

**LAPORAN PRAKTIKUM
PEMROGRAMAN MOBILE
MODUL II**



ANDROID BASICS IN KOTLIN

Oleh:

AULIA LITA

NIM. 2010817120003

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

LEMBAR PENGESAHAN
LAPORAN PRAKTIKUM PEMROGRAMAN MOBILE
MODUL II

Laporan Praktikum Pemrograman Mobile Modul II : Android Basics in Kotlin ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman Mobile. Laporan Praktikum ini dikerjakan oleh:

Nama Praktikan : Aulia Lita
NIM : 2010817120003

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	3
DAFTAR TABEL	3
SOAL	4
A. Source Code	6
B. Output Program	10
C. Pembahasan	12
D. Tautan Git	12

DAFTAR GAMBAR

Gambar 1 Tampilan awal Aplikasi	4
Gambar 2 Tampilan Aplikasi saat Nilai 2 dadu berbeda	5
Gambar 3 Tampilan Aplikasi saat Nilai 2 dadu sama	6
Gambar 4. Hasil Run Aplikasi	10
Gambar 5 Hasil Run Aplikasi saat Nilai 2 dadu berbeda	11
Gambar 6 Hasil Run Aplikasi saat Nilai 2 dadu sama	11

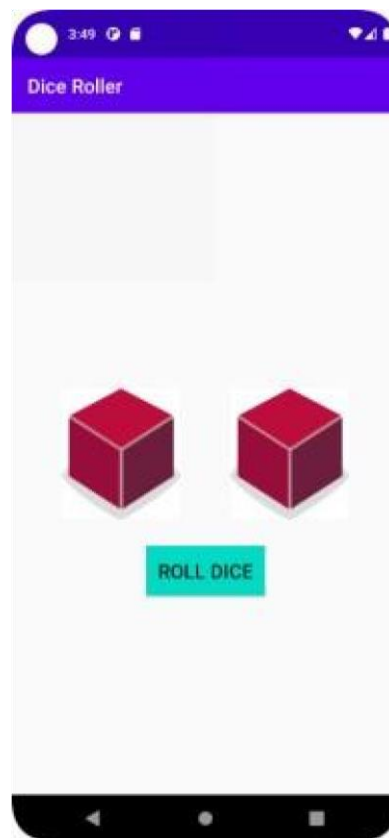
DAFTAR TABEL

Tabel 1. Source Code Kotlin	8
Tabel 2. Source Code xml	9

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 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 Aplikasi saat Nilai 2 dadu berbeda

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.
5. Untuk gambar dadu dapat didownload pada link berikut:
https://drive.google.com/u/0/uc?id=147HT2IIH5qin3z5ta7H9y2N_5OMW81Ll&export=download



Gambar 3 Tampilan Aplikasi saat Nilai 2 dadu sama

A. Source Code

```
1  
2 package com.example.diceroller  
3  
4 import android.os.Bundle  
5 import android.widget.Button  
6 import android.widget.ImageView  
7 import  
8 androidx.appcompat.app.AppCompatActivity  
9 import android.widget.Toast  
10  
11 class MainActivity : AppCompatActivity() {
```

```

12         override fun onCreate(savedInstanceState:
13 Bundle?) {
14             super.onCreate(savedInstanceState)
15             setContentView(R.layout.activity_main)
16
17
18             val rollButton: Button =
19 findViewById(R.id.button)
20             rollButton.setOnClickListener {
21 rollDice() }
22             dadu0()
23         }
24
25
26         private fun rollDice() {
27
28             val dice = Dice(6)
29             val diceRoll = dice.roll()
30
31
32             val diceImage: ImageView =
33 findViewById(R.id.imageView)
34             val drawableResource = when (diceRoll)
35 {
36                 1 -> R.drawable.dice_1
37                 2 -> R.drawable.dice_2
38                 3 -> R.drawable.dice_3
39                 4 -> R.drawable.dice_4
40                 5 -> R.drawable.dice_5
41                 else -> R.drawable.dice_6
42             }
43
44
45
46 diceImage.setImageResource(drawableResource)
47             diceImage.contentDescription =
48 diceRoll.toString()
49
50             //Roll baru
51             val dice2 = Dice(6)
52             val diceRoll2 = dice2.roll()
53
54
55             val diceImage2: ImageView =
56 findViewById(R.id.imageView2)
57

```

```

58         val         drawableResource2         =         when
59         (diceRoll2) {
60             1 -> R.drawable.dice_1
61             2 -> R.drawable.dice_2
62             3 -> R.drawable.dice_3
63             4 -> R.drawable.dice_4
64             5 -> R.drawable.dice_5
65             else -> R.drawable.dice_6
66         }
67
68
69
70         diceImage2.setImageResource(drawableResource2)
71         diceImage2.contentDescription         =
72         diceRoll2.toString()
73
74         if(diceRoll == diceRoll2) {
75             Toast.makeText(this, "Selamat anda
76 dapat         dadu         double!",
77 Toast.LENGTH_SHORT).show();
78         }
79
80         else if (diceRoll != diceRoll2)
81         {
82             Toast.makeText(this, "Anda belum
83 beruntung!", Toast.LENGTH_SHORT).show();
84         }
85     }
86     fun dadu0() {
87         val         diceImage3:         ImageView         =
88         findViewById(R.id.imageView)
89         val         diceImage4:         ImageView         =
90         findViewById(R.id.imageView2)
91
92         diceImage3.setImageResource(R.drawable.dice_0)
93
94         diceImage4.setImageResource(R.drawable.dice_0)
95
96
97         R.drawable.dice_0
98         print(R.drawable.dice_0)
99     }
100 }
101
102
103 class Dice(private val numSides: Int) {
104

