

**LAPORAN PRAKTIKUM  
PRAKTIKUM Ke-7  
PENGENALAN ANDROID STUDIO**



**Disusun oleh:**  
**OKTAVIANA SADAMA NUR AZIZAH**  
**24060121130060**

**DEPARTEMEN ILMU KOMPUTER/INFORMATIKA  
FAKULTAS SAINS DAN MATEMATIKA  
UNIVERSITAS DIPONEGORO  
2023**

## **KATA PENGANTAR**

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas limpahan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan penyusunan laporan praktikum yang berjudul “Pengenalan Android Studio”. Laporan ini disusun untuk menyelesaikan salah satu tugas praktikum mata kuliah Pemrograman Berbasis Platform di Lab A1 program studi Informatika Fakultas Sains dan Matematika.

Adapun maksud dan tujuan dibuatnya laporan praktikum ini adalah dapat mengoperasikan Android Studio dan membuat sebuah aplikasi penghitung volume balok. Dalam penyusunan laporan ini, penulis mengucapkan terima kasih kepada Kak Arjuna Wahyu Kusuma dan Kak Danang Wisnu Prayoga yang telah membantu penulis dalam memahami materi dan dasar-dasar praktikum.

Dalam penulisan laporan praktikum ini, penulis menyadari bahwa masih banyak kekurangan-kekurangan baik pada teknis penulisan maupun materi, mengingat akan kemampuan yang dimiliki penulis masih terbatas. Untuk itu penulis mengharapkan kritik dan saran yang membangun dari pembaca, agar laporan praktikum ini dapat menjadi lebih baik. Penulis berharap laporan ini dapat menjadi sumber informasi yang bermanfaat bagi para pembaca.

Semarang, 5 November 2023

Oktaviana Sadama Nur Azizah

# **BAB I**

## **PENDAHULUAN**

### **1.1 Rumusan Masalah**

Tambahkan fitur untuk menghitung luas permukaan dan keliling balok pada *project* aplikasi yang sudah dikerjakan pada latihan sebelumnya. Tampilkan hasil perhitungan pada `tv_result`.

### **1.2 Tujuan**

Setelah mengikuti praktikum ini mahasiswa diharapkan dapat memahami cara untuk:

- 1.2.1 Mengoperasikan Android Studio.
- 1.2.2 Membuat sebuah aplikasi penghitung volume balok.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Android**

Menurut Andi Juansyah (2015), android adalah sebuah sistem operasi perangkat *mobile* berbasis linux yang mencangkup sistem operasi, middleware, dan aplikasi. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Sedangkan menurut Ajith Abraham, dkk, android adalah sistem operasi yang sebelumnya bisa digunakan pada *mobile devices*, *notebook*, dan komputer.

#### **2.2 Android Studio**

Android Studio merupakan sebuah *Integrated Development Environment* (IDE) resmi untuk pengembangan aplikasi pada sistem operasi Android milik Google. Android Studio dibangun di atas IntelliJ IDEA milik JetBrains yang didesain khusus untuk pengembangan aplikasi berbasis Android. Android Studio saat ini tersedia pada sistem operasi Windows, macOS, dan Linux. Android Studio merupakan IDE utama dalam pengembangan aplikasi Android native menggantikan *Eclipse Android Development Tools* (E-ADT) (Modul praktikum 7, hal 2).

## **BAB III**

### **PEMBAHASAN**

#### **3.1 Soal**

Tambahkan fitur untuk menghitung luas permukaan dan keliling balok pada *project* aplikasi yang sudah dikerjakan pada latihan sebelumnya. Tampilkan hasil perhitungan pada `tv_result`.

