

# **LAPORAN PRAKTIKUM PENGEMBANGAN APLIKASI BERGERAK**

## **Android Beginner 2 - WEEK 4**



Disusun oleh:

Nama : Afif Imam Rahadi

Nim : L0122006

Kelas : A

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI DAN SAINS DATA**

**UNIVERSITAS SEBELAS MARET**

**2024**

## 1. Screenshot Source Code

Program kali ini menggunakan software Android Studio. Program yang saya buat adalah program profile pribadi yang memiliki 2 halaman yang masing-masing memiliki button dengan aksi yang berbeda. **Laporan praktikum 4 menggunakan laporan minggu lalu yang ditambahkan beberapa materi untuk melengkapi tugas praktikum 4.**

### MainActiviy.kt

#### A. Package dan Impor

```
1 package com.l0122006.afifimam.praktikum3
2
3 import android.content.Intent
4 import android.net.Uri
5 import androidx.appcompat.app.AppCompatActivity
6 import android.os.Bundle
7 import android.view.View
8 import android.widget.Button
9 import androidx.activity.enableEdgeToEdge
10 import androidx.core.view.ViewCompat
11 import androidx.core.view.WindowInsetsCompat
```

Pada gambar di atas terdapat package yang saya buat pada project yang saya buat untuk praktikum 3 Android beginner 1. Package yang saya buat menggunakan format yang sudah diberikan oleh asisten.

Selanjutnya terdapat import library yang saya gunakan untuk membantu penggunaan syntax pada android studio. Salah satu contoh yang saya gunakan adalah intent yang digunakan untuk memulai aktivitas baru, uri untuk url dan masih banyak lagi.

#### B. Class MainActivity dan override fun onCreate

```
class MainActivity : AppCompatActivity(), View.OnClickListener {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->
            val systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars())
            v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom)
            insets
        }

        val btnProfile : Button = findViewById(R.id.btn_profile)
        btnProfile.setOnClickListener(this)
        val btnGithub : Button = findViewById(R.id.btn_github)
        btnGithub.setOnClickListener(this)
    }
}
```

Pada gambar di atas terdapat activity yang merupakan turunan dari AppCompatActivity() dan mengimplementasikan View.OnClickListener. Class ini merupakan komponen penting pada android. Main activity ini mengatur logika dari jalannya aplikasi.

Selanjutnya terdapat override fun onCreate yang merupakan metode yang nantinya akan dipanggil ketika aktivitas dibuat. Metode ini juga memeriksa apakah ada data yang disimpan dari aktivitas sebelumnya. Di dalam onCreate ini, terdapat enableEdgeToEdge() yang digunakan untuk penggunaan tampilan layar sebesar seluruh ruang layar.

Selanjutnya terdapat ViewCompat.setOnApplyWindowInsetsListener (findViewById(R.id.main)). Bagian ini mengatur pendengar yang akan dipanggil saat menerima inset. Selanjutnya terdapat deklarasi variable untuk button. Terdapat dua button pada main yaitu btnProfile dan btnGithub. Pendeklarasian ini mencari tombol dengan ID yang sesuai dari tata letak.

### C. Override fun onClick

```
override fun onClick(view: View?) {
    when(view?.id){
        R.id.btn_profile -> {
            val profileIntent = Intent(packageContext: this@MainActivity, ProfileActivity::class.java)
            profileIntent.putExtra(ProfileActivity.EXTRA_NIM, value: "L0122006")
            profileIntent.putExtra(ProfileActivity.EXTRA_NAME, value: "Afif Imam Rahadi")
            profileIntent.putExtra(ProfileActivity.EXTRA_BATCH, value: "Informatika, 2022")
            profileIntent.putExtra(ProfileActivity.EXTRA_DESCRIPTION, value: "I am currently studying at Sebelas Maret University")
            startActivity(profileIntent)
        }
        R.id.btn_github -> {
            val githubIntent = Intent(Intent.ACTION_VIEW, Uri.parse(uriString: "https://github.com/afifrahadi"))
            startActivity(githubIntent)
        }
    }
}
```

Pada gambar di atas, terdapat pengimplementasian metode onClick dari view.OnClickListener. Method ini akan dipanggil Ketika ada klik pada salah satu tombol yang sudah ditetapkan. Dalam method ini, terdapat when yang digunakan untuk menentukan Tindakan yang akan diambil berdasarkan ID.