```


	fun roll(): Int {
	return (1..numSides).random()
105	}
106	}
107	
108	
109	
110	

Tabel 1 Source Code Kotlin

1	<?xml version="1.0" encoding="utf-8"?>
2	
3	<androidx.constraintlayout.widget.ConstraintLayout
4	
5	xmlns:android="http://schemas.android.com/apk/res/android"
6	xmlns:app="http://schemas.android.com/apk/res-auto"
7	xmlns:tools="http://schemas.android.com/tools"
8	android:layout_width="match_parent"
9	android:layout_height="match_parent"
10	tools:context=".MainActivity">
11	
12	
13	<ImageView
14	android:id="@+id/imageView"
15	android:layout_width="160dp"
16	android:layout_height="200dp"
17	app:layout_constraintBottom_toBottomOf="parent"
18	app:layout_constraintEnd_toEndOf="parent"
19	app:layout_constraintHorizontal_bias="0.119"
20	app:layout_constraintStart_toStartOf="parent"
21	app:layout_constraintTop_toTopOf="parent"
22	app:layout_constraintVertical_bias="0.474"
23	tools:srcCompat="@drawable/dice_1" />
24	
25	<ImageView
26	android:id="@+id/imageView2"
27	android:layout_width="160dp"
28	android:layout_height="200dp"
29	app:layout_constraintBottom_toBottomOf="parent"
30	app:layout_constraintEnd_toEndOf="parent"
31	app:layout_constraintHorizontal_bias="0.86"
32	app:layout_constraintStart_toStartOf="parent"
33	app:layout_constraintTop_toTopOf="parent"
34	app:layout_constraintVertical_bias="0.474"
35	tools:srcCompat="@drawable/dice_1" />
36	

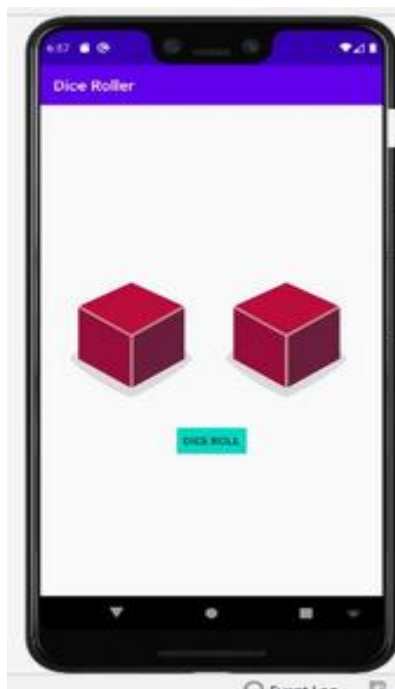
```

37     <Button
38         android:backgroundTint="@color/colorAccent"
39         android:id="@+id/button"
40         android:layout_width="wrap_content"
41         android:layout_height="wrap_content"
42         android:layout_marginTop="16dp"
43         android:text="DICE ROLL"
44         app:layout_constraintEnd_toEndOf="parent"
45         app:layout_constraintStart_toStartOf="parent"
46
47 app:layout_constraintTop_toBottomOf="@+id/imageView" />
48
49
50
51 </androidx.constraintlayout.widget.ConstraintLayout>

```

Tabel 2 Source Code Xml

B. OUTPUT PROGRAM



Gambar 4 Hasil Run Aplikasi



Gambar 5 Hasil Run Aplikasi saat Nilai 2 dadu berbeda



Gambar 4 Hasil Run Aplikasi saat Nilai 2 dadu sama

C. PEMBAHASAN

Baris [1] merupakan sintaks awal program merupakan sintaks yang digunakan untuk mengawali program, lalu baris ke [2] hingga [9] merupakan sintaks yang berguna untuk mengoperasikan program-program yang akan dibuat kedalam program, lalu kita buat class nya untuk menampilkan output yang dibuat dan class dice digunakan untuk membuat angka random pada dadu, lalu kita buat variabel yang sesuai dengan apa yang diinginkan dan buatlah variabel button roll yang berguna untuk mendeklarasikan tombol roll pada program, findViewById digunakan untuk merujuk pada image mana program itu. fungsi rollDice() dapat dijalankan dengan menggunakan setOnClickListener, apabila menekan tombol button. setImageResource digunakan untuk mereset imageView agar dadu kosong. dan sintaks ini drawable untuk mengacak dadu, seperti 1 -> R.drawable.dice_1 yang apabila dadu 1 muncul maka akan menampilkan pula gambar dadu 1 yang telah tersimpan sebelumnya, dan apabila diacak dan apabila mendapatkan fungsi yang sama maka output yang ditampilkan adalah “Selamat anda mendapatkan dadu double”, dan apabila tidak maka akan menampilkan output “Anda belum beruntung”, selanjutnya untuk menampilkan dadu awal kosong adalah dengan membuat fungsi baru serta buatlah variabel baru dan inisiasi beberapa fungsi serta cetak lah image dadu kosong sesuai nama dari image tersebut

Main xml hanya berfungsi untuk mengatur bagaimana tinggi lebar serta warna dari apa yang ditampilkan, misalnya imageView untuk mengatur ketentuan image, lalu button yang akan menampilkan sebuah button yang digunakan untuk mengacak dadu, serta kita atur warna dan tingginya agar sesuai dengan apa yang diinginkan.

D. TAUTAN GIT

Berikut adalah tautan untuk source code yang telah dibuat.

<https://github.com/AuliaLita/praktikummobile2/tree/main/MODUL2>