

**LAPORAN PRAKTIKUM
PEMROGRAMAN MOBILE
MODUL 1**



ANDROID BASIC WITH KOTLIN

Oleh:

Farisa Adelia

NIM. 2110817120010

**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 : Farisa Adelia
NIM : 2110817120010

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

DAFTAR ISI	3
DAFTAR GAMBAR.....	4
DAFTAR TABEL	5
SOAL 1.....	6
A. Source Code	6
B. Output Program.....	9
C. Pembahasan	10
D. Tautan Git	10

DAFTAR GAMBAR

Gambar 1. Output Dadu	9
-----------------------------	---

DAFTAR TABEL

Tabel 1. Source Code Activity	8
Tabel 2. Source Code Main	9

SOAL 1

Buatlah sebuah aplikasi Dice Roller yang dapat menampilkan sebuah dadu yang dapat berubah-ubah tampilannya.

A. Source Code

```
1 package com.example.praktikumobile1
2
3 import androidx.appcompat.app.AppCompatActivity
4 import android.os.Bundle
5 import android.widget.Button
6 import android.widget.ImageView
7 import android.widget.TextView
8 import android.widget.Toast
9
10 /**
11  * activity ini untuk user menekan tombol roll tersebut
12  * on the screen.
13  */
14 class MainActivity : AppCompatActivity() {
15     override fun onCreate(savedInstanceState: Bundle?) {
16         super.onCreate(savedInstanceState)
17         setContentView(R.layout.activity_main)
18         val rollButton: Button = findViewById(R.id.button)
19
20         rollButton.setOnClickListener { rollDice() }
21 //      di bawah ini untuk menampilkan dadu saat aplikasi pertama
22 kali di buka
23         rollDice()
24     }
25
26     //Roll the dice and update the screen with the result.
27     private fun rollDice() {
28         // Create new Dice object with 6 sides and roll it
29         val dice = Dice(6)
30         val diceRoll = dice.roll()
31         val diceRoll2 = dice.roll()
32
33
34         // Update the screen with the dice roll
35 //      val resultTextView: TextView =
36 findViewById(R.id.textView)
37         val diceImage: ImageView = findViewById(R.id.imageView)
```

```

38
39 //                                val    resultTextView2:    TextView    =
40 findViewById(R.id.textView2)
41     val                                diceImage2:                                ImageView                                =
42 findViewById(R.id.imageView2)
43
44     // Determine which drawable resource ID to use based on
45 the dice roll
46
47     val drawableResource = when (diceRoll) {
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         else -> R.drawable.dice_6
54     }
55     // Update the ImageView with the correct drawable
56 resource ID
57
58     diceImage.setImageResource(drawableResource)
59     val drawableResource2 = when (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     // Update the ImageView with the correct drawable
68 resource ID
69
70     diceImage2.setImageResource(drawableResource2)
71     // Update the content description
72
73     diceImage.contentDescription = diceRoll.toString()
74     diceImage2.contentDescription = diceRoll2.toString()
75
76     // pengkondisian terhadap dadu yang sama atau beda
77 terhadap pesan yang muncul ke layar
78     if (diceRoll == diceRoll2) {
79         val toast = Toast.makeText(
80             this,
81             "Anda dapat dadu double yang sama!",
82             Toast.LENGTH_SHORT
83         )
84         toast.show()

```

```

85         } else {
86             val toast2 = Toast.makeText(this, "Anda belum
87 beruntung!", Toast.LENGTH_SHORT)
88             toast2.show()
89         }
90     }
91
92 }
93
94
95 class Dice(private val numSides: Int) {
96
97     fun roll(): Int {
98         return (1..numSides).random()
99     }
100 }

```

Tabel 1. Source Code Activity

```

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:layout_width="match_parent"
7      android:layout_height="match_parent"
8      tools:context=".MainActivity">
9
10     <Button
11         android:id="@+id/button"
12         android:layout_width="wrap_content"
13         android:layout_height="wrap_content"
14         android:layout_marginTop="16dp"
15         android:text="@string/roll"
16         android:textSize="36sp"
17         app:layout_constraintEnd_toEndOf="parent"
18         app:layout_constraintStart_toStartOf="parent"
19         app:layout_constraintTop_toBottomOf="@+id/imageView" />
20
21     <ImageView
22         android:id="@+id/imageView"
23         android:layout_width="160dp"
24         android:layout_height="200dp"
25         app:layout_constraintBottom_toBottomOf="parent"
26         app:layout_constraintEnd_toEndOf="parent"
27         app:layout_constraintHorizontal_bias="0.498"

```



```

28         app:layout_constraintStart_toStartOf="parent"
29         app:layout_constraintTop_toTopOf="parent"
30         app:layout_constraintVertical_bias="0.499"
31         tools:srcCompat="@drawable/dice_1" />
32
33     <ImageView
34         android:id="@+id/imageView2"
35         android:layout_width="160dp"
36         android:layout_height="200dp"
37         app:layout_constraintBottom_toTopOf="@+id/imageView"
38         app:layout_constraintEnd_toEndOf="parent"
39         app:layout_constraintStart_toStartOf="parent"
40         app:layout_constraintTop_toTopOf="parent"
41         tools:srcCompat="@drawable/dice_1" />
42
43 </androidx.constraintlayout.widget.ConstraintLayout>

```

Tabel 2. Source Code Main

B. Output Program



Gambar 1. Output Dadu

C. Pembahasan

Code MainActivity.kt tersebut adalah sebuah program yang berfungsi untuk menampilkan gambar dadu di layar dengan angka acak setiap kali tombol "Roll" ditekan. Program ini juga mengimplementasikan beberapa fitur tambahan seperti menampilkan pesan toast jika hasil dadu adalah double atau tidak.

Activity_amin.xml adalah file layout XML yang digunakan oleh activity MainActivity. Layout XML ini mendefinisikan tata letak tampilan yang akan ditampilkan oleh aplikasi pada layar.

D. Tautan Git

Berikut adalah tautan untuk source code yang telah dibuat.

<https://github.com/farisadelia/Pemrograman-Web>