#### **3.2 Source Code**

##### **3.2.1 File: activity\_volume.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="match_parent"
        android:layout_height="match_parent">
    <LinearLayout
        android:orientation="vertical"
        android:padding="24dp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/width"
            android:labelFor="@+id/et_width"/>
        <EditText
            android:id="@+id/et_width"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginBottom="16dp"
            android:inputType="numberDecimal"
            android:autofillHints="width" />
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/length"
            android:labelFor="@+id/et_length"/>
        <EditText
            android:id="@+id/et_length"
            android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"
        android:inputType="numberDecimal"
        android:autofillHints="length" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/height"
        android:labelFor="@+id/et_height"/>
    <EditText
        android:id="@+id/et_height"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"
        android:inputType="numberDecimal"
        android:autofillHints="height" />
    <Button
        android:id="@+id/btn_volume"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/count_volume" />
    <Button
        android:id="@+id/btn_luas"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/count_luas" />
    <Button
        android:id="@+id/btn_keliling"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/count_keliling" />
    <TextView
        android:id="@+id/tv_result"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="18dp"
        android:text="@string/result"
        android:textAlignment="center"
        android:textSize="18sp"
        android:textStyle="bold" />
</LinearLayout>
</ScrollView>
```

### 3.2.1 File: VolumeActivity.kt

```
package com.example.myapplication

import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.TextView
import androidx.activity.ComponentActivity
import androidx.activity.compose.setContent
import android.view.View
import com.example.myapplication.ui.theme.MyApplicationTheme

class VolumeActivity : ComponentActivity(), View.OnClickListener {
    private lateinit var tvResult: TextView
    private lateinit var etWidth: EditText
    private lateinit var etLength: EditText
    private lateinit var etHeight: EditText
    private lateinit var btnVolume: Button
    private lateinit var btnLuas: Button
    private lateinit var btnKeliling: Button

    private val KEY_RESULT = "key_result"

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContent {
            MyApplicationTheme {
                setContentView(R.layout.activity_volume)

                tvResult = findViewById<TextView>(R.id.tv_result)
                etWidth = findViewById<EditText>(R.id.et_width)
                etLength = findViewById<EditText>(R.id.et_length)
                etHeight = findViewById<EditText>(R.id.et_height)
                btnVolume = findViewById<Button>(R.id.btn_volume)
                btnLuas = findViewById<Button>(R.id.btn_luas)
                btnKeliling = findViewById<Button>
                    (R.id.btn_keliling)

                btnVolume.setOnClickListener(this)
                btnLuas.setOnClickListener(this)
                btnKeliling.setOnClickListener(this)
            }
        }
    }
}
```

```

        if (savedInstanceState != null) {
            val result = savedInstanceState.getString
                (KEY_RESULT)
            tvResult.text = result
        }
    }
}

override fun onClick(view: View?) {
    if (view != null) {
        // Perintah hanya akan dieksekusi jika btn_volume ditekan
        when (view.id) {
            R.id.btn_volume -> {
                // Ambil nilai yang diberikan pengguna pada
                // seluruh EditText
                val inputLength: String = etLength.text.
                    toString().trim()
                val inputWidth: String = etWidth.text.toString()
                    .trim()
                val inputHeight: String = etHeight.text.
                    toString().trim()

                var isEmptyFields = false
                if (inputLength.isEmpty()) {
                    isEmptyFields = true
                    etLength.error = "Field ini tidak boleh
                        kosong"
                }

                if (inputWidth.isEmpty()) {
                    isEmptyFields = true
                    etWidth.error = "Field ini tidak boleh
                        kosong"
                }

                if (inputHeight.isEmpty()) {
                    isEmptyFields = true
                    etHeight.error = "Field ini tidak boleh
                        kosong"
                }
            }
        }
    }
}

```

```

        if(!isEmptyFields) {
            // Hitung volume balok
            val volume: Double = inputLength.toDouble()
                * inputWidth.toDouble() * inputHeight.
                toDouble();

            // Tampilkan hasil perhitungan ke TextView
            // -> tvResult
            tvResult.text = String.format("Volume: %s",
                volume.toString());
        }
    }