Untuk setiap tombolnya, terdapat R.id. (variable). Ini digunakan untuk jika button tersebut diklik. Untuk aksinya, dapat ditulin dalam blok kode kurung kurawalnya {}. Pada button btn\_profile. Jika button tersebut di klik, maka program pertama kali akan membuat objek intent untuk memulai aktivitas ProfileActivity. Di dalamnya terdapat penambahan data ke intent menggunakan metode putExtra(). Ini akan mengirim data ke halaman ProfileActivity. **Bagian ini merupakan implementasi dari explicit Intent.** Jangan lupa pada bagian akhir tambahkan startActivity untuk memulai aktivitas pada ProfileActivity.

Selanjutnya, terdapat button untuk R.id.btn\_github. Pada bagian ini, pertama terdapat pendeklarasian pembuatan obyek intent untuk membuka link ke github menggunakan URI yang sudah ditentukan. Dan pada bagian akhir, terdapat startActivity untuk memulai aktivitas yang akan menuju profile github menggunakan intent yang sudah dibuat.

## Activity\_main.xml

### A. Isi Kode

```
<?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/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_margin="16.dp"
    android:gravity="center_horizontal"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView"
        style="@style/TextContent.TitleMedium"
        android:text="@string/my_profile"
        app:layout_constraintBottom_toTopOf="@+id/imageView"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <ImageView
        android:id="@+id/imageView"
        style="@style/ImageView"
        android:contentDescription="@string/afif_imam"
        android:src="@drawable/fotoafif_foreground"
        app:layout_constraintBottom_toTopOf="@+id/textView2"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView" />
```

```
<TextView
    android:id="@+id/textView2"
    style="@style/TextContent.Hello"
    android:text="@string/hello_im_afif_imam_rahadi"
    app:layout_constraintBottom_toTopOf="@+id/textView3"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/imageView" />

<TextView
    android:id="@+id/textView3"
    style="@style/TextContent.Paragraf"
    android:text="@string/pertama"
    android:textAlignment="center"
    app:layout_constraintBottom_toTopOf="@+id/btn_profile"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView2" />
```

```

<Button
    android:id="@+id/btn_profile"
    style="@style/ButtonGeneral"
    android:text="@string/go_tomy_profile"
    app:layout_constraintBottom_toTopOf="@+id/btn_github"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView3" />

<Button
    android:id="@+id/btn_github"
    style="@style/ButtonGeneral.Github"
    android:text="@string/go_to_my_github"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/btn_profile" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

Pada bagian xml ini merupakan gambar tampilan pada hp tersebut. Pada bagian mainxml ini, kode dimulai dengan membuat elemen <constraintLayout> sebagai root dari layoutnya. Selanjutnya diikuti dengan atribut-atribut android seperti width, height dll.

Selanjutnya untuk menambahkan element text, dapat menggunakan <TextView>. Disini akan membentuk kotak tersendiri. Ini sama seperti <div> pada html. Setelah itu, kita dapat mengatur ukuran dan margin ataupun atribut lainnya sesuai dengan yang disukai. Selanjutnya untuk menuliskan kalimat/kata, kalimat tersebut akan disimpan ke dalam folder res dan di dalam folder values terdapat file strings.xml.

```

1 <resources>
2   <string name="app_name">Praktikum3</string>
3   <string name="go_tomy_profile">Go To My Profile</string>
4   <string name="hello_im_afif_imam_rahadi">Hello, Im Afif Imam Rahadi</string>
5   <string name="go_to_my_github">Go To My Github</string>
6   <string name="afif_imam">Afif Imam</string>
7   <string name="my_profile">My Profile</string>
8   <string name="share">Share</string>
9   <string name="deskripsi">I am an Informatics Undergraduate Student. I am an active person in acad
10  <string name="pertama">This is my first application program using Android Studio regarding persona
11 </resources>

```

Selanjutnya untuk menambahkan gambar dapat menggunakan <ImageView> di sini ukuran foto juga dapat diatur sesuka kita. Untuk pengambilan foto dapat menggunakan src yang nantinya akan mengambil foto dari folder drawable. Untuk menambahkan foto, pada folder drawable dapat mengklik mouse kanan dan memilih image asset, lalu disitu anda dapat menambahkan fotonya.

Berikutnya terdapat bagian activity main. Untuk penambahan button dapat menggunakan <button>. Atribut button wajib memiliki id karena agar button ini dapat melakukan aksi/perintah pada MainActivity.kt yang sebelumnya sudah diatur.

**Pada praktikum kali ini, terdapat perbedaan pada pengaturan style pada xmlnya.** Pada praktikum kali ini pengaturan style dilakukan dengan cara membuat class style dengan namanya masing-masing yang disimpan pada folder themes di dalam file themes. Untuk pembuatan class bisa dilihat melalui gambar di bawah ini

