

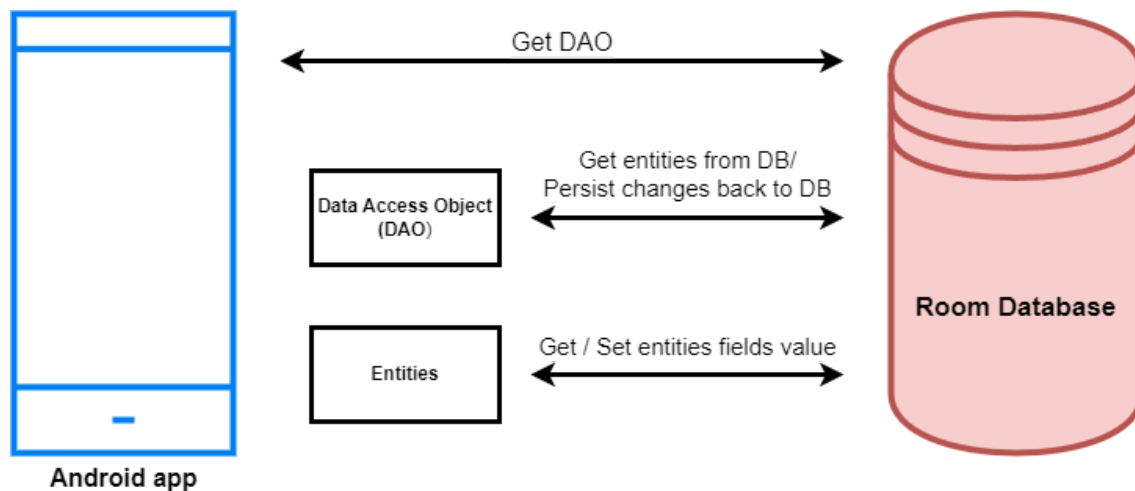


**Podcast Plus: A Redux-Inspired Podcast App with  
Dynamic Themes for Android  
Project Based Experiential Learning Program**

## Podcast Plus: A Redux-Inspired Podcast App with Dynamic Themes for Android

A project that demonstrates the use of Android Jetpack Compose to build a UI for a podcast player app. The app allows users to choose , play and pause podcasts.

### Architecture



### Learning Outcomes :

By end of this project:

- You'll be able to work on Android studio and build an app.
- You'll be able to integrate the database accordingly.

### Project Workflow:

- Users register into the application.
- After registration , user logins into the application.
- User enters into the main page
- The app allows users to choose , play and pause podcasts.

## Tasks:

- 1.Required initial steps
- 2.Creating a new project.
- 3.Adding required dependencies.
- 4.Creating the database classes.
- 5.Building application UI and connecting to database.
- 6.Modifying AndroidManifest.xml
- 7.Running the application.

### Task 1:

Required initial steps :

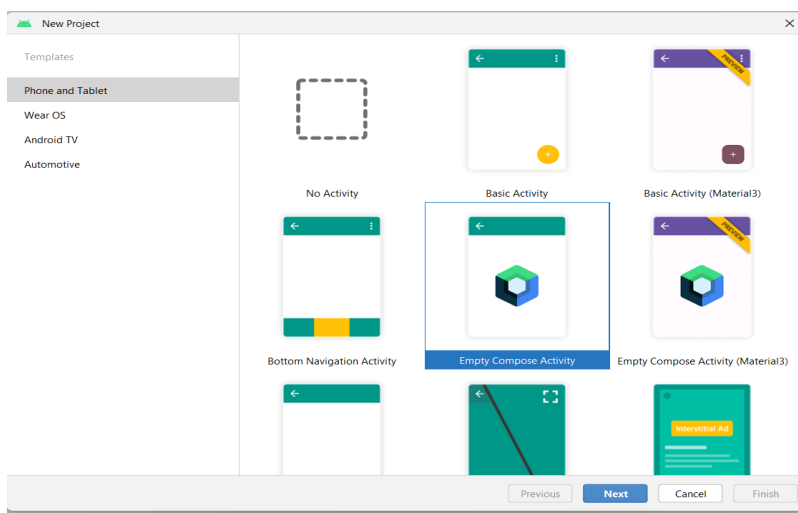
<https://developer.android.com/studio/install>

### Task 2 :

Creating a new project.

Step 1 : Android studio > File > New > New Project > Empty Compose Activity

Step 2 : Click on **Next** button.

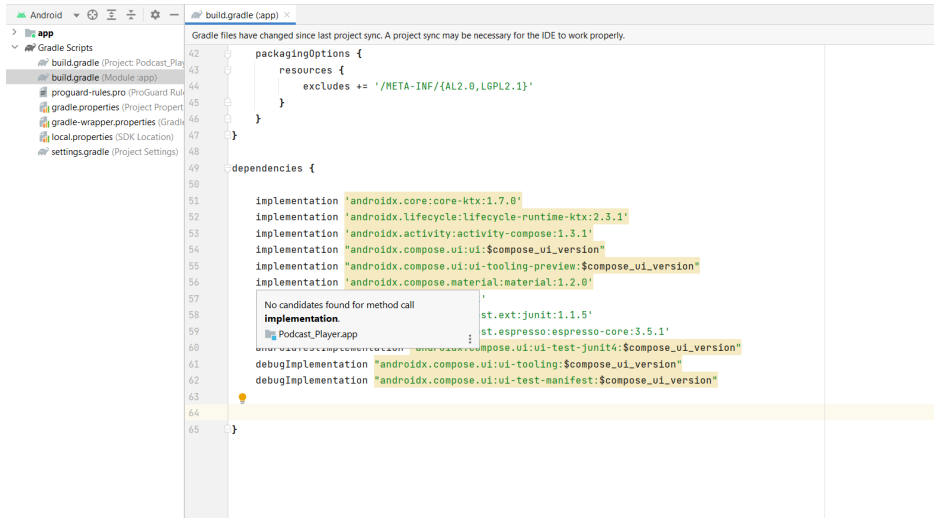


Step 3 : Give name to the new project.

