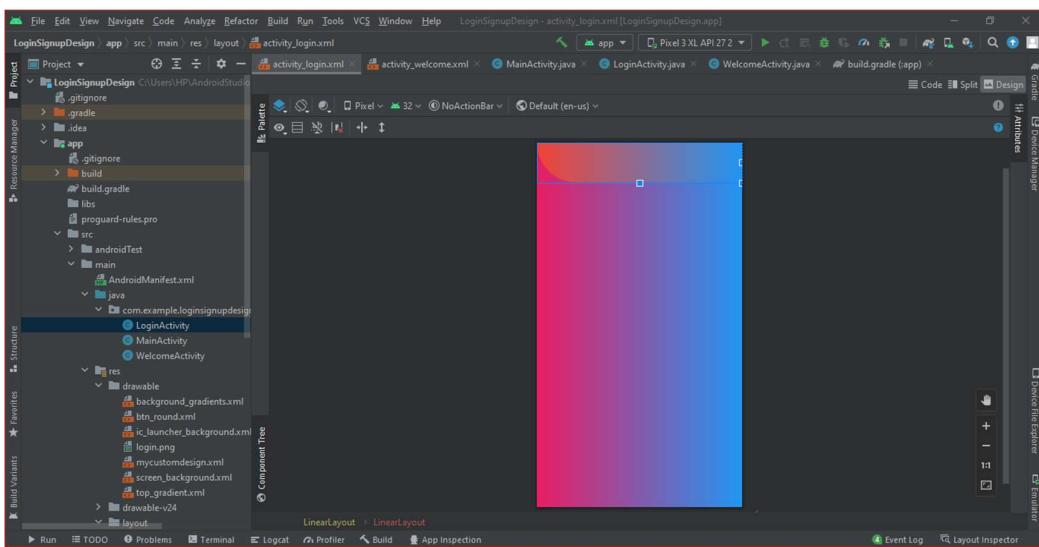
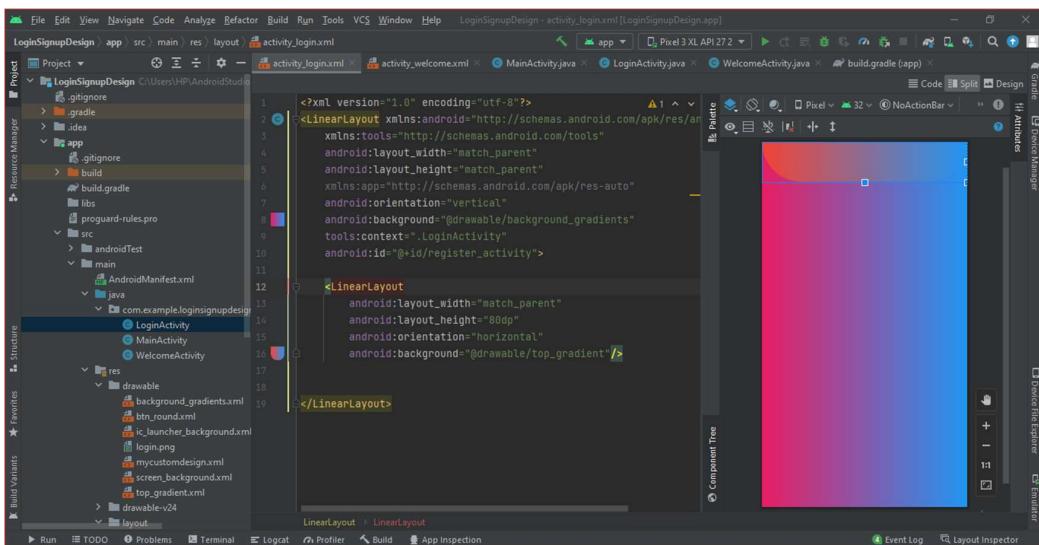


```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:apps="http://schemas.android.com/apk/res-auto"
    android:orientation="vertical"
    android:background="@drawable/background_gradients"
    tools:context=".LoginActivity"
    android:id="@+id/register_activity">

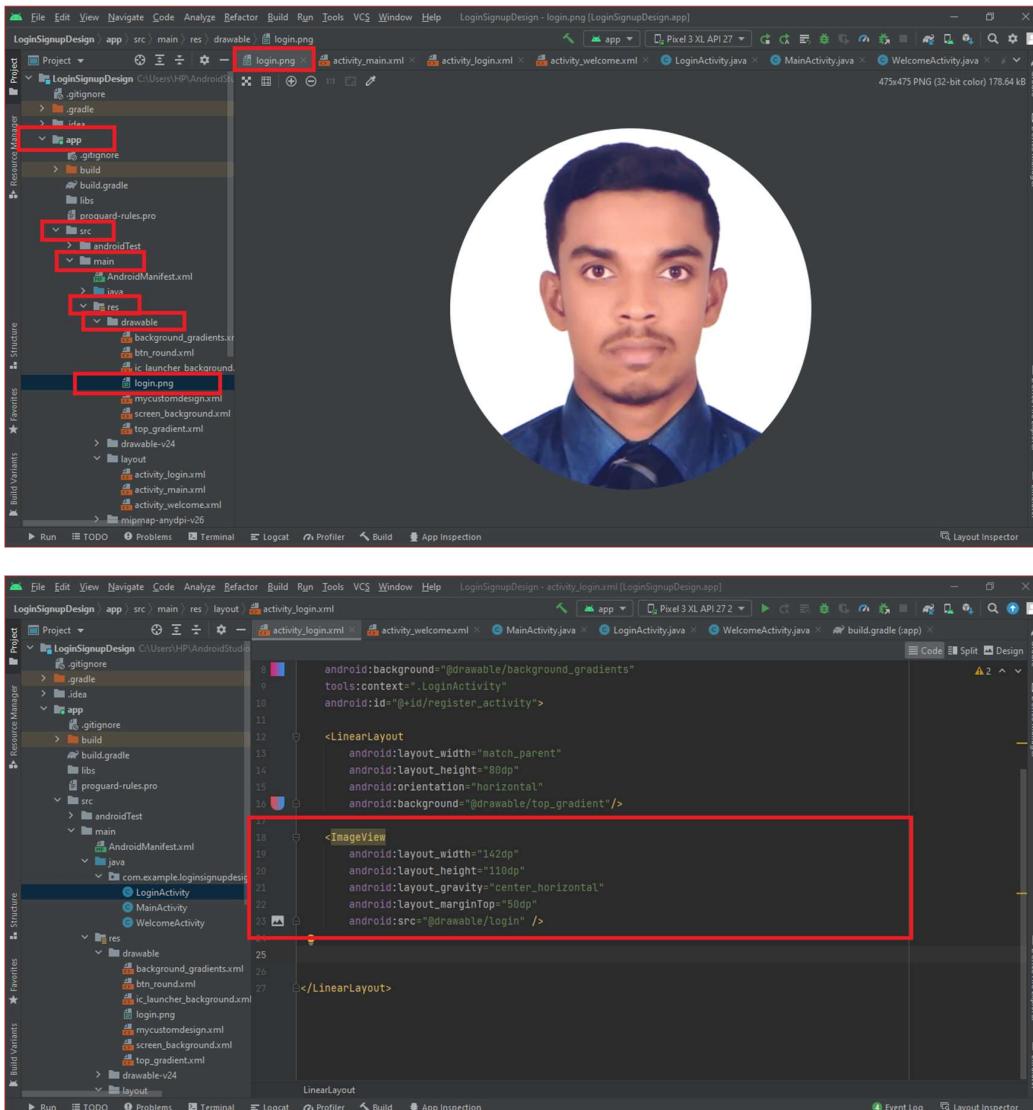
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="80dp"
        android:orientation="horizontal"
        android:background="@drawable/top_gradient"/>

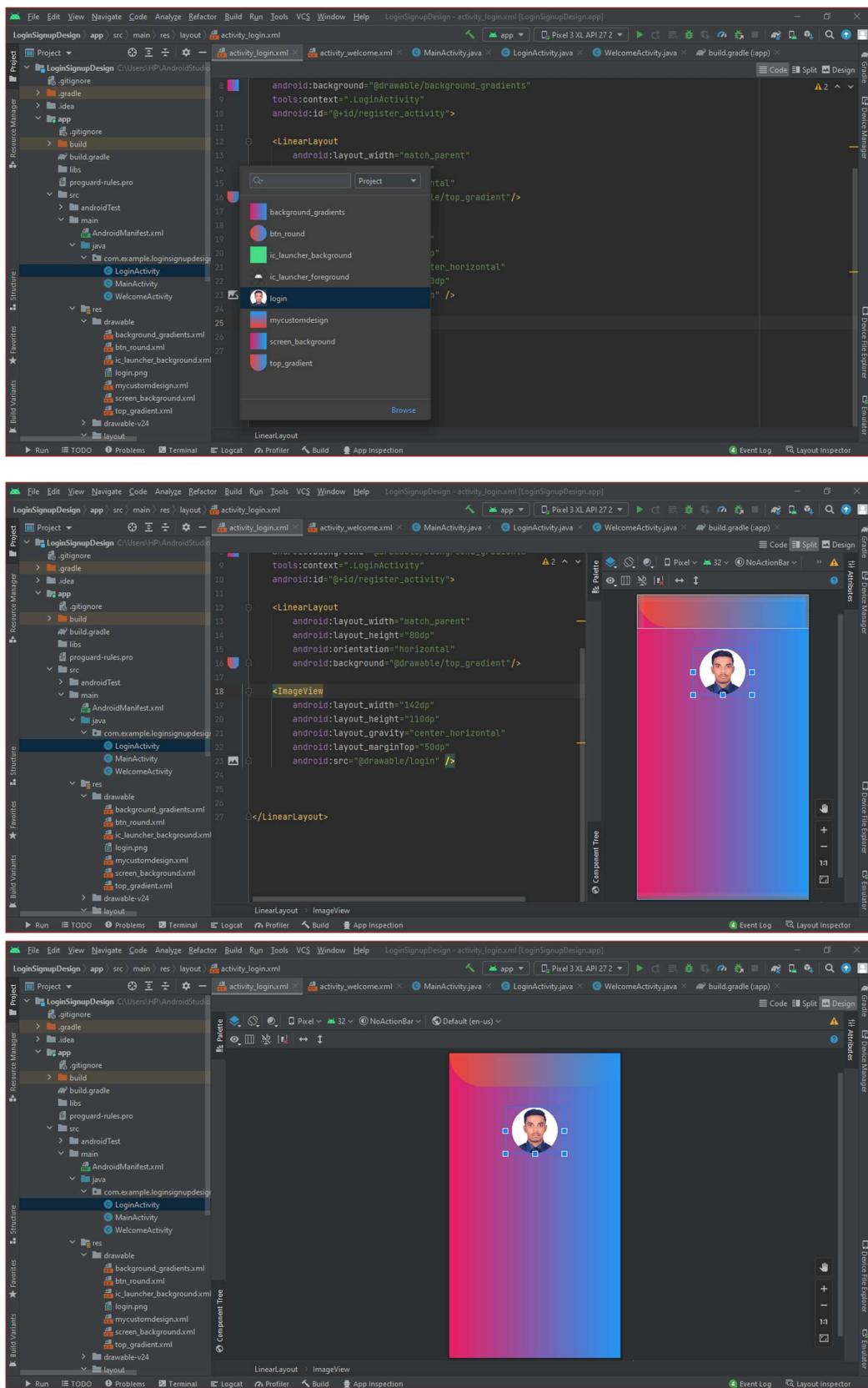
</LinearLayout>
```

The screenshot shows the Android Studio interface with the XML code for `activity_login.xml`. A red box highlights the section where the `background` attribute of the inner `LinearLayout` is set to `@drawable/top_gradient`. Below this, the `background_gradients` resource is expanded, showing it contains references to `background`, `btn_round`, `ic_launcher_background`, `ic_launcher_foreground`, `login`, `mycustomdesign`, `screen_background`, and `top_gradient`.



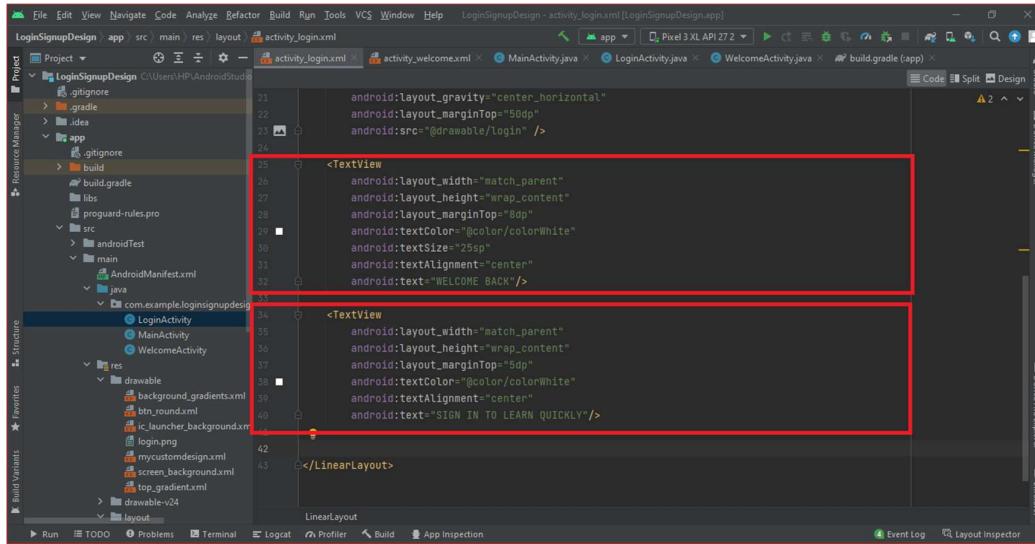
2.3. Now add my own picture in the front . put the picture into the drawable folder. Add a statement to ImageView code android:src="@drawable/login" />. as shown in the figure below.



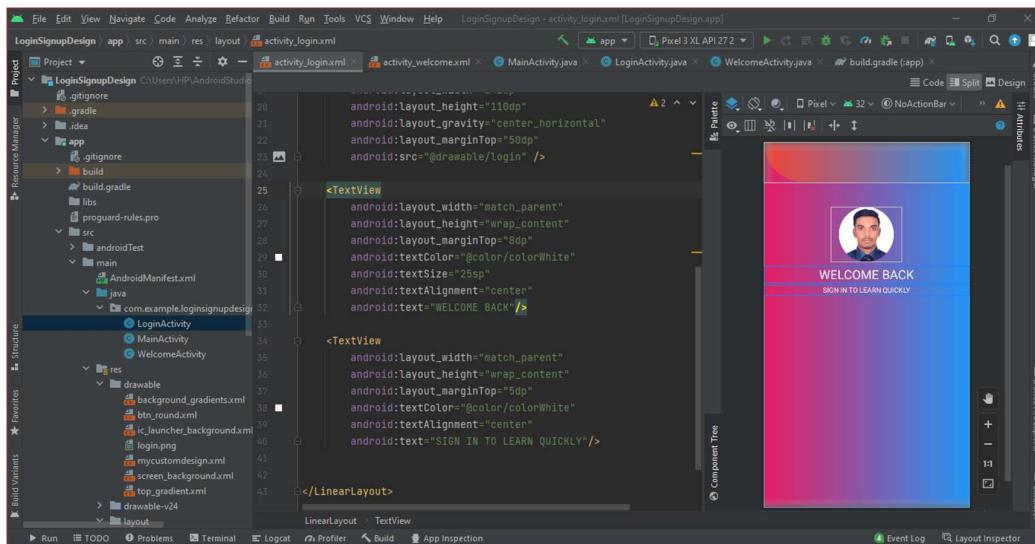


2.4. Now add to “WELCOME BACK” and “SIGN IN TO LEARN QUICKLY”

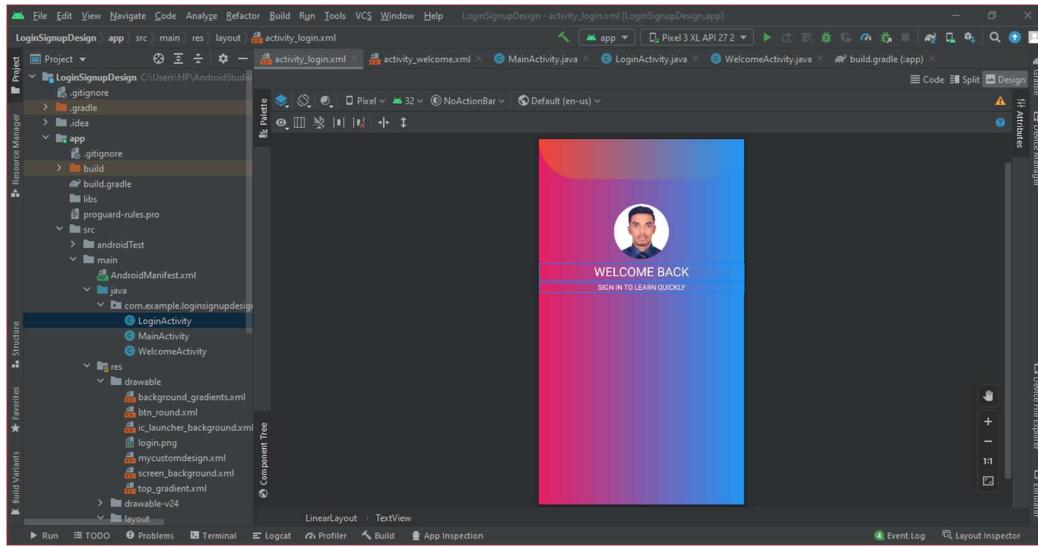
text below in the picture. as shown in the figure below.



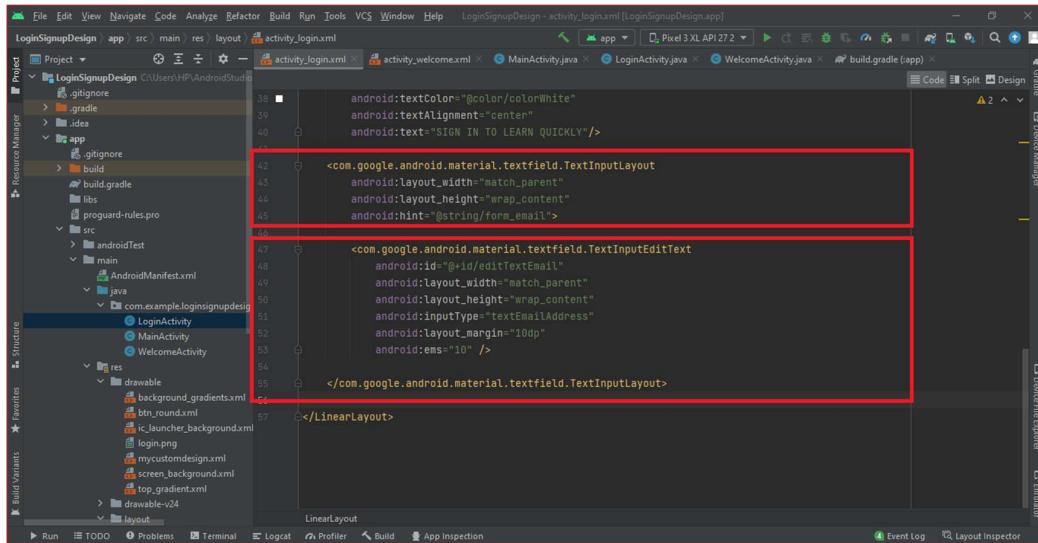
```
<TextView  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_marginTop="8dp"  
    android:textColor="@color/colorWhite"  
    android:textSize="25sp"  
    android:textAlignment="center"  
    android:text="WELCOME BACK"/>  
  
<TextView  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_marginTop="8dp"  
    android:textColor="@color/colorWhite"  
    android:textSize="25sp"  
    android:textAlignment="center"  
    android:text="SIGN IN TO LEARN QUICKLY"/>
```

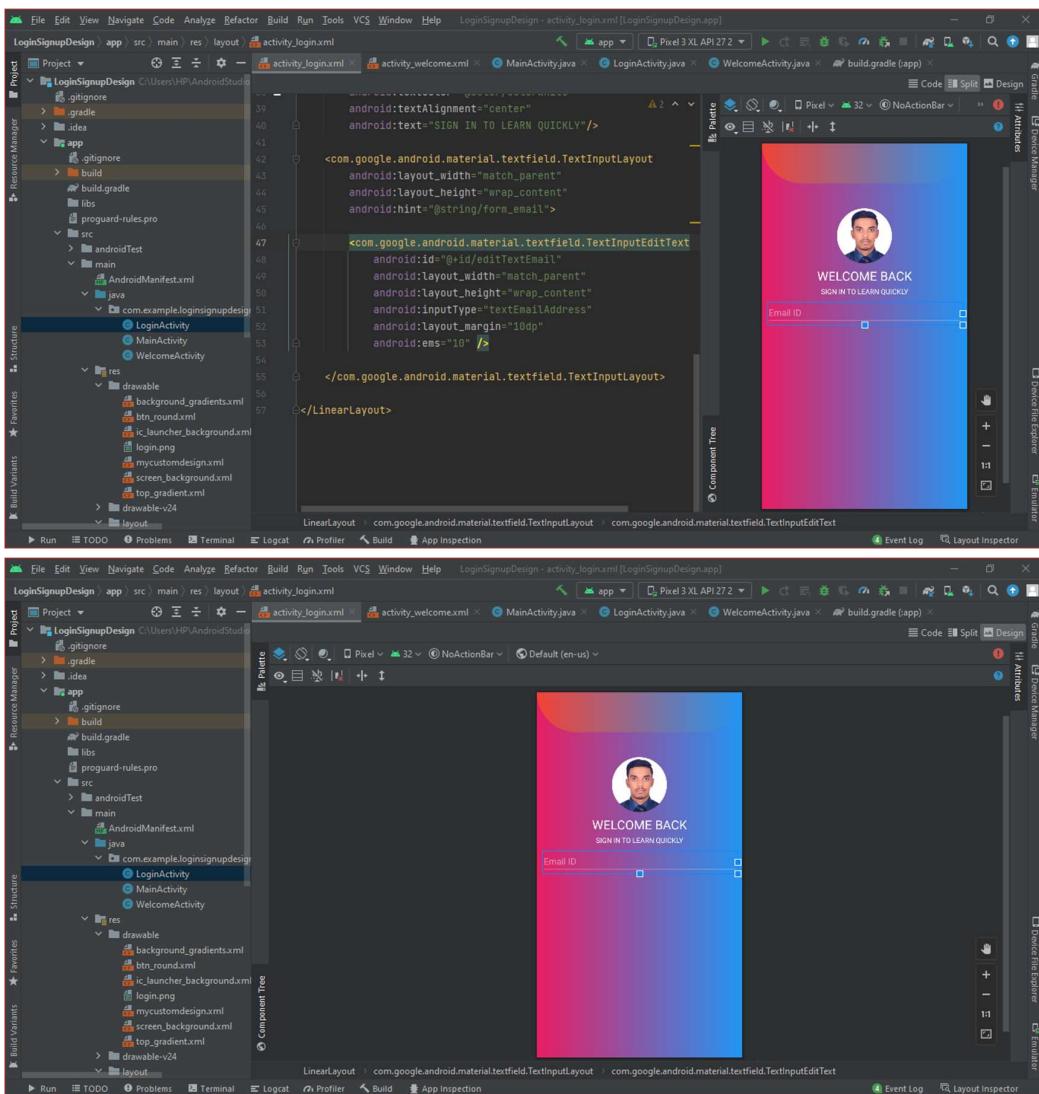


```
<TextView  
    android:layout_width="match_parent"  
    android:layout_height="110dp"  
    android:layout_gravity="center_horizontal"  
    android:layout_marginTop="50dp"  
    android:src="@drawable/login" />  
  
<TextView  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_marginTop="8dp"  
    android:textColor="@color/colorWhite"  
    android:textSize="25sp"  
    android:textAlignment="center"  
    android:text="WELCOME BACK"/>  
  
<TextView  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_marginTop="8dp"  
    android:textColor="@color/colorWhite"  
    android:textSize="25sp"  
    android:textAlignment="center"  
    android:text="SIGN IN TO LEARN QUICKLY"/>
```