```
<resources xmlns:tools="http://schemas.android.com/tools">
    <!-- Base application theme. -->
    <style name="Base.Theme.Praktikum3" parent="Theme.Material3.DayNight.NoActionBar">
        <!-- Customize your light theme here. -->
        <!-- <item name="colorPrimary">@color/my_light_primary</item> -->
        <item name="android:windowBackground">@color/lightSteelBlue</item>
    </style>

    <style name="TextContent">
        <item name="android:layout_width">wrap_content</item>
        <item name="android:layout_height">wrap_content</item>
        <item name="android:lineSpacingExtra">4dp</item>
    </style>

    <style name="TextContent.TitleMedium">
        <item name="android:textAppearance">@style/TextAppearance.Material3.TitleMedium</item>
        <item name="android:layout_marginTop">90dp</item>
        <item name="android:layout_marginBottom">20dp</item>
    </style>

    <style name="ImageView">
        <item name="android:layout_width">200dp</item>
        <item name="android:layout_height">200dp</item>
        <item name="android:layout_marginBottom">14dp</item>
    </style>

    <style name="ImageView.Circle">
        <item name="android:layout_width">200dp</item>
        <item name="android:layout_height">200dp</item>
        <item name="android:layout_marginBottom">20dp</item>
        <item name="android:layout_marginTop">200dp</item>
    </style>
</resources>
```

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFF</color>
    ⚡ <color name="lightSteelBlue">#B0C4DE</color>
</resources>
```

Pada gambar di atas, terdapat pendeklarasian style sesuai dengan class nya masing-masing. Penggunaan style pertama adalah background coclor pada light theme. Di sini saya membuat color terlebih dahulu di file color.xml dengan nama lightSteelBlue.

Contoh lain penggunaan style adalah <style name="TextContent">. Pada class ini terdapat atribut width dan height dengan nilai wrap\_content. Selanjutnya, terdapat class <style name="TextContent.TitleMedium">. Class ini akan diwarisi sifat style dari class TextContent sehingga tanpa mendeklarasikan width dan height, class ini sudah memiliki wrap\_content itu sendiri. Penggunaan style ini digunakan agar pada file .xml terlihat lebih ringkas dan rapih. Selain itu juga, ini dapat meminimalisir pemborosan penggunaan style. Dan terakhir jangan lupa untuk memasukkan styling pada bagian themes untuk nightnya.

Ini dilakukan agar saat user mengganti dark mode, syling pada light mode tidak berbeda dengan saat dark mode.

## ProfileActivity.kt

### A. Package dan Import

```
package com.l0122006.afifimam.praktikum3

import android.content.Intent
import android.os.Bundle
import android.view.View
import android.widget.Button
import android.widget.TextView
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat
```

Pada gambar di atas, terdapat package project dan juga import library yang akan digunakan pada file ini. Package masih sama dengan sebelumnya disebabkan masih dalam folder project yang sama. Untuk import disesuaikan dengan penggunaan pada program.

### B. Class ProfileActivity

```
class ProfileActivity : AppCompatActivity(), View.OnClickListener {
    companion object {
        const val EXTRA_NIM = "extra_nim"
        const val EXTRA_NAME = "extra_name"
        const val EXTRA_BATCH = "extra_batch"
        const val EXTRA_DESCRIPTION = "extra_description"
    }
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_profile)
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->
            val systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars())
            v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom)
            insets
        }
    }
}
```

