

**LAPORAN PRAKTIKUM
PEMROGRAMAN MOBILE
MODUL 1**



ANDROID BASIC WITH KOTLIN

Oleh:

Firda Khoirunisa

NIM. 2310817220025

**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 : Firda Khoirunisa
NIM : 2310817220025

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	12
D. Tautan Git	13

DAFTAR GAMBAR

Gambar 1. Screenshot Hasil Jawaban Soal 1	11
Gambar 2. Screenshot Hasil Jawaban Soal 1	11

DAFTAR TABEL

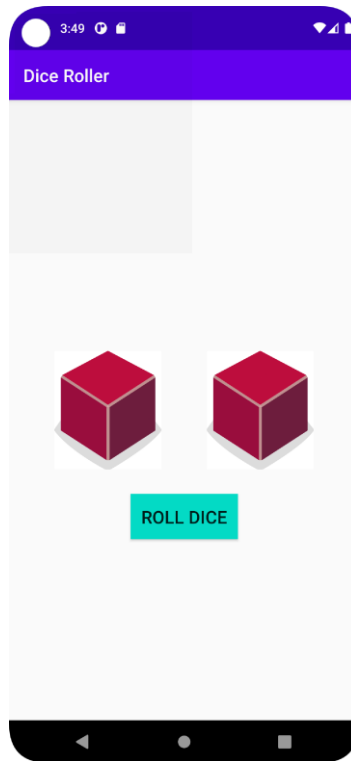
Tabel 1. Source Code Jawaban Soal 1.....	9
Tabel 2. Source Code Jawaban Soal 1.....	10

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 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.



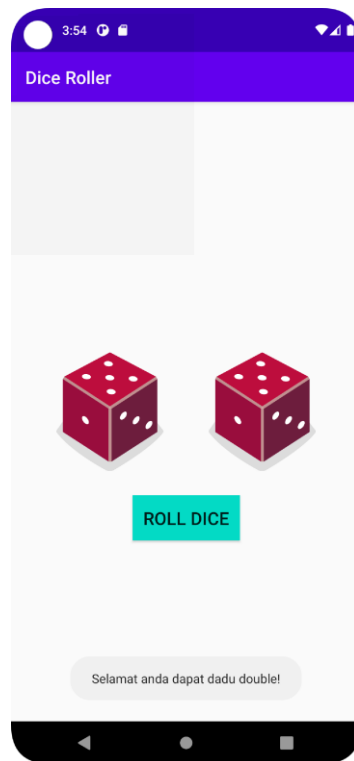
Gambar 1 Tampilan Awal Aplikasi

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.diceroller
2
3 import android.os.Bundle
4 import android.widget.Toast
5 import androidx.appcompat.app.AppCompatActivity
6 import com.example.diceroller.databinding.ActivityMainBinding
7
8 class MainActivity : AppCompatActivity() {
9     private lateinit var binding: ActivityMainBinding
10    private var dice1Value = 0
11    private var dice2Value = 0
12
13    override fun onCreate(savedInstanceState: Bundle?) {
14        super.onCreate(savedInstanceState)
15        binding = ActivityMainBinding.inflate(layoutInflater)
16        setContentView(binding.root)
17
18        binding.diceImage1.setImageResource(R.drawable.dice_0)
19        binding.diceImage2.setImageResource(R.drawable.dice_0)
20
21        savedInstanceState?.let {
22            dice1Value = it.getInt("DICE1")

```



