

**LAPORAN PRAKTIKUM
PEMROGRAMAN MOBILE
MODUL 1**



ANDROID BASIC WITH KOTLIN

Oleh:

Jovan Gilbert Natamasindah NIM. 2310817310002

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

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 : Jovan Gilbert Natamasindah
NIM : 2310817310002

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	12
C. Pembahasan	12
D. Tautan Git.....	16

DAFTAR GAMBAR

Tabel 1. Source Code Jawaban Soal 1.....	11
--	----

DAFTAR TABEL

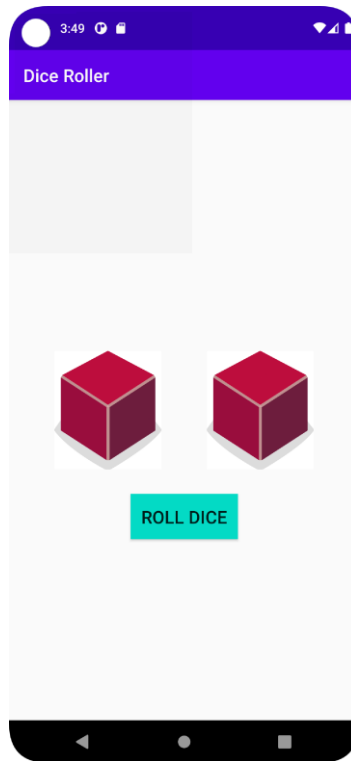
Tabel 1. Source Code Jawaban Soal 1	Error! Bookmark not defined.
Tabel 2. Source Code Jawaban Soal 1	10

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 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.



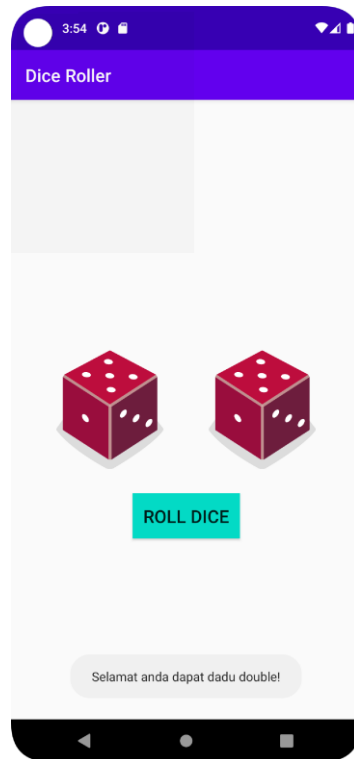
Gambar 1 Tampilan Awal Aplikasi

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.android.modul1
2
3 import android.media.Image
4 import android.os.Bundle
5 import android.widget.Toast
6 import androidx.activity.ComponentActivity
7 import androidx.activity.compose.setContent
8 import androidx.activity.enableEdgeToEdge
9 import androidx.compose.foundation.Image
10 import androidx.compose.foundation.background
11 import androidx.compose.foundation.layout.Arrangement
12 import androidx.compose.foundation.layout.Column
13 import androidx.compose.foundation.layout.Row
14 import androidx.compose.foundation.layout.fillMaxSize
15 import androidx.compose.foundation.layout.padding
16 import androidx.compose.foundation.layout.statusBarsPadding
17 import androidx.compose.foundation.layout.width
18 import androidx.compose.foundation.shape.RoundedCornerShape
19 import androidx.compose.material3.Button
20 import androidx.compose.material3.ButtonDefaults
21 import androidx.compose.material3.ExperimentalMaterial3Api
22 import androidx.compose.material3.Scaffold

```