Dalam class ProfileActivity, terdapat beberapa perintah yang dilakukan. Class ini masih sama dengan class pasc mainActivity yang merupakan turunan dari AppCompatActivity. Selanjutnya, terdapat pendeklarasian objek companion yang berisi konstanta string yang digunakan sebagai kunci untuk menambkan dan mengambil data dari intent. Berikutnya,

terdapat fungsi onCreate yang dipanggil Ketika aktivitas dibuat. Untuk bagian kodenya tidak berbeda jauh pada mainActivit dan memiliki fungsi yang sama.

```
val btnShare : Button = findViewById(R.id.btn_share)
btnShare.setOnClickListener(this)

val tvProfileData : TextView = findViewById(R.id.tv_profile_data)
val nim = intent.getStringExtra(EXTRA_NIM)
val name = intent.getStringExtra(EXTRA_NAME)
val batch = intent.getStringExtra(EXTRA_BATCH)
val description = intent.getStringExtra(EXTRA_DESCRIPTION)

val text = """
    $nim

    $name

    $batch

    Self Description

    $description
    """.trimIndent()

tvProfileData.text = text
}
```

Setelah itu terdapat pendeklarasian variable btnShare dengan id yang sesuai. Di bawahnya terdapat pendeklarasian pencarian textview untuk beberapa variable. Pencarian textview ini disesuaikan dengan ID untuk menampilkan teksnya. Data teksnya diambil dari intent menggunakan variable yang sesuai. Dan outputnya menggunakan rawstring yang seperti pada gambar.



### C. OnClick

```
override fun onClick(v: View?) {  
    when(v?.id) {  
        R.id.btn_share -> {  
            val intent = Intent(Intent.ACTION_SEND)  
            intent.type = "text/plain"  
            intent.putExtra(Intent.EXTRA_SUBJECT, value: "Name : Afif Imam Rahadi \n Nim : L0122006")  
            intent.putExtra(Intent.EXTRA_TEXT, value: "Name : Afif Imam Rahadi \n Nim : L0122006")  
  
            // Menampilkan daftar aplikasi yang dapat menangani intent share  
            startActivity(Intent.createChooser(intent, title: "Share via"))  
        }  
    }  
}
```

Di bawahnya, terdapat fungsi onclick untuk button share yang sudah dideklarasikan sebelumnya. Sama seperti sebelumnya, diawali dengan when, lalu jika id nya benar maka akan menjalankan perintah di bawahnya. Terdapat pendeklarasian objek intent untuk memulai Tindakan ACTION\_SEND dan menetapkan tipe konten intent menjadi text/plain. **Pada bagian Intent.ACTION\_SEND inilah yang merupakan implementasi dari implicit intent.** Selanjutnya, tidak lupa menambahkan data text menggunakan metode putExtra seperti judul dan teks yang ingin dibagikan. Jangan lupa diakhiri dengan startActivity untuk memulai aktivitas.

### Activity\_profile.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"  
    xmlns:tools="http://schemas.android.com/tools"  
    android:id="@+id/main"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:layout_gravity="center_horizontal"  
    android:layout_margin="16.dp"  
    tools:context=".ProfileActivity">  
  
    <de.hdodenhof.circleimageview.CircleImageView  
        android:id="@+id/profile_image"  
        style="@style/ImageView.Circle"  
        android:src="@drawable/fotoafif_foreground"  
        app:civ_border_color="#FF000000"  
        app:civ_border_width="2dp"  
        app:layout_constraintBottom_toTopOf="@+id/tv_profile_data"  
        app:layout_constraintEnd_toEndOf="parent"  
        app:layout_constraintStart_toStartOf="parent"  
        app:layout_constraintTop_toTopOf="parent" />  
  
    <TextView  
        android:id="@+id/tv_profile_data"  
        style="@style/TextContent.Paragraf"  
        android:gravity="center"  
        app:layout_constraintBottom_toTopOf="@+id/textView4"  
        app:layout_constraintEnd_toEndOf="parent"  
        app:layout_constraintStart_toStartOf="parent"  
        app:layout_constraintTop_toBottomOf="@+id/profile_image" />
```

```

<TextView
    android:id="@+id/textView4"
    style="@style/TextContent.Paragraf"
    android:text="I am an Informatics Undergraduate Student. I am an acti..."
    android:textAlignment="center"
    app:layout_constraintBottom_toTopOf="@+id/btn_share"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/tv_profile_data" />

<Button
    android:id="@+id/btn_share"
    style="@style/ButtonGeneral.Share"
    android:layout_marginBottom="233dp"
    android:text="Share"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView4" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

Sama seperti sebelumnya, file ini menggunakan constraintLayout juga. Untuk mengeluarkan text dapat menggunakan TextView, gambar menggunakan ImageView dan button menggunakan button. Yang berbeda terletak pada bagian textView pertama karena pada bagian tersebut terdapat id untuk menerima data dari profile data yang sebelumnya menggunakan raw string.

