

**LAPORAN PRAKTIKUM
PEMROGRAMAN MOBILE
MODUL 2**



TIP CALCULATOR APP

Oleh:

Farisa Adelia

NIM. 2110817120010

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS TEKNIK
UNIVERSITAS LAMBUNG MANGKURAT
APRIL 2024**

LEMBAR PENGESAHAN
LAPORAN PRAKTIKUM PEMROGRAMAN I
MODUL 2

Laporan Praktikum Pemrograman Mobile Modul 2: Tip Calculator App 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

| | |
|-------------------------|----|
| LEMBAR PENGESAHAN | 2 |
| DAFTAR ISI..... | 3 |
| DAFTAR GAMBAR | 4 |
| DAFTAR TABEL..... | 5 |
| SOAL 1 | 6 |
| A. Source Code | 6 |
| B. Output Program..... | 10 |
| C. Pembahasan..... | 13 |

DAFTAR GAMBAR

| | |
|--------------------------------------------------|----|
| Gambar 1. Output program awal | 10 |
| Gambar 2. Calculate button clicked..... | 11 |
| Gambar 3. Output all radio button selected | 12 |

DAFTAR TABEL

| | |
|------------------------------------------------|----|
| Tabel 1. Source Code MainActivity.kt | 6 |
| Tabel 2. Source Code activity_main.xml..... | 7 |
| Tabel 3. Source Code build.gradle (:app) | 9 |
| Tabel 4. Source Code strings.xml | 10 |

SOAL 1

Soal Praktikum:

Buat aplikasi kalkulator tip dengan membangun tata letak terlebih dahulu, kemudian terapkan logika untuk menghitung tip berdasarkan input pengguna dan tambahkan tampilan yang menarik pada aplikasi Tip Calculator menggunakan panduan Material Design.

A. Source Code

Tabel 1. Source Code MainActivity.kt

```
1 package com.example.tiptime
2
3 import androidx.appcompat.app.AppCompatActivity
4 import android.os.Bundle
5 import android.os.PersistableBundle
6 import com.example.tiptime.databinding.ActivityMainBinding
7 import java.text.NumberFormat
8
9 class MainActivity : AppCompatActivity() {
10
11     // Untuk mengambil referensi atribut yang dibuat pada file .xml
12     private lateinit var binding: ActivityMainBinding
13
14     // Untuk menginisialisasi objek binding
15     override fun onCreate(savedInstanceState: Bundle?) {
16         super.onCreate(savedInstanceState)
17
18         binding = ActivityMainBinding.inflate(layoutInflater)
19         setContentView(binding.root)
20
21         // untuk memanggil function calculateTip pada calculateButton
22         // ketika diklik
23         binding.calculateButton.setOnClickListener { calculateTip() }
24
25         // Function untuk memproses perhitungan tip
26         private fun calculateTip() {
27             val stringInTextField = binding.costOfService.text.toString()
28             val cost = stringInTextField.toDoubleOrNull()
29             if (cost == null) {
30                 binding.tipResult.text = ""
31                 return
32             }
33
34             // Untuk menghitung persenan
35             val tipPercentage = when
36             (binding.tipOptions.checkedRadioButtonId) {
37                 R.id.option_twenty_percent -> 0.20
38                 R.id.option_eighteen_percent -> 0.18
39                 else -> 0.15
40             }
```

| | |
|----|---------------------------------------------------------------|
| 39 | |
| 40 | // Untuk membulatkan angka keatas ketika switch diaktifkan |
| 41 | var tip = tipPercentage * cost |
| 42 | if (binding.roundUpSwitch.isChecked) { |
| 43 | tip = kotlin.math.ceil(tip) |
| 44 | } |
| 45 | |
| 46 | // Untuk memformat nilai mata uang sesuai setting mobile user |
| 47 | val formattedTip = |
| 48 | NumberFormat.getCurrencyInstance().format(tip) |
| 49 | binding.tipResult.text = getString(R.string.tip_amount, |
| 50 | formattedTip) |
| 51 | } |

