

**LAPORAN PRAKTIKUM
PEMROGRAMAN MOBILE
MODUL 1**



ANDROID BASICS IN KOTLIN

Oleh:

Randy Febrian

NIM. 2310817110013

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS TEKNIK
UNIVERSITAS LAMBUNG MANGKURAT
APRIL 2025**

LEMBAR PENGESAHAN
LAPORAN PRAKTIKUM PEMROGRAMAN MOBILE
MODUL 1

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

Nama Praktikan : Randy Febrian
NIM : 2310817110013

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.....	10
C. Pembahasan.....	13
SOAL 2.....	Error! Bookmark not defined.
A. Source Code	Error! Bookmark not defined.
B. Output Program.....	Error! Bookmark not defined.
C. Pembahasan.....	Error! Bookmark not defined.
SOAL 3.....	Error! Bookmark not defined.
A. Source Code	Error! Bookmark not defined.
B. Output Program.....	Error! Bookmark not defined.
C. Pembahasan.....	Error! Bookmark not defined.
SOAL 4.....	Error! Bookmark not defined.
A. Source Code	Error! Bookmark not defined.
B. Output Program.....	Error! Bookmark not defined.
C. Pembahasan.....	Error! Bookmark not defined.
SOAL 5.....	Error! Bookmark not defined.
A. Source Code	Error! Bookmark not defined.
B. Output Program.....	Error! Bookmark not defined.
C. Pembahasan.....	Error! Bookmark not defined.
TAUTAN GIT	15

DAFTAR GAMBAR

Gambar 1.1. Tampilan Awal Aplikasi.....	6
Gambar 1.2. Tampilan Dadu Setelah di Roll.....	7
Gambar 1.3. Tampilan Roll Dadu Double.....	7
Gambar 1.4. Screenshot Hasil Soal 1	12

DAFTAR TABEL

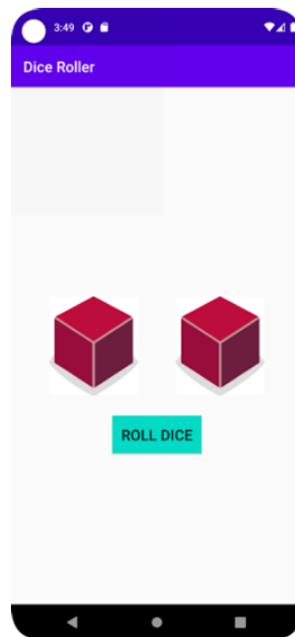
Tabel 1. Jawaban Soal 1 MainActivity.kt.....	8
Tabel 2. Jawaban Soal 1 DiceViewModel.kt	10
Tabel 3. Jawaban Soal 1 activity_main.xml	11

SOAL 1

Soal Praktikum

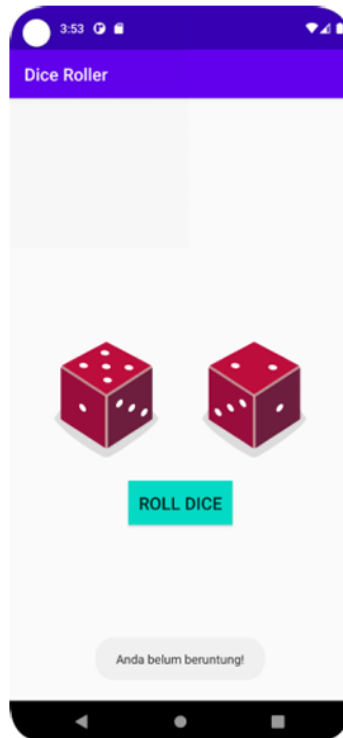
Buatlah sebuah aplikasi yang dapat menampilkan 2 (dua) buah dadu yang dapat berubah ubah tampilannya pada saat user menekan tombol “Roll Dice”. Aturan aplikasi yang akan dibangun adalah sebagaimana berikut:

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



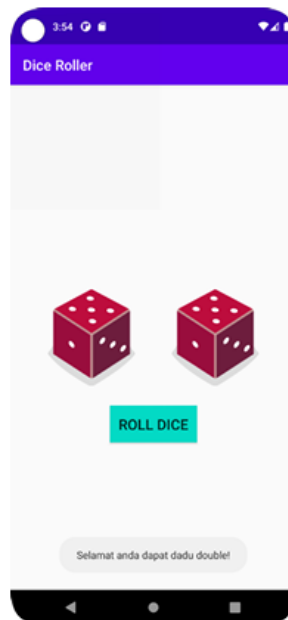
Gambar 1. Tampilan Awal Aplikasi

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 nilaidouble, maka aplikasi akan menampilkan pesan “Selamat anda dapat dadu double!” seperti dapat dilihat pada Gambar 3.



Gambar 3. Tampilan Roll Dadu Double

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/file/d/14V3qXGdFnuoYN4AGd_9SgFh8kw8X9ySm/view

A. Source Code

Tabel 1. Jawaban Soal 1 MainActivity.kt

```

1 package com.example.rolldice
2
3 import android.os.Bundle
4 import android.widget.Toast
5 import androidx.activity.viewModels
6 import androidx.appcompat.app.AppCompatActivity
7 import androidx.lifecycle.Observer
8 import
9 com.example.rolldice.databinding.ActivityMainBinding
10
11 class MainActivity : AppCompatActivity() {
12     private lateinit var binding: ActivityMainBinding
13
14     private val diceViewModel: DiceViewModel by
15     viewModels()
16
17     override fun onCreate(savedInstanceState: Bundle?)
18 {
19         super.onCreate(savedInstanceState)
20
21         binding =
22 ActivityMainBinding.inflate(layoutInflater)
23         setContentView(binding.root)
24
25         binding.btRoll.setOnClickListener {
26 diceViewModel.rollDice() }
27
28         diceViewModel.dice1Result.observe(this,
29 Observer { result ->
30             val diceImageResource = when(result) {
31                 1 -> R.drawable.dice_1
32                 2 -> R.drawable.dice_2
33                 3 -> R.drawable.dice_3
34                 4 -> R.drawable.dice_4
35                 5 -> R.drawable.dice_5
36                 6 -> R.drawable.dice_6
37                 else -> R.drawable.dice_1

```



```

38         }
39
40 binding.ivDadu1.setImageResource(diceImageResource)
41     })
42
43     diceViewModel.dice2Result.observe(this,
44 Observer { result ->
45         val diceImageResource = when(result) {
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             6 -> R.drawable.dice_6
52             else -> R.drawable.dice_1
53         }
54
55 binding.ivDadu2.setImageResource(diceImageResource)
56     })
57
58     diceViewModel.isDouble.observe(this, Observer
59 { isDouble ->
60         val message = if (isDouble) {
61             "Selamat anda dapat dadu double!"
62         } else {
63             "Anda belum beruntung!"
64         }
65         Toast.makeText(applicationContext,
66 message, Toast.LENGTH_SHORT).show()
67     })
68 }
69 }

```

Tabel 2. Jawaban Soal 1 DiceViewModel.kt

```
1 package com.example.rolldice
2
3 import androidx.lifecycle.LiveData
4 import androidx.lifecycle.MutableLiveData
5 import androidx.lifecycle.ViewModel
6
7 class DiceViewModel : ViewModel() {
8     private val _dice1Result = MutableLiveData<Int>()
9     val dice1Result: LiveData<Int> = _dice1Result
10
11     private val _dice2Result = MutableLiveData<Int>()
12     val dice2Result: LiveData<Int> = _dice2Result
13
14     private val _isDouble = MutableLiveData<Boolean>()
15     val isDouble: LiveData<Boolean> = _isDouble
16
17     class Dice(private val jumlahSisi: Int) {
18         fun roll(): Int {
19             return (1..jumlahSisi).random()
20         }
21     }
22
23     fun rollDice() {
24         val dice1 = Dice(6)
25         val dice2 = Dice(6)
26
27         val dice1Roll = dice1.roll()
28         val dice2Roll = dice2.roll()
29
30         _dice1Result.value = dice1Roll
31         _dice2Result.value = dice2Roll
32         _isDouble.value = dice1Roll == dice2Roll
33     }
34 }
```

