

An android Music Player APP

Final Project Report

By

Vikrant Kumar(171385)

Under the supervision of

Dr.Ravindra Bhatt



Department of CSE,Engineering And IT

JUIT, Wagnaghat, 173234, Himachal Pradesh, INDIA

Certificate

Candidate's Declaration

I hereby declare that the work presented here in the Report "An Android music Player application" in fulfillment of the necessities for the award of the degree of Bachelor of Technology in Computer Science and Information Technology submitted within Department of computer Science and Engineering And Information Technology, Jaypee University Of Information Technology Waknaghat is an authentic record of my very own dispensed over period from January 2021 to May 2021 under the Supervision of DR. Ravindra Bhatt.

Vikrant kumar (171385)

This is to ensure that the above assertion made by the up-and-comer is consistent with the best of my insight.



Dr. Ravindra Bhatt

signature



Vikrant kumar (171385)

Abstract

There are several music players applications ,one can download any of it as per their requirement . we also created a music player application named LoKGeeT which is a simple music player application that will fetch all the downloaded songs in your device in one location that is in LoKGeeT and you can listen to them from there.It will tackle with memory problems . Java is the Backend language we used here And the user interface is done with extensible markup language i.e XML.

Intro-duction

Android is an open source operating system that runs on mobile phones ,laptops ,tabs etc. with java interface.it is developed by OHA i.e a Google Firm named as Open Handset ALLiance.Android Application is a software that runs on Android operating system. There is a term used is called apk file which is used for application which means android package .this file is zip archive and provide information,metadata and app code.Android Application coding can be done in languages as java ,kotlin and c++.Here we created application a music player application that will fetch all the songs in your device and then fetch them to one location from where you can listen to them .we have used java as back end language and XML as front end language.the App will contain homepage where there are options as play ,pause, forward, previous, seekbar ,refresh. Then there is a navigation drawer too and the option in it are colors,themes,about application,timer.

Application Version



The version we have used in our music player application is from ice cream sandwich 4.0 To latest version so far it means our application will only work on devices that has android version of ice cream sandwich 4.0 and above ,devices which version lower than it will not able to run this application .

Before get to know Activity lifecycle we must have knowledge of **what an activity is** .so basically an activity provides a screen with which we can relate/interacts. And this screen consist of widgets and layouts.

Example of how Activity lifecycle is used in applications

Suppose we are using instagram and when we open the application Oncreate() is called at this time activity is created and then onstart() method is called at this time activity is visible to user and now onResume() method is called at this time user can interact with activity. Suppose from home page of instagram you jumped to message tab means you are switching to another activity at this time your previous activity will go in to onPause() and if you return to this activity in short time interval then it will go in to onResume() and again started but something as this happened that you got phone call during this work then the activity on onPause() will have to go to onRestart () and it starts again from beginning ,same thing happened when you do not return to activity for so long then it will go to restart to start again.

System Design

Here we want to make a music player application with the help of XML as front end language and JAVa as backend language.then we will connect these two files with the help of **intent** as we all know intent is acts as a bridge to connect two activities ... first we will connect all the XML files with each other an then we will connect all these activities in our main activity file that is Activity main ... In future we will add lyrics of various cultural songs so that ease trekkers ,travellers and explorer to know the meaning of these songs . the languages we are going to use for lyrics are Hindi and English .later on our aim will be to spread it worldwide.

Presentation of designing system structure

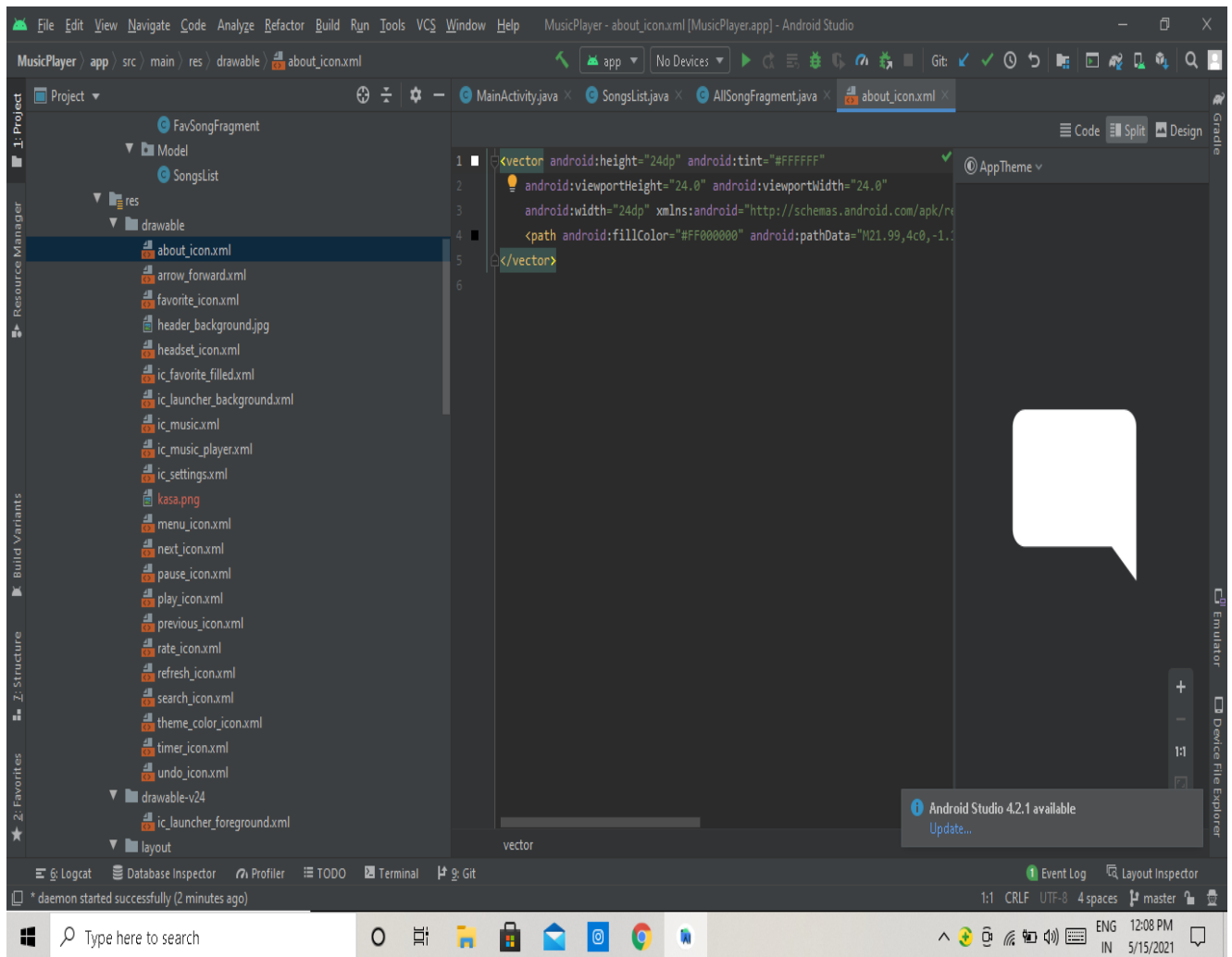
The fundamental design substance of Project incorporates: the SRC (source code), gen (steady that Android framework naturally creates), res (asset document), and the design of document and pictures in the principle stockpiling program interface,

Front end Design using XML

There is a resource file under which there are some terms that helps to make a flawless application and these are :

Drawables

Here we can add images in any format as jpeg,png but it can't be in format tiff etc and after pasting images in this option you can use them to give logo of your application and add these images to your application to make it look good.