Step 4 : Give the Minimum SDK value

Step 5 : Click Finish



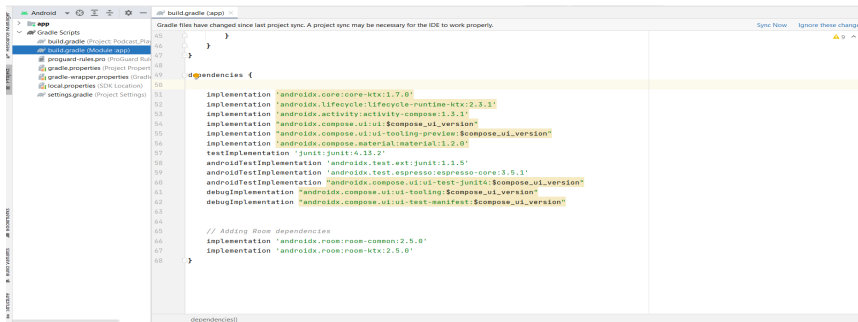


Step 2 : Adding room dependencies.  
Add the below code in dependencies

```

// Adding Room dependencies
implementation 'androidx.room:room-common:2.5.0'
implementation 'androidx.room:room-ktx:2.5.0'

```



Step 3 : Click on Sync now

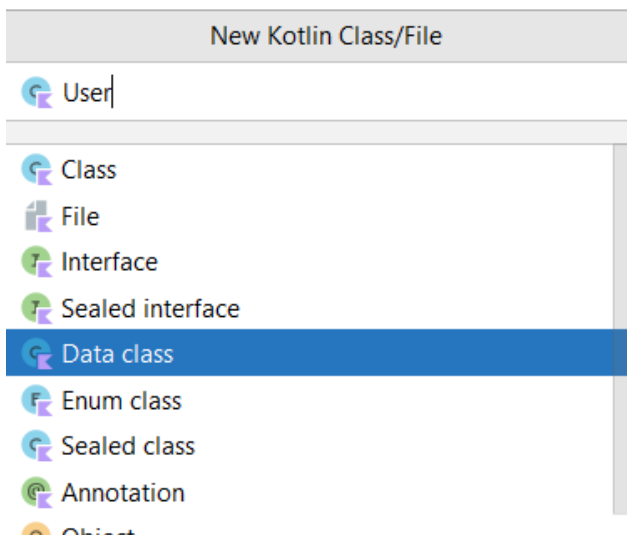
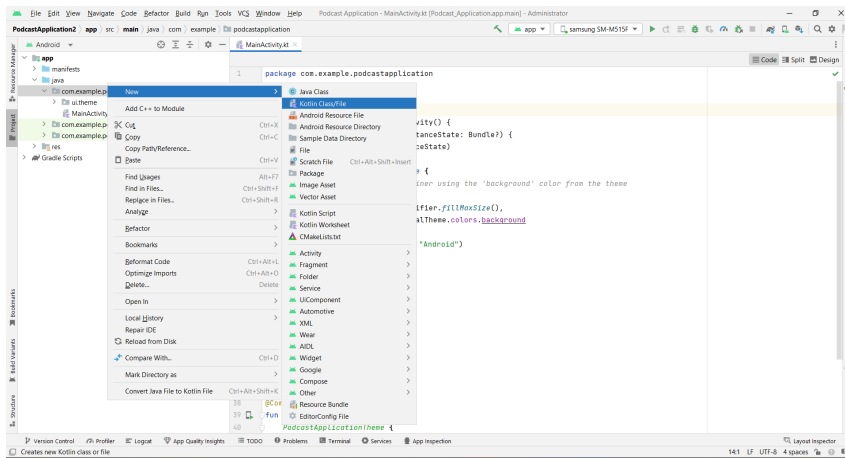
## Task 4:

Creating the database classes.

- To learn more about Database follow this link:

<https://gorillalogic.com/blog/android-room-tutorial-simplifying-how-you-work-with-app-database/>