```

23 import androidx.compose.material3.Text
24 import androidx.compose.material3.TopAppBar
25 import androidx.compose.material3.TopAppBarDefaults
26 import androidx.compose.runtime.Composable
27 import androidx.compose.runtime.mutableStateOf
28 import androidx.compose.runtime.getValue
29 import androidx.compose.runtime.saveable.rememberSaveable
30 import androidx.compose.runtime.setValue
31 import androidx.compose.ui.Alignment
32 import androidx.compose.ui.Modifier
33 import androidx.compose.ui.platform.LocalContext
34 import androidx.compose.ui.res.colorResource
35 import androidx.compose.ui.res.painterResource
36 import androidx.compose.ui.unit.dp
37 import com.android.modull1.ui.theme.Modull1Theme
38
39 class MainActivity : ComponentActivity() {
40     override fun onCreate(savedInstanceState: Bundle?) {
41         super.onCreate(savedInstanceState)
42         enableEdgeToEdge()
43         setContent {
44             Modull1Theme {
45                 Display()
46             }
47         }
48     }
49 }
50
51 @Composable
52 fun Display() {
53     Scaffold(topBar = { TopBar() }) { innerPadding ->
54         RollDice(modifier = Modifier.padding(innerPadding))
55     }
56 }
57
58 @OptIn(ExperimentalMaterial3Api::class)
59 @Composable
60 fun TopBar( ) {
61     TopAppBar(
62         title = { Text("Dice Roller") },
63         colors = TopAppBarDefaults.topAppBarColors(
64             containerColor = colorResource(id =
65 R.color.blue_dark),
66             titleContentColor = colorResource(id =
67 R.color.white)
68         ),
69         modifier = Modifier
70             .statusBarsPadding()
71     )
72 }
73
74 @Composable

```

```

73 fun RollDice(modifier: Modifier = Modifier ) {
74
75     var Dice1 by rememberSaveable { mutableStateOf(0) }
