

**LAPORAN PRAKTIKUM
PEMROGRAMAN MOBILE
MODUL 1**



ANDROID BASIC WITH KOTLIN

Oleh:

Alya Rosaan NIM. 2310817320006

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

Nama Praktikan : Alya Rosaan
NIM : 2310817320006

Menyetujui,
Asisten Praktikum

Mengetahui,
Dosen Penanggung Jawab Praktikum

Muhammad Raka Azwar
NIM. 2210817210012

Andreyan Rizky Baskara, S.Kom., M.Kom.
NIP. 19930703 201903 01 011

DAFTAR ISI

LEMBAR PENGESAHAN	2
DAFTAR ISI	3
DAFTAR GAMBAR.....	4
DAFTAR TABEL	5
SOAL 1	6
A. Source Code XML.....	8
B. Output Program	10
C. Pembahasan	11
D. Source Code Compose.....	13
E. Output Program	15
F. Pembahasan	15
G. Tautan Git.....	16

DAFTAR GAMBAR

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

DAFTAR TABEL

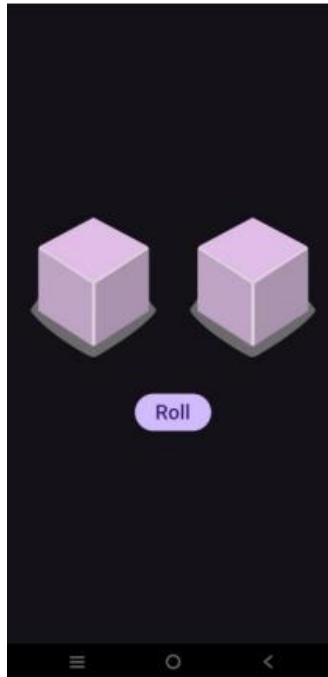
Tabel 1. Source Code Jawaban Soal 1	9
Tabel 2. Source Code Jawaban Soal 1	10

SOAL 1

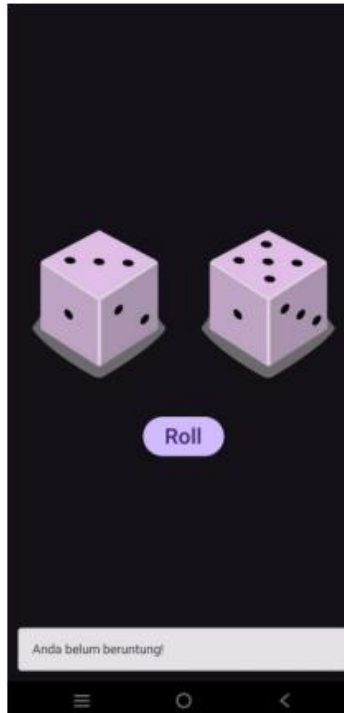
Soal Praktikum:

Buatlah sebuah aplikasi yang dapat menampilkan 2 (dua) buah dadu yang dapat berubah-ubah tampilannya pada saat user menekan tombol “Roll”. 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.



2. Setelah user menekan tombol “Roll” 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.



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. Buatlah aplikasi tersebut menggunakan XML dan Jetpack Compose.

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

A. Source Code XML

MainActivity.kt

```
1 package com.example.mydicerollerapp
2
3 import android.os.Bundle
4 import android.widget.Button
5 import android.widget.ImageView
6 import android.widget.Toast
7 import androidx.appcompat.app.AppCompatActivity
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             rollMyDice()
17         }
18     }
19
20     private fun rollMyDice() {
21         val dice = Dice(6)
22         val dice2 = Dice(6)
23         val diceRoll: Int = dice.roll()
24         val diceRoll2: Int = dice2.roll()
25         val diceImage: ImageView = findViewById(R.id.imageView)
26         val diceImage2: ImageView = findViewById(R.id.imageView2)
27         diceImage.setImageResource(R.drawable.dice_0)
28         diceImage2.setImageResource(R.drawable.dice_0)
29         when (diceRoll) {
30             1 -> diceImage.setImageResource(R.drawable.dice_1)
31             2 -> diceImage.setImageResource(R.drawable.dice_2)
32             3 -> diceImage.setImageResource(R.drawable.dice_3)
33             4 -> diceImage.setImageResource(R.drawable.dice_4)
34             5 -> diceImage.setImageResource(R.drawable.dice_5)
35             else -> diceImage.setImageResource(R.drawable.dice_6)
36         }
37         when (diceRoll2) {
38             1 -> diceImage2.setImageResource(R.drawable.dice_1)
39             2 -> diceImage2.setImageResource(R.drawable.dice_2)
```