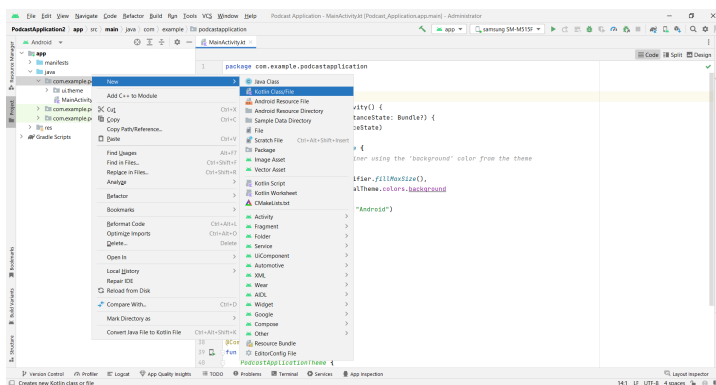
Step 1 : Create User data class

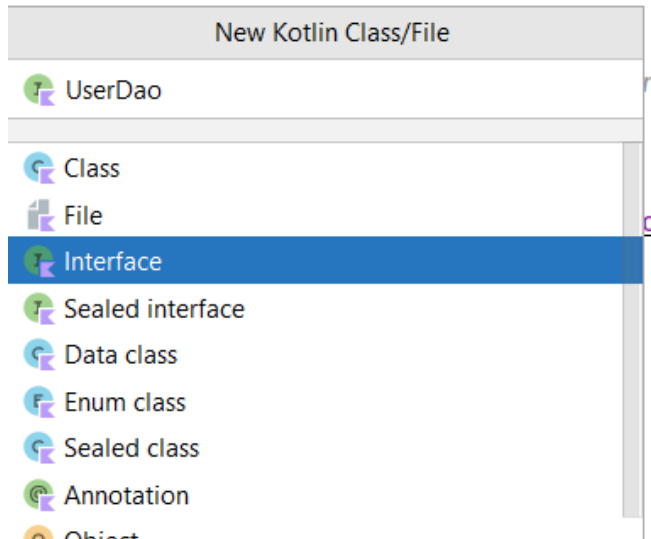


User class code:

<https://github.com/smartinternz02/PodcastPlayer/blob/master/app/src/main/java/com/example/podcastplayer/User.kt>

Step 2 : Create an UserDao interface

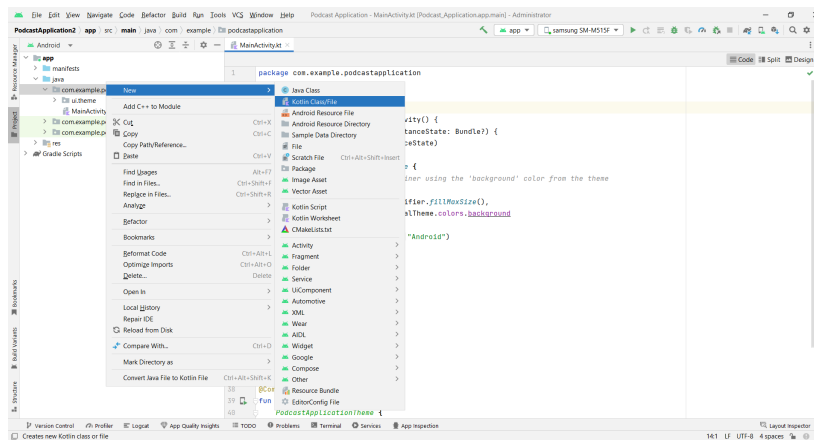


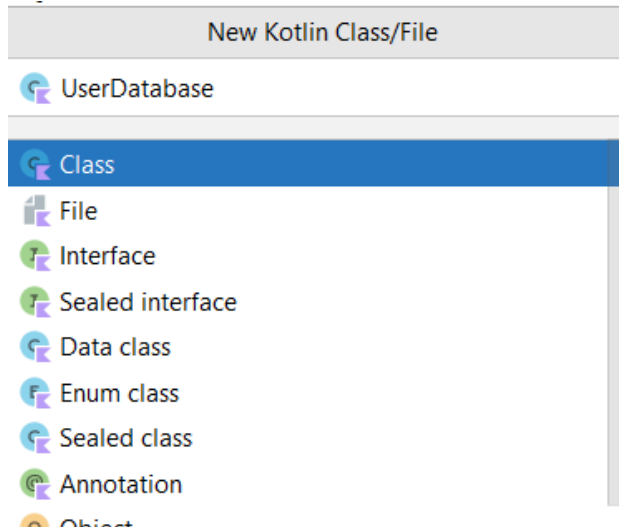


UserDao interface code:

<https://github.com/smartinternz02/PodcastPlayer/blob/master/app/src/main/java/com/example/podcastplayer/UserDao.kt>

Step 3 : Create an UserDatabase class

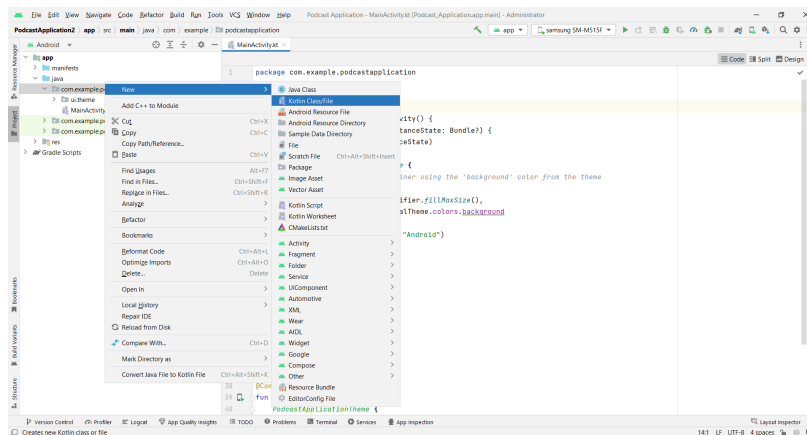




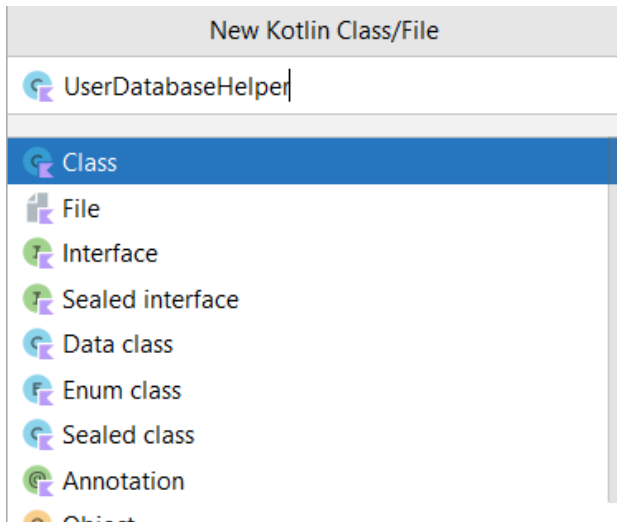
UserDatabase class code :