76     var Dice2 by rememberSaveable { mutableStateOf(0) }
77     val context = LocalContext.current
78
79     val (img1, img2) = diceImg(Dice1, Dice2)
80     val Img1 = img1
81     val Img2 = img2
82
83     Column(
84         modifier = modifier
85             .background(colorResource(id = R.color.cream))
86             .fillMaxSize(),
87         verticalArrangement = Arrangement.Center,
88         horizontalAlignment = Alignment.CenterHorizontally
89     ) {
90         Row {
91             Image(
92                 painter = painterResource(Img1),
93                 contentDescription = "Dice 1 image",
94                 modifier = Modifier
95                     .width(150.dp)
96             )
97             Image(
98                 painter = painterResource(Img2),
99                 contentDescription = "Dice 2 image",
100                modifier = Modifier
101                    .width(150.dp)
102            )
103        }
104        Button(onClick = {
105            Dice1 = randomDice()
106            Dice2 = randomDice()
107
108            if(Dice1 == Dice2) {Toast.makeText(context,
109 "Selamat anda dapat dadu double!", Toast.LENGTH_SHORT).show()}
110            else{Toast.makeText(context, "Anda belum
111 beruntung!", Toast.LENGTH_SHORT).show()}
112        },
113            colors = ButtonDefaults.buttonColors(
114                contentColor = colorResource(id =
115 R.color.white),
116                containerColor = colorResource(id =
117 R.color.blue_darker)
118            ),
119            shape = RoundedCornerShape(10.dp)
120        ) {
121            Text("ROLL DICE")
122        }
123    }
124 }

```

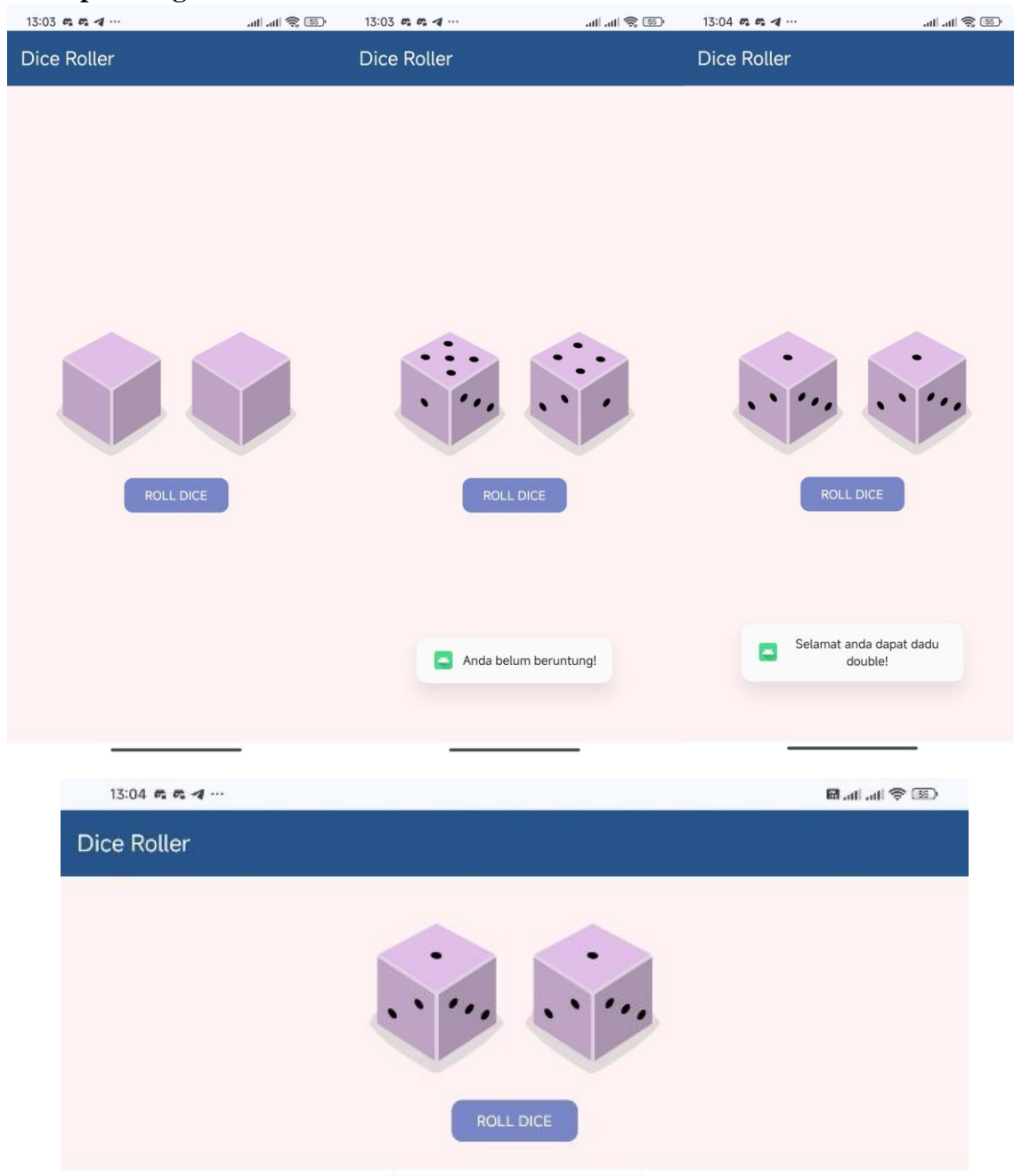
Tabel 1. Source Code Jawaban Soal 1

2. RandomDice.kt