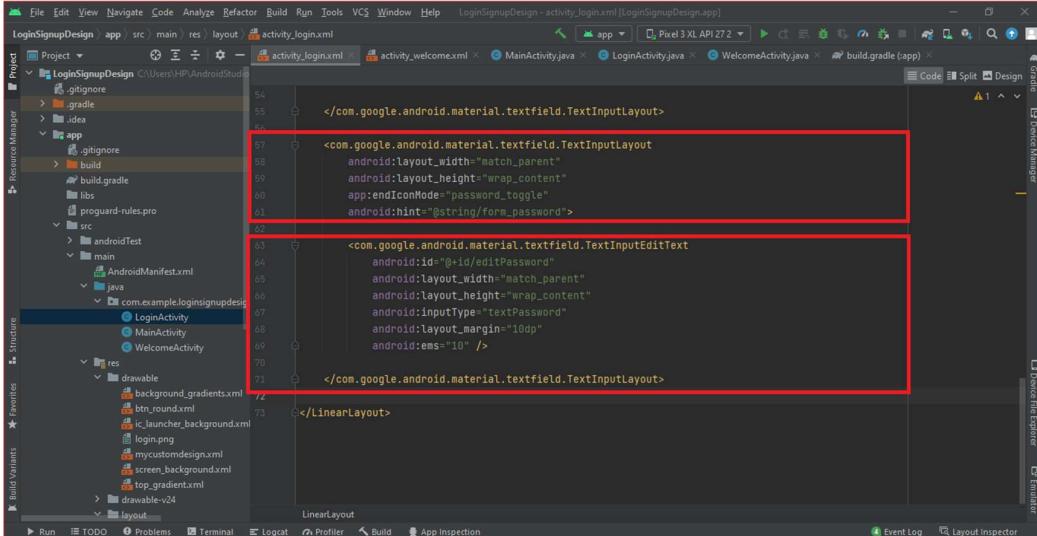


2.5. Add “Email ID” input box in the layout. as shown in the figure below.





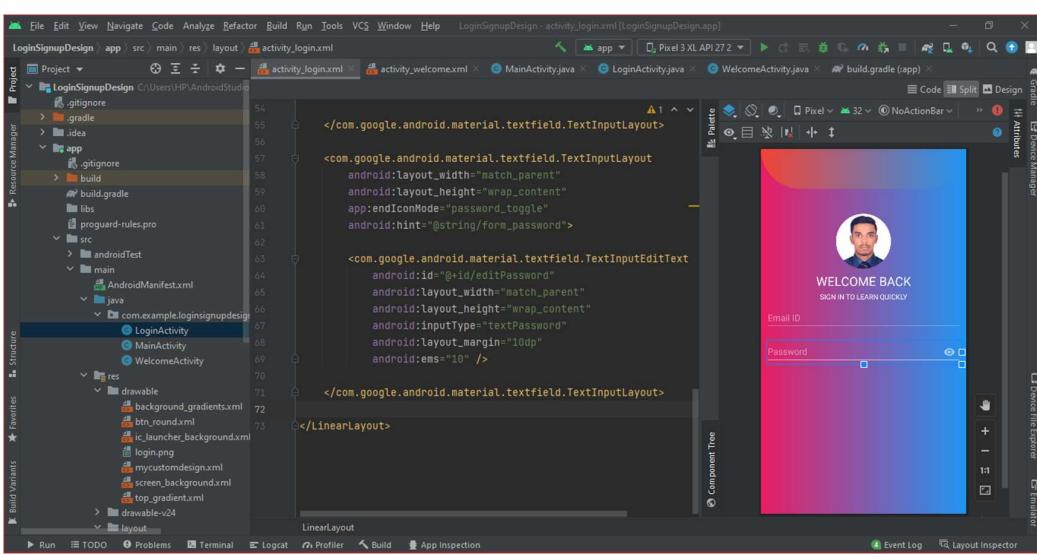
2.6. Add “Password” input box in the layout. as shown in the figure below.

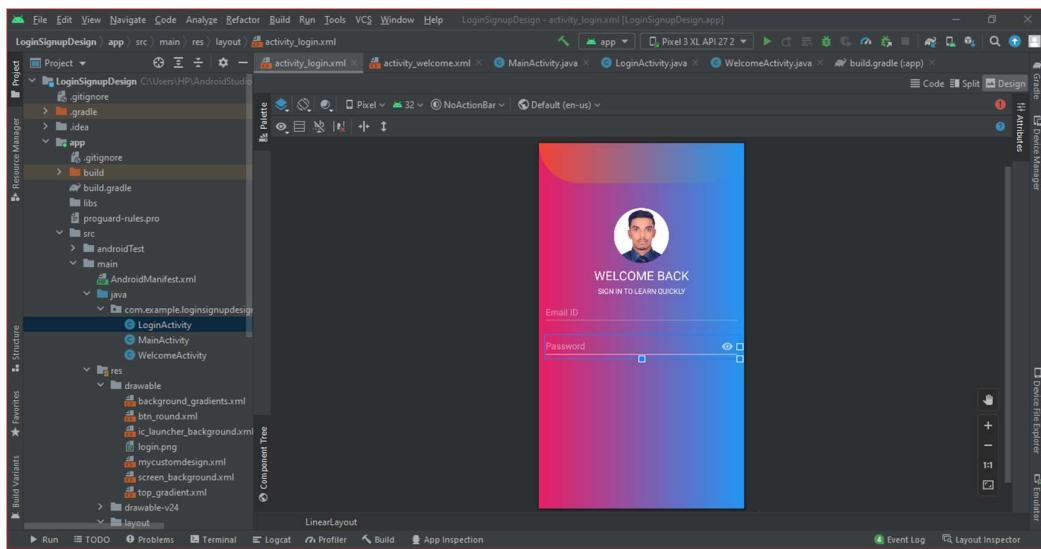


```

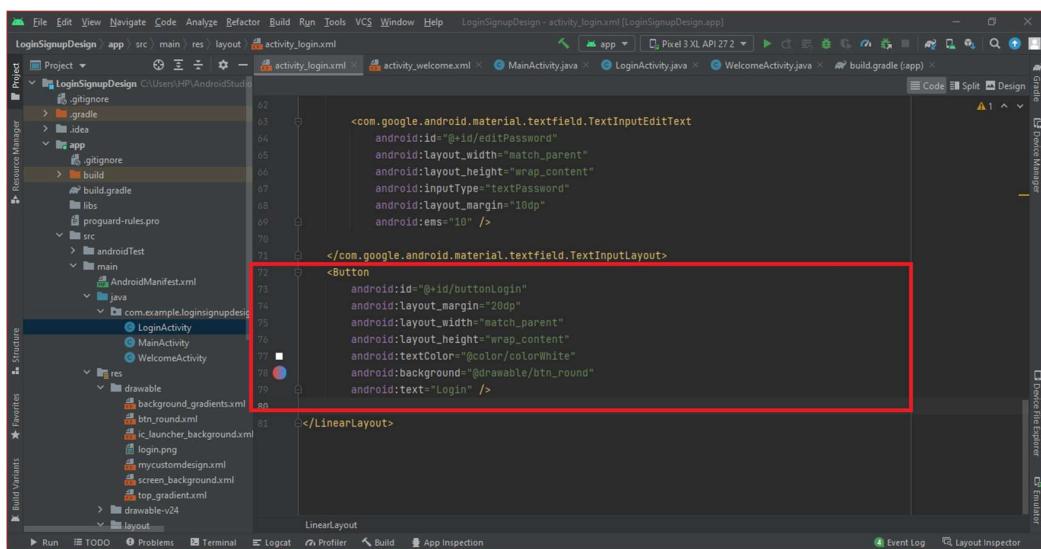
<com.google.android.material.textfield.TextInputLayout>
    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        app:endIconMode="password_toggle"
        android:hint="@string/form_password">
        <com.google.android.material.textfield.TextInputEditText
            android:id="@+id/editTextPassword"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:inputType="textPassword"
            android:layout_margin="10dp"
            android:ems="10" />
    </com.google.android.material.textfield.TextInputLayout>
</LinearLayout>

```





2.7. Add “Sing In” button in the layout . Sing up text, text color, and button color.



The image shows three screenshots of the Android Studio interface, illustrating the design process for a login screen.

Screenshot 1: The "activity_login.xml" layout file is open in the XML editor. A context menu is displayed over the code, specifically targeting the `com.google.android.material.textfield.TextInputLayout` element. The menu lists several resources from the project's `res` directory, including `background_gradients`, `btn_round`, `ic_launcher_background`, `ic_launcher_foreground`, `login`, and `mycustomdesign`. This indicates that the developer is trying to apply a specific gradient or shape to the input field.

```

<com.google.android.material.textfield.TextInputLayout
    android:id="@+id/editPassword"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:inputType="textPassword"
    android:layout_margin="10dp"/>

```

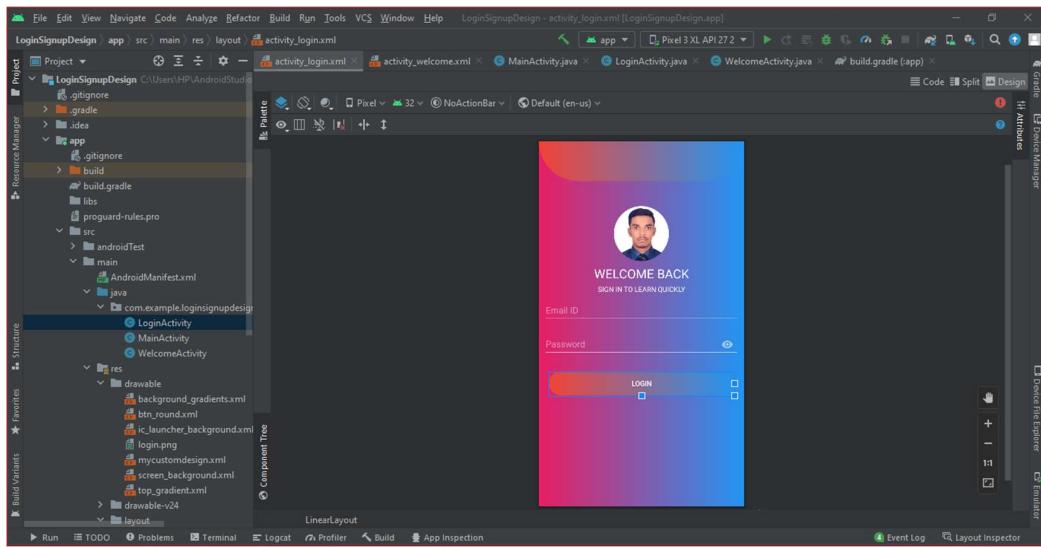
Screenshot 2: The same layout file is shown, but the context menu has been closed. Instead, a floating palette is visible, showing a preview of the current layout. The preview displays a white rectangular area with rounded corners and a blue gradient background, representing the state of the `btn_round` resource.

Screenshot 3: The layout file is now fully visible, showing the complete XML code for the login screen. The developer has successfully applied the `btn_round` resource to the `TextInputLayout` element, giving it a rounded button appearance. The preview in the floating palette shows the final design with the rounded corners and gradient.

```

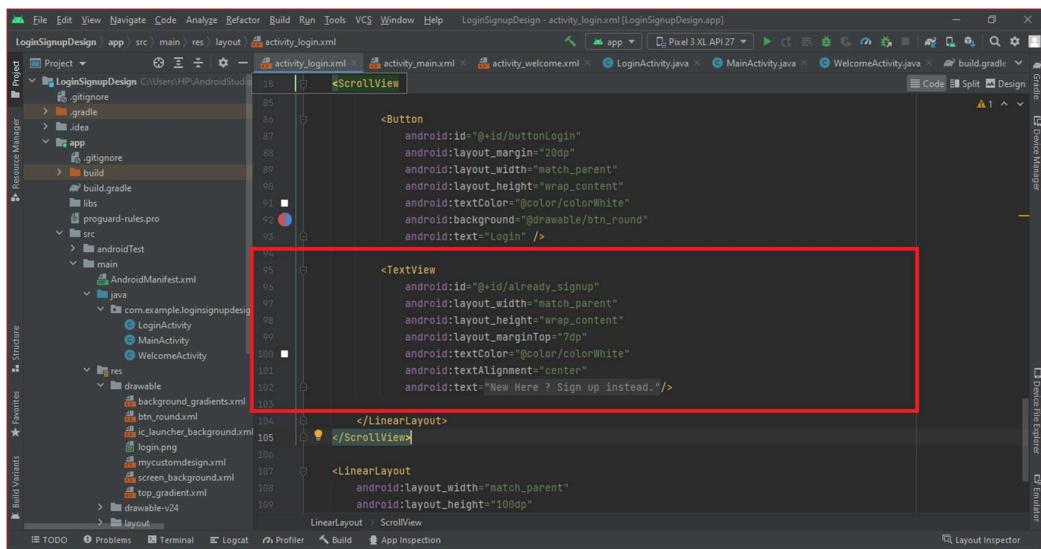
<com.google.android.material.textfield.TextInputLayout
    android:id="@+id/editPassword"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:inputType="textPassword"
    android:layout_margin="10dp"/>

```

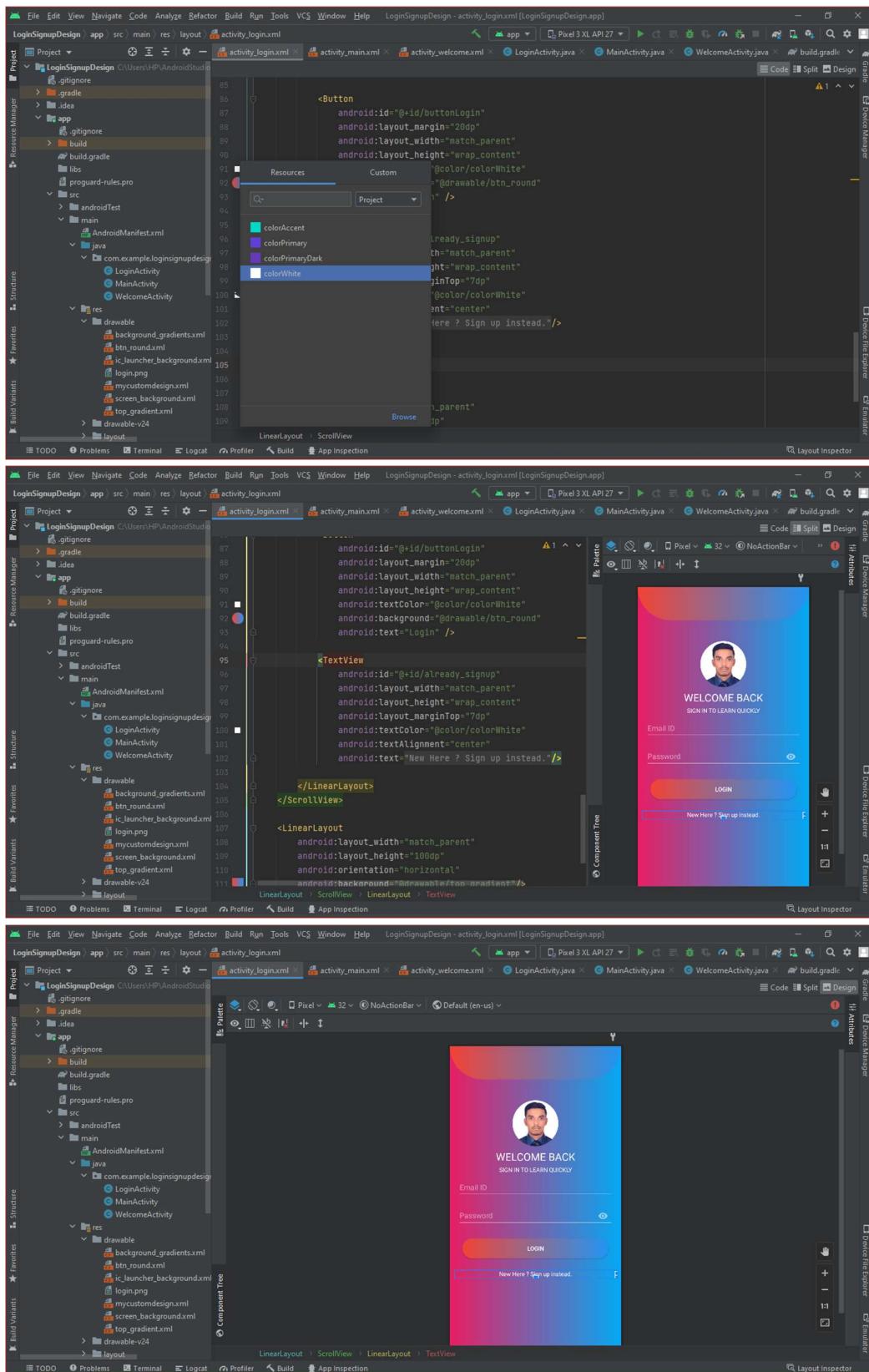


2.8. Add “New Here? Sing up instead” text below in the sing In button.

as shown in the figure below.



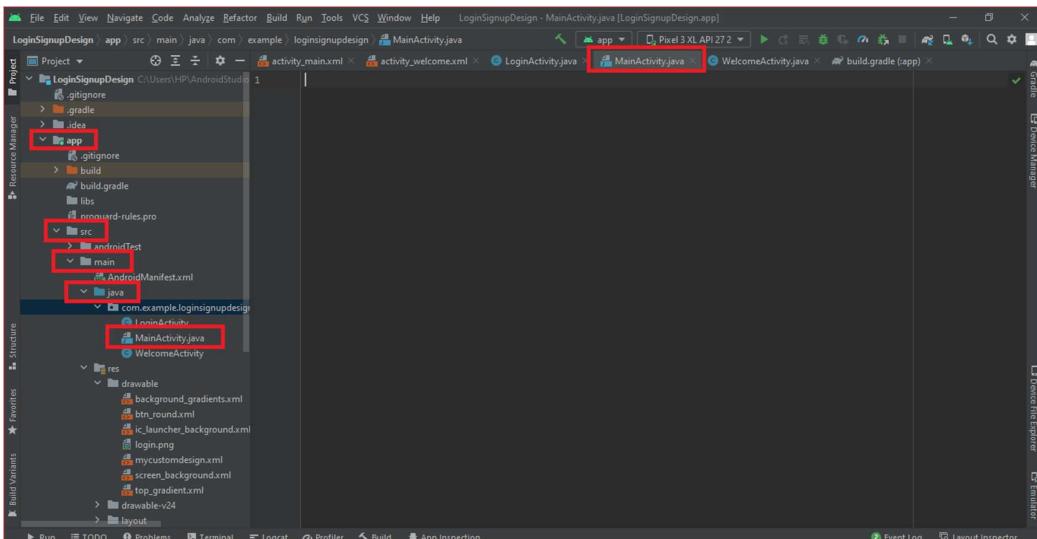
```
        <Button  
            android:id="@+id/buttonLogin"  
            android:layout_margin="20dp"  
            android:layout_width="match_parent"  
            android:layout_height="wrap_content"  
            android:textColor="#color/colorWhite"  
            android:background="@drawable/btn_round"  
            android:text="Login" />  
  
        <TextView  
            android:id="@+id/already_signup"  
            android:layout_width="match_parent"  
            android:layout_height="wrap_content"  
            android:layout_marginTop="7dp"  
            android:textColor="#color/colorWhite"  
            android:textAlignment="center"  
            android:text="New Here ? Sign up instead."/>  
    
```



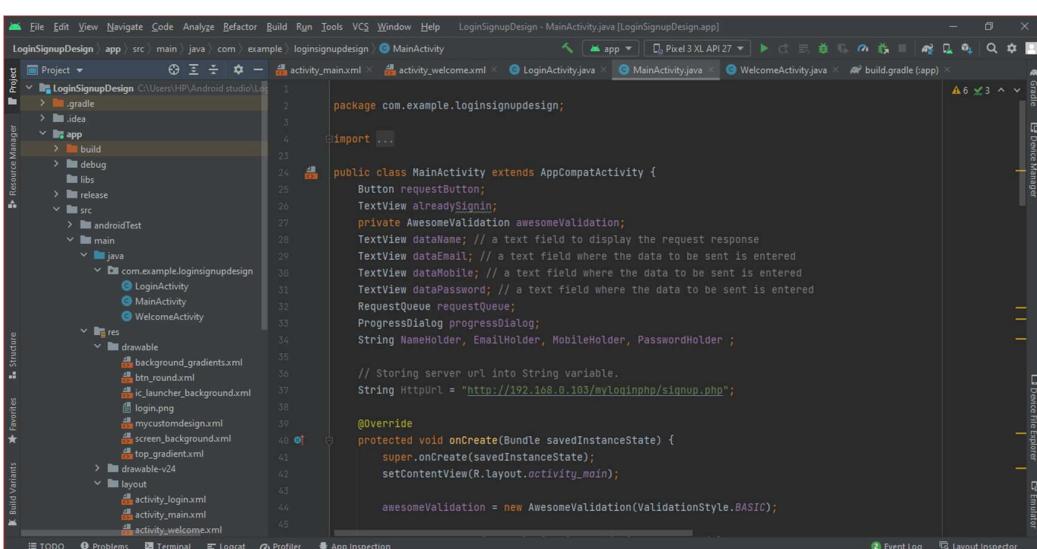
3.Implement the login process

3.1. Open the folder app > SRC > main > java > com > example > CODE

WITH ARAFAT Singup from, delete all the files in the folder, and right-click to create MainActivity.java File, start writing login Logic. as shown in the figure below.