40	3 -> diceImage2.setImageResource(R.drawable.dice_3)
41	4 -> diceImage2.setImageResource(R.drawable.dice_4)
42	5 -> diceImage2.setImageResource(R.drawable.dice_5)
43	else -> diceImage2.setImageResource(R.drawable.dice_6)
44	
45	}
46	if (diceRoll == diceRoll2) {
47	Toast.makeText(this, "Selamat, anda dapat dadu
	double!", Toast.LENGTH_SHORT).show()
48	} else {
49	Toast.makeText(this, "Anda belum beruntung!",
	Toast.LENGTH_SHORT).show()
50	}
51	}
52	}
53	class Dice(private val numSides: Int) {
54	fun roll(): Int {
55	return (1..numSides).random()
56	}
57	}

Tabel 1. Source Code Jawaban Soal 1

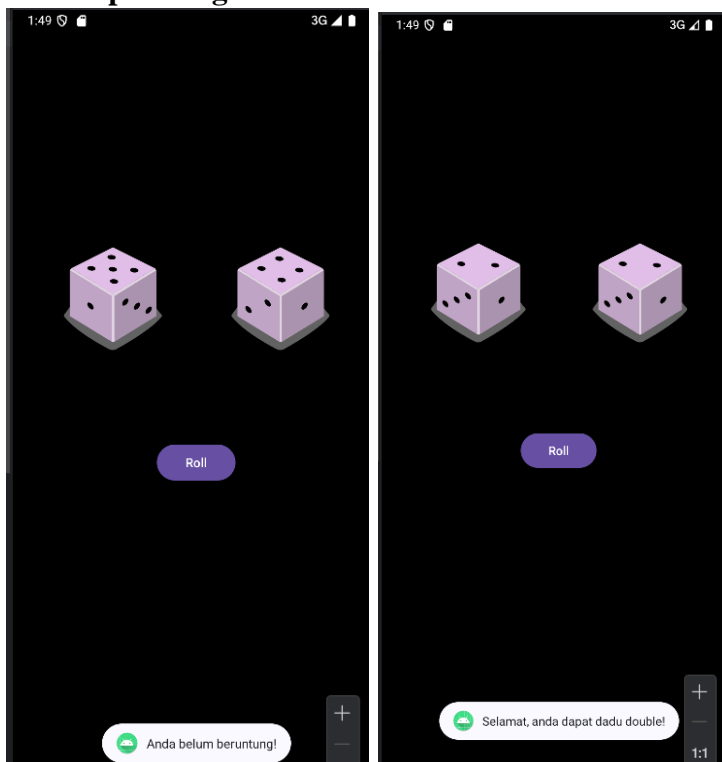
activity_main.xml

1	<?xml version="1.0" encoding="utf-8"?>
2	<androidx.constraintlayout.widget.ConstraintLayout
3	xmlns:android="http://schemas.android.com/apk/res/android"
4	xmlns:app="http://schemas.android.com/apk/res-auto"
5	xmlns:tools="http://schemas.android.com/tools"
6	android:id="@+id/main"
7	android:layout_width="match_parent"
8	android:layout_height="match_parent"
9	android:background="@color/black"
10	tools:context=".MainActivity">
11	
12	<!-- Dadu Kiri -->
13	
14	<!-- Dadu Kanan -->
15	
16	<ImageView
17	android:id="@+id/imageView"
18	android:layout_width="150dp"
19	android:layout_height="150dp"
20	android:layout_marginTop="248dp"
21	app:layout_constraintEnd_toStartOf="@+id/imageView2"
22	app:layout_constraintHorizontal_chainStyle="spread"
23	app:layout_constraintStart_toStartOf="parent"
24	app:layout_constraintTop_toTopOf="parent"
25	app:srcCompat="@drawable/dice_0" />
26	
27	<ImageView

