Mobile Application Development Practical File

A PRACTICAL FILE SUBMITTED IN PARTIAL FULFILLMENT OF COURSE

Mobile Application Development Laboratory

BACHELORS OF TECHNOLOGY

Department of Information Technology

SUBMITTED BY

Hima Gandhi — 1921041 —1905339

SUBMITTED TO

Prof. Ranjodh Kaur

Session - July-Dec 2022

GURU NANAK DEV ENGINEERING COLLEGE

LUDHIANA-141006, INDIA

Contents

List of Figures

1 To study design aspects of development environment like Android, IOS.

1.0.1 Android

Android's Java environment can be broken down into a handful of key sections. When you understand the contents in each of these sections, the Javadoc reference material that ships with the SDK becomes a real tool and not just a pile of seemingly unrelated material. You might recall that Android isn't a strictly Java ME software environment, but there's some commonality between the Android platforms and other Java development platforms. The next few sections review some of the Java packages (core and optional) in the Android SDK and where you can use them. Android Studio is the official IDE for Android development, and with a single download it includes everything you need to begin developing Android apps as you can see below

- IntelliJ IDE + Android Studio plugin
- Android SDK Tools
- Android Platform-tools
- Android Emulator with an Android system image including Google Play Services

Android Studio provides tools for building apps on every type of Android device. Code editing, debugging, performance tooling, a flexible build system, and an instant build or deploy system are included in Android studio.

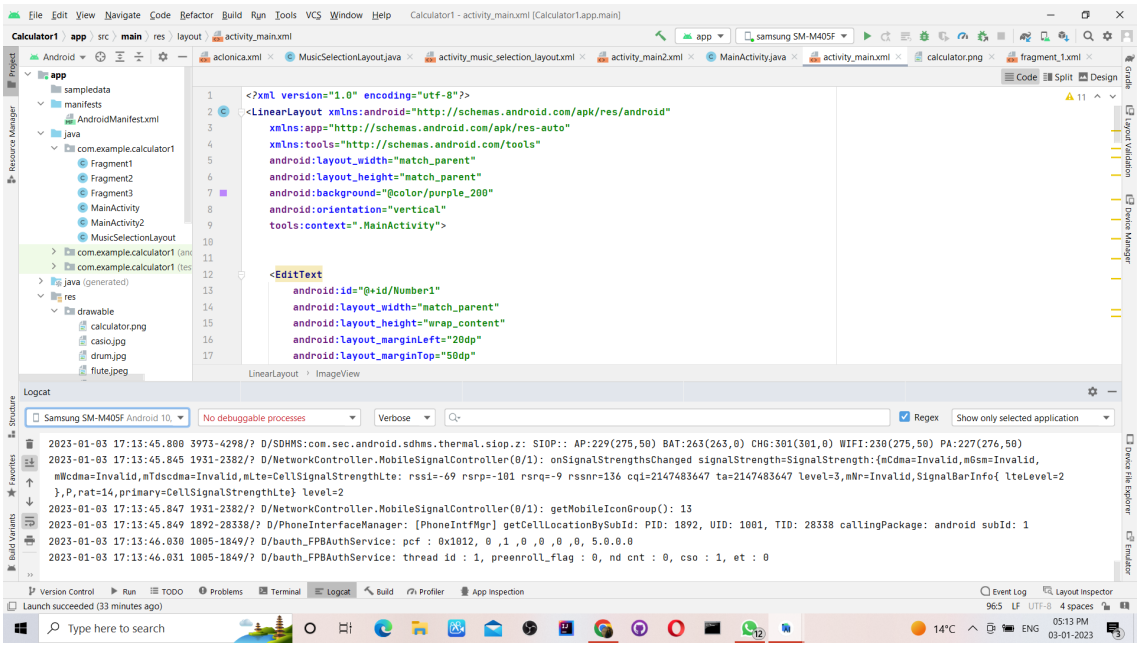

Figure 1.1: Android Studio

1. Command Line Tools:- The Android SDK tools available from the SDK Manager provide additional command-line tools to help you during your Android development. The tools are classified into two groups: SDK tools and platform tools. SDK tools are platform

independent and are required no matter which Android platform you are developing on. Platform tools are customized to support the features of the latest Android platform.

2. SDK Tools:- The SDK tools are installed with the SDK starter package and are periodically updated. The SDK tools are required if you are developing Android applications. The most important SDK tools include the Android SDK Manager (Android sdk), the AVD Manager (Android AVD) the emulator (emulator), and the Dalvik Debug Monitor Server (DDMS).

3. Virtual Device Tools:-

   - Android Virtual Device Manager
   - Android Emulator (emulator)
   - mksdcard

4. Development Tools:- Hierarchy Viewer (hierarchyviewer) - Provides a visual representation of the layout's View hierarchy with performance information for each node in the layout, and a magnified view of the display to closely examine the pixels in your layout.

5. SDK Manager:- SDK Manager lets you manage SDK packages, such as installed platforms and system images. sqlite3 - Lets you access the SQLite data files created and used by Android applications.

6. Debugging tools:-

   - Android Monitor
   - adb
   - Dalvik Debug Monitor Server (DDMS)
   - Device Monitor
   - systrace

7. Build Tools:-

   - apksigner
   - JOBB
   - ProGuard
   - zipalign

1.0.2 iOS

iOS application development is the process of making mobile applications for Apple hardware, including iPhone, iPad and iPod Touch. The software is written in the Swift programming language or Objective-C and then deployed to the App Store for users to download.

Requirements for iOS development environment:-

- An Apple Mac computer running the latest version of macOS.

- Xcode, which is the integrated development environment (IDE) for macOS, available as a free download from the Mac App Store.
- An active Apple Developer account, which requires a USD 99 annual fee.

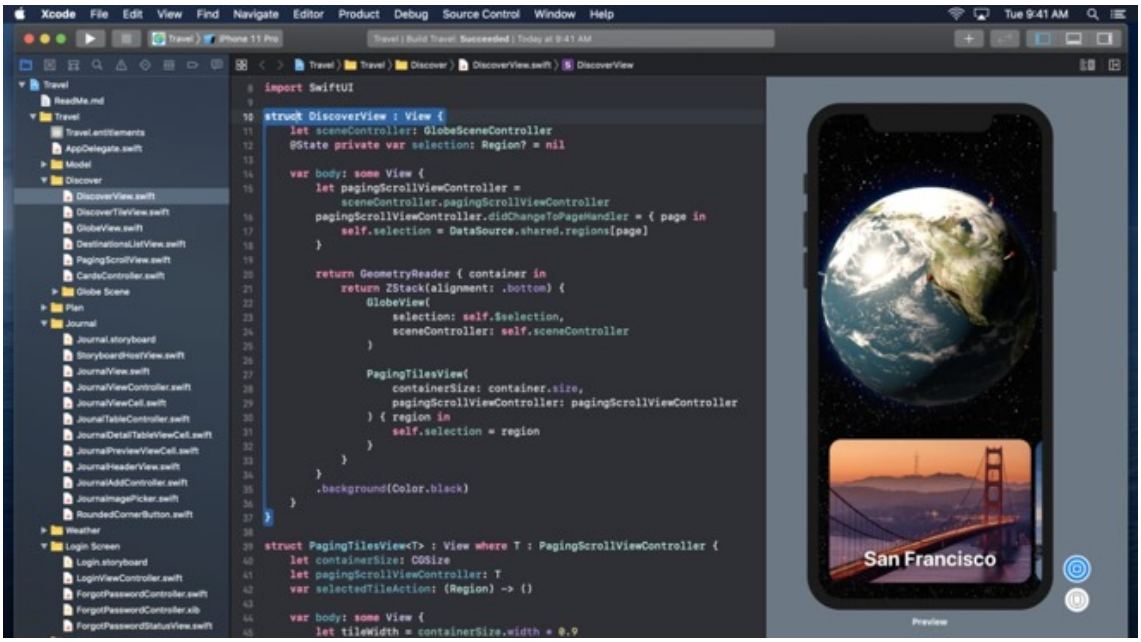

Figure 1.2: xCode iOS IDE

iOS development programming languages:-

- **Objective-C:** Developed in the early 1980s, Objective-C was the primary programming language for all Apple products for decades. Derived from the C language, Objective-C is an object-oriented programming language centered on passing messages to different processes (as opposed to invoking a process in traditional C programming). Many developers choose to maintain their legacy applications written in Objective-C instead of integrating them into the Swift framework, which was introduced in 2014.
- **Swift:** The Swift programming language is the new "official" language of iOS. While it has many similarities to Objective-C, Swift is designed to use a simpler syntax and is more focused on security than its predecessor. Because it shares a run time with Objective-C, you can easily incorporate legacy code into updated apps. Swift is easy to learn, even for people just beginning to program. Because Swift is faster, more secure and easier to use than Objective-C, you should plan to use it to develop your iOS app unless you have a compelling reason to stick with Objective-C.

One of the major advantages of iOS app development is the extensive collection of developer resources available to you. Because of the standardization, functionality and consistency of iOS app development, Apple is able to release native APIs and libraries as kits that are stable, feature-rich and easy to use. You can use these iOS SDKs to seamlessly integrate your app into Apple's existing infrastructure.

For example, if you're working on an app controller for a smart toaster oven, you can use HomeKit to standardize the communication between the toaster and the phone. There are kits for game development (such as SpriteKit, GameplayKit and ReplayKit), health apps, maps, cameras, as well as Siri, Apple's virtual assistant.

2 Android Development Environment: To setup Android studio2 and study its basic components.

1. Head over to https://developer.android.com/studiodownloads to get the Android Studio executable or zip file.
2. Click on the Download Android Studio Button.

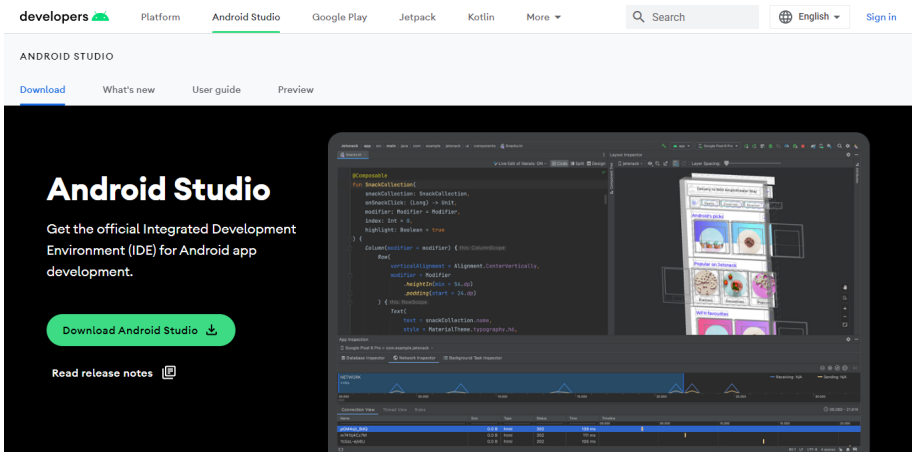

Figure 2.1: Official Android Studio download page

Click on the "I have read and agree with the above terms and conditions" checkbox followed by the download button.

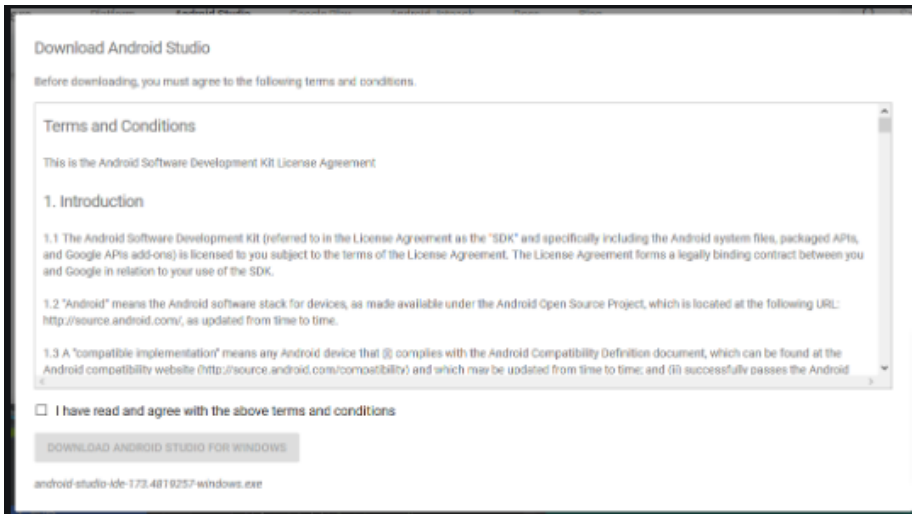

Figure 2.2: Terms and conditions

Click on the Save file button in the appeared prompt box and the file will start downloading.

3. After the downloading has finished, open the file from downloads and run it. It will prompt the following dialog box.

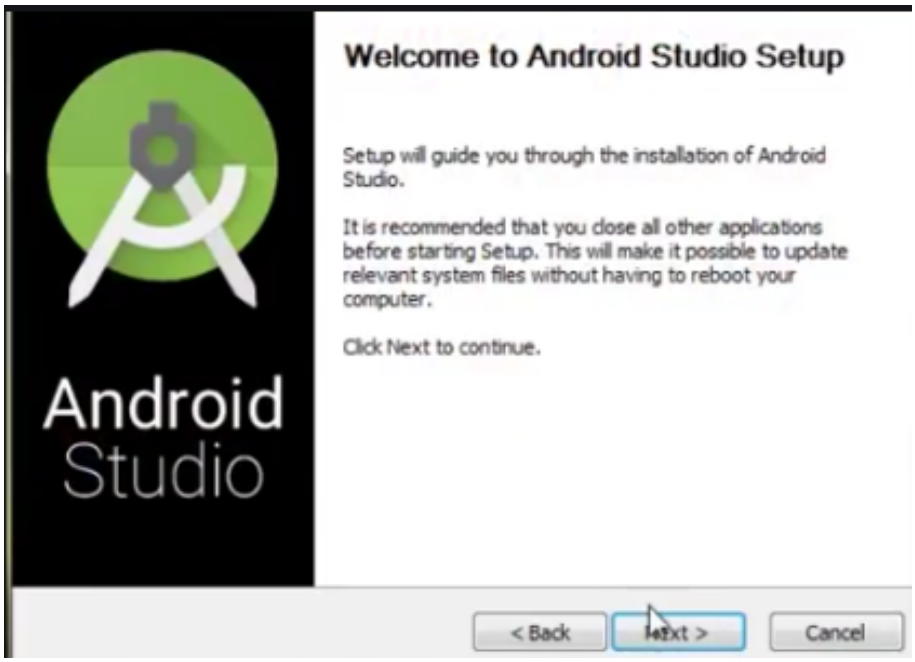

Figure 2.3: Download completed

Click on next. In the next prompt, it'll ask for a path for installation. Choose a path and hit next.

4. It will start the installation, and once it is completed, it will be like the image shown below.

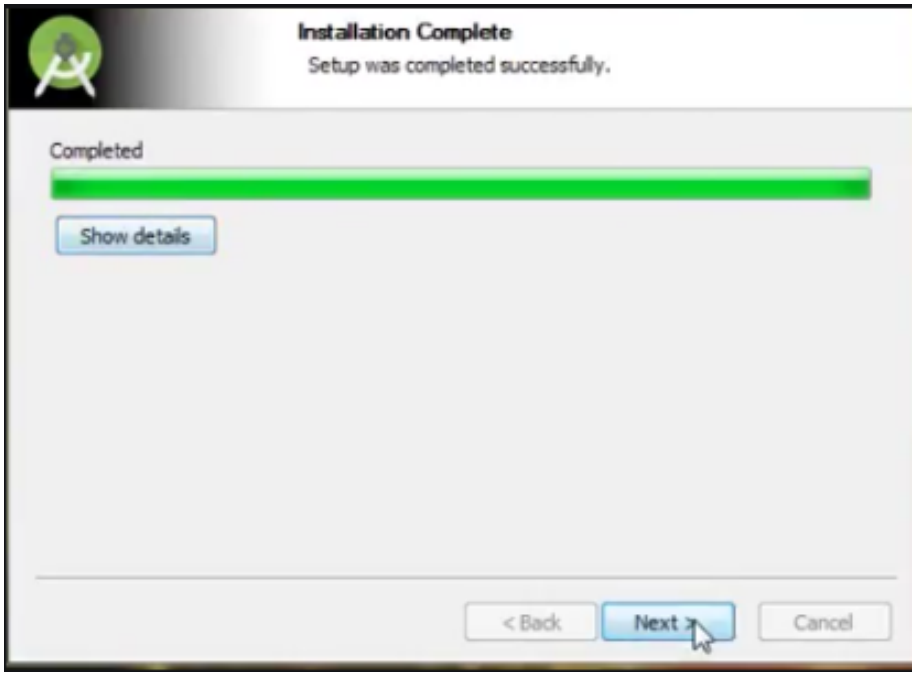

Figure 2.4: Installation process

Click on next.

Figure 2.5: Installation process

5. Once "Finish" is clicked, it will ask whether the previous settings need to be imported [if the android studio had been installed earlier], or not. It is better to choose the 'Don't import Settings option'.

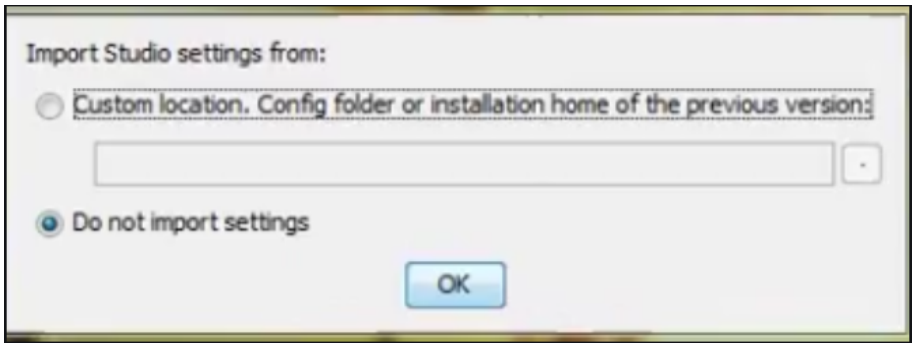

Figure 2.6: Installation process

   Click on OK.

