

**LAPORAN PRAKTIKUM
PEMROGRAMAN MOBILE
MODUL 2**



ANDROID BASICS IN KOTLIN

Oleh:

Firda Aulia

NIM. 2010817220010

**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 Basic in Kotlin ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman Mobile. Laporan Praktikum ini dikerjakan oleh:

Nama Praktikan : Firda Aulia
NIM : 2010817220010

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
DAFTAR TABEL	5
SOAL 1	6
A. Source Code	8
B. Output Program.....	11
C. Pembahasan.....	13
D. Tautan Git	15

DAFTAR GAMBAR

Gambar 1. Tampilan Awal Aplikasi.....	11
Gambar 2. Tampilan Dadu Setelah di roll.....	12
Gambar 3. Tampilan Roll Dadu Double.....	13

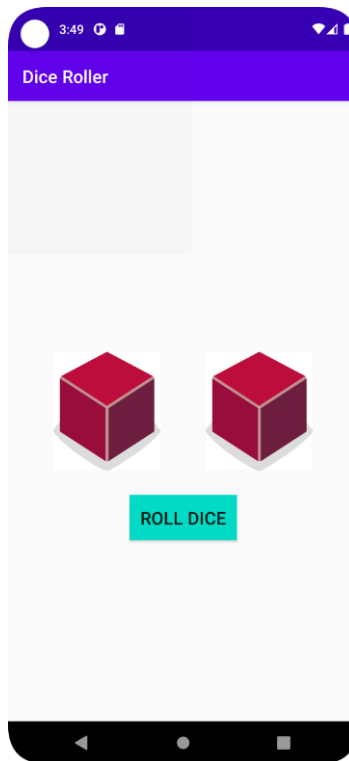
DAFTAR TABEL

Table 1. Source Code .xml	9
Table 2. Source Code .kt	10

SOAL 1

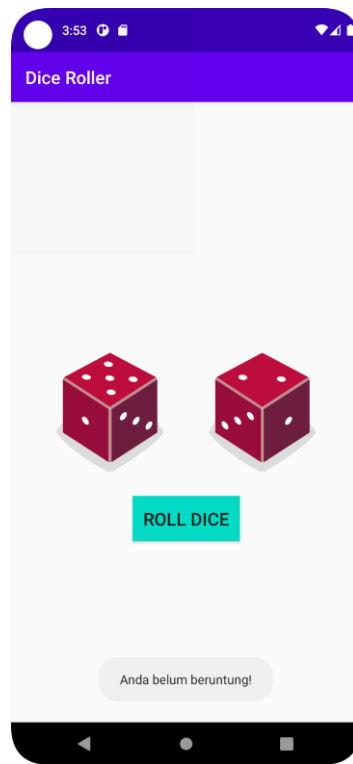
Buatlah sebuah aplikasi yang dapat menampilkan 2 (dua) buah dadu yang dapat berubah-ubah tampilannya pada saat user menekan tombol “Roll Dice”. Aturan aplikasi yang akan dibangun adalah sebagaimana berikut:

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



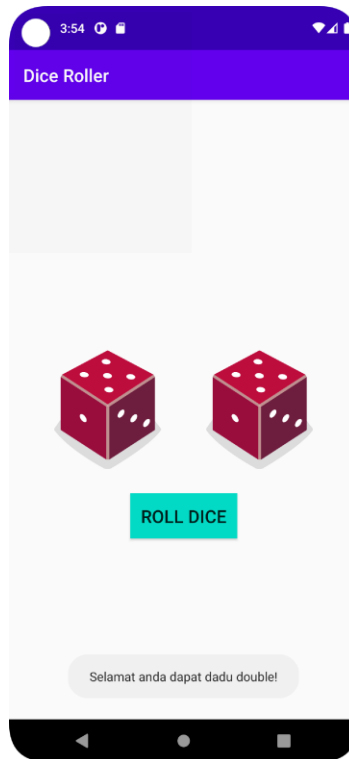
Gambar 1 Tampilan Awal Aplikasi

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 Tampilan Dadu Setelah Di Roll

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 Module 2 dalam bentuk project**. Jangan lupa untuk melakukan **Clean Project** sebelum mengupload pekerjaan anda pada repo.



