A Sleep Tracking App for a Better Night's Rest

Project Based Experiential Learning Program

Handbook

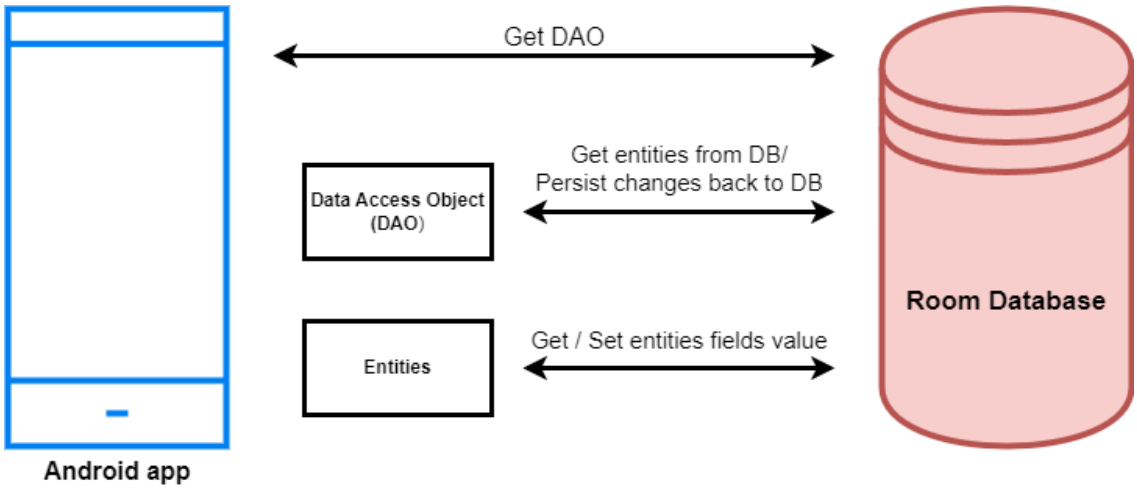
SmartBridge Educational Services Pvt. Ltd.

A Sleep Tracking App for a Better Night's Rest

A project that demonstrates the use of Android Jetpack Compose to build a UI for a sleep tracking app. The app allows users to track their sleep.With the “Sleep Tracker” app, you can assess the quality of sleep they have had in a day. It has been time and again proven that a good quality sleep is pretty essential for effective functioning of both mind and body.

“Sleep Tracker” application enables you to start the timer when they are in the bed and about to fall asleep. The timer will keep running in the background until it is stopped, whenever the user wakes up. Based on the sleep experience, you can rate your sleep quality. Finally , the app will display an analysis of the kind of sleep , you had the previous night.

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
- User can track the sleep timing and he record the time

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.Using AndroidManifest.xml
7.Running the application.

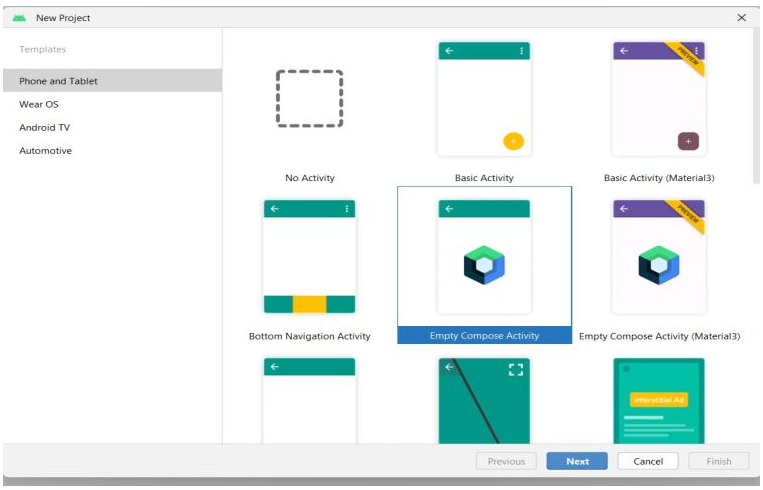
Task 1:

Required initial steps :

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

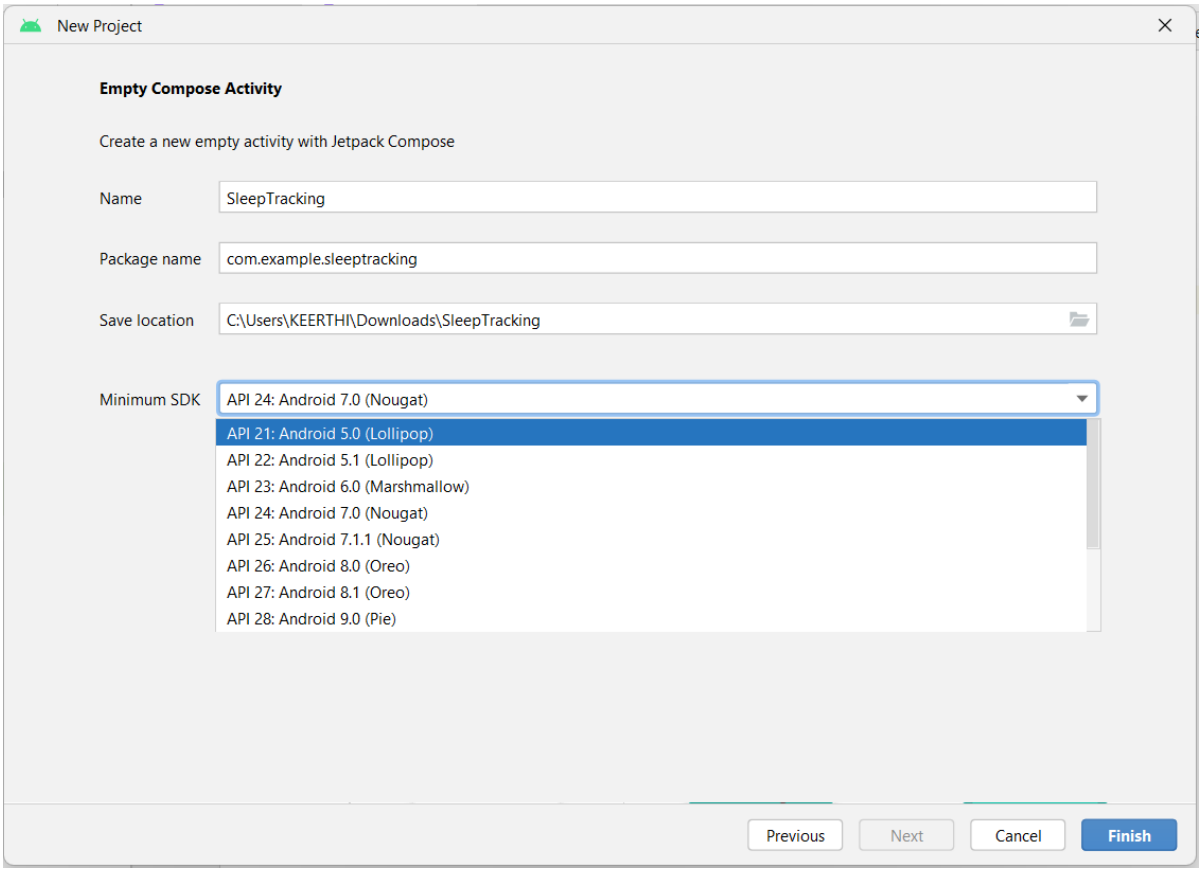
Task 2 :

Creating a new project.
Step 1 : Android studio > File > New > New Project > Empty Compose Activity
Step 2 : Click on **Next** button.

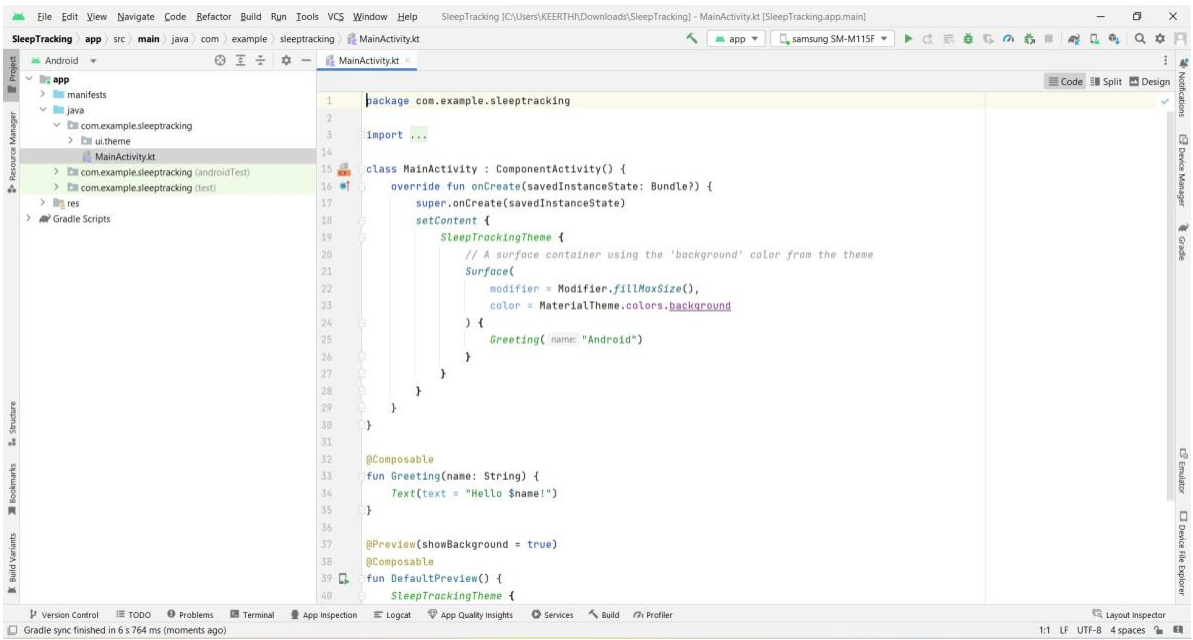


Step 3 : Give name to the new project.