It contains various icons as :

About icon : gives info. About music player.

Arrow forward: to jump to next music .

Favourite icon : if you like any song,you can mark it as favouritr.

Headset icon : to make app look stylist.

Menu icon : to jump to menus .

Play icon : to play song.

Pause icon : to pause song .

Previous icon : to jump to previous song.

Search icon : to search any song.

Timer icon : to set sleep timer .

Undo icon : to undo your actions .

Layouts

It defines where to add widgets and tools in th application.

Different layout used

Linear layout is ViewGroup which used to provide child view elements either horizontally or vertically.

Relative layout is used to specify the position of child view element.

Constraint layout is used to specify the position of layout constraint for every child view .

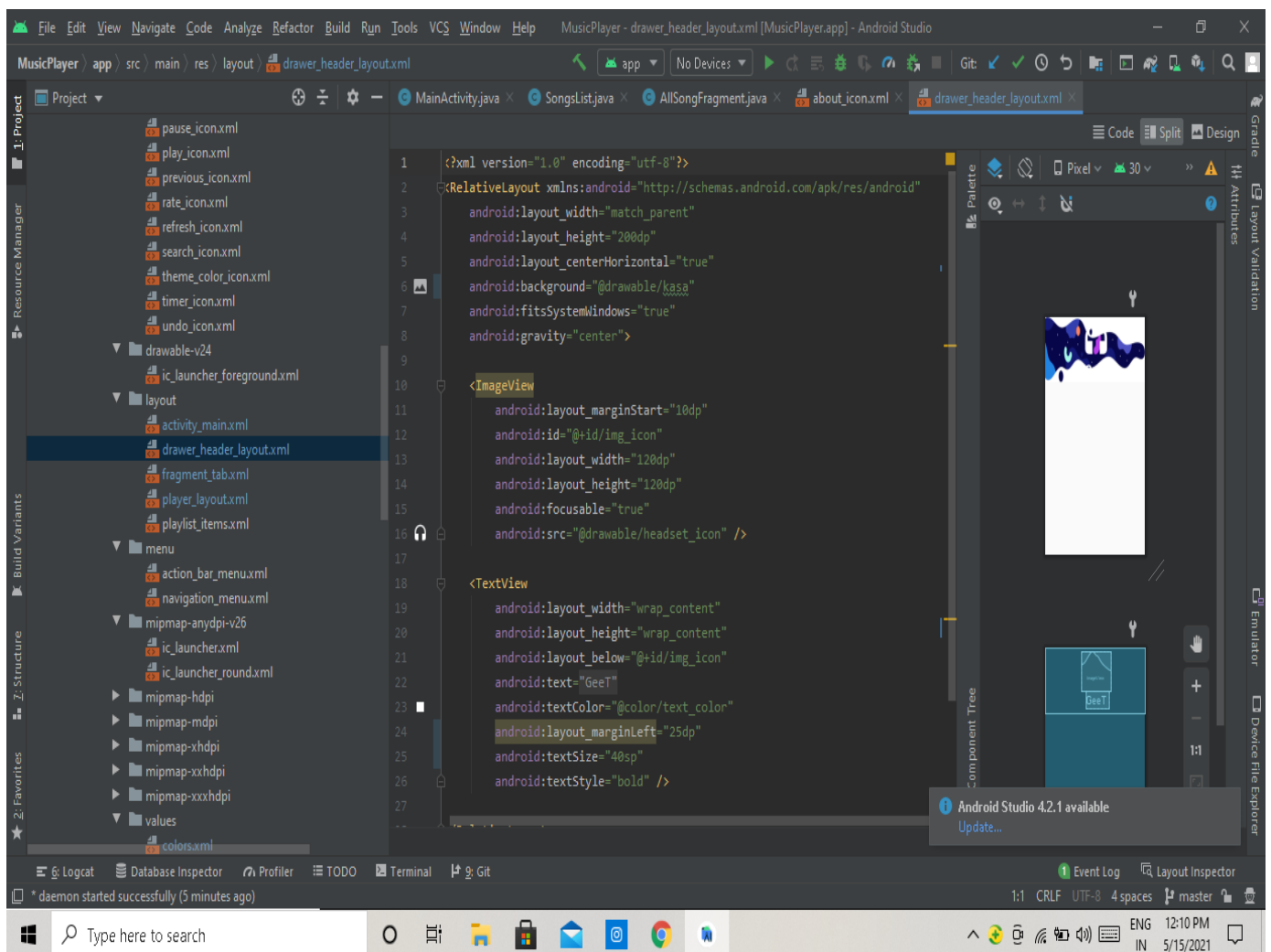
Frame Layout is used to specify position of view elements it contains pn the top of each other .

Table layout is used to display the web pages in activity layout .