The screenshot shows the Android Studio Project Structure. The 'app' directory is selected. Inside 'app', there is a 'src' folder which contains 'main', 'java', and 'AndroidManifest.xml'. The 'java' folder is expanded, showing 'com.example.loginsignupdesign' package, which contains 'MainActivity.java' and 'WelcomeActivity.java'. Both files are highlighted with red boxes. The 'activity_main.xml' and 'activity_welcome.xml' files are also visible in the list.



The screenshot shows the Java code for 'MainActivity.java'. The code is as follows:

```
package com.example.loginsignupdesign;

import ...

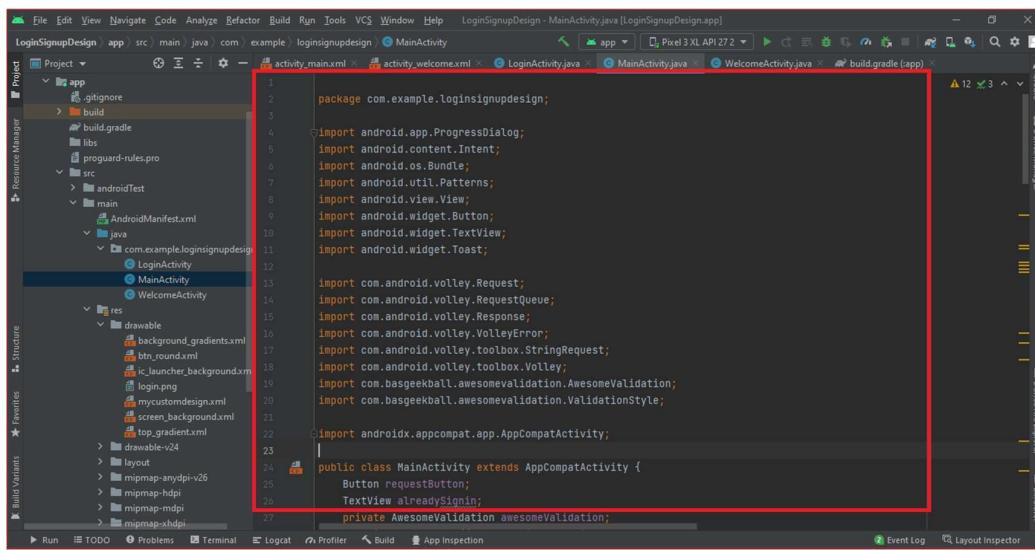
public class MainActivity extends AppCompatActivity {
    Button requestButton;
    TextView alreadySignIn;
    private AwesomeValidation awesomeValidation;
    TextView dataName; // a text field to display the request response
    TextView dataEmail; // a text field where the data to be sent is entered
    TextView dataMobile; // a text field where the data to be sent is entered
    TextView dataPassword; // a text field where the data to be sent is entered
    RequestQueue requestQueue;
    ProgressDialog progressDialog;
    String NameHolder, EmailHolder, MobileHolder, PasswordHolder;

    // Storing server url Into String variable.
    String HttpUrl = "http://192.168.0.103/myloginphp/signup.php";

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

        awesomeValidation = new AwesomeValidation(ValidationStyle.BASIC);
    }
}
```

3.2. Write the framework code of the program, and fill in the layout name of the login interface in the setcontentview function (default is activity_main). as shown in the figure below.



The screenshot shows the Android Studio interface with the project 'LoginSignupDesign' open. The 'MainActivity.java' file is selected in the Project Structure sidebar. A red box highlights the code area where the framework code is written. The code imports various Android libraries and classes, including AppCompatActivity, RequestQueue, Volley, and AwesomeValidation. It defines a class MainActivity that extends AppCompatActivity and includes methods for handling button clicks and validating text fields.

```
1 package com.example.loginsignupdesign;
2
3 import android.app.ProgressDialog;
4 import android.content.Intent;
5 import android.os.Bundle;
6 import android.util.Patterns;
7 import android.view.View;
8 import android.widget.Button;
9 import android.widget.TextView;
10 import android.widget.Toast;
11
12 import com.android.volley.Request;
13 import com.android.volley.RequestQueue;
14 import com.android.volley.Response;
15 import com.android.volley.VolleyError;
16 import com.android.volley.toolbox.StringRequest;
17 import com.android.volley.toolbox.Volley;
18 import com.basgeekball.awesomevalidation.AwesomeValidation;
19 import com.basgeekball.awesomevalidation.ValidationStyle;
20
21 import androidx.appcompat.app.AppCompatActivity;
22
23 public class MainActivity extends AppCompatActivity {
24     Button requestButton;
25     TextView alreadySignedIn;
26     private AwesomeValidation awesomeValidation;
27 }
```

3.3. Write the response function (listening function) corresponding to the login button and the function to obtain the information in the input box. as shown in the figure below.

```

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help LoginSignupDesign - MainActivity.java [LoginSignupDesign.app]
LoginSignupDesign app src main java com.example.loginsignupdesign MainActivity onCreate app Pixel 3 XL API 27 2
activity_main.xml activity_welcome.xml LoginActivity.java MainActivity.java WelcomeActivity.java build.gradle (app)
1 package com.example.loginsignupdesign;
2
3 import android.app.ProgressDialog;
4 import android.content.Intent;
5 import android.os.Bundle;
6 import android.util.Patterns;
7 import android.view.View;
8 import android.widget.Button;
9 import android.widget.TextView;
10 import android.widget.Toast;
11
12 import com.android.volley.Request;
13 import com.android.volley.RequestQueue;
14 import com.android.volley.Response;
15 import com.android.volley.VolleyError;
16 import com.android.volley.toolbox.StringRequest;
17 import com.android.volley.toolbox.Volley;
18 import com.basgeekball.awesomewebview.AwesomeValidation;
19 import com.basgeekball.awesomewebview.ValidationStyle;
20
21 import androidx.appcompat.app.AppCompatActivity;
22
23 public class MainActivity extends AppCompatActivity {
24     Button requestButton;
25     TextView alreadysignin;
26     private AwesomeValidation awesomeValidation;
27 }

```



```

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help LoginSignupDesign - MainActivity.java [LoginSignupDesign.app]
LoginSignupDesign app src main java com.example.loginsignupdesign MainActivity onCreate app Pixel 3 XL API 27 2
activity_main.xml activity_welcome.xml LoginActivity.java MainActivity.java WelcomeActivity.java build.gradle (app)
25
26     Button requestButton;
27     TextView alreadysignin;
28     private AwesomeValidation awesomeValidation;
29     TextView dataName; // a text field to display the request response
30     TextView dataEmail; // a text field where the data to be sent is entered
31     TextView dataMobile; // a text field where the data to be sent is entered
32     TextView dataPassword; // a text field where the data to be sent is entered
33     RequestQueue requestQueue;
34     ProgressDialog progressDialog;
35     String NameHolder, EmailHolder, MobileHolder, PasswordHolder ;
36
37     // Storing server url into String variable.
38     String httpUrl = "http://192.168.0.103/myloginphp/signup.php";
39
40     @Override
41     protected void onCreate(Bundle savedInstanceState) {
42         super.onCreate(savedInstanceState);
43         setContentView(R.layout.activity_main);
44
45         awesomeValidation = new AwesomeValidation(ValidationStyle.BASIC);
46
47         requestButton = (Button) findViewById(R.id.buttonSubmit);
48         alreadysignin = (TextView) findViewById(R.id.already_signin);
49         requestButton.setOnClickListener(mMyListener);
50         alreadysignin.setOnClickListener(mMyListener);
51
52         dataName = (TextView) findViewById(R.id.editTextName);
53
54     }

```

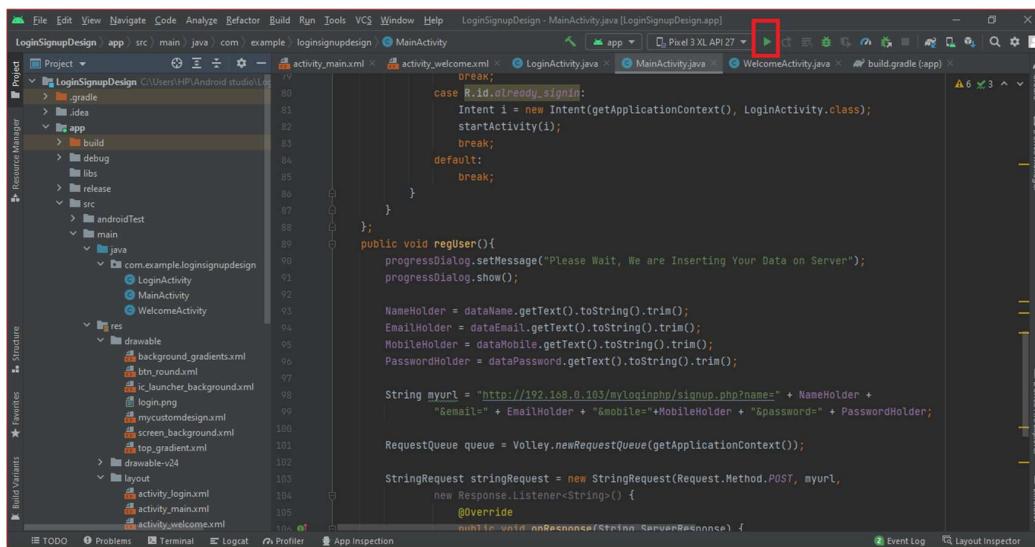
```

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help LoginSignupDesign - MainActivity.java [LoginSignupDesign.app]
LogInSignupDesign app src main java com.example.loginsignupdesign MainActivity onCreate app Pixel 3 XL API 27
activity_main.xml activity_welcome.xml LoginActivity.java MainActivity.java WelcomeActivity.java build.gradle (app)

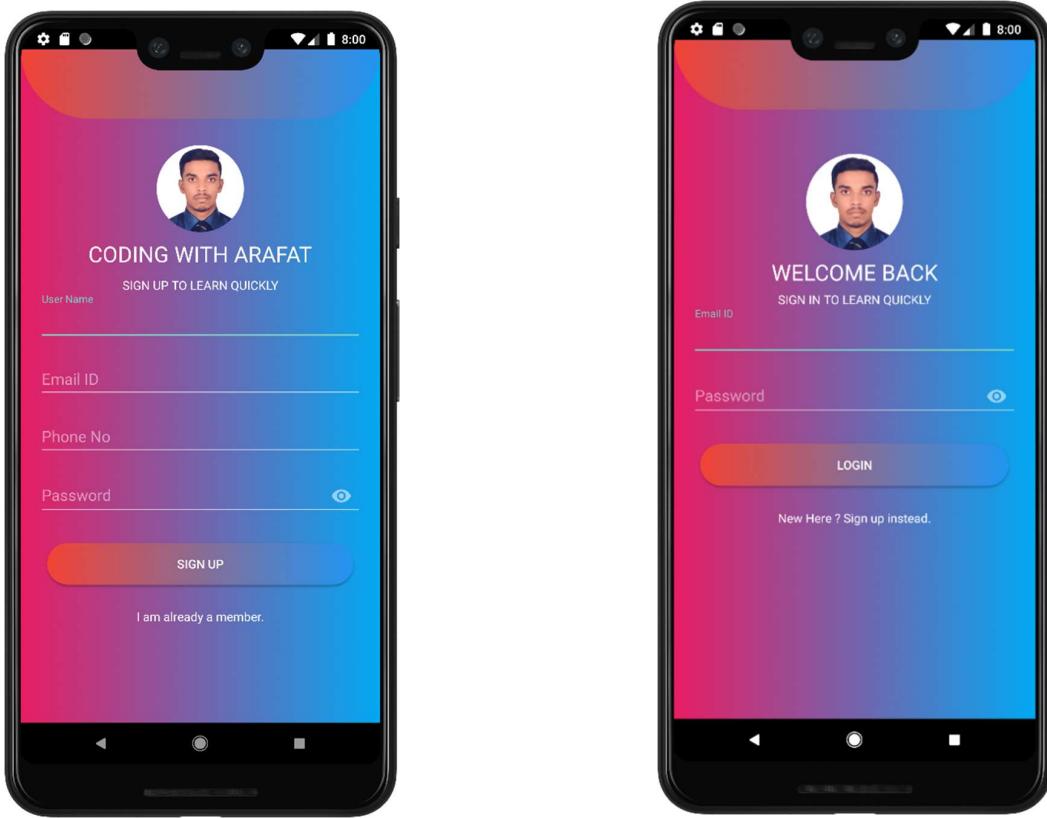
49 alreadySignIn.setOnClickListener(mMyListener);
50
51 dataName = (EditText) findViewById(R.id.editTextName);
52 dataEmail = (EditText) findViewById(R.id.editTextEmail);
53 dataMobile = (EditText) findViewById(R.id.editTextMobile);
54 dataPassword = (EditText) findViewById(R.id.editTextPassword);
55
56
57 String regexPassword = "^(?=.*[\\S+]{5,20})+(?=[^\\s]+(?=\\d)(?=[^\\w\\d]{1})(?=[^@\\w\\d]{1})(?=[^=\\w\\d]{1}){8,20}$";
58 //String regexPassword = "(?=[^\\s]+(?=\\d)(?=[^\\w\\d]{1})(?=[^@\\w\\d]{1})(?=[^=\\w\\d]{1}){8,20}$";
59 //adding validation to edittexts
60 awesomeValidation.addValidation(activity: this, R.id.editTextName, "^(?=.*[\\S+]{5,20})+(?=[^\\s]+(?=\\d)(?=[^\\w\\d]{1})(?=[^@\\w\\d]{1})(?=[^=\\w\\d]{1}){8,20}$");
61 //alternative "[a-zA-Z0-9]([_-.]?[a-zA-Z0-9])"
62 awesomeValidation.addValidation(activity: this, R.id.editTextEmail, Patterns.EMAIL_ADDRESS, R.string.emailerror);
63 awesomeValidation.addValidation(activity: this, R.id.editTextMobile, "[0-9]{8,11}", R.string.mobileerror);
64 awesomeValidation.addValidation(activity: this, R.id.editTextPassword, regexPassword, R.string.passworderror);
65
66 // Creating Volley newRequestQueue .
67 requestQueue = Volley.newRequestQueue( context: MainActivity.this);
68
69 progressDialog = new ProgressDialog( context: MainActivity.this);
70
71 }
72
73 private View.OnClickListener mMyListener = new View.OnClickListener() {
74     public void onClick(View v) {
75         switch (v.getId()) {
76
77             case R.id.buttonSubmit:
78                 if (awesomeValidation.validate()) {
79                     regUser();
80                 }
81                 break;
82             case R.id.already_signin:
83                 Intent i = new Intent(getApplicationContext(), LoginActivity.class);
84                 startActivity(i);
85                 break;
86             default:
87                 break;
88         }
89     }
90 }
91
92 public void regUser(){
93     progressDialog.setMessage("Please Wait, We are Inserting Your Data on Server");
94     progressDialog.show();
95
96     NameHolder = dataName.getText().toString().trim();
97     EmailHolder = dataEmail.getText().toString().trim();
98     MobileHolder = dataMobile.getText().toString().trim();
99     PasswordHolder = dataPassword.getText().toString().trim();
100
101     String myurl = "http://192.168.0.103/myloginphp/signup.php?name=" + NameHolder +
102         "&email=" + EmailHolder + "&mobile=" + MobileHolder + "&password=" + PasswordHolder;
103
104     RequestQueue queue = Volley.newRequestQueue(getApplicationContext());
105
106     StringRequest stringRequest = new StringRequest(Request.Method.POST, myurl,
107         new Response.Listener<String>() {
108             @Override
109             public void onResponse(String ServerResponse) {
110                 // Hiding the progress dialog after all task complete.
111                 progressDialog.dismiss();
112                 // Showing response message coming from server.
113                 Toast.makeText(context: MainActivity.this, ServerResponse, Toast.LENGTH_LONG).show();
114             }
115         },
116         new Response.ErrorListener() {
117             @Override
118             public void onErrorResponse(VolleyError volleyError) {
119                 // Hiding the progress dialog after all task complete.
120                 progressDialog.dismiss();
121                 // Showing error message if something goes wrong.
122                 Toast.makeText(context: MainActivity.this, volleyError.toString(), Toast.LENGTH_LONG).show();
123             }
124         });
125     queue.add(stringRequest);
126 }

```

3.4. At this time, click the icon  in the upper right corner to run the program, and the virtual machine will pop up. as shown in the figure below.

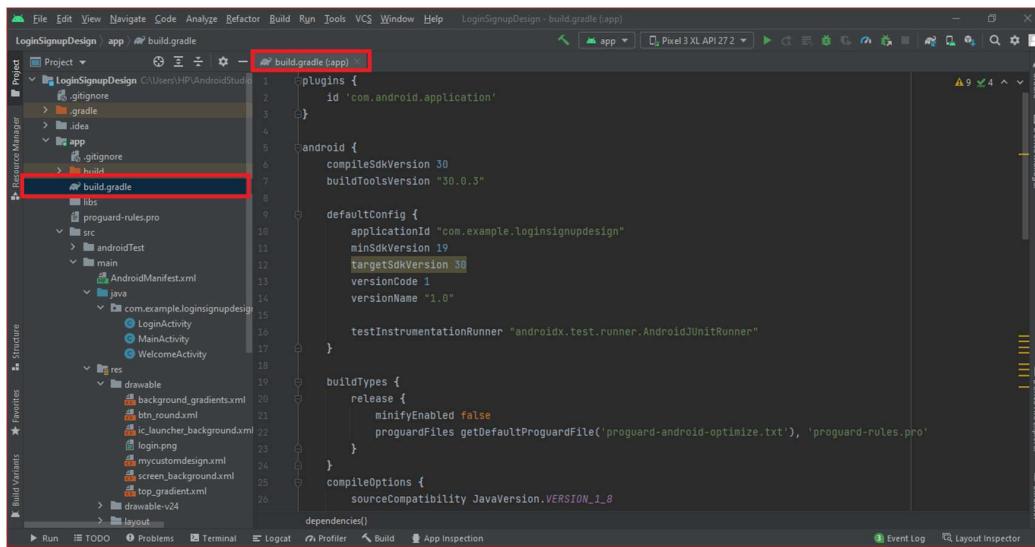


3.5. After clicking the button on the virtual machine, the console will display the contents of the input box. as shown in the figure below.



4. Pack into.Apk file and run on your phone:

4.1. Open the build.gradle File, which can modify the version number, name and other information of APK. as shown in the figure below.



The screenshot shows the Android Studio interface with the project 'LoginSignupDesign' open. The 'build.gradle (app)' file is selected in the Project structure. The code in the editor is as follows:

```
plugins {
    id 'com.android.application'

    android {
        compileSdkVersion 30
        buildToolsVersion "30.0.3"

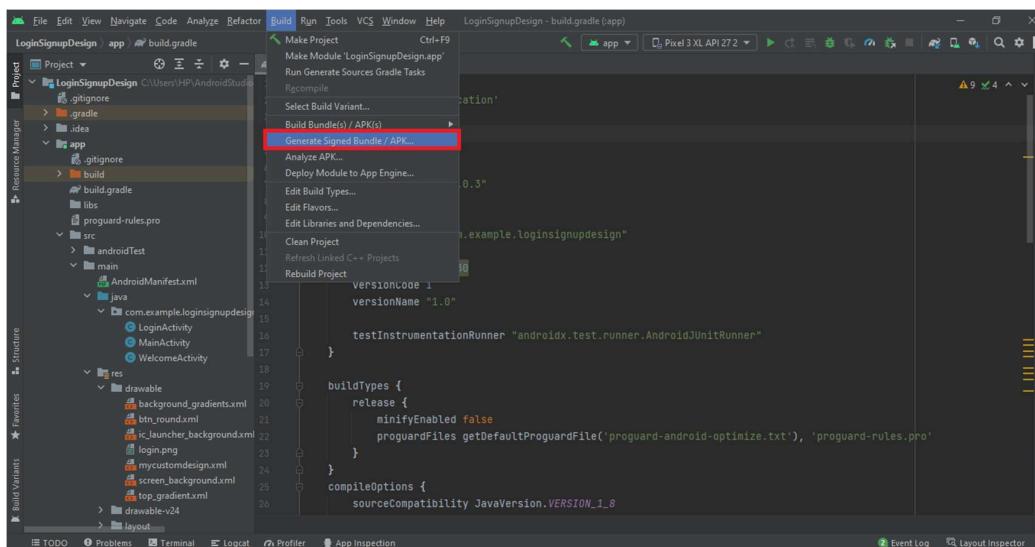
        defaultConfig {
            applicationId "com.example.loginsignupdesign"
            minSdkVersion 19
            targetSdkVersion 30
            versionCode 1
            versionName "1.0"

            testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
        }

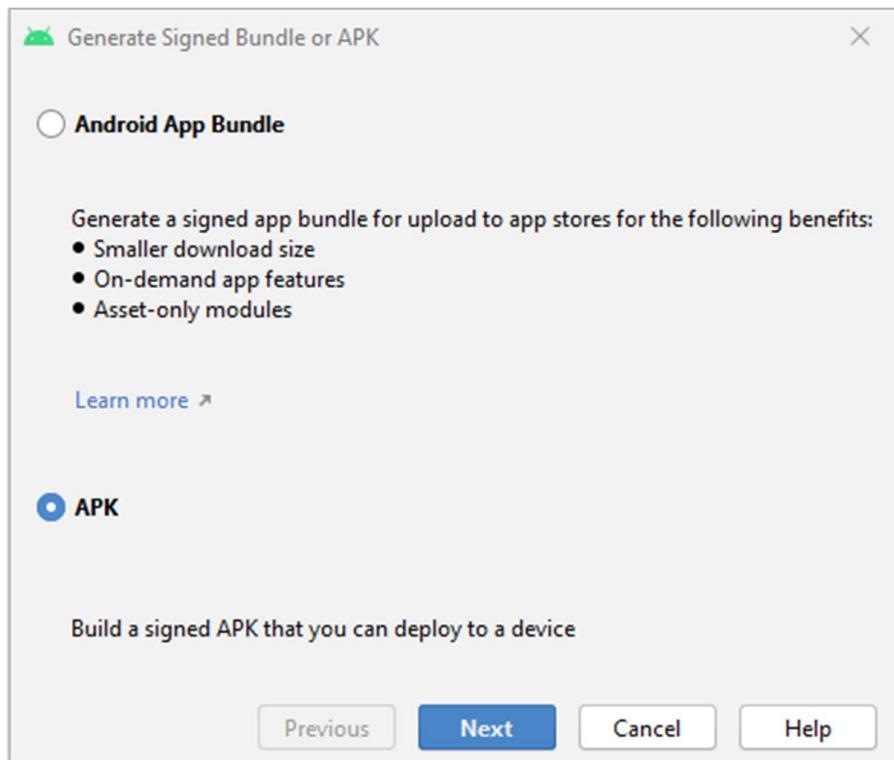
        buildTypes {
            release {
                minifyEnabled false
                proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
            }
        }

        compileOptions {
            sourceCompatibility JavaVersion.VERSION_1_8
            targetCompatibility JavaVersion.VERSION_1_8
        }
    }
}
```