6. This will start the Android Studio.

Figure 2.7: Installation completed

7. After it has found the SDK components, it will redirect to the Welcome dialog box.

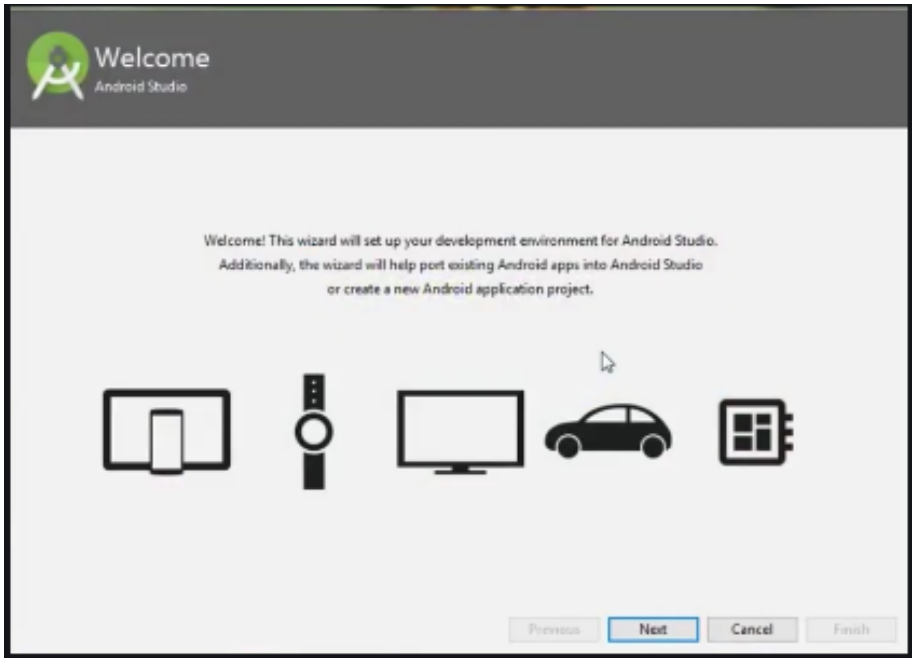

Figure 2.8: Welcome screen

8. Choose Standard and click on Next. Now choose the theme, whether the Light theme or the Dark one. The light one is called the IntelliJ theme whereas the dark theme is called Dracula. Choose as required. Click on Next.

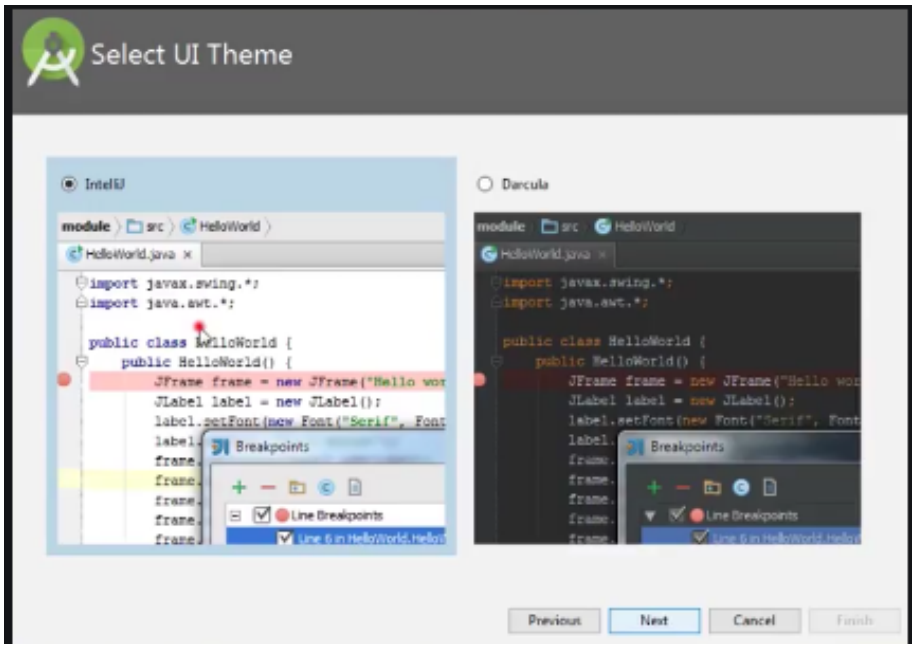

Figure 2.9: Theme selection

9. Now it is time to download the SDK components.

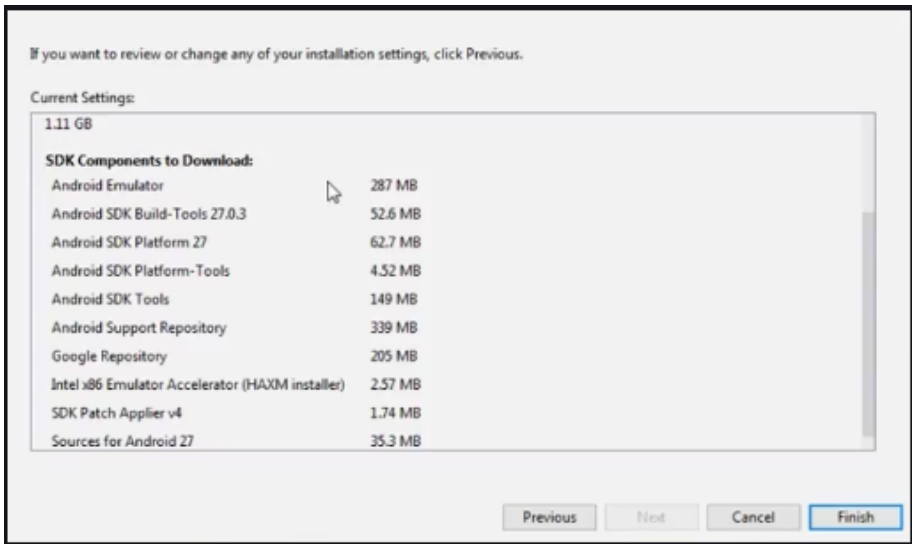

Figure 2.10: Installation process

Click on Finish. Components begin to download let it complete.

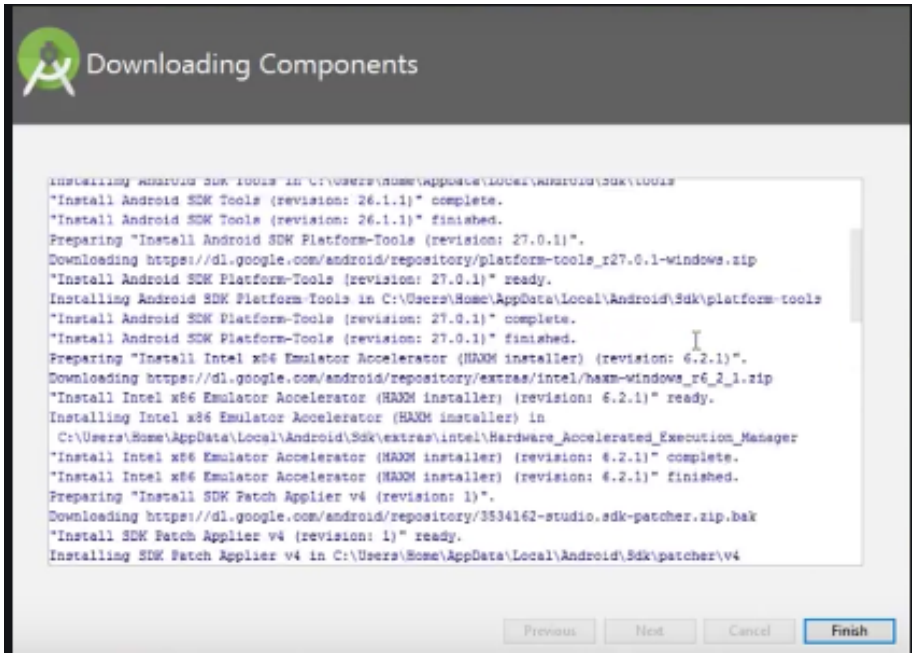

Figure 2.11: Installation process

10. Click on Start a new Android Studio project to build a new app.

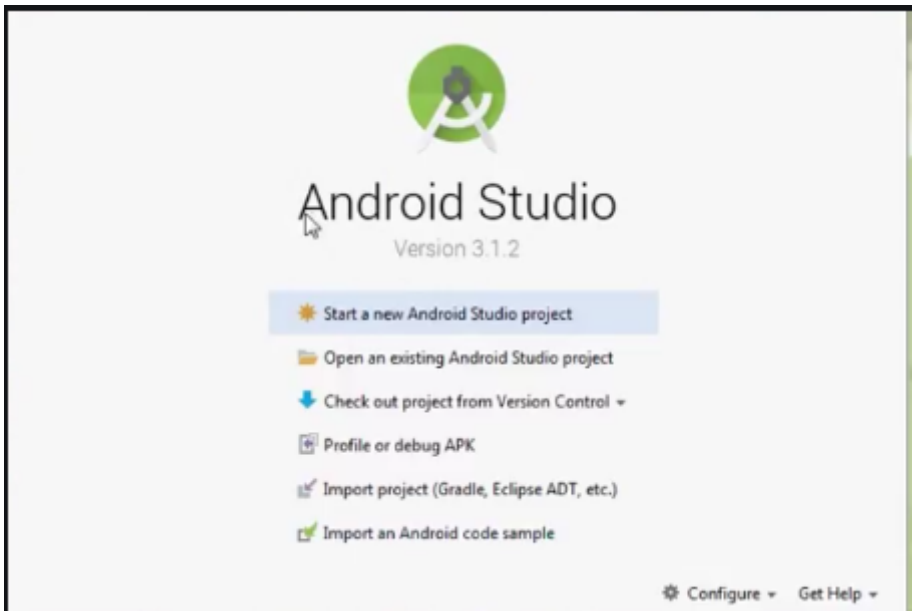

Figure 2.12: First Project

3 Android User Interface Design: To study various XML files needed for interface design.

XML stands for Extensible Markup Language. XML is a markup language much like HTML used to describe data. It is derived from Standard Generalized Markup Language(SGML). In Android, the XML is used to implement UI-related data, and it's a lightweight markup language that doesn't make layout heavy. XML only contains tags, while implementing they need to be just invoked. Syntax for XML tag:-

```
<tag_name>Hello World!</tag_name>
```

Different XML files serve different purposes in Android Studio. The list of various XML files in Android Studio with their purposes is discussed below.

1. **Layout XML files in android:** The Layout XML files are responsible for the actual User Interface of the application. It holds all the widgets or views like Buttons, TextViews, EditTexts, etc. which are defined under the ViewGroups. The Location of the layout files in Android is:

   ```
   app -> src -> main -> res -> layout
   ```

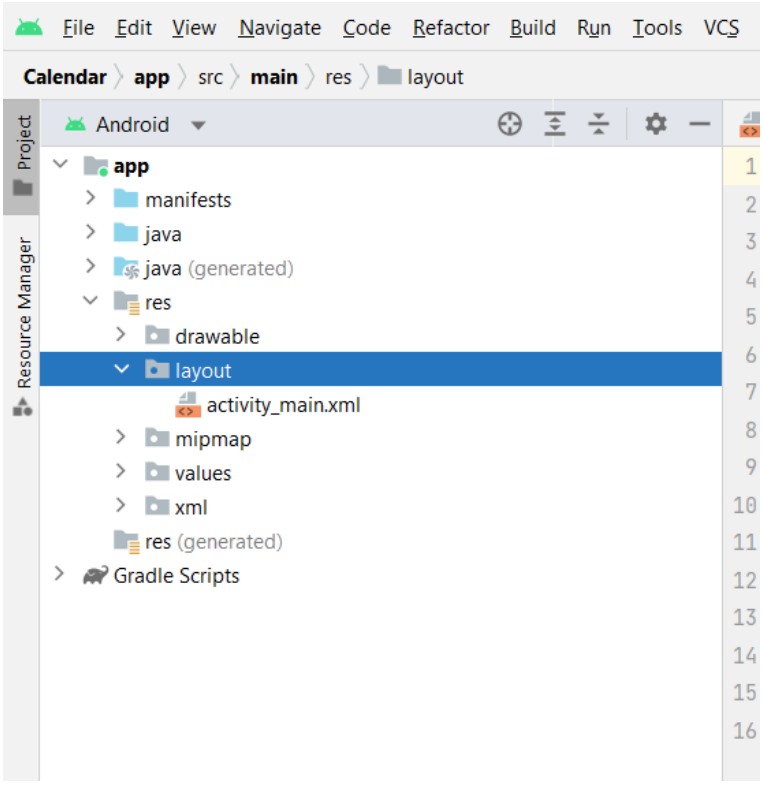

Figure 3.1: Layout XML location

2. **AndroidManifest.xml file:** This file describes the essential information about the application's, like the application's package names which matches code's namespaces, a

component of the application like activities, services, broadcast receivers, and content providers. Permission required by the user for the application features also mentioned in this XML file. Location of the AndroidManifest.xml file:

```
app -> src -> main
```

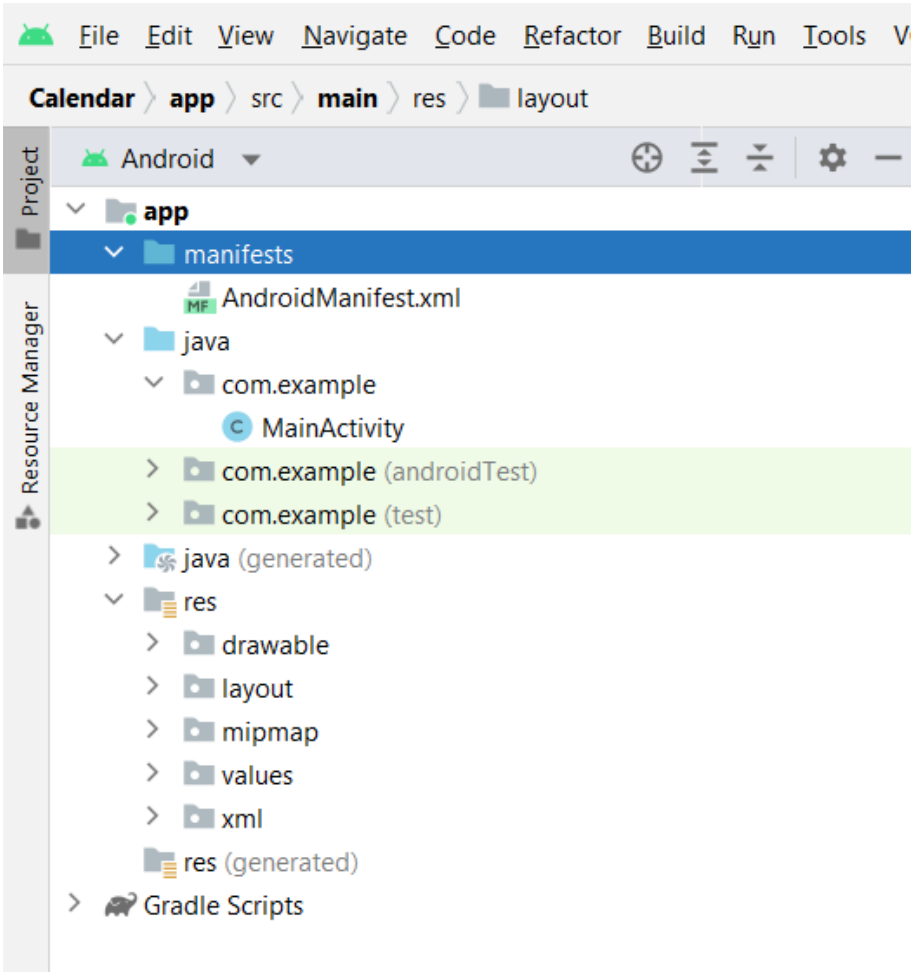

Figure 3.2: Manifest XML location

3. **strings.xml file:** This file contains texts for all the TextViews widgets. This enables reusability of code, and also helps in the localization of the application with different languages. The strings defined in these files can be used to replace all hardcoded text in the entire application. Location of the strings.xml file

```
app -> src -> main -> res -> values
```

