

**LAPORAN TUGAS BESAR  
PEMROGRAMAN MOBILE  
APLIKASI PEMUTAR MUSIK**

**DISUSUN OLEH :**

**ELVIN PITER**

**1809075020**



**PROGRAM STUDI TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS MULAWARMAN**

**SAMARINDA**

**2021**

# **BAB I**

## **PENDAHULUAN**

### **A. Latar Belakang**

Musik merupakan suara atau nada yang disatukan sedemikian rupa sehingga menghasilkan irama, lagu dan keharmonisan yang dihasilkan dari alat-alat musik yang menghasilkan bunyi. Musik adalah suatu hal yang tidak bisa dipisahkan dari manusia pada saat ini. Kalau dilihat dari perkembangannya, pada zaman dahulu musik digunakan hanya untuk upacara-upacara ritual keagamaan atau adat istiadat. Tetapi sekarang ini musik seperti sudah menjadi kebutuhan hidup manusia. Mendengarkan musik yang disukai dapat mengubah suasana hati, menghilangkan stres dan dapat membuat kita menjadi lebih rileks/santai. Mendengarkan musik dapat dilakukan dimana saja dan di waktu kapan saja, misalnya waktu bekerja, mengerjakan tugas, berolahraga, maupun sebelum tidur.

Sekarang ada banyak jenis media yang dapat digunakan untuk memutar musik yang ingin kita dengar diantaranya radio, MP3, laptop, maupun smartphone. Semua media ini mungkin ada salah satunya yang kita milikinya.

Perkembangan teknologi informasi mendorong perkembangan perangkat lunak smartphone untuk dapat memutar musik kita dimana ini membutuhkan sebuah aplikasi yang dapat memutar musik tersebut, dengan menggunakan berbagai macam tools, sekarang ada banyak aplikasi yang dapat memutar musik baik yang berbayar maupun gratis.

Mungkin kita sering menggunakan aplikasi pemutar musik di smartphone kita yang memiliki berbagai tools yang dapat kita gunakan dalam memutar musik. Namun apakah kita mengetahui apa saja tools apa saja yang digunakan untuk memutar musik tersebut sehingga musik dapat di putar.

Oleh karena itu, pada project akhir ini saya akan membuat aplikasi sederhana pemutar musik yang di buat dengan menggunakan aplikasi Android Studio dengan beberapa tools.

**B. Rumusan Masalah Manfaat**

Untuk Penelitian ini akan dibatasi pada hal-hal berikut:

- a. Memaparkan pembuatan sistem aplikasi Android yang bisa berjalan di sistem operasi Android.
- b. Aplikasi berjalan dalam kondisi offline

**C. Tujuan Pembuatan Aplikasi Android**

Tujuan dari penelitian ini adalah untuk merancang dan membuat aplikasi pemutar musik sederhana dapat berjalan pada sistem operasi Android.

**D. Manfaat**

- a. Dapat mengetahui pemutar musik sederhana yang dibangun hanya dengan beberapa tools.

## **BAB II**

### **LANDASAN TEORI**

#### **A. SeekBar**

SeekBar adalah salah satu widget ,yang berfungsi untuk menampilkan ekstensi dari widget progressBar, namun dengan cara di drag dengan jari atau touch kita (draggable).Widget ini umumnya dipakai untuk keperluan setting atau pengaturan. Seperti saat kita mengatur volume sound atau brightness(pencahayaan) di perangkat Android.

SeekBar merupakan turunan dari ProgressBar yang digunakan untuk menampilkan indikasi kemajuan suatu proses, hampir sama dengan progressBar biasanya, yang membedakan hanyalah indikasi kemajuan pada SeekBar dapat kita geser maju atau mundur dengan cara di drag dengan jari.

#### **B. ImageView**

ImageView adalah kelas digunakan untuk menampilkan file gambar di app. ImageView hadir dengan pilihan konfigurasi yang berbeda untuk mendukung jenis skala yang berbeda, diantaranya properti konfigurasi scaleTypes, center, center\_crop, fit\_xy, dan fitStart.

#### **C. Image Button**

Image Button adalah Button yang dengan tampilan sebuah gambar/image yang dapat dibuat dengan ImageButton. Gambar yang kita jadikan sebagai tombol dapat kita atur sesuai keinginan kita.

Image Button merupakan tombol yang dibuat dengan mengubah tombol objek tersebut menjadi sebuah gambar/icon. Image Button biasanya digunakan untuk mempermudah pengguna memahami tujuan dari tombol yang akan dipilih dan dapat ditampilkan sesuai aksinya.