Step 4 : Give the Minimum SDK value
Step 5 : Click Finish



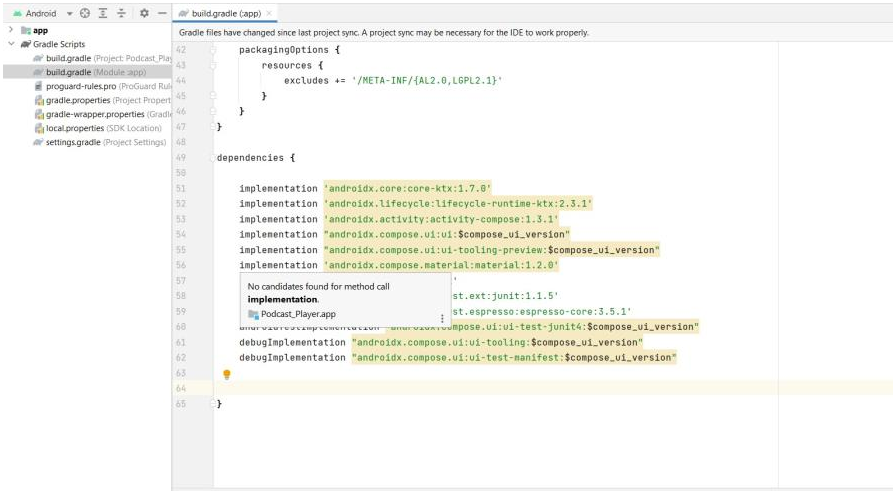
Main activity file



Task 3 :

Adding required dependencies.

Step 1 : Gradle scripts > build.gradle(Module :app)



Step 2 : Adding room dependencies.

Add the below code in dependencies

```
dependencies {
    implementation("androidx.core:core-ktx:1.9.0")
    implementation("androidx.lifecycle:lifecycle-runtime-ktx:2.6.1")
    implementation("androidx.activity:activity-compose:1.7.2")
    implementation(platform("androidx.compose:compose-bom:2023.03.00"))
    implementation("androidx.compose.ui:ui")
    implementation("androidx.compose.ui:ui-graphics")
    implementation("androidx.compose.ui:ui-tooling-preview")
    implementation("androidx.compose.material3:material3")
    testImplementation("junit:junit:4.13.2")
    androidTestImplementation("androidx.test.ext:junit:1.1.5")
    androidTestImplementation("androidx.test.espresso:espresso-core:3.5.1")
    androidTestImplementation(platform("androidx.compose:compose-bom:2023.03.00"))
    androidTestImplementation("androidx.compose.ui:ui-test-junit4")
    debugImplementation("androidx.compose.ui:ui-tooling")
    debugImplementation("androidx.compose.ui:ui-test-manifest")
}
```

Step 3 : Click on Sync now

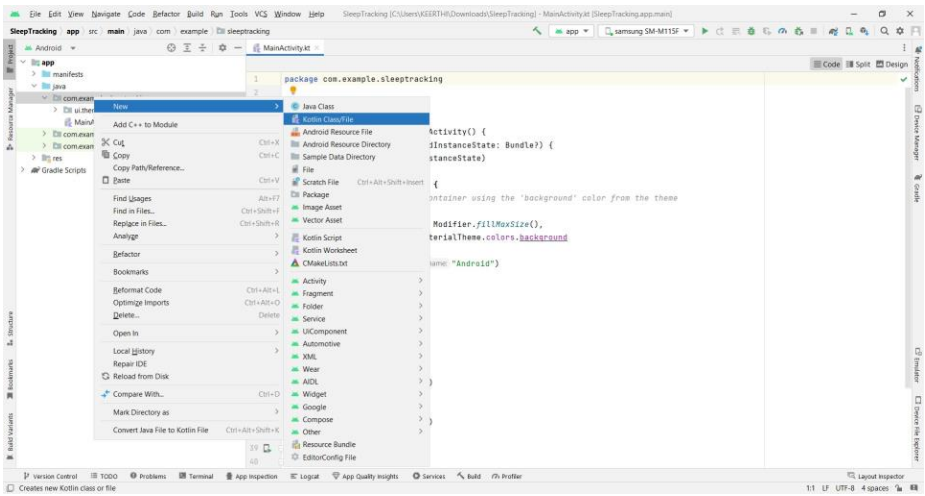
Task 4:

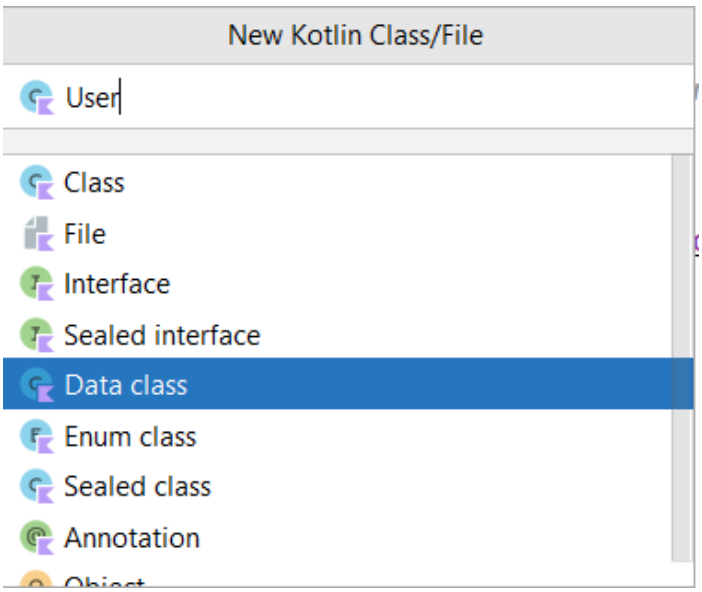
Creating the database classes.

In this project we will be having two databases, one is for user registration and login and other is for tracking the sleep of the user.

Database 1

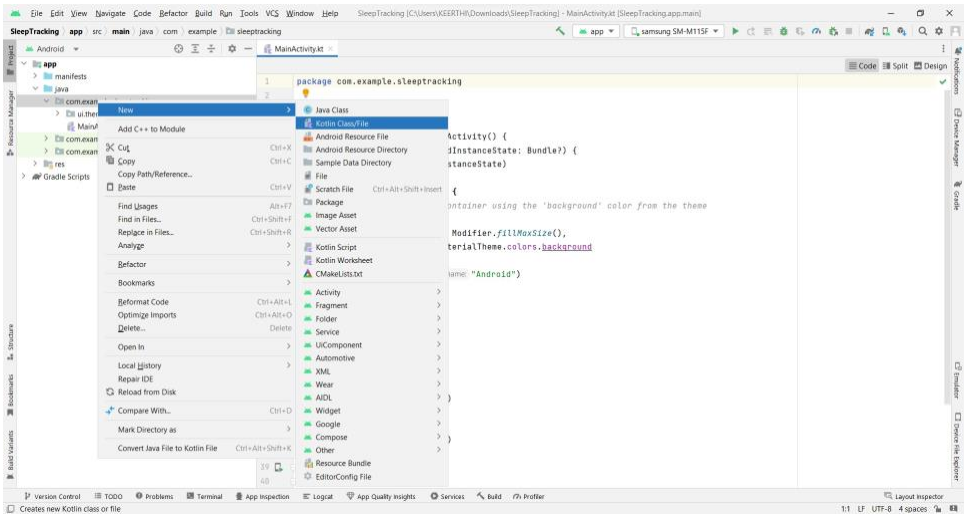
Step 1 : Create User data class

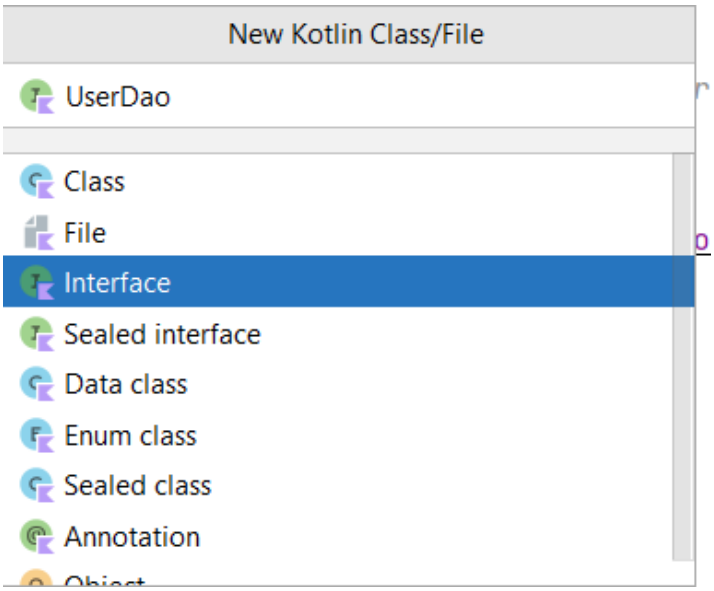




User class code:

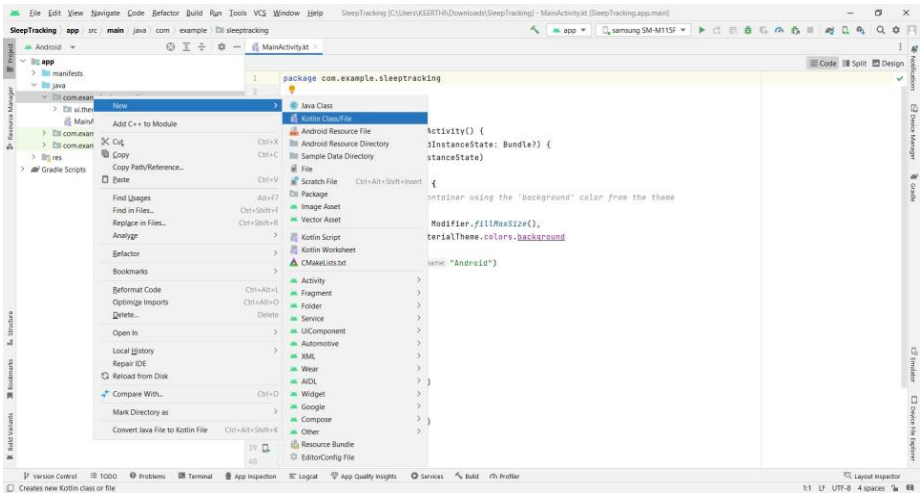
Step 2 : Create an UserDao interface

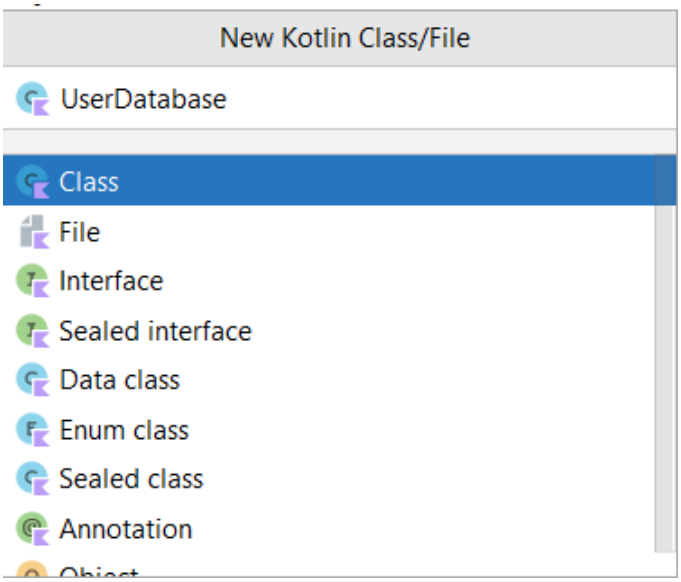




UserDao interface code:

Step 3 : Create an UserDatabase class

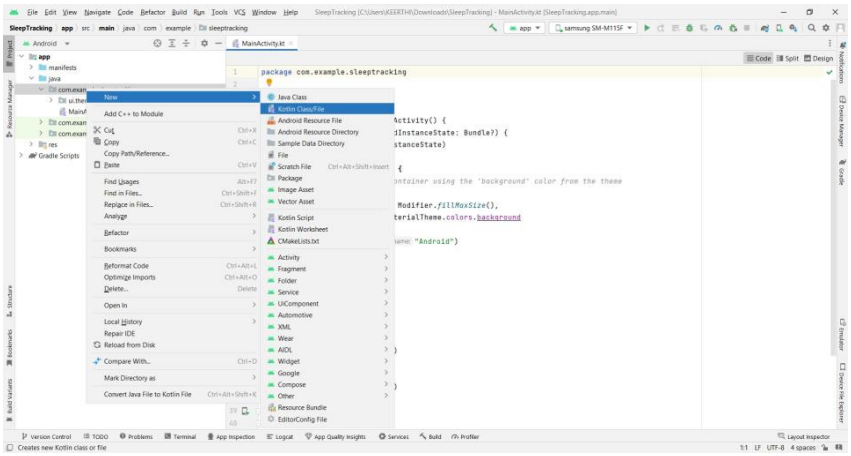


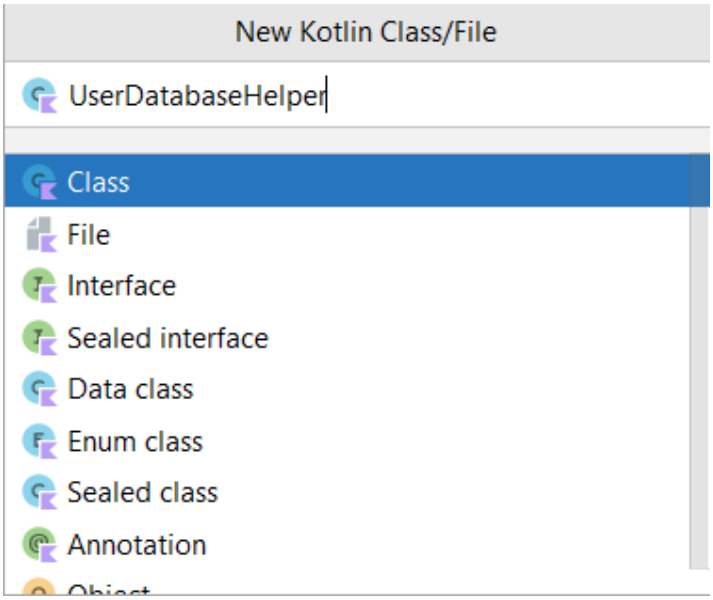


UserDatabase class code :

https://github.com/kondasivaprasad/Sleep-Tracker/blob/master/app/src/main/java/com/example/sleeptracking/UserDatabase.kt

Step 4 : Create an UserDatabaseHelper class

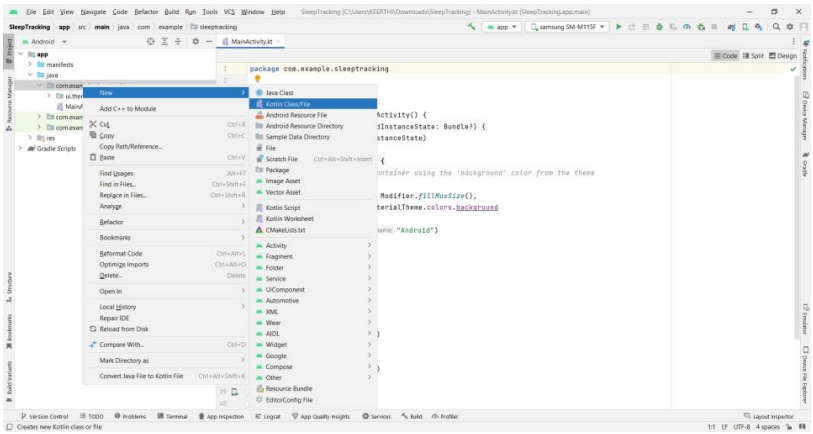


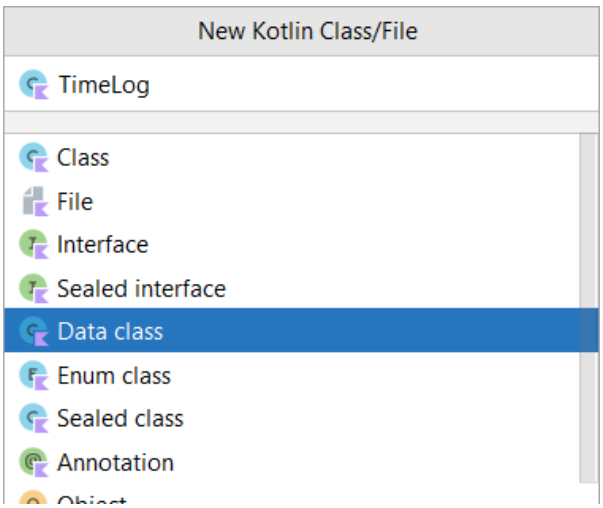


UserDatabaseHelper class code : https://github.com/kondasivaprasad/Sleep-Tracker/blob/master/app/src/main/java/com/example/sleeptracking/UserDatabaseHelper.kt