<https://github.com/smartinternz02/PodcastPlayer/blob/master/app/src/main/java/com/example/podcastplayer/UserDatabase.kt>

Step 4 : Create an UserDatabaseHelper class







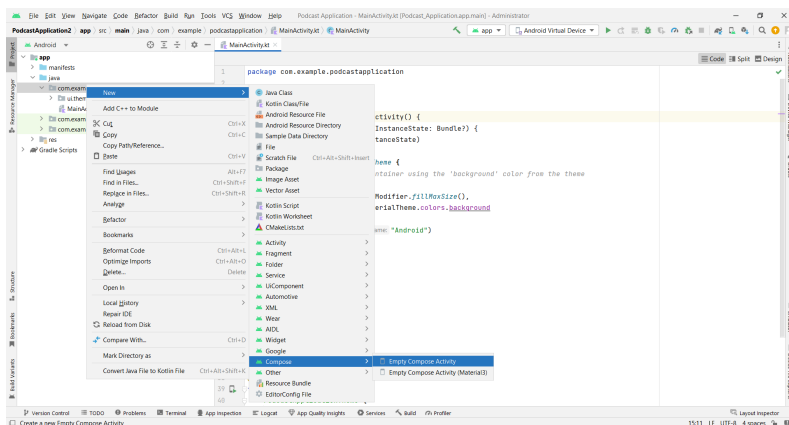
UserDatabaseHelper class code :

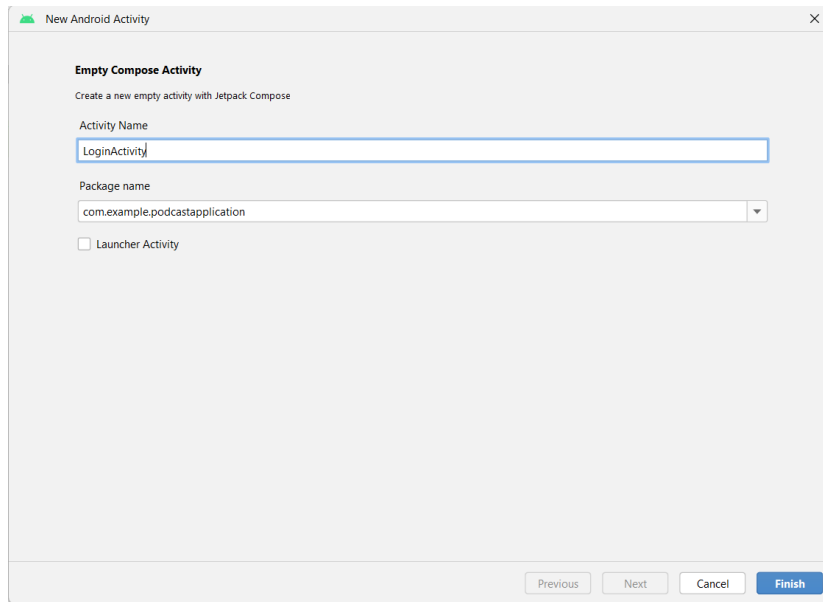
<https://github.com/smartinternz02/PodcastPlayer/blob/master/app/src/main/java/com/example/podcastplayer/UserDatabaseHelper.kt>

## Task 5:

Building application UI and connecting to database.