Tabel 3. Jawaban Soal 1 activity_main.xml

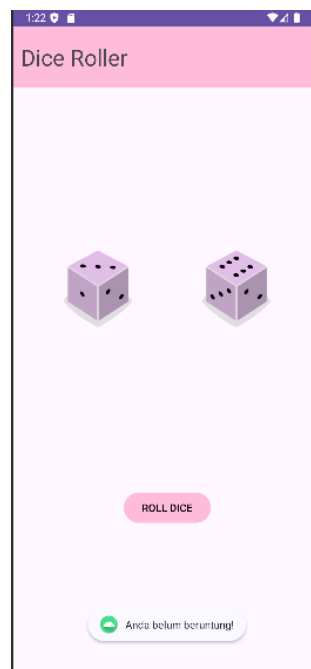
1	<?xml	version="1.0"	encoding="utf-8"?>
2	<androidx.constraintlayout.widget.ConstraintLayout		
3	xmlns:android="http://schemas.android.com/apk/res/andro		
4	id"		
5	xmlns:app="http://schemas.android.com/apk/res-auto"		
6	xmlns:tools="http://schemas.android.com/tools"		
7	android:id="@+id/main"		
8	android:layout_width="match_parent"		
9	android:layout_height="match_parent"		
10	tools:context=".MainActivity">		
11			
12	<TextView		
13	android:layout_width="match_parent"		
14	android:layout_height="wrap_content"		
15	app:layout_constraintTop_toTopOf="parent"		
16	app:layout_constraintStart_toStartOf="parent"		
17	android:text="Dice	Roller"	
18	android:textSize="30sp"		
19	android:textColor="#000000"		
20	android:background="#FFBCDA"		
21	android:paddingVertical="20dp"		
22	android:paddingStart="10dp"		
23	/>		
24			
25	<ImageView		
26	android:id="@+id/ivDadu1"		
27	android:layout_width="144dp"		
28	android:layout_height="133dp"		
29	app:layout_constraintBottom_toBottomOf="parent"		
30	app:layout_constraintStart_toStartOf="parent"		
31	app:layout_constraintTop_toTopOf="parent"		
32	app:srcCompat="@drawable/dice_0"		
33	android:layout_marginStart="30dp"/>		
34			
35	<ImageView		
36	android:id="@+id/ivDadu2"		
37	android:layout_width="147dp"		
38	android:layout_height="134dp"		
39	app:layout_constraintBottom_toBottomOf="parent"		
40	app:layout_constraintEnd_toEndOf="parent"		
41	app:layout_constraintTop_toTopOf="parent"		
42	app:srcCompat="@drawable/dice_0"		
43	android:layout_marginEnd="30dp"/>		
44			
45	<Button		
46	android:id="@+id/btRoll"		

```

47         android:layout_width="wrap_content"
48         android:layout_height="wrap_content"
49         android:text="ROLL DICE"
50         app:layout_constraintEnd_toEndOf="parent"
51         app:layout_constraintStart_toStartOf="parent"
52
53     app:layout_constraintTop_toBottomOf="@+id/ivDadu2"
54         android:backgroundTint="#00FFFF"
55         android:textColor="@color/black" />
56
57 </androidx.constraintlayout.widget.ConstraintLayout>