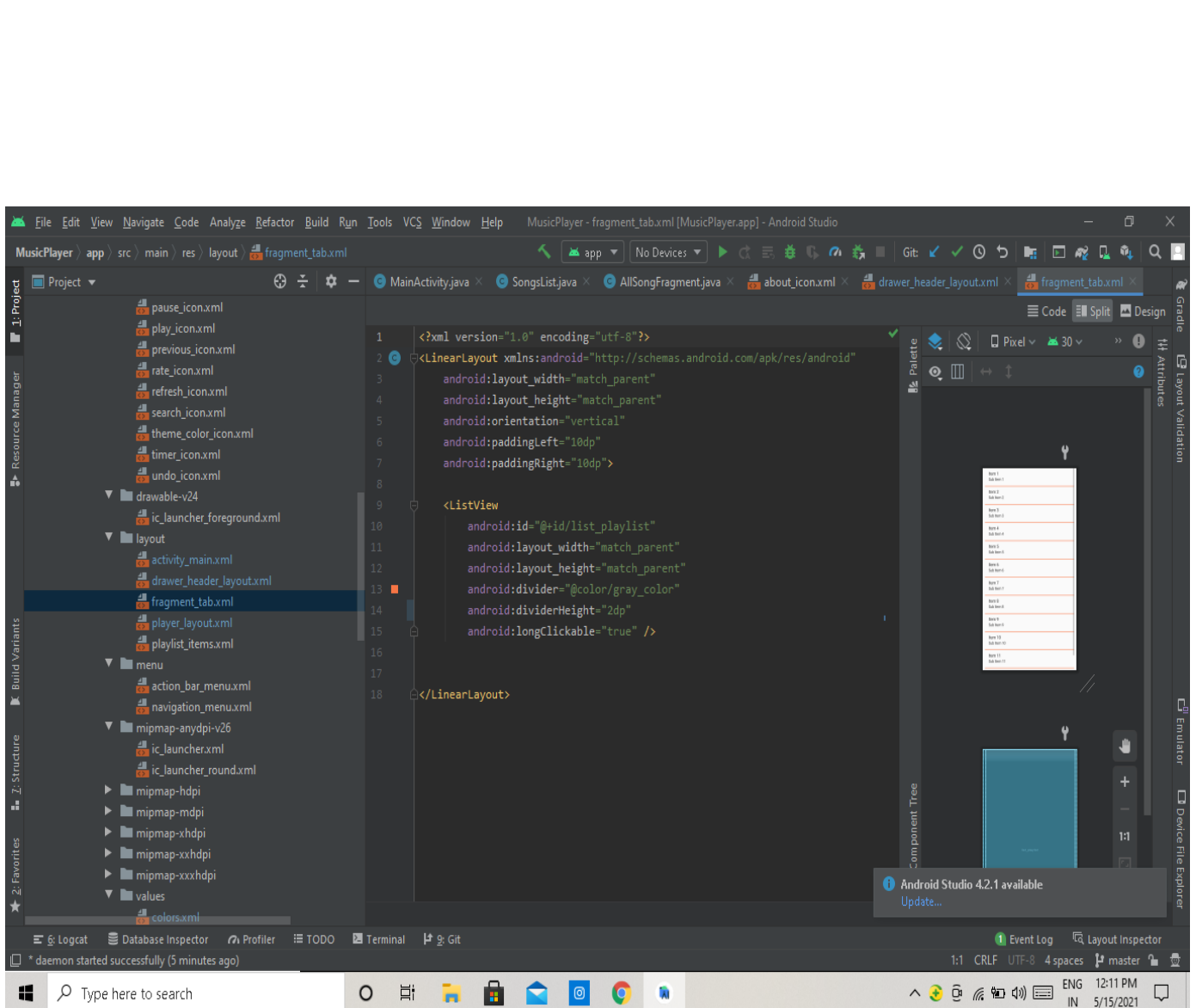
List View is used to display list of items in single column.

Grid view is used to display a scrollable list of items in a grid view of rows and columns.

