

**LAPORAN PRAKTIKUM  
PEMROGRAMAN MOBILE  
MODUL 2**



**ANDROID LAYOUT WITH COMPOSE**

**Oleh:**

**Harry Pratama Yunus    NIM. 2310817210010**

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

**LEMBAR PENGESAHAN**  
**LAPORAN PRAKTIKUM PEMROGRAMAN I**  
**MODUL 2**

Laporan Praktikum Pemrograman Mobile Modul 2: Android Layout With Compose ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman Mobile. Laporan Praktikum ini dikerjakan oleh:

Nama Praktikan : Harry Pratama Yunus  
NIM : 2310817210010

Menyetujui,  
Asisten Praktikum

Mengetahui,  
Dosen Penanggung Jawab Praktikum

Muhammad Raka Azwar  
NIM. 2210817210012

Andreyan Rizky Baskara, S.Kom., M.Kom.  
NIP. 19930703 201903 01 011

## DAFTAR ISI

LEMBAR PENGESAHAN .....	2
DAFTAR ISI .....	3
DAFTAR GAMBAR.....	4
DAFTAR TABEL .....	5
SOAL 1.....	6
A. Source Code.....	8
B. Output Program .....	16
C. Pembahasan .....	24
SOAL 2.....	27
A. Pembahasan .....	27
Tautan Git .....	28

## DAFTAR GAMBAR

Gambar 1. Tampilan awal aplikasi .....	6
Gambar 2. Tampilan Pilihan Presentasi Tip .....	7
Gambar 3. Tampilan Aplikasi Setelah Dijalankan .....	7
Gambar 4. Screenshot Hasil Jawaban Soal 1 XML bagian a .....	16
Gambar 5. Screenshot Hasil Jawaban Soal 1 XML bagian b .....	17
Gambar 6. Screenshot Hasil Jawaban Soal 1 XML bagian c .....	18
Gambar 7. Screenshot Hasil Jawaban Soal 1 XML bagian d .....	19
Gambar 8. Screenshot Hasil Jawaban Soal 1 compose bagian a .....	20
Gambar 9. Screenshot Hasil Jawaban Soal 1 compose bagian b.....	21
Gambar 10 . Screenshot Hasil Jawaban Soal 1 compose bagian c .....	22
Gambar 11. Screenshot Hasil Jawaban Soal 1 compose bagian d.....	23

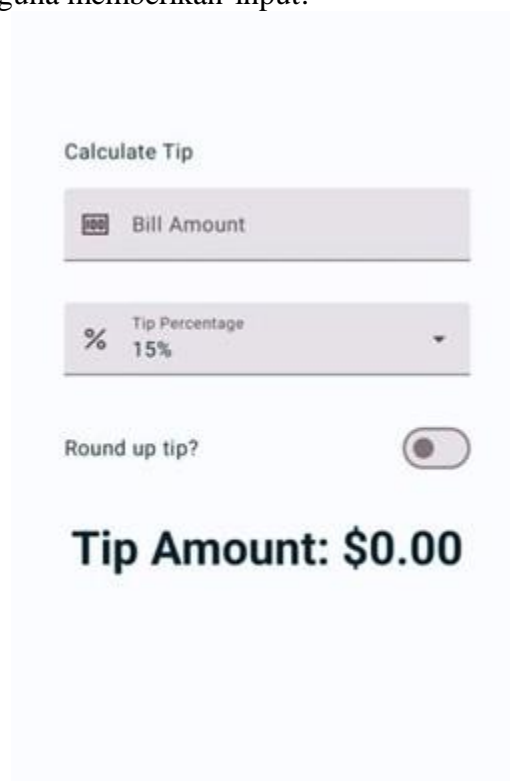
## DAFTAR TABEL

Table 1 source code soal 1 mainactivity.kt.....	8
Table 2 source code soal 1 acivity_main.xml.....	9
Table 3 source code soal 1 mainactivity.kt(compose) .....	12

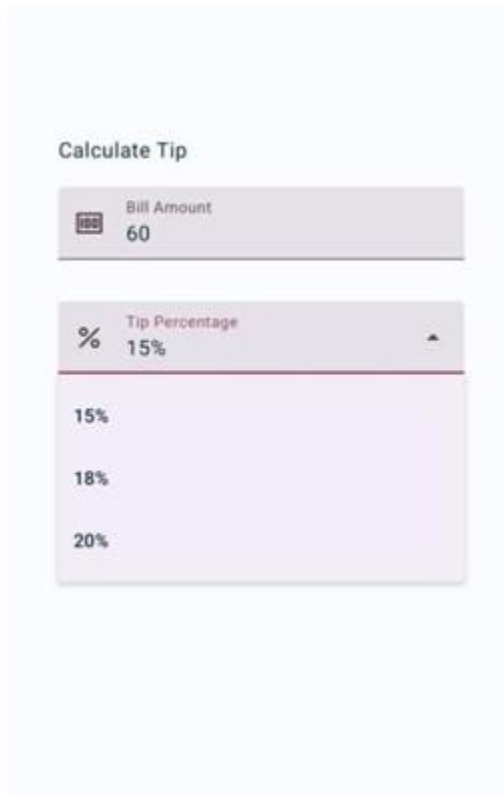
## SOAL 1

### Soal Praktikum:

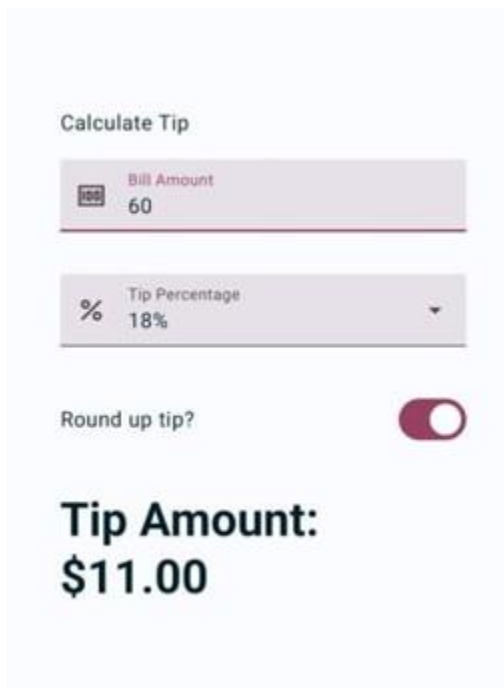
1. Buatlah sebuah aplikasi kalkulator tip menggunakan XML dan Jetpack Compose yang dirancang untuk membantu pengguna menghitung tip yang sesuai berdasarkan total biaya layanan yang mereka terima. Fitur-fitur yang diharapkan dalam aplikasi ini mencakup:
  - a. Input biaya layanan: Pengguna dapat memasukkan total biaya layanan yang diterima dalam bentuk nominal.
  - b. Pilihan persentase tip: Pengguna dapat memilih persentase tip yang diinginkan.
  - c. Pengaturan pembulatan tip: Pengguna dapat memilih untuk membulatkan tip ke angka yang lebih tinggi.
  - d. Tampilan hasil: Aplikasi akan menampilkan jumlah tip yang harus dibayar secara langsung setelah pengguna memberikan input.



Gambar 1. Tampilan awal aplikasi



Gambar 2. Tampilan Pilihan Presentasi Tip



Gambar 3. Tampilan Aplikasi Setelah Dijalankan

## A. Source Code

### XML :

### MainActivity.kt

Table 1 source code soal 1 mainactivity.kt

```
1 package com.pemrogamanmobile.mycalculatortipapp
2
3 import android.os.Bundle
4 import android.text.Editable
5 import android.text.TextWatcher
6 import android.widget.*
7 import androidx.appcompat.app.AppCompatActivity
8 import kotlin.math.ceil
9
10 class MainActivity : AppCompatActivity() {
11
12     private lateinit var billAmountInput: EditText
13     private lateinit var tipPercentageDropdown:
14     AutoCompleteTextView
15     private lateinit var roundUpSwitch: Switch
16     private lateinit var tipResult: TextView
17
18     private val tipOptions = listOf("15%", "18%", "20%")
19
20     override fun onCreate(savedInstanceState: Bundle?) {
21         super.onCreate(savedInstanceState)
22         setContentView(R.layout.activity_main)
23
24         billAmountInput = findViewById(R.id.billAmountInput)
25         tipPercentageDropdown =
26         findViewById(R.id.tipPercentageDropdown)
27         roundUpSwitch = findViewById(R.id.roundUpSwitch)
28         tipResult = findViewById(R.id.tipResult)
29
30         val adapter = ArrayAdapter(this,
31         android.R.layout.simple_dropdown_item_1line, tipOptions)
32         tipPercentageDropdown.setAdapter(adapter)
33
34         tipPercentageDropdown.setText(tipOptions[0], false)
35
36         tipPercentageDropdown.setOnClickListener {
37             tipPercentageDropdown.showDropDown()
38         }
39
40         tipPercentageDropdown.setOnItemClickListener { _, _, _
41         ->
42             calculateTip()
43         }
44
45         billAmountInput.addTextChangedListener(object :
46         TextWatcher {
```



42	override fun afterTextChanged(s: Editable?) {	
43	calculateTip()	
44	}	
45		
46	override fun beforeTextChanged(s: CharSequence?,	
	start: Int, count: Int, after: Int) {}	
47	override fun onTextChanged(s: CharSequence?, start:	
	Int, before: Int, count: Int) {}	
48	})	
49		
50	roundUpSwitch.setOnCheckedChangeListener { _, _ ->	
51	calculateTip()	
52	}	
53		
54	calculateTip()	
55	}	
56		
57	private fun calculateTip() {	
58	val billAmount	=
	billAmountInput.text.toString().toDoubleOrNull() ?: 0.0	
59	val tipPercentText	=
	tipPercentageDropdown.text.toString()	
60	val tipPercent	=
	tipPercentText.removeSuffix("%").toDoubleOrNull() ?: 0.0	
61		
62	var tip = billAmount * (tipPercent / 100)	
63	if (roundUpSwitch.isChecked) {	
64	tip = ceil(tip)	
65	}	
66		
67	tipResult.text = "Tip Amount: \$%.2f".format(tip)	
68	}	
69	}	

## activity\_main.xml

Table 2 source code soal 1 activity\_main.xml

1	<?xml version="1.0" encoding="utf-8"?>
2	<ScrollView
3	xmlns:android="http://schemas.android.com/apk/res/android"
4	xmlns:app="http://schemas.android.com/apk/res-auto"
5	android:layout_width="match_parent"
6	android:layout_height="match_parent"
7	android:padding="24dp"
8	android:background="#F8FFFB">
9	
10	<LinearLayout
11	android:layout_width="match_parent"
12	android:layout_height="wrap_content"
13	android:gravity="center"
14	android:orientation="vertical">

```

15
16         <TextView
17             android:id="@+id/title"
18             android:layout_width="wrap_content"
19             android:layout_height="wrap_content"
20             android:layout_gravity="start"
21             android:layout_marginBottom="16dp"
22             android:text="Calculate Tip"
23             android:textColor="#090909"
24             android:textSize="20sp" />
25
26         <LinearLayout
27             android:layout_width="match_parent"
28             android:layout_height="70dp"
29             android:layout_marginBottom="16dp"
30             android:background="@drawable/box_background"
31             android:gravity="center_vertical"
32             android:orientation="horizontal"
33             android:padding="10dp">
34
35             <ImageView
36                 android:layout_width="24dp"
37                 android:layout_height="24dp"
38                 android:layout_gravity="center_vertical"
39                 android:src="@drawable/money" />
40
41
42         <com.google.android.material.textfield.TextInputLayout
43             android:layout_width="0dp"
44             android:layout_height="match_parent"
45             android:layout_marginStart="8dp"
46             android:layout_weight="1"
47             android:hint="Bill Amount"
48             app:boxBackgroundMode="none">
49
50
51         <com.google.android.material.textfield.TextInputEditText
52             android:id="@+id/billAmountInput"
53             android:layout_width="match_parent"
54             android:layout_height="wrap_content"
55
56             android:background="@android:color/transparent"
57             android:inputType="numberDecimal"
58             android:paddingVertical="8dp" />
59
60     </com.google.android.material.textfield.TextInputLayout>
61     </LinearLayout>
62
63     <LinearLayout
64         android:layout_width="match_parent"
65         android:layout_height="70dp"
66         android:layout_marginBottom="16dp"

```

```

67         android:background="@drawable/box_background"
68         android:gravity="center_vertical"
69         android:orientation="horizontal"
70         android:padding="10dp">
71
72         <ImageView
73             android:layout_width="24dp"
74             android:layout_height="24dp"
75             android:layout_gravity="center_vertical"
76             android:src="@drawable/percent" />
77
78
79     <com.google.android.material.textfield.TextInputLayout
80         android:layout_width="0dp"
81         android:layout_height="match_parent"
82         android:layout_marginStart="8dp"
83         android:layout_weight="1"
84         android:hint="Tip Percentage"
85         app:boxBackgroundMode="none">
86
87         <AutoCompleteTextView
88             android:id="@+id/tipPercentageDropdown"
89             android:layout_width="match_parent"
90             android:layout_height="match_parent"
91             android:gravity="center_vertical"
92
93             android:background="@android:color/transparent"
94             android:inputType="none"
95             android:dropDownWidth="match_parent"
96             android:paddingVertical="8dp" />
97
98     </com.google.android.material.textfield.TextInputLayout>
99     </LinearLayout>
100
101     <LinearLayout
102         android:layout_width="match_parent"
103         android:layout_height="50dp"
104         android:layout_marginBottom="24dp"
105         android:orientation="horizontal">
106
107         <TextView
108             android:layout_width="wrap_content"
109             android:layout_height="match_parent"
110             android:gravity="center_vertical"
111             android:text="Round up tip?"
112             android:textColor="#3C3A3A"
113             android:textSize="20dp" />
114
115         <Switch
116             android:id="@+id/roundUpSwitch"
117             android:layout_width="match_parent"
118             android:layout_height="match_parent" />

```

119	</LinearLayout>
120	
121	<TextView
122	android:id="@+id/tipResult"
123	android:layout_width="wrap_content"
	android:layout_height="wrap_content"
	android:text="Tip Amount: \$0.00"
	android:textColor="#000000"
	android:textSize="34dp"
	android:textStyle="bold" />
	</LinearLayout>
	</ScrollView>

## COMPOSE :

### MainActivity.kt

Table 3 source code soal 1 mainactivity.kt(compose)