Gambar 3 Tampilan Roll Dadu Double

A. Source 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	<Button
11	android:id="@+id/button"
12	android:layout_width="wrap_content"
13	android:layout_height="wrap_content"
14	android:text="@string/roll"
15	app:layout_constraintBottom_toBottomOf="parent"
16	app:layout_constraintEnd_toEndOf="parent"
17	app:layout_constraintHorizontal_bias="0.498"
18	app:layout_constraintStart_toStartOf="parent"
19	app:layout_constraintTop_toTopOf="parent"
20	app:layout_constraintVertical_bias="0.729" />
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="0.175"
29	app:layout_constraintStart_toStartOf="parent"
30	app:layout_constraintTop_toTopOf="parent"
31	app:layout_constraintVertical_bias="0.499"
32	tools:srcCompat="@drawable/dice_1"
33	android:contentDescription="TODO" />
34	
35	<ImageView
36	android:id="@+id/imageView2"
37	android:layout_width="160dp"
38	android:layout_height="200dp"
39	app:layout_constraintBottom_toTopOf="@+id/button"
40	app:layout_constraintEnd_toEndOf="parent"
41	app:layout_constraintHorizontal_bias="0.798"
42	app:layout_constraintStart_toStartOf="parent"
43	app:layout_constraintTop_toTopOf="parent"
44	app:layout_constraintVertical_bias="0.499"
45	tools:srcCompat="@drawable/dice_1" />
46	
47	</androidx.constraintlayout.widget.ConstraintLayout>
48	

Table 1. Source Code .xml

1	package com.example.diceroller
2	import androidx.appcompat.app.AppCompatActivity
3	import android.os.Bundle
4	import android.widget.Button
5	import android.widget.ImageView
6	import android.widget.TextView
7	import android.widget.Toast
8	
9	class MainActivity : AppCompatActivity() {
10	override fun onCreate(savedInstanceState: Bundle?) {
11	super.onCreate(savedInstanceState)
12	setContentView(R.layout.activity_main)
13	
14	val rollButton: Button = findViewById(R.id.button)
15	rollButton.setOnClickListener {
16	rollDice()
17	}
18	
19	rollDice()
20	}
21	
22	private fun rollDice() {
23	val dice1 = Dice(6)
24	val diceRoll1 = dice1.roll()

