

**LAPORAN PRAKTIKUM
PEMROGRAMAN MOBILE
MODUL 1**



ANDROID BASIC IN KOTLIN

Oleh:

Regina Silva M

NIM. 2310817220011

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS TEKNIK
UNIVERSITAS LAMBUNG MANGKURAT
APRIL 2024**

LEMBAR PENGESAHAN
LAPORAN PRAKTIKUM PEMROGRAMAN I
MODUL 1

Laporan Praktikum Pemrograman Mobile Modul 1: Android Basic in Kotlin ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman Mobile. Laporan Praktikum ini dikerjakan oleh:

Nama Praktikan : Regina Silva Maharatini
NIM : 2310817220011

Menyetujui,
Asisten Praktikum

Mengetahui,
Dosen Penanggung Jawab Praktikum

Zulfa Auliya Akbar
NIM. 2210817210026

Muti`a Maulida S.Kom M.T.I
NIP. 19881027 201903 20 13

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	11
D. Tautan Git	13

DAFTAR GAMBAR

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

DAFTAR TABEL

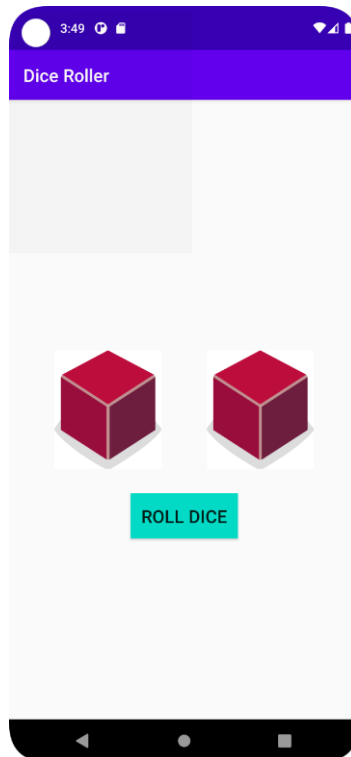
Tabel 1. Source Code Jawaban Soal 1.....	9
Tabel 2. Source Code Jawaban Soal 1.....	10
Tabel 3. Source Code Jawaban Soal 1.....	11

SOAL 1

Soal Praktikum:

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.



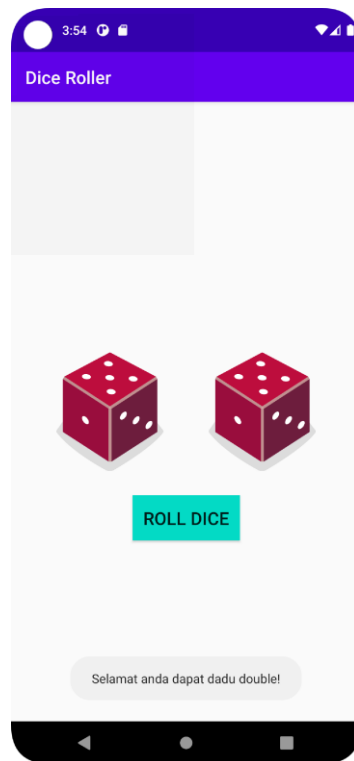
Gambar 1 Tampilan Awal Aplikasi

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 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.
5. Untuk gambar dadu dapat didownload pada link berikut:
https://drive.google.com/u/0/uc?id=147HT2IIH5qin3z5ta7H9y2N_5OMW81LI&export=download



Gambar 3 Tampilan Roll Dadu Double

A. Source Code

1. MainActivity.kt

```

1 package com.example.daduapp
2
3 import android.os.Bundle
4 import android.widget.Toast
5 import androidx.activity.viewModels
6 import androidx.appcompat.app.AppCompatActivity
7 import com.example.daduapp.databinding.ActivityMainBinding
8
9 class MainActivity : AppCompatActivity() {
10
11     private lateinit var binding: ActivityMainBinding
12
13     private val viewModel: DiceViewModel by viewModels()
14
15     override fun onCreate(savedInstanceState: Bundle?) {
16         super.onCreate(savedInstanceState)
17
18         binding = ActivityMainBinding.inflate(layoutInflater)
19         setContentView(binding.root)
20
21         updateDiceImages(viewModel.dice1, viewModel.dice2)
22     }