```
1 package com.android.modul1
2
3 import kotlin.random.Random
4
5 fun randomDice(): Int {return Random.nextInt(1,7)}
6
7 fun diceImg(Dice1: Int, Dice2: Int): Pair<Int, Int> {
8     var img1 = R.drawable.dice_0
9     var img2 = R.drawable.dice_0
10
11     when(Dice1) {
12         1 -> img1 = R.drawable.dice_1
13         2 -> img1 = R.drawable.dice_2
14         3 -> img1 = R.drawable.dice_3
15         4 -> img1 = R.drawable.dice_4
16         5 -> img1 = R.drawable.dice_5
17         6 -> img1 = R.drawable.dice_6
18     }
19     when(Dice2) {
20         1 -> img2 = R.drawable.dice_1
21         2 -> img2 = R.drawable.dice_2
22         3 -> img2 = R.drawable.dice_3
23         4 -> img2 = R.drawable.dice_4
24         5 -> img2 = R.drawable.dice_5
25         6 -> img2 = R.drawable.dice_6
26     }
27     return Pair(img1, img2)
28 }
```

Tabel 2. Source Code Jawaban Soal 1

B. Output Program



Gambar 1. Screenshot Hasil Jawaban Soal 1

C. Pembahasan

1. MainActivity.kt:

- Pada baris 1, **package com.android.modul1** pendeklarasian nama package file Kotlin.

- Pada baris 3-37, **import** berfungsi untuk mengimpor package atau kelas untuk mengakses komponen dan fungsi dari android SDK dan jetpack compose.
- Pada baris 39, **class MainActivity : ComponentActivity()**, merupakan titik mula yang menjadi kelas utama dan akan dijalankan pertama kali saat aplikasi dibuka
- Pada baris 40, **override fun onCreate(savedInstanceState: Bundle?)**, berfungsi untuk menimpa (override) fungsi onCreate dari ComponentActivity.
- Pada baris 41, **super.onCreate(savedInstanceState)**, berfungsi untuk memanggil superclass dari fungsi onCreate untuk memastikan bahwa proses inisialisasi standar dari Android dijalankan sebelum logika saya dijalankan.
- Pada baris 42, **enableEdgeToEdge()**, berfungsi agar tampilan aplikasi dapat menggunakan seluruh layar dari status bar sampai navigation bar atau fullscreen layout.
- Pada baris 43, **setContent()**, digunakan untuk menampilkan UI berbasis jetpack compose ke dalam activity
- Pada baris 44, **Modul1Theme**, merupakan fungsi yang berisi tema custom yang membungkus seluruh UI untuk memberikan style yang konsisten.
- Pada baris 45, **Display()**, merupakan fungsi yang dibuat untuk menampilkan aplikasi pengocok dadu ke dalam activity.
- Pada baris 51-72, **@composable**, merupakan annotation untuk menandai bahwa fungsi yang dibuat dibawahnya merupakan fungsi UI di jetpack compose.
- Pada baris 52, **fun Display()**, merupakan fungsi yang dibuat untuk menampung semua component UI yang ingin ditampilkan ke dalam activity, seperti TopBar() dan RollDice().
- Pada baris 53, **scaffold()**, merupakan komponen layout yang menjadi kerangka dasar dari tampilan yang mempermudah dalam mengatur struktur UI karena sudah disediakan slot-slot bawannya seperti topBar yang sudah saya isi dengan TopBar() dan content saya isi dengan RollDice().
- Pada baris 53, **TopBar()**, berfungsi untuk memanggil fungsi TopBar() di dalam scaffold dan dimasukkan ke dalam slot topBar untuk ditampilkan dalam activity.
- Pada baris 54, **RollDice()**, berfungsi untuk memanggil fungsi RollDice() dengan parameter Modifier.padding(innerPadding) untuk memberikan padding bawaan dari scaffold.
- Pada baris 58, **@OptIn(ExperimentalMaterial3Api::class)**, berfungsi untuk memberitahukan compiler bahwa fungsi composable dibawahnya masih menggunakan API yang bersifat eksperimental untuk menghindari error saat kompilasi.

- Pada baris 60, **fun TopBar()**, merupakan fungsi composable yang dibuat untuk menampung komponen UI TopAppBar yang akan ditampilkan ke activity tepatnya dibagian atas layar.
- Pada baris 61, **TopAppBar()**, komponen yang berfungsi untuk menampilkan elemen UI di bagian atas layar.
 - Pada baris 62, **title**, merupakan parameter dalam TopAppBar() untuk menampilkan judul .
 - Pada baris 63, **colors** merupakan parameter dalam TopAppBar() untuk mengatur warna background (containerColor) dan warna teks judul (titleContentColor).
 - Pada baris 67, **modifier** merupakan parameter dalam TopAppBar() untuk menerapkan modifikasi terhadap tampilan komponen, salah satunya memberikan .statusBarPadding() untuk menambahkan padding secara otomatis di atas layar guna menghindari tabrakan elemen dengan status bar.
