**LAPORAN PRAKTIKUM**

**PEMROGRAMAN MOBILE**

**MODUL 2**

****

**ANDROID BASICS IN KOTLIN**

**Oleh:**

**Aurelia Monica Sari NIM. 2010817220001**

**PROGRAM STUDI TEKNOLOGI INFORMASI**

**FAKULTAS TEKNIK**

**UNIVERSITAS LAMBUNG MANGKURAT**

**MARET 2022**

# LEMBAR PENGESAHAN

**LAPORAN PRAKTIKUM PEMROGRAMAN MOBILE**

**MODUL 2**

Laporan Praktikum Pemrograman Mobile Modul 2: *Android Basics In Kotlin* ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman Mobile. Laporan Prakitkum ini dikerjakan oleh:

Nama Praktikan : Aurelia Monica Sari

NIM : 2010817220001

|  |  |
| --- | --- |
| Menyetujui,  Asisten Praktikum  Rezi Rahadianor  NIM. 1810817210019 | Mengetahui,  Dosen Penanggung Jawab Praktikum  Andreyan Rizky Baskara, S.Kom., M.Kom.  NIP. 19930703 201903 1 011 |

# DAFTAR ISI

[LEMBAR PENGESAHAN 2](#_Toc99625131)

[DAFTAR ISI 3](#_Toc99625132)

[DAFTAR GAMBAR 4](#_Toc99625133)

[A. Soal 5](#_Toc99625134)

[B. Source Code 8](#_Toc99625135)

[C. Output Program 11](#_Toc99625136)

[D. Pembahasan 13](#_Toc99625137)

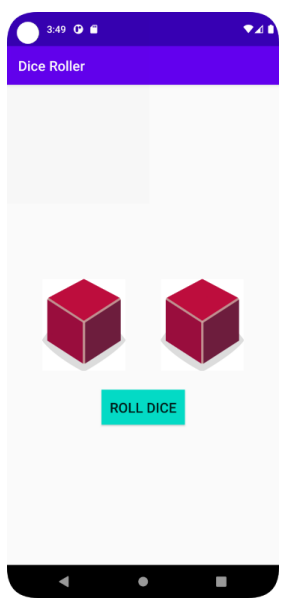
# DAFTAR GAMBAR

[Gambar 1. Screenshot Hasil Jawaban Soal 1 6](#_Toc96727062)

## Soal

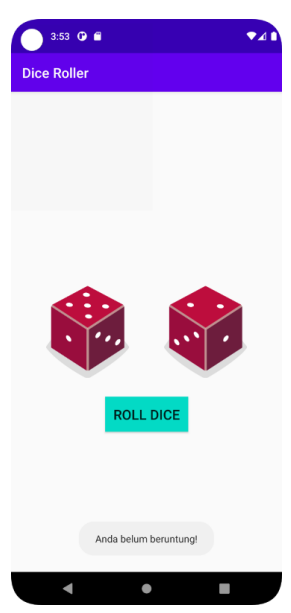
Buatlah sebuah aplikasi yang dapat menampilkan 2 (dua) buah dadu yang dapat berubah-ubah tampilannya pada saat user menekan tombol “Roll Dice”. Aturan aplikasi yang akan dibangun adalah sebagaimana berikut:

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



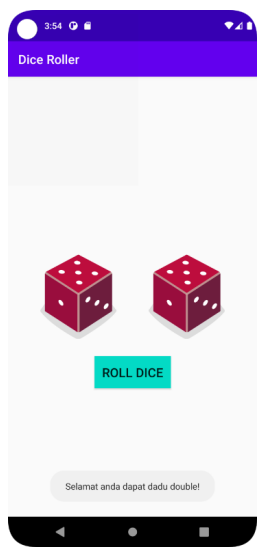
Gambar 1. Contoh Tampilan Awal Aplikasi

1. 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. Contoh Tampilan Dadu Setelah Diroll

1. 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.
2. Upload aplikasi yang telah anda buat kedalam repository github ke dalam folder Modul 2 dalam bentuk project. Jangan lupa untuk melakukan Clean Project sebelum mengupload pekerjaan anda pada repo.
3. Untuk gambar dadu dapat didownload pada link berikut: <https://drive.google.com/u/0/uc?id=147HT2lIH5qin3z5ta7H9y2N_5OMW81Ll&export=download>



