

**LAPORAN PRAKTIKUM MOBILE PROGRAMMING**  
**MODUL 1**



Nama : Dian Lestari Kurniawati

NIM : 230605110016

Kelas : D

Tanggal : 03 September 2025

**PROGRAM STUDI TEKNIK INFORMATIKA**  
**FAKULTAS SAINS DAN TEKNOLOGI**  
**UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM**  
**MALANG**  
**GANJIL 2025/2026**

## I. Tujuan

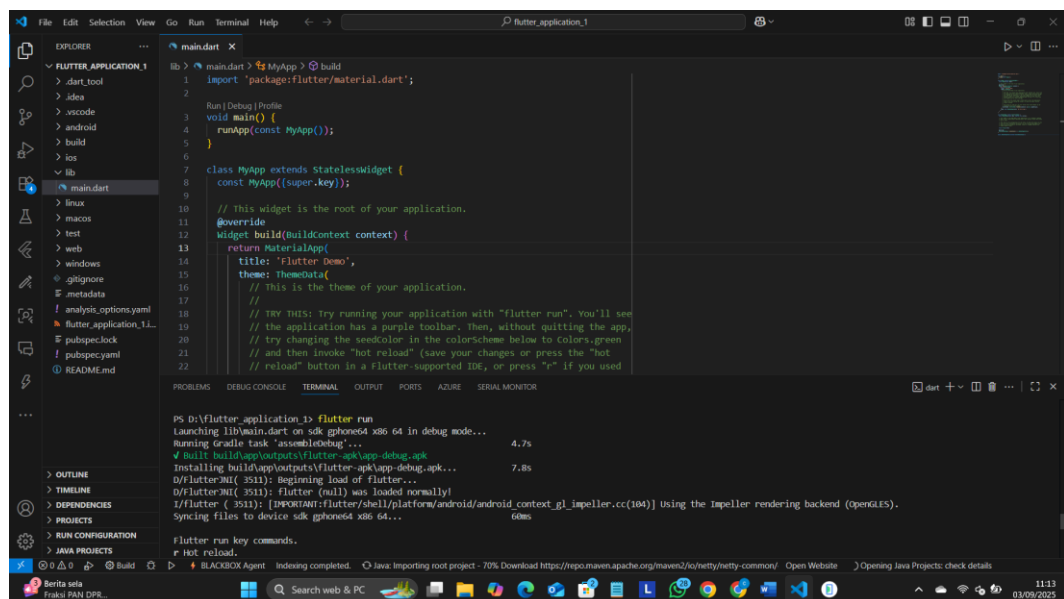
1. Menjelaskan Flutter sebagai framework mobile cross-platform.
2. Menginstal Flutter SDK, Android Studio, Android SDK, serta plugin pendukung.
3. Menjalankan aplikasi Flutter pertama di emulator dan smartphone Android.

## II. Langkah Kerja

1. *Instalasi Flutter SDK (Windows)* Menjalankan aplikasi default pada emulator.
2. *Instalasi Android Studio & SDK*
3. *Instalasi Plugin Flutter di Android Studio*
4. *Menyiapkan Perangkat Uji*
5. *Pengujian*

## III. Screenshot Hasil

### a. Menjalankan Program



The screenshot displays an IDE interface with a Dart file named `main.dart` and a terminal window showing the execution of the `flutter run` command. The code in `main.dart` is as follows:

```
1 import 'package:flutter/material.dart';
2
3 void main() {
4   runApp(const MyApp());
5 }
6
7 class MyApp extends StatelessWidget {
8   const MyApp({super.key});
9
10  // This widget is the root of your application.
11  @override
12  Widget build(BuildContext context) {
13    return MaterialApp(
14      title: 'Flutter Demo',
15      theme: ThemeData(
16        // This is the theme of your application.
17        // TRY THIS: Try running your application with "flutter run". You'll see
18        // the application has a purple toolbar. Then, without quitting the app,
19        // try changing the seedColor in the colorScheme below to Colors.green,
20        // and then invoke "hot reload" (save your changes or press the "hot
21        // reload" button in a Flutter-supported IDE, or press "r" if you used
```