```


23	binding.rollbuton.setOnClickListener	{
24	viewModel.dice1	= Dice(6).roll()
25	viewModel.dice2	= Dice(6).roll()
26		
27	updateDiceImages(viewModel.dice1, viewModel.dice2)	
28	}	
29	}	
30		
31	private fun updateDiceImages(dice1: Int, dice2: Int) {	
32		
33	binding.diceImage1.setImageResource(getDiceDrawable(dice1))	
34		
35	binding.diceImage2.setImageResource(getDiceDrawable(dice2))	
36		
37	if (dice1 == dice2 && dice1 != 0) {	
38	Toast.makeText(this, "Selamat anda dapat dadu	
39	double!", Toast.LENGTH_SHORT).show()	
40	} else if (dice1 != 0 && dice2 != 0) {	
41	Toast.makeText(this, "Anda belum beruntung!",	
42	Toast.LENGTH_SHORT).show()	
43	}	
44	}	
45		
46	private fun getDiceDrawable(roll: Int): Int {	
47	return when (roll) {	
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	6 -> R.drawable.dice_6	
54	else -> R.drawable.dice_0	
55	}	
56	}	
57	}	
58		
59	class Dice(private val numSides: Int) {	
60	fun roll(): Int = (1..numSides).random()	
61	}	
62		

Tabel 1. Source Code Jawaban Soal 1

2. DaduView.kt

1	package	com.example.daduapp
2		
3	import	androidx.lifecycle.ViewModel
4		
5	class	DiceViewModel : ViewModel() {
6		