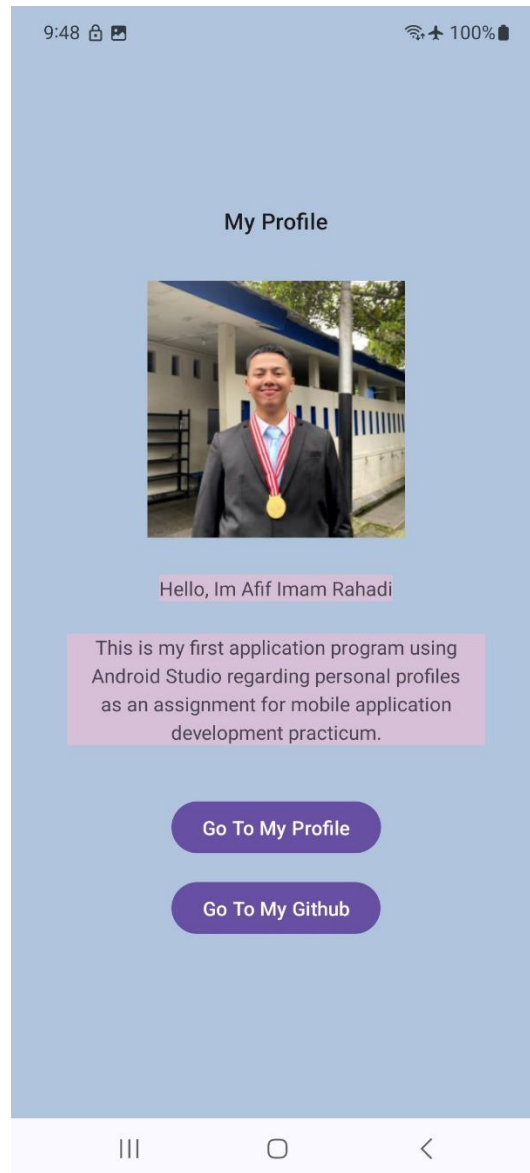
Perbedaannya selanjutnya terdapat pada imagenya, pengeluaran image menggunakan cara yang berbeda karena image akan berbentuk bulat. Atribut yang digunakan dapat dilihat pada gambar di atas.

Pada praktikum kali ini, styling dipisah dari xml nya dan dimasukkan ke dalam folder themes file themes. **Cara penggunaannya sama seperti pada file activity\_main.xml dengan menggunakan @style.** Dengan pengaturan style yang terpisah, kode program pada xml akan terlihat lebih ringkas dan rapih.