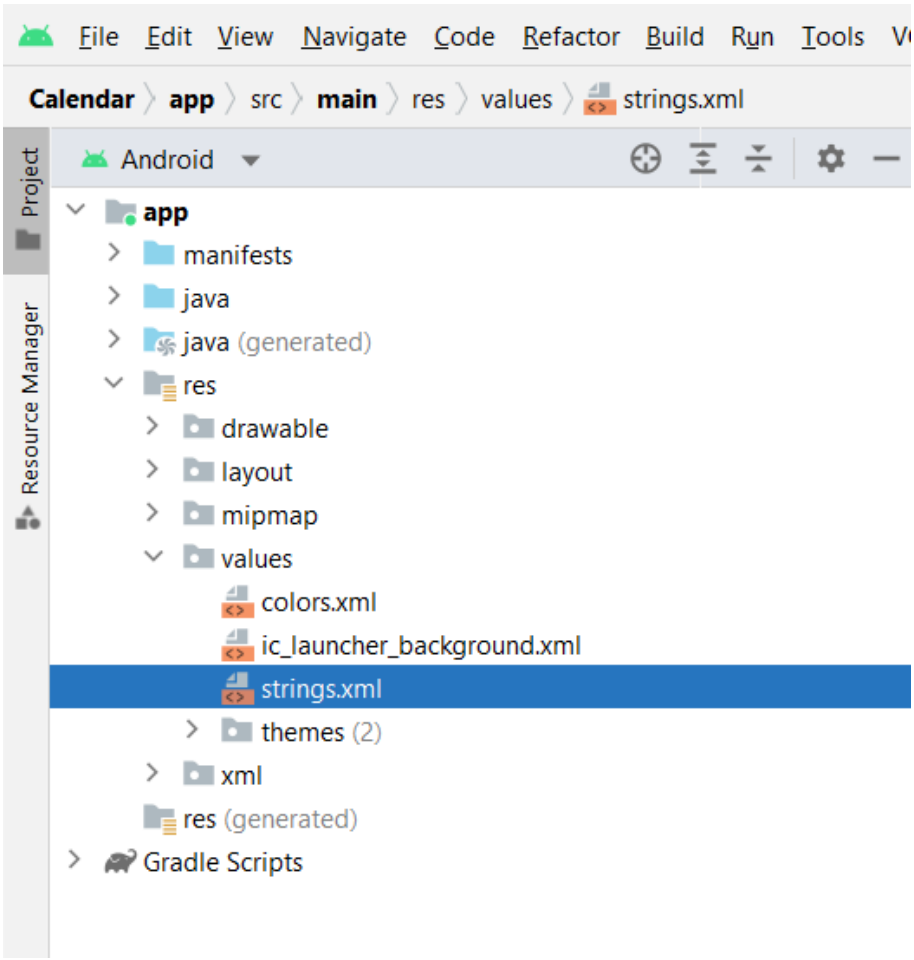

Figure 3.3: String XML location

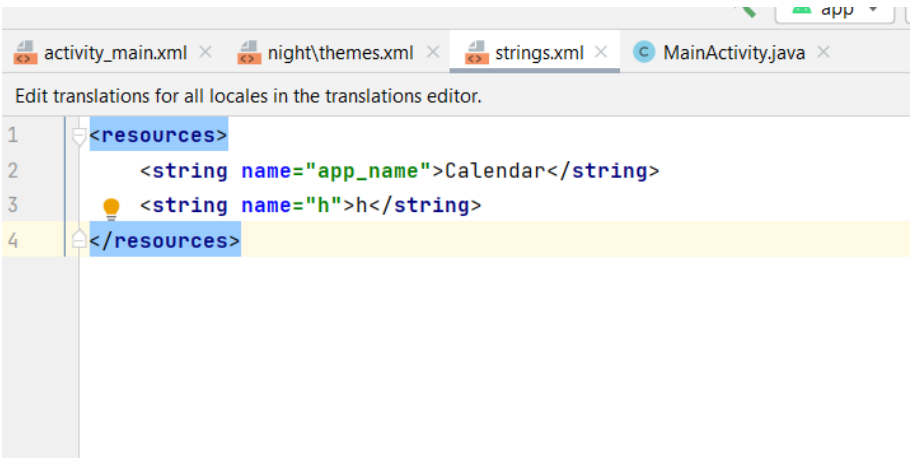

Figure 3.4: String XML Code

4. **themes.xml file:** This file defines the base theme and customized themes of the application. It also used to define styles and looks for the UI(User Interface) of the application. By defining styles we can customize how the views or widgets look on the User Interface. Location of styles.xml file.

```
app -> src -> main -> res -> values
```

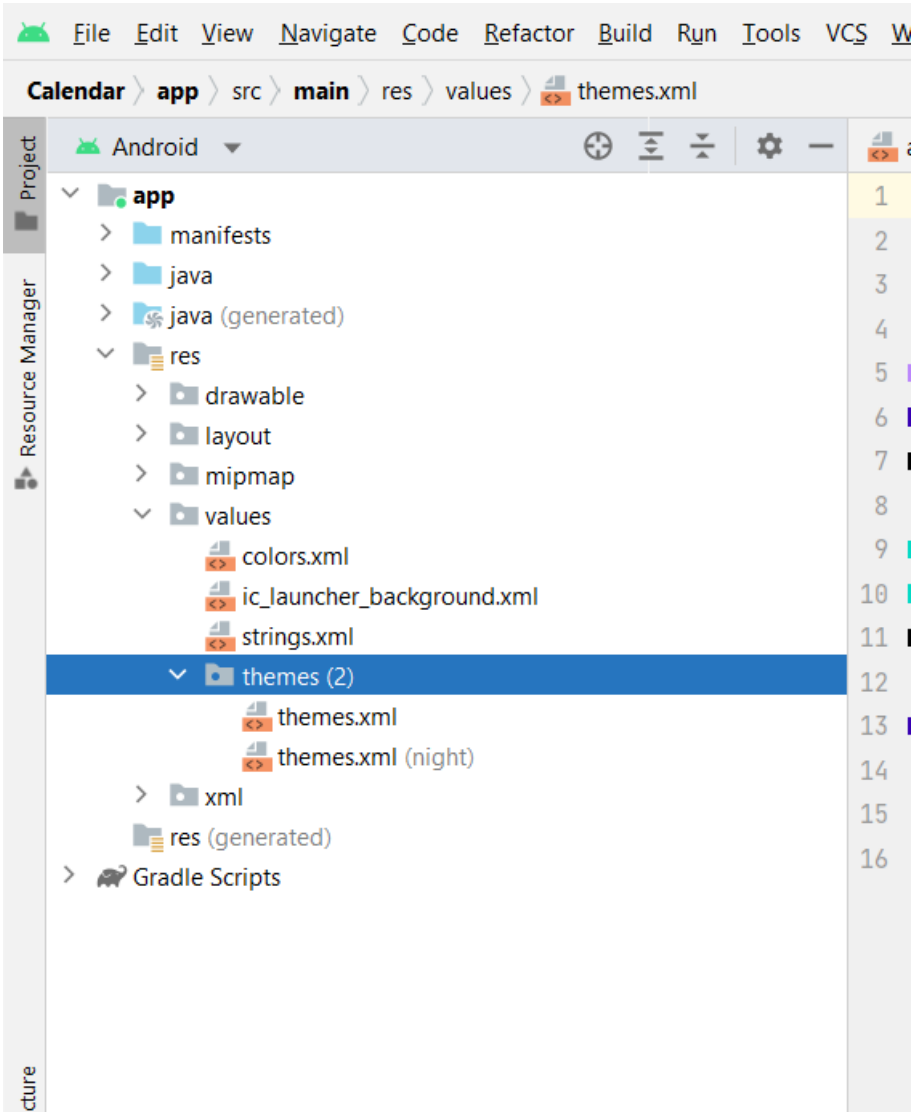

Figure 3.5: Themes XML location

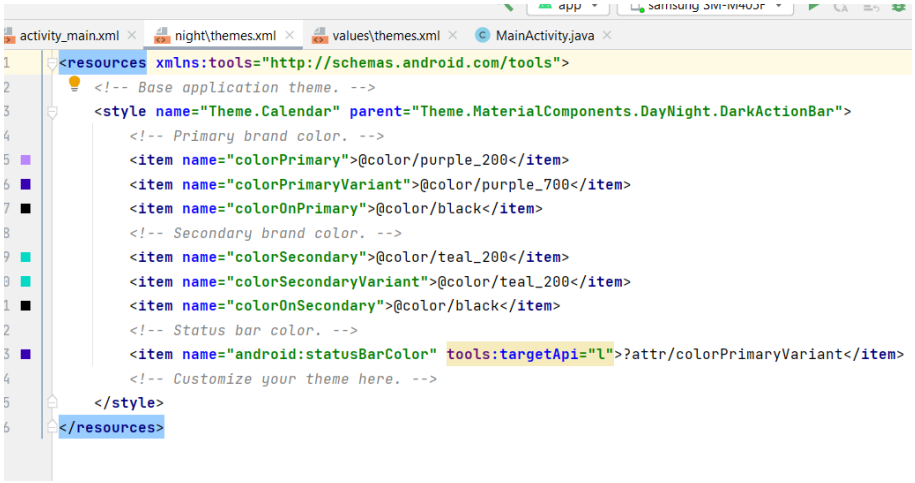

Figure 3.6: Themes XML code

5. **Drawable XML files:** These are the XML files that provide graphics to elements like custom background for the buttons and its ripple effects, also various gradients can be created. This also holds the vector graphics like icons. Using these files custom layouts can be constructed for EditTexts. Location for the Drawable files are:

   `app -> src -> main -> res -> drawable`

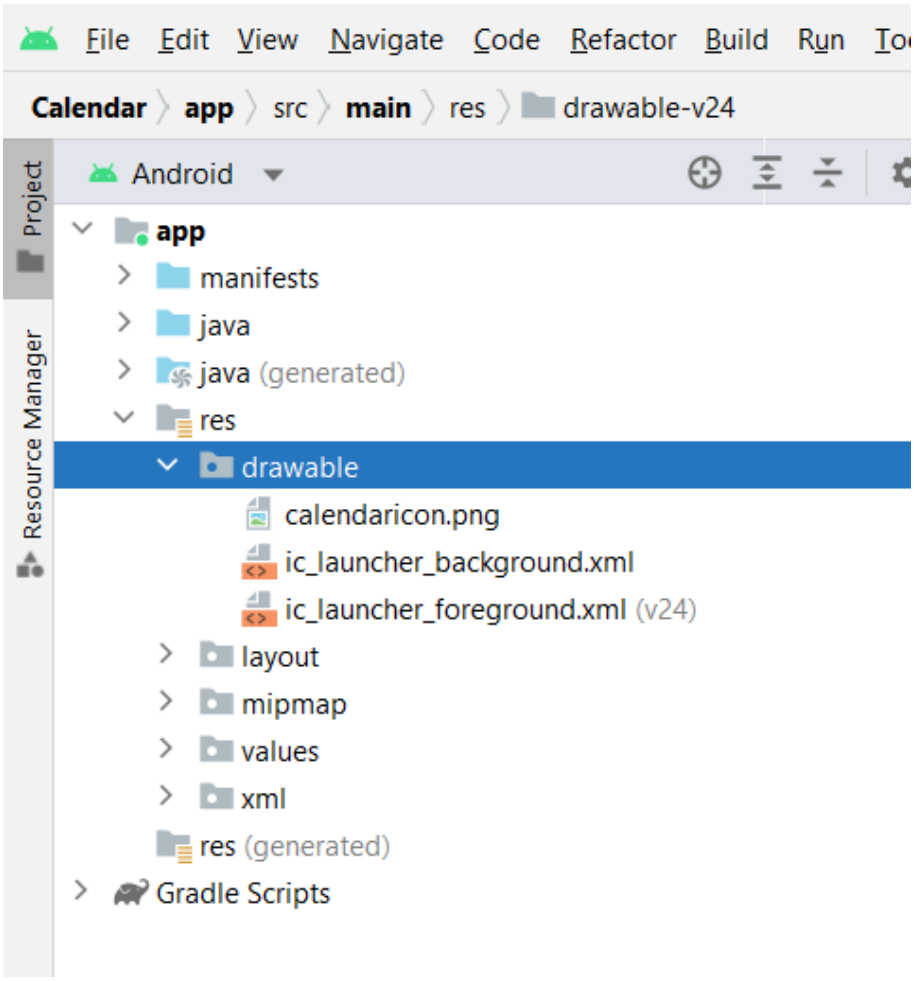

Figure 3.7: Drawable XML location

6. **colors.xml file:** The colors.xml file is responsible to hold all the types of colors required for the application. It may be primary brand color and its variants and secondary brand color and its variants. The colors help uphold the brand of the applications. So the colors need to be decided cautiously as they are responsible for the User Experience. The colors need to be defined in the hex code format. Location of colors.xml file:

   ```
   app -> src -> main -> res -> values
   ```

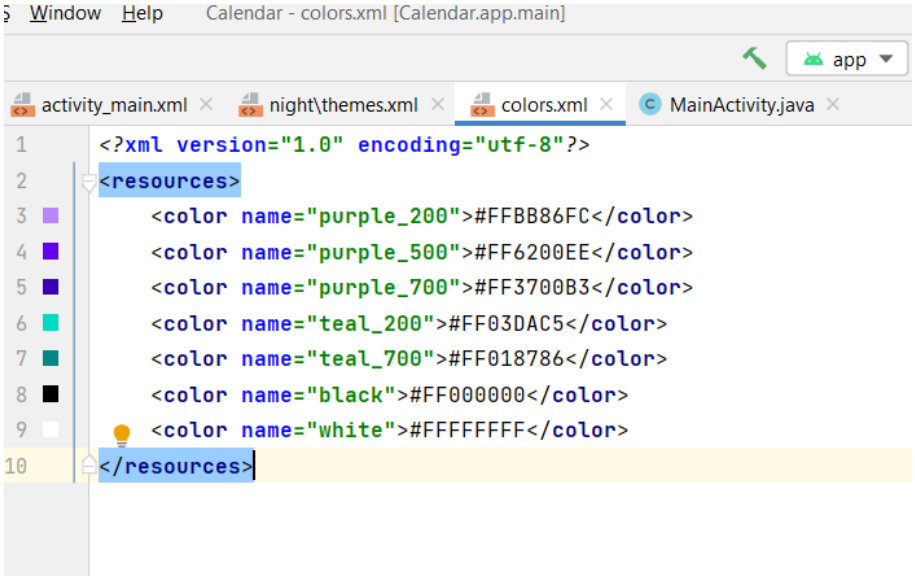

Figure 3.8: colors XML code

7. **dimens.xml file:** As the file name itself suggests that the file is responsible to hold the entire dimensions for the views. it may be the height of the Button, padding of the views, the margin for the views, etc. The dimensions need to in the format of pixel density(dp) values. Which replaces all the hard-coded dp values for the views. This file needs to be created separately in the values folder. Location to create dimens.xml file:

   ```
   app -> src -> main -> res -> values
   ```

3.0.1 Calculator UI development - XML Code

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    android:background="@color/purple_200"
    android:orientation="vertical"
    tools:context=".MainActivity">


    <EditText
        android:id="@+id/Number1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginTop="50dp"
        android:layout_marginRight="20dp"
        android:ems="10"
        android:hint="Enter 1st Number"
        android:inputType="number"
        android:minHeight="48dp" />

    <EditText
        android:id="@+id/Number2"
        android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="50dp"
    android:layout_marginRight="20dp"
    android:ems="10"
    android:hint="Enter 2nd Number"
    android:inputType="number"
    android:minHeight="48dp" />

<Button
    android:id="@+id/Add"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="20dp"
    android:layout_marginRight="20dp"
    android:text="Addition" />

<Button
    android:id="@+id/Sub"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="20dp"
    android:layout_marginRight="20dp"
    android:text="Subtraction" />

<Button
    android:id="@+id/Mul"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="20dp"
    android:layout_marginRight="20dp"
    android:text="Multiplication" />

<Button
    android:id="@+id/Div"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="20dp"
    android:layout_marginRight="20dp"
    android:text="DIVISION" />

<TextView
    android:id="@+id/Result"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="20dp"
    android:layout_marginRight="20dp"
    android:text="Result= "
    android:textColor="#FFFFFF"
    android:textSize="20sp"
    android:textStyle="bold" />

<Button
    android:id="@+id/Reset"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"

    android:layout_marginLeft="20dp"
    android:layout_marginTop="20dp"
    android:layout_marginRight="20dp"
    android:backgroundTint="@color/design_default_color_error"
    android:text="Reset"
    android:textColor="@color/white" />
```

```
    <ImageView
        android:id="@+id/imageView3"
        android:layout_width="425dp"
        android:layout_height="127dp"
        android:layout_marginTop="10dp"
        app:srcCompat="@drawable/calculator" />
    <!-- <Button
          android:id="@+id/Page"
          android:layout_width="wrap_content"
          android:layout_height="60dp"
          android:layout_marginStart="140dp"
          android:layout_marginTop="20dp"
          android:backgroundTint="#E91E63"
          android:text="Click Here!"
          android:textAlignment="center" />
   -->
</LinearLayout>
```

3.0.2 Output Screen

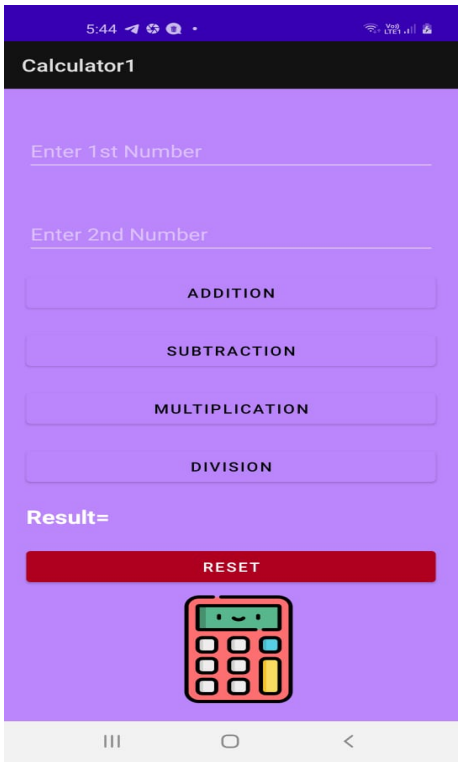

Figure 3.9: Calculator Design

3.0.3 Calendar UI development - XML Code

```
<?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:id="@+id/layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
```

```
    android:background="#E49E9E"
    tools:context=".MainActivity">

    <CalendarView
        android:id="@+id/cv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.239" />

    <TextView
        android:id="@+id/tv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="30sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/cv"
        app:layout_constraintVertical_bias="0.077" />

    <TextView
        android:id="@+id/tv2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Select Date"
        android:textSize="30sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.365"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/tv"
        app:layout_constraintVertical_bias="0.368" />