Gambar 3. Contoh Tampilan Roll Dadu Double

## Source Code

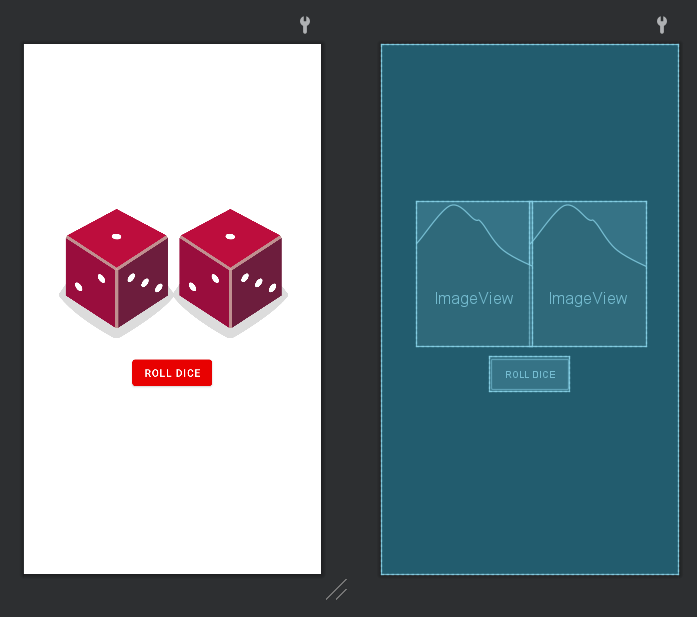
1. MainActivity.kt

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76 | package com.example.diceroller  import androidx.appcompat.app.AppCompatActivity  import android.os.Bundle  import android.widget.Button  import android.widget.ImageView  import android.widget.Toast  class MainActivity : AppCompatActivity() {  override fun onCreate(savedInstanceState: Bundle?) {  super.onCreate(savedInstanceState)  setContentView(R.layout.activity\_main)  val rollButton: Button = findViewById(R.id.button)  rollButton.setOnClickListener {  rollDice()  }  rollDice()  }  private fun rollDice() {  val dice1 = Dice(6)  val diceRoll1 = dice1.roll()  val diceImage1: ImageView = findViewById(R.id.imageView1)  val dice2 = Dice(6)  val diceRoll2 = dice2.roll()  val diceimage2: ImageView = findViewById(R.id.imageView2)  if (diceRoll1 == diceRoll2) {  val toast = Toast.makeText(this, "Selamat anda dapat dadu double!", Toast.LENGTH\_SHORT)  toast.show()  } else {  val toast = Toast.makeText(this, "Anda belum beruntung", Toast.LENGTH\_SHORT)  toast.show()  }  val drawableResource1 = when (diceRoll1) {  1 -> R.drawable.dice\_1  2 -> R.drawable.dice\_2  3 -> R.drawable.dice\_3  4 -> R.drawable.dice\_4  5 -> R.drawable.dice\_5  else -> R.drawable.dice\_6  }  val drawableResource2 = when (diceRoll2) {  1 -> R.drawable.dice\_1  2 -> R.drawable.dice\_2  3 -> R.drawable.dice\_3  4 -> R.drawable.dice\_4  5 -> R.drawable.dice\_5  else -> R.drawable.dice\_6  }  diceImage1.setImageResource(drawableResource1)  diceImage1.contentDescription = diceRoll1.toString()  diceimage2.setImageResource(drawableResource2)  diceimage2.contentDescription = diceRoll2.toString()  }  }  class Dice(private val numSides: Int) {  fun roll(): Int {  return (1..numSides).random()  }  } |

1. activity\_main.xml (code)

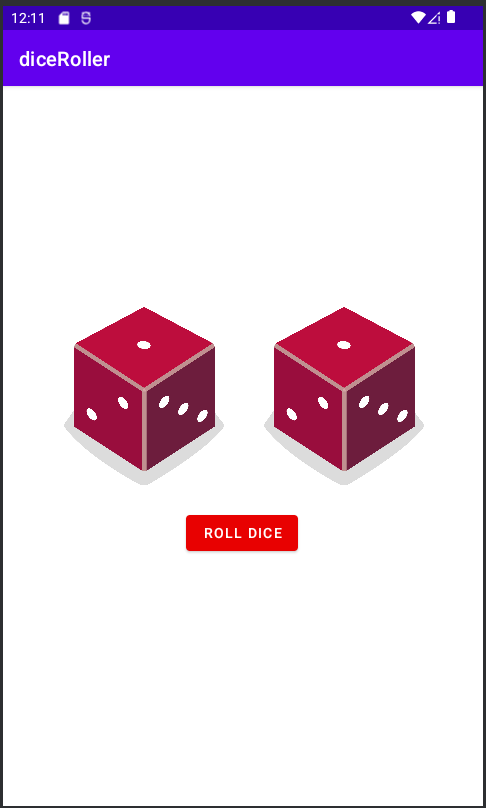
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47 | <?xml version="1.0" encoding="utf-8"?>  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".MainActivity">  <ImageView  android:id="@+id/imageView2"  android:layout\_width="160dp"  android:layout\_height="200dp"  app:layout\_constraintBottom\_toTopOf="@+id/button"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintHorizontal\_bias="0.816"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"  app:layout\_constraintVertical\_bias="0.943"  tools:srcCompat="@drawable/dice\_1" />  <ImageView  android:id="@+id/imageView1"  android:layout\_width="160dp"  android:layout\_height="200dp"  app:layout\_constraintBottom\_toTopOf="@+id/button"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintHorizontal\_bias="0.191"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"  app:layout\_constraintVertical\_bias="0.943"  tools:srcCompat="@drawable/dice\_1" />  <Button  android:id="@+id/button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="@string/roll\_dice"  app:backgroundTint="#E80101"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintHorizontal\_bias="0.498"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"  app:layout\_constraintVertical\_bias="0.629" />  </androidx.constraintlayout.widget.ConstraintLayout> |