#### **D. Drawable**

Drawable adalah abstraksi umum untuk *sesuatu yang bisa digambar*. Berbagai subclass berguna dalam skenario gambar tertentu, dan Anda dapat memperluasnya untuk menentukan objek drawable Anda sendiri yang berperilaku secara unik.

Ada dua cara untuk menentukan dan membuat instance Drawable selain menggunakan class constructor antara lain: meluaskan resource gambar (file bitmap) yang tersimpan di project dan m  
eluaskan resource XML yang menentukan properti drawable.

#### E. LinearLayout

LinearLayout adalah kelompok tampilan yang menyejajarkan semua turunan dalam satu arah, baik vertikal maupun horizontal. Anda bisa menetapkan arah tata letak dengan atribut android:orientation.

Semua turunan LinearLayout akan ditumouk satu sama lain, sehingga daftar vertikal hanya akan memiliki satu turunan perbaris, berapa pun lebarnya, dan daftar horizontal hanya akan setinggi sati baris (tinggi turunan yang tertinggi, ditambah pengisi). LinearLayout mematuhi margin antara turunan dan gravitasi (sejajar kanan, tengah, atau kiri) setiap turunan.

## BAB III

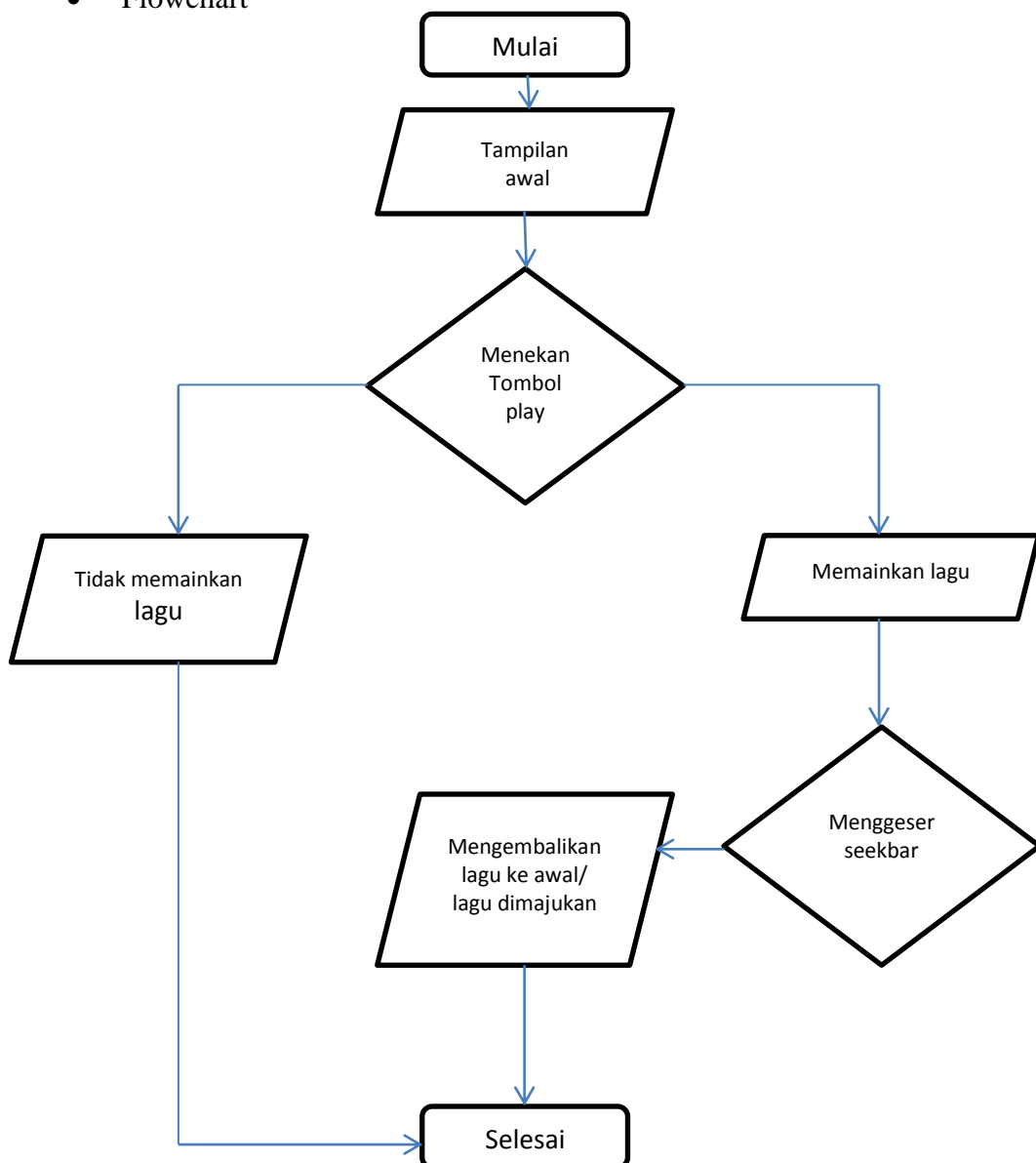
### PERANCANGAN APLIKASI

#### A. Perancangan Sistem

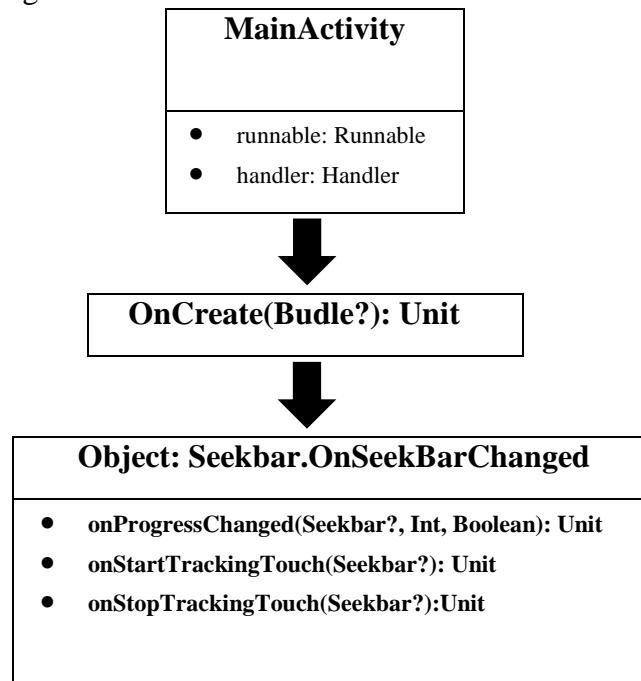
- Spesifikasi Minimum

Aplikasi Pengingat Tugas ini dapat berjalan dengan spek minimum Android versi 6.0 (Marshmallow)

- Flowchart



- Class Diagram



## B. Perancangan UI/UX

- Component

Component yang digunakan dalam aplikasi ini ada LinearLayout, TextView, ImageView, Seekbar, ImageButton

- Layout

Layout yang digunakan dalam aplikasi ini activity\_main.xml

## C. Coding

- 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:gravity="center"
    tools:context=".MainActivity">
  
```

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Music Title"
    android:textSize="26sp"
    android:textColor="#000"/>
<ImageView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@drawable/cover"
    android:layout_marginVertical="12dp"/>

<SeekBar
    android:layout_width="300dp"
    android:layout_height="wrap_content"
    android:id="@+id/seekbar"
    android:progress="50" />

<LinearLayout
    android:layout_width="300dp"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:gravity="center">
    <ImageButton
        android:layout_width="50dp"