Tabel 2. Source Code 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 | android:padding="16dp" |
| 9 | tools:context=".MainActivity"> |
| 10 | |
| 11 | <!-- Untuk membuat kolom teks yang dapat diubah oleh user di dalam |
| 12 | aplikasi --> |
| 13 | <EditText |
| 14 | android:id="@+id/cost_of_service" |
| 15 | android:layout_width="160dp" |
| 16 | android:layout_height="wrap_content" |
| 17 | android:hint="@string/cost_of_service" |
| 18 | android:inputType="numberDecimal" |
| 19 | app:layout_constraintStart_toStartOf="parent" |
| 20 | app:layout_constraintTop_toTopOf="parent" /> |
| 21 | |
| 22 | <!-- Untuk membuat teks yang tampil pada layar --> |
| 23 | <TextView |
| 24 | android:id="@+id/service_question" |
| 25 | android:layout_width="wrap_content" |
| 26 | android:layout_height="wrap_content" |
| 27 | android:text="@string/how_was_the_service" |
| 28 | app:layout_constraintStart_toStartOf="parent" |
| 29 | app:layout_constraintTop_toBottomOf="@+id/cost_of_service"/> |
| 30 | |
| 31 | <!-- Untuk mengelompokkan button yang dibuat menjadi 1 grup --> |
| 32 | <RadioGroup |
| 33 | android:id="@+id/tip_options" |
| 34 | android:layout_width="wrap_content" |
| 35 | android:layout_height="wrap_content" |
| 36 | android:checkedButton="@id/option_twenty_percent" |
| | android:orientation="vertical" |

```

37         app:layout_constraintStart_toStartOf="parent"
38         app:layout_constraintTop_toBottomOf="@id/service_question">
39
40         <!-- Tambahkan Radio Buttons disini -->
41         <RadioButton
42             android:id="@+id/option_twenty_percent"
43             android:layout_width="wrap_content"
44             android:layout_height="wrap_content"
45             android:text="@string/amazing_20" />
46
47         <RadioButton
48             android:id="@+id/option_eighteen_percent"
49             android:layout_width="wrap_content"
50             android:layout_height="wrap_content"
51             android:text="@string/good_18" />
52
53         <RadioButton
54             android:id="@+id/option_fifteen_percent"
55             android:layout_width="wrap_content"
56             android:layout_height="wrap_content"
57             android:text="@string/okay_15" />
58
59     </RadioGroup>
60
61     <!-- Untuk membuat tombol switch -->
62     <Switch
63         android:id="@+id/round_up_switch"
64         android:layout_width="0dp"
65         android:layout_height="wrap_content"
66         android:checked="true"
67         android:text="@string/round_up_tip"
68         app:layout_constraintEnd_toEndOf="parent"
69         app:layout_constraintStart_toStartOf="@+id/tip_options"
70         app:layout_constraintTop_toBottomOf="@+id/tip_options" />
71
72     <!-- Untuk membuat button berbentuk persegi -->
73     <Button
74         android:id="@+id/calculate_button"
75         android:layout_width="0dp"
76         android:layout_height="wrap_content"
77         android:text="@string/calculate"
78         app:layout_constraintEnd_toEndOf="parent"
79         app:layout_constraintStart_toStartOf="parent"
80         app:layout_constraintTop_toBottomOf="@+id/round_up_switch" />
81
82     <TextView
83         android:id="@+id/tip_result"
84         android:layout_width="wrap_content"
85         android:layout_height="wrap_content"
86         app:layout_constraintEnd_toEndOf="parent"
87         app:layout_constraintTop_toBottomOf="@+id/calculate_button"
88         tools:text="Tip Amount: $10" />
89
90 </androidx.constraintlayout.widget.ConstraintLayout>