R.id.btn_luas -> {
    // Ambil nilai yang diberikan pengguna pada
    // seluruh EditText
    val inputLength: String = etLength.text.
        toString().trim();
    val inputWidth: String = etWidth.text.
        toString().trim();
    val inputHeight: String = etHeight.text.
        toString().trim();

    var isEmptyFields = false
    if (inputLength.isEmpty()) {
        isEmptyFields = true
        etLength.error = "Field ini tidak boleh
            kosong"
    }

    if (inputWidth.isEmpty()) {
        isEmptyFields = true
        etWidth.error = "Field ini tidak boleh
            kosong"
    }

    if (inputHeight.isEmpty()) {
        isEmptyFields = true
        etHeight.error = "Field ini tidak boleh
            kosong"
    }

    if (!isEmptyFields) {
        // Hitung luas permukaan balok

```

```

        val luas: Double = 2 * (inputLength.
            toDouble() * inputWidth.toDouble() +
            inputLength.toDouble() * inputHeight.
            toDouble() + inputWidth.toDouble() *
            inputHeight.toDouble()))

        // Tampilkan hasil perhitungan ke TextView
        // -> tvResult
        tvResult.text = String.format("Luas
            Permukaan: %s", luas.toString());
    }
}

R.id.btn_keliling -> {
    // Ambil nilai yang diberikan pengguna pada
    // seluruh EditText
    val inputLength: String = etLength.text.
        toString().trim();
    val inputWidth: String = etWidth.text.toString()
        .trim();
    val inputHeight: String = etHeight.text.
        toString().trim();

    var isEmptyFields = false
    if (inputLength.isEmpty()) {
        isEmptyFields = true
        etLength.error = "Field ini tidak boleh
            kosong"
    }

    if (inputWidth.isEmpty()) {
        isEmptyFields = true
        etWidth.error = "Field ini tidak boleh
            kosong"
    }

    if (inputHeight.isEmpty()) {
        isEmptyFields = true
        etHeight.error = "Field ini tidak boleh
            kosong"
    }

    if (!isEmptyFields) {
        // Hitung keliling balok

```

```

        val keliling: Double = 4 * (inputLength.
            toDouble() + inputWidth.toDouble() +
            inputHeight.toDouble())

        // Tampilkan hasil perhitungan ke TextView
        // -> tvResult
        tvResult.text = String.format("Keliling:
            %s", keliling.toString());
    }
}

}

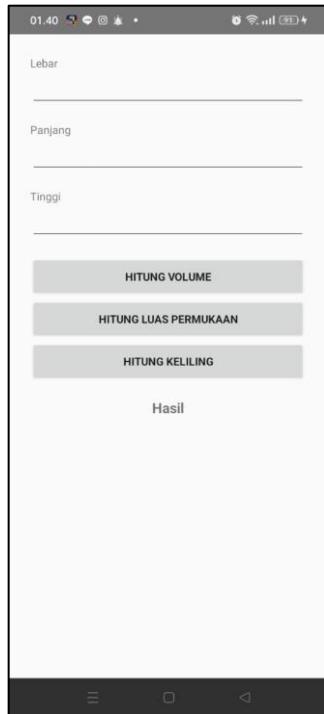
}

override fun onSaveInstanceState(outState: Bundle) {
    super.onSaveInstanceState(outState)