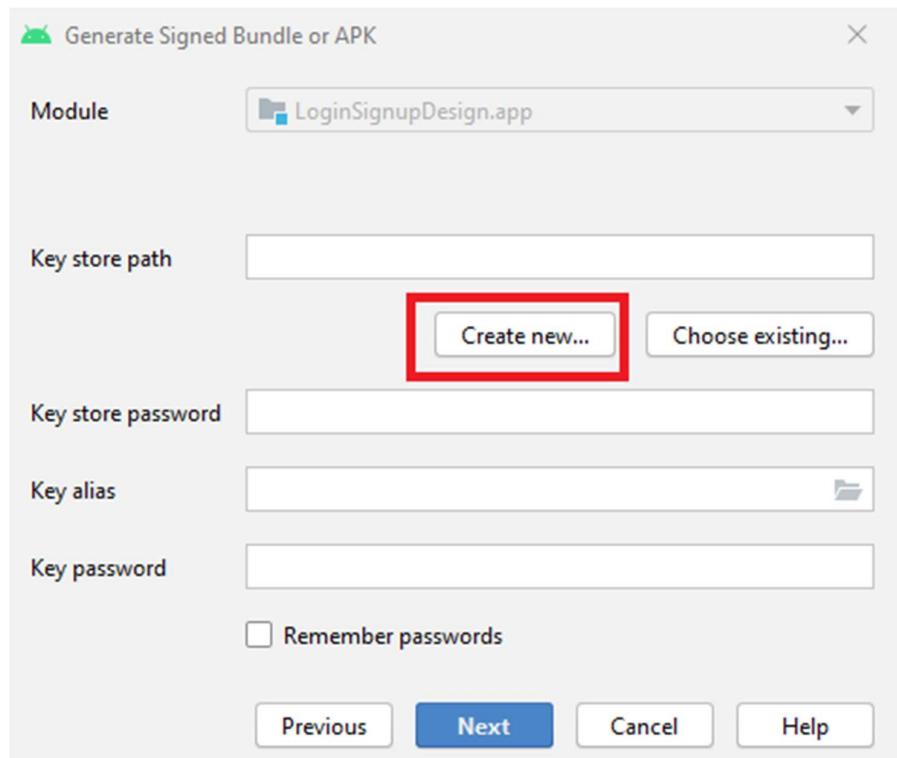
4.2. Click 'Build->Generate Signed APK'



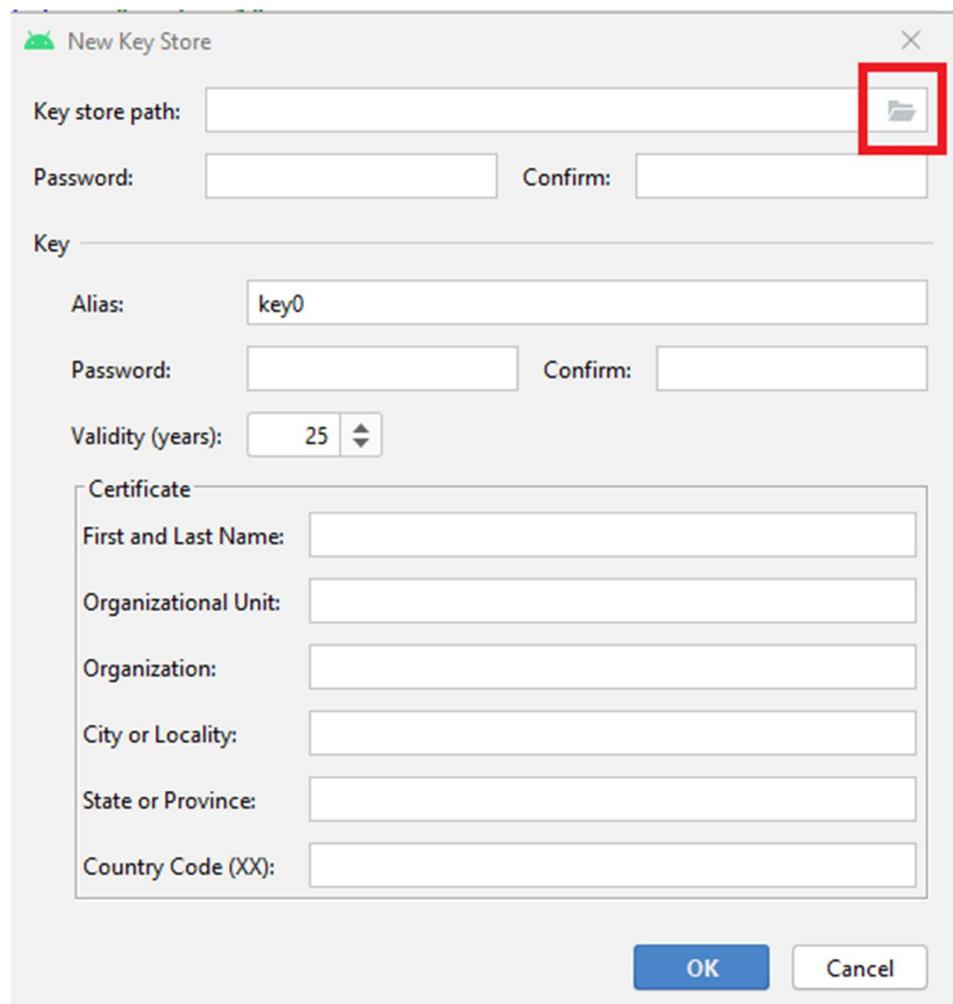
4.3. Select APK after opening

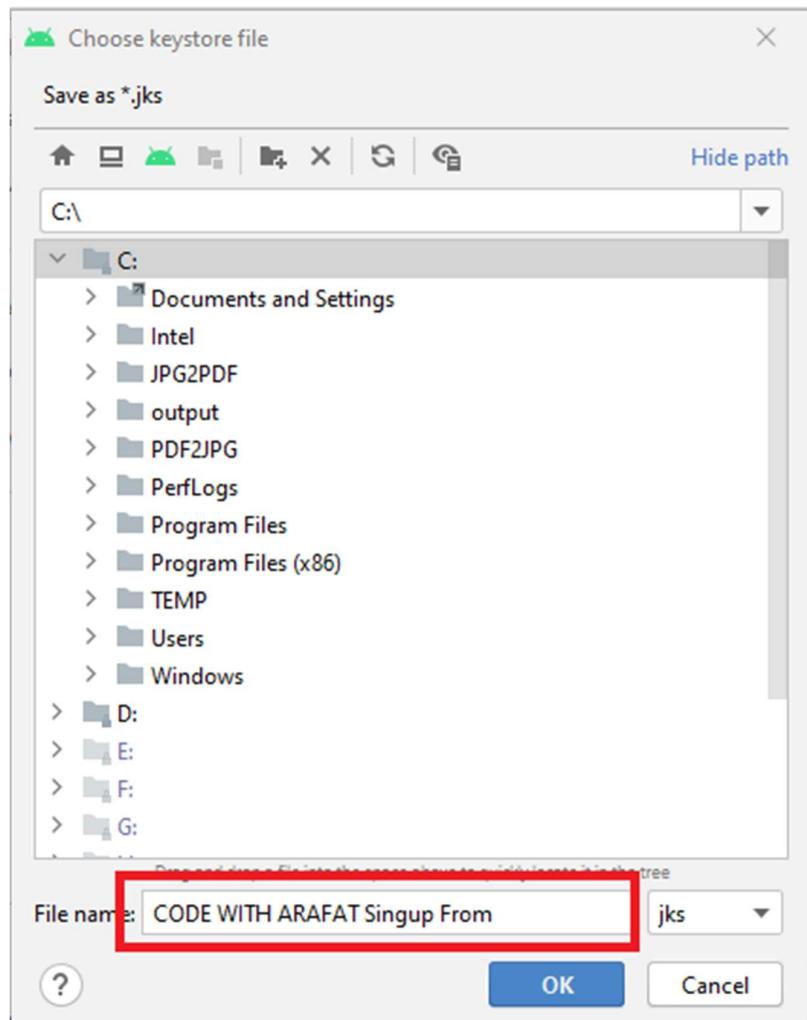


4.4. Click the Button 'create new...'

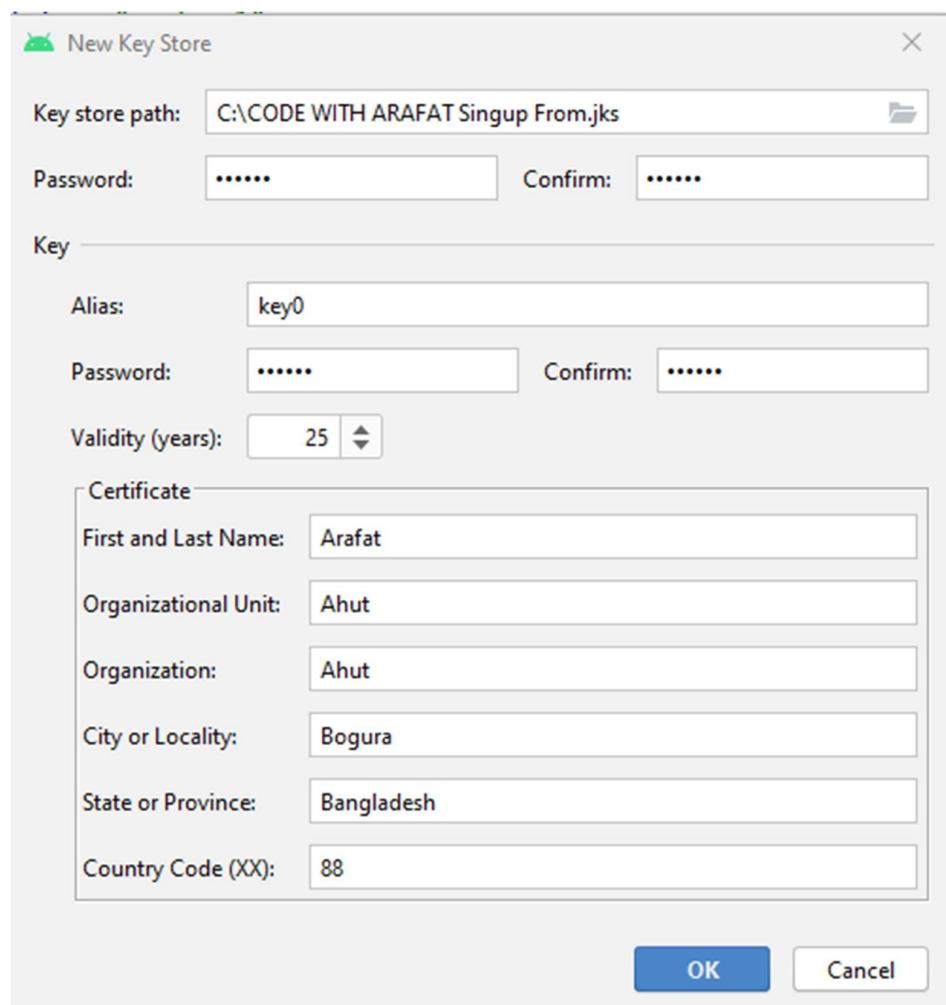


4.5. Create a key file with suffix.JKS

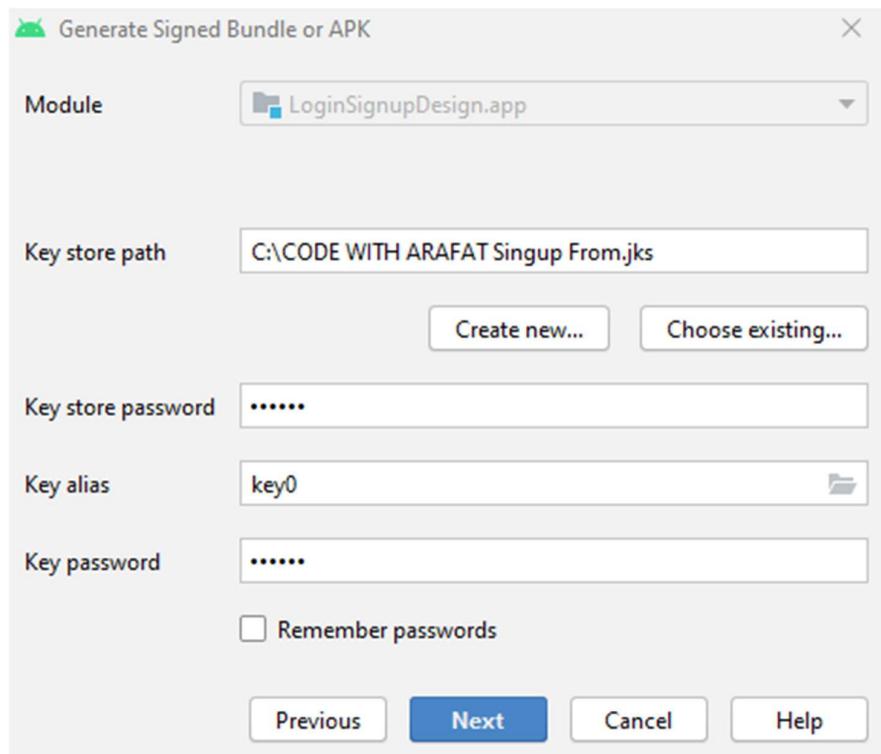


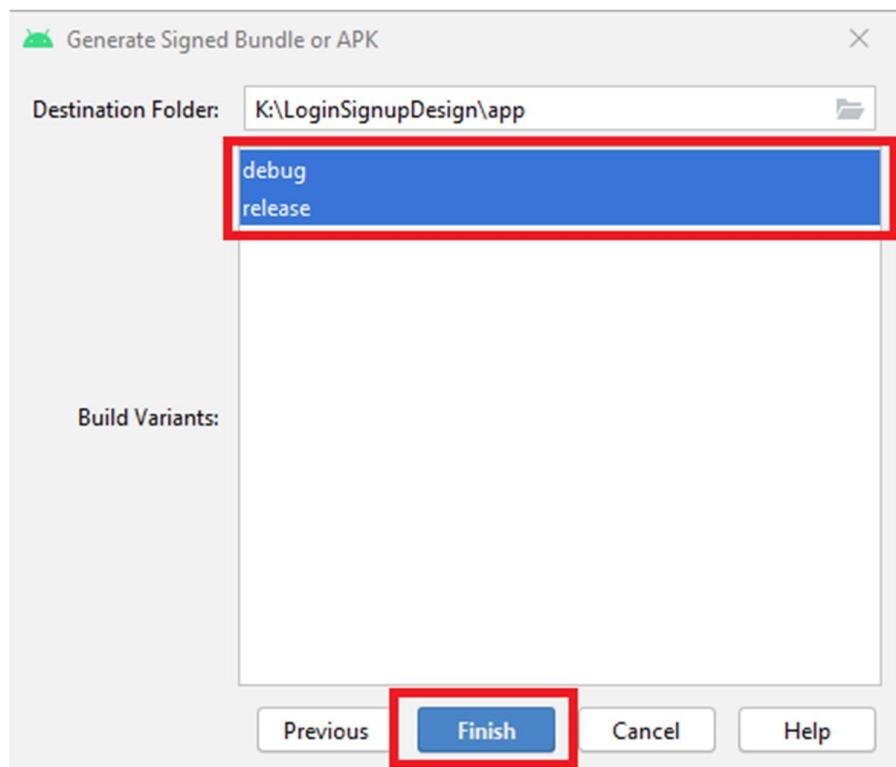


4.6. Fill in other information

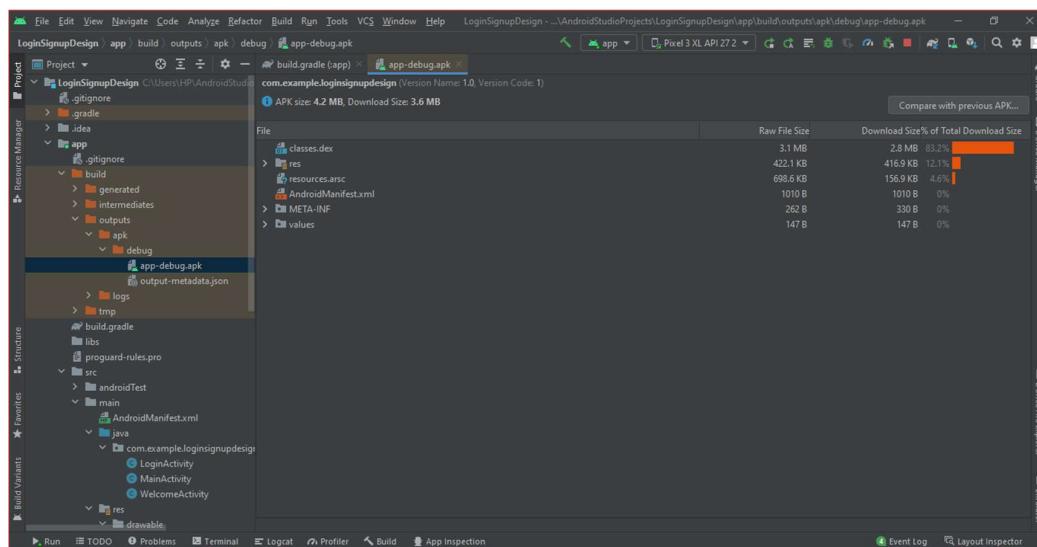


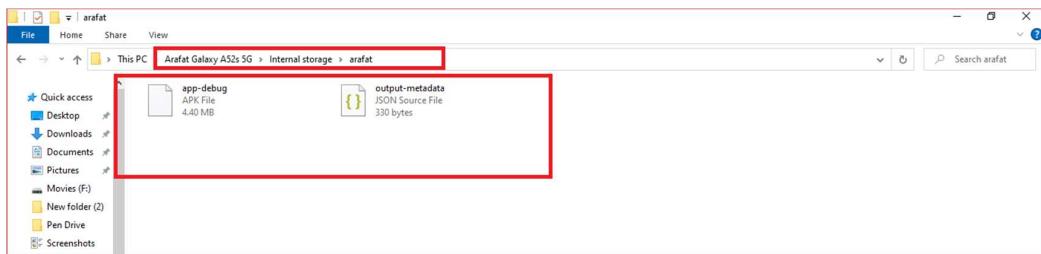
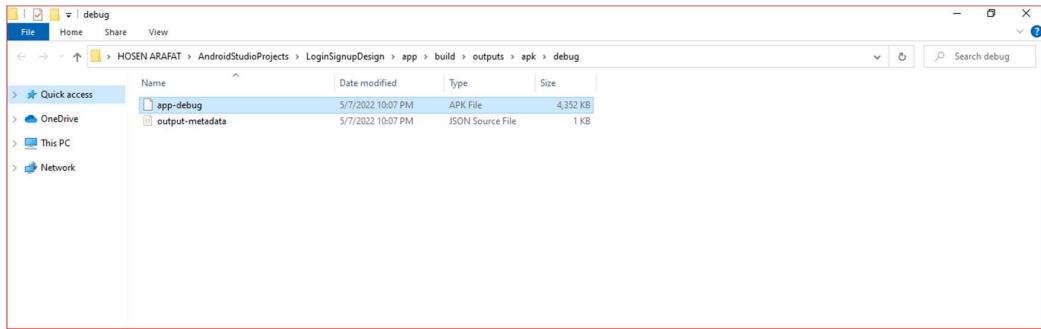
4.7. Click next, select Build Variants,Signature Version and then click finish



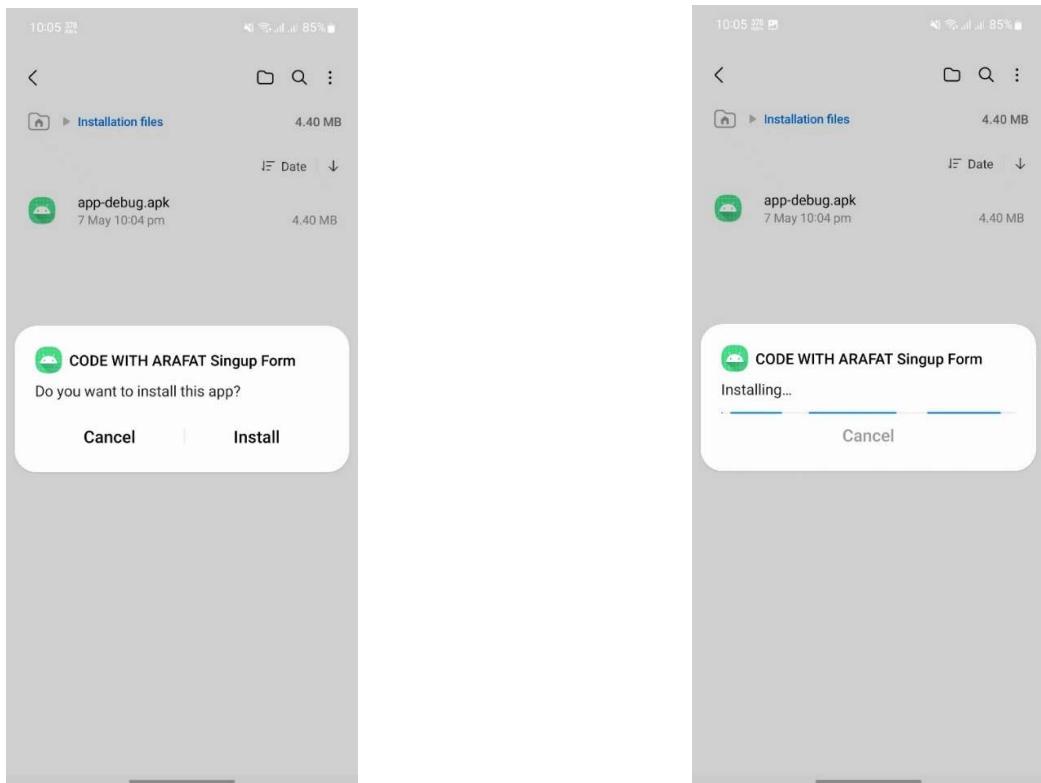


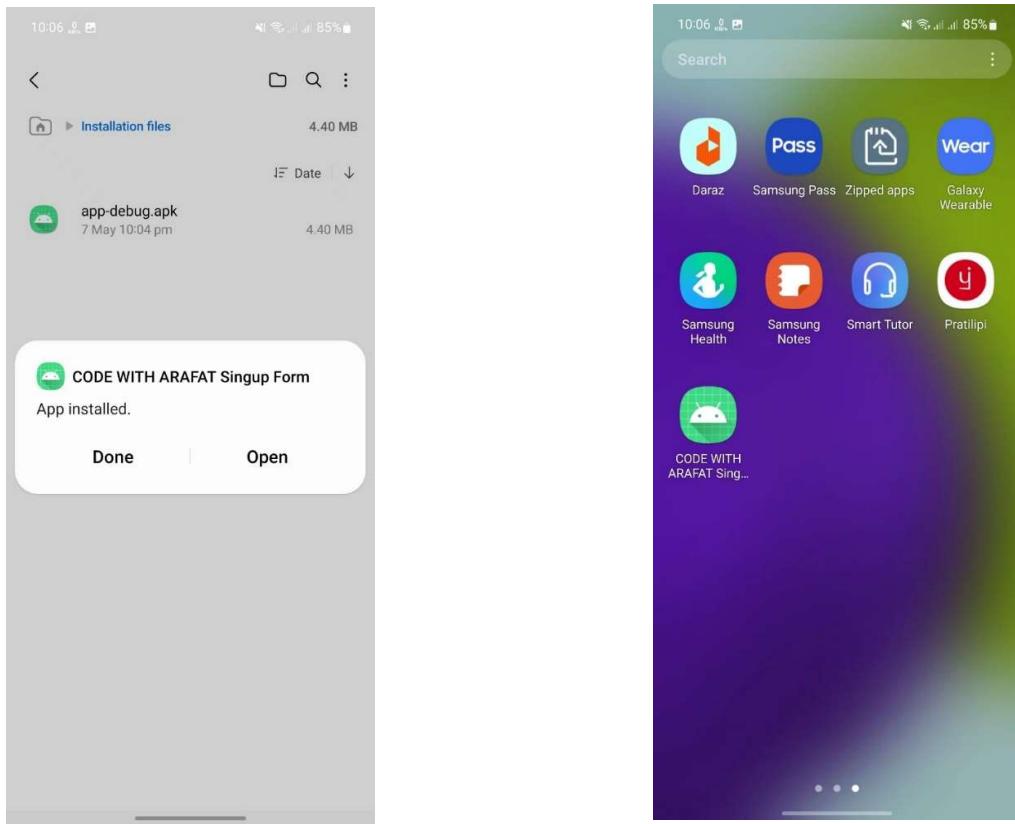
4.8. Click app > debug to find the APK file and copy it to the phone.