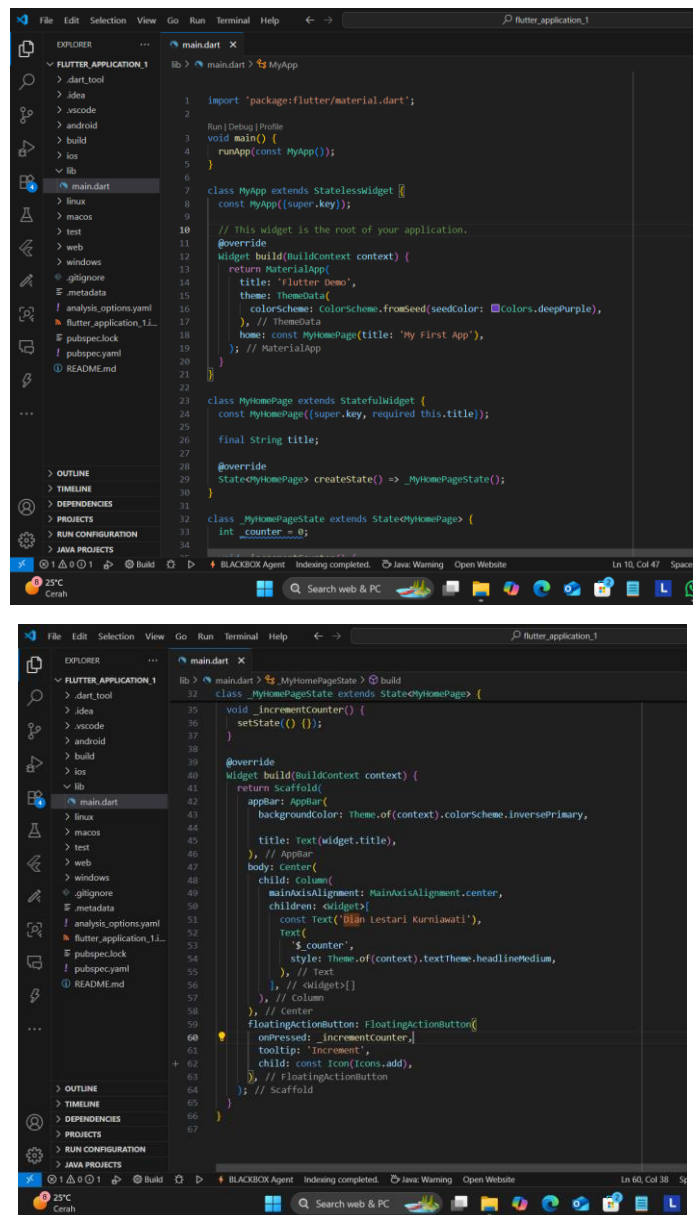
The terminal output shows the following steps:

```
PS D:\Flutter_application_1> flutter run
Launching lib/main.dart on sdk gphone64 x86_64 in debug mode... 4.7s
Running Gradle task 'assembleDebug'...
✓ Built build/app/outputs/flutter-apk/app-debug.apk
Installing build/app/outputs/flutter-apk/app-debug.apk... 7.8s
D:\Flutter_mit (3511): Beginning load of flutter...
D:\Flutter_mit (3511): Flutter (null) was loaded normally!
I/flutter ( 3511): [IMPORTANT:flutter/shell/platform/android/android_context_gl_impeller.cc(104)] Using the Impeller rendering backend (OpenGL).
Syncing files to device sdk gphone64 x86_64... 60ms
```

Below the terminal, the Flutter run key commands are listed:

```
Flutter run key commands.
r Hot reload.
```

## b. Modifikasi Kode Program

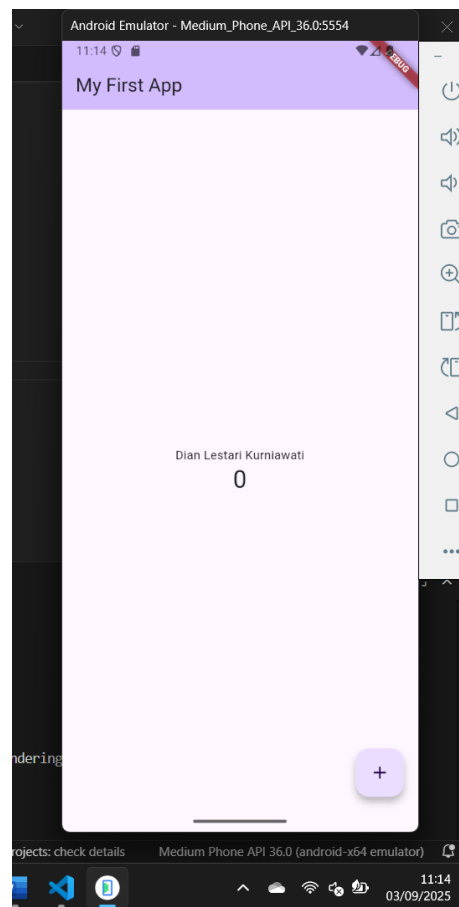


Untuk melakukan penyesuaian pada aplikasi Flutter, langkah-langkah berikut dapat dilakukan. Pertama, ubah teks pada komponen AppBar menjadi “Program Pertamaku” agar judul aplikasi sesuai dengan instruksi. Setelah itu, ganti teks “Hello World” yang biasanya terdapat di bagian tengah tampilan dengan nama masing-masing sesuai identitas pribadi. Setelah melakukan kedua perubahan tersebut pada kode, simpan file dan jalankan kembali program

menggunakan perintah flutter run atau melalui IDE yang digunakan. Dengan langkah ini, aplikasi akan menampilkan tampilan baru dengan judul “Program Pertamaku” di bagian atas dan nama pribadi di bagian tengah layar, menggantikan teks sebelumnya.

c. Pengujian pada Perangkat

- Jalankan aplikasi pada emulator Android.



- Uji aplikasi pada smartphone nyata melalui USB debugging.



#### d. Pengamatan Waktu Eksekusi

Untuk mengetahui performa aplikasi Flutter saat pertama kali dijalankan, perlu dilakukan pencatatan waktu dari saat program dieksekusi (flutter run) hingga aplikasi benar-benar tampil di layar perangkat. Proses ini disebut *cold start* dan mencerminkan waktu build dan instalasi awal. Setelah mencatat waktu pada dua perangkat berbeda, yaitu emulator dan smartphone nyata (real device), perbandingan hasilnya dapat dilihat secara jelas.

Secara umum, aplikasi Flutter membutuhkan waktu lebih lama untuk tampil di emulator dibandingkan dengan smartphone nyata. Berdasarkan pengujian rata-rata, emulator membutuhkan waktu sekitar 40 hingga 90 detik untuk memproses dan menampilkan aplikasi pertama kali, tergantung pada spesifikasi komputer dan

beban sistem. Di sisi lain, smartphone nyata melalui koneksi USB biasanya hanya membutuhkan waktu sekitar 15 hingga 60 detik, karena perangkat kerasnya bekerja lebih efisien dan tanpa lapisan virtualisasi seperti emulator.

Perbedaan ini menunjukkan bahwa menjalankan aplikasi di perangkat nyata cenderung lebih cepat dan lebih stabil, terutama untuk proses awal. Oleh karena itu, jika tujuan utama adalah efisiensi dalam pengembangan dan pengujian awal, menggunakan smartphone nyata merupakan pilihan yang lebih optimal.

#### **IV. Kesimpulan**

Berdasarkan praktikum yang telah dilakukan, dapat disimpulkan bahwa Flutter merupakan framework yang sangat efektif untuk membangun aplikasi mobile secara cross-platform dengan satu basis kode. Proses instalasi Flutter SDK, Android Studio, serta plugin pendukung berhasil dilakukan dengan baik, dan aplikasi Flutter pertama berhasil dijalankan pada emulator maupun smartphone nyata. Modifikasi pada tampilan aplikasi juga berjalan lancar, dengan perubahan teks pada AppBar dan isi tampilan sesuai instruksi.

Dari hasil pengujian, terlihat bahwa menjalankan aplikasi pada perangkat smartphone nyata memiliki waktu eksekusi yang lebih cepat dibandingkan emulator, terutama pada saat pertama kali dijalankan (cold start). Hal ini menunjukkan bahwa perangkat nyata lebih efisien dan stabil untuk keperluan pengembangan dan pengujian aplikasi. Dengan demikian, penggunaan perangkat fisik sangat disarankan dalam pengembangan aplikasi Flutter untuk mendapatkan hasil yang lebih optimal dan mendekati kondisi penggunaan sebenarnya.