Database 2

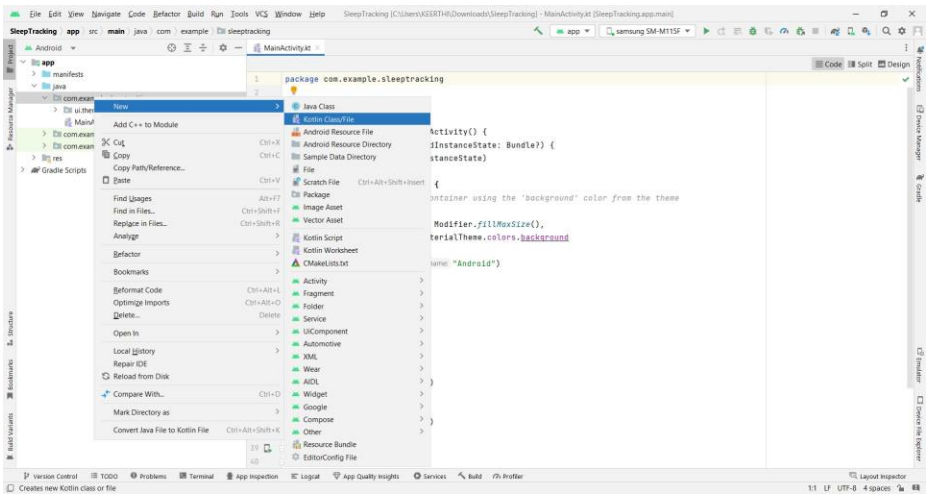
Step 1 : Create TimeLog data class

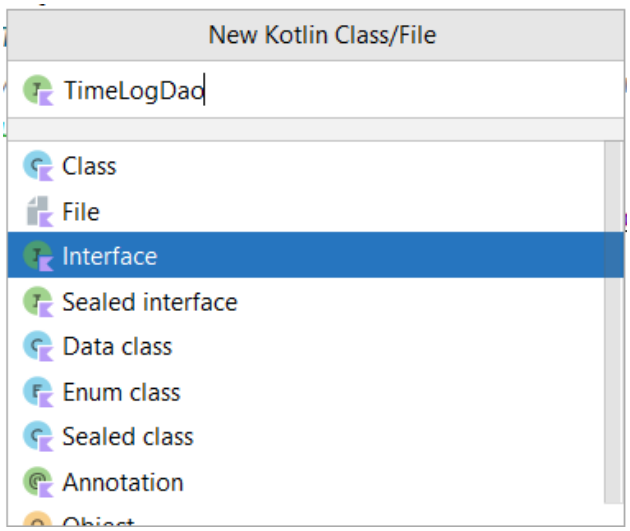




TimeLog data class code: https://github.com/kondasivaprasad/Sleep-Tracker/blob/master/app/src/main/java/com/example/sleeptracking/TimeLog.kt

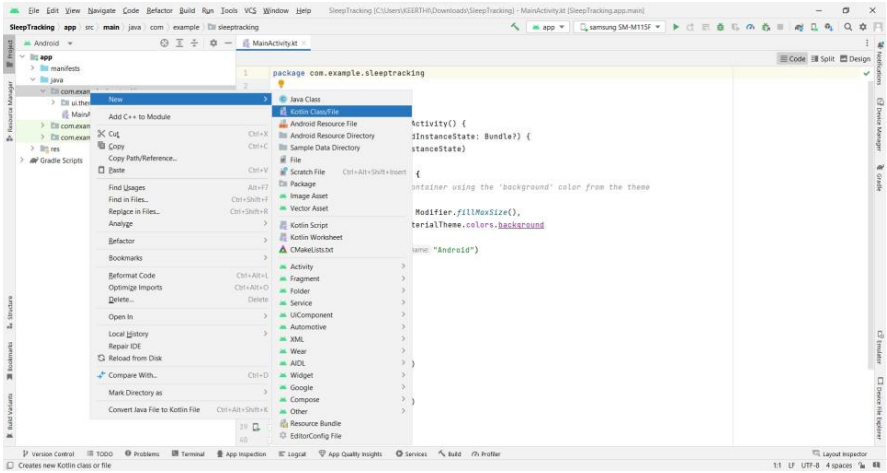
Step 2 : Create an TimeLogDao interface

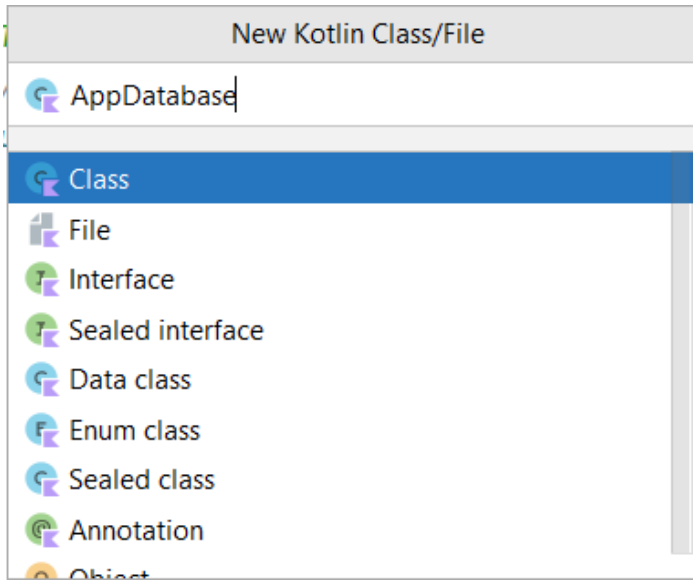




TimeLogDao interface code: https://github.com/kondasivaprasad/Sleep-Tracker/blob/master/app/src/main/java/com/example/sleeptracking/TimeLogDao.kt

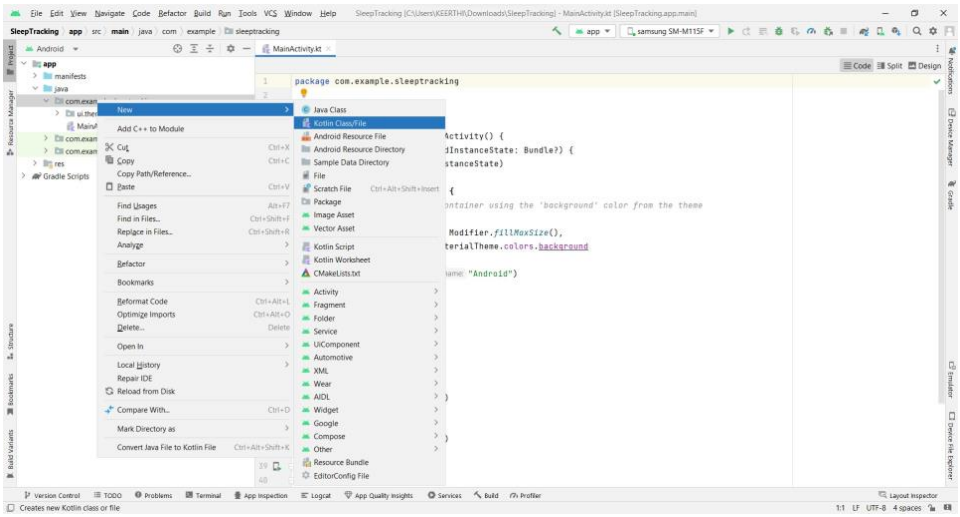
Step 3 : Create an AppDatabase class

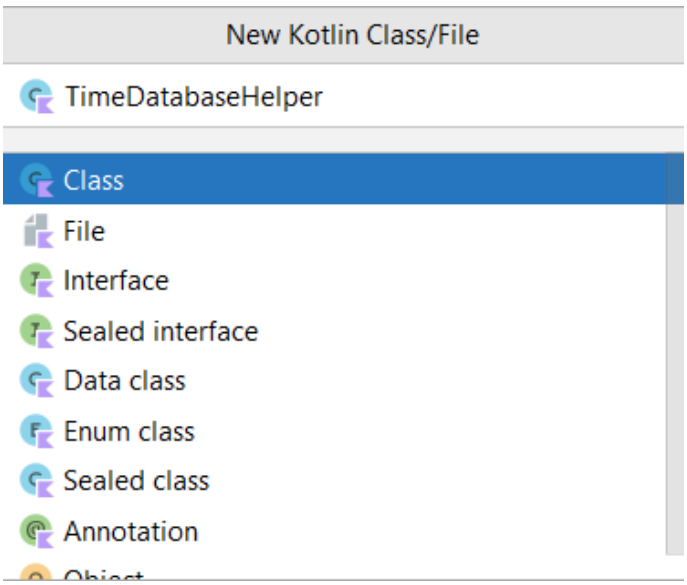




AppDatabase class code:
https://github.com/kondasivaprasad/Sleep-Tracker/blob/master/app/src/main/java/com/example/sleeptracking/UserDatabase.kt

