

**LAPORAN PRAKTIKUM
PEMROGRAMAN MOBILE
MODUL 2**



ANDROID BASICS IN KOTLIN

Oleh:

Aurelia Monica Sari NIM. 2010817220001

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS TEKNIK
UNIVERSITAS LAMBUNG MANGKURAT
MARET 2022**

LEMBAR PENGESAHAN
LAPORAN PRAKTIKUM PEMROGRAMAN MOBILE
MODUL 2

Laporan Praktikum Pemrograman Mobile Modul 2: *Android Basics In Kotlin* ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman Mobile. Laporan Praktikum ini dikerjakan oleh:

Nama Praktikan : Aurelia Monica Sari
NIM : 2010817220001

Menyetujui,
Asisten Praktikum

Mengetahui,
Dosen Penanggung Jawab Praktikum

Rezi Rahadianor
NIM. 1810817210019

Andreyan Rizky Baskara, S.Kom., M.Kom.
NIP. 19930703 201903 1 011

DAFTAR ISI

LEMBAR PENGESAHAN	2
DAFTAR ISI	3
DAFTAR GAMBAR.....	4
A. Soal	5
B. Source Code	8
C. Output Program.....	11
D. Pembahasan.....	13

DAFTAR GAMBAR

Gambar 1. Screenshot Hasil Jawaban Soal 1	11
-------------------------------------------------	----

A. Soal

Buatlah sebuah aplikasi yang dapat menampilkan 2 (dua) buah dadu yang dapat berubah-ubah tampilannya pada saat user menekan tombol “Roll Dice”. Aturan aplikasi yang akan dibangun adalah sebagaimana berikut:

1. Tampilan awal aplikasi setelah dijalankan akan menampilkan 2 buah dadu kosong seperti dapat dilihat pada Gambar 1.



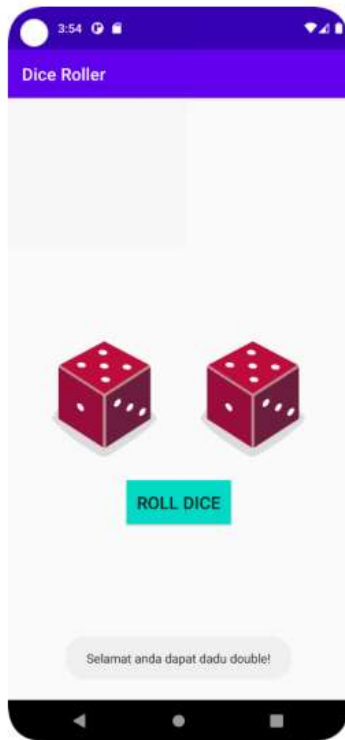
Gambar 1. Contoh Tampilan Awal Aplikasi

2. Setelah user menekan tombol “Roll Dice” maka masing-masing dadu akan memunculkan sisi dadu masing-masing dengan angka antara 1 s/d 6. Apabila user mendapatkan nilai dadu yang berbeda antara Dadu 1 dengan Dadu 2 maka akan menampilkan pesan “Anda belum beruntung!” seperti dapat dilihat pada Gambar 2.



Gambar 2. Contoh Tampilan Dadu Setelah Diroll

3. Apabila user mendapatkan nilai dadu yang sama antara Dadu 1 dan Dadu 2 atau nilai double, maka aplikasi akan menampilkan pesan “Selamat anda dapat dadu double!” seperti dapat dilihat pada Gambar 3.
4. Upload aplikasi yang telah anda buat kedalam repository github ke dalam folder Modul 2 dalam bentuk project. Jangan lupa untuk melakukan Clean Project sebelum mengupload pekerjaan anda pada repo.
5. Untuk gambar dadu dapat didownload pada link berikut:
https://drive.google.com/u/0/uc?id=147HT2lIH5qin3z5ta7H9y2N_5OMW81Ll&export=download