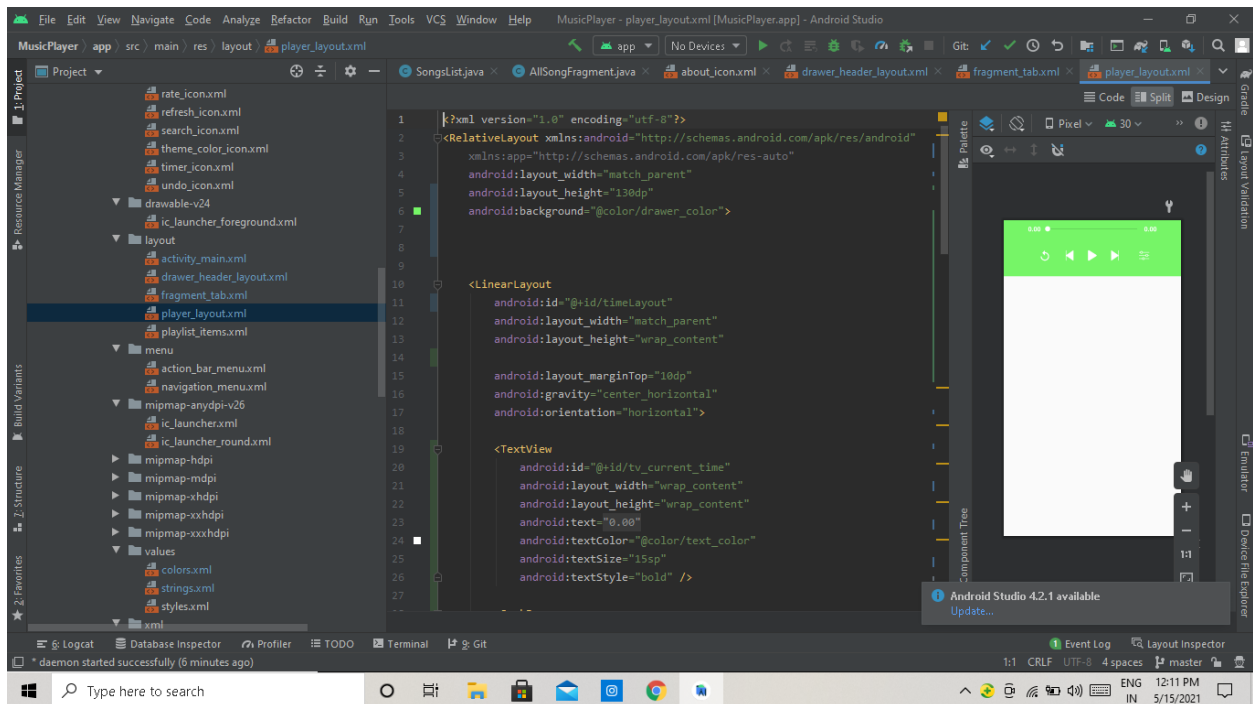
Drawer Header



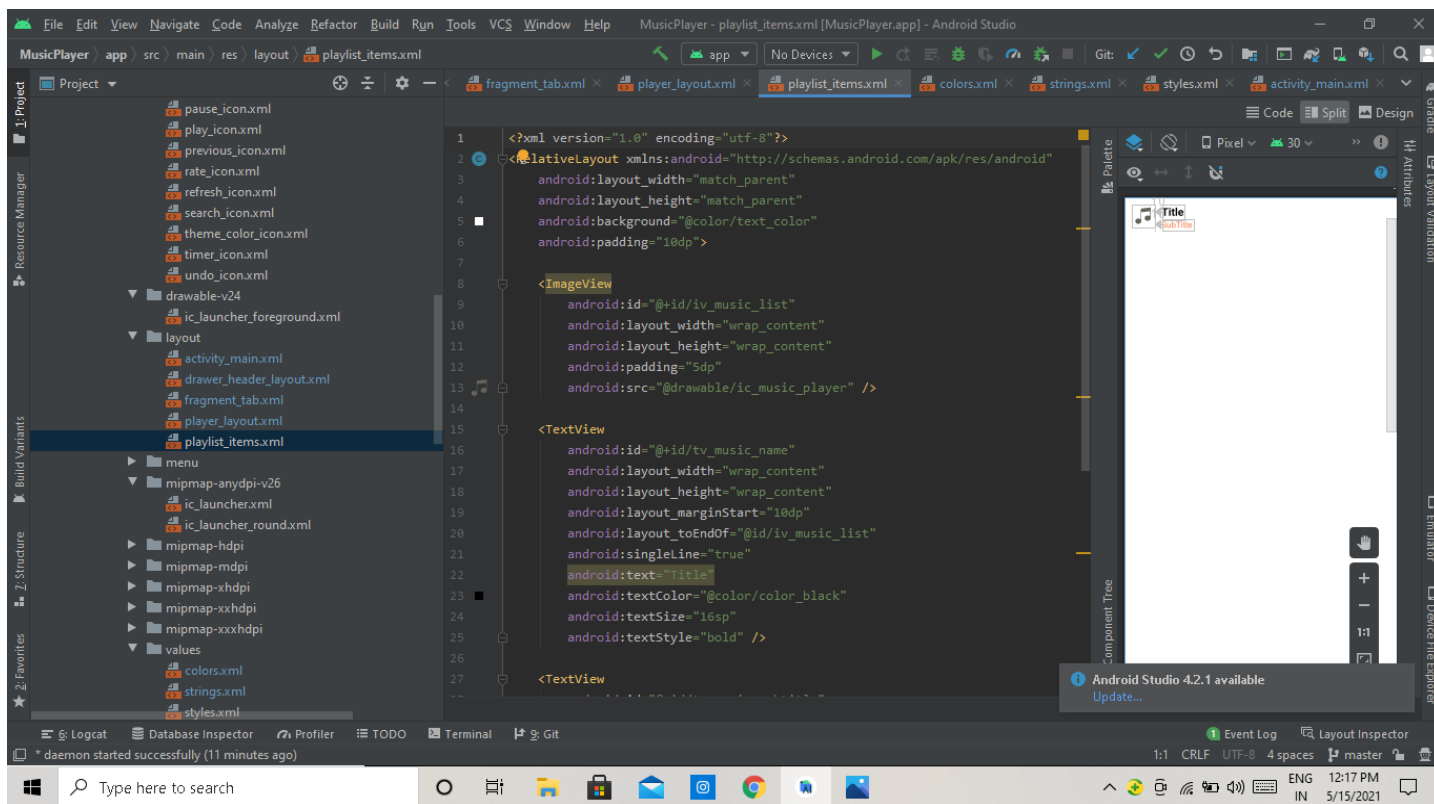
Fragment tab



Player Layout



Playlist item



Main activity

The screenshot displays the Android Studio interface for a project named "MusicPlayer". The main window shows the XML layout for the "activity_main.xml" file. The layout is defined as a `android.support.v4.widget.DrawerLayout` with the following structure:

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.v4.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res-auto"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/drawer_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".Activity.MainActivity">
    <RelativeLayout
        android:id="@+id/content_frame"
        android:layout_width="match_parent"
        android:layout_height="match_parent">
        <android.support.v7.widget.Toolbar
            android:id="@+id/toolbar"
            android:layout_width="match_parent"
            android:layout_height="?attr/actionBarSize"
            android:background="@color/color_black"
            android:theme="@style/ThemeOverlay.AppCompat.ActionBar" />
        <LinearLayout
            android:id="@+id/ll_include_controls"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@id/toolbar">
```

The interface also shows a resource manager on the left with a tree view of the project's resources, including icons, drawables, and layout files. On the right, there is a preview of the app's UI on a mobile device, showing a green progress bar and playback controls. The bottom status bar indicates the system time as 12:15 PM on 5/15/2021.

Value resource files

color.xml

It helps you to choose that which color do you want to use in application.
There are major three parts

Primary color

Primary color Dark

Accent color

These tones are utilized in some predefined assets of the android studio too. These tones on a case by case basis to be set hazy else it could bring about certain exemptions for emerge.

MusicPlayer > app > src > main > res > values > colors.xml

app No Devices

Project

- activity_main.xml
- drawer_header_layout.xml
- fragment_tab.xml
- player_layout.xml
- playlist_items.xml
- menu
- mipmap-anydpi-v26
 - ic_launcher.xml
 - ic_launcher_round.xml
- mipmap-hdpi
- mipmap-mdpi
- mipmap-xhdpi
- mipmap-xxhdpi
- mipmap-xxxhdpi
- values
 - colors.xml
 - strings.xml
 - styles.xml
 - xml
 - searchable.xml
- AndroidManifest.xml
- test
- .gitignore
- build.gradle
- proguard-rules.pro
- gradle
 - .gitignore
 - build.gradle
 - gradle.properties
 - gradlew

Resource Manager

Build Variants

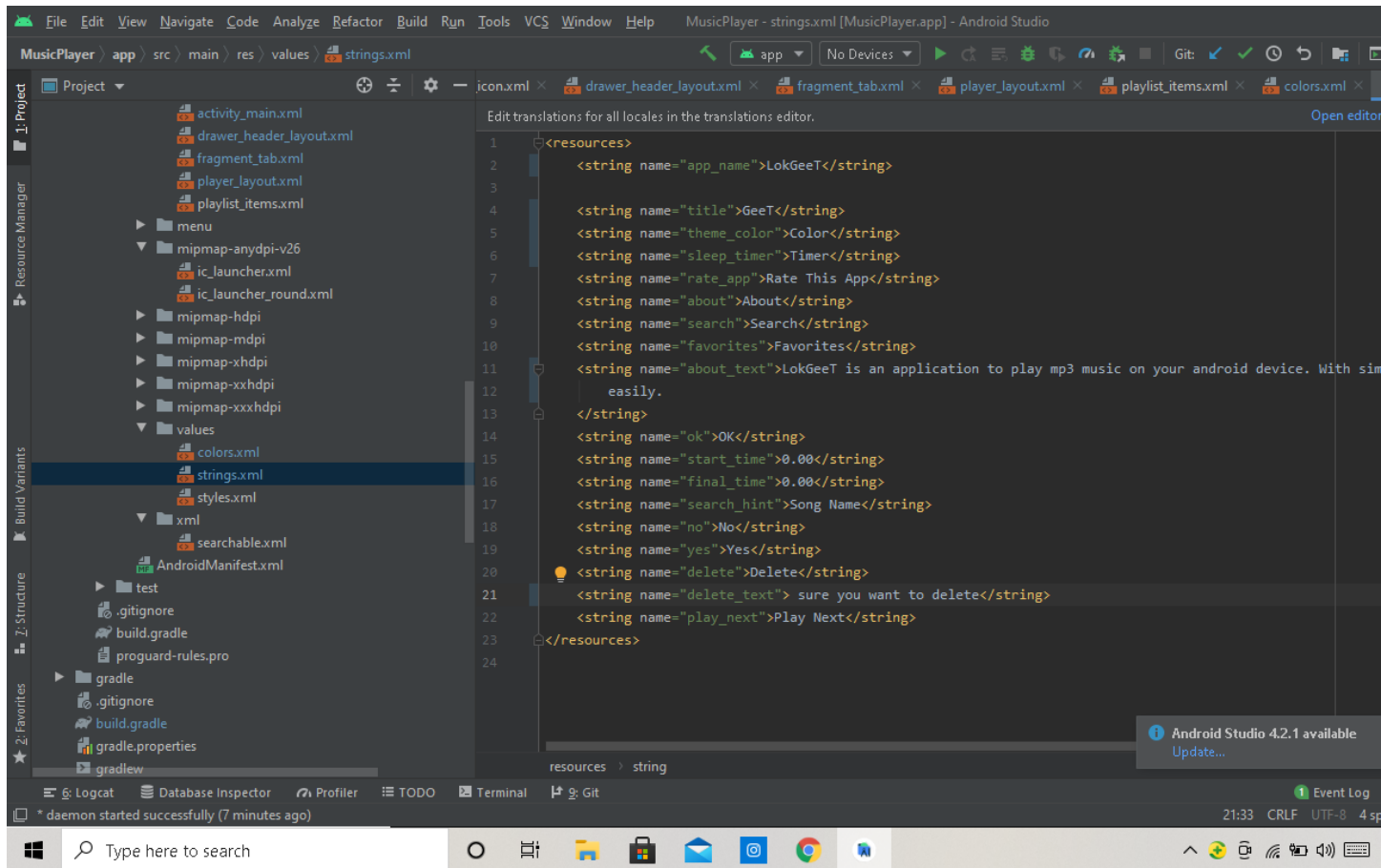
Structure

Favorites

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <resources>
3   <color name="colorPrimary">#76F567</color>
4   <color name="colorPrimaryDark">#F67545</color>
5   <color name="colorAccent">#000000</color>
6
7   <color name="text_color">#ffffff</color>
8   <color name="color_black">#000000</color>
9   <color name="drawer_color">#76F567</color>
10  <color name="light_color">#fafafa</color>
11  <color name="off_color">#dfdada</color>
12  <color name="gray_color">#F67545</color>
13
14 </resources>
15
```