        android:layout_height="50dp"
        android:src="@drawable/ic_skip_previous"
        android:background="@drawable/btn_bg"/>
    <ImageButton
        android:id="@+id/play_btn"
        android:layout_width="50dp"
        android:layout_height="50dp"
        android:src="@drawable/ic_play"
        android:background="@drawable/btn_bg"/>
    <ImageButton
        android:layout_width="50dp"
        android:layout_height="50dp"
        android:src="@drawable/ic_skip_next"
        android:background="@drawable/btn_bg"/>
</LinearLayout>
</LinearLayout>

```

- MainActivity

**package** com.example.pemutarmusik



```

import android.media.MediaPlayer
import android.os.Bundle
import android.os.Handler
import android.widget.SeekBar
import androidx.appcompat.app.AppCompatActivity
import kotlinx.android.synthetic.main.activity_main.*

class MainActivity : AppCompatActivity() {

    //change the seekbar position while the song is playing
    //to do this we need to create a runnable object and a handler
    lateinit var runnable: Runnable
    private var handler = Handler()

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

        // create a raw folder inside our res directory and import our music
        val mediaPlayer = MediaPlayer.create(this, R.raw.tulus_gajah)
        // add seekbar functionalities
        seekbar.progress = 0
        //add the maximum value of our seekbar the duration of the music
        seekbar.max = mediaPlayer.duration
        // create play button event
        play_btn.setOnClickListener {
            // check that the media player is not playing
            if(!mediaPlayer.isPlaying) {
                mediaPlayer.start()
                // change the button image
                play_btn.setImageResource(R.drawable.ic_pause)
            } else { // the media player is playing and we can pause it
                mediaPlayer.pause()
                play_btn.setImageResource(R.drawable.ic_play)
            }
        }
        //add the seek bar event
        // when we change our seek bar progress the song will change the
        position
        seekbar.setOnSeekBarChangeListener(object:
        SeekBar.OnSeekBarChangeListener{
            override fun onProgressChanged(seekBar: SeekBar?, pos: Int,
            changed: Boolean) {