```


Tabel 3. Source Code build.gradle (:app)

| | |
|----|-------------------------------------------------------------------|
| 1 | plugins { |
| 2 | id 'com.android.application' |
| 3 | id 'org.jetbrains.kotlin.android' |
| 4 | } |
| 5 | |
| 6 | android { |
| 7 | namespace 'com.example.tiptime' |
| 8 | compileSdk 33 |
| 9 | |
| 10 | defaultConfig { |
| 11 | applicationId "com.example.tiptime" |
| 12 | minSdk 24 |
| 13 | targetSdk 33 |
| 14 | versionCode 1 |
| 15 | versionName "1.0" |
| 16 | |
| 17 | testInstrumentationRunner |
| 18 | "androidx.test.runner.AndroidJUnitRunner" |
| 19 | } |
| 20 | buildTypes { |
| 21 | release { |
| 22 | minifyEnabled false |
| 23 | proguardFiles getDefaultProguardFile('proguard-android- |
| 24 | optimize.txt'), 'proguard-rules.pro' |
| 25 | } |
| 26 | compileOptions { |
| 27 | sourceCompatibility JavaVersion.VERSION_1_8 |
| 28 | targetCompatibility JavaVersion.VERSION_1_8 |
| 29 | } |
| 30 | kotlinOptions { |
| 31 | jvmTarget = '1.8' |
| 32 | } |
| 33 | buildFeatures { |
| 34 | viewBinding = true |
| 35 | } |
| 36 | } |
| 37 | |
| 38 | dependencies { |
| 39 | |
| 40 | implementation 'androidx.core:core-ktx:1.7.0' |
| 41 | implementation 'androidx.appcompat:appcompat:1.6.1' |
| 42 | implementation 'com.google.android.material:material:1.8.0' |
| 43 | implementation 'androidx.constraintlayout:constraintlayout:2.1.4' |
| 44 | testImplementation 'junit:junit:4.13.2' |
| 45 | androidTestImplementation 'androidx.test.ext:junit:1.1.5' |
| 46 | androidTestImplementation 'androidx.test.espresso:espresso- |
| 47 | core:3.5.1' |
| | } |

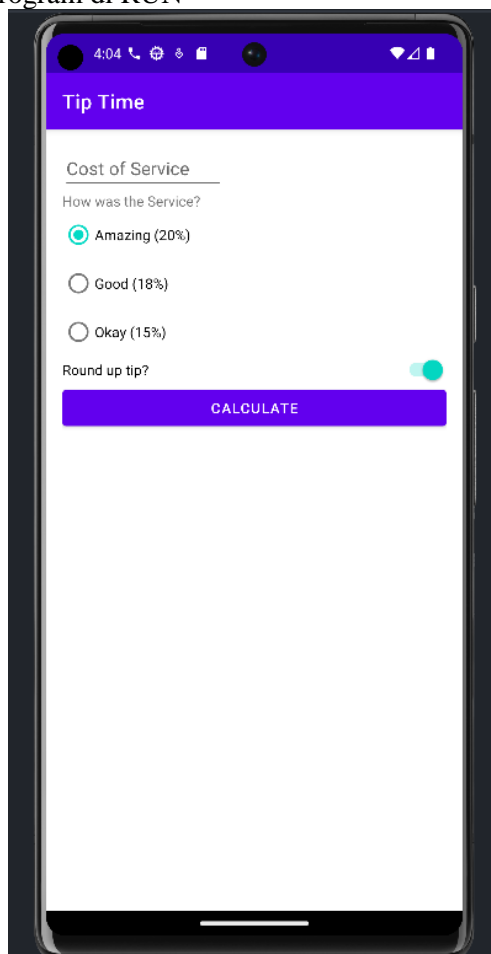
Tabel 4. Source Code strings.xml

| | |
|----|------------------------------------------------------------------|
| 1 | <resources> |
| 2 | <string name="app_name">Tip Time</string> |
| 3 | <string name="tip_amount">Tip Amount : %s</string> |
| 4 | <string name="calculate">Calculate</string> |
| 5 | <string name="cost_of_service">Cost of Service</string> |
| 6 | <string name="how_was_the_service">How was the Service?</string> |
| 7 | <string name="amazing_20">Amazing (20%)</string> |
| 8 | <string name="good_18">Good (18%)</string> |
| 9 | <string name="okay_15">Okay (15%)</string> |
| 10 | <string name="round_up_tip">Round up tip?</string> |
| 11 | </resources> |