Step 4 : Create an TimeDatabaseHelper class



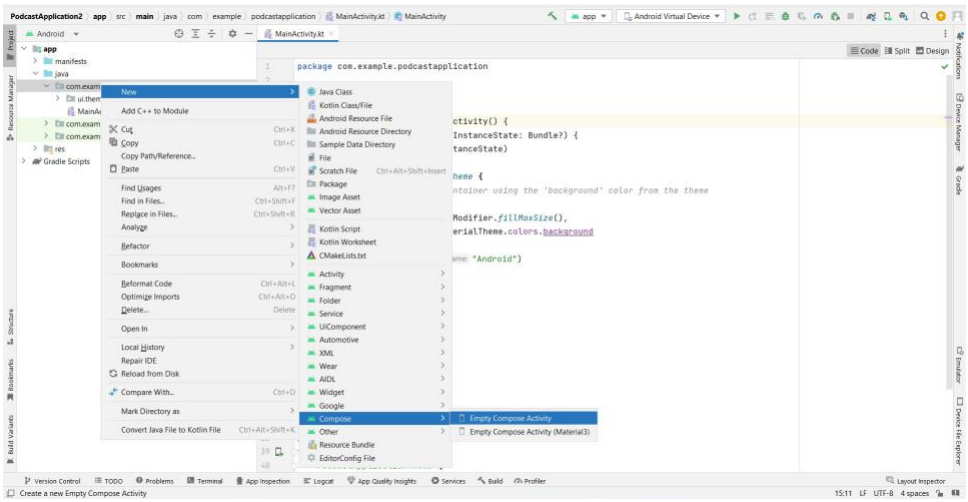


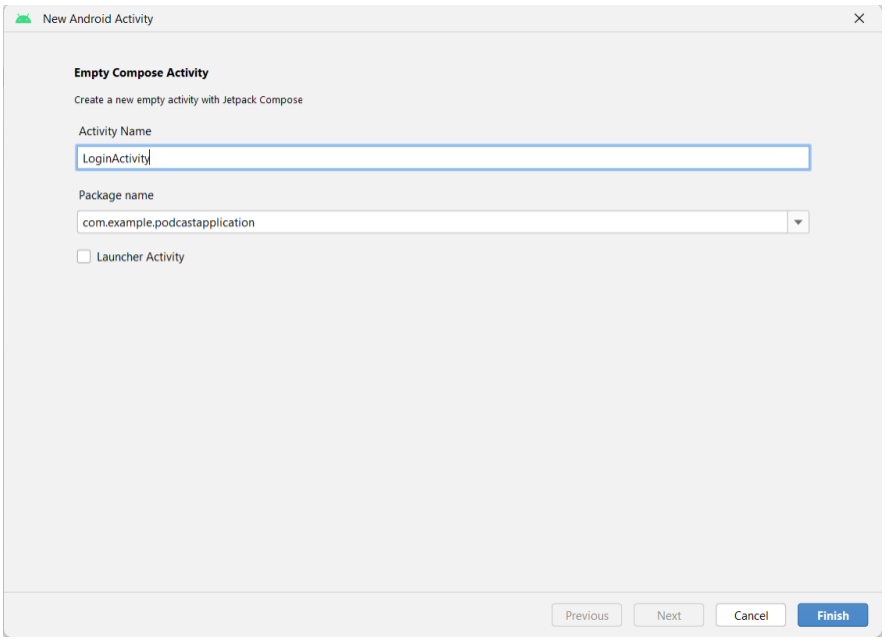
TimeDatabaseHelper class code:
https://github.com/kondasivaprasad/Sleep-Tracker/blob/master/app/src/main/java/com/example/sleeptracking/UserDatabaseHelper.kt

Task 5:

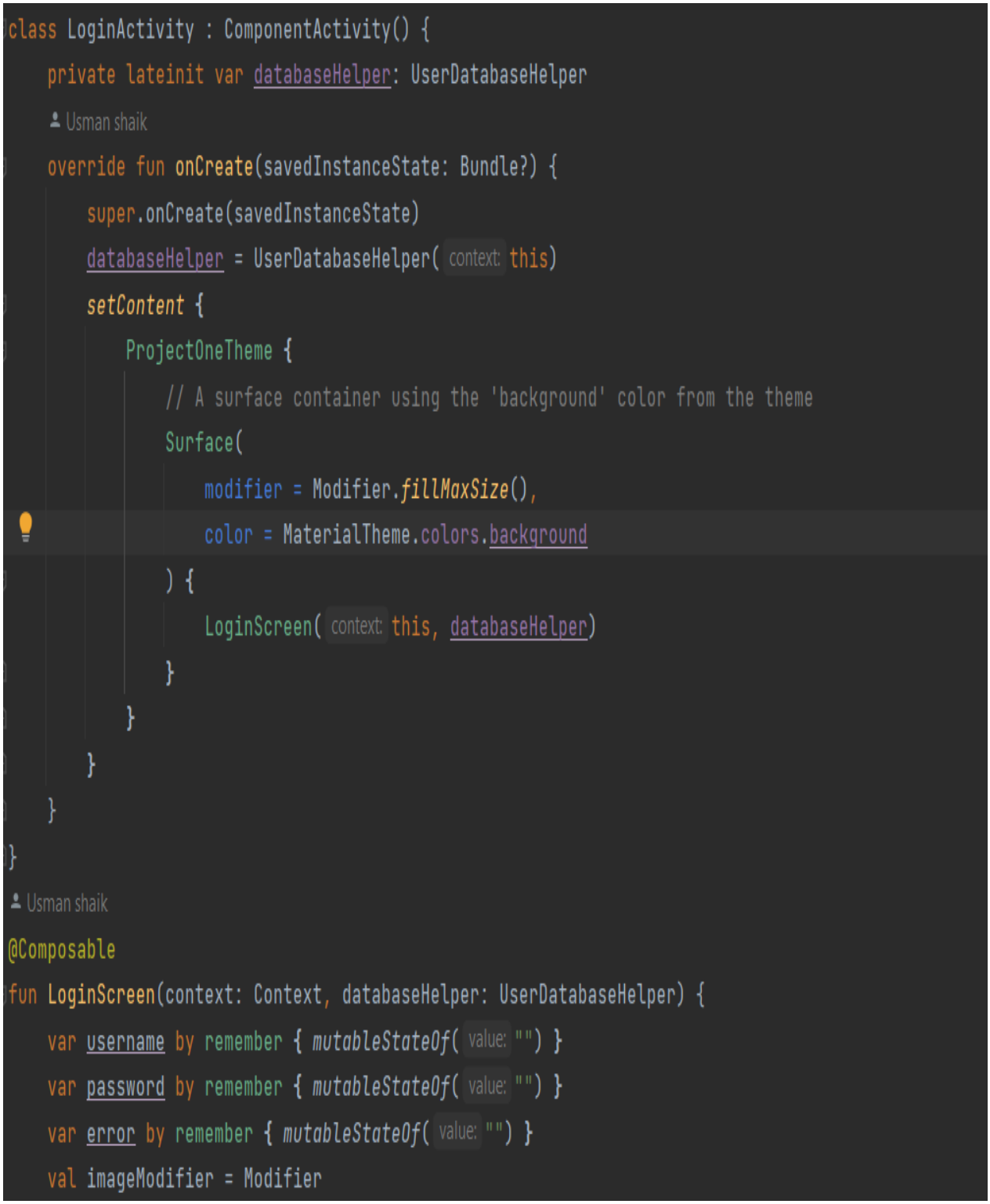
Building application UI and connecting to database.

