Taha Aflouk

A00300601

Media Player Assignment

Mobile Apps

Table of Contents

[CODE 1](#_Toc150709599)

[XML CODE 1](#_Toc150709600)

[Java Code 3](#_Toc150709601)

[output 10](#_Toc150709602)

# 

# Issues

My project was working fine in college it was playing music and the sound was working fine but now it does not play any sound I think there is no problem in the code like all functionality working fine but maybe it is mac issue and on the map API doing the same thing

# CODE

## 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:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity"**>  
  
 <**TextView  
 android:id="@+id/songDetails"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Hello World!"  
 android:textSize="25dp"  
 app:layout\_constraintBottom\_toTopOf="@id/image"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.498"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="1.0"** />  
  
  
 <**ImageView  
 android:id="@+id/image"  
 android:layout\_width="300dp"  
 android:layout\_height="300dp"  
 android:text="Hello World!"  
 app:layout\_constraintBottom\_toTopOf="@id/layout"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.495"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"** />  
  
 <**TextView  
 android:id="@+id/track\_info"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Hello World!"  
 android:textSize="25dp"  
 app:layout\_constraintBottom\_toTopOf="@id/seek\_bar"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.498"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@id/track\_info"  
 app:layout\_constraintVertical\_bias="1.0"** />  
  
 <**SeekBar  
 android:id="@+id/seek\_bar"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="8dp"  
 android:padding="15dp"  
 app:layout\_constraintBottom\_toTopOf="@+id/layout"  
 tools:layout\_editor\_absoluteX="-16dp"** />  
  
 <**LinearLayout  
 android:id="@+id/layout"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:orientation="vertical"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@id/image"**>  
  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:orientation="horizontal"**>  
  
 <**Button  
 android:layout\_margin="5dp"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:onClick="RewindPressed"  
 android:text="Rewind"** />  
  
 <**Button  
 android:layout\_margin="5dp"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:onClick="playPressed"  
 android:text="Play"** />  
  
 <**Button  
 android:layout\_margin="5dp"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:onClick="forwardPressed"  
 android:text="Forward"** />  
 </**LinearLayout**>  
  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:orientation="horizontal"**>  
  
 <**Button  
 android:layout\_margin="5dp"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:onClick="previousSongPressed"  
 android:text="previous Song"** />  
  
 <**Button  
 android:layout\_margin="5dp"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:onClick="pausePressed"  
 android:text="Pause"** />  
  
 <**Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:onClick="nextSongPressed"  
 android:layout\_margin="5dp"  
 android:text="Next Song"** />  
 </**LinearLayout**>  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:orientation="horizontal"**>  
  
 <**Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:onClick="VolumeDownPressed"  
 android:layout\_margin="5dp"  
 android:text="Volume down"** />  
 <**Button  
 android:id="@+id/muteButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:onClick="mutePressed"  
 android:text="Mute"  
 android:layout\_margin="5dp"** />  
  
 <**Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:onClick="VolumeUpPressed"  
 android:layout\_margin="5dp"  
 android:text="Volume up"** />  
  
  
 </**LinearLayout**>  
  
  
 </**LinearLayout**>  
  
  
</**androidx.constraintlayout.widget.ConstraintLayout**>

## Java Code

**package** com.example.mediaproject2;  
  
**import** androidx.appcompat.app.AppCompatActivity;  
**import** androidx.collection.CircularArray;  
  
**import** android.content.Context;  
**import** android.media.AudioDeviceCallback;  
**import** android.media.AudioManager;  
**import** android.media.MediaPlayer;  
**import** android.os.Bundle;  
**import** android.os.Handler;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.ImageView;  
**import** android.widget.SeekBar;  
**import** android.widget.TextView;  
  
**import** java.util.ArrayList;  
**import** java.util.concurrent.TimeUnit;  
  
**public class** MainActivity **extends** AppCompatActivity {  
  
 **private** MediaPlayer **mediaPlayer**;  
 **private** AudioManager **audioManager**;  
 **private** TextView **songDetails**;  
 **private** TextView **trackInfo**;  
  
  
 ArrayList<Integer> **songs** = **new** ArrayList<>();  
  
  
 **private** ImageView **imageView**;  
  
 **private** SeekBar **seek\_bar**;  
 **int currentIndex** = 0;  
  
  
 **private** Runnable **trackLengthUpdate** = **new** Runnable() {  
 @Override  
 **public void** run() {  
 **if** (**mediaPlayer** != **null**) {  
 **int** duration = **mediaPlayer**.getCurrentPosition();  
 String timeString = String.*format*(**"%2d:%2d:%2d"**,  
 TimeUnit.***MILLISECONDS***.toHours(duration),  
 TimeUnit.***MILLISECONDS***.toMinutes(duration),  
 TimeUnit.***MILLISECONDS***.toSeconds(duration) % 60);  
 **trackInfo**.setText(timeString);  
 **seek\_bar**.setProgress(duration);  
 **handler**.postDelayed(**this**, 500);  
 **seek\_bar**.setEnabled(**true**);  
  