B. Output Program

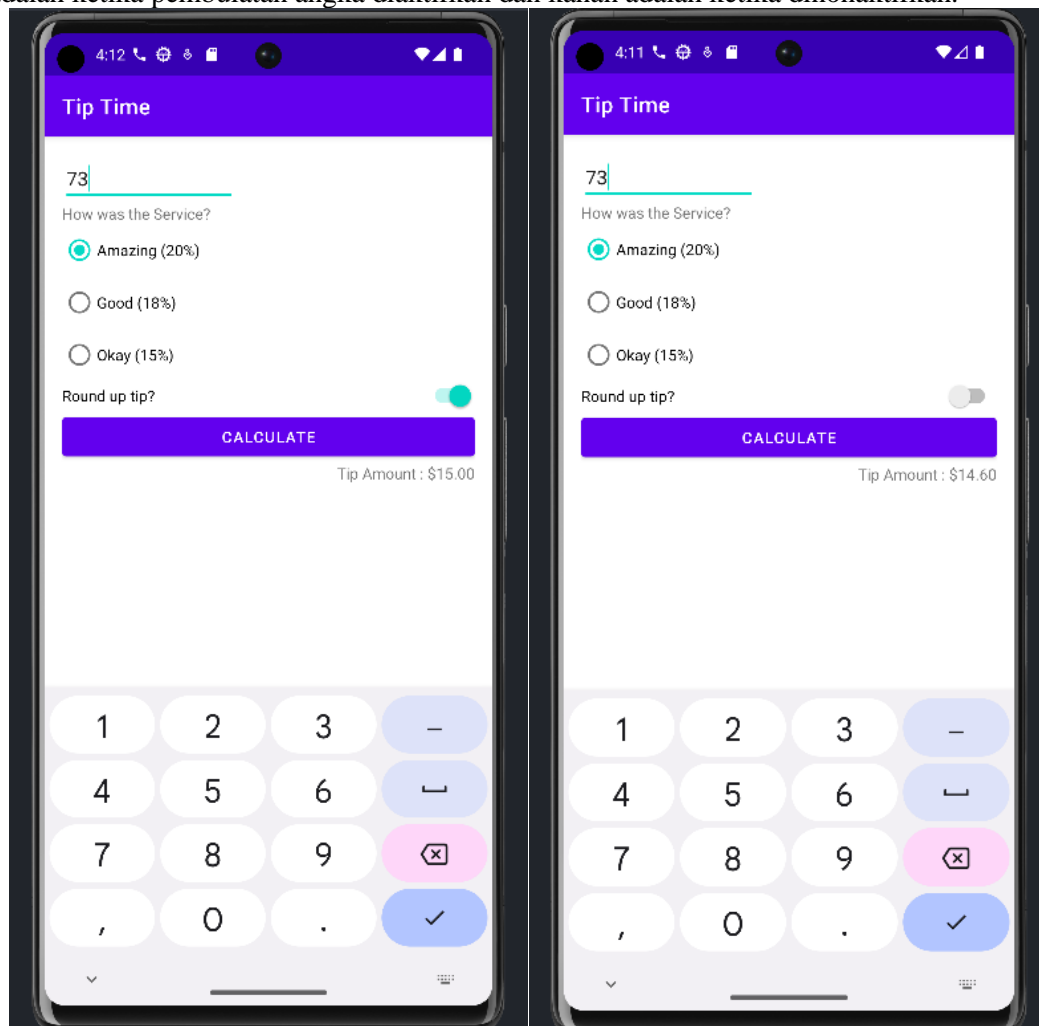
Adapun untuk output dari program yang telah dibuat adalah sebagai berikut :