1	package com.pemrogamanmobile.mycalculatortipapp
2	
3	import android.os.Bundle
4	import androidx.activity.ComponentActivity
5	import androidx.activity.compose.setContent
6	import androidx.compose.foundation.layout.Arrangement
7	import androidx.compose.foundation.layout.Column
8	import androidx.compose.foundation.layout.Row
9	import androidx.compose.foundation.layout.fillMaxSize
10	import androidx.compose.foundation.layout.fillMaxWidth
11	import androidx.compose.foundation.layout.padding
12	import androidx.compose.foundation.text.KeyboardOptions
13	import androidx.compose.material.icons.Icons
14	import androidx.compose.material.icons.filled.Money
15	import androidx.compose.material.icons.filled.Percent
16	import androidx.compose.material3.DropdownMenu
17	import androidx.compose.material3.DropdownMenuItem
18	import androidx.compose.material3.ExperimentalMaterial3Api
19	import androidx.compose.material3.ExposedDropdownMenuBox
20	import androidx.compose.material3.ExposedDropdownMenuDefaults
21	import androidx.compose.material3.Icon
22	import androidx.compose.material3.Text
23	import androidx.compose.material3.TextField
24	import androidx.compose.runtime.Composable
25	import androidx.compose.runtime.getValue
26	import androidx.compose.runtime.mutableStateOf
27	import androidx.compose.runtime.remember
28	import androidx.compose.runtime.setValue
29	import androidx.compose.ui.Alignment
30	import androidx.compose.ui.Modifier
31	import androidx.compose.ui.text.font.FontWeight
32	import androidx.compose.ui.text.input.KeyboardType

```

33 import androidx.compose.ui.unit.dp
34 import androidx.compose.ui.unit.sp
35
36
37 class MainActivity : ComponentActivity() {
38     override fun onCreate(savedInstanceState: Bundle?) {
39         super.onCreate(savedInstanceState)
40         setContent {
41             Mycalculatortip(
42                 )
43         }
44     }
45 }
46
47 @OptIn(ExperimentalMaterial3Api::class)
48 @Composable
49 fun Mycalculatortip() {
50     val options = listOf("15%", "18%", "20%")
51     var expanded by remember { mutableStateOf(false) }
52     var selectedOptionText by remember {
53         mutableStateOf(options[0]) }
54     var billAmount by remember { mutableStateOf("") }
55     var roundUp by remember { mutableStateOf(false) }
56
57     val tipPercentage =
58     selectedOptionText.removeSuffix("%").toDoubleOrNull() ?: 0.0
59     val bill = billAmount.toDoubleOrNull() ?: 0.0
60     var tip = bill * (tipPercentage / 100.0)
61     if (roundUp) {
62         tip = kotlin.math.ceil(tip)
63     }
64
65     androidx.compose.foundation.layout.Box(
66         modifier = Modifier
67             .fillMaxSize()
68             .padding(16.dp)
69     ) {
70         Column(
71             verticalArrangement = Arrangement.Center,
72             modifier = Modifier.fillMaxSize()
73         ) {
74             Text(
75                 text = "Calculate Tip",
76                 fontSize = 24.sp,
77                 modifier = Modifier
78                     .padding(bottom = 16.dp)
79             )
80
81             TextField(
82                 value = billAmount,
83                 onValueChange = { billAmount = it },
84                 label = { Text("Bill Amount") },

```

```

82         leadingIcon = {
83             Icon(Icons.Filled.Money,
contentDescription = "Money Icon")
84         },
85         keyboardOptions                                =
86         KeyboardOptions.Default.copy(keyboardType      =
KeyboardType.Number),
87         singleLine = true,
88         modifier = Modifier
89             .fillMaxWidth()
90             .padding(bottom = 16.dp),
91         colors                                          =
92         ExposedDropdownMenuDefaults.textFieldColors()
93     )
94
95     ExposedDropdownMenuBox(
96         expanded = expanded,
97         onExpandedChange = { expanded = !expanded },
98         modifier = Modifier
99             .fillMaxWidth()
100             .padding(bottom = 16.dp)
101     ) {
102         TextField(
103             readOnly = true,
104             value = selectedOptionText,
105             onValueChange = {},
106             label = { Text("Tip Percentage") },
107             leadingIcon = {
108                 Icon(Icons.Filled.Percent,
contentDescription = "Percent Icon")
109             },
110             trailingIcon = {
111                 ExposedDropdownMenuDefaults.TrailingIcon(expanded = expanded)
112             },
113             modifier = Modifier
114                 .menuAnchor()
115                 .fillMaxWidth(),
116             colors                                          =
117             ExposedDropdownMenuDefaults.textFieldColors()
118         )
119
120         DropdownMenu(
121             expanded = expanded,
122             onDismissRequest = { expanded = false },
123             modifier = Modifier.fillMaxWidth()
124         ) {
125             options.forEach { selectionOption ->
126                 DropdownMenuItem(
127                     text = { Text(selectionOption) },
128                     onClick = {

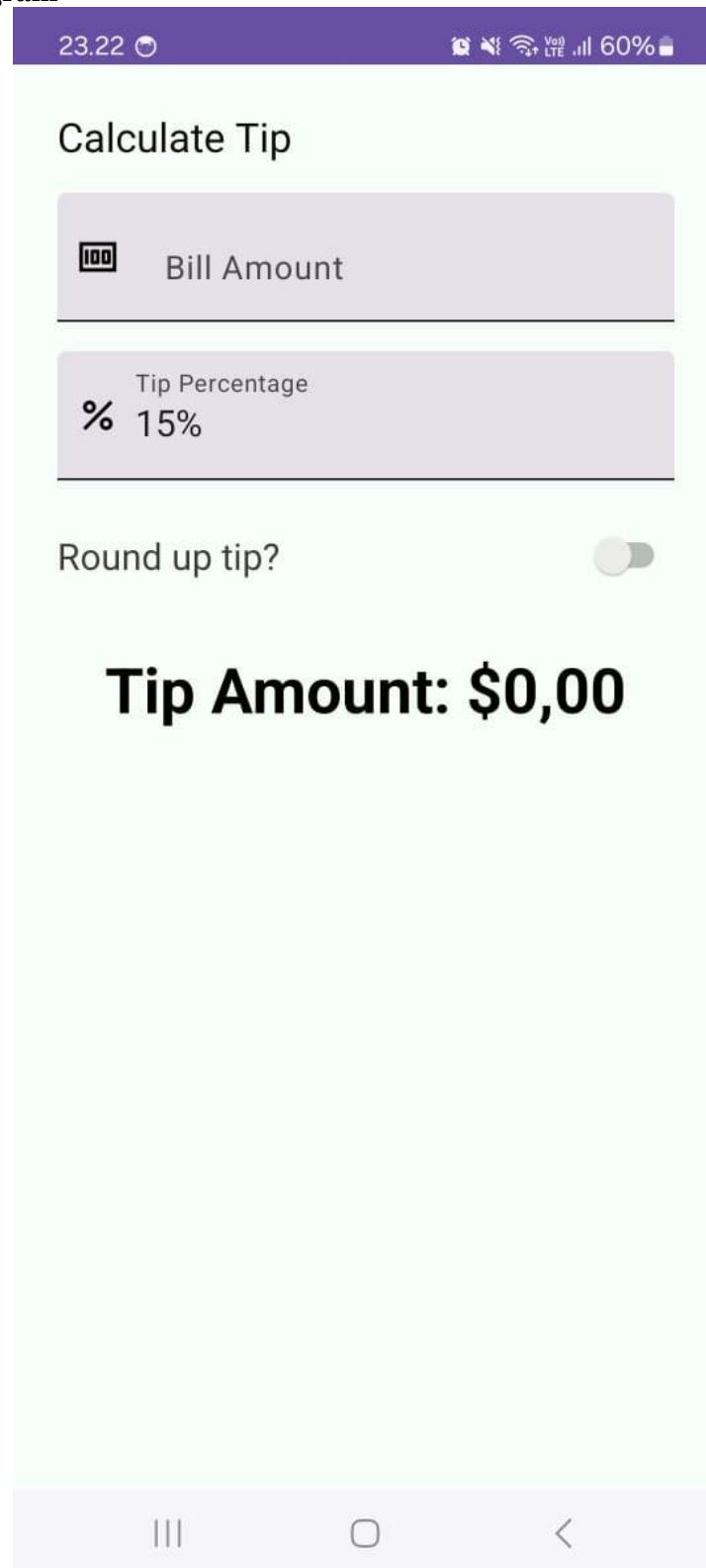
```

```

128                                     selectedOptionText           =
129 selectionOption
130                                     expanded = false
131                                     }
132                                     )
133                                     }
134                                     }
135                                     }
136
137             Row(
138                 horizontalArrangement
139 Arrangement.SpaceBetween,
140                 modifier = Modifier
141                     .fillMaxWidth()
142                     .padding(vertical = 16.dp)
143             ) {
144                 Text(text = "Round up tip?", fontSize = 20.sp)
145                 androidx.compose.material3.Switch(
146                     checked = roundUp,
147                     onChangeChange = { roundUp = it }
148                 )
149
150                 Text(
151                     text = "Tip Amount: $${"%.2f".format(tip)}",
152                     fontSize = 38.sp,
153                     fontWeight = FontWeight.Bold,
154                     modifier = Modifier
155                         .padding(top = 16.dp)
156                         .align(Alignment.CenterHorizontally)
157                 )
158             }
159 }

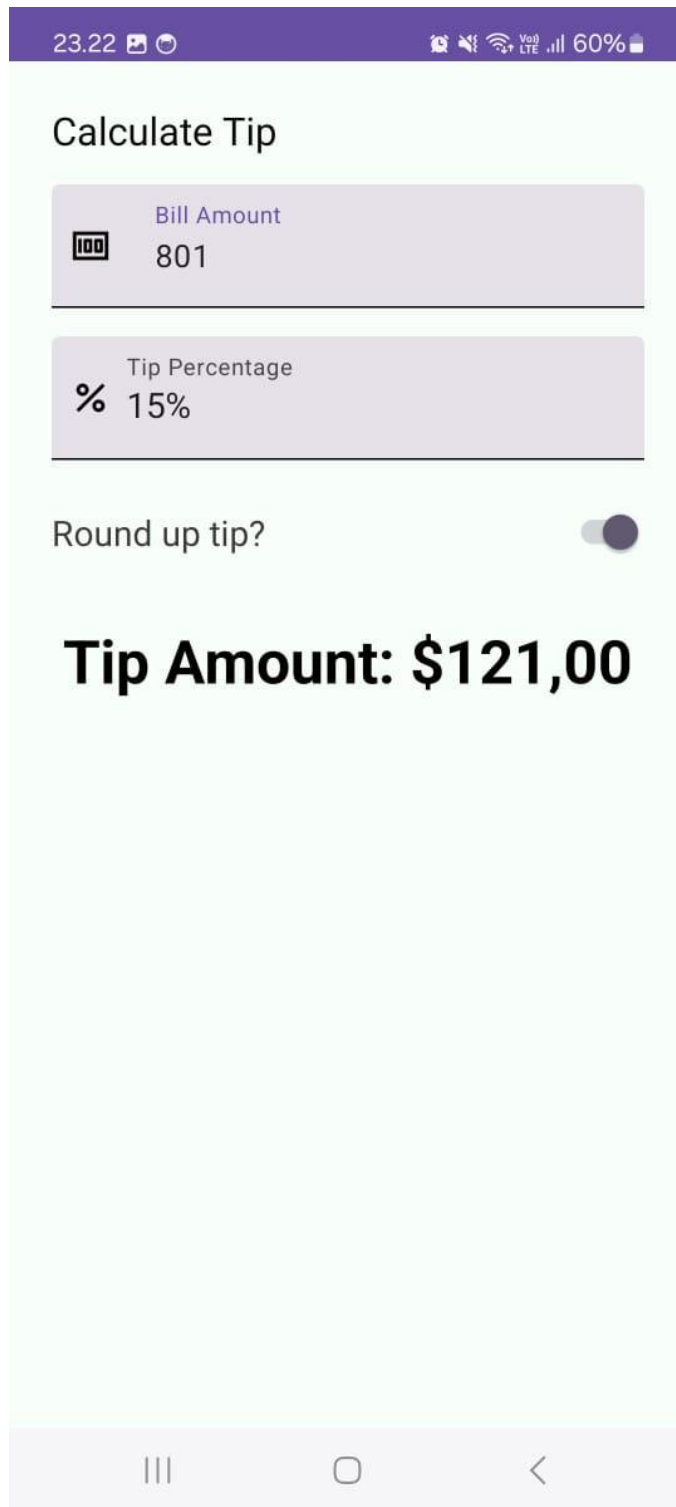
```

## B. Output Program

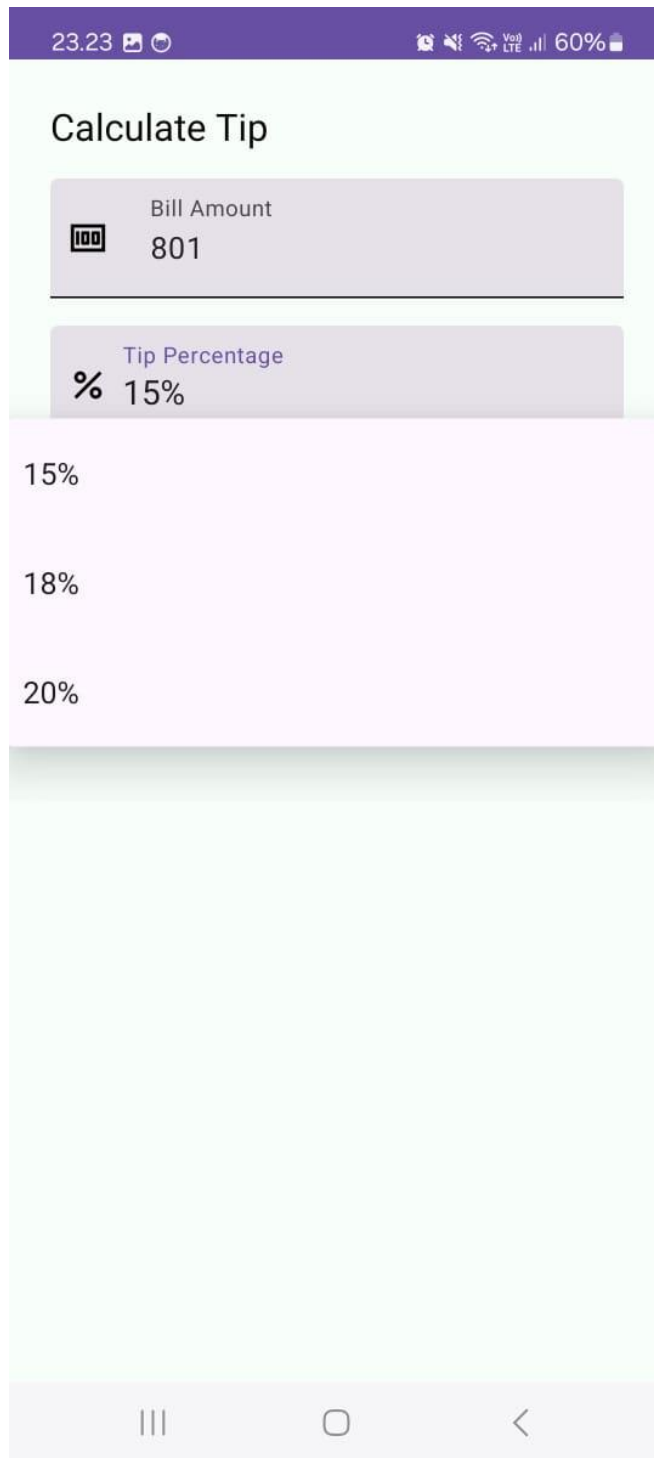


Gambar 4. Screenshot Hasil Jawaban Soal 1 XML bagian a

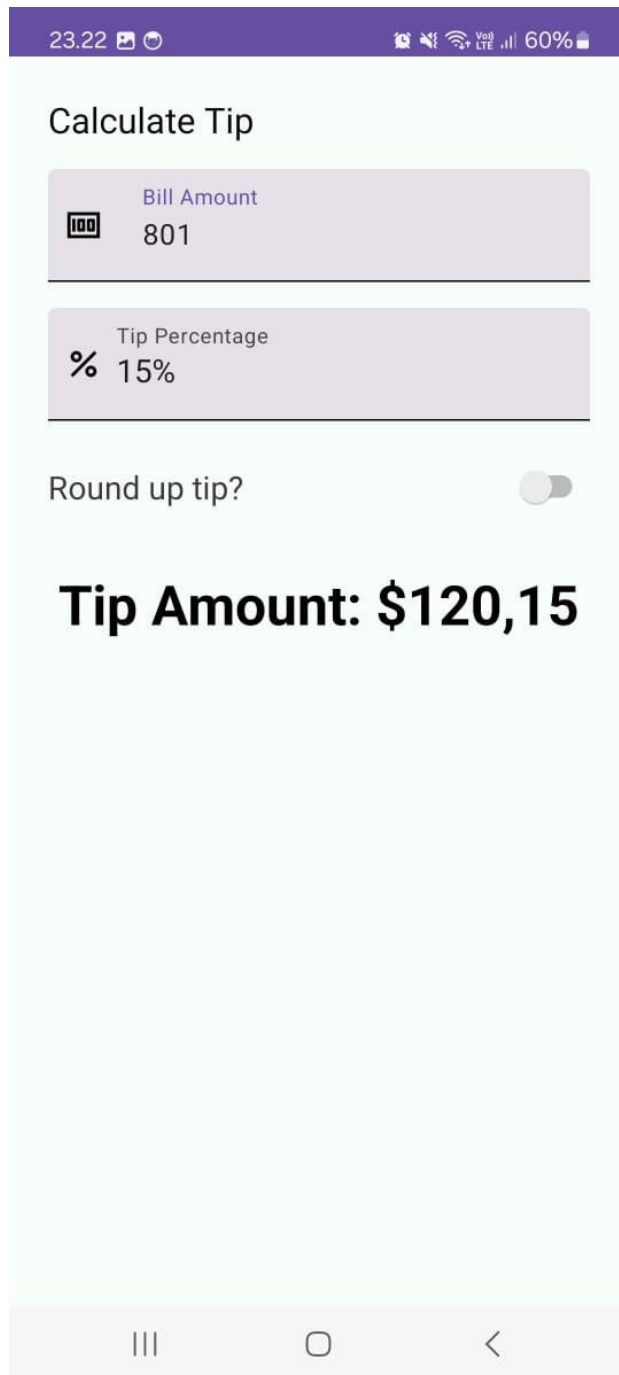




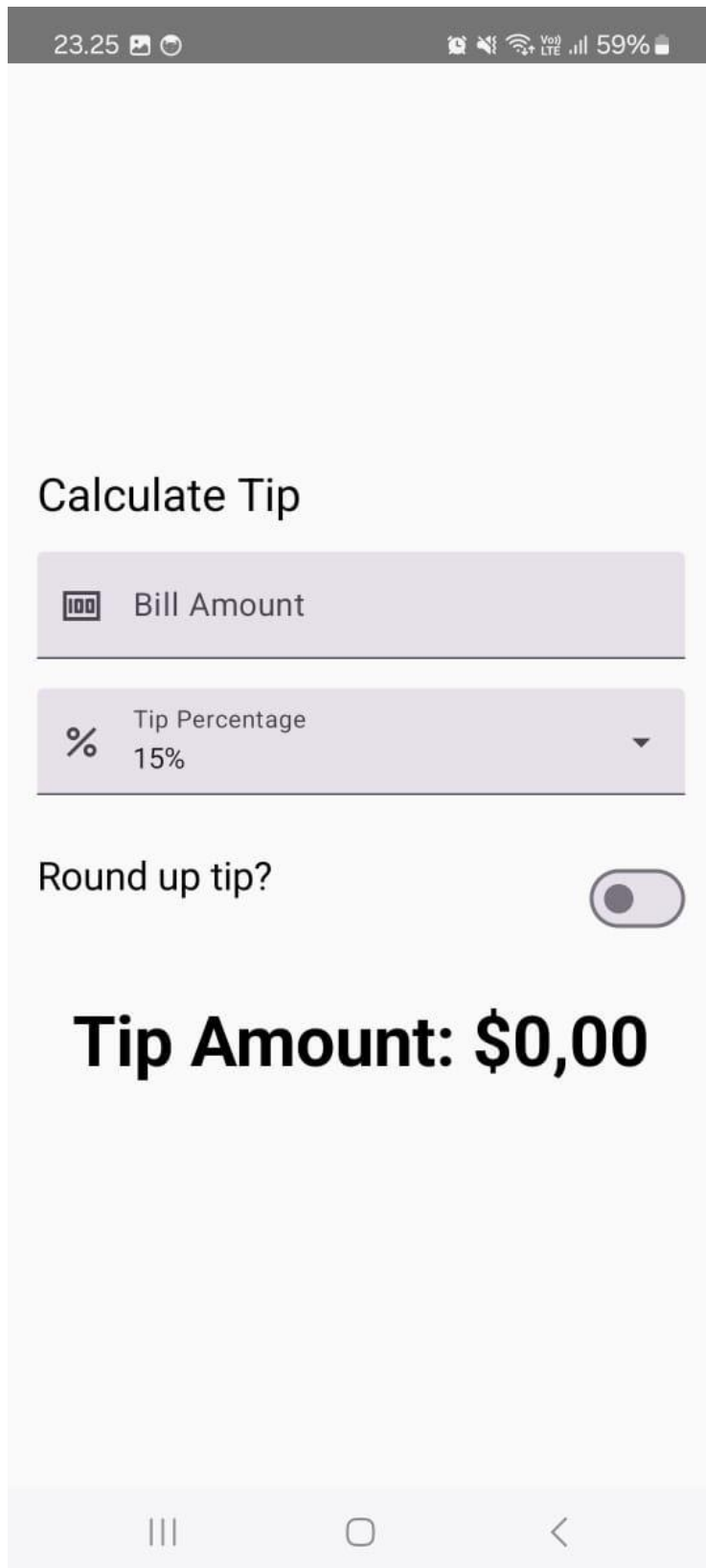
*Gambar 5. Screenshot Hasil Jawaban Soal 1 XML bagian b*



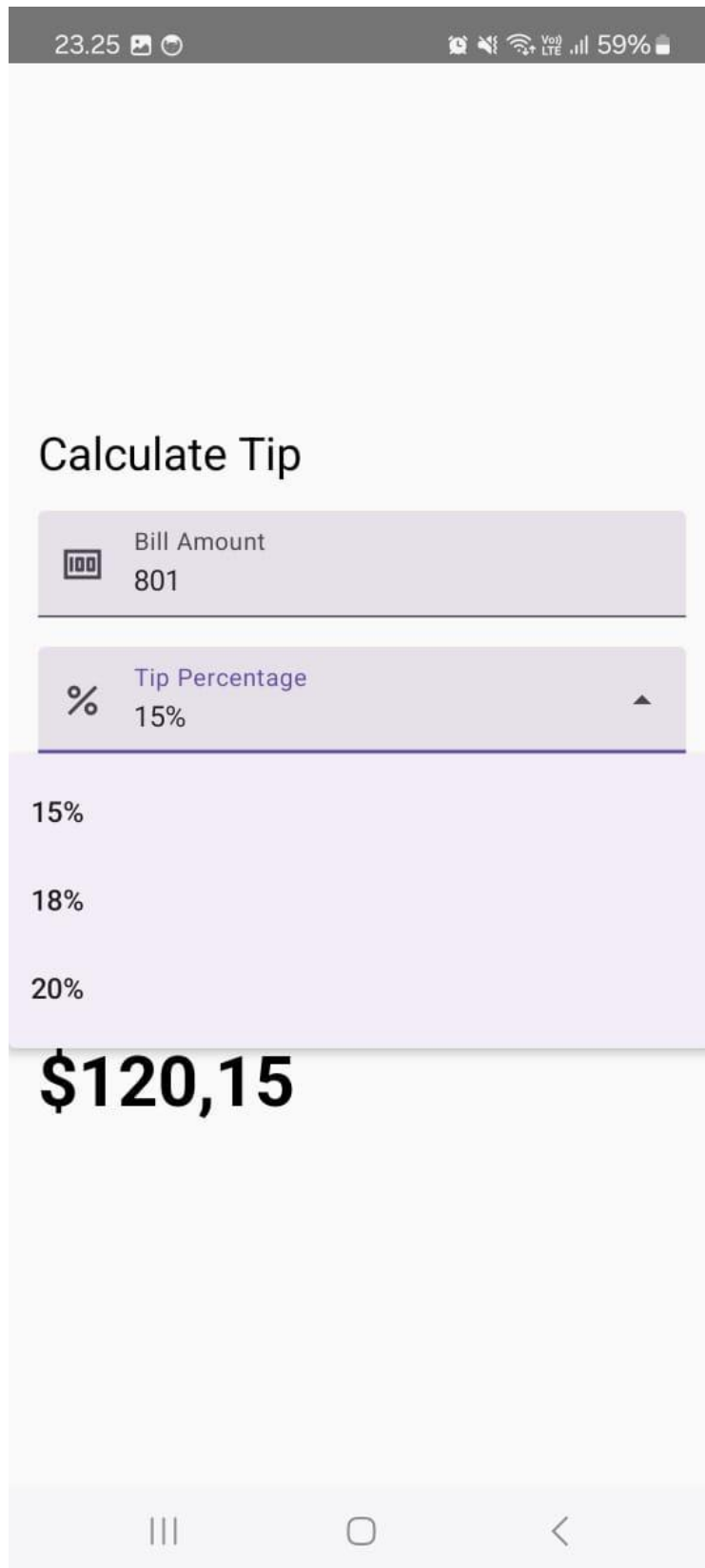
Gambar 6. Screenshot Hasil Jawaban Soal 1 XML bagian c



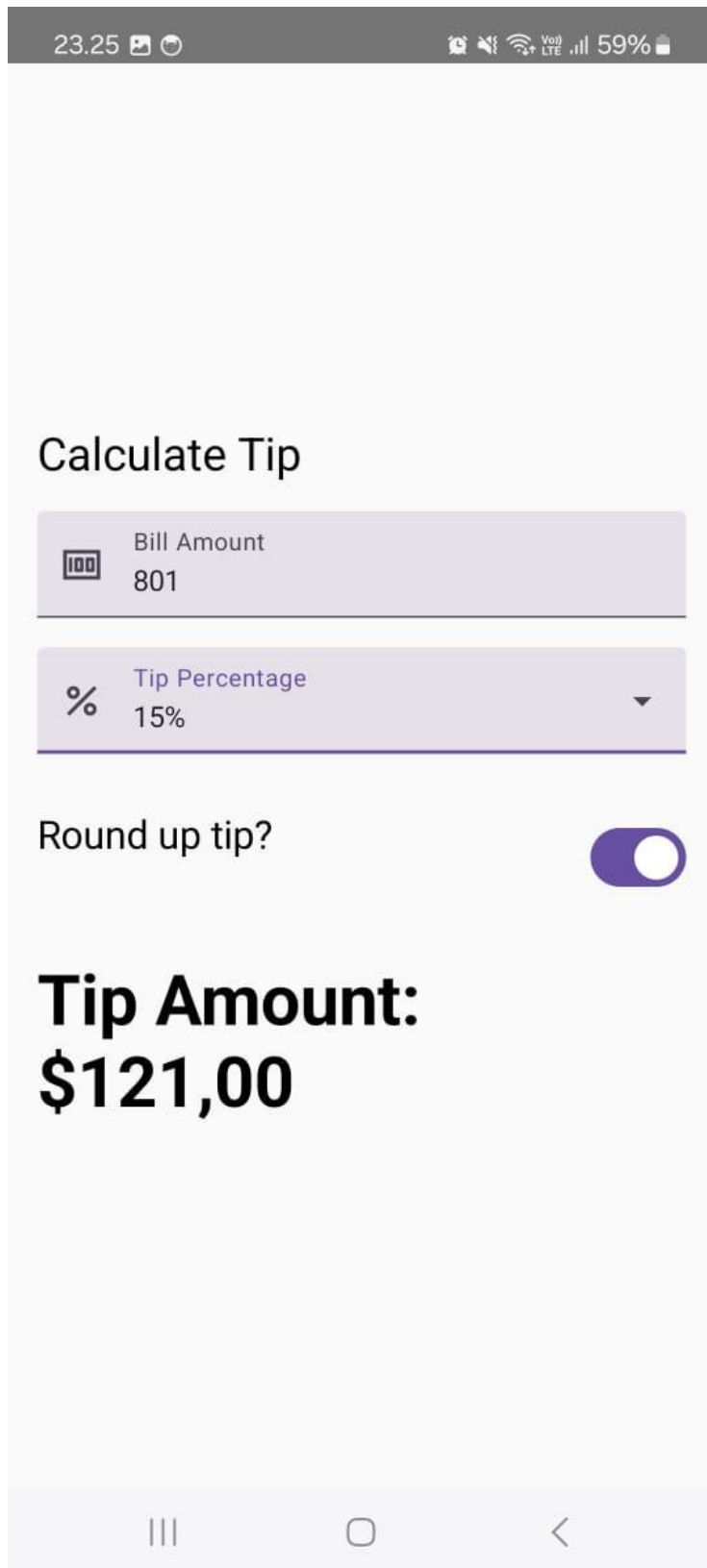
Gambar 7. Screenshot Hasil Jawaban Soal 1 XML bagian d