```

B. Output Program



Gambar 4. Screenshot Hasil Soal 1

C. Pembahasan

MainActivity.kt

Baris [1] berfungsi untuk menyatakan bahwa file MainActivity.kt berada dalam package com.example.rolldice. Baris [3 – 12] digunakan untuk mengimpor berbagai library penting, seperti komponen ViewModel (viewModels, ViewModel, LiveData, MutableLiveData), View Binding (ActivityMainBinding), serta komponen UI seperti Toast. Baris [14 – 18] mendefinisikan kelas MainActivity yang merupakan turunan dari AppCompatActivity. Di dalamnya terdapat dua properti penting, yaitu binding yang digunakan untuk menghubungkan layout ke kode menggunakan View Binding, dan diceViewModel yang dideklarasikan menggunakan by viewModels() untuk mengakses ViewModel sesuai siklus hidup activity.

Baris [21 – 23] menginisialisasi binding menggunakan layout inflater dan menampilkan tampilan utama dengan setContentView(binding.root). Baris [25 - 26] memberikan aksi pada tombol btRoll agar saat ditekan, fungsi rollDice() dari ViewModel dijalankan. Baris [28 – 41] mengamati perubahan nilai pada dice1Result. Jika terjadi perubahan, maka akan dipilih gambar dadu yang sesuai berdasarkan hasil, lalu ditampilkan pada ivDadu1. Hal serupa juga dilakukan pada baris [43 – 56] untuk dadu kedua dan ditampilkan di ivDadu2. Baris [58 – 69] mengamati nilai dari isDouble. Jika nilainya true, maka ditampilkan pesan "Selamat anda dapat dadu double!" menggunakan Toast, sedangkan jika false, akan muncul pesan "Anda belum beruntung!".

DiceViewModel.kt

Baris [1] adalah package dari aplikasi Dice Roller. Baris [3 - 5] adalah import yang diperlukan untuk ViewModel yang digunakan dalam aplikasi, meliputi; LiveData, MutableLiveData dan ViewModel. Baris [7] adalah pendeklarasian Class DiceViewModel dengan tipe Viewmodel(). Baris [9 - 15] berfungsi untuk menyimpan tiga buah objek LiveData, yaitu _dice1Result untuk hasil lemparan dadu pertama, _dice2Result untuk hasil lemparan dadu kedua, dan _isDouble untuk status apakah kedua dadu menunjukkan angka yang sama (double).

Pada baris [17 - 33], di dalam DiceViewModel, terdapat kelas Dice yang menerima jumlah sisi dadu sebagai parameter dan memiliki fungsi roll() untuk menghasilkan angka acak antara 1 hingga jumlah sisi tersebut. Fungsi rollDice() akan membuat dua objek Dice

dengan enam sisi, melempar keduanya menggunakan fungsi roll(), lalu memperbarui nilai _dice1Result, _dice2Result, dan _isDouble berdasarkan hasil lemparan tersebut.

Activity_main.xml

Baris [1 - 10] berfungsi untuk mengatur layout pada aplikasi android. Kemudian baris [12 – 23] berfungsi untuk menampilkan teks Dice Roller dengan warna pink pada bagian atas layer aplikasi, dengan ukuran teks 30 sp dan menggunakan padding atas, kiri, dan bawah untuk memberikan jarak. Kemudian, baris [25 – 43] berfungsi untuk menampilkan gambar dadu kosong dengan size yang sama dengan constraint atas bawah dari parent sehingga berada di tengah, dengan menggunakan chain agar dadu saling terikat bahkan pada saat mode handphone dibuat landscape. Gambar ini nantinya akan diubah secara acak dari kode MainActivity.kt. Baris [32 - 42] adalah tombol atau button yang digunakan untuk roll dadu dengan height dan widthnya disesuaikan dengan teks yang ada dan diletakkan dibawah gambar dadu.

TAUTAN GIT

Berikut adalah tautan untuk source code yang telah dibuat.

https://github.com/rnddfn/Pemrograman_Mobile