    <ImageView
        android:id="@+id/iv"
        android:layout_width="95dp"
        android:layout_height="130dp"
        android:layout_marginEnd="40dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.928"
        app:layout_constraintStart_toStartOf="@+id/tv2"
        app:layout_constraintTop_toBottomOf="@+id/tv"
        app:layout_constraintVertical_bias="0.284"
        app:srcCompat="@mipmap/calendar" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

3.0.4 Output Screen

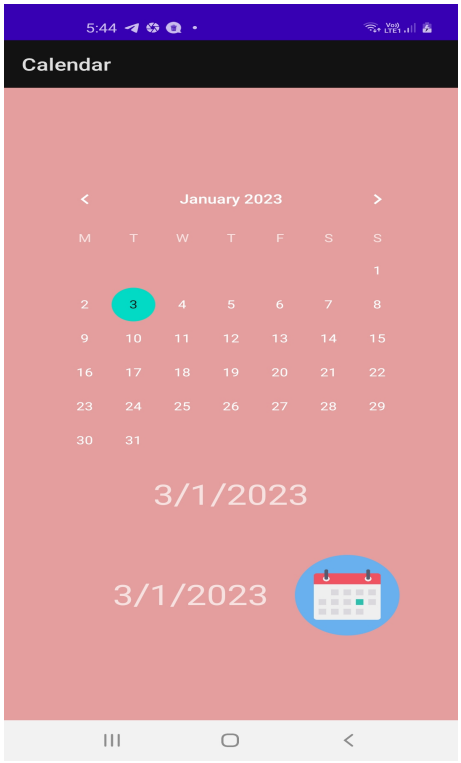

Figure 3.10: Calendar Design

4 Android User Interface Design: To implement different type of layouts like relative, grid, linear and table.

4.0.1 UI development - XML Code

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    android:background="#ECBACB"
    android:orientation="vertical"
    tools:context=".MainActivity3">

    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="#75ECE7E9">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical">

            <TextView
                android:id="@+id/textView"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginTop="20dp"
                android:layout_marginBottom="20dp"
                android:fontFamily="sans-serif-black"
                android:text="MUSIC BUZZ"
                android:textAlignment="center"
                android:textColor="?attr/colorError"
                android:textSize="34sp" />

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:layout_marginBottom="20dp"
                android:orientation="horizontal">

                <ImageView
                    android:id="@+id/imageView"
                    android:layout_width="169dp"
                    android:layout_height="151dp"
                    android:layout_marginLeft="10dp"
                    android:layout_marginRight="20dp"
                    android:layout_weight="1"
                    android:clickable="true"
                    android:contentDescription="@string/app_name"
                    android:onClick="drum"
                    app:srcCompat="@drawable/drum" />

                <ImageView
                    android:id="@+id/imageView2"
                    android:layout_width="169dp"
                    android:layout_height="151dp"
                    android:layout_marginRight="10dp"
                    android:layout_weight="1"
                    android:clickable="true"
                    android:contentDescription="@string/app_name"
                    android:onClick="piano"
                    app:srcCompat="@drawable/piano" />
```

```
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginBottom="20dp"
    android:orientation="horizontal">

    <ImageView
        android:id="@+id/imageView8"
        android:layout_width="169dp"
        android:layout_height="151dp"
        android:layout_marginLeft="10dp"
        android:layout_marginRight="20dp"
        android:layout_weight="1"
        android:clickable="true"
        android:contentDescription="@string/app_name"
        android:onClick="guitar"
        android:scaleType="centerCrop"
        app:srcCompat="@drawable/guitar" />

    <ImageView
        android:id="@+id/imageView9"
        android:layout_width="169dp"
        android:layout_height="match_parent"
        android:layout_marginRight="10dp"
        android:layout_weight="1"
        android:clickable="true"
        android:contentDescription="@string/app_name"
        android:onClick="flute"
        android:scaleType="centerCrop"
        app:srcCompat="@drawable/flute" />
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginBottom="20dp"
    android:orientation="horizontal">

    <ImageView
        android:id="@+id/imageView10"
        android:layout_width="169dp"
        android:layout_height="155dp"
        android:layout_marginLeft="10dp"
        android:layout_marginRight="20dp"
        android:layout_weight="1"
        android:clickable="true"
        android:onClick="casio"
        android:scaleType="centerCrop"
        app:srcCompat="@drawable/casio"
        tools:ignore="SpeakableTextPresentCheck" />

    <ImageView
        android:id="@+id/imageView11"
        android:layout_width="169dp"
        android:layout_height="match_parent"
        android:layout_marginRight="10dp"
        android:layout_weight="1"
        android:clickable="true"
        android:contentDescription="@string/app_name"
        android:onClick="trumpet"
        android:scaleType="centerCrop"
        app:srcCompat="@drawable/trumpet" />
</LinearLayout>

<EditText
    android:id="@+id/addnew1"
```

```
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginLeft="90dp"
            android:layout_marginRight="80dp"
            android:ems="10"
            android:fontFamily="sans-serif-black"
            android:hint="Add New Instrument!"
            android:inputType="textPersonName"
            android:minHeight="48dp"

            android:textAlignment="center"
            android:textColor="#2B2727"
            android:textColorHint="#100F0F" />

        <Button
            android:id="@+id/addnew"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginLeft="90dp"
            android:layout_marginRight="90dp"
            android:text="ADD" />



        <TextView
            android:id="@+id/textView3"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="#C87D7D"
            android:fontFamily="cursive"
            android:text="MUSIC GALLERY"
            android:textAlignment="center"
            android:textSize="34sp" />
    </LinearLayout>
</ScrollView>

</LinearLayout>
```

4.0.2 Java Code

```
package com.example.spotifyform;

import androidx.appcompat.app.AppCompatActivity;
import android.annotation.SuppressLint;
import android.content.Intent;
import android.media.MediaPlayer;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;


public class MainActivity3 extends AppCompatActivity {
    MediaPlayer m,g,c,p,f,t;
    public void drum(View view)
    {
        m.start();
        Toast.makeText(this,"Drum",Toast.LENGTH_SHORT).show();
        //return true;
    }
    public void guitar(View view)
    {
        g.start();
        Toast.makeText(this,"Guitar",Toast.LENGTH_SHORT).show();
```

```
    }
    public void casio(View view)
    {
        c.start();
        Toast.makeText(this, "Casio", Toast.LENGTH_SHORT).show();

    }
    public void piano(View view)
    {
        p.start();
        Toast.makeText(this, "Piano", Toast.LENGTH_SHORT).show();
    }
    public void flute(View view)
    {
        f.start();
        Toast.makeText(this, "Flute", Toast.LENGTH_SHORT).show();
    }
    public void trumpet(View view)
    {
        t.start();
        Toast.makeText(this, "Trumpet", Toast.LENGTH_SHORT).show();
    }
    @SuppressLint("WrongViewCast")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        Button addnew;

        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main3);
        addnew = findViewById(R.id.addnew);

        //code for media player
        m = MediaPlayer.create(this, R.raw.drumvoice);
        g = MediaPlayer.create(this, R.raw.guitarvoice);
        c = MediaPlayer.create(this, R.raw.casiovoice);
        p = MediaPlayer.create(this, R.raw.pianovoice);
        f = MediaPlayer.create(this, R.raw.flutevoice);
        t = MediaPlayer.create(this, R.raw.trumpetvoice);



        addnew.setOnClickListener(new View.OnClickListener() {
            @Override
            // public void onClick(View view) {
            // addnew.setText(null);
            // addnew.dispatchDisplayHint(View.VISIBLE);
            //addnew.setOnClickListener(new View.OnClickListener() {

            //}
            public void onClick(View view) {
                Intent i = new Intent(MainActivity3.this, MainActivity4.class);
                startActivity(i);
            }
        });


        // }
        //});

    }
}
```

4.0.3 Output Screen

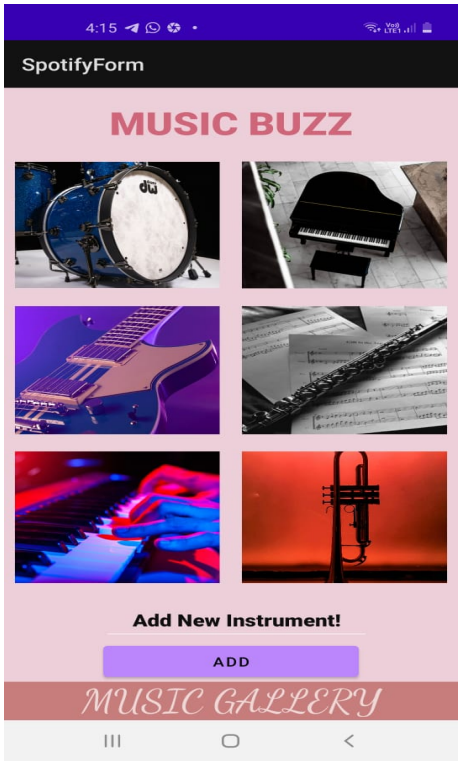

Figure 4.1: Multi Layout including grid

5 Apps Interactivity in Android: To incorporate element of interactivity using Android Fragment and Intent Class

5.0.1 UI development (XML files)

1. **activity_main4.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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="wrap_content"
    android:orientation="vertical"
    tools:context=".MainActivity4">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="#050505"
        android:orientation="horizontal">

        <Button
            android:id="@+id/btn1"
            android:layout_width="108dp"
            android:layout_height="wrap_content"
            android:layout_marginLeft="10dp"
            android:layout_weight="1"
            android:backgroundTint="#FBC8C8"
            android:text="Songs"
            android:textColor="#9C27B0" />

        <Button
            android:id="@+id/btn2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="10dp"
            android:layout_weight="1"
            android:backgroundTint="#FBC8C8"
            android:text="Playlist"
            android:textColor="#9C27B0" />

        <Button
            android:id="@+id/btn3"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="10dp"
            android:layout_marginRight="10dp"
            android:layout_weight="1"
            android:backgroundTint="#FBC8C8"
            android:text="Artists"
            android:textColor="#9C27B0" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="685dp"
        android:orientation="vertical">

        <androidx.fragment.app.FragmentContainerView
            android:id="@+id/fragmentContainerView"
            android:name="com.example.spotifyform.Fragment1"
            android:layout_width="match_parent"
            android:layout_height="match_parent" />
    </LinearLayout>
```

```
</LinearLayout>
```

2. fragment_1.xml:-

```
    <?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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="#ff1324"
    tools:context=".Fragment1">


    <TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_gravity="center"
        android:gravity="center"
        android:textColor="#ffff"
        android:text="Fragment 1" />

</FrameLayout>
```

3. fragment_2.xml:-

```
   <?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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="#3F51B5"
    tools:context=".Fragment2">

    <TextView
        android:id="@+id/textView2"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_gravity="center"
        android:gravity="center"
        android:text="Fragment 2"
        android:textColor="#ffff" />
</FrameLayout>
```

4. fragment_3.xml

```
   <?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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="#FFC107"
    tools:context=".Fragment3">

    <TextView
        android:id="@+id/textView4"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_gravity="center"
        android:gravity="center"
        android:text="Fragment 3"
        android:textColor="#ffff" />
```

```
</FrameLayout>
```

5.0.2 Java Code

1. MainActivity4.java

```
package com.example.spotifyform;

import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.FragmentManager;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class MainActivity4 extends AppCompatActivity {
    Button btn1, btn2, btn3;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main4);

        //Fragment Manager to manage code
        //FragmentManager fragmentManager = getSupportFragmentManager();

        btn1=findViewById(R.id.btn1);
        btn2=findViewById(R.id.btn2);
        btn3=findViewById(R.id.btn3);

        ///Code for Songs Button to Switch Fragments(Fragment1)
        btn1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

                FragmentManager fragmentManager = getSupportFragmentManager();
                fragmentManager.beginTransaction()
                        .replace(R.id.fragmentContainerView,Fragment1.class, null)
                        .setReorderingAllowed(true).addToBackStack("name")
                        .commit();
            }
        });
        ///Code for Playlist Button to Switch Fragments(Fragment2)
        btn2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                FragmentManager fragmentManager = getSupportFragmentManager();
                fragmentManager.beginTransaction()
                        .replace(R.id.fragmentContainerView,Fragment2.class, null)
                        .setReorderingAllowed(true).addToBackStack("name")
                        .commit();
            }
        });
        ///Code for Artists Button to Switch Fragments(Fragment3)
        btn3.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                FragmentManager fragmentManager = getSupportFragmentManager();
                fragmentManager.beginTransaction()
                        .replace(R.id.fragmentContainerView,Fragment3.class, null)
                        .setReorderingAllowed(true).addToBackStack("name")
                        .commit();
            }
        });
    }
}
```

2. Fragment1.java:

```
package com.example.spotifyform;

import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

import androidx.fragment.app.Fragment;


public class Fragment1 extends Fragment {



    public Fragment1() {
        // Required empty public constructor
    }


    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        return inflater.inflate(R.layout.fragment_1, container, false);
    }
}
```

3. Fragment2.java:

```
package com.example.spotifyform;

import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

import androidx.fragment.app.Fragment;


public class Fragment2 extends Fragment {


    public Fragment2() {
        // Required empty public constructor
    }




    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        return inflater.inflate(R.layout.fragment_2, container, false);
    }
}
```

4. Fragment3.java:

```
package com.example.spotifyform;

import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

import androidx.fragment.app.Fragment;


public class Fragment3 extends Fragment {


    public Fragment3() {
        // Required empty public constructor
    }


    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        return inflater.inflate(R.layout.fragment_3, container, false);
    }
}
```

5.0.3 Output Screens:-

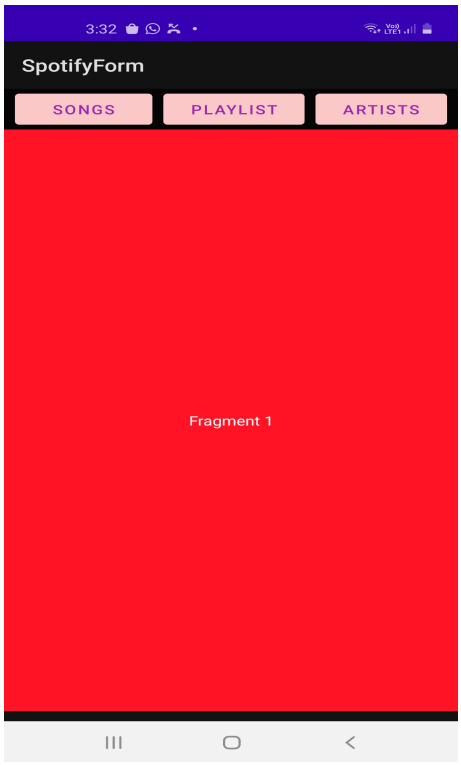

Figure 5.1: Fragment 1

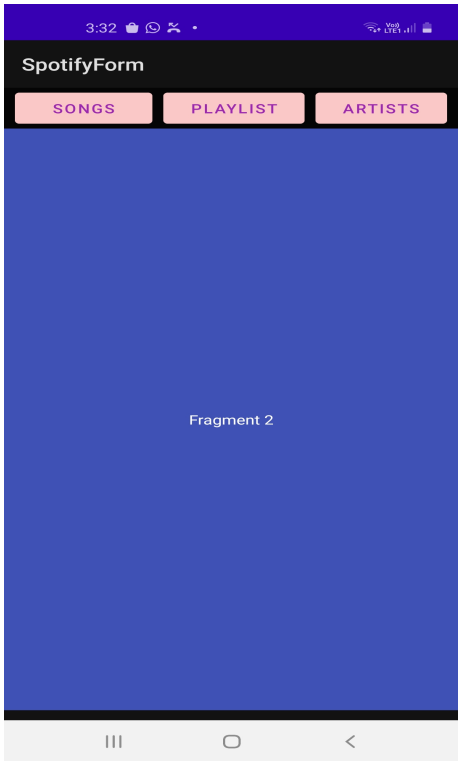

Figure 5.2: Fragment 2

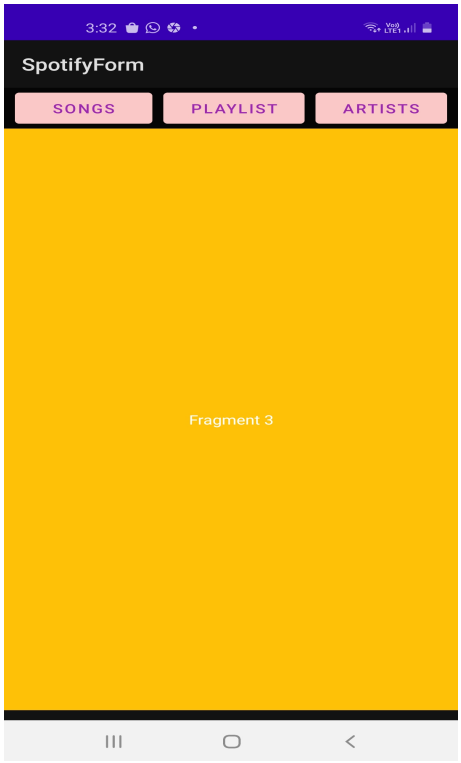

Figure 5.3: Fragment 3

5.0.4 Intent Class

Explicit Intent

1. **activity_main.xml:**

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

    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:paddingTop="15dp">

            <ImageView
                android:id="@+id/imageView2"
                android:layout_width="match_parent"
                android:layout_height="112dp"
                app:srcCompat="@drawable/spotify" />

            <View
                android:id="@+id/divider2"
                android:layout_width="match_parent"
                android:layout_height="1dp"
                android:layout_marginTop="20dp"
                android:background="?android:attr/listDivider" />

            <Button
                android:id="@+id/button2"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginStart="50dp"
                android:layout_marginTop="10dp"
                android:layout_marginEnd="40dp"
                android:text="SIGN UP WITH FACEBOOK "
                app:cornerRadius="23pt" />