1. activity\_main.xml (design)

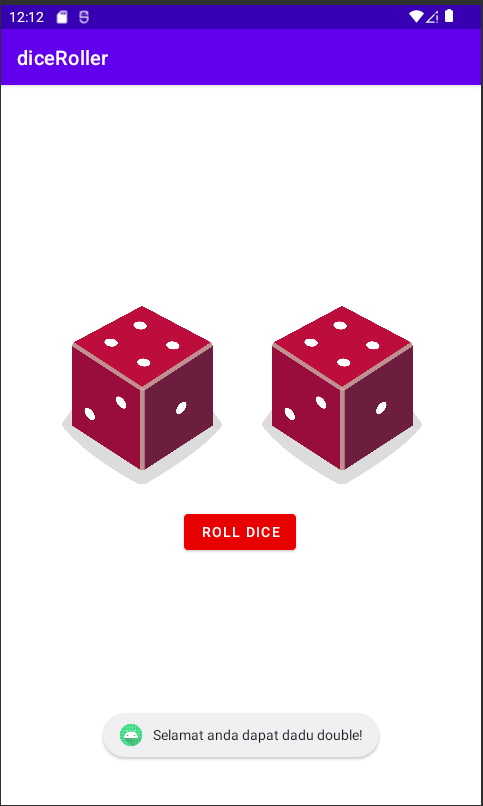
 

Gambar 4. Screenshot Design Pada activity\_main.xml. Gambar 5. Screenshot Component Tree

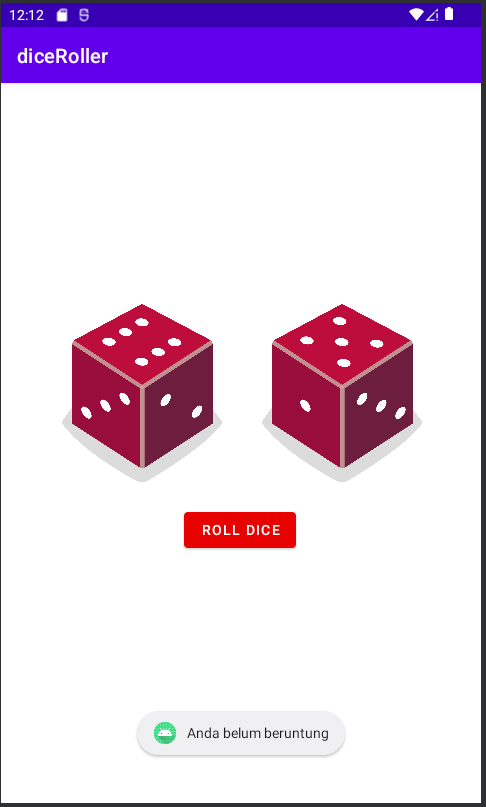
## Output Program



Gambar 6. Tampilan Awal Aplikasi



Gambar 7. Tampilan Aplikasi Setelah Dadu Di-roll



Gambar 8. Tampilan Aplikasi Saat Dadu Double

## Pembahasan

Pada baris 15 terdapat findViewById() untuk menemukan Button dalam tata letak. R.id.button adalah ID resource untuk Button yang merupakan ID uniknya. Kode menyimpan reference ke objek Button dalam variabel yang disebut rollButton, bukan objek Button itu sendiri.

Pada baris 17 terdapat objek rollButton dan click listener pada objek tersebut dengan memanggil metode setOnClickListener().

Pada baris 18 terdapat rollDice() untuk memanggil fungsi rollDice().

Pada baris 25 terdapat objek dice1 untuk menginisialisasi dadu dengan 6 sisi.

Pada baris 26 terdapat objek diceRoll1 untuk mengacak nilai pada dadu.

Pada baris 27 terdapat objek untuk menampilkan gambar dadu.

Pada baris 35-43 terdapat kondisional apabila kedua dadu menampilkan nilai yang sama maka akan menampilkan toast “Selamat anda mendapatkan dadu double”, dan sebaliknya apabila kedua dadu tidak menampilkan nilai yang sama maka akan menampilkan toast “Anda belum beruntung”.

Pada baris 45-68 terdapat kode untuk menampilkan gambar sesuai nilai dadu menggunakan kondisional when.

Pada baris 71 terdapat class Dice agar dapat menerima bilangan bulat yang disebut numSides.

Pada baris 73-74 terdapat funsi untuk mengacak nilai dadu.