7	var	dicel:	Int	=	0
8	var	dice2:	Int	=	0
9	}				
10					

Tabel 2. Source Code Jawaban Soal 1

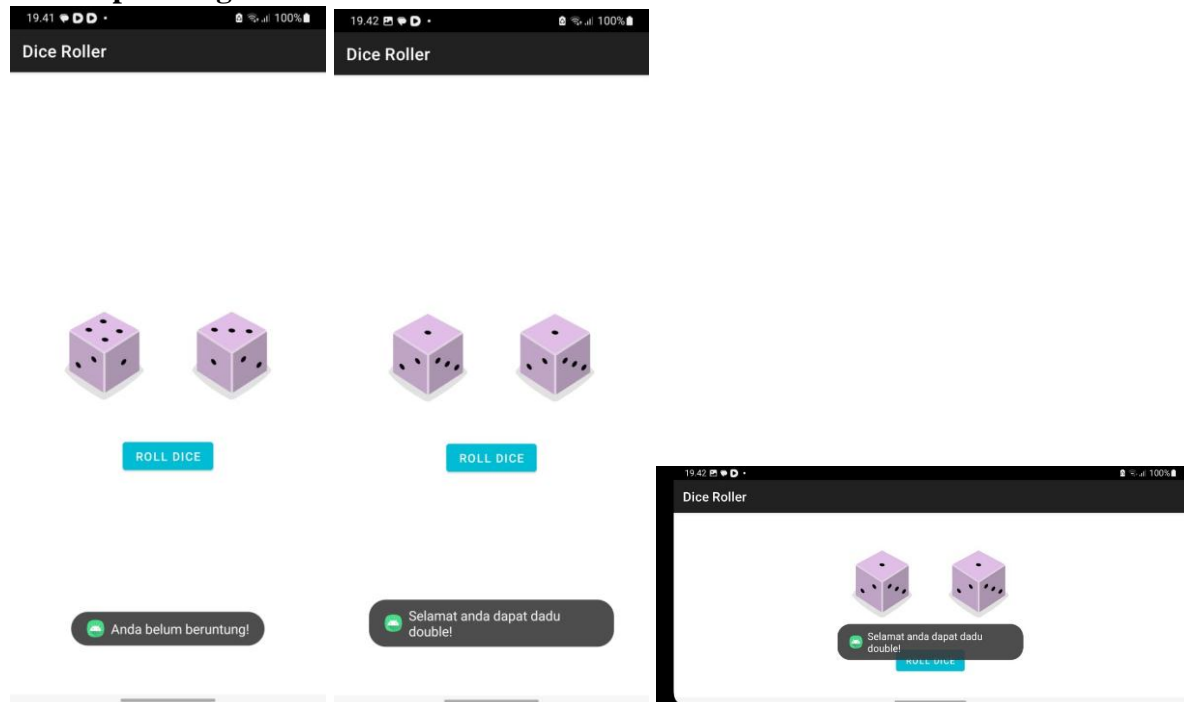
3. activity_main.xml

1	<?xml version="1.0" encoding="utf-8"?>				
2	<LinearLayout				
3	xmlns:android="http://schemas.android.com/apk/res/android"				
4	xmlns:tools="http://schemas.android.com/tools"				
5	android:id="@+id/mainLayout"				
6	android:layout_width="match_parent"				
7	android:layout_height="match_parent"				
8	android:orientation="vertical"				
9	android:gravity="center"				
10	android:background="#FFFFFF"				
11	android:padding="16dp"				
12	tools:context=".MainActivity">				
13					
14					
15	<LinearLayout				
16	android:layout_width="wrap_content"				
17	android:layout_height="wrap_content"				
18	android:orientation="horizontal"				
19	android:gravity="center">				
20					
21					
22	<ImageView				
23	android:id="@+id/dice_image1"				
24	android:layout_width="150dp"				
25	android:layout_height="150dp"				
26	android:src="@drawable/dice_0"				
27	android:layout_margin="8dp"/>				
28					
29					
30	<ImageView				
31	android:id="@+id/dice_image2"				
32	android:layout_width="150dp"				
33	android:layout_height="150dp"				
34	android:src="@drawable/dice_0"				
35	android:layout_margin="8dp"/>				
36	</LinearLayout>				
37					
38	<Button				
39	android:id="@+id/rollbuton"				
40	android:layout_width="wrap_content"				
41	android:layout_height="wrap_content"				
42	android:text="ROLL				
43	android:textColor="#FFFFFF" DICE"				
44					

45	<code>android:backgroundTint="#00BCD4"</code>
46	<code>android:layout_marginTop="24dp"/></code>
47	<code></LinearLayout></code>
48	

Tabel 3. Source Code Jawaban Soal 1

B. Output Program



Gambar 1. Screenshot Hasil Jawaban Soal 1

C. Pembahasan

1. MainActivity.kt:

Variabel binding digunakan untuk menginisialisasi ViewBinding dari layout `activity_main.xml`, sehingga seluruh elemen UI dapat diakses langsung tanpa menggunakan `findViewById`. Objek `viewModel` merupakan instance dari kelas `DiceViewModel`, yang digunakan untuk menyimpan nilai dua dadu (`dice1` dan `dice2`) agar data tidak hilang saat terjadi perubahan konfigurasi, seperti rotasi layar. Pada fungsi `onCreate`, layout dihubungkan dengan aktivitas menggunakan `setContentViews(binding.root)`, dan fungsi `updateDiceImages()` dipanggil untuk menampilkan nilai dadu terakhir yang disimpan di `ViewModel`. Saat tombol `ROLL DICE` ditekan, dua nilai acak dihasilkan dari objek `Dice`, lalu disimpan ke dalam `viewModel.dice1` dan `viewModel.dice2`. Fungsi `updateDiceImages()`

digunakan untuk menampilkan gambar dadu berdasarkan hasil roll, serta menampilkan pesan melalui `Toast`. Jika kedua nilai dadu sama dan bukan nol, akan ditampilkan pesan “Selamat anda dapat dadu double!”, sedangkan jika berbeda, pesan yang muncul adalah “Anda belum beruntung!”. Fungsi `getDiceDrawable()` bertugas mengembalikan resource gambar dadu (`dice_1` hingga `dice_6`) sesuai angka hasil lemparan. Kelas `Dice` sendiri digunakan untuk menghasilkan angka acak dari 1 sampai jumlah sisi dadu, dengan default 6 sisi.

2. DaduView.kt

Kelas `DiceViewModel` merupakan turunan dari kelas `ViewModel`, yang digunakan untuk menyimpan data agar tetap bertahan saat terjadi perubahan konfigurasi, seperti rotasi layar. Di dalamnya terdapat dua variabel, yaitu `dice1` dan `dice2`, yang masing-masing menyimpan nilai lemparan dadu pertama dan kedua. Dengan menyimpan nilai ini di dalam `ViewModel`, aplikasi tetap dapat menampilkan hasil dadu terakhir tanpa ter-reset meskipun activity dibuat ulang.

3. activity_main.xml

Layout menggunakan elemen `LinearLayout` sebagai container utama dengan orientasi vertikal dan posisi elemen ditengah (`gravity="center"`), serta latar belakang putih dan padding sebesar 16dp. Di dalamnya terdapat `LinearLayout` horizontal yang berfungsi untuk menampilkan dua gambar dadu secara sejajar. Setiap dadu ditampilkan menggunakan elemen `ImageView` dengan ukuran 150dp x 150dp dan jarak antar gambar diatur melalui atribut `layout_margin`. Gambar awal dadu diatur menggunakan `@drawable/dice_0`, yang merupakan gambar dadu kosong sebagai tampilan default. Di bawah gambar dadu terdapat elemen `Button` dengan ID `rollbutton`, berfungsi untuk memicu aksi "ROLL DICE". Tombol ini memiliki teks berwarna putih dan latar belakang berwarna biru kehijauan (`#00BCD4`), serta jarak atas sebesar 24dp agar tampilan lebih rapi.

D. Tautan Git

Berikut adalah tautan untuk source code yang telah dibuat.

<https://github.com/rgnsilvas/Pemrograman-Mobile/tree/master/MODUL1>