```

```

        //Now when change the position of seekbar the music will go to
the position
        if(changed){
            mediaPlayer.seekTo(pos)

        }
    }

    override fun onStartTrackingTouch(seekBar: SeekBar?) {

    }

    override fun onStopTrackingTouch(seekBar: SeekBar?) {

    }

    })

    runnable = Runnable {
        seekbar.progress = mediaPlayer.currentPosition
        handler.postDelayed(runnable,1000)
    }
    handler.postDelayed(runnable, 1000)
    // we want that when the music finish to play the seekbar will back
to 0 and the button image change
    mediaPlayer.setOnCompletionListener {
        play_btn.setImageResource(R.drawable.ic_play)
        seekbar.progress = 0
    }

    }
}

```

- ic\_pause.xml

```

<vector
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:height="50dp"
    android:tint="#0095FF"
    android:viewportHeight="24.0"
    android:viewportWidth="24.0"
    android:width="50dp">
    <path
        android:fillColor="#FF000000"

```

```

        android:pathData="M6,19h4L10,5L6,5v14zM14,5v14h4L18,5h-
4z"/>
    </vector>

```

- ic\_play.xml

```

<vector
    android:height="50dp"
    android:tint="#0095FF"
    android:viewportHeight="24.0"
    android:viewportWidth="24.0"
    android:width="50dp"
    xmlns:android="http://schemas.android.com/apk/res/android">
    <path
        android:fillColor="#FF000000"
        android:pathData="M8,5v14l11,-7z"/>
</vector>

```

- ic\_skip\_next.xml

```

<vector
    android:height="50dp"
    android:tint="#0095FF"
    android:viewportHeight="24.0"
    android:viewportWidth="24.0"
    android:width="50dp"
    xmlns:android="http://schemas.android.com/apk/res/android">
    <path
        android:fillColor="#FF000000"
        android:pathData="M6,18l8.5,-6L6,6v12zM16,6v12h2V6h-2z"/>
</vector>

```

- ic\_skip\_previous

```

<vector
    android:height="50dp"
    android:tint="#0095FF"
    android:viewportHeight="24.0"
    android:viewportWidth="24.0"
    android:width="50dp"
    xmlns:android="http://schemas.android.com/apk/res/android">
    <path
        android:fillColor="#FF000000"
        android:pathData="M6,6h2v12L6,18zM9.5,12l8.5,6L18,6z"/>
</vector>

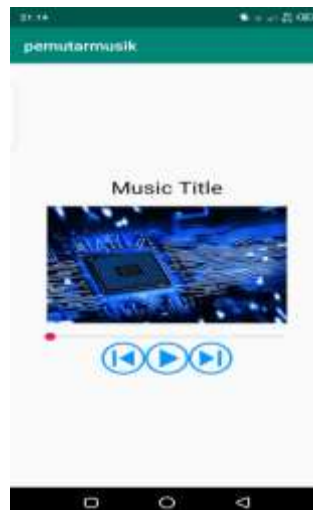
```

- btn\_bg

```
<?xml version="1.0" encoding="utf-8"?>
<shape
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:shape="oval">
  <stroke
    android:color="#0295FF"
    android:width="3dp"/>
</shape>
```

#### D. Testing

1. Tampilan awal menjalankan aplikasi



2. Tampilan saat menekan button play



### 3. Tampilan saat menggeser seekbar



### E. Debuging

Adapun masalah dalam membuat aplikasi ini adalah emulator yang digunakan untuk menampilkan hasil pengcodingan aplikasi mengalami kendala disebabkan membutuhkan penggunaan RAM/memory yang besar untuk menjalankannya. Jadi, upaya yang saya lakukan adalah dengan menggunakan Smartphone yang saya miliki sebagai emulatoanya

## **BAB IV**

### **KESIMPULAN DAN SARAN**

#### **A. Kesimpulan**

Dengan adanya aplikasi pemutar musik sederhana ini kita bisa belajar cara sederhana dalam membuat aplikasi pada android studio, dan dengan pembuatan aplikasi ini kita dapat belajar apa saja komponen yang digunakan dalam membangun sebuah aplikasi sederhana.

#### **B. Saran**

Untuk pengembangan aplikasi pemutar musik selanjutnya agar dapat menambahkan pilihan list lagu, beberapa tools lain yang dapat membuat aplikasi ini menjadi semakin menarik dan juga tampilan dari aplikasi.

## **LAMPIRAN**

<https://youtu.be/mQ0-PN4SLyY>

<https://youtu.be/mT0-NToumZU>