Step 1: Creating LoginActivity.kt with database





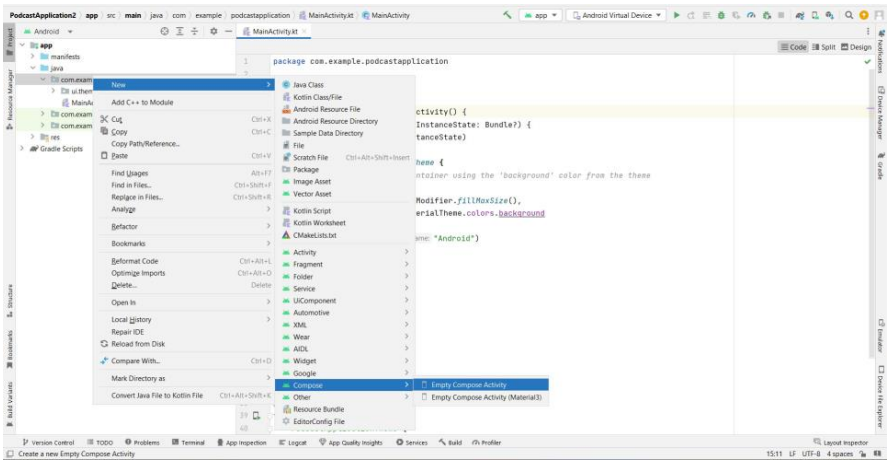
Database connection in LoginActivity.kt:

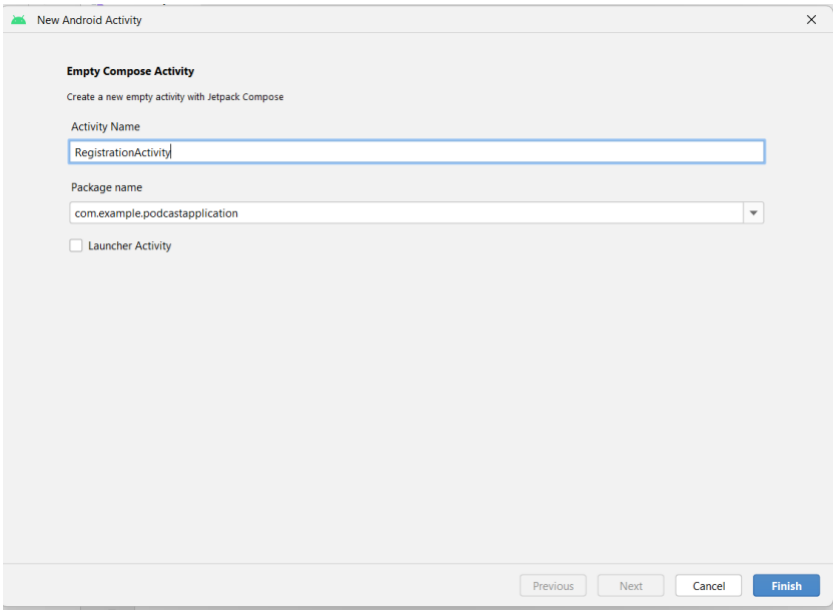


```
class LoginActivity : ComponentActivity() {
    private lateinit var databaseHelper: UserDatabaseHelper
    Usman shaik
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        databaseHelper = UserDatabaseHelper( context: this)
        setContent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

Complete code in below link:
https://github.com/kondasivaprasad/Sleep-Tracker/blob/master/app/src/main/java/com/example/sleeptracking/LoginActivity.kt

Step 2 : Creating RegistrationActivity.kt with database





Database connection in RegistrationActivity.kt

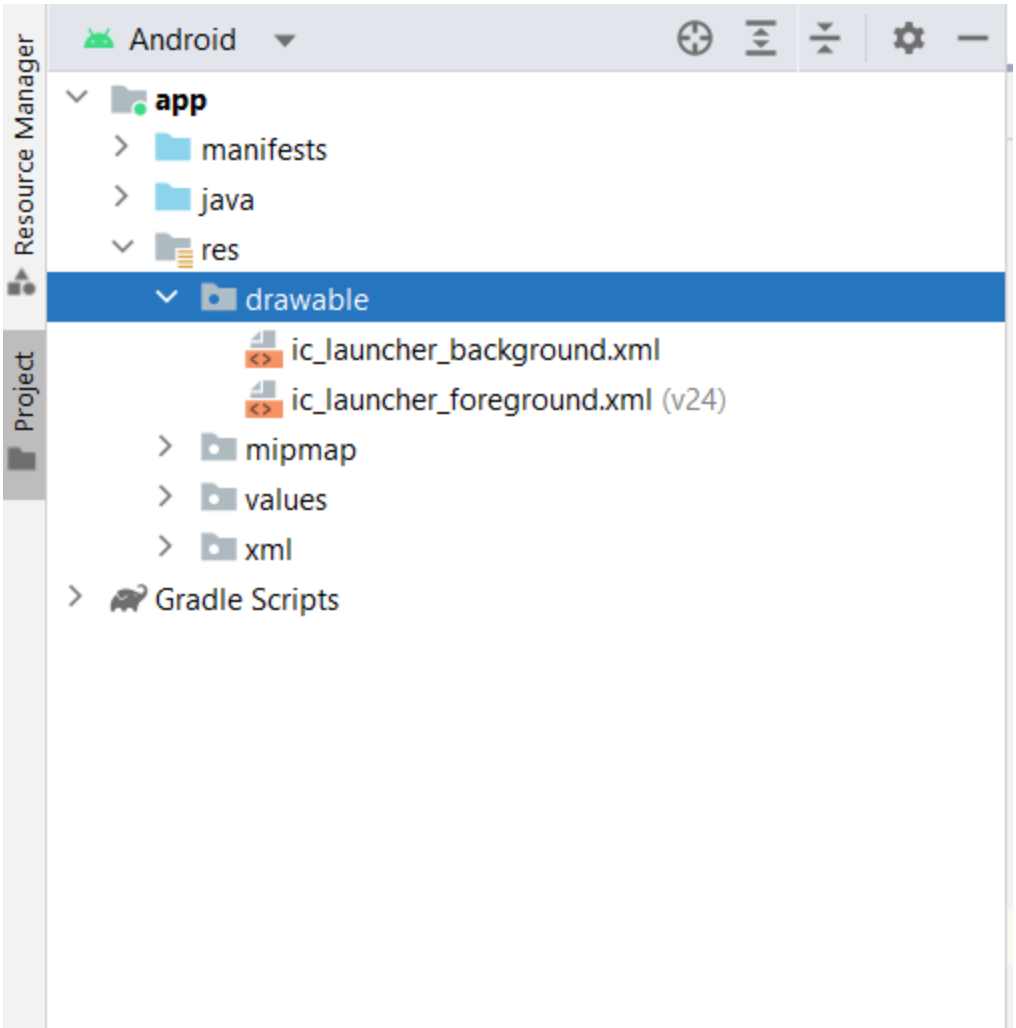
```
class MainActivity2 : ComponentActivity() {
    private lateinit var databaseHelper: UserDatabaseHelper
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        databaseHelper = UserDatabaseHelper(this)
        setContent {
            ProjectOneTheme {
                // A surface container using the 'background' color from the theme
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colors.background
                ) {

                    RegistrationScreen(this,databaseHelper)
                }
            }
        }
    }
}


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

    val imageModifier = Modifier
    Image(
        painterResource(id = R.drawable.sleeptracking),
        contentScale = ContentScale.FillHeight,
        contentDescription = "",
```

Complete code in below link:
https://github.com/kondasivaprasad/Sleep-Tracker/blob/master/app/src/main/java/com/example/sleeptracking/RegistrationActivity.kt

Step 3 : Creating MainActivity.kt file

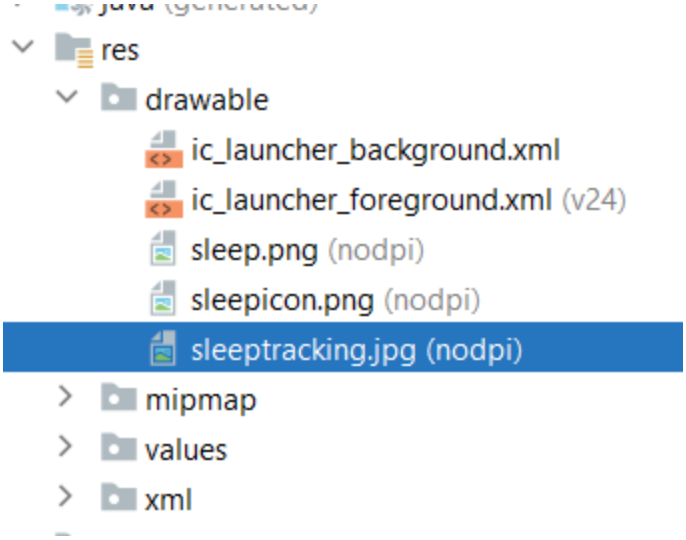
In MainActivity.kt file the main application is developed

- 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/kondasivaprasad/Sleep-Tracker/tree/master/app/src/main/res/drawable

Required drawables

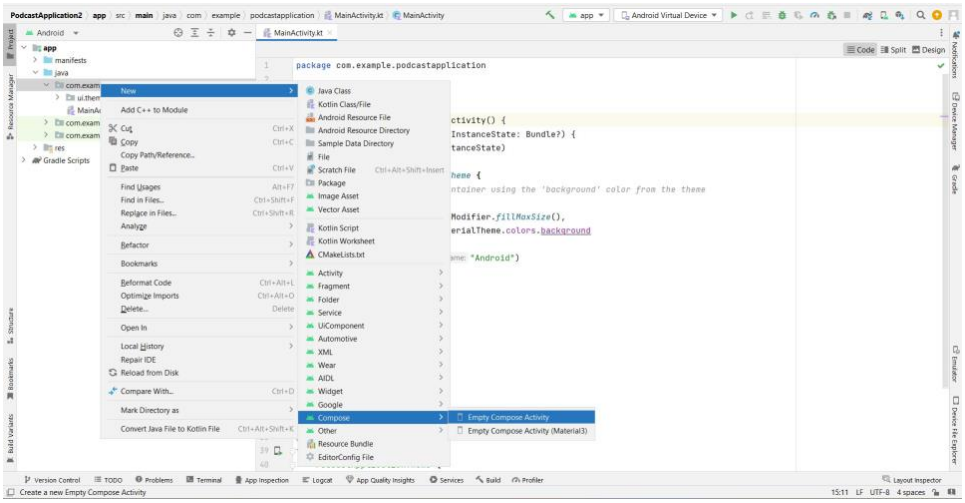


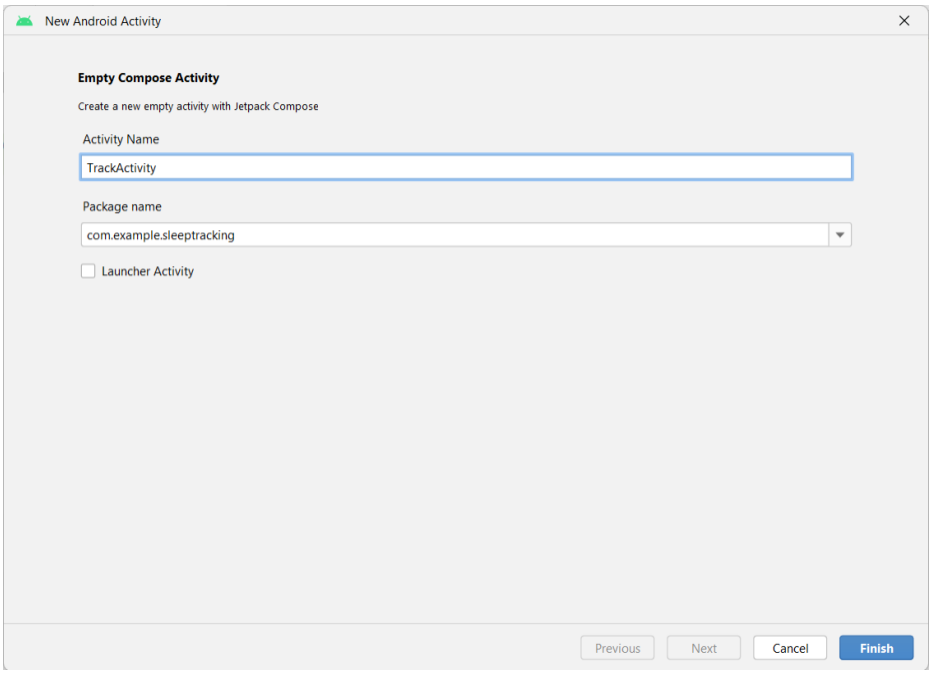
MainActivity.kt