```

23         dice2Value = it.getInt("DICE2")
24         updateDiceImages()
25     }
26
27     binding.buttonRoll.setOnClickListener {
28         rollDice()
29     }
30 }
31
32 private fun rollDice() {
33     dice1Value = (1..6).random()
34     dice2Value = (1..6).random()
35     updateDiceImages()
36
37     val message = if (dice1Value == dice2Value) {
38         "Selamat anda dapat dadu double!"
39     } else {
40         "Anda belum beruntung!"
41     }
42
43     Toast.makeText(this, message,
44         Toast.LENGTH_SHORT).show()
45 }
46
47 private fun updateDiceImages() {
48
49     binding.diceImage1.setImageResource(getDiceImage(dice1Value))
50
51     binding.diceImage2.setImageResource(getDiceImage(dice2Value))
52 }
53
54 private fun getDiceImage(value: Int): Int {
55     return when (value) {
56         1 -> R.drawable.dice_1
57         2 -> R.drawable.dice_2
58         3 -> R.drawable.dice_3
59         4 -> R.drawable.dice_4
60         5 -> R.drawable.dice_5
61         6 -> R.drawable.dice_6
62         else -> R.drawable.dice_0
63     }
64 }
65
66 override fun onSaveInstanceState(outState: Bundle) {
67     super.onSaveInstanceState(outState)
68     outState.putInt("DICE1", dice1Value)
69     outState.putInt("DICE2", dice2Value)
70 }
71 }

```

Tabel 1. Source Code Jawaban Soal 1

2. activity_main.xml

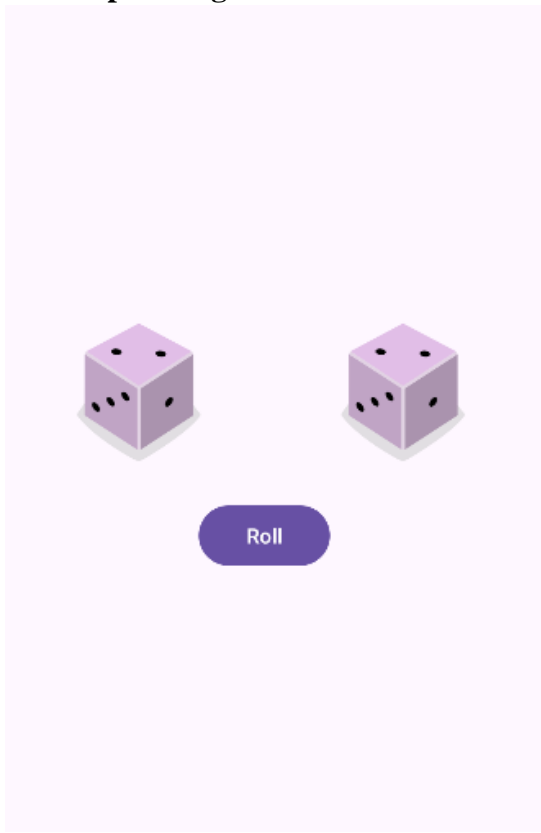
```

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:layout_width="match_parent"
7      android:layout_height="match_parent"
8      tools:context=".MainActivity">
9
10     <ImageView
11         android:id="@+id/dice_image_2"
12         android:layout_width="120dp"
13         android:layout_height="120dp"
14         android:layout_marginTop="392dp"
15         android:contentDescription="@string/dice_2"
16         android:src="@drawable/dice_0"
17         app:layout_constraintEnd_toEndOf="parent"
18         app:layout_constraintStart_toEndOf="@+id/dice_image_1"
19         app:layout_constraintTop_toTopOf="parent"          />
20
21     <ImageView
22         android:id="@+id/dice_image_1"
23         android:layout_width="120dp"
24         android:layout_height="120dp"
25         android:layout_marginTop="392dp"
26         android:contentDescription="@string/dice_1"
27         android:src="@drawable/dice_0"
28         app:layout_constraintEnd_toStartOf="@+id/dice_image_2"
29         app:layout_constraintStart_toStartOf="parent"
30         app:layout_constraintTop_toTopOf="parent"          />
31
32     <Button
33         android:id="@+id/button_roll"
34         android:layout_width="wrap_content"
35         android:layout_height="wrap_content"
36         android:layout_marginTop="12dp"
37         android:text="@string/roll"
38         app:layout_constraintEnd_toEndOf="parent"
39         app:layout_constraintHorizontal_bias="0.486"
40         app:layout_constraintStart_toStartOf="parent"
41         app:layout_constraintTop_toBottomOf="@+id/dice_image_1"
42     />
43
44 </androidx.constraintlayout.widget.ConstraintLayout>