1. Tampilan Awal ketika program di RUN



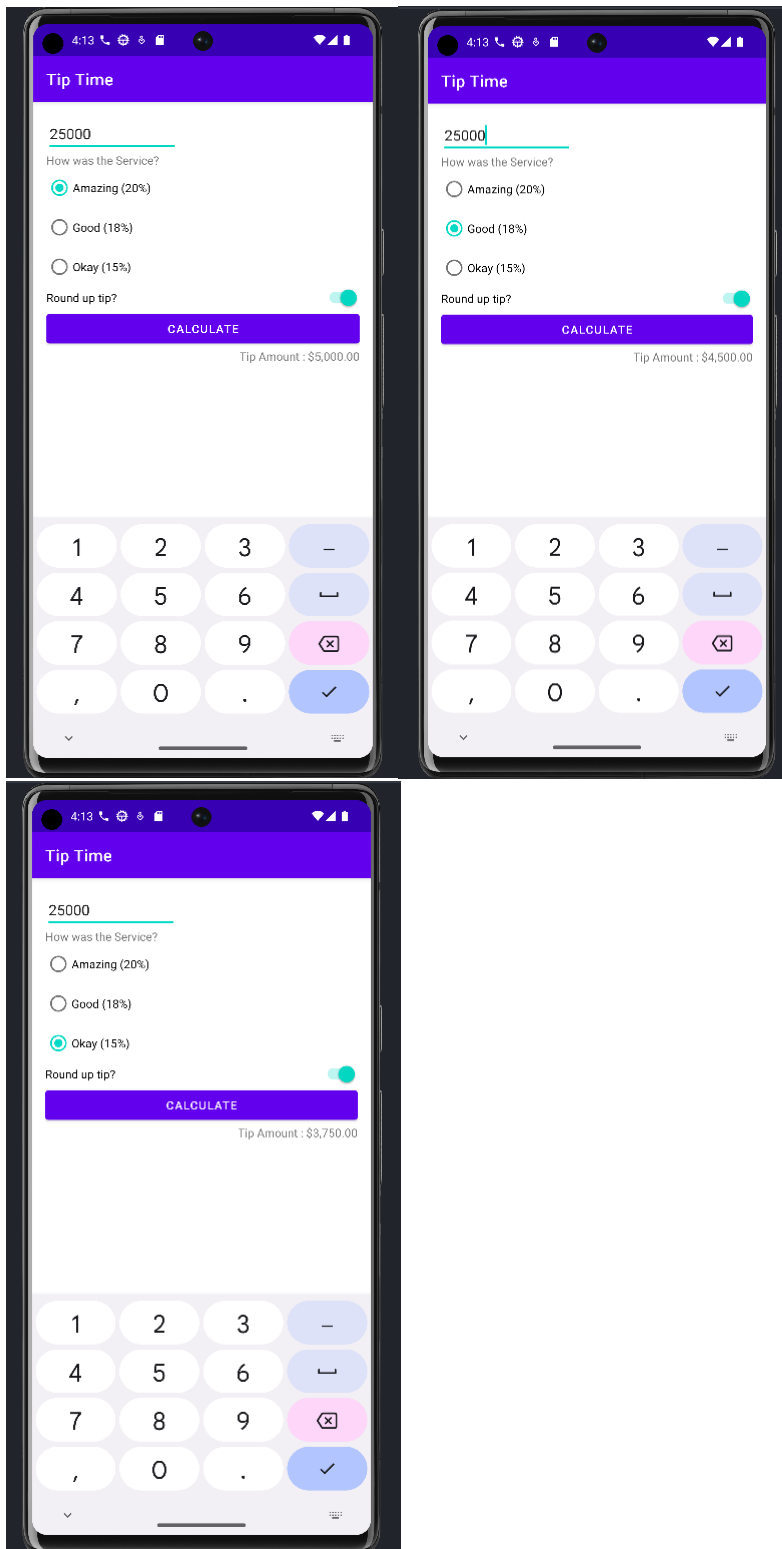
Gambar 1. Output program awal

2. Tampilan ketika memasukkan input angka dan menekan Calculate Button. Gambar kiri adalah ketika pembulatan angka diaktifkan dan kanan adalah ketika dinonaktifkan.