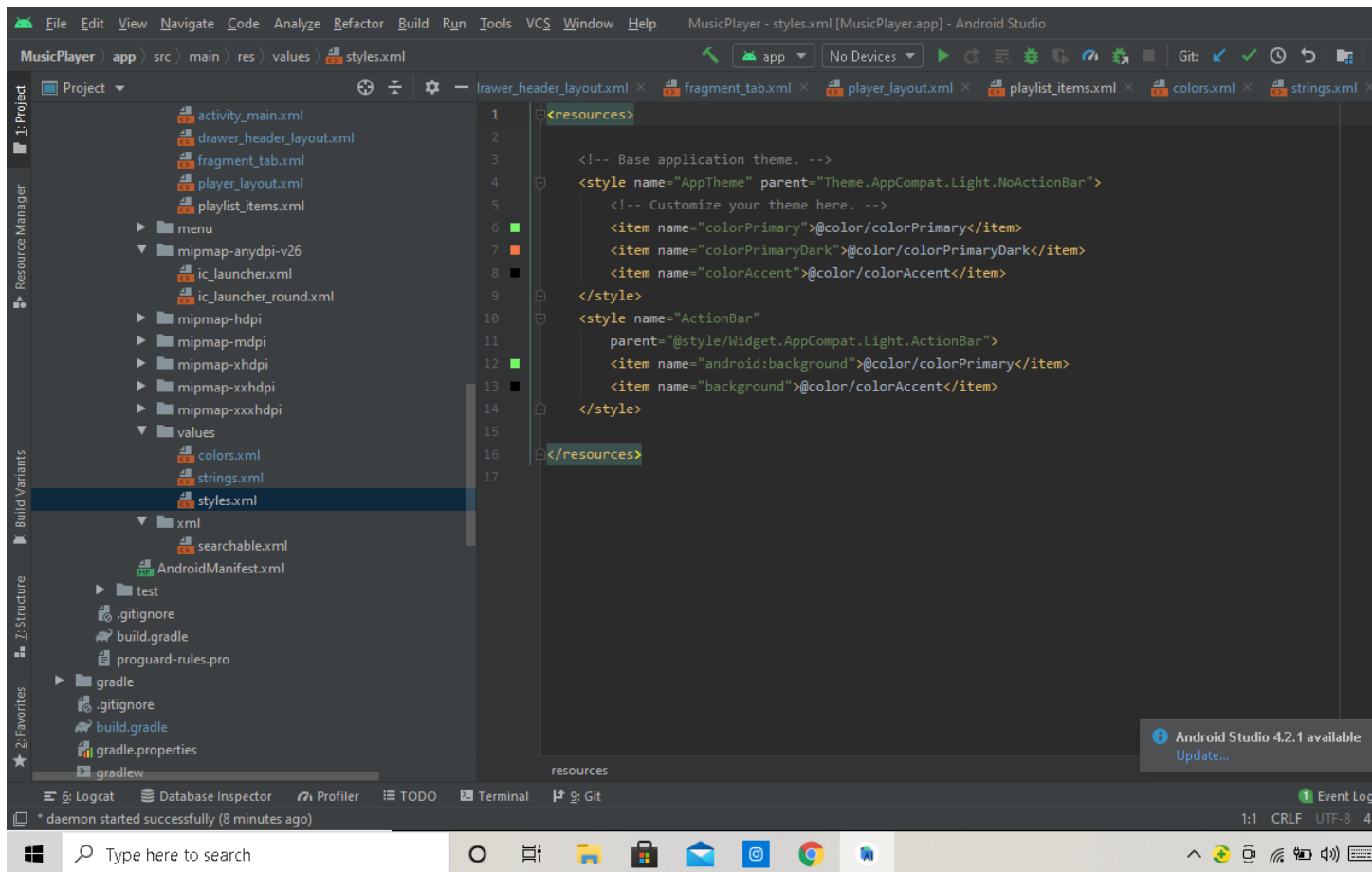

string.xml

It helps to stylize name of application , name of toolbar , name of any thing that is happening in our application layout all of them can be changed from here.