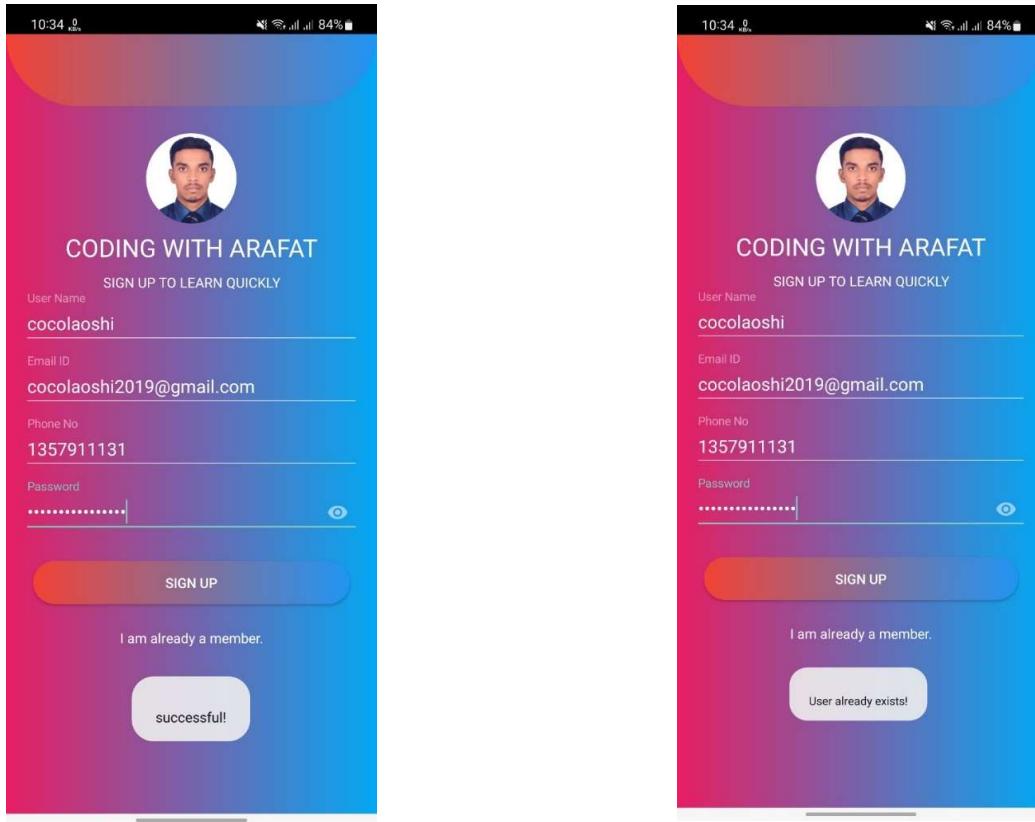


4.9. Download and install the program on my mobile.





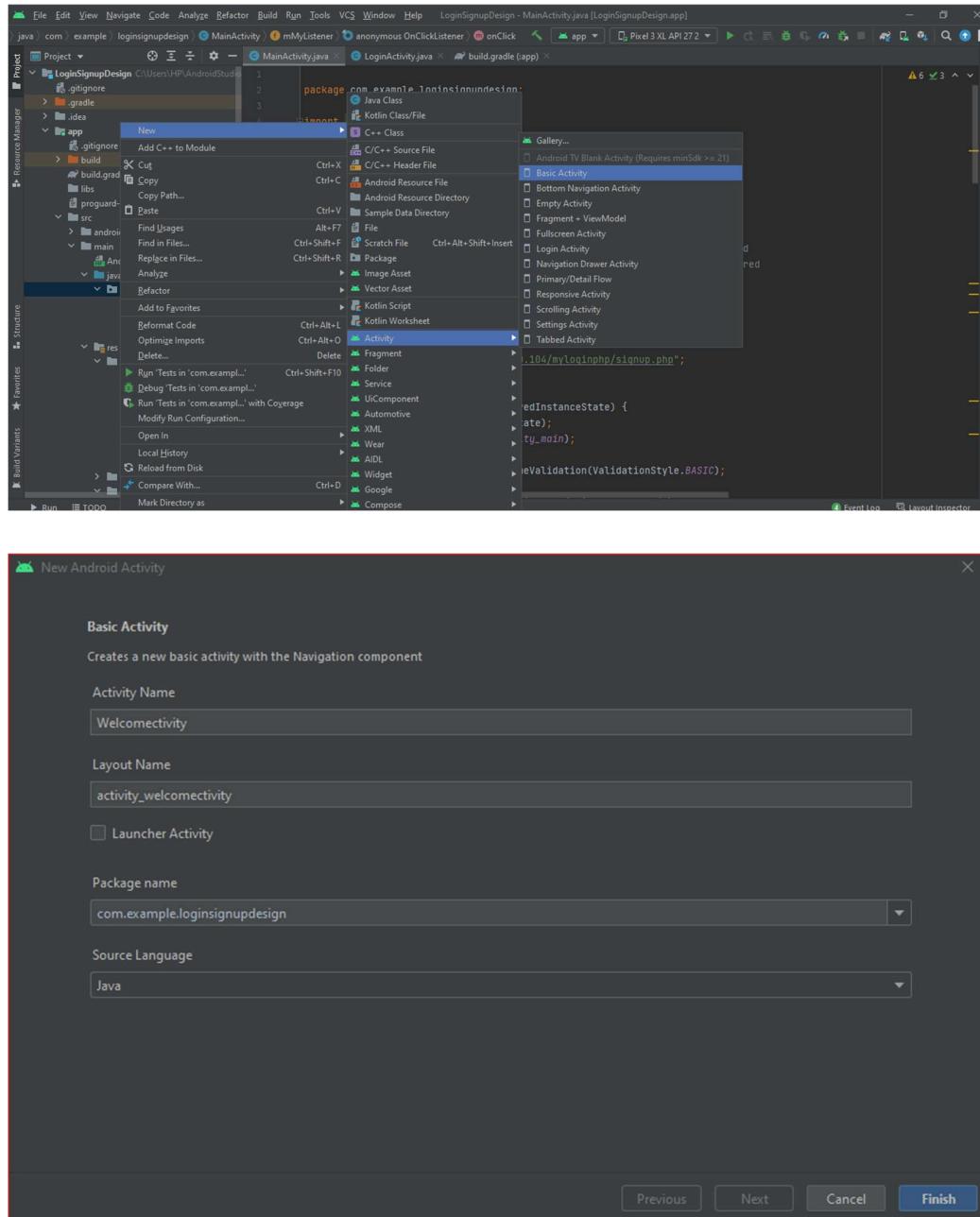
4.10. My first app is finished!



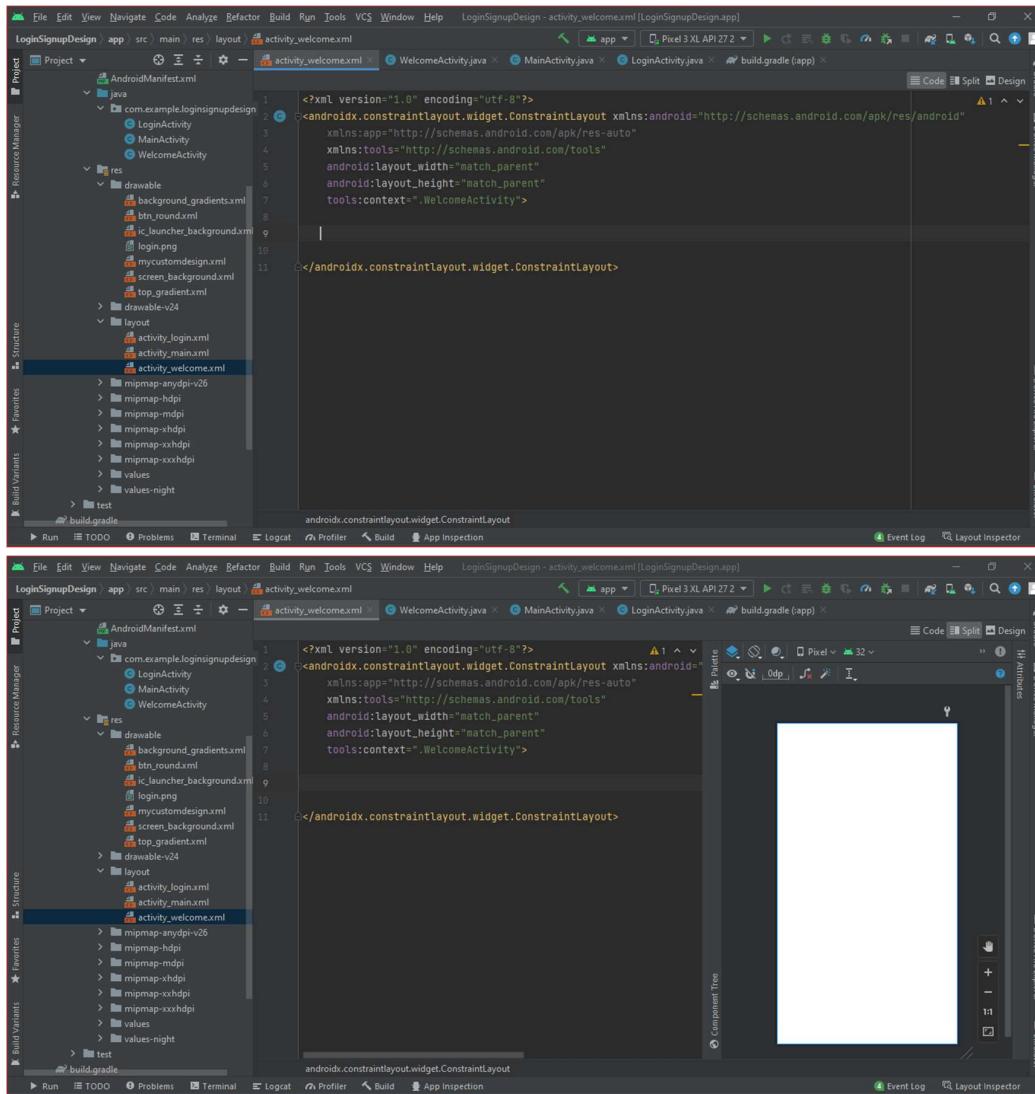
III. building MYSQL database environment

1. Complete the basic user authentication and jump

1.1. New login success page.



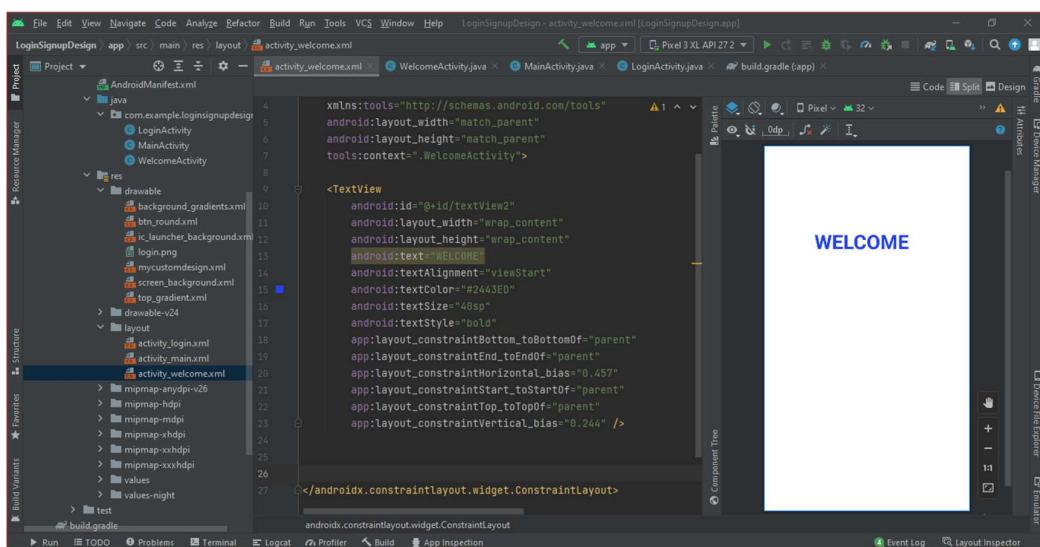
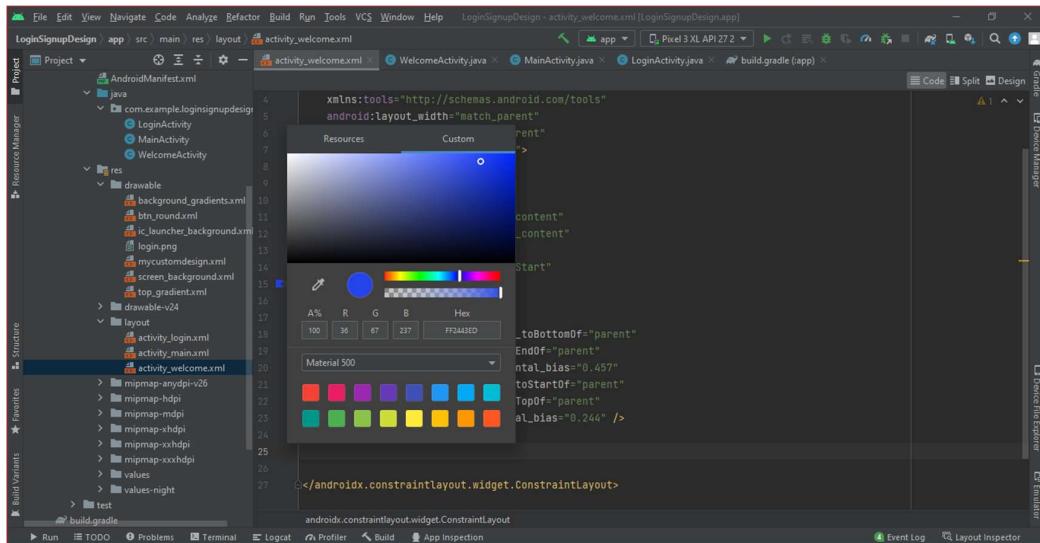
1.2. Modify the login Activity_welcome.xml page, and modify the code to display "Welcome by ARAFAT (199076003)". as shown in the figure below.

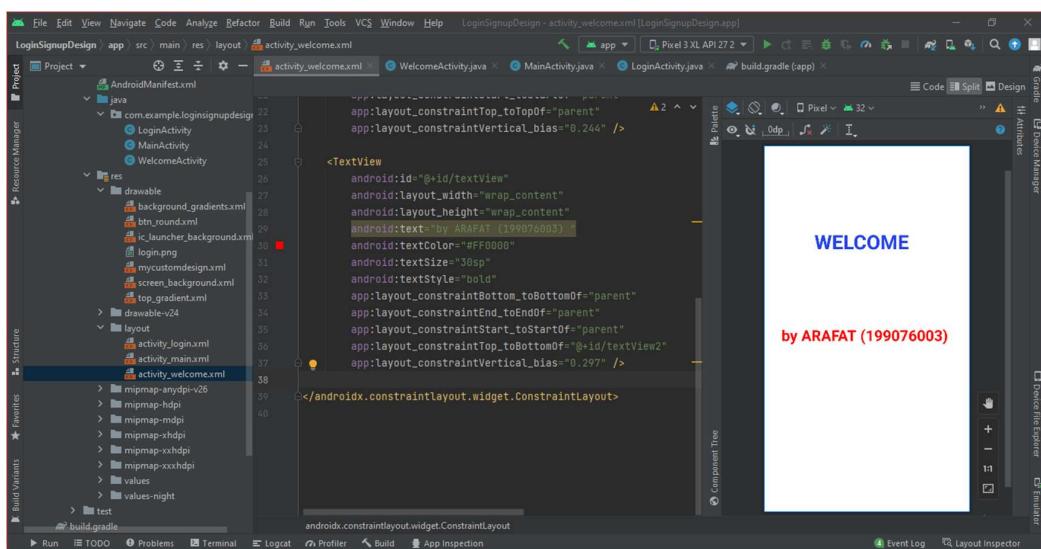
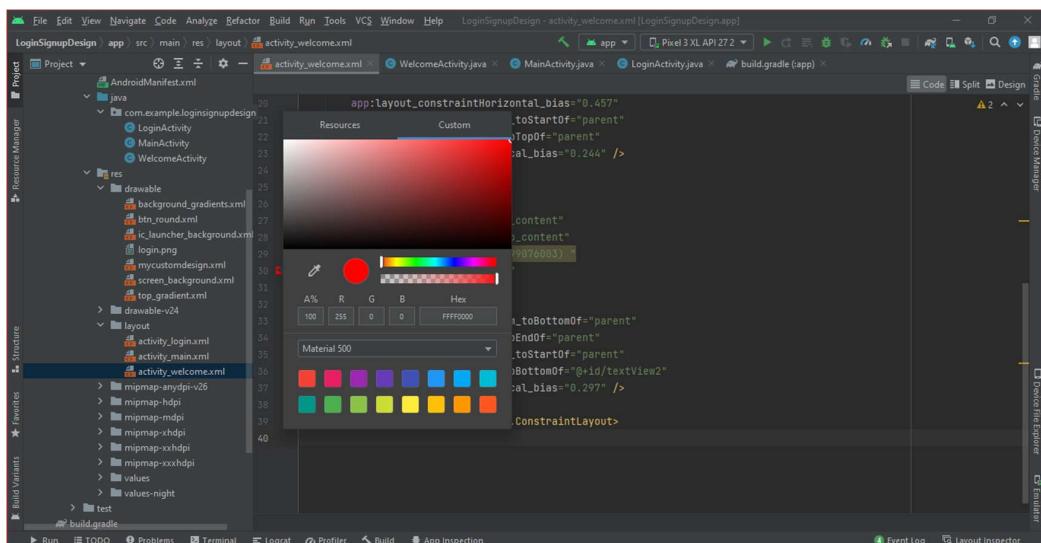
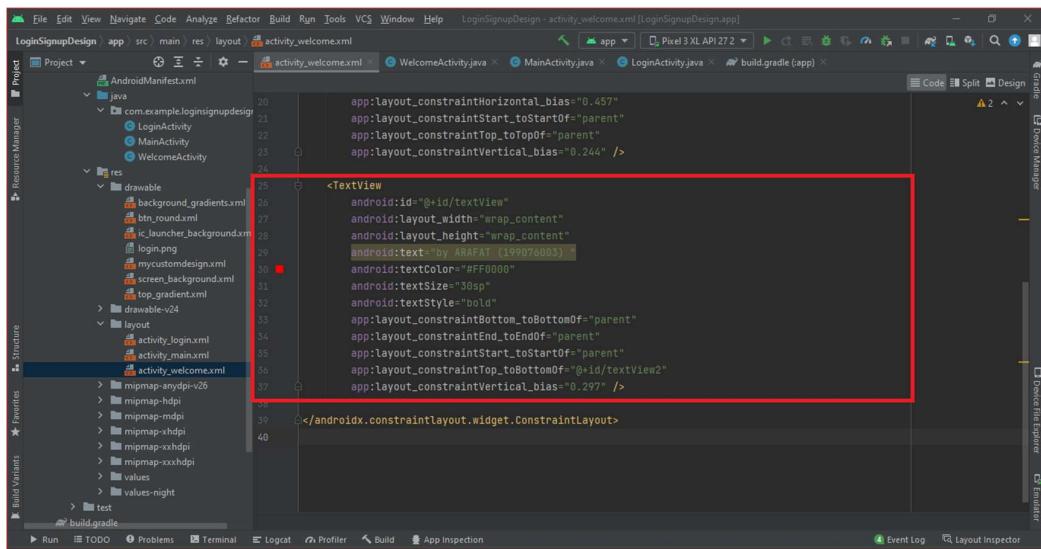