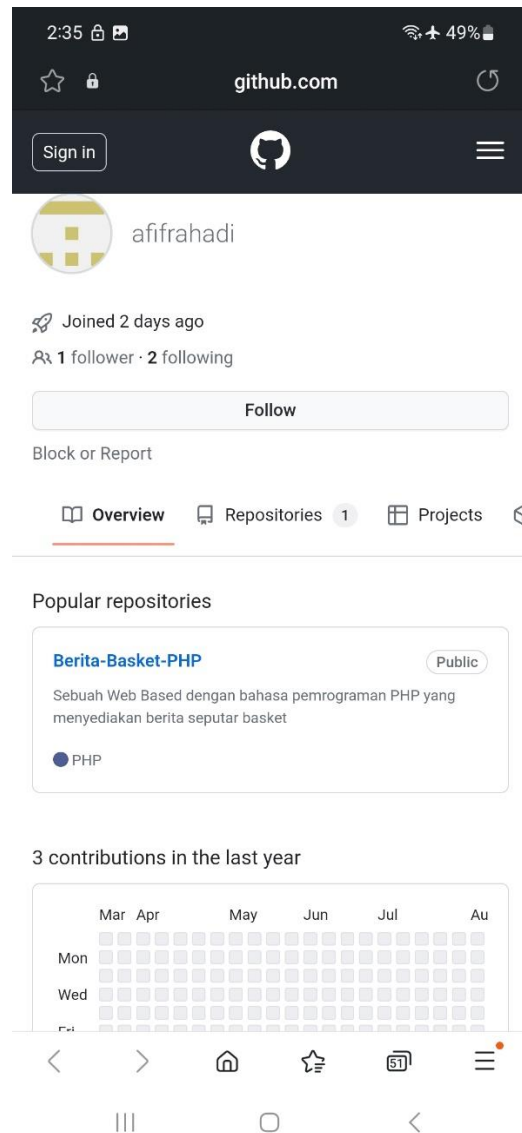
## 2. Screenshot Emulator

Hasil kali ini akan diambil dari emulator. Akan tetapi, saya menggunakan device external yaitu hp saya sendiri yang hasilnya nanti akan sama seperti emulator pada android studio.

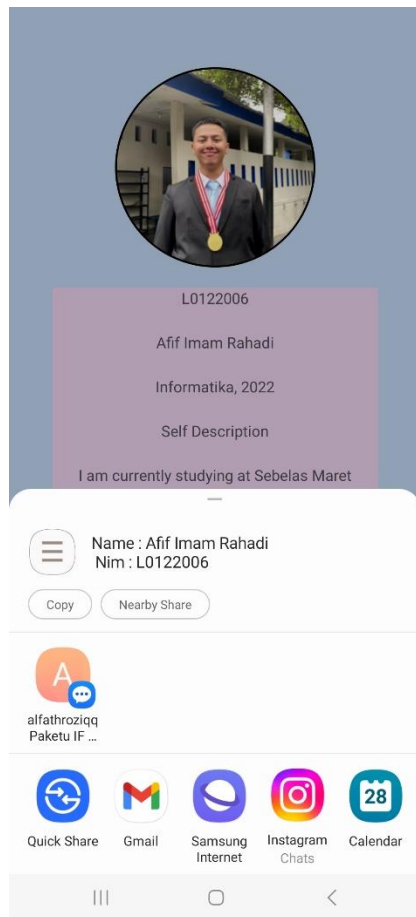
### A. Light Theme



Gambar di atas merupakan hasil program dari MainActivity. Terdapat judul, gambar, sedikit teks, dan terdapat dua button yang masing-masing memiliki aksi nya sendiri. Jika button Go To My Profile di pencet, maka akan masuk ke halaman ProfileActivity. Jika memencet button Go To My Github, maka akan masuk ke web github saya.

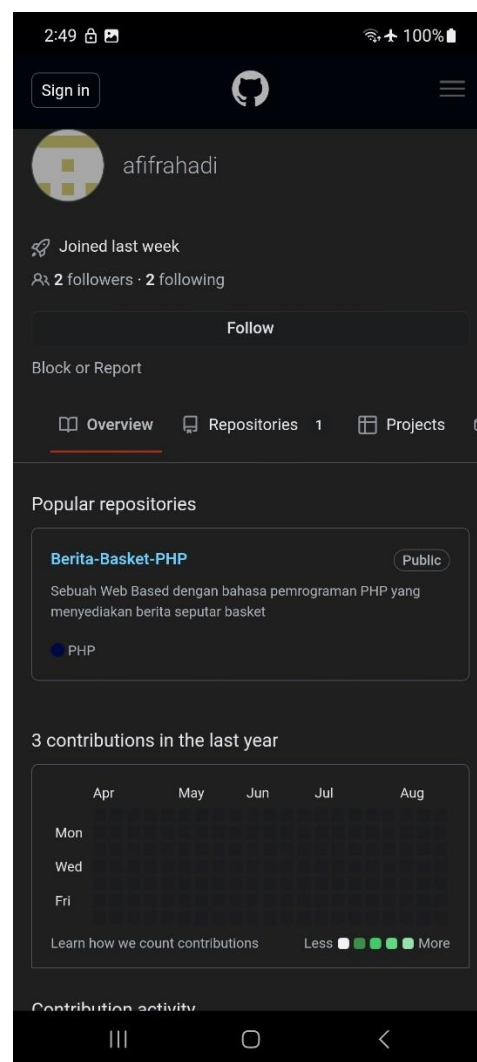
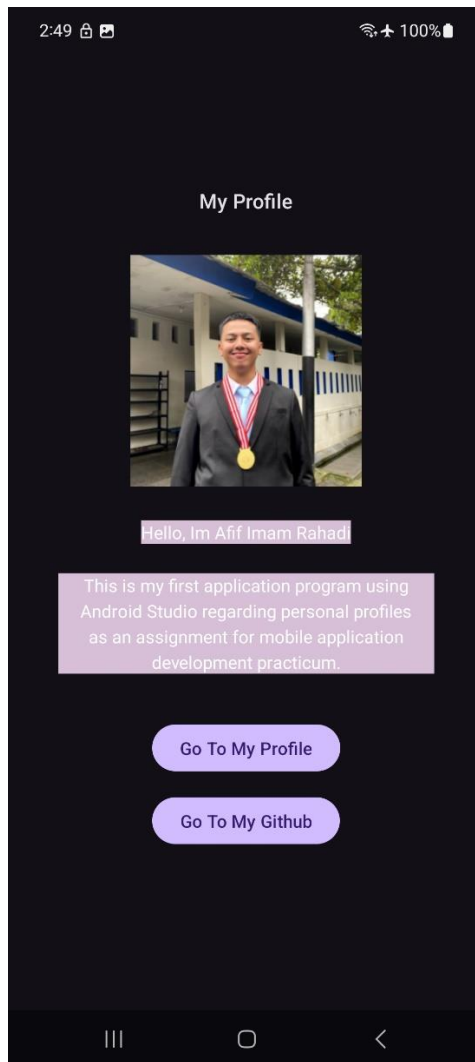