Styles.xml

Here you can stylize your application as there are many many options present at style file

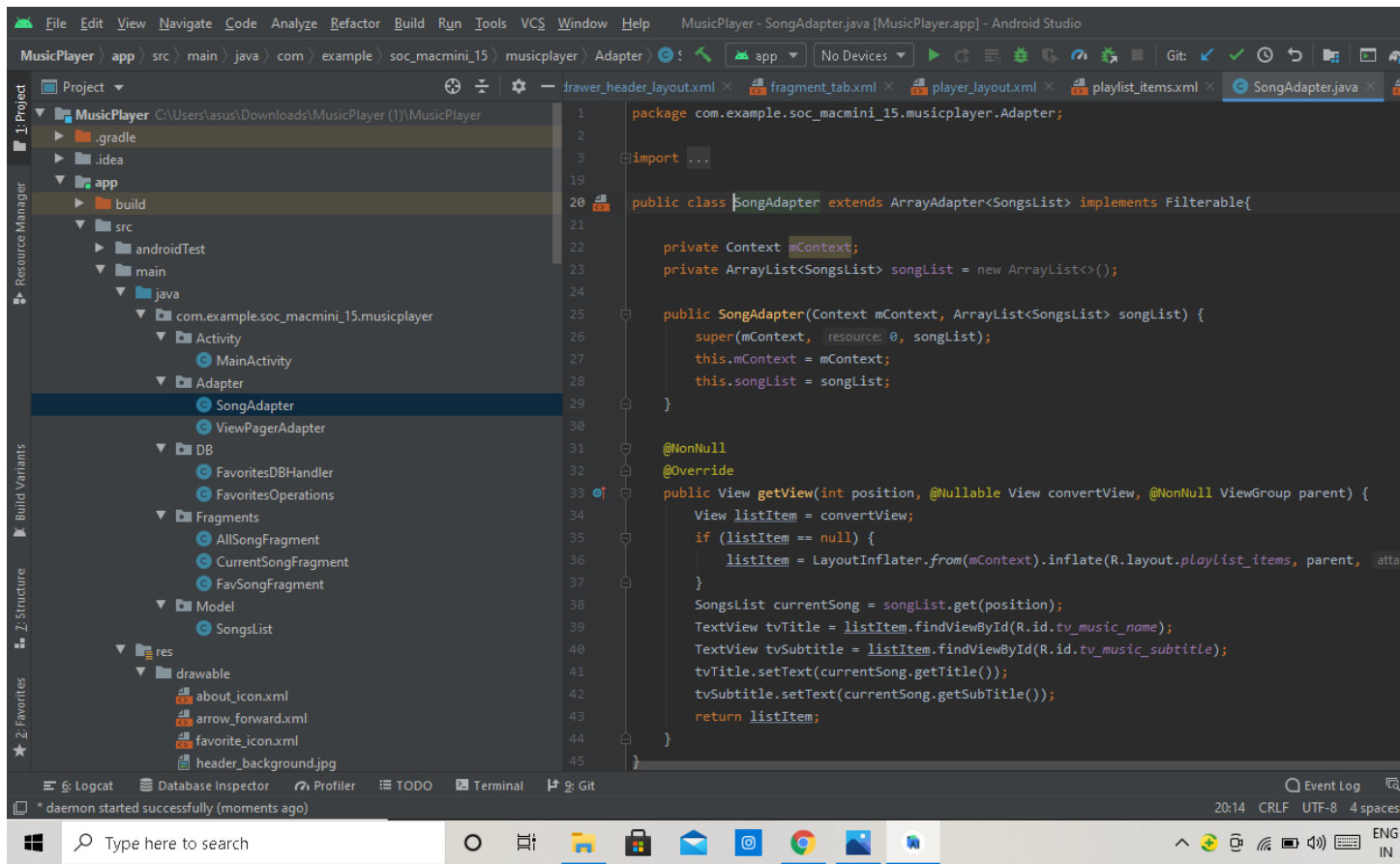


Back-End

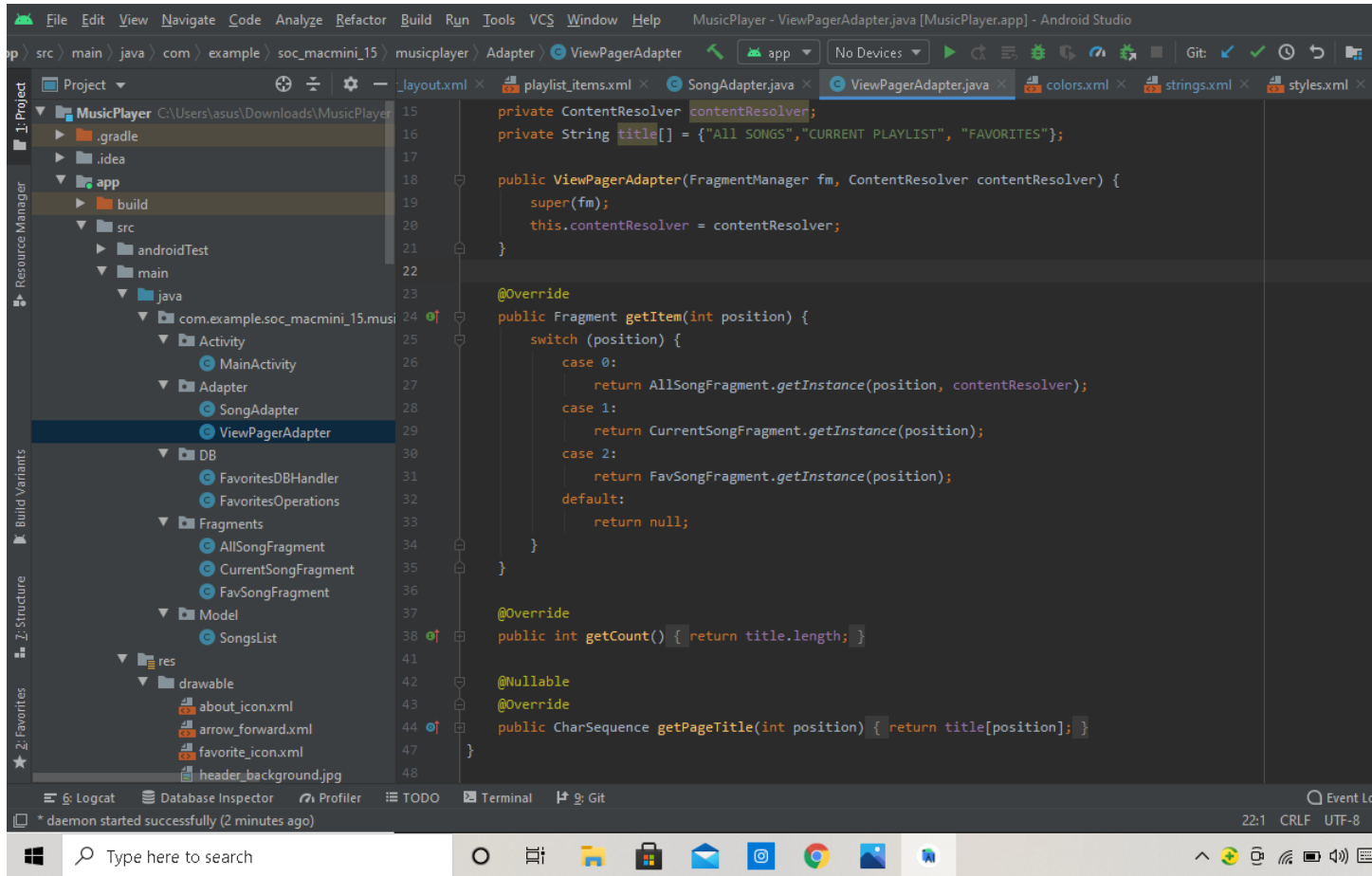
The language we used in back end is java programming language.