Step 1: Creating LoginActivity.kt with database





## Database connection in LoginActivity.kt

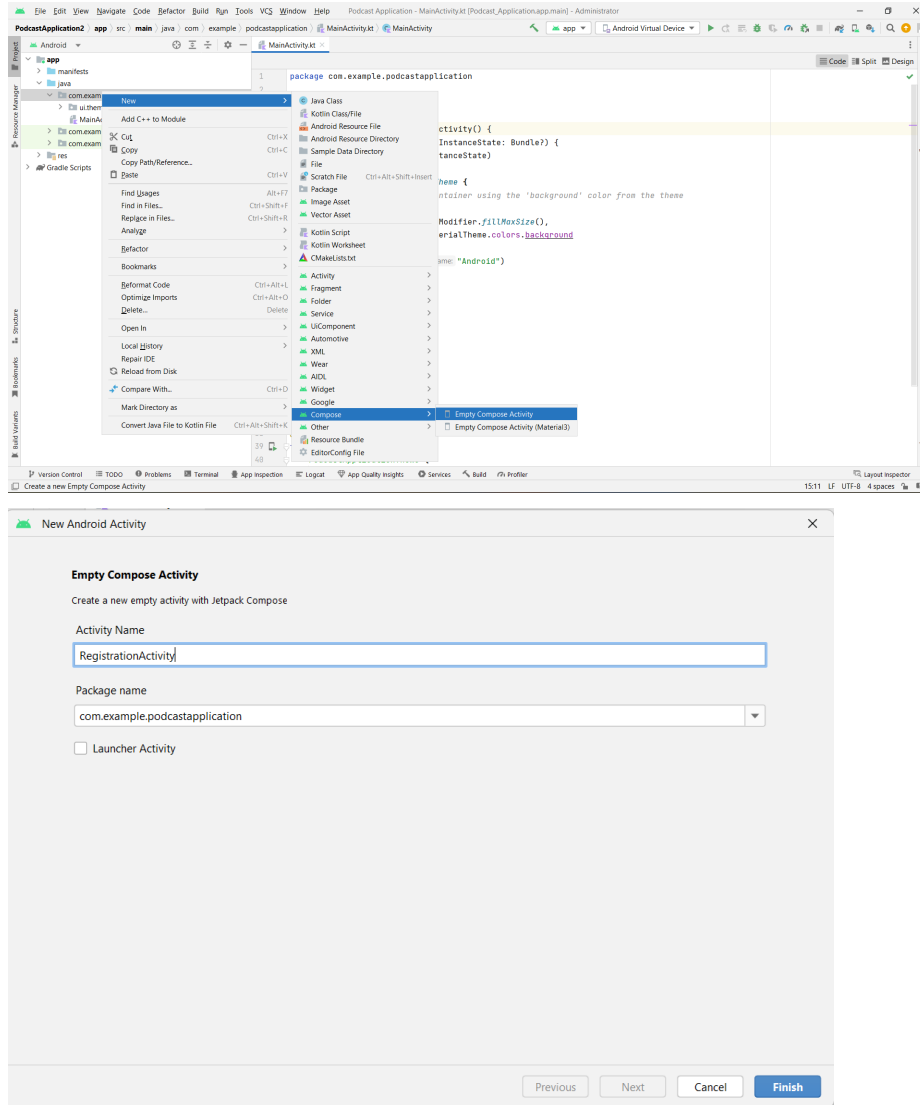
```
class LoginActivity : ComponentActivity() {
    private lateinit var databaseHelper: UserDatabaseHelper
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        databaseHelper = UserDatabaseHelper(context = this)
        setContent {
            PodcastPlayerTheme {
                // A surface container using the 'background' color from the theme
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colors.background
                ) {
                    LoginScreen(context = this, databaseHelper)
                }
            }
        }
    }
}

@Composable
fun LoginScreen(context: Context, databaseHelper: UserDatabaseHelper) {
    var username by remember { mutableStateOf(value: "") }
    var password by remember { mutableStateOf(value: "") }
    var error by remember { mutableStateOf(value: "") }
```

Complete code in below link

<https://github.com/smartinternz02/PodcastPlayer/blob/master/app/src/main/java/com/example/podcastplayer/LoginActivity.kt>

## Step 2 : Creating RegistrationActivity.kt with database



## Database connection in RegistrationActivity.kt

```

class RegistrationActivity : AppCompatActivity() { private lateinit var databaseHelper: UserDatabaseHelper
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        databaseHelper = UserDatabaseHelper(context: this)
        setContent {
            PodcastPlayerTheme {
                // A surface container using the 'background' color from the theme
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colors.background
                ) {
                    RegistrationScreen(context: this, databaseHelper)
                }
            }
        }
    }
}

```

```

@Composable
fun RegistrationScreen(context: Context, databaseHelper: UserDatabaseHelper) {
    var username by remember { mutableStateOf(value: "") }
    var password by remember { mutableStateOf(value: "") }
    var email by remember { mutableStateOf(value: "") }
    var error by remember { mutableStateOf(value: "") }
}

```

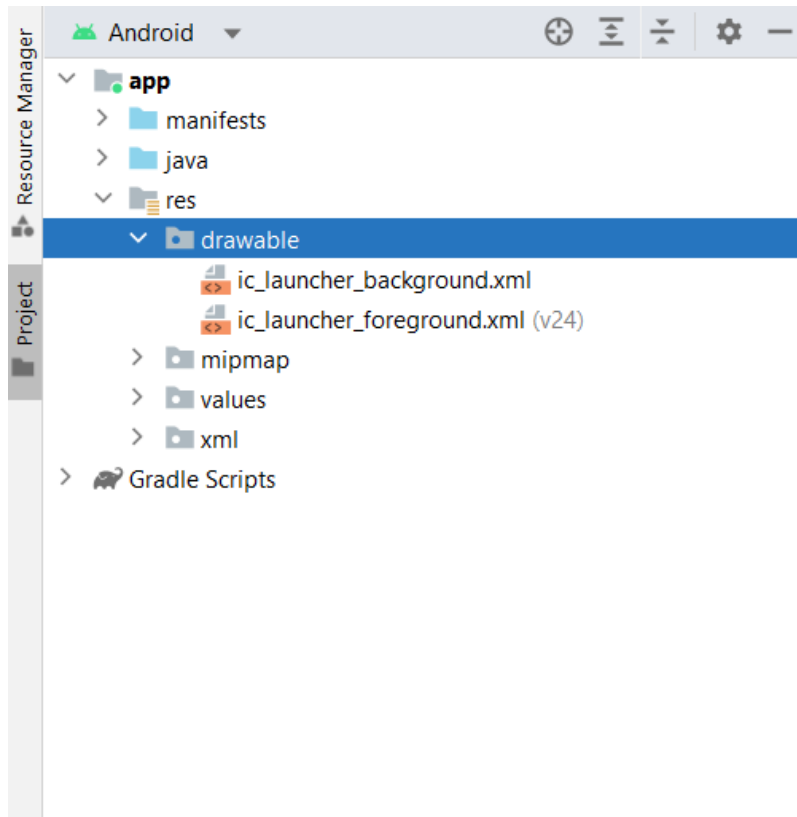
Complete code in below link:

<https://github.com/smartinternz02/PodcastPlayer/blob/master/app/src/main/java/com/example/podcastplayer/RegistrationActivity.kt>