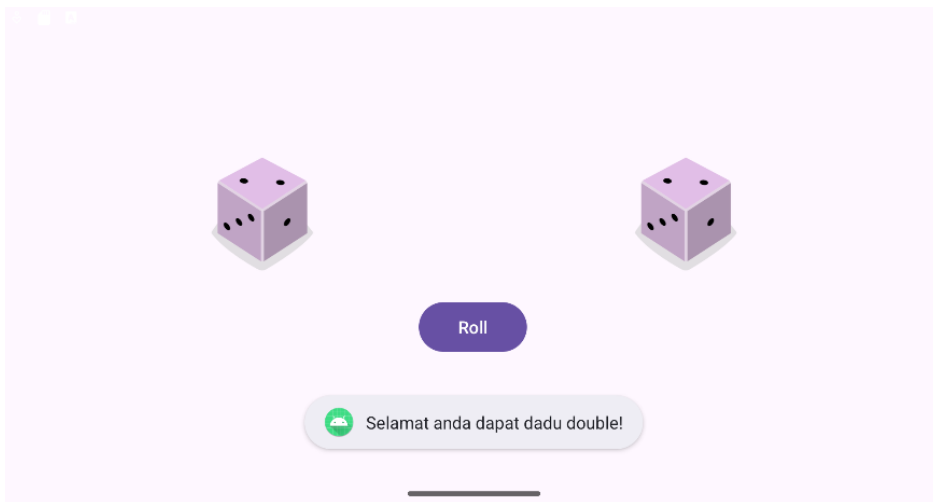
```

Tabel 2. Source Code Jawaban Soal 1

B. Output Program



Gambar 1. Screenshot Hasil Jawaban Soal 1



Gambar 2. Screenshot Hasil Jawaban Soal 1

C. Pembahasan

1. MainActivity.kt:

Pada baris [3-6], mengimpor beberapa library penting, yaitu Bundle, Toast, dan AppCompatActivity, serta ActivityMainBinding untuk menghubungkan layout XML ke kode Kotlin. Kemudian, di baris [8 - 11], dideklarasikan kelas MainActivity yang mewarisi AppCompatActivity, serta dua variabel: binding sebagai penghubung layout dan dice1Value serta dice2Value untuk menyimpan nilai acak dadu. Pada baris [13-28], method onCreate() digunakan sebagai titik awal saat Activity dijalankan. Di dalamnya, binding diinisialisasi dengan memanggil ActivityMainBinding.inflate(layoutInflater) (baris 15), dan ditetapkan sebagai tampilan utama dengan setContentView(binding.root) pada baris 16. Setelah itu pada baris [18-19], gambar dadu diatur menggunakan gambar awal dice_0. Jika ada data tersimpan dari instance sebelumnya, akan mengambil kembali nilai-nilai sebelumnya dengan getInt() lalu memperbarui gambar dadu menggunakan updateDiceImages(). Pada baris [27-28], mengatur listener pada tombol buttonRoll agar ketika tombol ditekan, metode rollDice() dipanggil. Method rollDice() didefinisikan di baris 32-44. Dalam metode ini, dua angka acak antara 1 dan 6 dihasilkan menggunakan (1..6).random() dan disimpan ke dalam dice1Value dan dice2Value (baris 33-34). Kemudian, gambar dadu diperbarui (baris 35) dan kondisi dicek—jika kedua dadu menunjukkan angka yang sama, akan muncul Toast dengan pesan keberuntungan; jika tidak, akan muncul pesan biasa (baris 37-40). Pada baris [47-51], metode updateDiceImages() mengatur ulang gambar pada kedua ImageView berdasarkan nilai dadu yang sekarang, dengan bantuan metode getDiceImage() yang akan mencocokkan angka dan mengembalikan resource drawable yang sesuai. Method getDiceImage() sendiri didefinisikan di baris [54-64]. Fungsi ini menerima value dan mencocokkannya dengan angka 1 hingga 6 untuk menentukan gambar yang sesuai (dice_1 sampai dice_6). Jika nilainya tidak sesuai, akan dikembalikan gambar dice_0. Terakhir, pada baris [66-69], metode onSaveInstanceState() menyimpan nilai dadu terakhir (DICE1 dan DICE2) ke dalam Bundle agar dapat dipulihkan jika aplikasi di-restart atau orientasi layar berubah.

2. activity_main.xml

Pada baris [2-8], struktur awal layout dideklarasikan menggunakan ConstraintLayout, yang memungkinkan kita mengatur posisi elemen UI secara fleksibel. Di baris [10-19],

ImageView pertama (dice_image_2) dibuat dengan ukuran 120dp x 120dp dan menggunakan gambar awal dice_0. Elemen ini diletakkan di sebelah kanan dice_image_1, ditentukan oleh constraint layout_constraintStart_toEndOf. Di baris [21-30], elemen ImageView kedua (dice_image_1) didefinisikan. Posisinya berada di kiri dan diatur agar berdampingan dengan dice_image_2. Kedua gambar dadu ini diberi deskripsi konten dari string resource (@string/dice_1 dan @string/dice_2) agar lebih ramah aksesibilitas. Baris [32-41] berisi elemen Button (button_roll) yang akan digunakan untuk memicu lemparan dadu. Tombol ini diatur agar berada di tengah-tengah layar secara horizontal (layout_constraintHorizontal_bias) dan berada tepat di bawah gambar dadu pertama. Dengan pendekatan layout ini, dua gambar dadu akan tampil berdampingan di bagian atas layar, sedangkan tombol "Roll" berada di bawahnya, siap ditekan oleh pengguna untuk menghasilkan angka acak.

D. Tautan Git

Berikut adalah tautan untuk source code yang telah dibuat.

[Pemrograman-Mobile/DiceRoller at main · firdakhrns/Pemrograman-Mobile](https://github.com/firdakhrns/Pemrograman-Mobile)