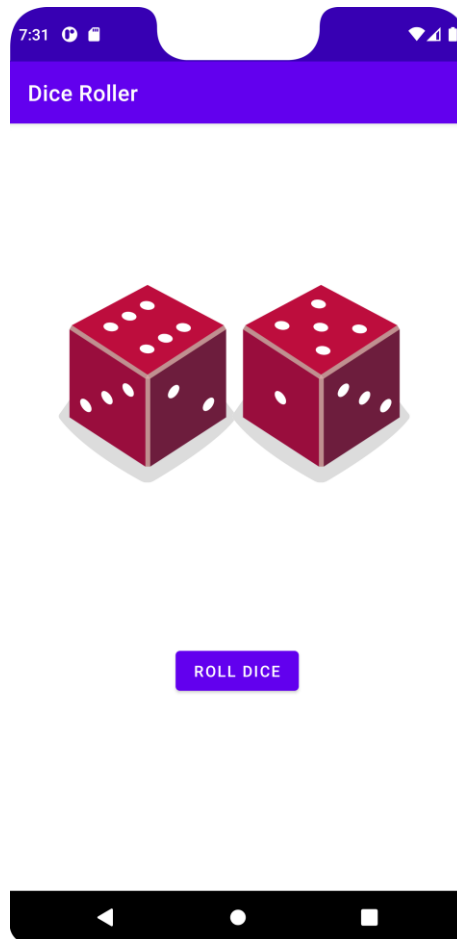
```

25     val diceImage1: ImageView = findViewById(R.id.imageView1)
26
27     val dice2 = Dice (6)
28     val diceRoll2 = dice2.roll()
29     val diceimage2: ImageView = findViewById(R.id.imageView2)
30
31     if(diceRoll1 == diceRoll2){
32         val toast = Toast.makeText(this, "selamat", Toast.LENGTH_SHORT)
33         toast.show()
34     } else {
35         val toast = Toast.makeText(this, "maaf anda belum beruntung", Toast.LENGTH_SHORT)
36         toast.show()
37     }
38
39     val drawableResource1 = when (diceRoll1) {
40         1 -> R.drawable.dice_1
41         2 -> R.drawable.dice_2
42         3 -> R.drawable.dice_3
43         4 -> R.drawable.dice_4
44         5 -> R.drawable.dice_5
45         else -> R.drawable.dice_6
46     }
47     val drawableResource2 = when (diceRoll2) {
48         1 -> R.drawable.dice_1
49         2 -> R.drawable.dice_2
50         3 -> R.drawable.dice_3
51         4 -> R.drawable.dice_4
52         5 -> R.drawable.dice_5
53         else -> R.drawable.dice_6
54     }
55     diceImage1.setImageResource(drawableResource1)
56     diceImage1.contentDescription = diceRoll1.toString()
57
58     diceimage2.setImageResource(drawableResource2)
59     diceimage2.contentDescription = diceRoll2.toString()
60 }
61 }
62
63 class Dice(private val numSides: Int) {
64
65     fun roll(): Int {
66         return (1..numSides).random()
67     }
68 }
69
70
71
72
73

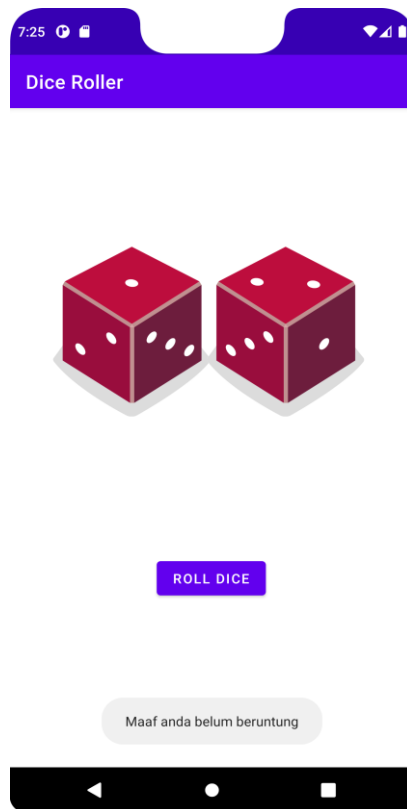
```

Table 2. Source Code .kt

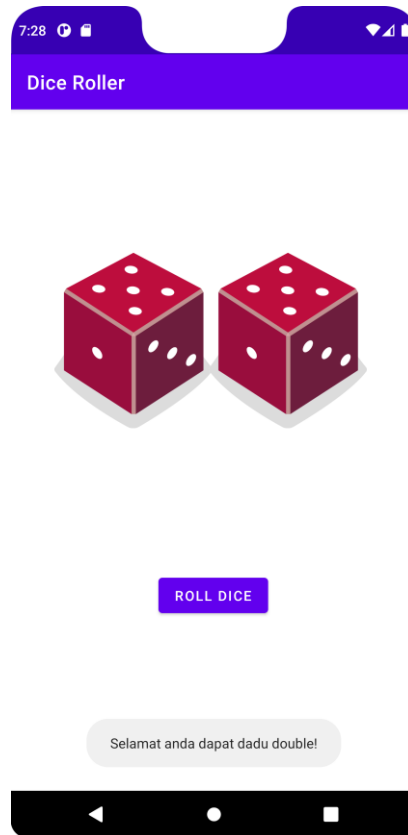
B. Output Program



Gambar 1. Tampilan Awal Aplikasi



Gambar 2. Tampilan Dadu Setelah di roll



Gambar 3. Tampilan Roll Dadu Double

C. Pembahasan

File .xml

Diawali baris [1] menunjukkan bahwa jenis file adalah XML.

Tag `ConstrainLayout` baris [2] setelah itu `xmlns` adalah singkatan dari namespace XML, dan setiap baris mendefinisikan skema atau kosakata untuk atribut yang terkait dengan kata-kata tersebut. Namespace `android:`, misalnya, menandai atribut yang ditentukan oleh sistem Android. Semua atribut dalam XML tata letak dimulai dengan salah satu namespace tersebut.