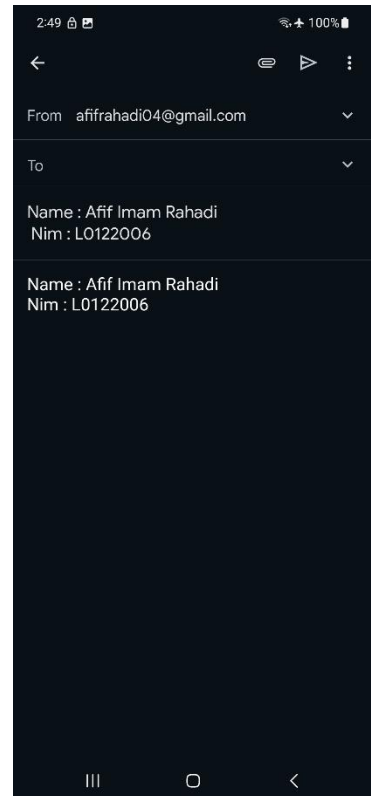
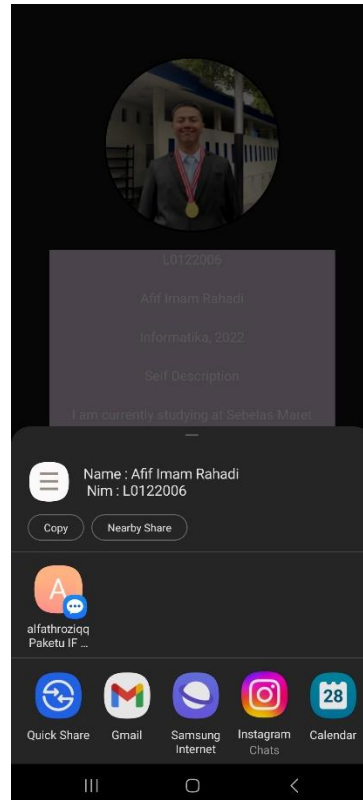
Pada halaman profile Activity, terdapat button share. Button ini akan langsung melakukan aksi dengan memunculkan aplikasi-aplikasi seperti gambar di bawah. Pada praktikum 4 ini, saya menambahkan background color berupa light steel blue yang hanya diaplikasikan pada saat menggunakan light theme.



Gambar di atas merupakan hasil dari tombol share. Pada gambar disebelah kiri terdapat aplikasi untuk melakukan share. Saya memilih gmail dan pada bagian kanan untuk subject dan content nya sudah diisi karena pada program profile activity saya sudah menambahkan menggunakan PutExtra.

## B. Dark Theme





Bagian ini memiliki hasil dan fungsi yang sama dengan light theme. Letak perbedaanya terdapat pada warna saja. Pada dark mode, background tidak diubah sama sekali sehingga hasil sesuai dengan themenya yaitu dark mode/hitam. Selain itu, semua masih sama.