28	android:id="@+id/imageView2"
29	android:layout_width="150dp"
30	android:layout_height="150dp"
31	app:layout_constraintTop_toTopOf="@+id/imageView"
32	app:layout_constraintStart_toEndOf="@+id/imageView"
33	app:layout_constraintEnd_toEndOf="parent"
34	app:srcCompat="@drawable/dice_0" />
35	
36	<!-- Tombol Roll -->
37	<Button
38	android:id="@+id/button"
39	android:layout_width="wrap_content"
40	android:layout_height="wrap_content"
41	android:layout_marginTop="84dp"
42	android:text="@string/roll"
43	app:layout_constraintEnd_toEndOf="parent"
44	app:layout_constraintHorizontal_bias="0.498"
45	app:layout_constraintStart_toStartOf="parent"
46	
47	app:layout_constraintTop_toBottomOf="@+id/imageView" />
48	
49	</androidx.constraintlayout.widget.ConstraintLayout>

Tabel 2. Source Code Jawaban Soal 1

B. Output Program



Gambar 1. Screenshot Hasil Jawaban Soal 1

C. Pembahasan

MainActivity.kt:

1. Pada package com.example.mydicerollerapp, nama package dari aplikasi
2. Pada bagian Import seperti
 - Button = Tombol
 - ImageView = Gambar
 - Toast = Pesan kecil muncul sementara
 - AppCompatActivity = Kelas dasar untuk aktivitas Android
3. Pada bagian class MainActivity : AppCompatActivity(), dideklarasikan kelas utama bernama MainActivity diaman laya pertma aplikasi dijalankan
4. Pada bagian fungsi onCreate() akan dipanggil saat aplikasi dibuka pertama kali,
 - menampilkan tampilan dari file activity_main.xml → setContentView(...), lalu
 - val rollButton: Button = findViewById(R.id.button) menghubungkan tombol dari XML ke kode Kotlin
 - rollButton.setOnClickListener {
rollMyDice() } -> Memberi aksi ke tombol: kalau diklik, maka jalankan fungsi rollMyDice():
5. Pada bagian Fungsi rollMyDice() dipanggil apda saat tombol diklik. Didalamnya terdapat
 - val dice = Dice(6)
val dice2 = Dice(6) -> untuk membuat 2 objek dadu dengan 6 sisi
 - val diceRoll: Int = dice.roll()
val diceRoll2: Int = dice2.roll() -> untuk melakukan lemparan dadu
 - val diceImage: ImageView = findViewById(R.id.imageView)
val diceImage2: ImageView = findViewById(R.id.imageView2) -> untuk mengambil 2 imageView dari layout
 - val diceImage: ImageView = findViewById(R.id.imageView)
val diceImage2: ImageView = findViewById(R.id.imageView2) -> untuk mengeset awal gambar dadu jadi kosong
 - when (diceRoll) {
1 -> diceImage.setImageResource(R.drawable.dice_1) dst...} -> untuk mengganti gambar sesuai angka dadu begitu juga untuk diceImage2
 - Toast.makeText(this, "Selamat, anda dapat dadu double!",
Toast.LENGTH_SHORT).show() -> lalu ini untuk mnegcek apakah angka dadu sama, jika tidak akan menmampilkan pesan anda belum beruntung
6. Pada bagian class Dice terdapat Dice dengan variabel private numSides yang nilainya integer
 - class Dice(private val numSides: Int) {
fun roll(): Int {
return (1..numSides).random()} } yaitu artinya fungsi roll akan mengembalikan hasil random antara 1 dan numSides sampai jumlah sisi dadu

activity_main.xml:

1. pada Baris 1 `<?xml version="1.0" encoding="utf-8"?>` : Penanda awal bahwa file ini adalah file XML dengan encoding UTF-8.
2. Pada Baris 2–10:
 - `<androidx.constraintlayout.widget.ConstraintLayout ->` Membuka layout utama `ConstraintLayout`.
 - `xmlns:android="http://schemas.android.com/apk/res/android"`
`xmlns:app="http://schemas.android.com/apk/res-auto"`
`xmlns:tools="http://schemas.android.com/tools ->` Menyediakan namespace (xmlns).
 - `android:id="@+id/main" ->` ID layout: main.
 - `android:layout_width="match_parent"`
`android:layout_height="match_parent" ->` Ukuran penuh layar (`match_parent`).
 - `android:background="@color/black" ->` Latar belakang hitam.
 - `tools:context=".MainActivity"> ->` Untuk preview di Android Studio (`tools:context`).
3. Pada Baris 12–21 untuk Gambar dadu kiri.
 - `android:id="@+id/imageView" ->` untuk ID: `imageView`.
 - `android:layout_width="150dp"`
`android:layout_height="150dp" ->` Ukuran: 150x150dp.
 - `android:layout_marginTop="248dp" ->` Jarak dari atas: 248dp.
 - `app:layout_constraintEnd_toStartOf="@+id/imageView2"`
`app:layout_constraintHorizontal_chainStyle="spread"`
`app:layout_constraintStart_toStartOf="parent"`
`app:layout_constraintTop_toTopOf="parent" ->` Dikonstraint ke dadu kanan (`imageView2`).
 - `app:srcCompat="@drawable/dice_0" /> ->` Gambar awal: `dice_0` (kosong).
4. Pada Baris 23–30 untuk Gambar dadu kanan.
 - `android:layout_width="150dp"`
`android:layout_height="150dp" ->` ID: `imageView2`.
 - `app:layout_constraintTop_toTopOf="@+id/imageView"`
`app:layout_constraintStart_toEndOf="@+id/imageView" ->` Ukuran: sama seperti kiri.
 - `app:layout_constraintEnd_toEndOf="parent" ->` Diletakkan di sebelah kanan dadu kiri.
 - `app:srcCompat="@drawable/dice_0" /> ->` Gambar awal: `dice_0`.
5. Pada Baris 32–41 untuk Tombol untuk roll dadu
 - `android:id="@+id/button" untuk` ID: `button`.
 - `android:layout_width="wrap_content"`
`android:layout_height="wrap_content" ->` Ukuran mengikuti isi teks.
 - `android:layout_marginTop="84dp" Jarak` dari dadu: 84dp
 - `android:text="@string/roll" ->` Teks tombol ambil dari file `strings.xml`.
 - `app:layout_constraintEnd_toEndOf="parent"`
`app:layout_constraintHorizontal_bias="0.498"`
`app:layout_constraintStart_toStartOf="parent"`
`app:layout_constraintTop_toBottomOf="@+id/imageView" ->` Posisinya di tengah horizontal.

D. Source Code Compose

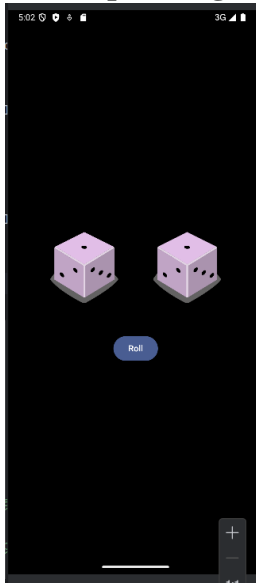
MainActivity.kt

```
1 package com.example.myrollerdiceappcompose
2
3 import android.os.Bundle
4 import android.widget.Toast
5 import androidx.activity.ComponentActivity
6 import androidx.activity.compose.setContent
7 import androidx.compose.foundation.Image
8 import androidx.compose.foundation.background
9 import androidx.compose.foundation.layout.*
10 import androidx.compose.material3.Button
11 import androidx.compose.material3.Text
12 import androidx.compose.runtime.*
13 import androidx.compose.ui.Alignment
14 import androidx.compose.ui.Modifier
15 import androidx.compose.ui.graphics.Color
16 import androidx.compose.ui.platform.LocalContext
17 import androidx.compose.ui.res.painterResource
18 import androidx.compose.ui.unit.dp
19 import
20 com.example.myrollerdiceappcompose.ui.theme.MyRollerDiceAppCompose
21 Theme
22
23 class MainComposeActivity : ComponentActivity() {
24     override fun onCreate(savedInstanceState: Bundle?) {
25         super.onCreate(savedInstanceState)
26         setContent {
27             MyRollerDiceAppComposeTheme {
28
29                 DiceRollerScreen(modifier =
30 Modifier.fillMaxSize())
31             }
32         }
33     }
34 }
35
36 @Composable
37 fun DiceRollerScreen(modifier: Modifier = Modifier) {
38     var dice1 by remember { mutableIntStateOf(1) }
39     var dice2 by remember { mutableIntStateOf(1) }
40
41     val context = LocalContext.current
42
43     val diceImages = listOf(
44         R.drawable.dice_0,
45         R.drawable.dice_1,
46         R.drawable.dice_2,
```