Gambar 2. Calculate button clicked

3. Tampilan ketika memasukkan input angka dan menggunakan pilihan radio button satu persatu.



Gambar 3. Output all radio button selected

C. Pembahasan

Pada praktikum ini membahas bagaimana membuat program calculator sederhana untuk menghitung uang tip. Program ini dijalankan dengan menggunakan bahasa kotlin dan Data Binding untuk menghubungkan layout XML pada Android Studio. Adapun penjelasan dari code-code yang ada pada program ini adalah :

MainActivity.kt

- Pada baris ke-3, dilakukan import library yang diperlukan untuk membangun aplikasi ini, termasuk **androidx.appcompat.app.AppCompatActivity** yang merupakan activity dasar dari aplikasi Android.
- Baris ke 4 - 7, merupakan deklarasi nama package dari aplikasi, dan import untuk library yang akan digunakan untuk activity.
- Baris ke 9 - 11, adalah deklarasi kelas MainActivity yang merupakan kelas dasar dari aplikasi.
- Baris ke 14 - 16, adalah deklarasi sebuah variabel **binding** dengan tipe **ActivityMainBinding**. Variabel ini digunakan untuk mengambil referensi atribut yang telah dibuat pada file XML.
- Baris ke 19 - 25, adalah fungsi **onCreate** yang digunakan untuk menginisialisasi objek binding dan mengatur layout activity yang akan digunakan pada aplikasi. Fungsi ini juga menambahkan event listener pada button **calculateButton** yang akan memanggil fungsi **calculateTip()** ketika diklik.
- Baris ke 28 - 41, adalah fungsi **calculateTip()** yang digunakan untuk memproses perhitungan tip berdasarkan input dari pengguna seperti jumlah biaya dari service, persentase tip, dan pilihan untuk membulatkan tip ke atas.

Dalam fungsi **calculateTip()**, terdapat beberapa langkah yang dilakukan untuk memproses perhitungan tip, yaitu:

1. Mengambil input dari pengguna berupa jumlah biaya dari service pada text field **costOfService**.
2. Mengubah input tersebut menjadi tipe data **Double** dengan menggunakan fungsi **toDoubleOrNull()**.
3. Jika input dari pengguna tidak valid, maka output pada **tipResult** akan dikosongkan dan fungsi dihentikan menggunakan **return**.
4. Jika input dari pengguna valid, maka akan dihitung persentase tip berdasarkan opsi yang dipilih pada radio button **tipOptions**.
5. Jika opsi untuk membulatkan tip ke atas diaktifkan, maka tip akan dibulatkan ke atas menggunakan fungsi **kotlin.math.ceil()**.
6. Nilai tip yang dihasilkan akan diformat menggunakan **NumberFormat.getCurrencyInstance()** agar sesuai dengan format mata uang yang digunakan oleh pengguna.
7. Hasil perhitungan tip akan ditampilkan pada **tipResult** menggunakan fungsi **getString()**.

activity_main.xml :

1. EditText
Kolom teks yang terdapat di layout ini adalah EditText dengan id "cost_of_service". EditText ini digunakan untuk memasukkan angka yang merepresentasikan biaya pelayanan yang diterima.
2. TextView
Teks yang tampil pada layar adalah TextView dengan id "service_question". TextView ini memberikan pertanyaan kepada pengguna mengenai kualitas pelayanan yang diterima.
3. RadioGroup dan RadioButton
RadioGroup dan RadioButton digunakan untuk memberikan opsi tip yang dapat dipilih oleh pengguna. Terdapat 3 opsi tip yang disediakan, yaitu 15%, 18%, dan 20%.
4. Switch
Switch dengan id "round_up_switch" digunakan untuk membulatkan jumlah tip ke angka yang lebih besar jika diaktifkan oleh pengguna.
5. Button
Button dengan id "calculate_button" digunakan untuk menghitung jumlah tip berdasarkan nilai yang diinput oleh pengguna.
6. TextView
TextView dengan id "tip_result" digunakan untuk menampilkan hasil perhitungan jumlah tip yang diberikan kepada pelayan. Tampilan tip yang ditampilkan akan diformat ke dalam mata uang yang sesuai dengan setting mobile user.