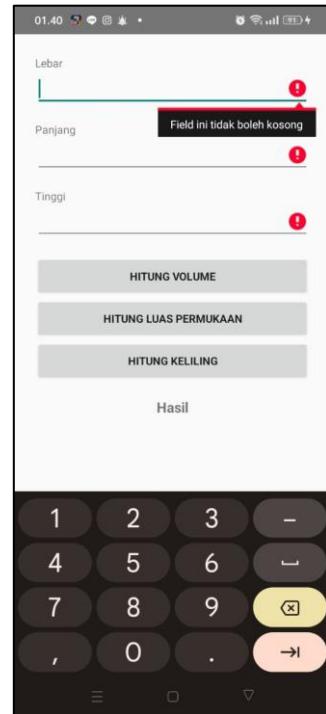
    val calculationResult = tvResult.text.toString()
    outState.putString(KEY_RESULT, calculationResult)
}
}

```

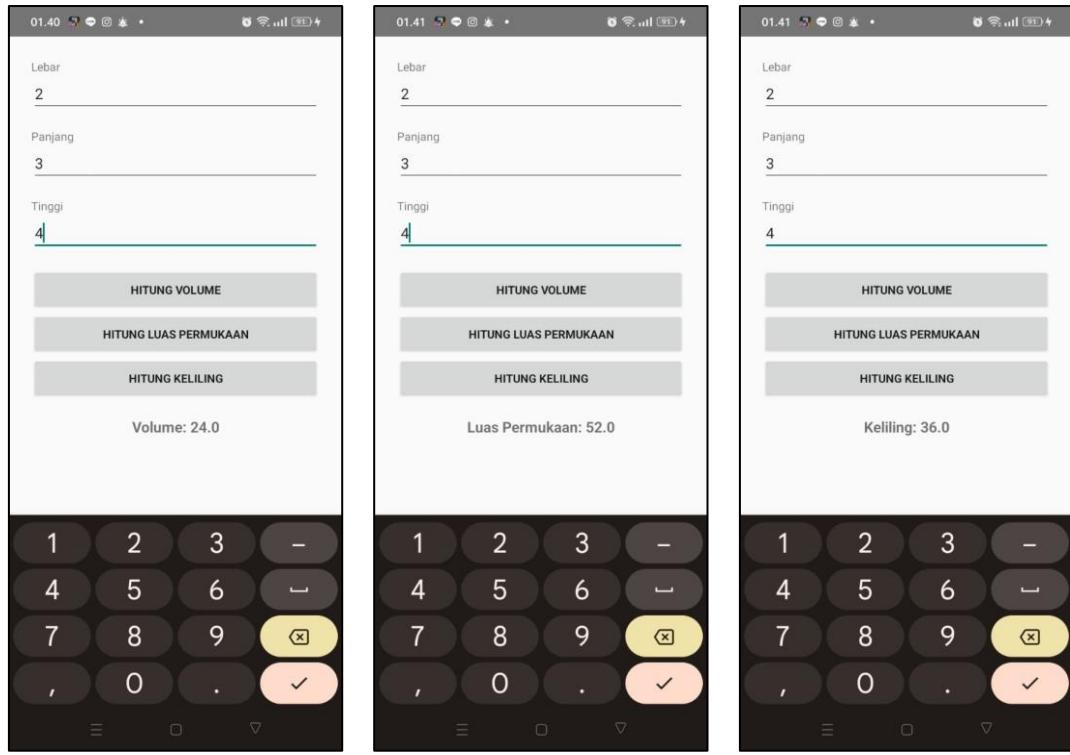
### 3.3 Screenshot Program



Tampilan Basic



Apabila tidak memenuhi validasi



Mengitung Volume

Menghitung Luas  
Permukaan

Menghitung Keliling

### 3.4 Penjelasan

Dalam penggeraan tugas pembuatan kalkulator untuk menghitung volume, luas permukaan, dan keliling balok kita bekerja menggunakan dua file, yaitu file `activity_volume.xml` dan `VolumeActivity.kt`. File `activity_volume.xml` menangani masalah UI atau tampilan dari aplikasi android yang ingin kita buat, sedangkan file `VolumeActivity.kt` berisi logika yang digunakan untuk menangani perhitungan volume, luas permukaan, dan keliling balok.

#### 3.4.1 File: `activity_volume.xml`

```
<?xml version="1.0" encoding="utf-8"?> merupakan deklarasi XML  
yang menunjukkan versi XML dan encoding yang digunakan dalam file  
XML. <ScrollView xmlns:android="http://schemas.android.com/  
apk/res/android ...> merupakan elemen utama yang mewadahi semua  
UI di dalamnya.
```

Tag `ScrollView` digunakan untuk membuat tampilan yang dapat di-*scroll* atau digulir jika kontennya lebih besar dari layar perangkat. Atribut `xmlns:android` mendefinisikan namespace "android" yang digunakan dalam dokumen XML.

Tag `<LinearLayout>` merupakan wadah untuk mengatur elemen-elemen UI dalam satu arah. Disini elemen UI diatur secara vertikal menggunakan `android:orientation="vertical"`.

Tag `<TextView>` digunakan untuk menampilkan teks label. Tag ini menunjukkan label "Width", "Length", dan "Height". Atribut `android:text` digunakan untuk mengatur teks yang akan ditampilkan dan atribut `android:labelFor` digunakan untuk menentukan elemen yang akan dihubungkan oleh aksesibilitas.

Dalam membuat *layout* "Width" dan "Height" digunakan nilai `match_parent` dimana elemen tersebut akan mengisi seluruh ruang yang tersedia dalam wadah induknya dalam arah yang sesuai. Terdapat pula nilai `wrap_content` dimana elemen tersebut akan secara otomatis menyesuaikan ukurannya untuk mencakup konten dengan presisi yang diperlukan.

Tag `<EditText>` digunakan untuk memasukkan input teks. Masing-masing memiliki ID yang unik (`android:id`) dan atribut `android:autofillHints` digunakan untuk memberi petunjuk *autofill* yang berkaitan dengan masing-masing elemen. Adapun tipe data yang diterima yaitu `numberDecimal`.

Tag `<Button>` digunakan untuk membuat tombol aksi dimana masing-masing tombol memiliki ID yang unik dan teks yang akan ditampilkan. Hasil perhitungan akan ditampilkan di `TextView` dengan ID `tv_result`.

### 3.4.2 File: VolumeActivity.kt

```
package com.example.myapplication
```

Baris tersebut mendefinisikan paket (*package*) tempat kelas berada, yaitu berada di `com.example.myapplication`.

```
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.TextView
import androidx.activity.ComponentActivity
import androidx.activity.compose.setContent
import android.view.View
import com.example.myapplication.ui.theme.MyApplicationTheme
```

Disini kita melakukan impor untuk berbagai kelas dan komponen yang diperlukan untuk mengembangkan aplikasi Android. Beberapa kelas yang di impor antara lain, `Bundle` untuk mengelola data dalam siklus hidup aktivitas, `Button`, `EditText`, dan `TextView` untuk elemen-elemen antarmuka pengguna, dan beberapa komponen dari pustaka Android seperti `ComponentActivity`.

```
class VolumeActivity : ComponentActivity(), View.OnClickListener {
    ...
}
```

Baris kode diatas merupakan deklarasi kelas `VolumeActivity`. Kelas ini adalah turunan (subkelas) dari `ComponentActivity` dengan mengimplementasikan antarmuka `View.OnClickListener`. Hal ini berarti `VolumeActivity` adalah sebuah aktivitas dalam aplikasi Android dan akan menangani tindakan pengguna, seperti mengklik tombol, karena mengimplementasikan `View.OnClickListener`. Di dalam kelas ini kita mendeklarasikan variabel-variabel yang merujuk ke file `activity_volume.xml` yang sudah kita definisikan sebelumnya.

```
private val KEY_RESULT = "key_result"
```

Dalam kelas `VolumeActivity` dideklarasikan pula baris kode diatas untuk menentukan kunci yang akan digunakan saat menyimpan dan mengambil data dari `Bundle`. Variabel ini digunakan dalam siklus hidup aktivitas untuk mengatur ulang tampilan jika aplikasi ditutup dan dibuka kembali.

```
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView {
        MyApplicationTheme {
            setContentView(R.layout.activity_volume)
            ...
            if (savedInstanceState != null) { val result =
                savedInstanceState.getString(KEY_RESULT) tvResult.text = result
            }
        }
    }
}
```

Baris kode diatas merupakan bagian dari metode `onCreate` dalam kelas `VolumeActivity` yang merupakan bagian dari siklus hidup aktivitas Android. Metode ini dipanggil saat aktivitas pertama kali dibuat. Dalam kode ini, metode `onCreate` di `override` dari kelas induk (`ComponentActivity`) untuk mengatur perilaku khusus aktivitas ini.

Dalamnya, didefinisikan tampilan aktivitas menggunakan `setContent`, yang merupakan fungsi ekstensi yang digunakan untuk menentukan tampilan aktivitas berdasarkan tema yang ditentukan (`MyApplicationTheme`) dan layout XML yang didefinisikan dalam file `activity_volume.xml`. Selain itu, dalam blok ini, elemen-elemen antarmuka pengguna diinisialisasi dengan mengambil referensi dari tampilan menggunakan `findViewById`.

Kemudian *conditional if* digunakan untuk memeriksa apakah ada data yang disimpan dalam `savedInstanceState`. `savedInstanceState` adalah objek `Bundle` yang digunakan untuk menyimpan data ketika aktivitas dihancurkan dan kemudian dibangun kembali. Jika ada data yang tersimpan dengan kunci yang sesuai (`KEY_RESULT`), data tersebut akan diambil dan ditampilkan di elemen `TextView` dengan nama `tvResult`.