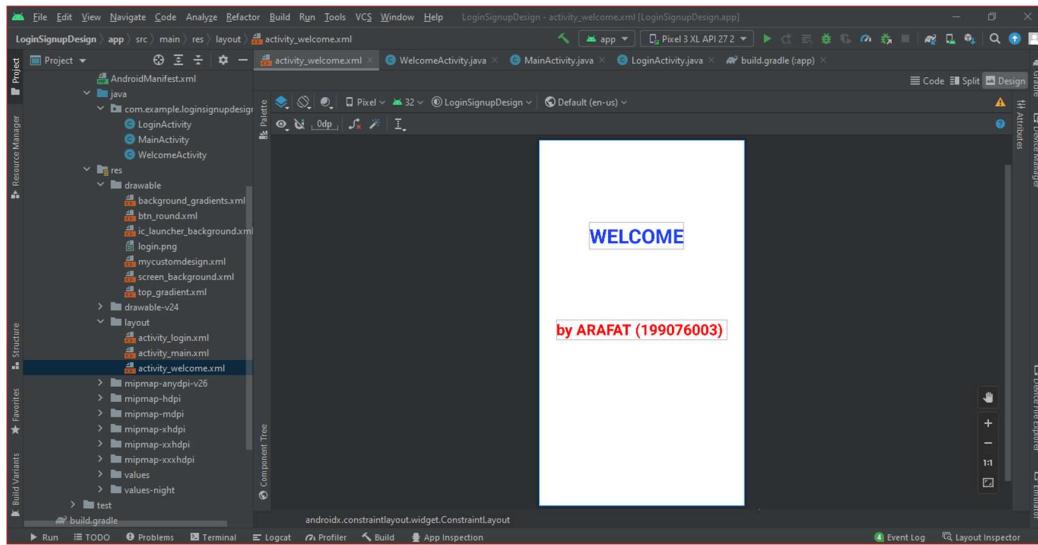
The screenshot shows the Android Studio interface with the project 'LoginSignUpDesign' open. The left sidebar displays the project structure, including the AndroidManifest.xml file and several Java files like LoginActivity, MainActivity, and WelcomeActivity. The main editor area shows the XML code for the 'activity_welcome.xml' layout. A red box highlights the section of code where the text 'WELCOME' is defined within a `<textView>` element.

```
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="WELCOME"
        android:textAlignment="ViewStart"
        android:textColor="#2445ED"
        android:textSize="40sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.457"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.244" />
    </androidx.constraintlayout.widget.ConstraintLayout>

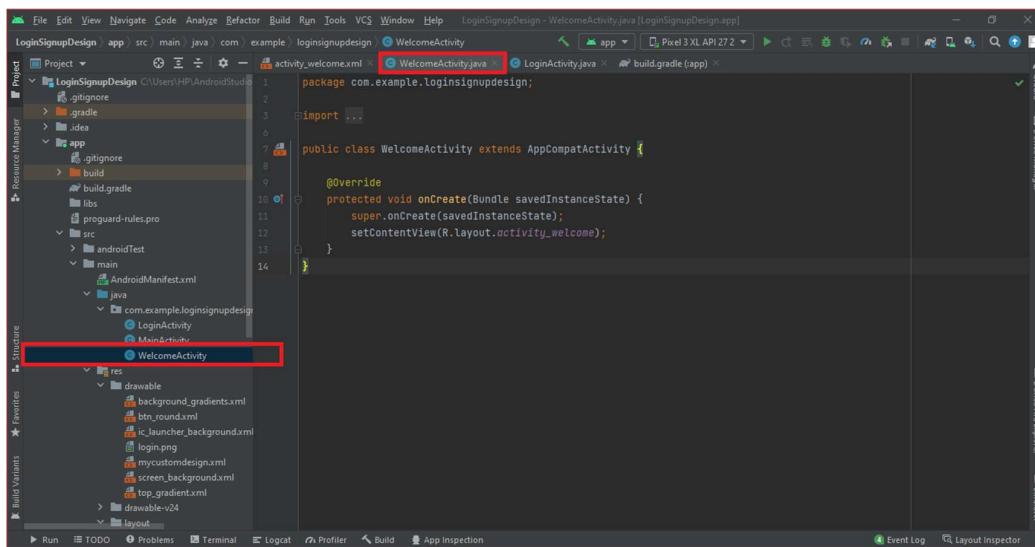
```







1.3.In and MainActivity.java Create in the same folder welcomeactivity.java File, and write code, guide the activity_welcome.xml Page written. as shown in the figure below.



The screenshot shows the Android Studio interface with the following details:

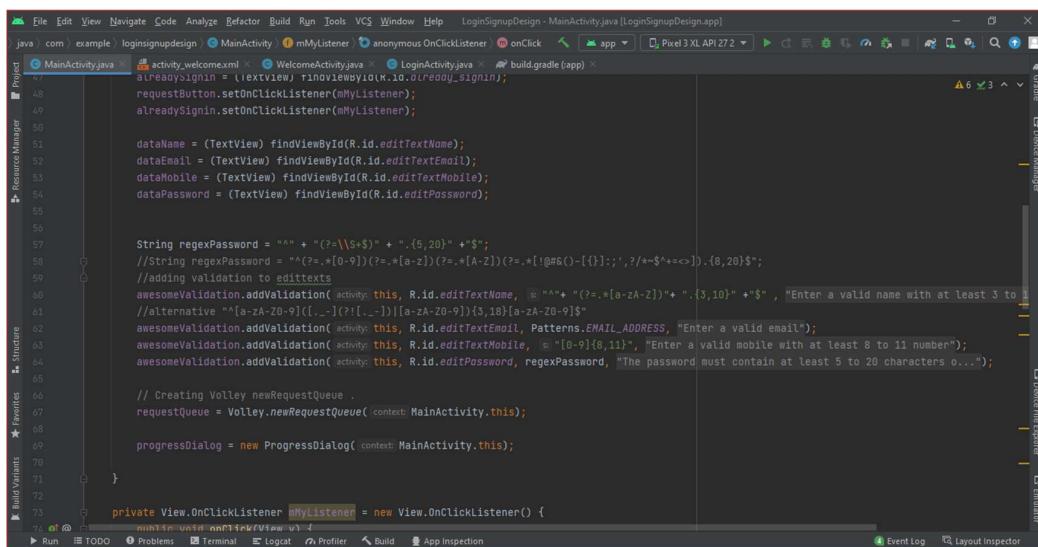
- Project Bar:** Shows the project name "LoginSignupDesign" and the file "WelcomeActivity.java" is selected.
- Project Tree:** Displays the project structure:
 - app
 - build
 - src
 - AndroidManifest.xml
 - java
 - com.example.loginsignupdesign
 - LoginActivity
 - MainActivity
 - WelcomeActivity
 - res
 - drawable
 - background_gradients.xml
 - btn_round.xml
 - ic_launcher_background.xml
 - login.png
 - mycustomdesign.xml
 - screen_background.xml
 - top_gradient.xml
 - layout
 - Code Editor:** Shows the Java code for WelcomeActivity.java:

```
package com.example.loginsignupdesign;
import ...;

public class WelcomeActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_welcome);
    }
}
```
 - Bottom Navigation:** Includes tabs for Run, TODO, Problems, Terminal, Logcat, Profiler, App Inspection, Event Log, and Layout Inspector.

1.4.stay MainActivity.java When I write user name, valid email id, valid phone no, and password must contain at least 5 to 20 characters. Then click sing up button. Show the “successful!” in the below. When i click Sing up button again show the “user already exists”. judge whether the user name= Yafei, email id=yafei2023@gmail.com, phone no= 24487654567, and password= Yafei23Ahut . as shown in the figure below.



The screenshot shows the Android Studio interface with the code editor open to MainActivity.java. The code implements a sign-up logic using Volley for network requests and AwesomeValidation for form validation. It includes validation rules for name, email, mobile, and password, and handles user existence checks via a progress dialog and toast messages.

```

java com.example.loginSignupDesign MainActivity.java
activity_welcome.xml WelcomeActivity.java LoginActivity.java build.gradle(app)
alreadySignIn = (TextView) findViewById(R.id.textView);
requestButton.setOnClickListener(mMyListener);
alreadySignIn.setOnClickListener(mMyListener);

dataName = (EditText) findViewById(R.id.editTextName);
dataEmail = (EditText) findViewById(R.id.editTextEmail);
dataMobile = (EditText) findViewById(R.id.editTextMobile);
dataPassword = (EditText) findViewById(R.id.editPassword);

String regexPassword = "^(?=\\S+$|^.{5,20}$|^$";
//String regexPassword = "(?=.*[0-9])(?=.**[a-z])(?=.**[A-Z])(?=.*[@#&]{1})"; //^.*$|^$";
//Adding validation to edittexts
awesomeValidation.addValidation(activity: this, R.id.editTextName, "(?=.*[a-zA-Z])^.{3,10}$", "Enter a valid name with at least 3 to 10 characters");
//Alternative "[a-zA-Z0-9]([._-]?[a-zA-Z0-9]){$,18}[a-zA-Z0-9]$"
awesomeValidation.addValidation(activity: this, R.id.editTextEmail, Patterns.EMAIL_ADDRESS, "Enter a valid email");
awesomeValidation.addValidation(activity: this, R.id.editTextMobile, "[0-9]{8,11}", "Enter a valid mobile with at least 8 to 11 numbers");
awesomeValidation.addValidation(activity: this, R.id.editPassword, regexPassword, "The password must contain at least 5 to 20 characters 0...");

// Creating Volley newRequestQueue .
requestQueue = Volley.newRequestQueue(context: MainActivity.this);

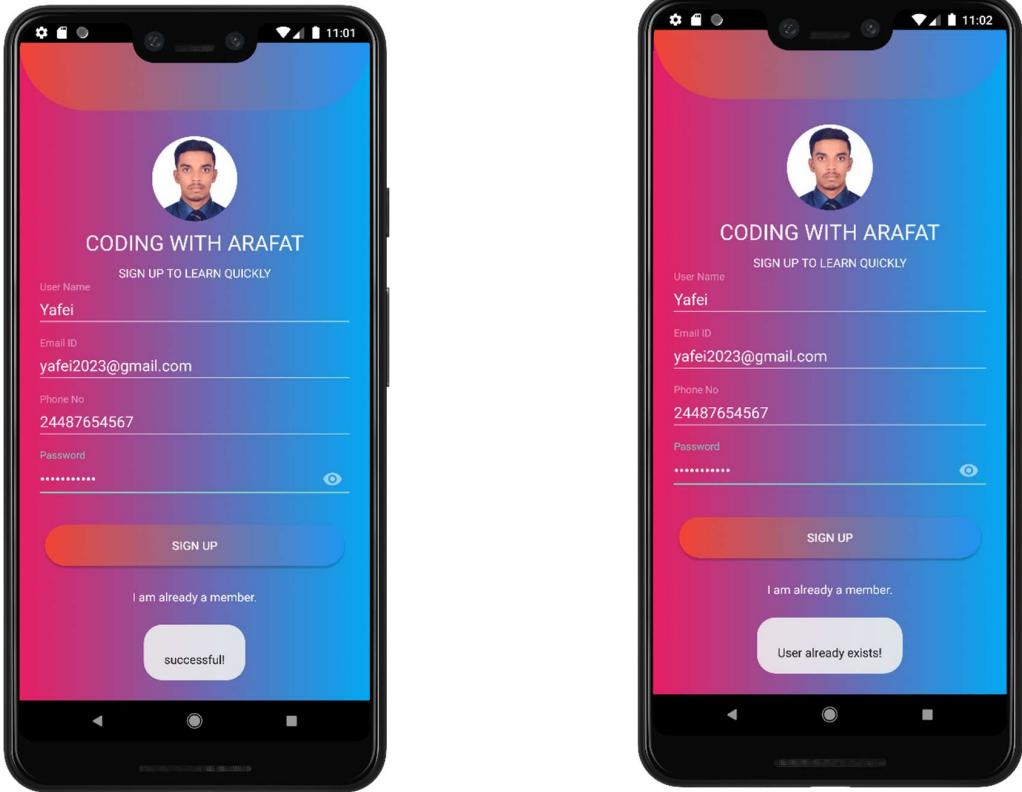
progressDialog = new ProgressDialog(context: MainActivity.this);
}

private View.OnClickListener mMyListener = new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if (v.getId() == R.id.buttonSingUp) {
            String name = dataName.getText().toString();
            String email = dataEmail.getText().toString();
            String mobile = dataMobile.getText().toString();
            String password = dataPassword.getText().toString();

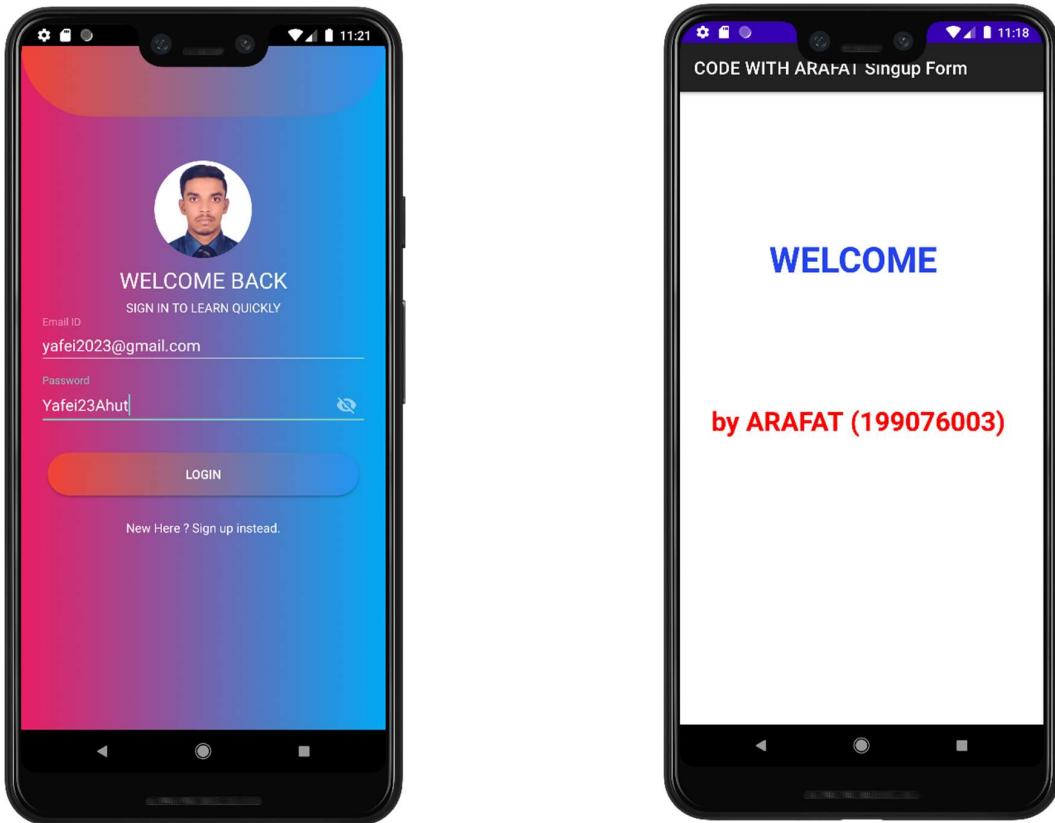
            if (!AwesomeValidation.validateName(name)) {
                AwesomeValidation.showNameError(dataName);
                return;
            }
            if (!AwesomeValidation.validateEmail(email)) {
                AwesomeValidation.showEmailError(dataEmail);
                return;
            }
            if (!AwesomeValidation.validateMobile(mobile)) {
                AwesomeValidation.showMobileError(dataMobile);
                return;
            }
            if (!AwesomeValidation.validatePassword(password)) {
                AwesomeValidation.showPasswordError(dataPassword);
                return;
            }

            AwesomeValidation.showSuccessToast("Successful!");
            AwesomeValidation.showSuccessToast("User already exists!");
        }
    }
}

```



1.5. Test with virtual machine, when the Email id and password are **Email id= yafei2023@gmail.com** and **password= Yafei23Ahut**. Then click the login button, then immediately jump to the activity welcome page .



IV. Configure Login and registration to database using PHP and MySQL:

1. Create database

1.1 Download and install XAMPP (The whole process needs network)

1.1.1. Open the website of XAMPP , Click DOWNLOAD

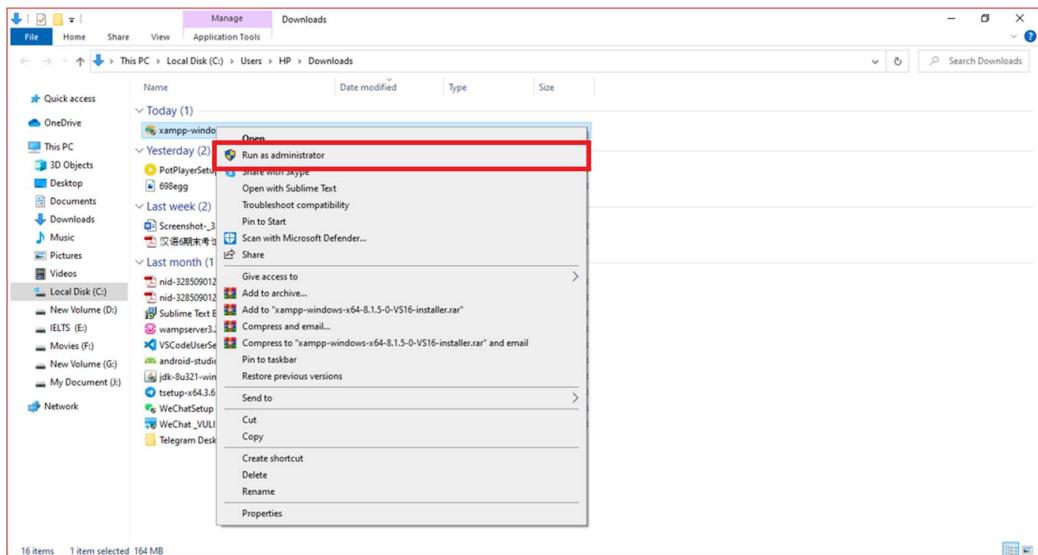
XAMPP to download.

(<https://www.apachefriends.org/download.html>)

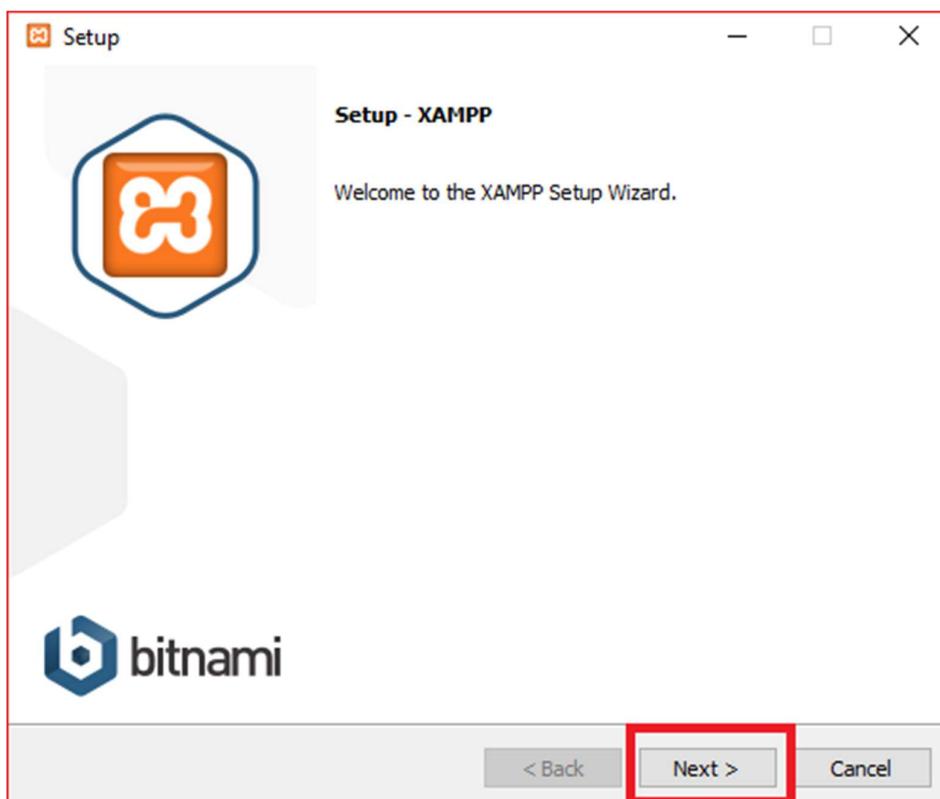
The screenshot shows the Apache Friends Download page. The 'Download' button in the top navigation bar is highlighted with a red box. The main content area is titled 'Download' and features a heading 'XAMPP'. Below it, a text block says 'XAMPP is an easy to install Apache distribution containing MariaDB, PHP, and Perl. Just download and start the installer. It's that easy.' To the right, there's a 'Documentation/FAQs' sidebar with a note about no manual and links to forums and Stack Overflow. The central part of the page displays a table for XAMPP for Windows versions 7.4.29, 8.0.18, and 8.1.5. The table columns are 'Version', 'Checksum', and 'Size'. Each row includes a 'What's Included?' link, an MD5/SHA1 hash pair, and a 'Download (64 bit)' button. The 8.1.5 row is also highlighted with a red box.

Version	Checksum	Size		
7.4.29 / PHP 7.4.29	What's Included?	md5 sha1	Download (64 bit)	159 Mb
8.0.18 / PHP 8.0.18	What's Included?	md5 sha1	Download (64 bit)	161 Mb
8.1.5 / PHP 8.1.5	What's Included?	md5 sha1	Download (64 bit)	164 Mb

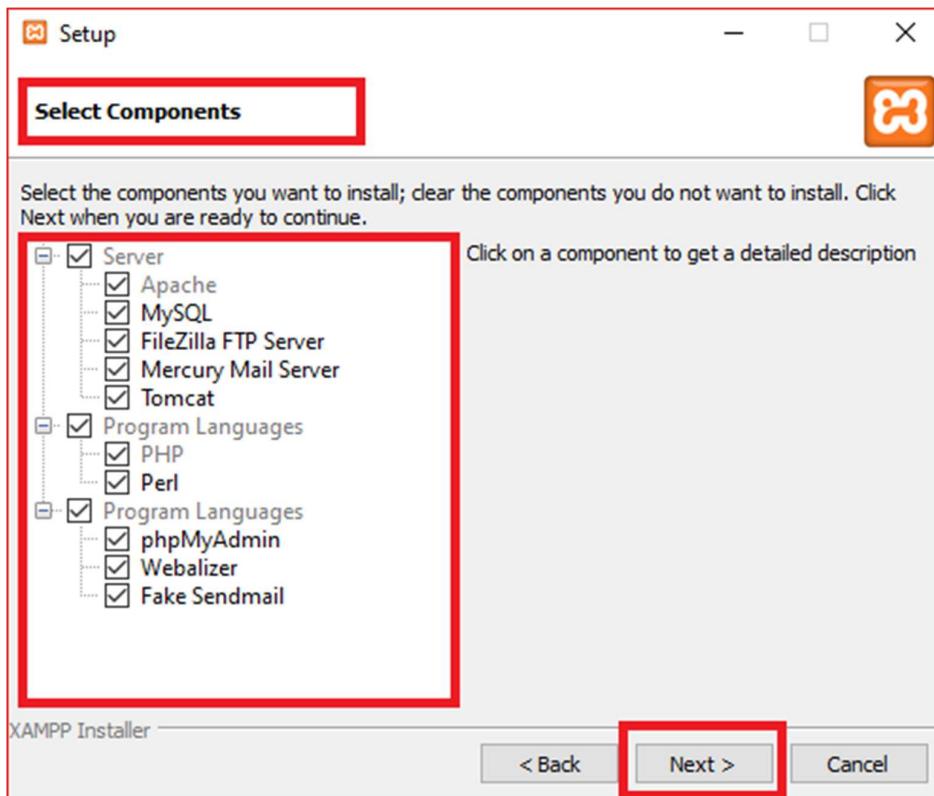
1.1.2. Run the installation package (in administrator mode).