47	R.drawable.dice_3,
48	R.drawable.dice_4,
49	R.drawable.dice_5,
50	R.drawable.dice_6
51)
52	
53	Column(
54	modifier = modifier
55	.background(Color.Black)
56	.padding(16.dp),
57	horizontalAlignment = Alignment.CenterHorizontally,
58	verticalArrangement = Arrangement.Center
59) {
60	Row(
61	horizontalArrangement = Arrangement.spacedBy(16.dp)
62) {
63	Image(
64	painter = painterResource(id = diceImages[dice1]),
65	contentDescription = "Dadu 1",
66	modifier = Modifier.size(150.dp)
67)
68	Image(
69	painter = painterResource(id = diceImages[dice2]),
70	contentDescription = "Dadu 2",
71	modifier = Modifier.size(150.dp)
72)
73	}
74	
75	Spacer(modifier = Modifier.height(32.dp))
76	
77	Button(onClick = {
78	dice1 = (1..6).random()
79	dice2 = (1..6).random()
80	
81	if (dice1 == dice2) {
82	Toast.makeText(context, "Selamat, anda dapat dadu
83	double!", Toast.LENGTH_SHORT).show()
84	} else {
85	Toast.makeText(context, "Anda belum beruntung!",
86	Toast.LENGTH_SHORT).show()
87	}
88	}) {
89	Text("Roll")
90	}
	}

Tabel 2. Source Code Jawaban Soal 1

E. Output Program



Gambar 1. Screenshot Hasil Jawaban Soal 1

F. Pembahasan

MainActivity.kt:

1. Pada package `com.example.mydicerollerapp`, nama package dari aplikasi
2. Pada bagian class “`class MainComposeActivity`”, dideklarasikan kelas utama bernama `MainComposeActivity` diaman layar pertama aplikasi dijalankan
3. Pada bagian fungsi `onCreate()` akan dipanggil saat aplikasi dibuka pertama kali,
 - `setContent` = baris ini menampilkan UI dari aplikasi pakai Compose
 - `MyRollerDiceAppComposeTheme` = untuk mengatur tampilan aplikasi
 - `DiceRollerScreen(modifier = Modifier.fillMaxSize())` = fungsi ini akan menampilkan layer utama dengan dua dadu dan tombol roll
4. Pada bagian `@Composable fun DiceRollerScreen(...)` menggambarkan tampilan aplikasi utama aplikasi terdapat
 - `var dice1 by remember { mutableIntStateOf(1) }`
`var dice2 by remember { mutableIntStateOf(1) }`
 - `Dice1` dan `Dice2` menyimpan angka hasil lempar dadu, kalau `remember` itu fungsinya untuk menyimpan nilai walau UI nya di rebuild
5. Pada bagian `val diceImages = listOf` berisi `R.drawable.dice_0...` itu untuk menyimpan daftar gambar dadu dalam bentuk array
6. Pada bagian tampilan UI seperti `Column(...)` = susunan dari atas ke bawah, `Row(...)` Susunan dadu Secara Horizontal, `Image(...)` Untuk menampilkan gambar dadu, `Button(...)` = tombol untuk melempar dadu didalam terdapat baris kode `dice1 = (1..6).random()` `dice2 = (1..6).random()` ada ada kondisional jadi pada saat tombol diklik dua angka acak akan dibuat, kalau angka sama maka akan keluar kondisi `if` jika dadu beda maka akan eksekusi kondisi `else` menggunakan `Toast` untuk memunculkan pesan singkat

G. Tautan Git

Berikut adalah tautan untuk source code yang telah dibuat.

<https://github.com/ALYAROSAAN/Pemrograman-Mobile>