Gambar 3. Contoh Tampilan Roll Dadu Double

B. Source Code

1. MainActivity.kt

```
1 package com.example.diceroller
2
3 import androidx.appcompat.app.AppCompatActivity
4 import android.os.Bundle
5 import android.widget.Button
6 import android.widget.ImageView
7 import android.widget.Toast
8
9 class MainActivity : AppCompatActivity() {
10     override fun onCreate(savedInstanceState: Bundle?)
11     {
12         super.onCreate(savedInstanceState)
13         setContentView(R.layout.activity_main)
14
15         val rollButton: Button =
16         findViewById(R.id.button)
17         rollButton.setOnClickListener {
18             rollDice()
19         }
20
21         rollDice()
22     }
23
24     private fun rollDice() {
25         val dice1 = Dice(6)
26         val diceRoll1 = dice1.roll()
27         val diceImage1: ImageView =
28         findViewById(R.id.imageView1)
29
30         val dice2 = Dice(6)
31         val diceRoll2 = dice2.roll()
32         val diceImage2: ImageView =
33         findViewById(R.id.imageView2)
34
35         if (diceRoll1 == diceRoll2) {
36             val toast = Toast.makeText(this, "Selamat
37 anda dapat dadu double!", Toast.LENGTH_SHORT)
38             toast.show()
39         } else {
40             val toast = Toast.makeText(this, "Anda
41 belum beruntung", Toast.LENGTH_SHORT)
42             toast.show()
43         }
44     }
```



```

45         val drawableResource1 = when (diceRoll1) {
46             1 -> R.drawable.dice_1
47             2 -> R.drawable.dice_2
48             3 -> R.drawable.dice_3
49             4 -> R.drawable.dice_4
50             5 -> R.drawable.dice_5
51             else -> R.drawable.dice_6
52         }
53         val drawableResource2 = when (diceRoll2) {
54             1 -> R.drawable.dice_1
55             2 -> R.drawable.dice_2
56             3 -> R.drawable.dice_3
57             4 -> R.drawable.dice_4
58             5 -> R.drawable.dice_5
59             else -> R.drawable.dice_6
60         }
61         diceImage1.setImageResource(drawableResource1)
62         diceImage1.contentDescription =
63 diceRoll1.toString()
64
65         diceimage2.setImageResource(drawableResource2)
66         diceimage2.contentDescription =
67 diceRoll2.toString()
68     }
69 }
70
71 class Dice(private val numSides: Int) {
72
73     fun roll(): Int {
74         return (1..numSides).random()
75     }
76 }

```

2. activity_main.xml (code)

