Mobile Appilication Development – 2CEIT5PE18

AIM : **AIM :**

CREATING A MINI SPOTIFY MUSIC PLAYER APPLICATION

Mini Spotify: A Comprehensive Overview

Mini Spotify is a simplified Android-based music streaming application that allows users to browse songs, play audio tracks, pause, resume, and navigate between tracks. It demonstrates the usage of **MediaPlayer**, **RecyclerView**, and basic UI components in **Android using Kotlin**.

Key Features of Mini Spotify Application

• Displays a list of songs using RecyclerView

• Play, Pause, and Stop audio functionality

• Supports local audio files

• Shows currently playing song name

• User-friendly interface similar to Spotify

• Lightweight and fast music playback

Technical Concepts Used

Mobile Appilication Development –

- **Programming Language:** Kotlin

- **UI Design:** XML Layout

- **Media Handling:** MediaPlayer class

- **List Display:** RecyclerView

- **Android Components:** Activity, Adapter

System Requirements

- Android Studio

- Android version 6.0 or above

- Kotlin support enabled

CODE :

Mobile Appilication Development –

Activity Main.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="#121212"
    android:gravity="center"
    android:padding="20dp">

    <!-- App Title -->

    <!-- Album Art -->

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:paddingBottom="20dp"
        android:text="Mini Spotify"
        android:textColor="#1DB954"
        android:textSize="26sp"
        android:textStyle="bold" />

    <ImageView
        android:layout_width="220dp"
        android:layout_height="220dp"
        android:layout_marginBottom="20dp"
        android:src="@drawable/ic_album"
        tools:srcCompat="@mipmap/album" />
```

Mobile Appilication Development –

```xml
<!-- Song Title -->
<TextView
    android:id="@+id/tvSong"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Believer"
    android:textColor="#FFFFFF"
    android:textSize="20sp"
    android:textStyle="bold"/>


<!-- Artist -->
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Imagine Dragons"
    android:textColor="#B3B3B3"
    android:textSize="14sp"
    android:paddingBottom="30dp"/>


<!-- Controls -->
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:gravity="center">
```

```xml
<!-- Controls -->
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:gravity="center">

    <ImageButton
        android:id="@+id/btnPrev"
        android:src="@drawable/ic_prev"
        android:background="@null"
        android:layout_width="60dp"
        android:layout_height="60dp"
        android:tint="#FFFFFF"/>

    <ImageButton
        android:id="@+id/btnPlay"
        android:src="@drawable/ic_play"
        android:background="@drawable/play_btn_bg"
        android:layout_width="80dp"
        android:layout_height="80dp"
        android:layout_margin="15dp"
        android:tint="#FFFFFF"/>

    <ImageButton
        android:id="@+id/btnNext"
        android:src="@drawable/ic_next"
        android:background="@null"
```

Mobile Appilication Development –

MAINACTIVITY:

```kotlin
class MainActivity : AppCompatActivity() {

    private lateinit var mediaPlayer: MediaPlayer
    private var isPlaying = false

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        val playBtn = findViewById<ImageButton>(R.id.btnPlay)

        mediaPlayer = MediaPlayer.create(this, R.raw.song)

        playBtn.setOnClickListener {
            if (!isPlaying) {
                mediaPlayer.start()
                playBtn.setImageResource(R.drawable.ic_pause)
            } else {
                mediaPlayer.pause()
                playBtn.setImageResource(R.drawable.ic_play)
            }
            isPlaying = !isPlaying
        }
    }
}
```

Mobile Appilication Development –

1. OUTPUT: