

## **Project UTS Android Aplikasi Kalkulator**



Dosen Pengampu:

Rizal M.Kom.

Disusun Oleh:

1. Riko Febrian (11220910000004)
2. Muhammad Alif Ramadhani (11220910000021)
3. Andika Pratama (11220910000018)

**TEKNIK INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH JAKARTA**

**2024**

## 1. Penjelasan Aplikasi Kalkulator

Aplikasi Kalkulator Android ini adalah sebuah aplikasi sederhana yang memungkinkan pengguna untuk melakukan operasi matematika dasar seperti penjumlahan, pengurangan, perkalian, dan pembagian. Aplikasi ini dibangun menggunakan bahasa pemrograman Java dan Android SDK.

Proyek terdiri dari beberapa file utama:

1. MainActivity.java: File utama yang berisi logika aplikasi
2. activity\_main.xml: Layout utama aplikasi
3. strings.xml: File resource untuk string
4. colors.xml: File resource untuk warna

### A. Penjelasan tentang MainActivity

- Struktur Kelas

```
public class MainActivity extends AppCompatActivity implements  
View.OnClickListener
```

Kelas ini merupakan aktivitas utama aplikasi yang mewarisi dari AppCompatActivity dan mengimplementasikan interface View.OnClickListener untuk menangani klik pada tombol.

- Deklarasi Variabel

```
TextView resultTv, solutionTv;  
MaterialButton buttonC, buttonBrackOpen, buttonBrackClose;  
MaterialButton buttonDivide, buttonMultiply, buttonPlus, buttonMinus,  
buttonEquals;  
MaterialButton button0, button1, button2, button3, button4, button5, button6, button7,  
button8, button9;  
MaterialButton buttonAC, buttonDot;
```

Variabel-variabel ini mendeklarasikan semua elemen UI yang akan digunakan dalam kalkulator, termasuk TextView untuk menampilkan hasil dan perhitungan, serta MaterialButton untuk semua tombol operasi dan angka.

- Metode onCreate()

```
@Override  
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
    resultTv = findViewById(R.id.result_tv);
```

```

solutionTv = findViewById(R.id.solution_tv);

// Pemanggilan assignId untuk setiap tombol
assignId(buttonC, R.id.button_c);
// ... (pemanggilan assignId untuk tombol lainnya)
}

```

Metode ini dipanggil saat aktivitas dibuat. Di sini, layout diatur dan semua elemen UI diinisialisasi dengan memanggil `findViewById()` dan metode `assignId()`.

- Metode `assignId()`

```

void assignId(MaterialButton btn, int id) {
    btn = findViewById(id);
    btn.setOnClickListener(this);
}

```

Metode ini menginisialisasi tombol dengan ID yang diberikan dan menetapkan `OnClickListener` untuk tombol tersebut.

- Metode `onClick()`

```

@Override
public void onClick(View view) {
    MaterialButton button = (MaterialButton) view;
    String buttonText = button.getText().toString();
    String dataToCalculate = solutionTv.getText().toString();

    // Logika untuk menangani klik tombol
    // ...
}

```

Metode ini menangani semua klik tombol. Ini mencakup logika untuk:

- Menghapus semua input (AC)
- Menampilkan hasil (=)
- Menghapus karakter terakhir (C)
- Menambahkan input ke ekspresi yang akan dihitung
- Mengevaluasi dan menampilkan hasil

- Metode `getResult()`

```

String getResult(String data) {

```

```

try {
    Context context = Context.enter();
    context.setOptimizationLevel(-1);
    Scriptable scriptable = context.initStandardObjects();
    String finalResult = context.evaluateString(scriptable, data, "Javascript", 1,
null).toString();
    if (finalResult.endsWith(".0")) {
        finalResult = finalResult.replace(".0", "");
    }
    return finalResult;
} catch (Exception e) {
    return "Err";
}
}

```

Metode ini menggunakan Rhino (engine JavaScript untuk Java) untuk mengevaluasi ekspresi matematika yang diinput. Ini memungkinkan kalkulator untuk menghitung ekspresi kompleks.

## B. Penjelasan tentang Activity\_main

- Struktur Utama

Layout ini menggunakan RelativeLayout sebagai root element, yang memungkinkan penempatan elemen-elemen secara relatif satu sama lain

```

<RelativeLayout
    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">

```

- Komponen Utama
  - TextView untuk Solusi

```

<TextView
    android:id="@+id/solution_tv"
    ...
/>

```

- Menampilkan ekspresi matematika yang diinput.
- Terletak di atas area hasil.

- TextView untuk Hasil

```
<TextView
    android:id="@+id/result_tv"
    ...
/>
```

- Menampilkan hasil perhitungan.
- Terletak di atas area tombol.

- Layout Tombol

```
<LinearLayout
    android:id="@+id/buttons_layout"
    ...
>
```

- Berisi semua tombol kalkulator.
- Menggunakan LinearLayout vertikal dengan beberapa LinearLayout horizontal di dalamnya.

- Tombol-tombol

Semua tombol menggunakan MaterialButton dengan style dan atribut yang konsisten:

```
<com.google.android.material.button.MaterialButton
    android:layout_width="72dp"
    android:layout_height="72dp"
    app:cornerRadius="36dp"
    style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
    android:textSize="32dp"
    android:textColor="@color/white"
    android:layout_margin="12dp"
    ...
/>
```

- Tombol diatur dalam 5 baris, masing-masing dalam LinearLayout horizontal.
- Tombol operasi ( $\div$ ,  $\times$ ,  $+$ ,  $-$ ) memiliki warna latar belakang oranye (#FF9800).
- Tombol C dan AC memiliki warna latar belakang merah (#F44336).
- Tombol angka memiliki warna latar belakang default (biru).

- Desain Responsif

- Penggunaan match\_parent untuk lebar dan tinggi memastikan layout mengisi seluruh layar.
- Margin dan padding digunakan untuk memberikan ruang antar elemen.
- Ukuran teks dan tombol yang konsisten memberikan tampilan yang seragam.

### C. Penjelasan colors.xml

```
<resources>
  <color name="purple_200">#FFBB86FC</color>
  <color name="purple_500">#FF6200EE</color>
  <color name="purple_700">#FF3700B3</color>
  <color name="teal_200">#FF03DAC5</color>
  <color name="teal_700">#FF018786</color>
  <color name="black">#FF000000</color>
  <color name="white">#FFFFFFFF</color>
</resources>
```

File colors.xml ini adalah bagian dari resources Android yang digunakan untuk mendefinisikan warna-warna yang akan digunakan dalam aplikasi. Berikut penjelasannya:

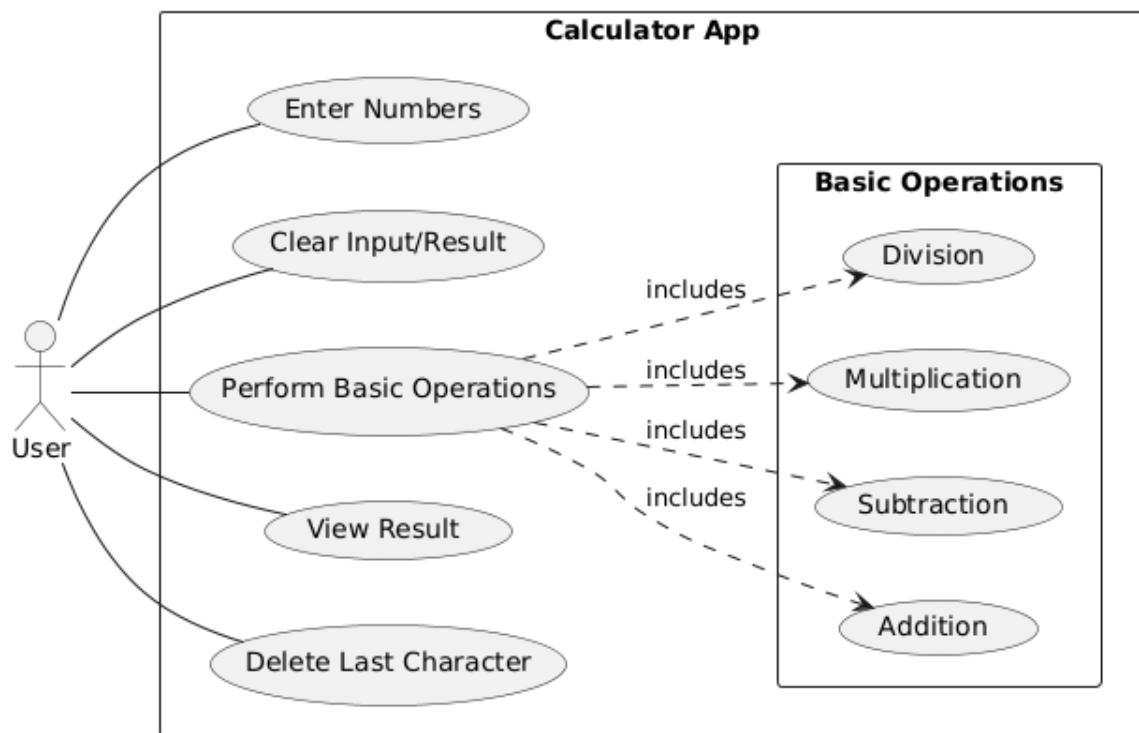
1. `<resources>`: Tag ini menandakan bahwa file ini berisi resource-resource Android.
2. Setiap warna didefinisikan dengan format:  
`<color name="nama_warna">#KODE_WARNA</color>`
3. Kode warna menggunakan format ARGB (Alpha, Red, Green, Blue) hexadecimal, di mana:
  - a. FF pertama mewakili nilai alpha (opasitas)
  - b. Enam digit berikutnya mewakili nilai RGB
4. Warna-warna yang didefinisikan:
  - a. purple\_200: Ungu muda (#FFBB86FC)
  - b. purple\_500: Ungu medium (#FF6200EE)
  - c. purple\_700: Ungu tua (#FF3700B3)
  - d. teal\_200: Teal muda (#FF03DAC5)
  - e. teal\_700: Teal tua (#FF018786)
  - f. black: Hitam (#FF000000)
  - g. white: Putih (#FFFFFFFF)
5. Warna-warna ini merupakan bagian dari tema default Material Design untuk Android, yang mencakup warna primer, warna aksen, dan warna netral.

### D. Cara Kerja Aplikasi

1. Pengguna menekan tombol angka untuk memasukkan angka.
2. Pengguna memilih operator matematika.
3. Pengguna memasukkan angka kedua.
4. Saat tombol "=" ditekan, aplikasi melakukan perhitungan.
5. Hasil ditampilkan di TextView.
6. Tombol "C" digunakan untuk mereset kalkulator.

## 2. UML Aplikasi Kalkulator

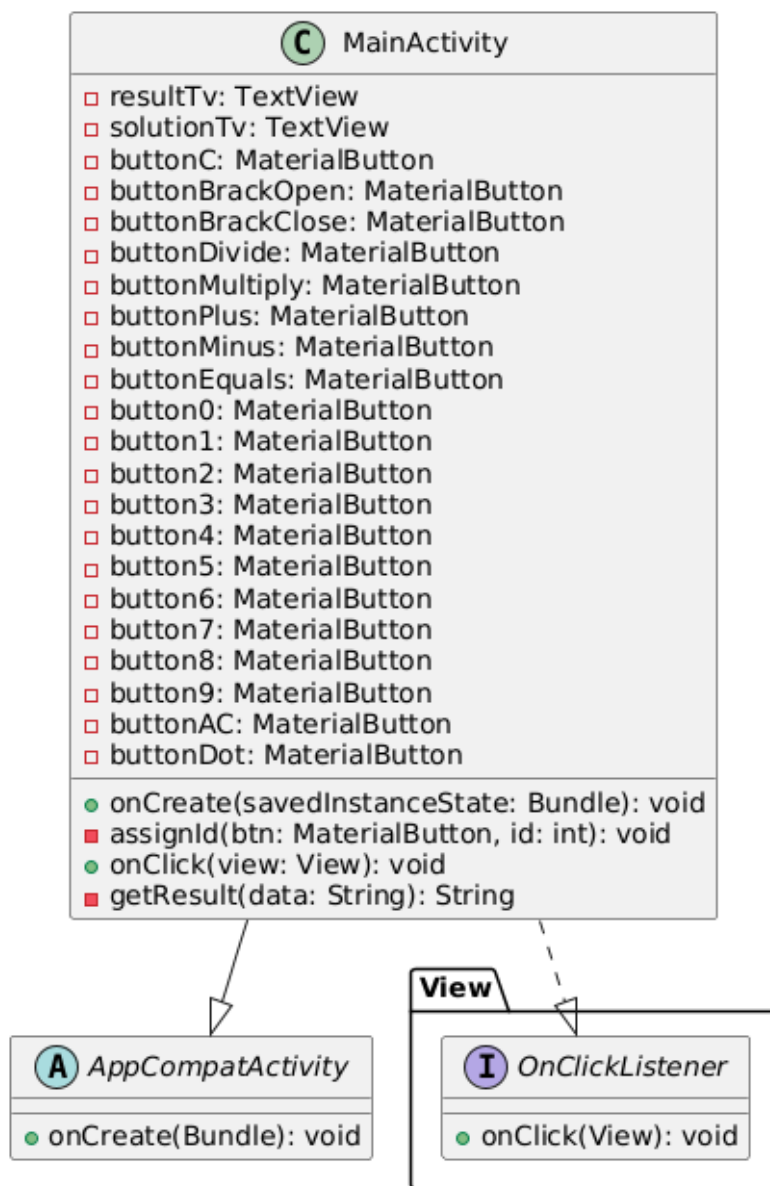
### A. Use Case Diagram



#### Penjelasan Use Case Diagram:

1. Enter Numbers: User dapat memasukkan angka (0-9) dan titik desimal
2. Clear Input/Result: User dapat menghapus semua input (AC) atau menghapus karakter terakhir (C)
3. Perform Basic Operations: User dapat melakukan operasi dasar matematika
  - a. Addition (+)
  - b. Subtraction (-)
  - c. Multiplication (×)
  - d. Division (÷)
4. View Result: User dapat melihat hasil perhitungan
5. Delete Last Character: User dapat menghapus karakter terakhir yang dimasukkan

## B. Class Diagram



Penjelasan Class Diagram:

1. **MainActivity:**
  - a. Merupakan kelas utama yang mewarisi `AppCompatActivity`
  - b. Mengimplementasi interface `View.OnClickListener` untuk menangani event klik button
  - c. Memiliki atribut-atribut berupa:
    - i. `TextView` untuk menampilkan hasil dan input
    - ii. `MaterialButton` untuk semua tombol kalkulator
  - d. Memiliki method-method:
    - i. `onCreate()`: Inisialisasi UI dan komponen

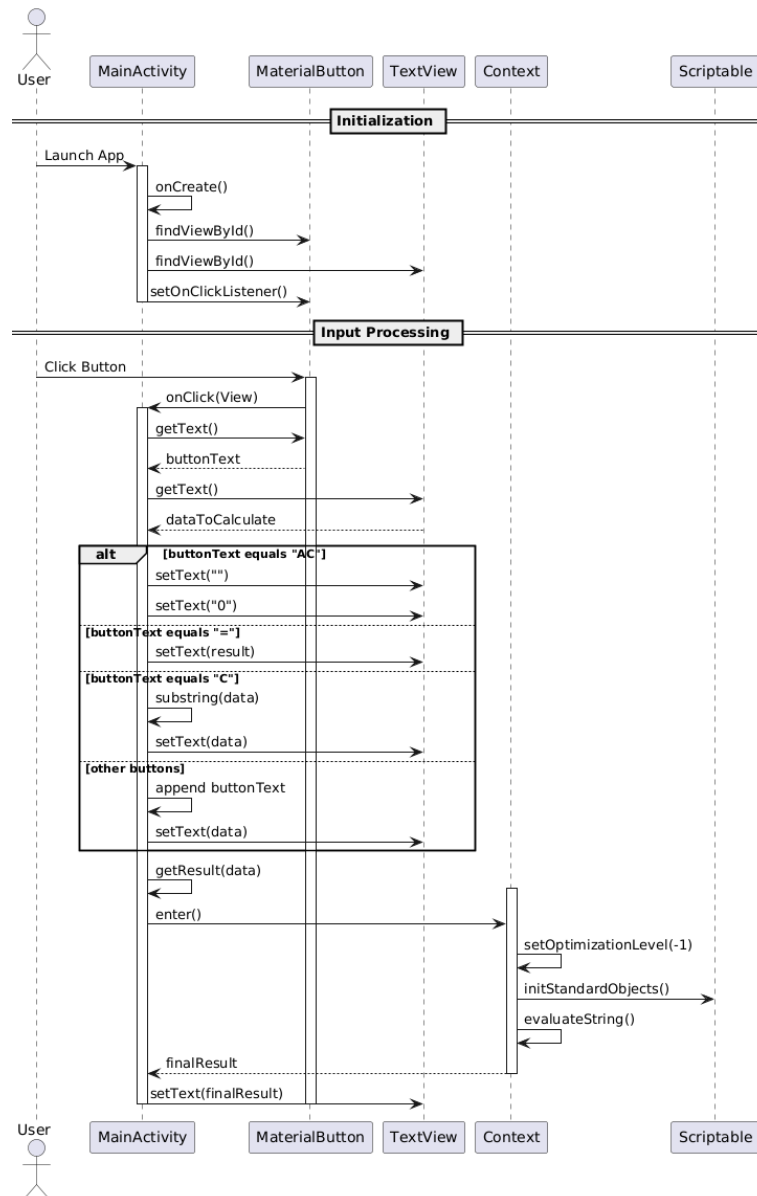


- ii. `assignId()`: Method helper untuk assign `onClick` listener ke button
  - iii. `onClick()`: Menangani event klik button
  - iv. `getResult()`: Menghitung hasil operasi matematika
- 2. `AppCompatActivity`:
  - a. Kelas abstract dari Android Framework
  - b. `MainActivity` mewarisi kelas ini untuk fungsionalitas Activity dasar
- 3. `View.OnClickListener`:
  - a. Interface dari Android Framework
  - b. `MainActivity` mengimplementasi interface ini untuk menangani event klik

Relasi antar kelas:

- `MainActivity` extends `AppCompatActivity` (inheritance)
- `MainActivity` implements `View.OnClickListener` (implementation)

## C Sequence Diagram



### Penjelasan Sequence Diagram:

1. Inisialisasi Aplikasi:
  - a. User membuka aplikasi
  - b. MainActivity melakukan onCreate()
  - c. Inisialisasi semua komponen UI menggunakan findViewById()
  - d. Set onClick listener untuk semua button
2. Proses Input dan Kalkulasi:
  - a. User menekan button (angka/operator)
  - b. MaterialButton memicu event onClick

- c. MainActivity mengambil teks dari button yang ditekan
- d. MainActivity mengambil data kalkulasi yang sudah ada
- e. Terdapat 4 kondisi berbeda berdasarkan input:
  - i. AC: Clear semua input dan hasil
  - ii. =: Menampilkan hasil kalkulasi
  - iii. C: Menghapus karakter terakhir
  - iv. Input lain: Menambahkan ke expression yang akan dihitung
- 3. Proses Kalkulasi:
  - a. MainActivity memanggil getResult()
  - b. Menggunakan Context untuk evaluasi expression
  - c. Hasil kalkulasi ditampilkan di TextView
- 4. Flow Alternatif:
  - a. Diagram menunjukkan berbagai alur berdasarkan jenis button yang ditekan
  - b. Setiap alur memiliki proses dan hasil yang berbeda
- 5. Interaksi antar Objek:
  - a. Menunjukkan komunikasi antara User, MainActivity, MaterialButton, TextView, Context, dan Scriptable
  - b. Menggambarkan alur data dan kontrol dalam aplikasi