Step 3 : Creating MainActivity.kt file

In MainActivity.kt file the main application is developed

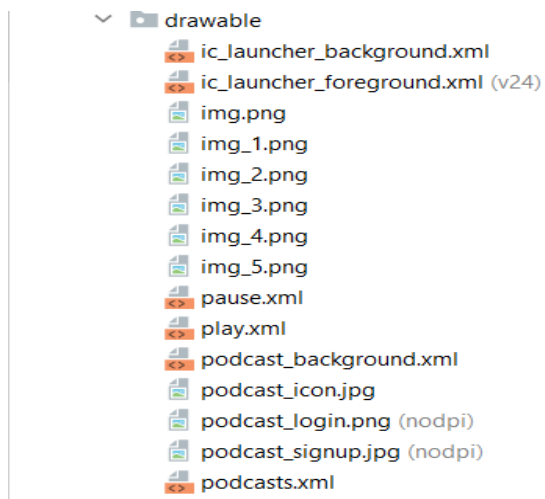
- Before creating UI we need to add some images in drawables which are in res



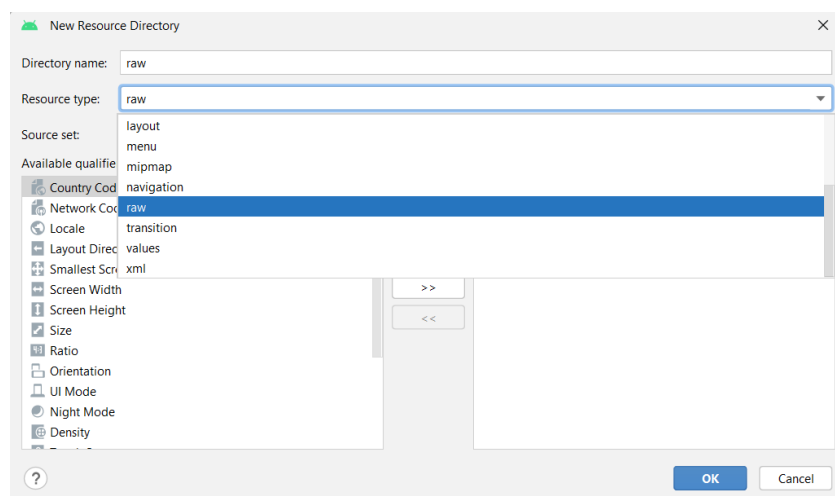
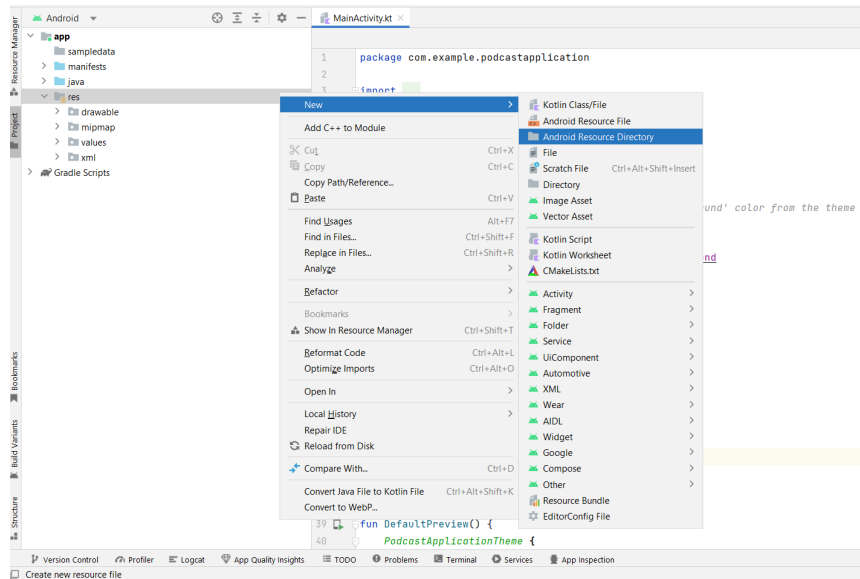
Download the required drawable from the code:

<https://github.com/smartinternz02/PodcastPlayer/tree/master/app/src/main/res/drawable>