            <RelativeLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_centerVertical="true">

                <TextView
                    android:id="@+id/tvText"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_centerInParent="true"
                    android:layout_marginLeft="10dp"
                    android:layout_marginRight="10dp"
                    android:text="or" />


                <View
                    android:layout_width="match_parent"
                    android:layout_height="1dp"
                    android:layout_centerVertical="true"
```

```
        android:layout_marginLeft="16dp"
        android:layout_toLeftOf="@id/tvText"
        android:background="?android:attr/listDivider" />

    />

    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:layout_centerVertical="true"
        android:layout_marginRight="16dp"
        android:layout_toRightOf="@id/tvText"
        android:background="?android:attr/listDivider" />
    />

</RelativeLayout>

<TextView
    android:id="@+id/textView3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="110dp"
    android:backgroundTint="#121111"
    android:text="Sign up with your email address"
    android:textColor="#070707"
    android:textStyle="bold" />

<EditText
    android:id="@+id/et1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="10dp"
    android:layout_marginRight="20dp"
    android:background="@drawable/image_border"
    android:ems="10"
    android:hint="Email"


    android:inputType="textEmailAddress"

    android:minHeight="48dp" />

<EditText
    android:id="@+id/et2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="10dp"
    android:layout_marginRight="20dp"
    android:background="@drawable/image_border"
    android:ems="10"
    android:hint="Confirm Email"
    android:inputType="textEmailAddress"

    android:minHeight="48dp" />

<EditText
    android:id="@+id/et3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"

    android:layout_marginTop="10dp"
    android:layout_marginRight="20dp"
    android:background="@drawable/image_border"
    android:ems="10"
```

```
    android:hint="Password"

    android:inputType="textPassword"
    android:minHeight="48dp"
    android:paddingTop="10dp"
    android:paddingBottom="10dp" />

<EditText
    android:id="@+id/et4"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="10dp"
    android:layout_marginRight="20dp"
    android:background="@drawable/image_border"
    android:ems="10"
    android:hint="What should we call you?"
    android:inputType="textPersonName"
    android:minHeight="48dp"
    android:paddingTop="10dp"
    android:paddingBottom="10dp" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginLeft="25dp"
    android:layout_marginTop="10dp"
    android:text="Date of Birth:" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="25dp"
    android:layout_marginEnd="25dp"

    android:orientation="horizontal">

    <Spinner
        android:id="@+id/spinner3"
        android:layout_width="272dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        tools:ignore="TouchTargetSizeCheck,SpeakableTextPresentCheck,SpeakableTextPresentCheck
            " />

    <Spinner
        android:id="@+id/spinner1"
        android:layout_width="100dp"
        android:layout_height="match_parent"
        tools:ignore="TouchTargetSizeCheck,SpeakableTextPresentCheck,SpeakableTextPresentCheck
            " />

    <EditText
        android:id="@+id/editTextTextPersonName"
        android:layout_width="wrap_content"
        android:layout_height="48dp"
        android:layout_weight="1"
        android:ems="10"
        android:hint="Year"
        android:inputType="textPersonName" />

</LinearLayout>

<TextView
    android:id="@+id/textView"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
```

```
    android:layout_marginLeft="25dp"
    android:layout_marginTop="10dp"
    android:text="Gender:" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:layout_weight="1"
    android:orientation="horizontal">


    <RadioGroup
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:orientation="horizontal">>

        <RadioButton
            android:id="@+id/radioButton5"
            android:layout_width="101dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Male" />

        <RadioButton
            android:id="@+id/radioButton6"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Female" />

        <RadioButton
            android:id="@+id/radioButton7"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Non-Binary" />
    </RadioGroup>

</LinearLayout>

<CheckBox
    android:id="@+id/checkBox"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:text="Share my registration data with Spotify's content providers for marketing
        purposes."
    android:textAlignment="center" />

<CheckBox
    android:id="@+id/checkBox2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="110dp"
    android:layout_marginRight="20dp"
    android:layout_marginBottom="10dp"
    android:text="I'm not a robot!"
    android:textSize="24sp" />

<TextView
    android:id="@+id/textView2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
```

```
            android:layout_marginBottom="10dp"
            android:backgroundTint="#121111"
            android:text="By clicking on Sign up, you agree to Spotify's Terms and Conditions of Use."
            android:textAlignment="center"
            android:textColor="#070707"
            android:textSize="14sp" />

        <TextView
            android:id="@+id/textView5"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginLeft="20dp"
            android:backgroundTint="#121111"
            android:text="To learn more about how Spotify collects, users, shares and protects your
                personal data please read Spotify's Privacy Policy."
            android:textAlignment="center"
            android:textColor="#070707"
            android:textSize="14sp" />

        <Button
            android:id="@+id/button"
            android:layout_width="334dp"
            android:layout_height="60dp"
            android:layout_marginStart="50dp"
            android:layout_marginTop="10dp"
            android:layout_marginEnd="40dp"
            android:backgroundTint="#4CAF50"
            android:text="SIGN UP"
            app:cornerRadius="23pt" />

        <Button
            android:id="@+id/button3"
            android:layout_width="334dp"
            android:layout_height="60dp"
            android:layout_marginStart="50dp"
            android:layout_marginTop="10dp"
            android:layout_marginEnd="40dp"
            android:backgroundTint="#3F51B5"
            android:text="LOG IN"
            app:cornerRadius="23pt" />


    </LinearLayout>
</ScrollView>

</LinearLayout>
```

5.0.5 Explicit Intent : Java Code

1. MainActivity.java

```
package com.example.spotifyform;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.Toast;
```

```
public class MainActivity extends AppCompatActivity implements AdapterView.OnItemSelectedListener {
    private Spinner spinner, spin;

    Button button, button2,button3;
    EditText et1, et2, et3, et4;
    database db;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        //database connectivity
        database g = new database(this); //g.getReadableDatabase();

        SQLiteDatabase db = g.getReadableDatabase();
        et1 = findViewById(R.id.et1);
        et2 = findViewById(R.id.et2);
        et3 = findViewById(R.id.et3);
        et4 = findViewById(R.id.et4);
        button = findViewById(R.id.button);
        button2 = findViewById(R.id.button2);
        button3 = findViewById(R.id.button3);
        button.setOnClickListener(new View.OnClickListener() {
            @Override

            public void onClick(View view) {
                String username1 = et1.getText().toString();
                String usernameconfirm1 = et2.getText().toString();
                String password1 = et3.getText().toString();
                String name1 = et4.getText().toString();
                if (username1.equals("") || usernameconfirm1.equals("") || password1.equals("") || name1.
                     equals("") ) {
                    Toast.makeText(MainActivity.this, "Enter all the fields", Toast.LENGTH_SHORT).show();
                } else {
                    Boolean i = g.insert_data(username1, usernameconfirm1, password1, name1);
                    if (i == true) {
                        Toast.makeText(MainActivity.this, "Successful", Toast.LENGTH_SHORT).show();
                    } else {
                        Toast.makeText(MainActivity.this, "Not Successful", Toast.LENGTH_SHORT).show();

                    }
                }
                et1.setText("");
                et2.setText("");
                et3.setText("");
                et4.setText("");
            }
        });
        //This is an example of Explicit Intent
        button3.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View view) {
                Intent i = new Intent(MainActivity.this, LoginformActivity.class);
                startActivity(i);
            }
        });

        spinner = findViewById(R.id.spinner3);
        spin = findViewById(R.id.spinner1);
        {
            ArrayAdapter<CharSequence> adapter = ArrayAdapter.createFromResource(this, R.array.spinner3,
                 android.R.layout.simple_spinner_item);
            adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);

            spinner.setAdapter(adapter);
```

```
            spinner.setOnItemSelectedListener(this);

        }
        {
            ArrayAdapter<CharSequence> adapter1 = ArrayAdapter.createFromResource(this, R.array.spinner1,
                  android.R.layout.simple_spinner_item);
            adapter1.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
            spin.setAdapter(adapter1);
            spin.setOnItemSelectedListener(this);

        }
    }

    @Override
    public void onItemSelected(AdapterView<?> adapterView, View view, int i, long l) {
        String choice = adapterView.getItemAtPosition(i).toString();
        Toast.makeText(getApplicationContext(), choice, Toast.LENGTH_SHORT).show();

        //Toast.makeText(getApplicationContext(),choice, Toast.LENGTH_LONG).show();
    }

    @Override
    public void onNothingSelected(AdapterView<?> adapterView) {

    }


}
```

5.0.6 Output Screens:-

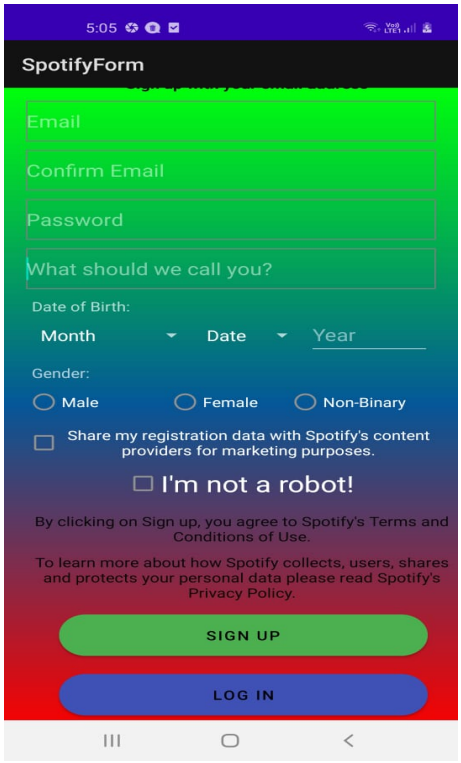

Figure 5.4: Click on LOG IN button

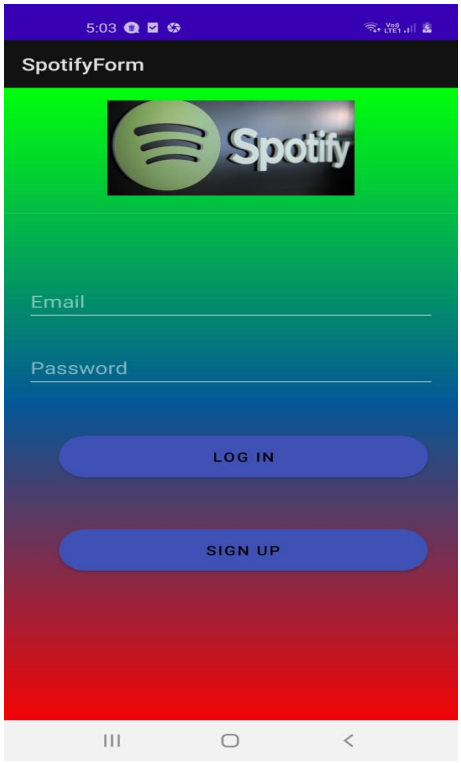

Figure 5.5: Explicit Intent Example

Implicit Intent

5.0.7 Implicit Intent : Java Code

(a) MainActivity.java

```
package com.example.spotifyform;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.database.sqlite.SQLiteDatabase;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity implements AdapterView.OnItemSelectedListener
    {
    private Spinner spinner, spin;

    Button button, button2,button3;
    EditText et1, et2, et3, et4;
    database db;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main);
        //database connectivity
        database g = new database(this); //g.getReadableDatabase();

        SQLiteDatabase db = g.getReadableDatabase();
        et1 = findViewById(R.id.et1);
        et2 = findViewById(R.id.et2);
        et3 = findViewById(R.id.et3);
        et4 = findViewById(R.id.et4);
        button = findViewById(R.id.button);
        button2 = findViewById(R.id.button2);
        button3 = findViewById(R.id.button3);
        button.setOnClickListener(new View.OnClickListener() {
            @Override

            public void onClick(View view) {
                String username1 = et1.getText().toString();
                String usernameconfirm1 = et2.getText().toString();
                String password1 = et3.getText().toString();
                String name1 = et4.getText().toString();
                if (username1.equals("") || usernameconfirm1.equals("") || password1.equals("") ||
                     name1.equals("") ) {
                    Toast.makeText(MainActivity.this, "Enter all the fields", Toast.LENGTH_SHORT).
                         show();
                } else {
                    Boolean i = g.insert_data(username1, usernameconfirm1, password1, name1);
                    if (i == true) {
                        Toast.makeText(MainActivity.this, "Successful", Toast.LENGTH_SHORT).show();
                    } else {
                        Toast.makeText(MainActivity.this, "Not Successful", Toast.LENGTH_SHORT).show
                             ();

                    }
                }
                et1.setText("");
                et2.setText("");
                et3.setText("");
                et4.setText("");
            }
        });
        //This is an example of Explicit Intent
        button3.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View view) {
                Intent i = new Intent(MainActivity.this, LoginformActivity.class);
                startActivity(i);
            }
        });
    //This is an example of Implicit Intent
        button2.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View view) {
                Intent i = new Intent(Intent.ACTION_VIEW, Uri.parse("https://www.facebook.com/"));
                startActivity(i);
            }
        });
        //
        spinner = findViewById(R.id.spinner3);
        spin = findViewById(R.id.spinner1);
        {
            ArrayAdapter<CharSequence> adapter = ArrayAdapter.createFromResource(this, R.array.
                 spinner3, android.R.layout.simple_spinner_item);
            adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);

            spinner.setAdapter(adapter);
```

```
            spinner.setOnItemSelectedListener(this);

        }
        {
            ArrayAdapter<CharSequence> adapter1 = ArrayAdapter.createFromResource(this, R.array.
                spinner1, android.R.layout.simple_spinner_item);
            adapter1.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
            spin.setAdapter(adapter1);
            spin.setOnItemSelectedListener(this);

        }
    }

    @Override
    public void onItemSelected(AdapterView<?> adapterView, View view, int i, long l) {
        String choice = adapterView.getItemAtPosition(i).toString();
        Toast.makeText(getApplicationContext(), choice, Toast.LENGTH_SHORT).show();

        //Toast.makeText(getApplicationContext(),choice, Toast.LENGTH_LONG).show();
    }