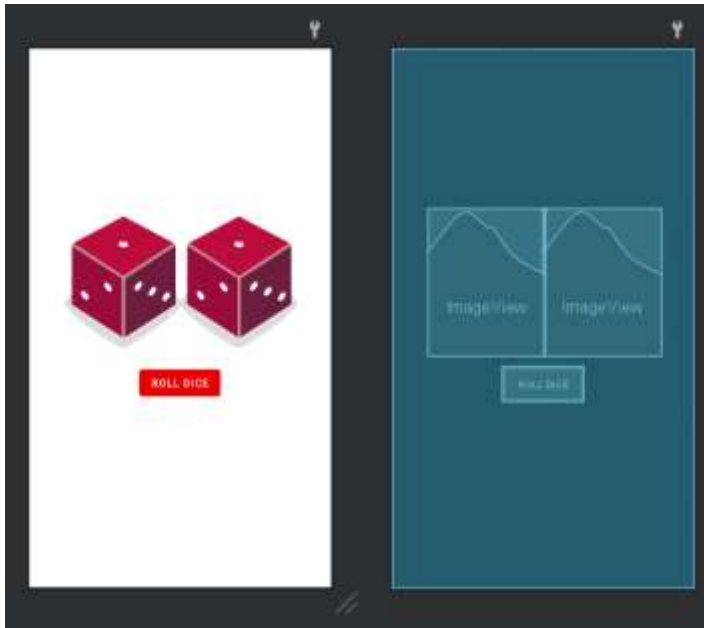
```

1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout
3  xmlns:android="http://schemas.android.com/apk/res/android"
4      xmlns:app="http://schemas.android.com/apk/res-auto"
5      xmlns:tools="http://schemas.android.com/tools"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      tools:context=".MainActivity">
9
10     <ImageView
11         android:id="@+id/imageView2"
12         android:layout_width="160dp"

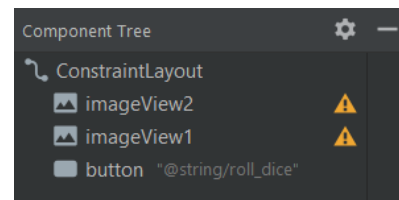
```

```
13         android:layout_height="200dp"
14         app:layout_constraintBottom_toTopOf="@+id/button"
15         app:layout_constraintEnd_toEndOf="parent"
16         app:layout_constraintHorizontal_bias="0.816"
17         app:layout_constraintStart_toStartOf="parent"
18         app:layout_constraintTop_toTopOf="parent"
19         app:layout_constraintVertical_bias="0.943"
20         tools:srcCompat="@drawable/dice_1" />
21
22     <ImageView
23         android:id="@+id/imageView1"
24         android:layout_width="160dp"
25         android:layout_height="200dp"
26         app:layout_constraintBottom_toTopOf="@+id/button"
27         app:layout_constraintEnd_toEndOf="parent"
28         app:layout_constraintHorizontal_bias="0.191"
29         app:layout_constraintStart_toStartOf="parent"
30         app:layout_constraintTop_toTopOf="parent"
31         app:layout_constraintVertical_bias="0.943"
32         tools:srcCompat="@drawable/dice_1" />
33
34     <Button
35         android:id="@+id/button"
36         android:layout_width="wrap_content"
37         android:layout_height="wrap_content"
38         android:text="@string/roll_dice"
39         app:backgroundTint="#E80101"
40         app:layout_constraintBottom_toBottomOf="parent"
41         app:layout_constraintEnd_toEndOf="parent"
42         app:layout_constraintHorizontal_bias="0.498"
43         app:layout_constraintStart_toStartOf="parent"
44         app:layout_constraintTop_toTopOf="parent"
45         app:layout_constraintVertical_bias="0.629" />
46
47 </androidx.constraintlayout.widget.ConstraintLayout>
```

3. activity_main.xml (design)



Gambar 4. Screenshot Design Pada activity_main.xml.



Gambar 5. Screenshot Component Tree

C. Output Program



Gambar 6. Tampilan Awal Aplikasi



Gambar 7. Tampilan Aplikasi Setelah Dadu Di-roll



Gambar 8. Tampilan Aplikasi Saat Dadu Double

D. Pembahasan

Pada baris 15 terdapat `findViewById()` untuk menemukan Button dalam tata letak. `R.id.button` adalah ID resource untuk Button yang merupakan ID uniknya. Kode menyimpan reference ke objek Button dalam variabel yang disebut `rollButton`, bukan objek Button itu sendiri.

Pada baris 17 terdapat objek `rollButton` dan click listener pada objek tersebut dengan memanggil metode `setOnClickListener()`.

Pada baris 18 terdapat `rollDice()` untuk memanggil fungsi `rollDice()`.

Pada baris 25 terdapat objek `dice1` untuk menginisialisasi dadu dengan 6 sisi.

Pada baris 26 terdapat objek `diceRoll1` untuk mengacak nilai pada dadu.

Pada baris 27 terdapat objek untuk menampilkan gambar dadu.

Pada baris 35-43 terdapat kondisional apabila kedua dadu menampilkan nilai yang sama maka akan menampilkan toast “Selamat anda mendapatkan dadu double”, dan sebaliknya apabila kedua dadu tidak menampilkan nilai yang sama maka akan menampilkan toast “Anda belum beruntung”.

Pada baris 45-68 terdapat kode untuk menampilkan gambar sesuai nilai dadu menggunakan kondisional `when`.

Pada baris 71 terdapat class `Dice` agar dapat menerima bilangan bulat yang disebut `numSides`.

Pada baris 73-74 terdapat fungsi untuk mengacak nilai dadu.