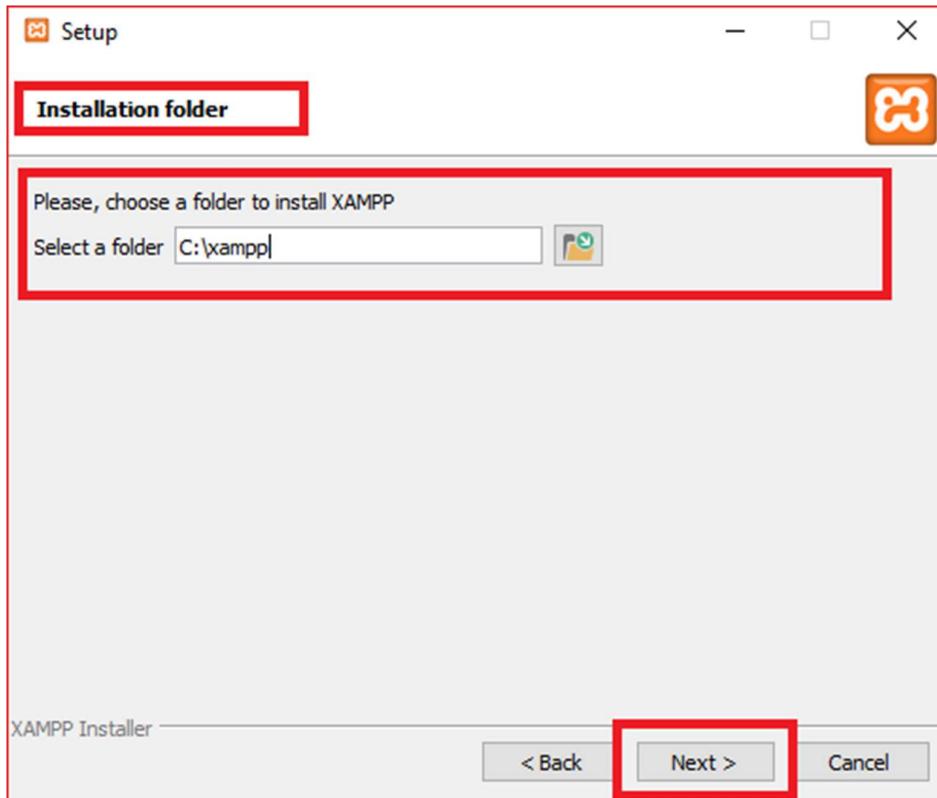
1.1.3. Always click 'next' to install.



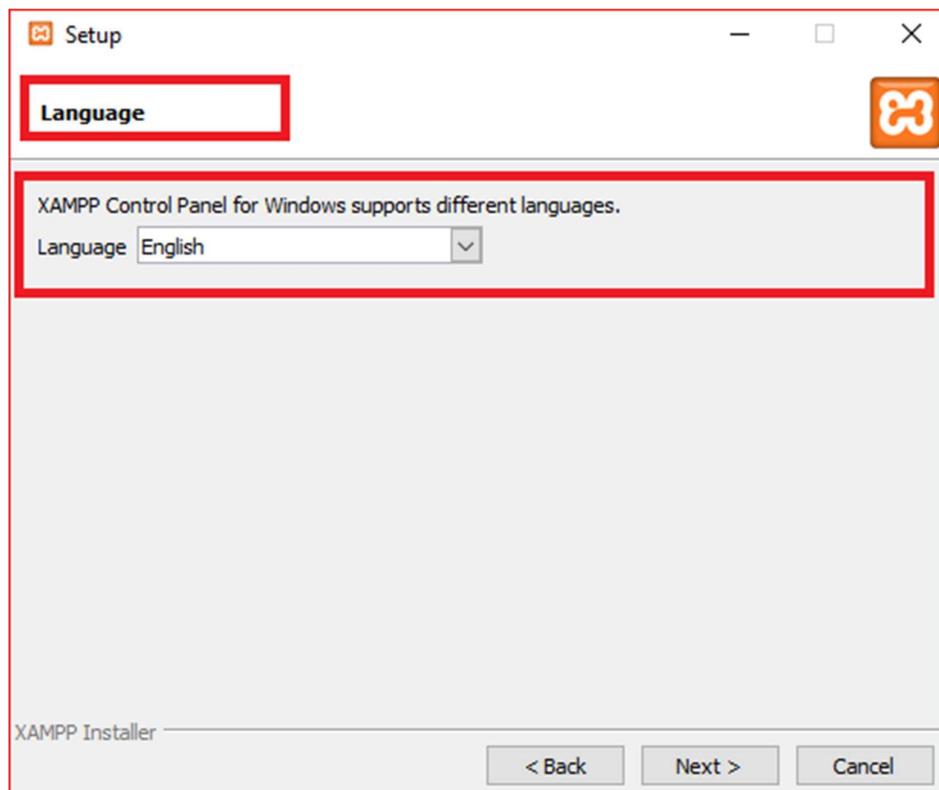
1.1.4. Select all Components “ XAMPP ” Then Click Next.



1.1.5. Choose Installation folder of the XAMPP . Then Click Next.

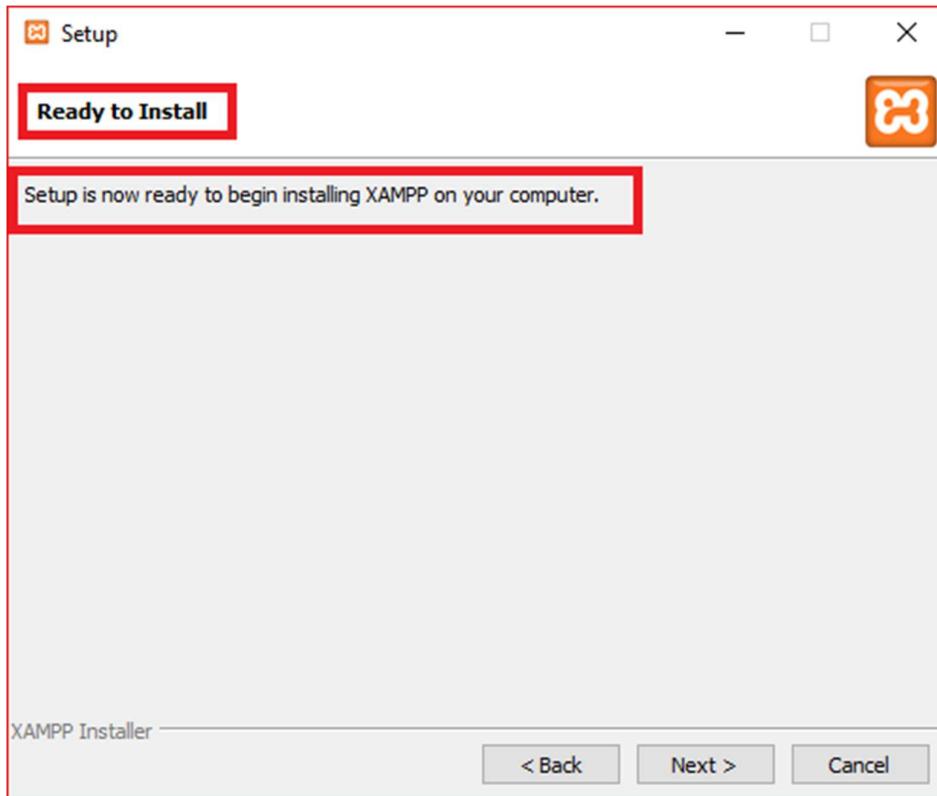


1.1.6. Set Language of the XAMPP . Then Click Next.

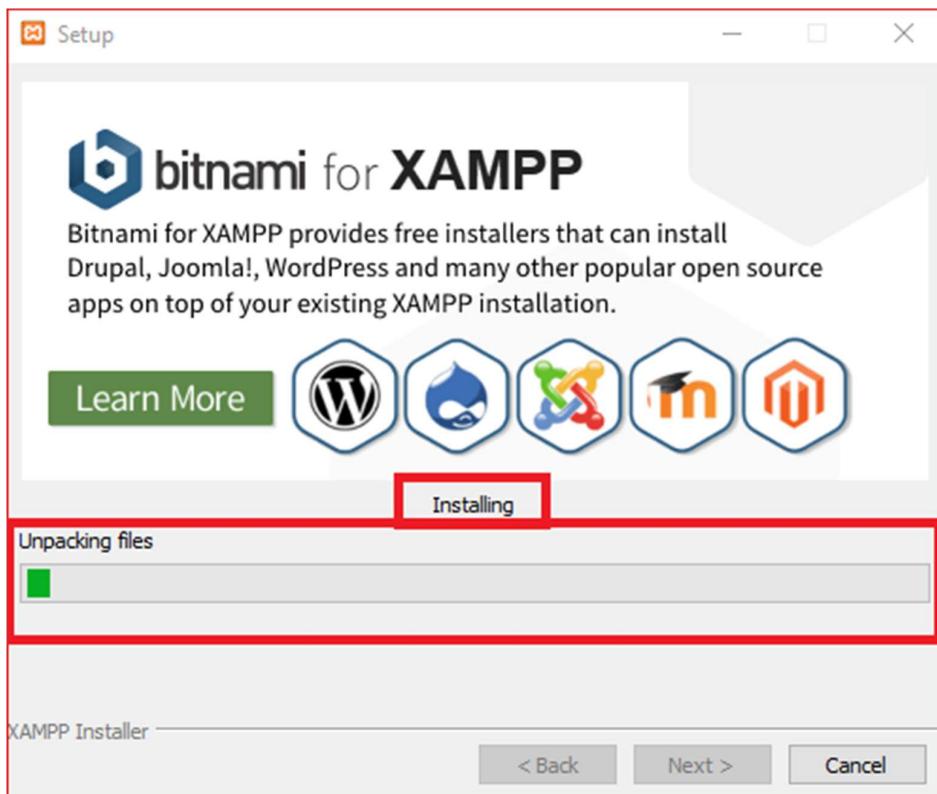


1.1.7. Setup is now Ready to being installing XAMPP of my computer.

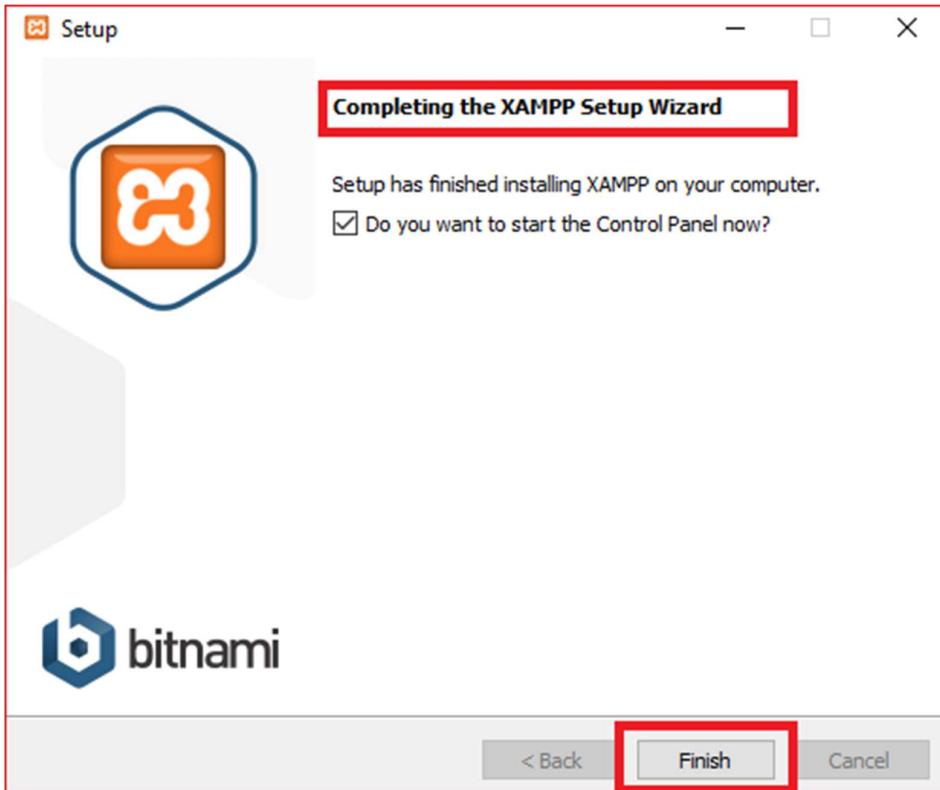
Then Click Next.



1.1.8. Installing XAMPP , Unpacking files.



1.1.9. Now finish the XAMPP installing. Click finish.



1.1.10. Now open XAMPP control panel.



```
2:25:28 PM [main] Initializing Control Panel
2:25:28 PM [main] Windows Version: Enterprise 64-bit
2:25:28 PM [main] XAMPP Version: 8.1.5
2:25:28 PM [main] Control Panel Version: 3.3.0 [ Compiled: Apr 6th 2021 ]
2:25:28 PM [main] You are not running with administrator rights! This will work for
2:25:28 PM [main] most application stuff but whenever you do something with services
2:25:28 PM [main] there will be a security dialogue or things will break! So think
2:25:28 PM [main] about running this application with administrator rights!
2:25:28 PM [main] XAMPP Installation Directory: "c:\xampp\"
2:25:28 PM [main] Checking for prerequisites
2:25:28 PM [main] All prerequisites found
2:25:28 PM [main] Initializing Modules
2:25:28 PM [main] Starting Check-Timer
2:25:28 PM [main] Control Panel Ready
```

1.1.11. Start Apache.

The screenshot shows the XAMPP Control Panel interface. The Apache service is listed with PID 6332 and Port 80,443. The 'Actions' column contains buttons for Stop, Admin, Config, Logs, and Shell. The 'Shell' button is highlighted with a red box. Below the table is a log window displaying the following text:

```
2.25.28 PM [main] Initializing Control Panel
2.25.28 PM [main] Windows Version: Enterprise 64-bit
2.25.28 PM [main] XAMPP Version: 8.1.5
2.25.28 PM [main] Control Panel Version: 3.3.0 [ Compiled: Apr 6th 2021 ]
2.25.28 PM [main] You are not running with administrator rights! This will work for
most application stuff but whenever you do something with services
there will be a security dialogue or things will break! So think
about running this application with administrator rights!
2.25.28 PM [main] XAMPP Installation Directory: "c:\xampp"
2.25.28 PM [main] Checking for prerequisites
2.25.28 PM [main] All prerequisites found
2.25.28 PM [main] Initializing Modules
2.25.28 PM [main] Starting Check-Timer
2.25.28 PM [main] Control Panel Ready
2.26.59 PM [Apache] Attempting to start Apache app...
2.26.59 PM [Apache] Status change detected: running
2.27.10 PM [mysql] Attempting to start MySQL app...
2.27.11 PM [mysql] Status change detected: running
2.33.48 PM [Apache] Attempting to stop Apache (PID: 5328)
2.33.48 PM [Apache] Attempting to stop Apache (PID: 976)
2.33.48 PM [Apache] Status change detected: stopped
2.33.49 PM [mysql] Attempting to stop MySQL app...
2.33.50 PM [mysql] Status change detected: stopped
2.34.25 PM [Apache] Attempting to start Apache app...
2.34.25 PM [Apache] Status change detected: running
```

1.1.12. Start MySQL

The screenshot shows the XAMPP Control Panel interface. The MySQL service is listed with PID 6836 and Port 3306. The 'Actions' column contains buttons for Stop, Admin, Config, Logs, and Explorer. The 'Stop' button is highlighted with a red box. Below the table is a log window displaying the following text:

```
2.25.28 PM [main] Initializing Control Panel
2.25.28 PM [main] Windows Version: Enterprise 64-bit
2.25.28 PM [main] XAMPP Version: 8.1.5
2.25.28 PM [main] Control Panel Version: 3.3.0 [ Compiled: Apr 6th 2021 ]
2.25.28 PM [main] You are not running with administrator rights! This will work for
most application stuff but whenever you do something with services
there will be a security dialogue or things will break! So think
about running this application with administrator rights!
2.25.28 PM [main] XAMPP Installation Directory: "c:\xampp"
2.25.28 PM [main] Checking for prerequisites
2.25.28 PM [main] All prerequisites found
2.25.28 PM [main] Initializing Modules
2.25.28 PM [main] Starting Check-Timer
2.25.28 PM [main] Control Panel Ready
2.26.59 PM [Apache] Attempting to start Apache app...
2.26.59 PM [Apache] Status change detected: running
2.27.10 PM [mysql] Attempting to start MySQL app...
2.27.11 PM [mysql] Status change detected: running
2.33.48 PM [Apache] Attempting to stop Apache (PID: 5328)
2.33.48 PM [Apache] Attempting to stop Apache (PID: 976)
2.33.48 PM [Apache] Status change detected: stopped
2.33.49 PM [mysql] Attempting to stop MySQL app...
2.33.50 PM [mysql] Status change detected: stopped
2.34.25 PM [Apache] Attempting to start Apache app...
2.34.25 PM [Apache] Status change detected: running
2.34.55 PM [mysql] Attempting to start MySQL app...
2.34.55 PM [mysql] Status change detected: running
```

1.1.13. Click Admin then go to “phpMyAdmin” .

XAMPP Control Panel v3.3.0 [Compiled: Apr 6th 2021]

Modules	Service	Module	PID(s)	Port(s)	Actions
		Apache	6332 4932	80, 443	Stop Admin Config Logs
		MySQL	6836	3306	Stop Admin Config Logs
		FileZilla			Start Admin Config Logs
		Mercury			Start Admin Config Logs
		Tomcat			Start Admin Config Logs

Log Output:

```

2.25.28 PM [main] Initializing Control Panel
2.25.28 PM [main] Windows Version: Enterprise 64-bit
2.25.28 PM [main] XAMPP Version: 8.1.5
2.25.28 PM [main] Control Panel Version: 3.3.0 [ Compiled: Apr 6th 2021 ]
2.25.28 PM [main] You are not running with administrator rights! This will work for
most application stuff but whenever you do something with services
there will be a security dialogue or things will break! So think
about running this application with administrator rights!
2.25.28 PM [main] XAMPP Installation Directory: "c:\xampp"
2.25.28 PM [main] Checking for prerequisites
2.25.28 PM [main] All prerequisites found
2.25.28 PM [main] Initializing Modules
2.25.28 PM [main] Starting Check-Timer
2.25.28 PM [main] Control Panel Ready
2.26.59 PM [Apache] Attempting to start Apache app...
2.26.59 PM [Apache] Status change detected: running
2.27.10 PM [mysql] Attempting to start MySQL app...
2.27.11 PM [mysql] Status change detected: running
2.33.48 PM [Apache] Attempting to stop Apache (PID: 5328)
2.33.48 PM [Apache] Status change detected: stopped
2.33.49 PM [mysql] Attempting to stop MySQL app...
2.33.50 PM [mysql] Status change detected: stopped
2.34.25 PM [Apache] Attempting to start Apache app...
2.34.25 PM [Apache] Status change detected: running
2.34.55 PM [mysql] Attempting to start MySQL app...
2.34.55 PM [mysql] Status change detected: running

```


phpMyAdmin

Recent Favorites

General settings

Appearance settings

Database server

Web server

phpMyAdmin

1.1.14. Now Create a Database Folder "singupdb" show query box. and create table users (name, email, mobile, password). as shown in the figure below.

```

CREATE DATABASE singupdb;
CREATE TABLE users(
    name VARCHAR(30) NOT NULL,
    email VARCHAR(30) NOT NULL,
    mobile VARCHAR(50) NOT NULL,
    password VARCHAR(50) NOT NULL
);

```

Server: MySQL_3306 > Database: signupdb > Table: users

Structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	name	varchar(30)	latin1_swedish_ci		No	None			Change Drop More
2	email	varchar(30)	latin1_swedish_ci		No	None			Change Drop More
3	mobile	varchar(50)	latin1_swedish_ci		No	None			Change Drop More
4	password	varchar(50)	latin1_swedish_ci		No	None			Change Drop More

Indexes

No index defined!