    @Override
    public void onNothingSelected(AdapterView<?> adapterView) {

    }


}
```

5.0.8 Output Screens:-

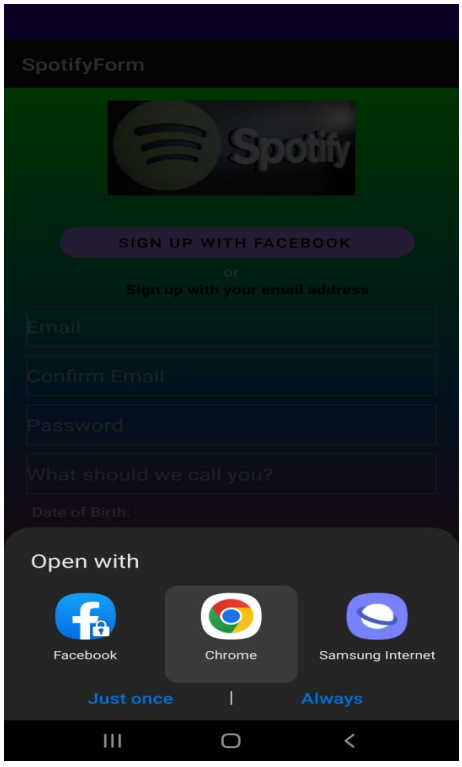

Figure 5.6: Click on SIGN UP WITH FACEBOOK button

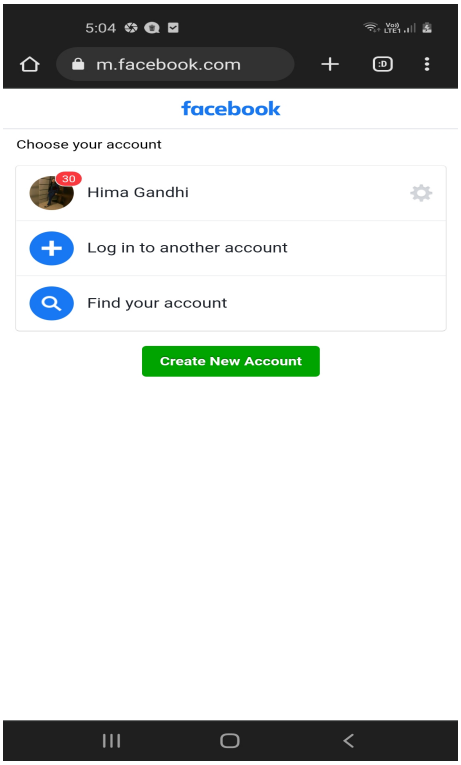

Figure 5.7: Implicit Intent Example

6 Persistent Data Storage: To perform database connectivity of android app using SQLite.

Login and Signup form is created using SQLite database.

6.0.1 UI Development (XML code)

i. SignUp form: activity_main.xml

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

    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:paddingTop="15dp">

            <ImageView
                android:id="@+id/imageView2"
                android:layout_width="match_parent"
                android:layout_height="112dp"
                app:srcCompat="@drawable/spotify" />

            <View
                android:id="@+id/divider2"
                android:layout_width="match_parent"
                android:layout_height="1dp"
                android:layout_marginTop="20dp"
                android:background="?android:attr/listDivider" />

            <Button
                android:id="@+id/button2"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginStart="50dp"
                android:layout_marginTop="10dp"
                android:layout_marginEnd="40dp"
                android:text="SIGN UP WITH FACEBOOK "
                app:cornerRadius="23pt" />

            <RelativeLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_centerVertical="true">

                <TextView
                    android:id="@+id/tvText"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_centerInParent="true"
                    android:layout_marginLeft="10dp"
                    android:layout_marginRight="10dp"
                    android:text="or" />
```

```
    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:layout_centerVertical="true"
        android:layout_marginLeft="16dp"
        android:layout_toLeftOf="@id/tvText"
        android:background="?android:attr/listDivider" />

    />

    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:layout_centerVertical="true"
        android:layout_marginRight="16dp"
        android:layout_toRightOf="@id/tvText"
        android:background="?android:attr/listDivider" />
    />

</RelativeLayout>

<TextView
    android:id="@+id/textView3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="110dp"
    android:backgroundTint="#121111"
    android:text="Sign up with your email address"
    android:textColor="#070707"
    android:textStyle="bold" />

<EditText
    android:id="@+id/et1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="10dp"
    android:layout_marginRight="20dp"
    android:background="@drawable/image_border"
    android:ems="10"
    android:hint="Email"


    android:inputType="textEmailAddress"

    android:minHeight="48dp" />

<EditText
    android:id="@+id/et2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="10dp"
    android:layout_marginRight="20dp"
    android:background="@drawable/image_border"
    android:ems="10"
    android:hint="Confirm Email"
    android:inputType="textEmailAddress"

    android:minHeight="48dp" />

<EditText
    android:id="@+id/et3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
```

```
        android:layout_marginTop="10dp"
        android:layout_marginRight="20dp"
        android:background="@drawable/image_border"
        android:ems="10"

        android:hint="Password"

        android:inputType="textPassword"
        android:minHeight="48dp"
        android:paddingTop="10dp"
        android:paddingBottom="10dp" />

    <EditText
        android:id="@+id/et4"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginTop="10dp"
        android:layout_marginRight="20dp"
        android:background="@drawable/image_border"
        android:ems="10"
        android:hint="What should we call you?"
        android:inputType="textPersonName"
        android:minHeight="48dp"
        android:paddingTop="10dp"
        android:paddingBottom="10dp" />

    <TextView
        android:id="@+id/textView4"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginLeft="25dp"
        android:layout_marginTop="10dp"
        android:text="Date of Birth:" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="25dp"
        android:layout_marginEnd="25dp"

        android:orientation="horizontal">

        <Spinner
            android:id="@+id/spinner3"
            android:layout_width="272dp"
            android:layout_height="match_parent"
            android:layout_weight="1"
            tools:ignore="TouchTargetSizeCheck,SpeakableTextPresentCheck,
                SpeakableTextPresentCheck" />

        <Spinner
            android:id="@+id/spinner1"
            android:layout_width="100dp"
            android:layout_height="match_parent"
            tools:ignore="TouchTargetSizeCheck,SpeakableTextPresentCheck,
                SpeakableTextPresentCheck" />

        <EditText
            android:id="@+id/editTextTextPersonName"
            android:layout_width="wrap_content"
            android:layout_height="48dp"
            android:layout_weight="1"
            android:ems="10"
            android:hint="Year"
            android:inputType="textPersonName" />
```

```
</LinearLayout>

<TextView
    android:id="@+id/textView"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginLeft="25dp"
    android:layout_marginTop="10dp"
    android:text="Gender:" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:layout_weight="1"
    android:orientation="horizontal">


    <RadioGroup
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:orientation="horizontal">>

        <RadioButton
            android:id="@+id/radioButton5"
            android:layout_width="101dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Male" />

        <RadioButton
            android:id="@+id/radioButton6"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Female" />

        <RadioButton
            android:id="@+id/radioButton7"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Non-Binary" />
    </RadioGroup>

</LinearLayout>

<CheckBox
    android:id="@+id/checkBox"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:text="Share my registration data with Spotify's content providers for
         marketing purposes."
    android:textAlignment="center" />

<CheckBox
    android:id="@+id/checkBox2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="110dp"
    android:layout_marginRight="20dp"
    android:layout_marginBottom="10dp"
    android:text="I'm not a robot!"
    android:textSize="24sp" />
```

```
            <TextView
                android:id="@+id/textView2"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginLeft="20dp"
                android:layout_marginBottom="10dp"
                android:backgroundTint="#121111"
                android:text="By clicking on Sign up, you agree to Spotify's Terms and
                    Conditions of Use."
                android:textAlignment="center"
                android:textColor="#070707"
                android:textSize="14sp" />

            <TextView
                android:id="@+id/textView5"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginLeft="20dp"
                android:backgroundTint="#121111"
                android:text="To learn more about how Spotify collects, users, shares and
                    protects your personal data please read Spotify's Privacy Policy."
                android:textAlignment="center"
                android:textColor="#070707"
                android:textSize="14sp" />

            <Button
                android:id="@+id/button"
                android:layout_width="334dp"
                android:layout_height="60dp"
                android:layout_marginStart="50dp"
                android:layout_marginTop="10dp"
                android:layout_marginEnd="40dp"
                android:backgroundTint="#4CAF50"
                android:text="SIGN UP"
                app:cornerRadius="23pt" />

            <Button
                android:id="@+id/button3"
                android:layout_width="334dp"
                android:layout_height="60dp"
                android:layout_marginStart="50dp"
                android:layout_marginTop="10dp"
                android:layout_marginEnd="40dp"
                android:backgroundTint="#3F51B5"
                android:text="LOG IN"
                app:cornerRadius="23pt" />


        </LinearLayout>
    </ScrollView>

</LinearLayout>
```

ii. Login Form: loginform.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"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="#16181C">

<ScrollView
    android:layout_width="match_parent"
```

```
android:layout_height="match_parent"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent">

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical" >

    <ImageView
        android:id="@+id/imageView3"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        app:srcCompat="@drawable/logo2" />

    <TextView
        android:id="@+id/textView2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="25dp"
        android:fontFamily="@font/squre"
        android:text="Login Into \nYour Wallet"
        android:textColor="#FFFFFF"
        android:textSize="35sp" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="32dp"
        android:layout_marginEnd="32dp"
        android:layout_marginBottom="32dp"
        android:background="@drawable/input_bg"
        android:orientation="vertical">

        <EditText
            android:id="@+id/login_email"
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:layout_marginStart="24dp"
            android:layout_marginTop="24dp"
            android:layout_marginEnd="24dp"
            android:background="@drawable/input_fields"
            android:drawableLeft="@drawable/email"
            android:drawablePadding="16dp"
            android:ems="10"
            android:hint="Email"
            android:inputType="textEmailAddress"
            android:paddingLeft="16dp"
            android:textColor="#FFFFFF"
            android:textColorHint="#FFFFFF" />

        <EditText
            android:id="@+id/login_pass"
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:layout_marginStart="24dp"
            android:layout_marginTop="24dp"
            android:layout_marginEnd="24dp"
            android:background="@drawable/input_fields"
            android:drawableLeft="@drawable/password"
            android:drawablePadding="16dp"
            android:ems="10"
            android:hint="Password"
            android:inputType="textPassword"
            android:paddingLeft="16dp"
            android:textColor="#FFFFFF"
```

```
            android:textColorHint="#FFFFFF" />

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:orientation="horizontal">

            <ImageView
                android:id="@+id/imageView5"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_margin="25dp"
                app:srcCompat="@drawable/facebook" />

            <ImageView
                android:id="@+id/imageView4"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_margin="25dp"
                app:srcCompat="@drawable/google" />

        </LinearLayout>

        <TextView
            android:id="@+id/register_link"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_marginBottom="20dp"
            android:layout_weight="1"
            android:text="Are you a new User? Register"
            android:textAlignment="center"
            android:textColor="#9D94FD"
            android:textSize="18sp"
            android:textStyle="bold" />

        <Button
            android:id="@+id/login_btn"
            style="@android:style/Widget.Button"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_marginStart="20dp"

            android:layout_marginEnd="20dp"
            android:layout_marginBottom="20dp"
            android:background="@drawable/bg_btn"
            android:text="Login"
            android:textColor="#FFFFFF"
            android:textSize="24sp"
            android:textStyle="bold" />

        <TextView
            android:id="@+id/textView4"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginBottom="32dp"
            android:text="Forgot Password? Recover it."
            android:textAlignment="center"
            android:textColor="#FFFFFF" />

    </LinearLayout>

</LinearLayout>
</ScrollView>
</androidx.constraintlayout.widget.ConstraintLayout>
```

6.0.2 Java Code

i. MainActivity.java

```
package com.example.spotifyform;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity implements AdapterView.
     OnItemSelectedListener {
    private Spinner spinner, spin;

    Button button, button3;
    EditText et1, et2, et3, et4;
    database db;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        //database connectivity
        database g = new database(this); //g.getReadableDatabase();

        SQLiteDatabase db = g.getReadableDatabase();
        et1 = findViewById(R.id.et1);
        et2 = findViewById(R.id.et2);
        et3 = findViewById(R.id.et3);
        et4 = findViewById(R.id.et4);
        button = findViewById(R.id.button);
        button3 = findViewById(R.id.button3);
        button.setOnClickListener(new View.OnClickListener() {
            @Override

            public void onClick(View view) {
                String username1 = et1.getText().toString();
                String usernameconfirm1 = et2.getText().toString();
                String password1 = et3.getText().toString();
                String name1 = et4.getText().toString();
                if (username1.equals("") || usernameconfirm1.equals("") || password1.equals("")
                      || name1.equals("") ) {
                    Toast.makeText(MainActivity.this, "Enter all the fields", Toast.
                         LENGTH_SHORT).show();
                } else {
                    Boolean i = g.insert_data(username1, usernameconfirm1, password1, name1);
                    if (i == true) {
                        Toast.makeText(MainActivity.this, "Successful", Toast.LENGTH_SHORT).
                             show();
                    } else {
                        Toast.makeText(MainActivity.this, "Not Successful", Toast.LENGTH_SHORT)
                             .show();

                    }
                }
                et1.setText("");
                et2.setText("");
                et3.setText("");
                et4.setText("");
```

```
            }
        });
        button3.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View view) {
                Intent i = new Intent(MainActivity.this, LoginformActivity.class);
                startActivity(i);
            }
        });
        //
        spinner = findViewById(R.id.spinner3);
        spin = findViewById(R.id.spinner1);
        {
            ArrayAdapter<CharSequence> adapter = ArrayAdapter.createFromResource(this, R.array
                 .spinner3, android.R.layout.simple_spinner_item);
            adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);

            spinner.setAdapter(adapter);

            spinner.setOnItemSelectedListener(this);

        }
        {
            ArrayAdapter<CharSequence> adapter1 = ArrayAdapter.createFromResource(this, R.
                 array.spinner1, android.R.layout.simple_spinner_item);
            adapter1.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
            spin.setAdapter(adapter1);
            spin.setOnItemSelectedListener(this);

        }
    }

    @Override
    public void onItemSelected(AdapterView<?> adapterView, View view, int i, long l) {
        String choice = adapterView.getItemAtPosition(i).toString();
        Toast.makeText(getApplicationContext(), choice, Toast.LENGTH_SHORT).show();

        //Toast.makeText(getApplicationContext(),choice, Toast.LENGTH_LONG).show();
    }

    @Override
    public void onNothingSelected(AdapterView<?> adapterView) {

    }


}
```

ii. LoginformActivity.java

```
package com.example.spotifyform;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class LoginformActivity extends AppCompatActivity {
    Button loginbtn, button5;
    EditText etemail,etpass;
```

```
    database db;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.loginform);
        button5 = findViewById(R.id.button5);
        db = new database(this);
        //database g = new database(this); //g.getReadableDatabase();

       // SQLiteDatabase db = g.getReadableDatabase();
        etemail = findViewById(R.id.etemail);
        etpass = findViewById(R.id.etpass);
        loginbtn = findViewById(R.id.loginbtn);


        loginbtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String username = etemail.getText().toString();
                String password = etpass.getText().toString();

                if(username.equals("") || password.equals(""))
                {
                    Toast.makeText(LoginformActivity.this, "Please enter the Credentials",
                         Toast.LENGTH_SHORT).show();
                }
                else
                {
                    Boolean result = db.checkusernamePassword(username,password);
                    if(result == true){
                    Intent i = new Intent(getApplicationContext(), MainActivity3.class);
                    startActivity(i);
                    }
                    else
                    {
                        Toast.makeText(LoginformActivity.this, "Invalid Credentials", Toast.
                            LENGTH_SHORT).show();
                    }
                }
            }
        });
        button5.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View view) {
                Intent i = new Intent(LoginformActivity.this, MainActivity.class);
                startActivity(i);
            }
        });

    }

}
```

iii. database.java

```
package com.example.spotifyform;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

import androidx.annotation.Nullable;
```

```
public class database extends SQLiteOpenHelper {
    private static final String dbname="signup.db";
    public database(@Nullable Context context) {
        super(context,dbname, null, 1);
    }

    @Override
    public void onCreate(SQLiteDatabase sqLiteDatabase) {
    String q = "create table users(_id integer primary key autoincrement, username text,
         usernameconfirm text, password text, name text)";
        sqLiteDatabase.execSQL(q);
    }

    @Override
    public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {
        sqLiteDatabase.execSQL("drop table if exists users");
        onCreate(sqLiteDatabase);
    }

    public boolean insert_data(String username, String usernameconfirm, String password,
         String name)
    {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues c = new ContentValues(); //for writing content
        c.put("username", username);
        c.put("usernameconfirm", usernameconfirm);
        c.put("password", password);
        c.put("name", name);
        long r = db.insert("users", null, c);
        if(r== -1)return false;
        else
            return true;
    }

    public Boolean checkusernamePassword(String username, String password) {
        SQLiteDatabase db = this.getWritableDatabase();
        Cursor cursor= db.rawQuery("select * from users where username = ? and password = ?",
             new String[] {username,password});
        if(cursor.getCount()>0)
        {
            return true;
        }
        else
        {
            return false;
        }
    }
}
```

6.0.3 Output Screens:

i. Register Form:-

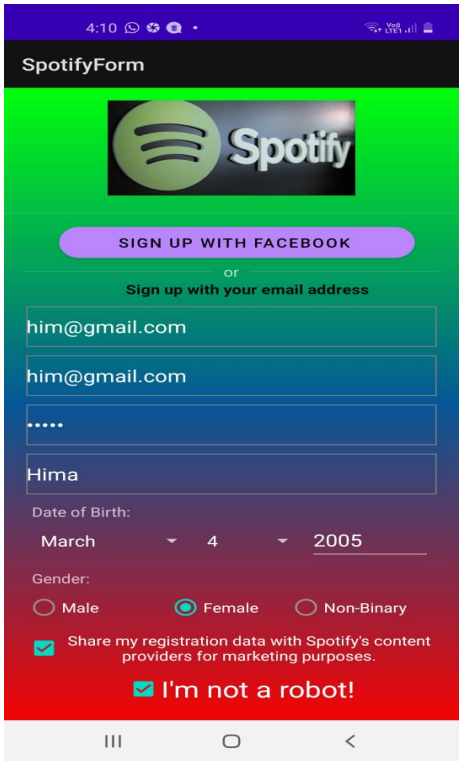

Figure 6.1: Registration form

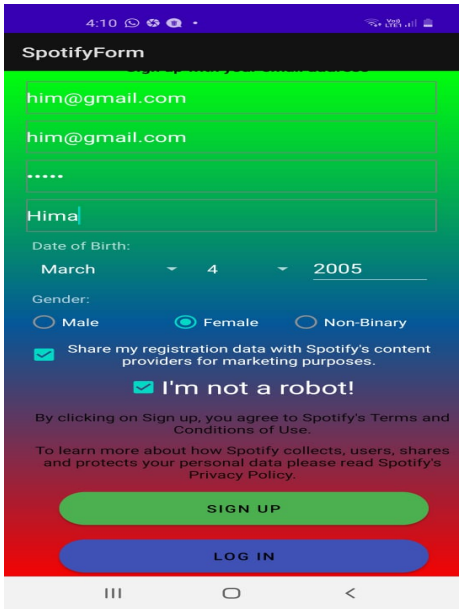

Figure 6.2: Registration form

ii. Login Form:-

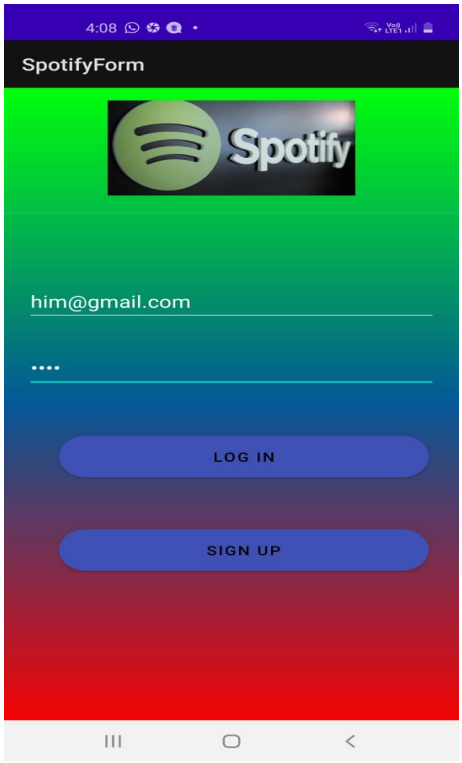

Figure 6.3: Login form

iii. Activity after login-

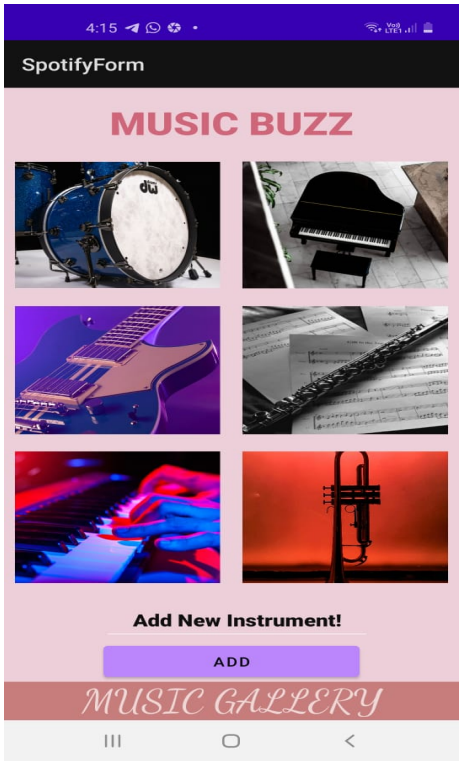

Figure 6.4: Main Activity

iv. Database -

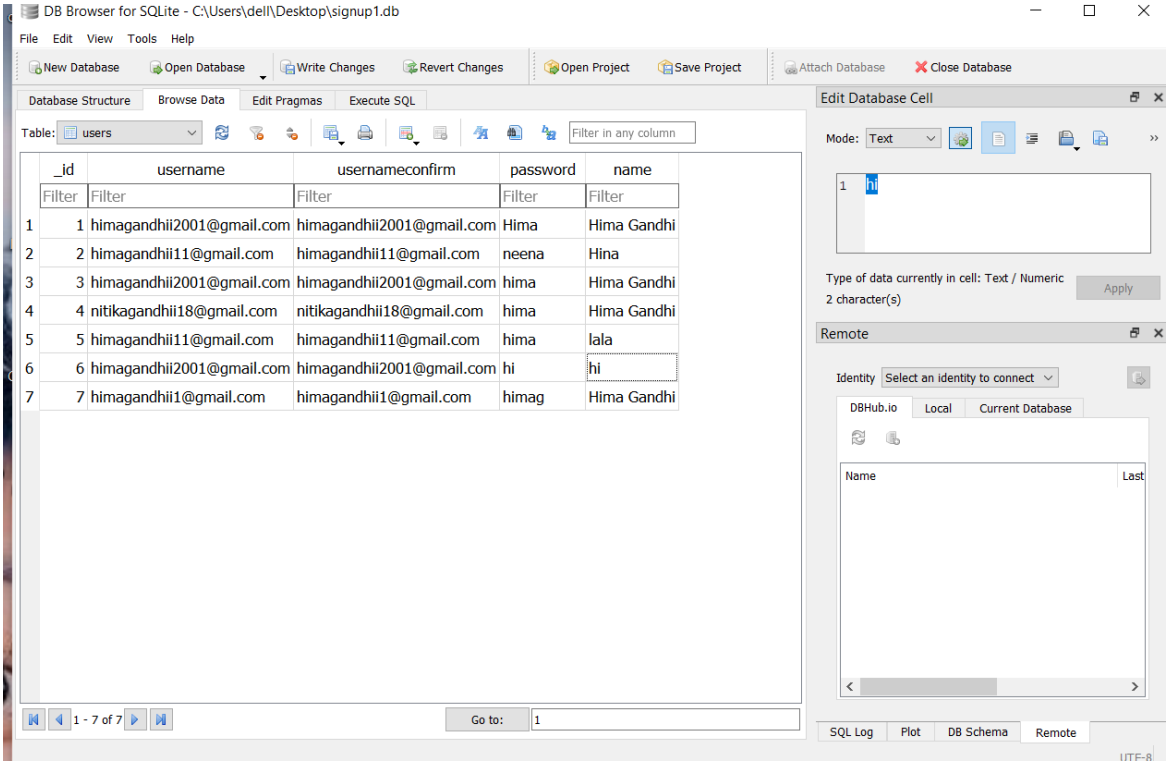

Figure 6.5: Database

7 Android Services and Threads: To implement the concept of multithreading using Android Service class

7.0.1 XML Code

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

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="314dp"
        android:layout_height="430dp"
        android:layout_gravity="center"
        android:layout_margin="50dp" />

        <Button
            android:id="@+id/button"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_margin="10dp"
            android:layout_gravity="center"
            android:background = "@drawable/btn"
            android:text="Load Image 1" />

        <Button
            android:id="@+id/button2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_margin="10dp"
            android:background = "@drawable/btn"
            android:layout_gravity="center"
            android:text="Load image 2" />

    </LinearLayout>
```