 }  
  
 }  
 };  
 **private** Handler **handler**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 **trackInfo** = findViewById(R.id.***track\_info***);  
 **seek\_bar** = findViewById(R.id.***seek\_bar***);  
 **imageView** = findViewById(R.id.***image***);  
 **songDetails** = findViewById(R.id.***songDetails***);  
 **songs**.add(0, R.raw.***adele\_\_hello***);  
 **songs**.add(1, R.raw.***burno\_mars***);  
 **songs**.add(2, R.raw.***hello\_baby\_alma***);  
 **songs**.add(3, R.raw.***ragbone\_man\_human***);  
 **songs**.add(4, R.raw.***sia\_unstoppable***);  
  
  
 **seek\_bar**.setMin(0);  
 **handler** = **new** Handler();  
  
 **seek\_bar**.setOnSeekBarChangeListener(**new** SeekBar.OnSeekBarChangeListener() {  
 @Override  
 **public void** onProgressChanged(SeekBar seekBar, **int** i, **boolean** b) {  
 **mediaPlayer**.seekTo(i);  
 }  
  
 @Override  
 **public void** onStartTrackingTouch(SeekBar seekBar) {  
 **handler**.removeCallbacks(**trackLengthUpdate**);  
 }  
  
 @Override  
 **public void** onStopTrackingTouch(SeekBar seekBar) {  
 **handler**.postDelayed(**trackLengthUpdate**, 500);  
 }  
 });  
 **seek\_bar**.setEnabled(**false**);  
  
 **audioManager** = (AudioManager) getApplicationContext().getSystemService(Context.***AUDIO\_SERVICE***);  
  
  
 }  
  
  
  
 @Override  
 **protected void** onStart() {  
 **super**.onStart();  
 **if** (**mediaPlayer** == **null**) {  
  
 **mediaPlayer** = MediaPlayer.*create*(getApplicationContext(), **songs**.get(**currentIndex**));  
 }  
 songDetails();  
 **mediaPlayer**.start();  
 **handler**.postDelayed(**trackLengthUpdate**, 500);  
 **seek\_bar**.setMax(**mediaPlayer**.getDuration());  
  
 }  
  
 @Override  
 **protected void** onPause() {  
 **super**.onPause();  
 **if** (**mediaPlayer** != **null**) {  
 **mediaPlayer**.release();  
 **mediaPlayer** = **null**;  
 }  
 }  
*// public void playPressed(View view) {  
// if (mediaPlayer == null) {  
// mediaPlayer = MediaPlayer.create(this, songs.get(currentIndex));  
// mediaPlayer.setOnCompletionListener(new MediaPlayer.OnCompletionListener() {  
// @Override  
// public void onCompletion(MediaPlayer mp) {  
// // Handle completion, if needed  
// }  
// });  
// }  
//  
// if (mediaPlayer != null && !mediaPlayer.isPlaying()) {  
// mediaPlayer.start();  
// }  
// }* **public void** playPressed(View view) {  
 **if** (**mediaPlayer** == **null**) {  
 *// If MediaPlayer is null, create and initialize it* **mediaPlayer** = MediaPlayer.*create*(**this**, **songs**.get(**currentIndex**));  
 **mediaPlayer**.setOnCompletionListener(**new** MediaPlayer.OnCompletionListener() {  
 @Override  
 **public void** onCompletion(MediaPlayer mp) {  
 *// Handle completion, if needed* }  
 });  
 **seek\_bar**.setMax(**mediaPlayer**.getDuration()); *// Set SeekBar max duration* **handler**.postDelayed(**trackLengthUpdate**, 500); *// Start updating track length* }  
  
 **if** (!**mediaPlayer**.isPlaying()) {  
 *// If MediaPlayer is not playing, start playback* **mediaPlayer**.start();  
 **seek\_bar**.setEnabled(**true**);  
 }  
 }  
  
  
*// public void playPressed(View view) {  
// if (mediaPlayer == null) {  
// mediaPlayer = MediaPlayer.create(this, songs.get(currentIndex));  
//  
// }  
// mediaPlayer.start();  
// }* **public void** pausePressed(View view) {  
 **if** (**mediaPlayer** != **null**) {  
 Button btn = (Button) view;  
 **if** (**mediaPlayer**.isPlaying()) {  
 **mediaPlayer**.pause();  
 btn.setText(**"Pause"**);  
 } **else** {  
 **mediaPlayer**.start();  
 btn.setText(**"Pause"**);  
 }  
  
 **mediaPlayer**.pause();  
 }  
 }  
*// public void pausePressed(View view) {  
// if (mediaPlayer != null) {  
// Button btn = (Button) view;  
// if (mediaPlayer.isPlaying()) {  
// mediaPlayer.pause();  
// btn.setText("Play"); // Change the button text to "Play" when paused  
// seek\_bar.setEnabled(false); // Freeze the SeekBar  
// } else {  
// mediaPlayer.start();  
// btn.setText("Pause");  
// seek\_bar.setEnabled(true); // Enable the SeekBar when playing  
// }  
// }  
// }* **public void** forwardPressed(View view) {  
 **if** (**mediaPlayer** != **null** && **mediaPlayer**.isPlaying()) {  
 **int** duration = **mediaPlayer**.getDuration();  
 **int** currentPosition = **mediaPlayer**.getCurrentPosition();  
 **int** nextPosition = currentPosition + 5000;  
  