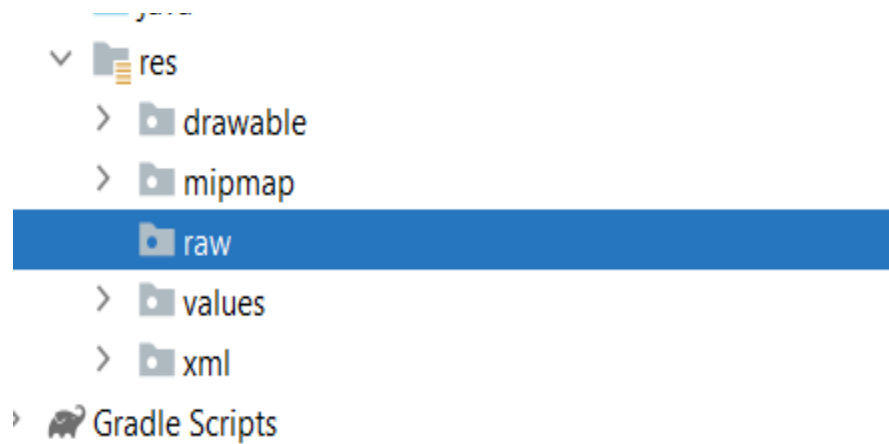
## Required drawables



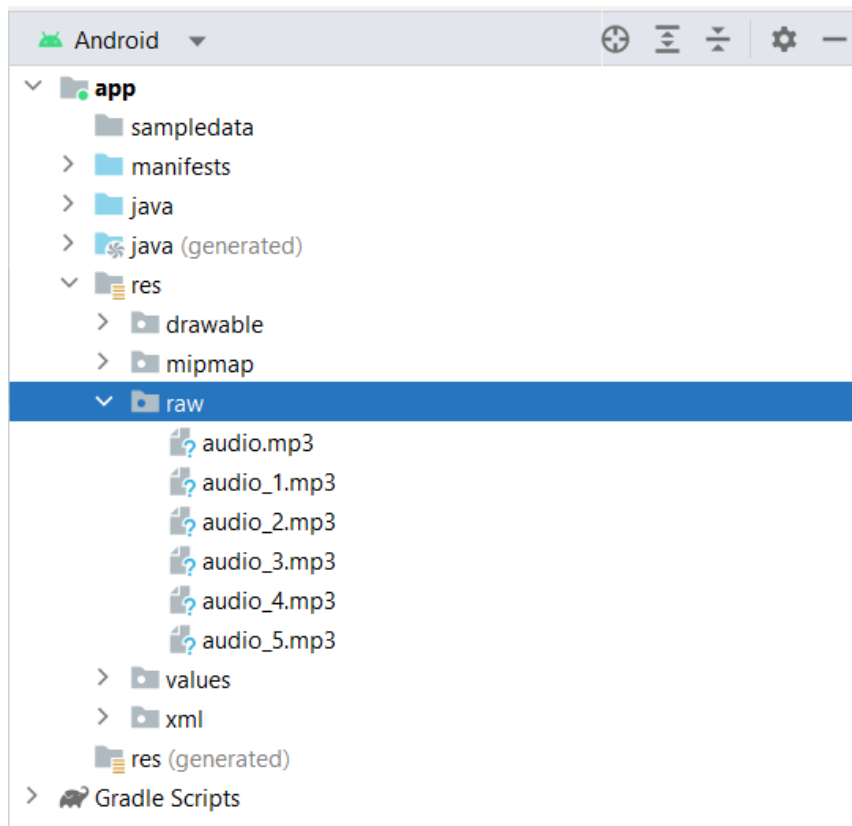
- After adding drawables we need to add the required audio files



Next click OK



Add the downloaded audio into raw by copy and paste of the file



You can download the files from below link

<https://github.com/smartinternz02/PodcastPlayer/tree/master/app/src/main/res/raw>

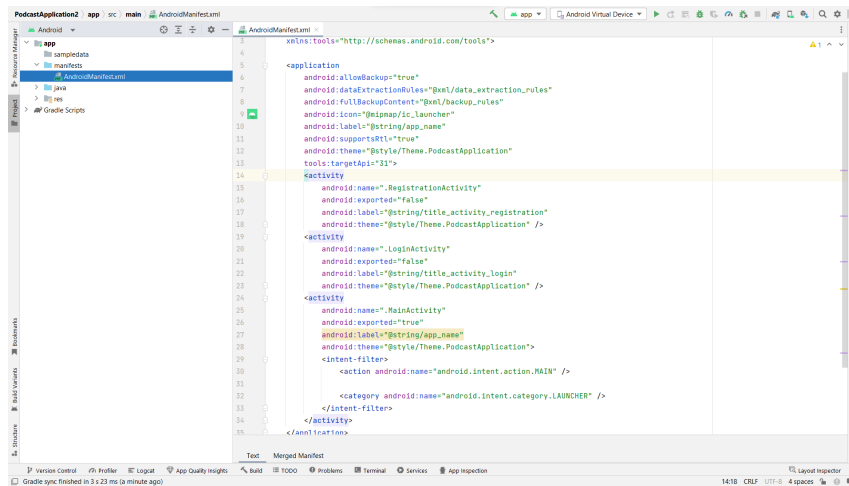
- After add all this we need to create the UI in MainActivity.kt file

MainActivity.kt file

Complete MainActivity.kt code:

<https://github.com/smartinternz02/PodcastPlayer/blob/master/app/src/main/java/com/example/podcastplayer/MainActivity.kt>

## Task 6: Modifying AndroidManifest.xml



When we run the app we will get the MainActivity.kt file as our first screen , but we want LoginActivity.kt , So we need to change in AndroidManifest.xml.



```

<activity
    android:name=".RegistrationActivity"
    android:exported="false"
    android:label="@string/title_activity_registration"
    android:theme="@style/Theme.PodcastApplication" />
<activity
    android:name=".LoginActivity"
    android:exported="false"
    android:label="@string/title_activity_login"
    android:theme="@style/Theme.PodcastApplication" />
<activity
    android:name=".MainActivity"
    android:exported="true"
    android:label="@string/app_name"
    android:theme="@style/Theme.PodcastApplication">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />

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

```

Changed AndroidManifest.xml.

```

tools:targetApi="31">
<activity
    android:name=".RegistrationActivity"
    android:exported="false"
    android:label="RegistrationActivity"
    android:theme="@style/Theme.PodcastPlayer" />
<activity
    android:name=".MainActivity"
    android:exported="false"
    android:label="MainActivity"
    android:theme="@style/Theme.PodcastPlayer" />
<activity
    android:name=".LoginActivity"
    android:exported="true"
    android:label="Podcast Player"
    android:theme="@style/Theme.PodcastPlayer">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />

        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
</application>

```

Complete code is given below:

<https://github.com/smartinternz02/PodcastPlayer/blob/master/app/src/main/AndroidManifest.xml>

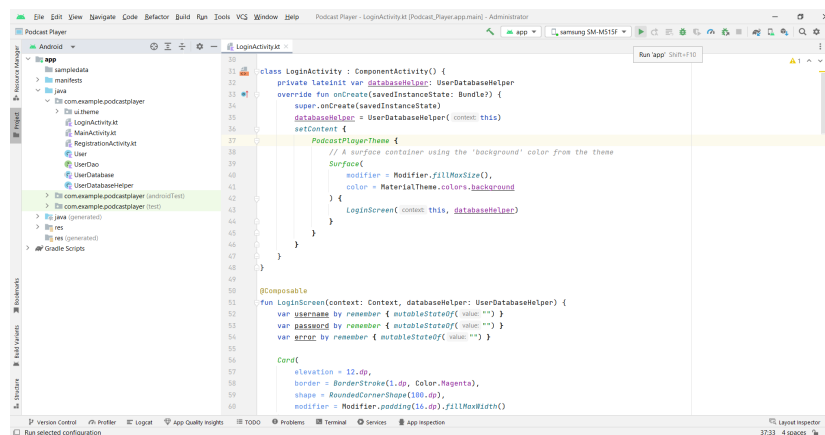
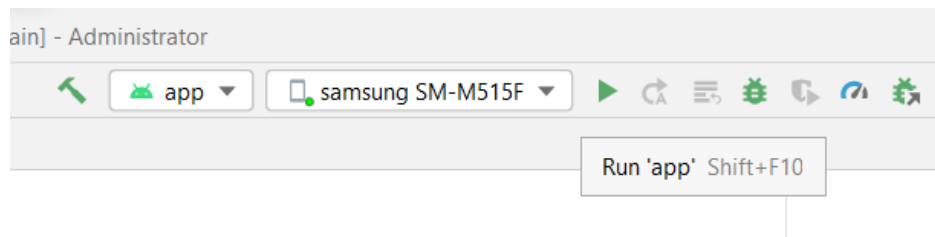
## Task 7:

Running the application.

Step 1: Run apps on a hardware device

<https://developer.android.com/studio/run/device>

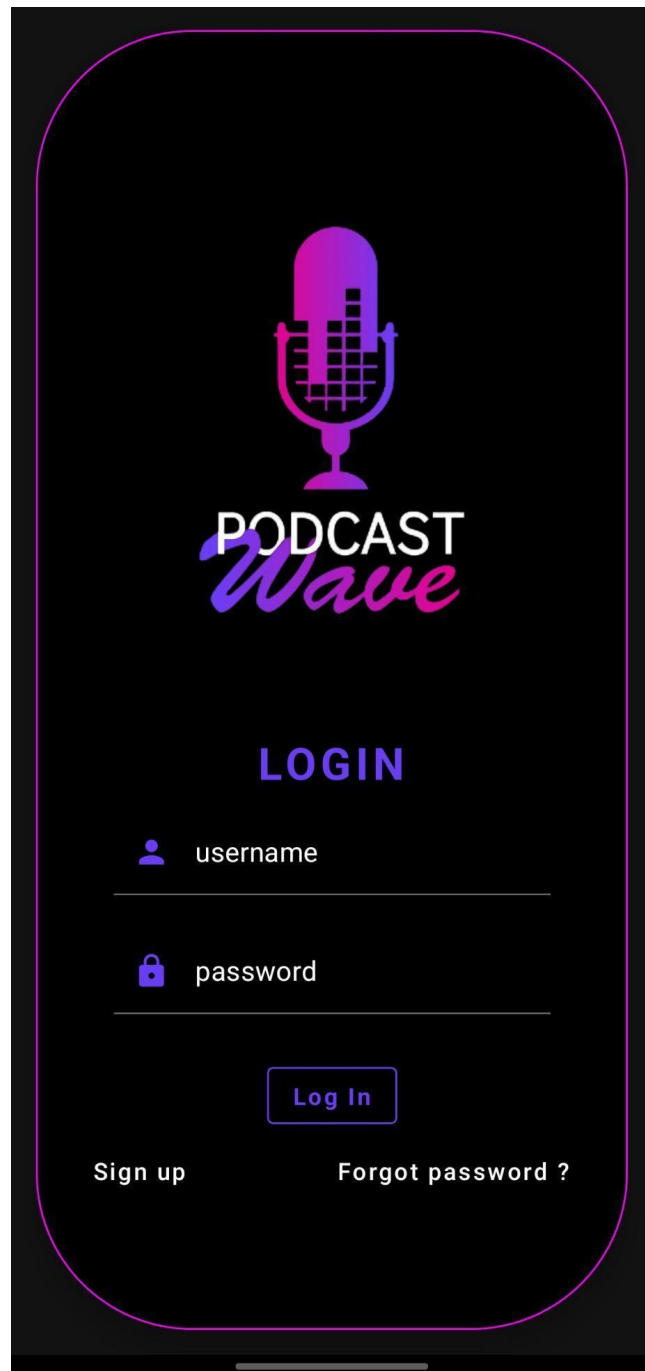
Step 2: Run the application in Mobile



**Complete Project Link:** <https://github.com/smartinternz02/PodcastPlayer>


## **Final Output of the Application :**


Login Page




Register Page :

## Sign Up




 username

---

 password

---

 email

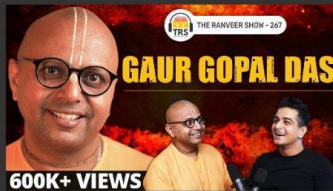
---

[Register](#)

Have an account? [Log in](#)

Main Podcast Page :

# PODCAST



GaurGopalDas Returns To TRS - Life, Monkhhood & Spirituality



Haunted Houses, Evil Spirits & The Paranormal Explained | Sarbajeet Mohanty



Kaali Mata ki kahani - Black Magic & Aghoris ft. Dr Vineet Aggarwal