7.0.2 Java Code

```
package com.example.multithread;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
public class MainActivity extends AppCompatActivity
{
    ImageView img;
    Button bt1,bt2;
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
```

```
        bt1 = (Button)findViewById(R.id.button);
        bt2= (Button) findViewById(R.id.button2);
        img = (ImageView)findViewById(R.id.imageView);

        bt1.setOnClickListener(new View.OnClickListener()
        {
            @Override
            public void onClick(View v)
            {
                new Thread(new Runnable()
                {
                    @Override
                    public void run()
                    {
                        img.post(new Runnable()
                        {
                            @Override
                            public void run()
                            {
                                img.setImageResource(R.drawable.pic1);
                            }
                        });
                    }
                }).start();
            }
        });

        bt2.setOnClickListener(new View.OnClickListener()
        {
            @Override
            public void onClick(View v)
            {
                new Thread(new Runnable()
                {
                    @Override
                    public void run()
                    {
                        img.post(new Runnable()
                        {
                            @Override
                            public void run()
                            {
                                img.setImageResource(R.drawable.pic2);
                            }
                        });
                    }
                }).start();
            }
        });
    }
}
```

7.0.3 Output Screens:

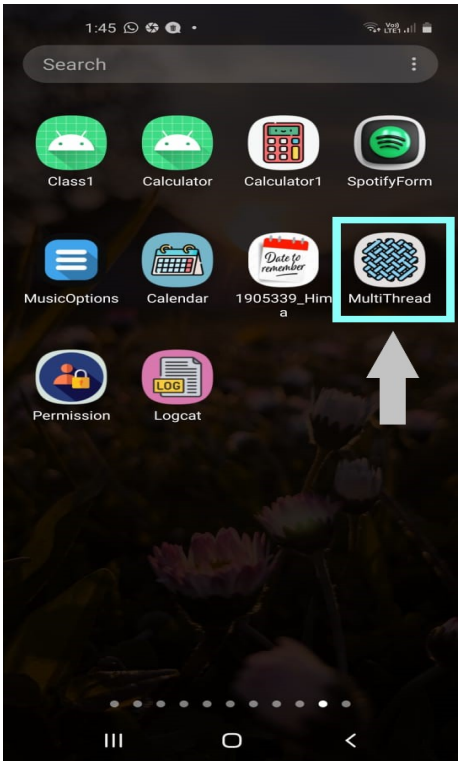

Figure 7.1: App Installed

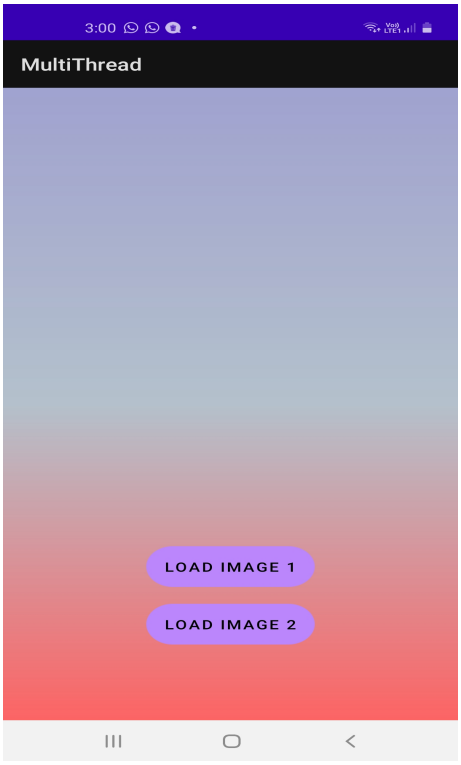

Figure 7.2: App screen

Figure 7.3: On clicking Load Image 1

Figure 7.4: On clicking Load Image 2

8 Android Security and Debugging: To implement concept of permission and perform request for permission to access different hardware components of mobile

i. Declare the permission in the Android Manifest file: In Android, permissions are declared in the AndroidManifest.xml file using the uses-permission tag.

```
<!--Declaring the required permissions-->
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.CAMERA" />
```

ii. Modify activity_main.xml file to Add two buttons to request permission on button click: Permission will be checked and requested on button click. Open the activity_main.xml file and add two buttons to it.

```
<!--Button to request storage permission-->
<?xml version="1.0" encoding="utf-8"?>
    <RelativeLayout
        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"
        android:background = "@drawable/gradient_drawable"
        tools:context=".MainActivity">

        <!-- To show toolbar-->
    <androidx.appcompat.widget.Toolbar
        android:id="@+id/toolbar"
        android:layout_width="match_parent"
        android:layout_height="?android:attr/actionBarSize"
        android:background="#9C27B0"
        app:title="Permission Example"
        app:titleTextColor="@android:color/white" />

        <!--Button to request storage permission-->

    <!--Button to request camera permission-->

    <Button

        android:id="@+id/storage"
        android:layout_width="122dp"
        android:layout_height="68dp"
        android:layout_below="@id/toolbar"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="207dp"
        android:backgroundTint="#E91E63"
        android:padding="8dp"
        android:text="Storage"
        app:cornerRadius="30pt" />

    <Button
        android:id="@+id/camera"
        android:layout_width="121dp"
        android:layout_height="67dp"
        android:layout_below="@id/storage"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="100dp"
        android:backgroundTint="#E91E63"
        android:padding="8dp"
        android:text="Camera"
```

```
        app:cornerRadius="30pt" />

</RelativeLayout>
```

iii. Check whether permission is already granted or not. If permission isn't already granted, request the user for the permission: In order to use any service or feature, the permissions are required. Hence we have to ensure that the permissions are given for that. If not, then the permissions are requested.

Check for permissions: Beginning with Android 6.0 (API level 23), the user has the right to revoke permissions from any app at any time, even if the app targets a lower API level. So to use the service, the app needs to check for permissions every time.

Request Permissions: When PERMISSION_DENIED is returned from the checkSelfPermission() method in the above syntax, we need to prompt the user for that permission. Android provides several methods that can be used to request permission, such as requestPermissions().

```
// Function to check and request permission
public void checkPermission(String permission, int requestCode)
{
    // Checking if permission is not granted
    if (ContextCompat.checkSelfPermission(MainActivity.this, permission) == PackageManager.
         PERMISSION_DENIED) {
        ActivityCompat.requestPermissions(MainActivity.this, new String[] { permission },
             requestCode);
    }
    else {
        Toast.makeText(MainActivity.this, "Permission already granted", Toast.LENGTH_SHORT).
             show();
    }
}
```

iv. Override onRequestPermissionsResult() method: onRequestPermissionsResult() is called when user grant or decline the permission. RequestCode is one of the parameters of this function which is used to check user action for the corresponding requests. Here a toast message is shown indicating the permission and user action.

```
@Override
public void onRequestPermissionsResult(int requestCode,
                                       @NonNull String[] permissions,
                                       @NonNull int[] grantResults)
{
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);

    if (requestCode == CAMERA_PERMISSION_CODE) {

        // Checking whether user granted the permission or not.
        if (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION_GRANTED) {

            // Showing the toast message
            Toast.makeText(MainActivity.this, "Camera Permission Granted", Toast.LENGTH_SHORT)
                 .show();
        }
        else {
            Toast.makeText(MainActivity.this, "Camera Permission Denied", Toast.LENGTH_SHORT).
                 show();
```

```
        }
    }
    else if (requestCode == STORAGE_PERMISSION_CODE) {
        if (grantResults.length > 0
            && grantResults[0] == PackageManager.PERMISSION_GRANTED) {
            Toast.makeText(MainActivity.this, "Storage Permission Granted", Toast.LENGTH_SHORT
                ).show();
        }
        else {
            Toast.makeText(MainActivity.this, "Storage Permission Denied", Toast.LENGTH_SHORT)
                .show();
        }
    }
}
```

v. MainActivity.java code:

```
package com.example.permission;

import android.Manifest;
import android.content.pm.PackageManager;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;

public class MainActivity extends AppCompatActivity {

    // Defining Buttons
    private Button storage, camera;

    // Defining Permission codes.
    // We can give any value
    // but unique for each permission.
    private static final int CAMERA_PERMISSION_CODE = 100;
    private static final int STORAGE_PERMISSION_CODE = 101;

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

        storage = findViewById(R.id.storage);
        camera = findViewById(R.id.camera);

        // Set Buttons on Click Listeners
        storage.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v)
            {
                checkPermission(Manifest.permission.WRITE_EXTERNAL_STORAGE,
                    STORAGE_PERMISSION_CODE);
            }
        });

        camera.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v)
            {
                checkPermission(Manifest.permission.CAMERA, CAMERA_PERMISSION_CODE);
```

```
            }
        });
    }

    // Function to check and request permission.
    public void checkPermission(String permission, int requestCode)
    {
        if (ContextCompat.checkSelfPermission(MainActivity.this, permission) == PackageManager
             .PERMISSION_DENIED) {

            // Requesting the permission
            ActivityCompat.requestPermissions(MainActivity.this, new String[] { permission },
                 requestCode);
        }
        else {
            Toast.makeText(MainActivity.this, "Permission already granted", Toast.LENGTH_SHORT
                 ).show();
        }
    }

    // This function is called when the user accepts or decline the permission.
    // Request Code is used to check which permission called this function.
    // This request code is provided when the user is prompt for permission.

    @Override
    public void onRequestPermissionsResult(int requestCode,
                                           @NonNull String[] permissions,
                                           @NonNull int[] grantResults)
    {
        super.onRequestPermissionsResult(requestCode,
                permissions,
                grantResults);

        if (requestCode == CAMERA_PERMISSION_CODE) {
            if (grantResults.length > 0 && grantResults[0] == PackageManager.
                 PERMISSION_GRANTED) {
                Toast.makeText(MainActivity.this, "Camera Permission Granted", Toast.
                     LENGTH_SHORT) .show();
            }
            else {
                Toast.makeText(MainActivity.this, "Camera Permission Denied", Toast.
                     LENGTH_SHORT) .show();
            }
        }
        else if (requestCode == STORAGE_PERMISSION_CODE) {
            if (grantResults.length > 0
                    && grantResults[0] == PackageManager.PERMISSION_GRANTED) {
                Toast.makeText(MainActivity.this, "Storage Permission Granted", Toast.
                     LENGTH_SHORT).show();
            } else {
                Toast.makeText(MainActivity.this, "Storage Permission Denied", Toast.
                     LENGTH_SHORT).show();
            }
        }
    }
}
```