Pada baris [9] terdapat tag `<button>`. Atribut `android:id` gunanya untuk menginisiasi id pada button tsb. Pada baris [11] dan [12] terdapat atribut untuk mengatur lebar dan tinggi pada widget. Pada baris [13] `android:text` untuk menambahkan text pada widget.

Pada baris [21] terdapat tag `<imageView>` untuk `imageView1`. Atribut `android:id` gunanya untuk menginisiasi id pada button tsb. Pada baris [23] dan [24] terdapat atribut untuk mengatur lebar dan tinggi pada `imageView1`.

Pada baris [33] terdapat tag `<imageView>` untuk `imageView2`. Atribut `android:id` gunanya untuk menginisiasi id pada button tsb. Pada baris [35] dan [36] terdapat atribut untuk mengatur lebar dan tinggi pada `imageView2`.

File .kt

Pertama untuk kode kotlin class MainActivity didefinisikan dengan kata kunci class. Sistem android akan memanggil metode onCreate() dari MainActivity. Metode findViewById() menemukan Button dalam tata letak. R.id.button adalah ID resource untuk Button yang merupakan ID unik untuknya. Kode menyimpan reference ke objek Button dalam variabel yang disebut rollButton, bukan objek Button itu sendiri.

Pada baris [23] fungsi rollDice untuk menampilkan hasil pelemparan dadu. Pada baris [24] val dice1 = Dice (6) untuk membuat dadu dengan 6 sisi. Setelah itu memanggil metode roll() lalu hasilnya disimpan pada variabel diceRoll1 pada baris[25]. Pada baris [26] val diceImage1: ImageView = findViewById(R.id.imageView1) berfungsi untuk menampilkan gambar imageView1.

Pada baris [28] val dice2 = dice (6) untuk membuat dadu dengan 6 sisi. Setelah itu memanggil metode roll() lalu hasilnya disimpan pada variabel diceRoll2 pada baris[29]. Pada baris [30] val diceImage1: ImageView = findViewById(R.id.imageView2) berfungsi untuk menampilkan gambar imageView2.

Pada baris [32] menggunakan logika If else, jika diceRoll1 == diceRoll2 maka output akan mengeluarkan “salamat”. Dan jika tidak maka output akan menghasilkan “maaf anda belum beruntung”.

Pada baris ke [40] pada dadu1 menggunakan logika when, jika user mendapat angka 1, maka tampilkan gambar dice_1. Jika user mendapat angka 2, maka tampilkan gambar dice_2. Jika user mendapat angka 3, maka tampilkan gambar dice_3. Jika user mendapatkan angka 4, maka tampilkan gambar dice_4. Jika user dapat angka 5, maka tampilkan gambar dice_5. Else tampilkan gambar dice_6.

Pada baris ke [48] pada dadu2 menggunakan logika when, jika user mendapat angka 1, maka tampilkan gambar dice_1. Jika user mendapat angka 2, maka tampilkan gambar dice_2. Jika user mendapat angka 3, maka tampilkan gambar dice_3. Jika user mendapatkan angka 4, maka tampilkan gambar dice_4. Jika user dapat angka 5, maka tampilkan gambar dice_5. Else tampilkan gambar dice_6.

Pada baris [56] kita memanggil diceImage1.setImageResource(drawableResource1) untuk menyimpan ID resource yang digunakan.

Pada baris [57] diceImage1.contentDescription = diceRoll.toString() kita mendeskripsikan konten sesuai pada imageView.

Pada baris [59] kita memanggil diceImage2.setImageResource(drawableResource2) untuk menyimpan ID resource yang digunakan.

Pada baris [60] `diceImage2.contentDescription = diceRoll.toString()` kita mendeskripsikan konten sesuai pada `imageView`,

Pada baris [64] membuat class `Dice` dengan menggunakan metode `roll()`. `numSides` sebagai `private` akan membuatnya hanya dapat diakses dalam class `Dice`. Karena satu-satunya kode yang akan menggunakan `numSides` ada di dalam class `Dice`, tidak masalah untuk membuat argumen ini `private` untuk class `Dice`

D. Tautan Git

Berikut adalah tautan untuk source code yang telah dibuat.

<https://github.com/Firdaaulia15/Praktikummobile2/tree/main/Modul2>