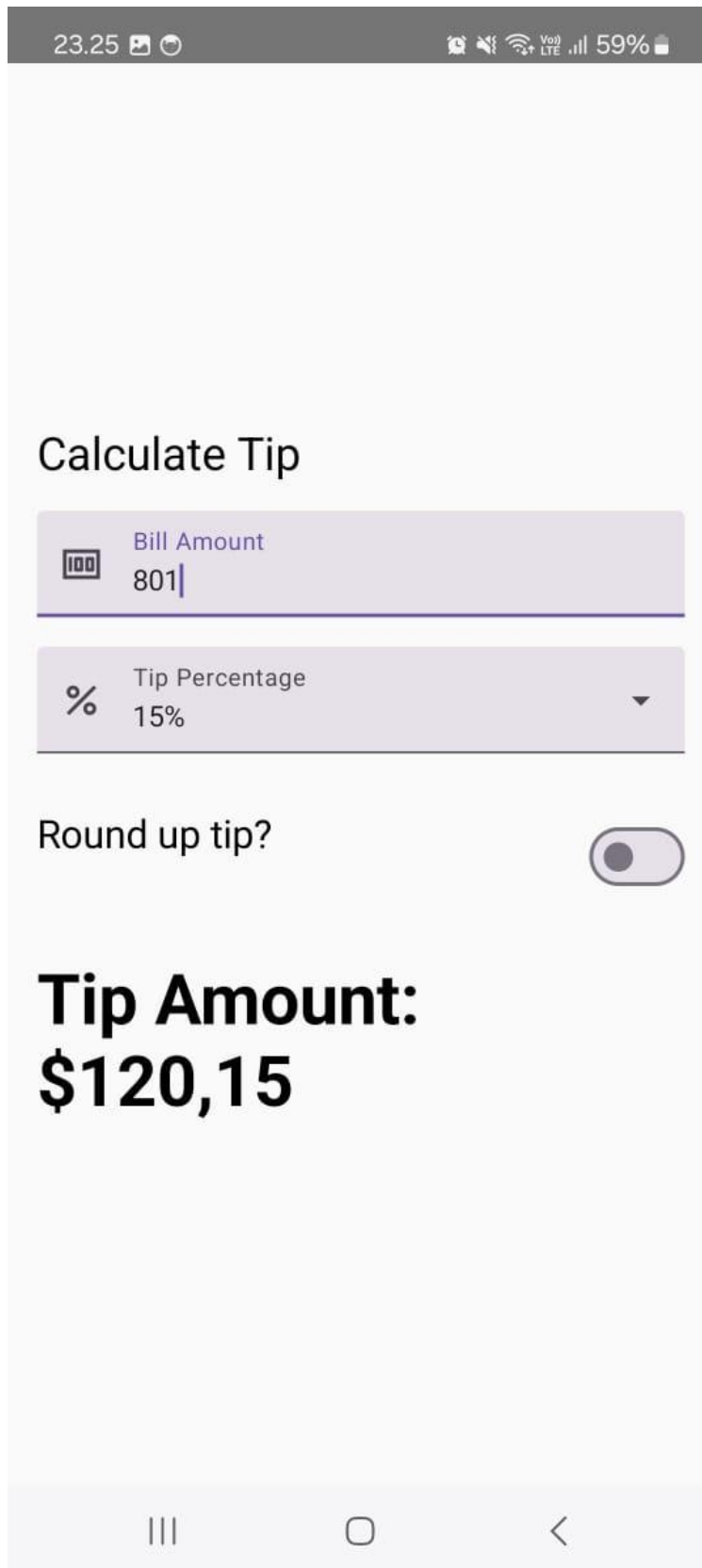
Gambar 8. Screenshot Hasil Jawaban Soal 1 compose bagian a



Gambar 9. Screenshot Hasil Jawaban Soal 1 compose bagian b



Gambar 10 . Screenshot Hasil Jawaban Soal 1 compose bagian c



Gambar 11. Screenshot Hasil Jawaban Soal 1 compose bagian d

## **C. Pembahasan**

### **XML :**

#### **MainActivity.kt:**

Pada baris ke [1] terdapat package `com.pemrogramanmobile.mycalculatoripapp` yang berfungsi sebagai nama paket dari aplikasi dan tempat menyimpan file.

Pada baris ke [3] sampai [8], berfungsi untuk mengimport beberapa kelas yang akan digunakan, yaitu `Bundle`, `Editable`, `TextWatcher`, `AutoCompleteTextView`, `EditText`, `Switch`, `TextView`, `ArrayAdapter`, `AppCompatActivity` dan `ceil` dari `kotlin.math`.

Pada baris ke [10] terdapat class `MainActivity : AppCompatActivity()` yang berfungsi untuk membuat class `MainActivity` yang merupakan activity utama aplikasi ini.

Pada baris ke [12] sampai [15] berfungsi untuk membuat variabel yang akan menyimpan inputan pengguna seperti jumlah tagihan, dropdown pilihan tip, switch pembulatan, dan hasil tip. (`lateinit` digunakan untuk menunda inisialisasi sebelum digunakan).

Pada baris ke [17] terdapat `private val tipOptions = listOf("15%", "18%", "20%")` yang berfungsi untuk membuat daftar pilihan persentase tip yang akan ditampilkan di dropdown.

Pada baris ke [19] sampai [21] berfungsi untuk memanggil activity yang pertama kali dibuat, kemudian juga menampilkan layout yang diambil dari `activity_main.xml`.

Pada baris ke [23] sampai [26] berfungsi untuk menghubungkan variabel di kode ini ke komponen yang ada di file xml berdasarkan id-nya.

Pada baris ke [28] berfungsi untuk membuat adapter yang menampilkan daftar persentase tip pada dropdown.

Pada baris ke [29] berfungsi untuk menetapkan adapter tersebut ke dropdown `tipPercentageDropdown`.

Pada baris ke [31] berfungsi untuk mengatur nilai default dari dropdown menjadi "15%" saat aplikasi dibuka pertama kali.

Pada baris ke [33] sampai [35] berfungsi untuk menampilkan pilihan dropdown ketika `tipPercentageDropdown` diklik oleh pengguna.

Pada baris ke [37] sampai [39] berfungsi untuk memanggil fungsi `calculateTip()` saat pengguna memilih salah satu pilihan dari dropdown.

Pada baris ke [41] sampai [48] berfungsi untuk memantau perubahan teks pada input jumlah tagihan, dan setiap kali teks berubah, fungsi `calculateTip()` akan dipanggil.

Pada baris ke [50] sampai [52] berfungsi untuk memantau status switch pembulatan, jika pengguna mengaktifkan atau menonaktifkan switch, maka fungsi `calculateTip()` akan dijalankan.

Pada baris ke [54] terdapat `calculateTip()` yang berfungsi untuk menghitung nilai tip secara langsung ketika aplikasi pertama kali dibuka.

Pada baris ke [58] berfungsi untuk mengambil nilai dari input tagihan dan mengubahnya menjadi angka, jika kosong maka diisi dengan nilai 0.0.

Pada baris ke [59] sampai [60] berfungsi untuk mengambil persentase tip dari dropdown dan menghapus karakter "%" lalu mengubahnya menjadi angka desimal.