8.0.1 Output Screens

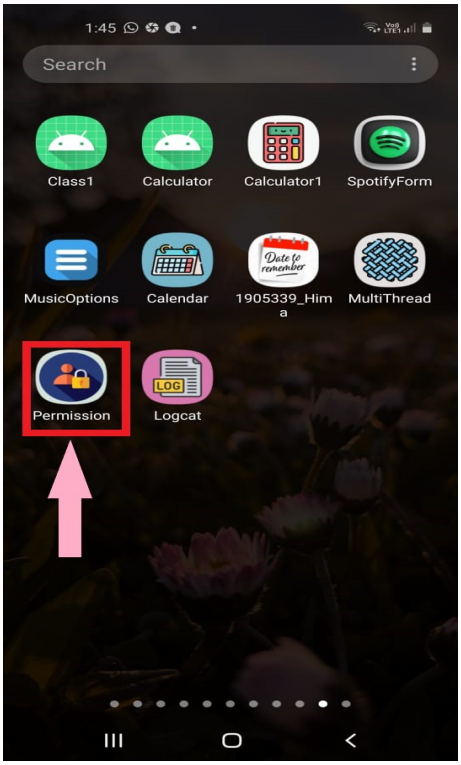

Figure 8.1: App installed

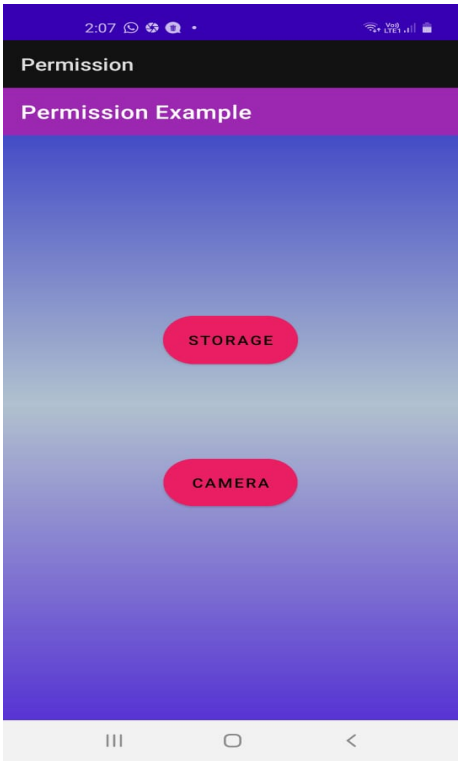

Figure 8.2: Main Activity

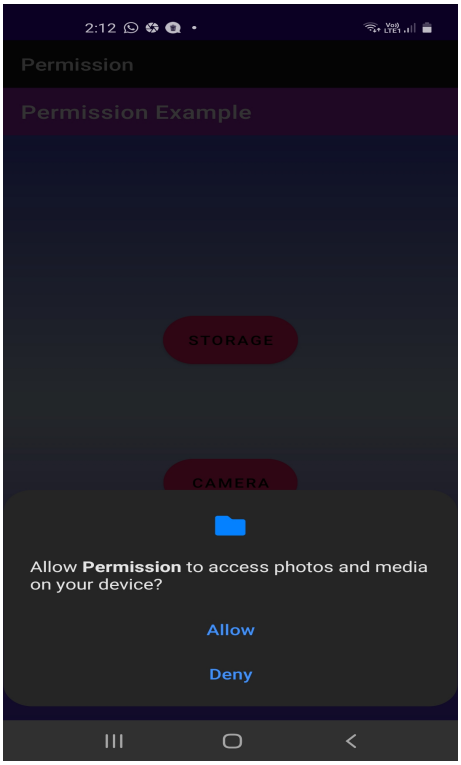

Figure 8.3: Permission dialog box

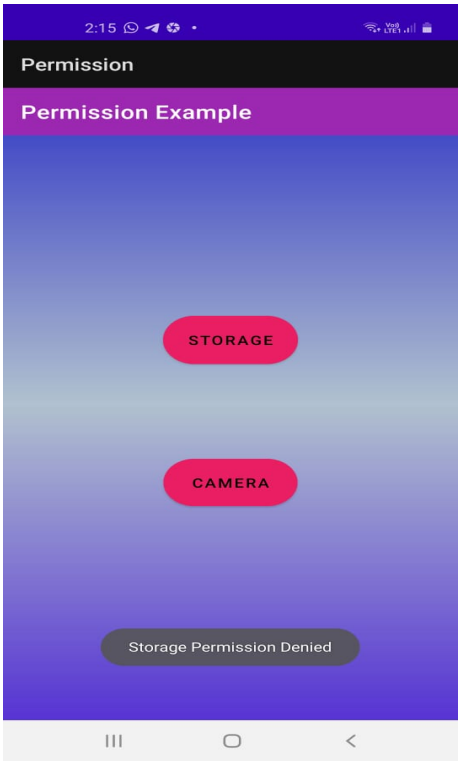

Figure 8.4: Toast message after permission denied

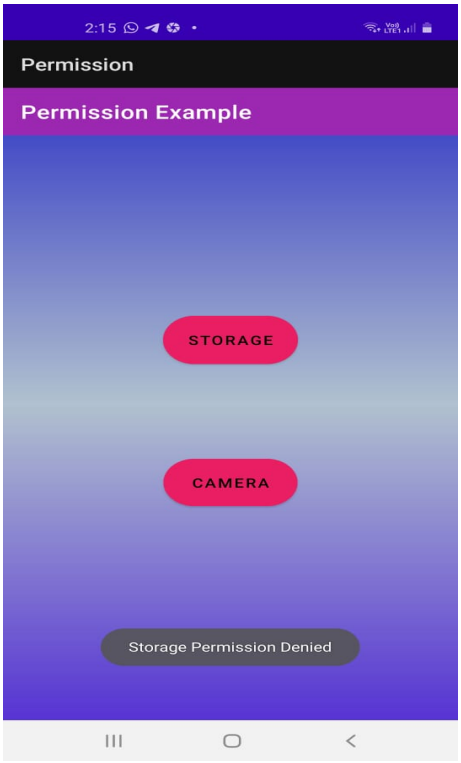

Figure 8.5: Toast message after storage permission granted

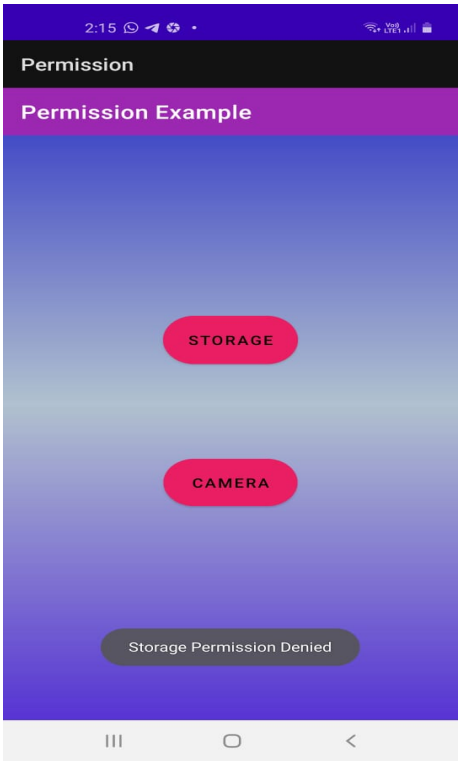

Figure 8.6: Toast message after camera permission granted

9 Android Security and Debugging: To perform debugging and testing of android app using tools like Logcat, Android debug bridge, DDMS

Logcat is an important tool in Android studio. Logcat helps developers to debug the code when an application is not running or has crashed. Steps to run Logact in Android Studio:-

i. Create new project in android studio and select empty activity.

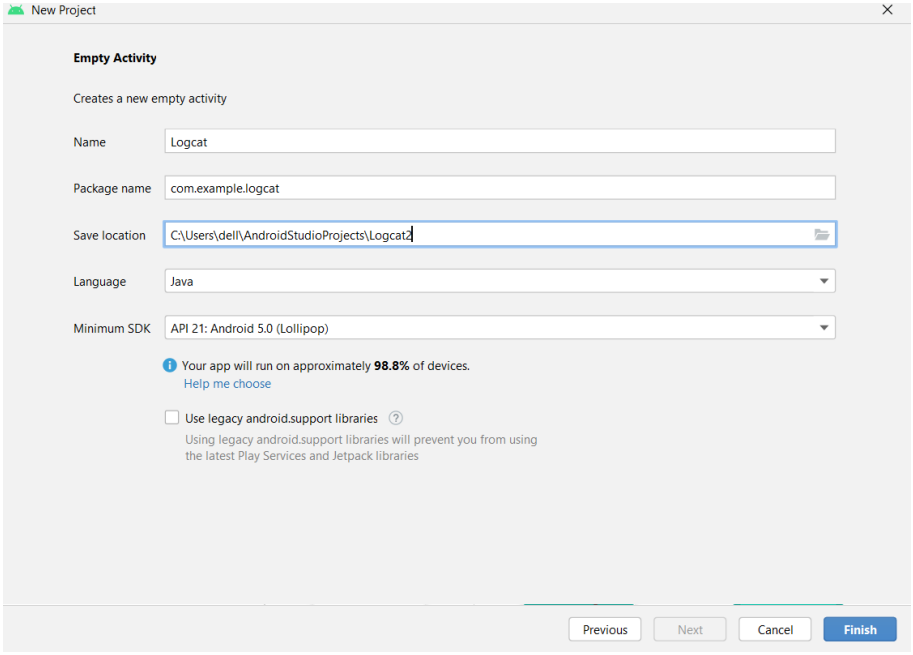

Figure 9.1: Create New Project

ii. Now go to activity_main.xml.

```
    <?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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"
    android:background="#EAD2E4"
    tools:context=".MainActivity">


    <TextView
        android:id="@+id/textView3"
        android:layout_width="316dp"
        android:layout_height="107dp"
        android:layout_above="@+id/Button1"
        android:layout_alignParentTop="true"
        android:layout_alignParentEnd="true"
        android:layout_marginTop="172dp"
        android:layout_marginEnd="23dp"
        android:layout_marginBottom="63dp"
        android:fontFamily="@font/finger_paint"
        android:text="Logcat Example"
        android:textColor="#E442EF"
        android:textSize="34sp"
```

```
        android:textStyle="bold" />

    <Button
        android:id="@+id/Button1"
        android:layout_width="365dp"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:backgroundTint="#53B5E1"
        android:onClick="methodClicked"
        android:text="Button1"
        android:textColor="@android:color/white"
        app:cornerRadius="30pt" />

    <Button
        android:id="@+id/Button2"
        android:layout_width="365dp"
        android:layout_height="wrap_content"
        android:layout_below="@id/Button1"
        android:layout_centerInParent="true"
        android:layout_marginTop="10dp"
        android:backgroundTint="#53B5E1"
        android:onClick="methodClicked"
        android:text="Button2"
        android:textColor="@android:color/white"
        app:cornerRadius="30pt" />

    <Button
        android:id="@+id/Button3"
        android:layout_width="365dp"
        android:layout_height="wrap_content"
        android:layout_below="@id/Button1"
        android:layout_centerInParent="true"
        android:layout_marginTop="70dp"
        android:backgroundTint="#53B5E1"
        android:onClick="methodClicked"
        android:text="Button3"
        android:textColor="@android:color/white"
        app:cornerRadius="30pt" />

    <Button
        android:id="@+id/Button4"
        android:layout_width="365dp"
        android:layout_height="wrap_content"
        android:layout_below="@id/Button1"
        android:layout_centerInParent="true"
        android:layout_marginTop="130dp"
        android:backgroundTint="#53B5E1"
        android:onClick="methodClicked"
        android:text="Button4"
        android:textColor="@android:color/white"
        app:cornerRadius="30pt" />

    <Button
        android:id="@+id/Button5"
        android:layout_width="365dp"
        android:layout_height="wrap_content"
        android:layout_below="@id/Button1"
        android:layout_centerInParent="true"
        android:layout_marginTop="190dp"
        android:backgroundTint="#53B5E1"
        android:onClick="methodClicked"
        android:text="Button5"
        android:textColor="@android:color/white"
        app:cornerRadius="30pt" />

</RelativeLayout>
```

iii. Now go to MainActivity.java.

```
package com.example.logcat;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.util.Log;
import android.view.View;

public class MainActivity extends AppCompatActivity {

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

    public void methodClicked(View view) {
        if(view.getId() == R.id.Button1){

            Log.e("TAG", "Button 1 is clicked");
        }
        else if(view.getId() == R.id.Button2)
        {
            Log.w("TAG", "Button 2 is clicked");
        }
        else if(view.getId() == R.id.Button3)
        {
            Log.i("TAG", "Button 3 is clicked");
        }
        else if(view.getId() == R.id.Button4)
        {
            Log.d("TAG", "Button 4 is clicked");
        }
        else if(view.getId() == R.id.Button5)
        {
            Log.v("TAG", "Button 5 is clicked");
        }
    }
}
```

iv. Now connect any physical or virtual device to android studio and click on Run button. The application is installed.

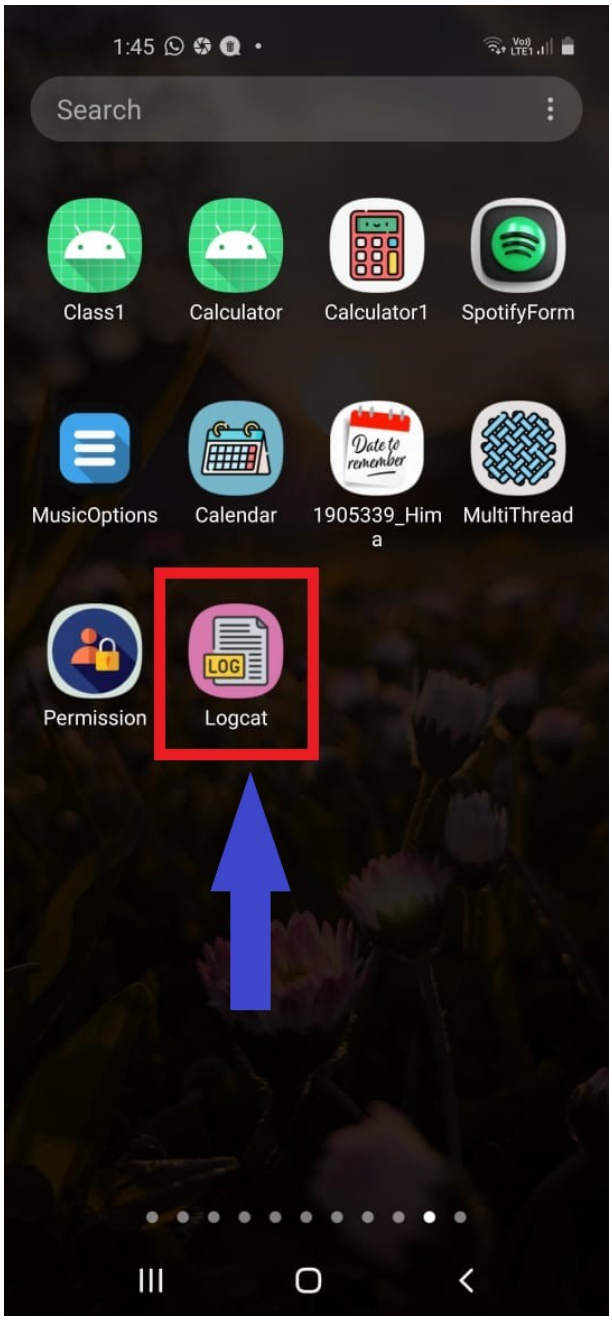

Figure 9.2: App installed successfully

v. Open Logcat and again click on run button. Now click on all the buttons one by one.

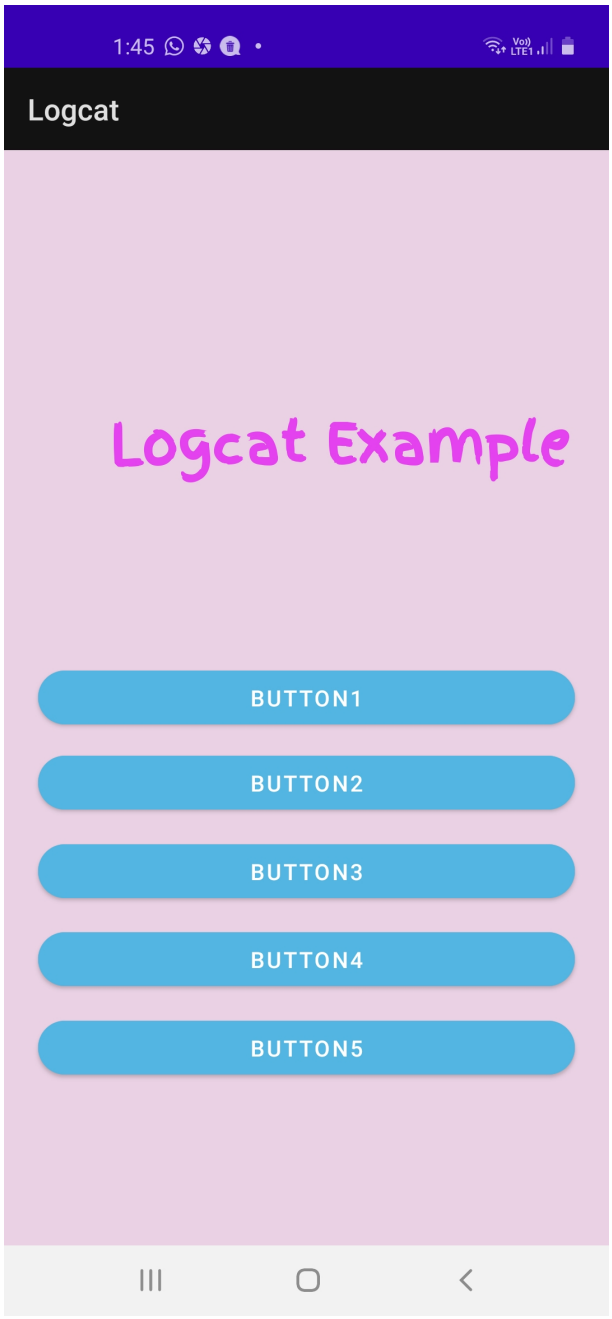

Figure 9.3: App Running

vi. Code for Logcat we have added:

```
Log.e("TAG", "Button 1 is clicked"); //error
Log.w("TAG", "Button 2 is clicked"); //warning
Log.i("TAG", "Button 3 is clicked"); //information
Log.d("TAG", "Button 4 is clicked"); //debug
Log.v("TAG", "Button 5 is clicked"); //verbose
```

vii. Now run the application and open logcat window. In logcat we have a search option where we can check easily if message is printed or not by going to Edit Filter Configuration and writing TAG name beside Log Tag option.

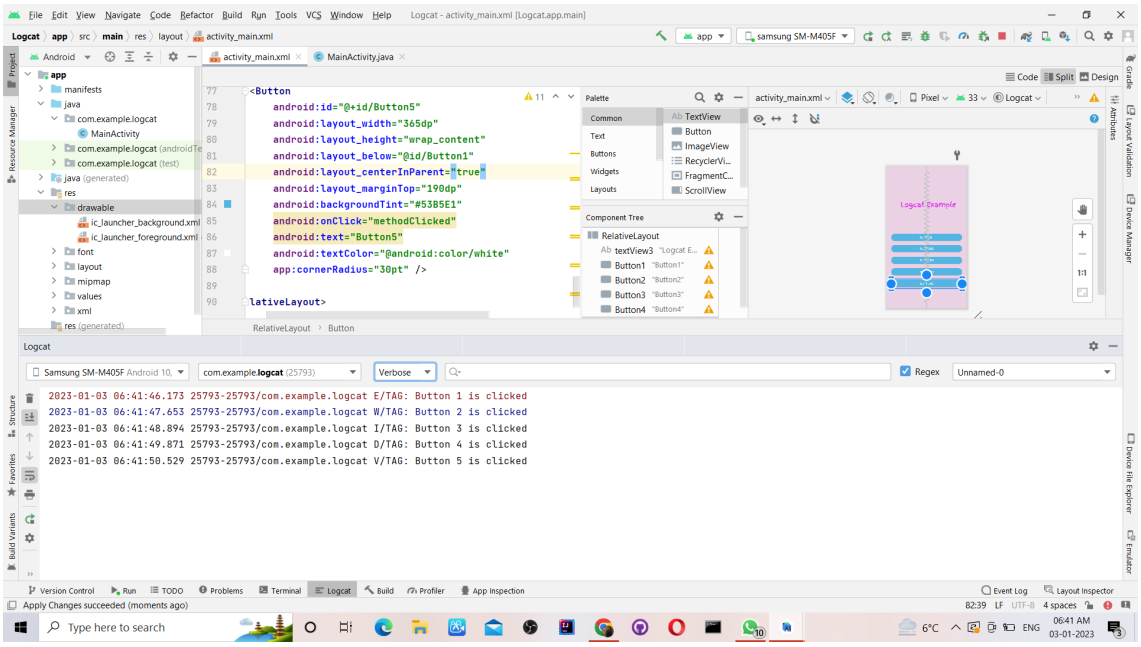

Figure 9.4: Logcat with custom message