Adapters

songAdapter

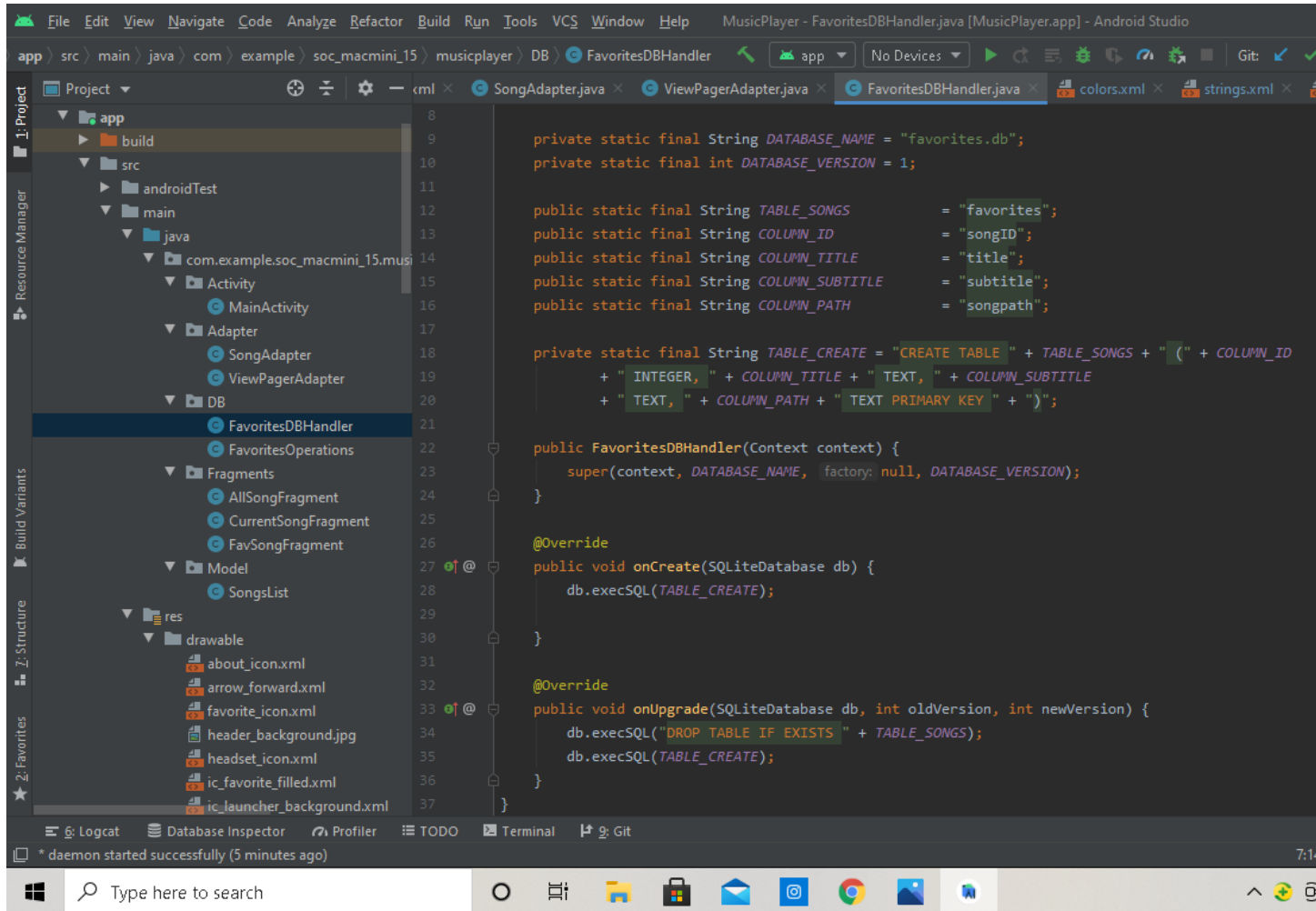


viewPagerAdapter



DataBase

DataBaseDBhandler



Fragments

There are three types of fragments we used in our application and these are:

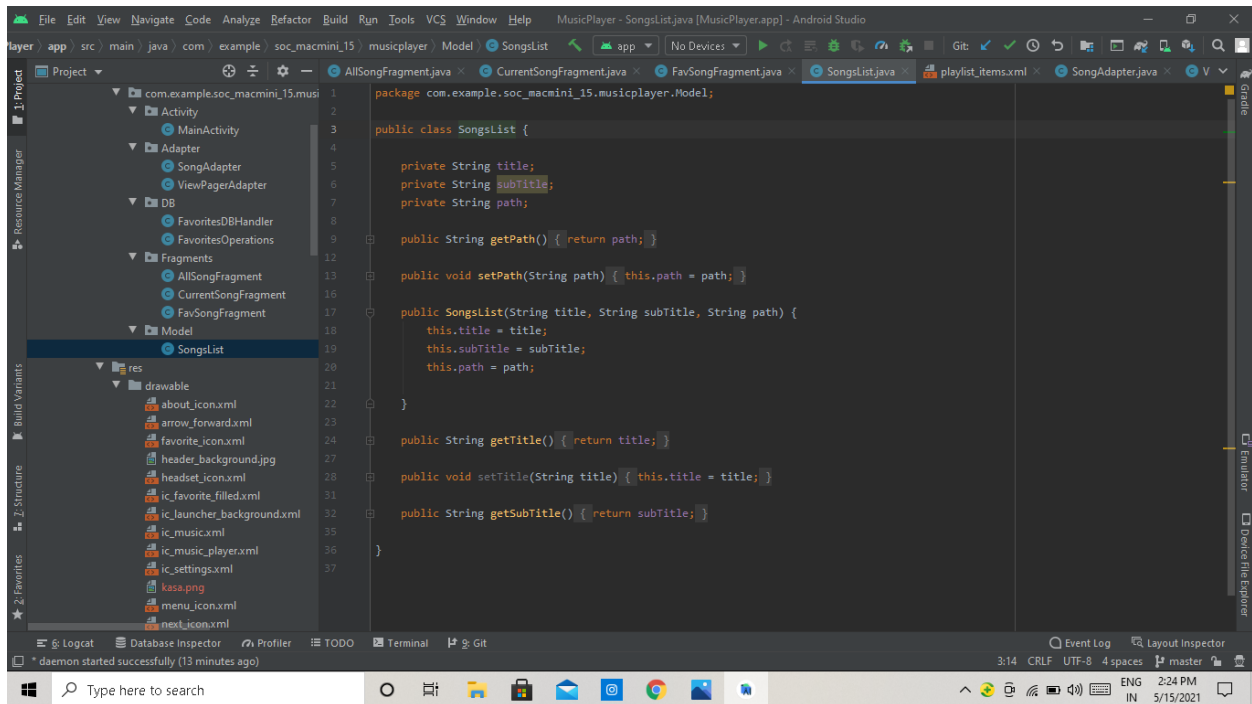
AllSongFragment

FavSongFragment

CurrentSongFragment

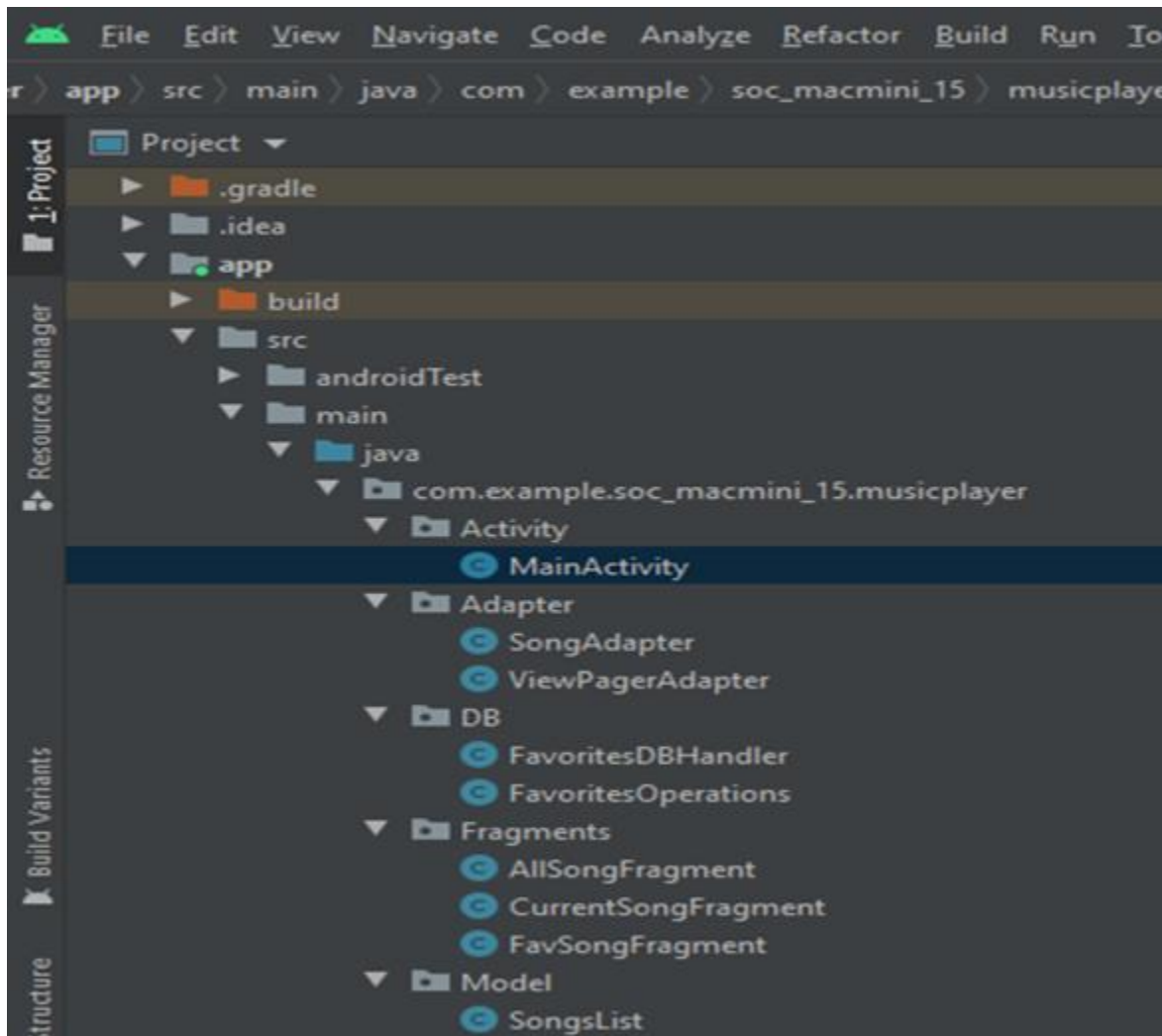
Models

SongList



Main Activity

This is the file where all activities are connected using intent, which acts as a bridge between Activities. Here are the list of files we included in our java programming :



Now with the help of Intent which acts as an intent between xml files and java files works as connector to connect two activities . we now join our main activity and activity.xml file as using syntax

```
setContentView(R.layout.activity_main);
```

Where activity_main is main Xml file

Conclusion

Music player framework understood the essential capacity of player: play, stop, rewind and fastforward a, volume change is performed through the Android System Itself, play mode, tune search, seekbar, This improvement embroiled the famous portable terminal advancement innovation. This is blend the executives of Java language in the open source versatile stage dependent on Linux framework arrangement record. The framework understood the music player programming. This plan of music player dependent on Android framework requires elaborate plan of the music player structure, by embracing ANDROID STUDIO 4.1.2 + Java language as specialized help of this framework,with the Android module apparatuses, and blend of Latest Android SDK rendition lead to the extensive and easily plan and advancement of the versatile terminal

References

1. <https://www.google.com/search?q=music+player+application+report&oq=music+player+application+report&aqs=chrome..69i57j33i160l2.9476j0j7&sourceid=chrome&ie=UTF-8>
2. <https://github.com/>
3. <https://projectworlds.in/android-projects-with-source-code/android-music-player-project-with-source-code/>
4. <https://www.youtube.com/playlist?list=PLwQLA73lSe1RfjMzbRLolkcIJBu25FnVJ>
5. <https://www.geeksforgeeks.org/android-tutorial/>