- Pada baris 73, **fun RollDice(modifier: Modifier = Modifier)**, merupakan fungsi composable yang dibuat untuk menampung beberapa logika sederhana mengenai aplikasi pengocok dadu, menerima parameter modifier, dan komponen UI Column, Row, Image, Button, Text yang akan ditampilkan ke activity.
- Pada baris 75 dan 76, **var** merupakan keyword untuk membuat suatu variable bersifat mutable yang menampung nilai dari berbagai tipe data.
- Pada baris 75 dan 76, **rememberSaveable** berfungsi untuk menyimpan dan mengingat nilai dari variable, dalam kasus ini Dice1 dan Dice2 agar selama komponen composable masih aktif dan jika terjadi perubahan konfigurasi layar misalnya saat dirotasi.
- Pada baris 77, **LocalContext.current** berfungsi untuk mendapatkan context dari activity saat ini dalam fungsi composable dan juga digunakan untuk menampilkan Toast sebagai notifikasi dari hasil dadu.
- Pada baris 77-81, **val** merupakan keyword untuk membuat suatu variable bersifat immutable yang menampung nilai dari berbagai tipe data
- Pada baris 83, **Column()** merupakan komponen layout mirip seperti scaffold, hanya saja Column digunakan untuk menyusun elemen UI secara vertical.
 - Pada baris, 84 **modifier** merupakan parameter dalam Column() untuk menerapkan modifikasi terhadap tampilan komponen, seperti *.background* untuk memberikan warna background dan *.fillMaxSize* agar column mengisi seluruh ukuran layar.

- Pada baris 87, **verticalArrangement** berfungsi untuk mengatur posisi vertical dari isi Column.
- Pada baris 88, **horizontalAlignment** berfungsi untuk mengatur posisi horizontal dari isi Column.
- Pada baris 90, **Row** merupakan komponen layout yang berlawanan dari Column, Dimana jika Column menyusun elemen UI secara Vertical, maka Row menyusun elemen UI secara horizontal.
- Pada baris 91 dan 97, **Image()** merupakan komponen UI yang berfungsi untuk menampilkan gambar.
 - Pada baris 92 dan 98, **painter** merupakan parameter dalam Image() untuk memasukkan gambar dari resource agar tampil di dalam Image()
 - Pada baris 93 dan 99, **contentDescription** merupakan parameter dalam Image() untuk memberikan deskripsi gambar.
 - Pada baris 94 dan 100, **modifier** merupakan parameter dalam Image() untuk memodifikasi tampilan gambar, dalam kasus ini memodifikasi widthnya menjadi 150.dp.
- Pada baris 104, **Button()** merupakan komponen UI yang berfungsi untuk membuat tombol pada aplikasi.
 - Pada baris 104, **onClick** merupakan parameter dalam Button() untuk memberikan aksi yang akan dijalankan ketika tombol ditekan.
 - Pada baris 108 dan 109, **if** dan **else** merupakan perkondisian Dimana jika kondisi if terpenuhi maka statementnya akan dijalankan, tetapi jika tidak maka else akan dijalankan.
 - Pada baris 111, **colors** merupakan parameter dalam Button() untuk mengatur warna background (containerColor) dan warna teks (titleContentColor).
 - Pada baris 115, **shape** merupakan parameter dalam Button() untuk mengatur sudut agar dapat membulat.
 - Pada baris 117, **Text()** merupakan komponen UI yang berfungsi menampilkan teks. Dalam kasus ini menampilkan teks di dalam tombol.

2. RandomDice.kt

- Pada baris 1, **package com.android.modul1** pendeklarasian nama package file Kotlin.
- Pada baris 3, **import** berfungsi untuk mengimpor package atau kelas untuk mengakses komponen dan fungsi dari android SDK dan jetpack compose

- Pada baris 5, **fun randomDice()** merupakan fungsi untuk mengacak nilai dadu dengan cara menggunakan fungsi random untuk mengacak angka dari 1 sampai 6 yang akan ditampilkan dalam bentuk gambar dadu.
- Pada baris 5 dan 27, **return** merupakan pengembalian terhadap suatu nilai di sebuah fungsi . jika di randomDice() akan mengembalikan angka random dan di diceImg() akan mengembalikan pasangan id gambar dadu
- Pada baris 7, **diceImg(Dice1: Int, Dice2: Int): Pair<Int, Int>** merupakan fungsi untuk mendapatkan gambar dadu berdasarkan angka random dari fungsi randomDice()
- Pada baris 8 dan 9, **var** merupakan keyword untuk membuat suatu variable bersifat mutable yang menampung nilai dari berbagai tipe data.
- Pada baris 11 dan 19, **when** merupakan perkondisian mirip seperti if else hanya saja syntaxnya lebih pendek dan mudah dipahami.

D. Tautan Git

Berikut adalah tautan untuk source code yang telah dibuat.

<https://github.com/AppleCraft2005/kuliah-santuy/tree/main/semesterIV/Pemrograman-Mobile>