```
class MainActivity : ComponentActivity() {

    private lateinit var databaseHelper: TimeLogDatabaseHelper

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        databaseHelper = TimeLogDatabaseHelper(this)
        databaseHelper.deleteAllData()
        setContent {
            ProjectOneTheme {
                // A surface container using the 'background' color from the theme
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colors.background
                ) {
                    MyScreen(this,databaseHelper)
                }
            }
        }
    }
}
@Composable
fun MyScreen(context: Context, databaseHelper: TimeLogDatabaseHelper) {
    var startTime by remember { mutableStateOf(0L) }
    var elapsedTime by remember { mutableStateOf(0L) }
    var isRunning by remember { mutableStateOf(false) }
    val imageModifier = Modifier
    Image(
        painterResource(id = R.drawable.sleeptracking),
        contentScale = ContentScale.FillHeight,
        contentDescription = "",
        modifier = imageModifier
            .alpha(0.3F),
```

Complete code in below link:
https://github.com/kondasivaprasad/Sleep-Tracker/blob/master/app/src/main/java/com/example/sleeptracking/MainActivity.kt

Step 4 : Creating TrackActivity.kt file





Database connection and fetching in TrackActivity.kt

```kotlin
class TrackActivity : ComponentActivity() {

    private lateinit var databaseHelper: TimeLogDatabaseHelper

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)

        databaseHelper = TimeLogDatabaseHelper(this)
        setContent {
            ProjectOneTheme {
                // A surface container using the 'background' color from the theme
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colors.background
                ) {
                    //ListListScopeSample(timeLogs)

                    val data=databaseHelper.getTimeLogs();
                    Log.d("Sandeep" ,data.toString())
                    val timeLogs = databaseHelper.getTimeLogs()
                    ListListScopeSample(timeLogs)
                }
            }
        }
    }
}


@Composable
fun ListListScopeSample(timeLogs: List<TimeLogDatabaseHelper.TimeLog>) {
    val imageModifier = Modifier
    Image(
```

Complete code in below link:
https://github.com/kondasivaprasad/Sleep-Tracker/blob/master/app/src/main/java/com/example/sleeptracking/TrackActivity.kt

Task 6:

Modifying AndroidManifest.xml

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

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/Theme.SleepTracking"
        tools:targetApi="31">
        <activity
            android:name=".TrackActivity"
            android:exported="false"
            android:label="@string/title_activity_track"
            android:theme="@style/Theme.SleepTracking" />
        <activity
            android:name=".MainActivity"
            android:exported="false"
            android:label="@string/app_name"
            android:theme="@style/Theme.SleepTracking" />
        <activity
            android:name=".MainActivity2"
            android:exported="false"
            android:label="RegistrationActivity"
            android:theme="@style/Theme.SleepTracking" />
        <activity
```

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.