Pada baris ke [62] berfungsi untuk menghitung nilai tip berdasarkan jumlah tagihan dan persentase tip.

Pada baris ke [63] sampai [65] berfungsi untuk membulatkan nilai tip ke atas jika switch pembulatan diaktifkan.

Pada baris ke [67] berfungsi untuk menampilkan hasil perhitungan tip ke dalam `TextView` dengan format dua angka di belakang koma.



### **activity\_main.xml:**

Pada baris ke [1] terdapat deklarasi `<?xml version="1.0" encoding="utf-8"?>` yang berfungsi untuk menyatakan versi XML yang digunakan.

Pada baris ke [2] sampai [7] terdapat elemen `ScrollView` yang berfungsi sebagai kontainer gulir vertikal agar isi layout bisa di-scroll jika melebihi layar, dengan lebar dan tinggi disesuaikan ke layar penuh, padding 24dp, dan latar belakang berwarna #F8FFFB.

Pada baris ke [9] sampai [13] terdapat `LinearLayout` sebagai wadah utama yang menampung seluruh komponen, dengan orientasi vertikal dan gravitasi ke tengah.

Pada baris ke [15] sampai [23] terdapat `TextView` dengan id title yang menampilkan teks “Calculate Tip” sebagai judul, dengan warna teks #090909, ukuran teks 20sp, dan margin bawah 16dp.

Pada baris ke [25] sampai [56] terdapat `LinearLayout` yang digunakan sebagai wadah input jumlah tagihan, dengan tinggi 70dp, margin bawah 16dp, latar belakang dari drawable `box_background`, orientasi horizontal, dan padding 10dp. Di dalamnya terdapat `ImageView` dengan ikon uang dan `TextInputLayout` yang menampung `TextInputEditText` untuk memasukkan jumlah tagihan dengan inputType `numberDecimal` dan padding vertikal 8dp.

Pada baris ke [58] sampai [91] terdapat `LinearLayout` serupa yang digunakan untuk memilih persentase tip, dengan struktur dan gaya yang hampir sama seperti input jumlah tagihan, namun berisi ikon persen dan `AutoCompleteTextView` untuk menampilkan daftar pilihan tip.

Pada baris ke [93] sampai [111] terdapat `LinearLayout` dengan tinggi 50dp dan margin bawah 24dp, yang menampilkan `TextView` “Round up tip?” di sebelah kiri dan `Switch` dengan id `roundUpSwitch` di sebelah kanan yang digunakan untuk mengaktifkan atau menonaktifkan pembulatan nilai tip.

Pada baris ke [113] sampai [120] terdapat `TextView` dengan id `tipResult` yang menampilkan hasil perhitungan tip awal yaitu “Tip Amount: \$0.00” dengan ukuran teks 34sp, tebal (bold), dan warna teks hitam.

### **COMPOSE :**

#### **MainActivity.kt:**

Pada baris ke [1] terdapat package `com.pemrogramanmobile.mycalculatoripapp` yang berfungsi sebagai nama paket dari aplikasi dan tempat menyimpan file.

Pada baris ke [3] sampai [34], berfungsi untuk mengimpor beberapa komponen penting dari Android seperti `Bundle`, `layout`, `icon`, `textfield`, dan komponen `Material3` yang digunakan dalam aplikasi ini.

Pada baris ke [37] terdapat class `MainActivity : ComponentActivity()` yang berfungsi untuk membuat class `MainActivity` sebagai activity utama aplikasi ini.

Pada baris ke [38] sampai [44] berfungsi untuk membuat fungsi `onCreate` yang akan dijalankan saat activity dimulai, lalu menampilkan `composable` bernama `Mycalculatorip` sebagai isi tampilan aplikasi.

Pada baris ke [47] terdapat anotasi `@OptIn(ExperimentalMaterial3Api::class)` yang digunakan untuk mengaktifkan fitur eksperimental dari `Material3`.

Pada baris ke [49] sampai [159] berfungsi untuk membuat fungsi `composable` bernama `Mycalculatorip` yang akan menjadi UI utama aplikasi ini.

Pada baris ke [50] terdapat variabel `options` yang berisi daftar pilihan persentase tip.

Pada baris ke [51] sampai [53] digunakan untuk mendeklarasikan dan mengingat state dari `dropdown`, nilai pilihan yang dipilih, dan jumlah tagihan.

Pada baris ke [54] terdapat variabel `roundUp` yang menyimpan status switch untuk membulatkan tip atau tidak.

Pada baris ke [56] sampai [61] berfungsi untuk menghitung nilai tip berdasarkan persentase yang dipilih dan jumlah tagihan, lalu jika `roundUp` bernilai `true` maka tip akan dibulatkan ke atas.

Pada baris ke [63] sampai [67] digunakan untuk menambahkan layout Box sebagai wadah utama dengan padding 16dp.

Pada baris ke [68] sampai [71] berfungsi untuk membuat kolom dengan konten yang ditampilkan secara vertikal dan berada di tengah layar.

Pada baris ke [72] sampai [77] digunakan untuk menampilkan judul aplikasi yaitu "Calculate Tip" dengan ukuran font 24sp.

Pada baris ke [79] sampai [92] berfungsi untuk menampilkan TextField untuk input jumlah tagihan, yang disertai icon uang, hanya menerima input angka, dan tampilannya diatur dengan padding dan lebar penuh.

Pada baris ke [94] sampai [100] digunakan untuk membuat ExposedDropDownMenuBox sebagai tampilan pilihan persentase tip.

Pada baris ke [101] sampai [116] berfungsi untuk menampilkan TextField readonly yang menunjukkan pilihan persentase, dengan icon persen, trailing icon dropdown, dan menggunakan warna default.

Pada baris ke [118] sampai [132] digunakan untuk menampilkan daftar DropdownMenuItem dari setiap item yang ada di list options, yang akan menutup menu dan menyimpan pilihan saat diklik.

Pada baris ke [136] sampai [147] digunakan untuk menampilkan Row yang berisi teks "Round up tip?" dan switch untuk mengaktifkan pembulatan, dengan jarak antar item disesuaikan.

Pada baris ke [149] sampai [156] digunakan untuk menampilkan hasil kalkulasi tip dalam format dua angka desimal, dengan ukuran font 38sp, tebal, dan berada di tengah secara horizontal.

## SOAL 2

### Soal Praktikum:

2. Jelaskan perbedaan dari implementasi XML dan Jetpack Compose beserta kelebihan dan kekurangan dari masing-masing implementasi.

### A. Pembahasan

Perbedaan dari implementasi XML dan Jetpack Compose, pada XML code bersifat imperativ Dimana tiap fungsi harus dibuat sendiri supaya membentuk suatu box yang diinginkan, sedangkan untuk compose lebih ke declarative Dimana code sudah banyak tersedia beberapa fungsi yang berguna untuk mengerjakan soal seperti `ExposedDropdownMenuBox`.

#### Kelebihan

XML : lebih fleksibel dalam membentuk tampilannya.

COMPOSE : lebih cepat dalam pengerjaan codenya.

#### Kekurangan

XML : lebih memakan waktu dalam pengerjaannya.

COMPOSE : tidak terlalu fleksible dibandingkan compose.

## **Tautan Git**

Berikut adalah tautan untuk source code yang telah dibuat.

[https://github.com/Harry154git/Pemrogaman\\_mobile/tree/main/Modul\\_2](https://github.com/Harry154git/Pemrogaman_mobile/tree/main/Modul_2)