Create an index on 1 columns Go

Server: MySQL_3306 > Database: signupdb > Table: users

Structure

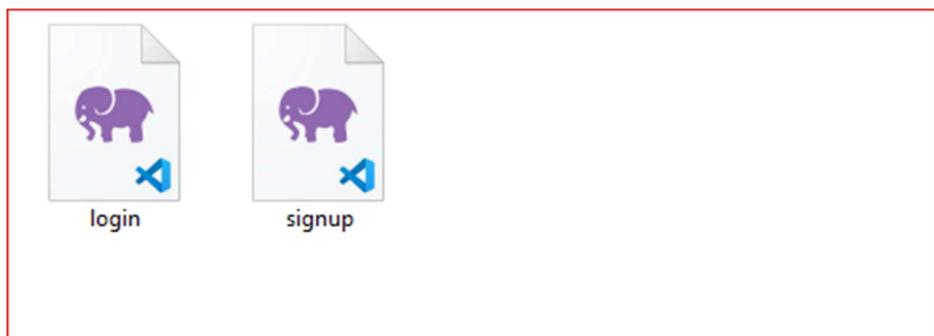
+ Options	name	email	mobile	password
Arafat	arafat2018@gmail.com	137448056068	Arafat1234	
Arafat	arafat2011@gmail.com	137448056067	Arafat12345	
baba	baba12@gmail.com	2345678901	baba12x	
maa	maa@gmail.com	12345678901	maa123456	
shakila	shakila@gmail.vom	12345678902	shakila12345	
shakila	shakila2014@gmail.com	01747679161	Shakila123@	
kayes	kayes2010@gmail.com	23456789051	kayes123@	
shazubi	shazubi2002@gmail.com	0178965434	shazubi2914	
himel	himel2001@gmail.com	1234567890	himel2001	
maruf	maruf2002@gmail.com	0178694534	maruf3456	
Tania	tania1996@gmail.com	0195647890	tania123456	
Arafat	arafat2001@gmail.com	15655561781	Arafat2914	
Arafat	Arfa420ahut@gmail.com	1234567894	Ahu2019	
cocoloshi	cocoloshi2019@gmail.com	1357911131	cocoloshi123456	
manki	manki2023@gmail.com	10987654367	manki980	
Yafei	yafei2023@gmail.com	24487654567	Yafei23Ahut	

Query results operations

Print Copy to clipboard Export Display chart Create view

2. Add insert function

2.1.Create a loginphp folder inside of the htdocs folder. And create loginin.php and signup.php inside of this folder. And write code for loginin.php and signup.php.



```
login.php ×
login.php
1 <?php
2
3 $con=new mysqli("localhost","root","","signupdb");
4 $st_check=$con->prepare("select * from users where email=? and password=?");
5 $st_check->bind_param("ss", $_GET["email"],$_GET["password"]);
6 $st_check->execute();
7 $rs=$st_check->get_result();
8 if($rs->num_rows==0)
9 |
echo "0";
10 else
11 |
echo "1";
12 ?>
```

```
signup.php ●
signup.php
1 <?php
2
3 $con=new mysqli("localhost","root","","signupdb");
4 $st_check=$con->prepare("select * from users where mobile=?");
5 $st_check->bind_param("s", $_GET["mobile"]);
6 $st_check->execute();
7 $rs=$st_check->get_result();
8 if($rs->num_rows==0)
9 {
10 $st=$con->prepare("insert into users values(?, ?, ?, ?)");
11 $st->bind_param("ssss", $_GET["name"],$_GET["email"],$_GET["mobile"],$_GET["password"]);
12 $st->execute();
13 | echo "successful!";
14 }
15 else
16 | echo "User already exists!";
17 ?>
```

3.connect to database using IP address

3.1. Now connect the Android app with database using my local server, I mean by using our ip address we can connect out android app with Signin.php and signup.php to database.

```
Microsoft Windows [Version 10.0.19044.1645]
(c) Microsoft Corporation. All rights reserved.

C:\Users\HP>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet 2:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix . :

Wireless LAN adapter Local Area Connection* 11:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix . :

Wireless LAN adapter Local Area Connection* 12:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix . :

Wireless LAN adapter Wi-Fi 2:

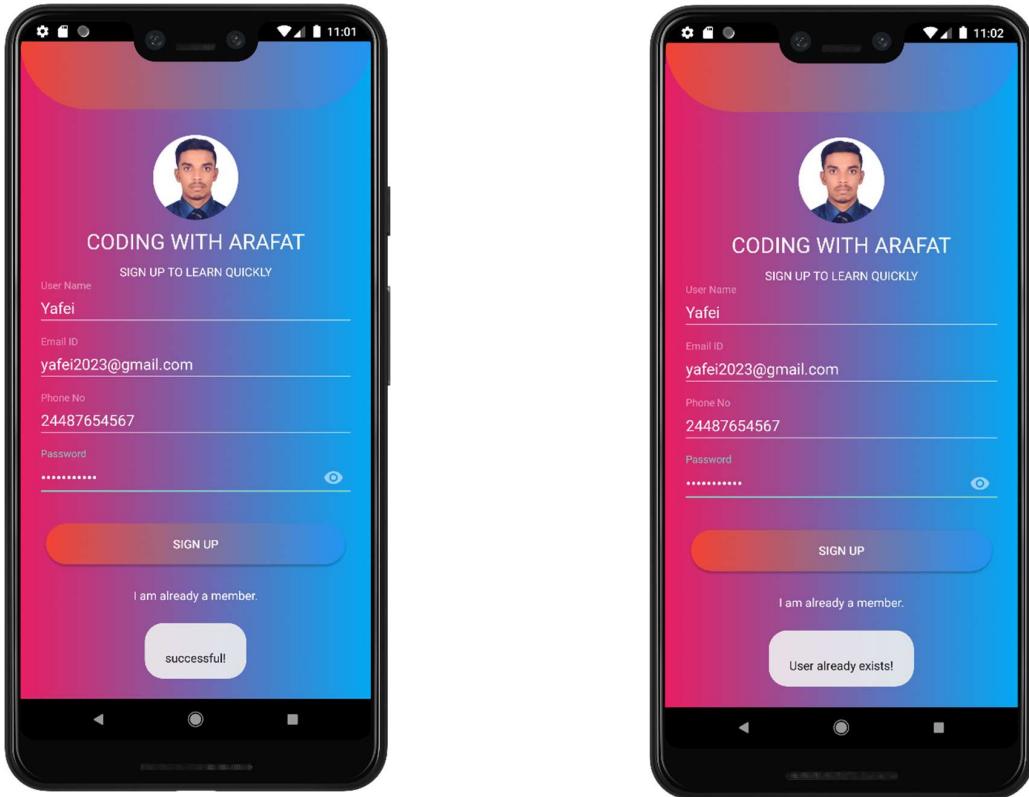
    Connection-specific DNS Suffix . :
    Link-local IPv6 Address . . . . . : fe80::790a:fca1:9a26:6a6c%9
    IPv4 Address . . . . . : 192.168.0.103
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.0.1
```

```
// Storing server url into String variable.
String HttpUrl = "http://192.168.0.103/myloginphp/signup.php";
```

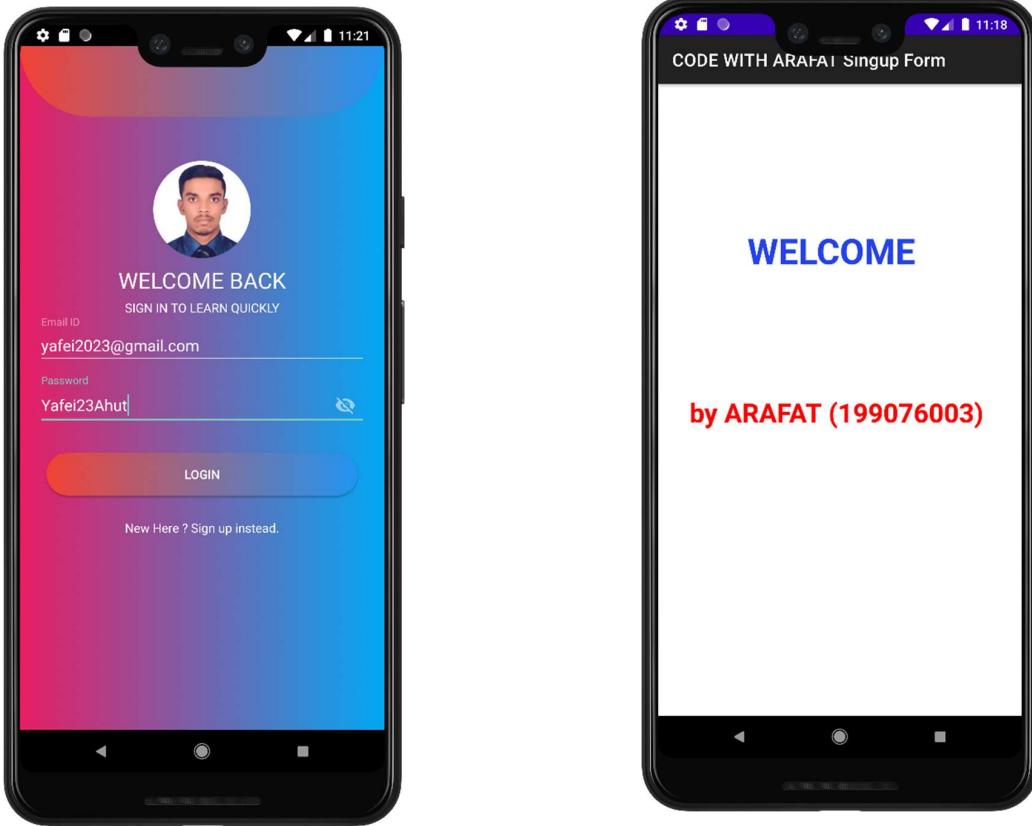
```
String myurl = "http://192.168.0.103/myloginphp/signup.php?name=" + NameHolder +
    "&email=" + EmailHolder + "&mobile=" + MobileHolder + "&password=" + PasswordHolder;
```

4. Testing

4.1. Login interface



4.2. Registration interface.



V. curriculum requirements

Project Name: login registration system

1. I user Using Java, php and Mysql database for storing users information
2. I create a user interface, Where user must have to input at least 3 characters or digits to maximum 10 characters or digit. If user input 2 or 3 characters /digits then they won't be able to register and login.

Regular expression for this username:

"^"(?=.*[a-zA-Z])".{3,10}"+"

4. Regular expression for password. Minimum 5 character/ digits to maximum 20 character/digits . Regular expression for this: "^(?=\\S+\\\$)".{5,20}"+"

5. I use Beautiful and practical software interface and amazing UI design in my app.

Experimental report submission:

I have submitted project folder, database folder myloginphp, database information and APK file.

ALL CODES:

login.php:

```
<?php

$con=new mysqli("localhost","root","","","signupdb");

$st_check=$con->prepare("select * from users where email=? and password=?");

$st_check->bind_param("ss", $_GET["email"],$_GET["password"]);

$st_check->execute();

$rs=$st_check->get_result();

if($rs->num_rows==0)

    echo "0";

else

    echo "1";

?>
```

signup.php:

```
<?php

$con=new mysqli("localhost", "root", "", "signupdb");

$st_check=$con->prepare("select * from users where mobile=?");

$st_check->bind_param("s", $_GET["mobile"]);

$st_check->execute();

$rs=$st_check->get_result();

if($rs->num_rows==0)

{

$st=$con->prepare("insert into users values(?,?,?,?,?)");
```

```
$st->bind_param("ssss", $_GET["name"],$_GET["email"],$_GET["mobile"],$_GET["password"]);  
  
$st->execute();  
  
echo "successful!";  
  
}  
  
else  
  
echo "User already exists!";  
  
?>
```

MainActivity.java:

```
package com.example.loginsignupdesign;

import android.app.AlertDialog;
import android.content.Intent;
import android.os.Bundle;
import android.util.Patterns;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;
import com.basgeekball.awesomevalidation.AwesomeValidation;
import com.basgeekball.awesomevalidation.ValidationStyle;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    Button requestButton;
```

```

    TextView alreadySignin;
    private AwesomeValidation awesomeValidation;
    TextView dataName; // a text field to display the request response
    TextView dataEmail; // a text field where the data to be sent is
entered
    TextView dataMobile; // a text field where the data to be sent is
entered
    TextView dataPassword; // a text field where the data to be sent is
entered
    RequestQueue requestQueue;
    ProgressDialog progressDialog;
    String NameHolder, EmailHolder, MobileHolder, PasswordHolder;

    // Storing server url into String variable.
    String HttpUrl = "http://192.168.0.103/myloginphp/signup.php";

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

        awesomeValidation = new
AwesomeValidation(ValidationStyle.BASIC);

        requestButton = (Button) findViewById(R.id.buttonSubmit);
        alreadySignin = (TextView) findViewById(R.id.already_signin);
        requestButton.setOnClickListener(mMyListener);
        alreadySignin.setOnClickListener(mMyListener);

        dataName = (TextView) findViewById(R.id.editTextName);
        dataEmail = (TextView) findViewById(R.id.editTextEmail);
        dataMobile = (TextView) findViewById(R.id.editTextMobile);
        dataPassword = (TextView) findViewById(R.id.editPassword);

        String regexPassword = "^(?=.*[0-9])(?=.*[a-z])(?=.*[A-Z])(?=.*[!@#&()-[\{}]:;'.?/*^=;<>]).{8,20}$";
        //String regexPassword =
//adding validation to edittexts
        awesomeValidation.addValidation(this, R.id.editTextName, "^(?=

```

```

"(?=.*[a-zA-Z])"+ ".{3,10}" +"$" , R.string.nameerror); //alternative
"^[a-zA-Z0-9]([._-](?![_-])|[a-zA-Z0-9]){3,18}[a-zA-Z0-9]$"
awesomeValidation.addValidation(this, R.id.editTextEmail,
Patterns.EMAIL_ADDRESS, R.string.emailerror);
awesomeValidation.addValidation(this, R.id.editTextMobile,
"[0-9]{8,11}" , R.string.mobileerror);
awesomeValidation.addValidation(this, R.id.editPassword,
regexPassword, R.string.passworderror);

// Creating Volley newRequestQueue .
requestQueue = Volley.newRequestQueue(MainActivity.this);

progressDialog = new ProgressDialog(MainActivity.this);

}

private View.OnClickListener mMyListener = new
View.OnClickListener() {
    public void onClick(View v) {
        switch (v.getId()) {
            case R.id.buttonSubmit:
                if (awesomeValidation.validate()) {
                    regUser();
                }
                break;
            case R.id.already_signin:
                Intent i = new Intent(getApplicationContext(),
LoginActivity.class);
                startActivity(i);
                break;
            default:
                break;
        }
    }
};

public void regUser() {
    progressDialog.setMessage("Please Wait, We are Inserting Your
Data on Server");
    progressDialog.show();

NameHolder = dataName.getText().toString().trim();

```

```

        EmailHolder = dataEmail.getText().toString().trim();
        MobileHolder = dataMobile.getText().toString().trim();
        PasswordHolder = dataPassword.getText().toString().trim();

        String myurl =
"http://192.168.0.103/myloginphp/signup.php?name=" + NameHolder +
                "&email=" + EmailHolder + "&mobile=" + MobileHolder +
"&password=" + PasswordHolder;

        RequestQueue queue =
Volley.newRequestQueue(getApplicationContext());

        StringRequest stringRequest = new
StringRequest(Request.Method.POST, myurl,
                new Response.Listener<String>() {
                    @Override
                    public void onResponse(String ServerResponse) {
                        // Hiding the progress dialog after all task
complete.

                        progressDialog.dismiss();
                        // Showing response message coming from server.
                        Toast.makeText(MainActivity.this,
ServerResponse, Toast.LENGTH_LONG).show();
                    }
                },
                new Response.ErrorListener() {
                    @Override
                    public void onErrorResponse(VolleyError
volleyError) {
                        // Hiding the progress dialog after all task
complete.

                        progressDialog.dismiss();
                        // Showing error message if something goes
wrong.
                        Toast.makeText(MainActivity.this,
volleyError.toString(), Toast.LENGTH_LONG).show();
                    }
                });
        queue.add(stringRequest);
    }
}

```

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    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"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:background="@drawable/screen_background"
    android:orientation="vertical"
    tools:context=".MainActivity"
    android:id="@+id/register_activity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="80dp"
        android:orientation="horizontal"
        android:background="@drawable/top_gradient"/>

    <ScrollView
        android:id="@+id/login_form"
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <LinearLayout
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:layout_marginLeft="10dp"
            android:layout_marginRight="10dp"
            android:orientation="vertical">

            <ImageView
                android:layout_width="133dp"
                android:layout_height="100dp"
                android:layout_gravity="center_horizontal"
                android:layout_marginTop="30dp"
                android:src="@drawable/login" />

            <TextView
```

```
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:text="CODING WITH ARAFAT"
        android:textAlignment="center"
        android:textColor="@color/colorWhite"
        android:textSize="25sp" />

    
```

```
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:text="SIGN UP TO LEARN QUICKLY"
        android:textAlignment="center"
        android:textColor="@color/colorWhite" />

    
```

```
    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/form_username">

    
```

```
<com.google.android.material.textfield.TextInputEditText
        android:id="@+id/editTextName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="10dp"
        android:ems="10"
        android:inputType="text" />

    
```

```
</com.google.android.material.textfield.TextInputLayout>

    
```

```
    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/form_email">

    
```

```
<com.google.android.material.textfield.TextInputEditText
        android:id="@+id/editTextEmail"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />
```

```
        android:layout_height="wrap_content"
        android:layout_margin="10dp"
        android:ems="10"
        android:inputType="textEmailAddress" />

    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/form_phone">

    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/editTextMobile"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="10dp"
        android:ems="10"
        android:inputType="phone" />

    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/form_password"
        app:endIconMode="password_toggle">

    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/editPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="10dp"
        android:ems="10"
        android:inputType="textPassword" />

    </com.google.android.material.textfield.TextInputLayout>

    <Button
```

```

        android:id="@+id/buttonSubmit"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="20dp"
        android:background="@drawable/btn_round"
        android:text="Sign Up"
        android:textColor="@color/colorWhite" />

    <TextView
        android:id="@+id/already_signin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="7dp"
        android:text="@string/already_member"
        android:textAlignment="center"
        android:textColor="@color/colorWhite" />

</LinearLayout>
</ScrollView>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="100dp"
    android:orientation="horizontal"
    android:background="@drawable/top_gradient"/>

</LinearLayout>
```

LoginActivity.java:

```

package com.example.loginsignupdesign;

import androidx.appcompat.app.AppCompatActivity;
import android.app.AlertDialog;
```

```
import android.content.Intent;
import android.os.Bundle;
import android.util.Patterns;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;
import com.basgeekball.awesomevalidation.AwesomeValidation;
import com.basgeekball.awesomevalidation.ValidationStyle;

public class LoginActivity extends AppCompatActivity {
    Button loginButton;
    TextView alreadySignup;
    private AwesomeValidation awesomeValidation;
    ProgressDialog progressDialog;
    String EmailHolder, PasswordHolder ;
    TextView dataEmail; // a text field where the data to be sent is
entered
    TextView dataPassword; // a text field where the data to be sent is
entered

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

        awesomeValidation = new
AwesomeValidation(ValidationStyle.BASIC);

        loginButton = (Button) findViewById(R.id.buttonLogin);
        alreadySignup = (TextView) findViewById(R.id.already_signup);
        loginButton.setOnClickListener(mMyListener);
        alreadySignup.setOnClickListener(mMyListener);
    }
}
```

```

        dataEmail = (TextView) findViewById(R.id.editTextEmail);
        dataPassword = (TextView) findViewById(R.id.editPassword);

        awesomeValidation.addValidation(this, R.id.editTextEmail,
Patterns.EMAIL_ADDRESS, R.string.emailerror);
        progressDialog = new ProgressDialog(LoginActivity.this);
    }

    private View.OnClickListener mMyListener = new
View.OnClickListener() {
    public void onClick(View v) {
        switch (v.getId() /*to get clicked view id**/) {
            case R.id.buttonLogin:
                if (awesomeValidation.validate()) {
//                    Toast.makeText(LoginActivity.this,
"Checking", Toast.LENGTH_LONG).show();
                    regUser();
                }
                break;
            case R.id.already_signup:
                Intent i = new Intent(getApplicationContext(),
MainActivity.class);
                startActivity(i);
                break;
            default:
                break;
        }
    }
};

    public void regUser(){
        progressDialog.setMessage("Please Wait, We are Inserting Your
Data on Server");
        progressDialog.show();

        EmailHolder = dataEmail.getText().toString().trim();
        PasswordHolder = dataPassword.getText().toString().trim();

        String myurl =
"http://192.168.0.103/myloginphp/login.php?email=" +
"&email=" + EmailHolder + "&password=" +
PasswordHolder;
    }
}

```

```

        RequestQueue queue =
Volley.newRequestQueue(getApplicationContext());
queue.add(stringRequest);
}

StringRequest stringRequest = new StringRequest(Request.Method.POST, myurl,
        new Response.Listener<String>() {
    @Override
    public void onResponse(String ServerResponse) {
        // Hiding the progress dialog after all task
        complete.
        progressDialog.dismiss();
        if (ServerResponse.equals("1")){
            Toast.makeText(LoginActivity.this,
ServerResponse, Toast.LENGTH_LONG).show();
            Intent i = new
Intent(getApplicationContext(), WelcomeActivity.class);
            startActivity(i);
        } else {
            Toast.makeText(LoginActivity.this,
ServerResponse, Toast.LENGTH_LONG).show();
        }
    }
},
        new Response.ErrorListener() {
    @Override
    public void onErrorResponse(VolleyError
volleyError) {
        // Hiding the progress dialog after all task
        complete.
        progressDialog.dismiss();
        // Showing error message if something goes
        wrong.
        Toast.makeText(LoginActivity.this,
volleyError.toString(), Toast.LENGTH_LONG).show();
    }
});
queue.add(stringRequest);
}
}

```

activity_login.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    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"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:orientation="vertical"
    android:background="@drawable/screen_background"
    tools:context=".LoginActivity"
    android:id="@+id/register_activity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="80dp"
        android:orientation="horizontal"
        android:background="@drawable/top_gradient"/>

    <ScrollView
        android:id="@+id/login_form"
        android:layout_width="match_parent"
        android:layout_height="match_parent" >

        <LinearLayout
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:layout_marginLeft="10dp"
            android:layout_marginRight="10dp"
            android:orientation="vertical">

            <ImageView
                android:layout_width="142dp"
                android:layout_height="110dp"
                android:layout_gravity="center_horizontal"
                android:layout_marginTop="50dp"
                android:src="@drawable/login" />
```

```
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:textColor="@color/colorWhite"
    android:textSize="25sp"
    android:textAlignment="center"
    android:text="WELCOME BACK"/>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="5dp"
    android:textColor="@color/colorWhite"
    android:textAlignment="center"
    android:text="SIGN IN TO LEARN QUICKLY"/>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="@string/form_email">

<com.google.android.material.textfield.TextInputEditText
    android:id="@+id/editTextEmail"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:inputType="textEmailAddress"
    android:layout_margin="10dp"
    android:ems="10" />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    app:endIconMode="password_toggle"
    android:hint="@string/form_password">

<com.google.android.material.textfield.TextInputEditText
    android:id="@+id/editPassword"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
```

```
        android:inputType="textPassword"
        android:layout_margin="10dp"
        android:ems="10" />

    </com.google.android.material.textfield.TextInputLayout>

    <Button
        android:id="@+id/buttonLogin"
        android:layout_margin="20dp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textColor="@color/colorWhite"
        android:background="@drawable/btn_round"
        android:text="Login" />

    <TextView
        android:id="@+id/already_signup"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="7dp"
        android:textColor="@color/colorWhite"
        android:textAlignment="center"
        android:text="@string/new_member"/>

    </LinearLayout>
</ScrollView>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="100dp"
    android:orientation="horizontal"
    android:background="@drawable/top_gradient"/>

</LinearLayout>
```

WelcomeActivity.java:

```
package com.example.loginsignupdesign;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class WelcomeActivity extends AppCompatActivity {

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

activity_welcome.xml:

```
<?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=".WelcomeActivity">

    <TextView
```

```
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="by ARAFAT (199076003) "
        android:textColor="#FF0000"
        android:textSize="30sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView2"
        app:layout_constraintVertical_bias="0.297" />

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="WELCOME"
        android:textAlignment="viewStart"
        android:textColor="#2443ED"
        android:textSize="40sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.457"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.244" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

AndroidManifest.xml:

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

    <uses-permission android:name="android.permission.INTERNET" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="CODE WITH ARAFAT Singup Form"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:usesCleartextTraffic="true"
        android:theme="@style/Theme.LoginSignupDesign">
        <activity android:name=".WelcomeActivity"></activity>
        <activity
            android:name=".LoginActivity"
            android:theme="@style/Theme.AppCompat.NoActionBar">
            </activity>
        <activity
            android:name=".MainActivity"
            android:theme="@style/Theme.AppCompat.NoActionBar">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category
                    android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```