Changed AndroidManifest.xml.

```
    <activity
        android:name=".TrackActivity"
        android:exported="false"
        android:label="@string/title_activity_track"
        android:theme="@style/Theme.SleepTracking" />
    <activity
        android:name=".MainActivity"
        android:exported="false"
        android:label="@string/app_name"
        android:theme="@style/Theme.SleepTracking" />
    <activity
        android:name=".MainActivity2"
        android:exported="false"
        android:label="RegistrationActivity"
        android:theme="@style/Theme.SleepTracking" />
    <activity
        android:name=".LoginActivity"
        android:exported="true"
        android:label="@string/app_name"
        android:theme="@style/Theme.SleepTracking">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />

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

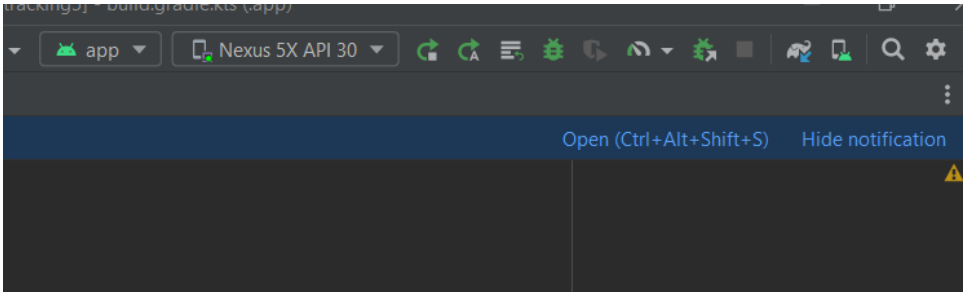
Complete AndroidManifest.xml code: https://github.com/kondasivaprasad/Sleep-Tracker/blob/master/app/src/main/AndroidManifest.xml

Task 7:

Running the application.

Step 1: Run apps on a hardware device
https://developer.android.com/studio/run/device

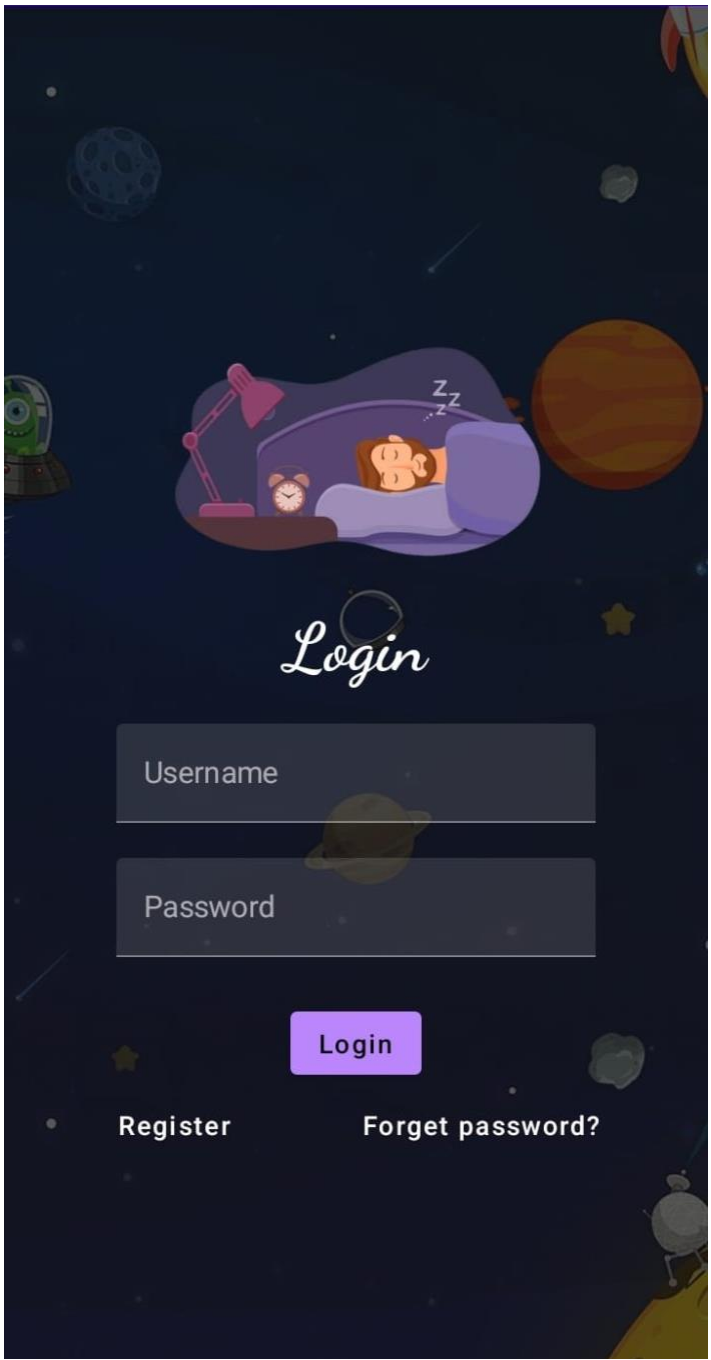
Step 2: Run the application in Mobile



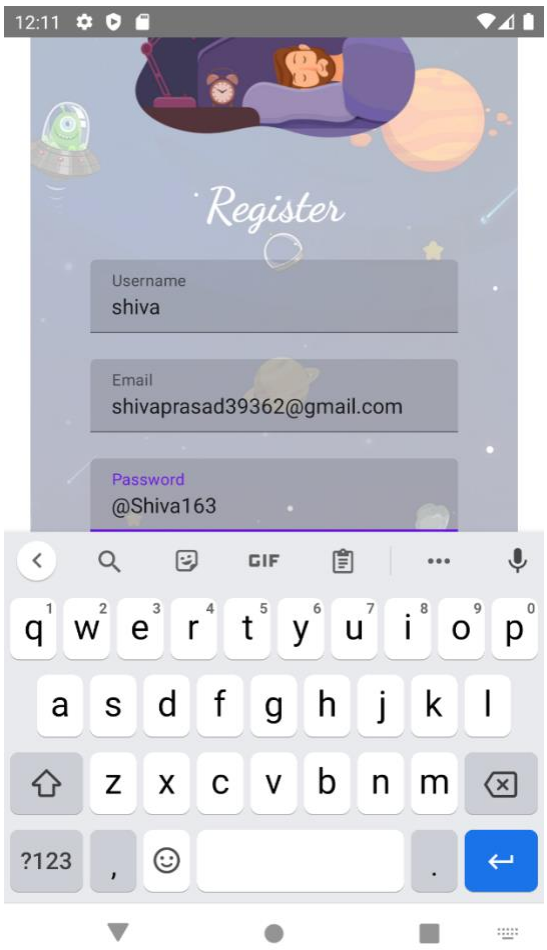
Complete Project Link: https://github.com/kondasivaprasad/Sleep-Tracker

Final Output of the Application :

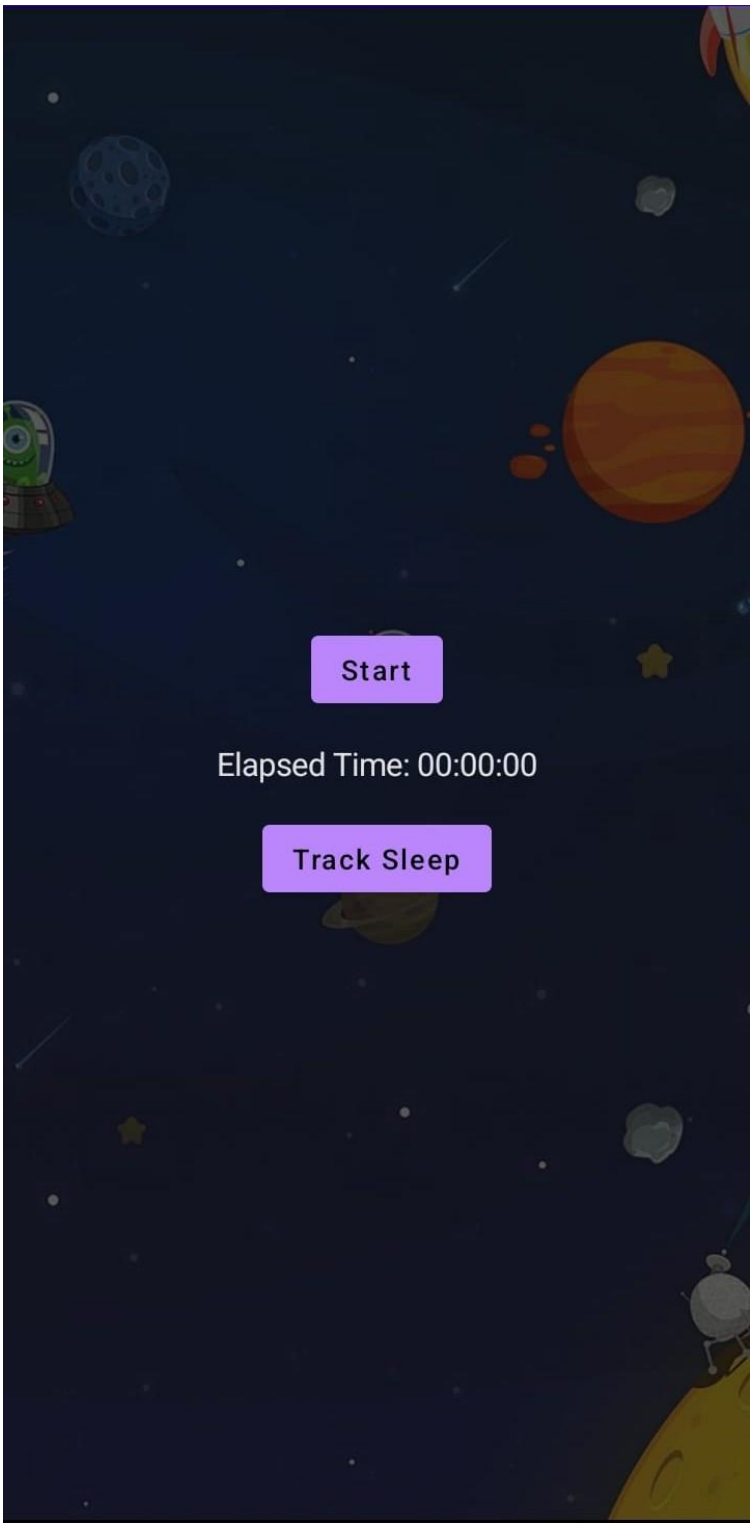
Login Page :



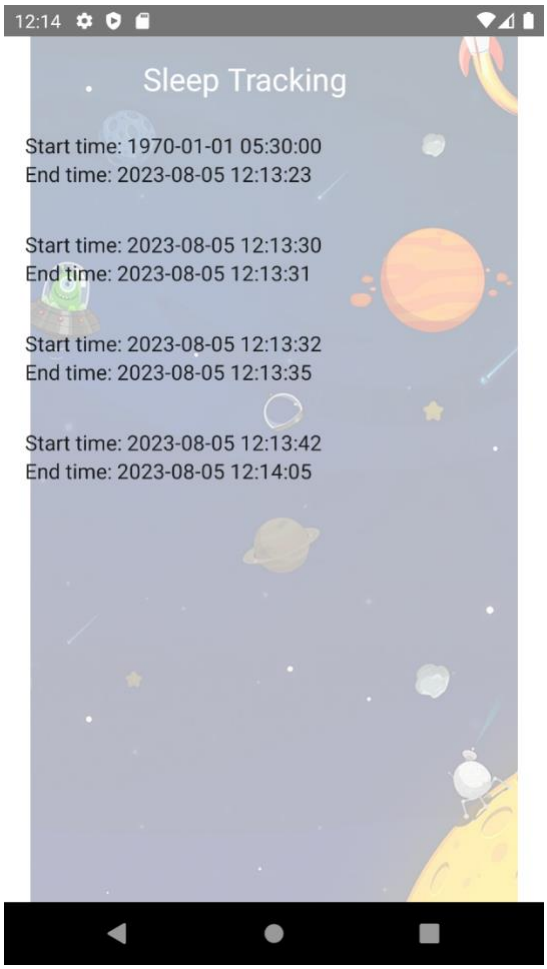
Registration Page:



Main Page:


Start
Elapsed Time: 00:00:00
Track Sleep

Track Sleep Page:



Submitted by:

Team ID: LTVIP2023TMID04297

Team Size: 4

Team Leader: Konda Sivaprasad

Team Member: Kolli Supraja

Team Member: Kolimi Lalusab

Team member: Kamireddy Pavankumarreddy