```
override fun onClick(view: View?) {
    ...
}
```

Baris kode diatas merupakan metode yang mengimplementasikan antarmuka `View.OnClickListener`. Metode ini akan menangani tindakan pengguna

saat tombol-tombol yang sesuai ditekan. Metode ini memeriksa ID dari elemen yang diklik dan melakukan perhitungan berdasarkan tombol yang diklik (tombol hitung volume, luas permukaan, atau keliling).

Dalam metode ini, *input* yang diberikan pengguna ditangkap setelah pengguna menekan tombol perhitungan dan logika perhitungan dijalankan untuk menghasilkan *output* yang sesuai. Dalam metode ini pula dijalankan fungsi validasi seperti akan mengeluarkan *alert* apabila pengguna menekan tombol perhitungan tanpa mengisikan *input* apapun atau terdapat *field* yang belum diisi *input*.

```
override fun onSaveInstanceState(outState: Bundle) {  
    super.onSaveInstanceState(outState)  
  
    val calculationResult = tvResult.text.toString()  
    outState.putString(KEY_RESULT, calculationResult)  
}
```

Baris kode diatas merupakan metode yang digunakan untuk menyimpan keadaan aktivitas sebelum aktifitas dihancurkan atau berhenti. Kemudian variabel `calculationResult` akan mengambil teks yang ditampilkan pada elemen `tvResult` dan menyimpannya. Variabel akan berisi hasil perhitungan yang ditampilkan di `TextView`.

Baris `outState.putString(KEY_RESULT, calculationResult)` akan menyimpan nilai `calculationResult` ke dalam objek `Bundle` yang diberi nama `outState`. `KEY_RESULT` yang sudah didefinisikan diawal digunakan untuk menyimpan hasil perhitungan.

## **BAB IV**

## **PENUTUP**

### **4.1 Kesimpulan**

Berdasarkan praktikum yang telah dilakukan, penulis dapat mengoperasikan Android Studio dan membuat program dengan bahasa kotlin. Adapun program yang berhasil penulis buat dan jalankan antara lain program untuk menghitung volume, luas permukaan, dan keliling balok. Dalam penggerjaannya, penulis menggunakan *device* android pribadi yang dihubungkan dengan Android Studio sebagai emulator.

### **4.2 Saran**

Terimakasih saya ucapan kepada asisten praktikum Lab A1, terkhusus Kak Danang Wisnu Prayoga yang selalu siap dan bersedia mendampingi penulis dalam proses pembelajaran Android Studio. Terimakasih atas bantuannya mulai dari penanganan *error* hingga membantu penulis menyambungkan *device* android sebagai emulator 😊 🙏.

## **DAFTAR PUSTAKA**

- Abraham, Ajith. Mauri, Jamie Lloret. Buford, John. 2011. *Advances in Computing and Communications Part III*. India: Springer Science & Business Media.
- Andi, J. 2015. *Pengembangan Aplikasi Child Tracker Berbasis Assisted – Global Positioning System (A-GPS) Dengan Platform Android*. Jurnal Ilmiah Komputer dan Informatika (KOMPUTA), Edisi 1 Volume 1, ISSN: 2089-9033.
- Praktikum PBP. 2023. *Introduction to Android Studio & Activity*. Prodi Informatika, Fakultas Sains dan Matematika, Universitas Diponegoro.