 **if** (nextPosition <= duration) {  
 **mediaPlayer**.seekTo(nextPosition);  
 }  
 }  
 }  
  
*// public void forwardPressed(View view) {  
// if (mediaPlayer != null && mediaPlayer.isPlaying()) {  
// // Check if the player is in a valid state  
// int duration = mediaPlayer.getDuration();  
// int currentPosition = mediaPlayer.getCurrentPosition();  
//  
// // Calculate the next position  
// int nextPosition = currentPosition + 5000;  
//  
// // Check if the next position is within the bounds of the duration  
// if (nextPosition <= duration) {  
// // Seek to the next position  
// mediaPlayer.seekTo(nextPosition);  
// }  
// }  
// }* **public void** mutePressed(View view) {  
 **if** (**mediaPlayer** != **null**) {  
 AudioManager audioManager = (AudioManager) getSystemService(Context.***AUDIO\_SERVICE***);  
 **if** (audioManager != **null**) {  
 **boolean** isMute = audioManager.getStreamVolume(AudioManager.***STREAM\_MUSIC***) == 0;  
 **if** (isMute) {  
 *// Unmute* audioManager.setStreamVolume(AudioManager.***STREAM\_MUSIC***, 50, 0); *// Adjust volume level as needed* ((Button) view).setText(**"Mute"**);  
 } **else** {  
 *// Mute* audioManager.setStreamVolume(AudioManager.***STREAM\_MUSIC***, 0, 0);  
 ((Button) view).setText(**"Unmute"**);  
 }  
 }  
 }  
 }  
 **public void** RewindPressed(View view) {  
 **if** (**mediaPlayer** != **null**) {  
 *//get the current track position* **int** position = **mediaPlayer**.getCurrentPosition();  
 **int** nextPosition = position - 5000;  
 **if** (nextPosition > 0) {  
 *//decrease the position* **mediaPlayer**.seekTo(nextPosition);  
 }  
 }  
 }  
  
 **public void** previousSongPressed(View view) {  
 **if** (**mediaPlayer** != **null**) {  
 **if** (**currentIndex** > 0) {  
 **currentIndex**--;  
 } **else** {  
 **currentIndex** = **songs**.size() - 1;  
 }  
 **if** (**mediaPlayer**.isPlaying()) {  
 **mediaPlayer**.stop();  
 }  
 **mediaPlayer** = MediaPlayer.*create*(**this**, **songs**.get(**currentIndex**));  
 **mediaPlayer**.start();  
 songDetails();  
 }  
 }  
  
 **public void** nextSongPressed(View view) {  
 **if** (**mediaPlayer** != **null**) {  
 **if** (**currentIndex** < **songs**.size() - 1) {  
 **currentIndex**++;  
 } **else** {  
 **currentIndex** = 0;  
 }  
 **if** (**mediaPlayer**.isPlaying()) {  
 **mediaPlayer**.stop();  
 }  
 **mediaPlayer** = MediaPlayer.*create*(**this**, **songs**.get(**currentIndex**));  
 **mediaPlayer**.start();  
 songDetails();  
  
  
 }  
 }  
  
 **private void** songDetails() {  
 **switch** (**currentIndex**) {  
 **case** 0:  
 **imageView**.setImageResource(R.drawable.***adele***);  
 **songDetails**.setText(**"Adele - Hello"**);  
 **break**;  
 **case** 1:  
 **imageView**.setImageResource(R.drawable.***burnomars***);  
 **songDetails**.setText(**"Bruno Mars - All That"**);  
 **break**;  
 **case** 2:  
 **imageView**.setImageResource(R.drawable.***almacogan***);  
 **songDetails**.setText(**"Alma Cogan - Creative Mind"**);  
 **break**;  
 **case** 3:  
 **imageView**.setImageResource(R.drawable.***ragnboneman***);  
 **songDetails**.setText(**"Rag'n'Bone Man - Human"**);  
 **break**;  
 **case** 4:  
 **imageView**.setImageResource(R.drawable.***sia***);  
 **songDetails**.setText(**"Sia - Unstoppable"**);  
 **break**;  
 **default**:  
 *// Handle the case where currentIndex is out of bounds* **break**;  
 }  
 }  
  
  
 **public void** VolumeDownPressed(View view) {  
 **if** (**mediaPlayer** != **null**) {  
 **audioManager**.adjustVolume(AudioManager.***ADJUST\_LOWER***, AudioManager.***FLAG\_PLAY\_SOUND***);  
 }  
 }  
  
 **public void** VolumeUpPressed(View view) {  
 **if** (**mediaPlayer** != **null**) {  
 **audioManager**.adjustVolume(AudioManager.***ADJUST\_RAISE***, AudioManager.***FLAG\_PLAY\_SOUND***);  
 }  
 }  
}

# output

