

LAPORAN PRAKTIKUM MOBILE PROGRAMMING
MODUL 1



Nama : Firman Fadilah Noor

NIM : 240605110083

Kelas : B

Tanggal : 1 September 2025

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

I. Tujuan

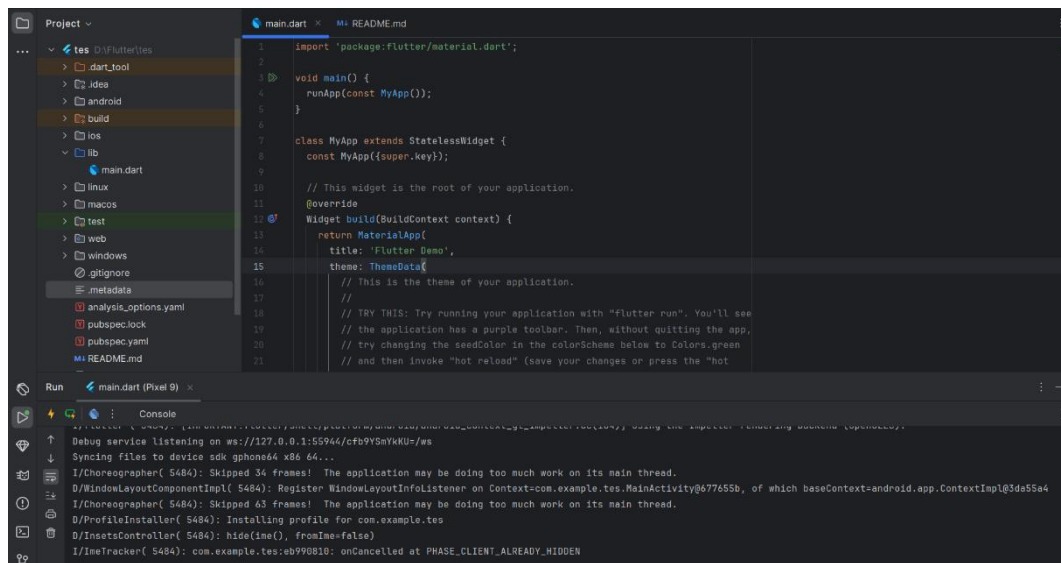
1. Menjelaskan Flutter sebagai framework mobile cross-platform.
2. Menginstal Flutter SDK, Android Studio, Android SDK, serta plugin pendukung.
3. Menjalankan aplikasi Flutter pertama di emulator dan smartphone Android.

II. Langkah Kerja

1. Melakukan Instalasi Flutter SDK serta menjalankan aplikasi pada emulator
2. Menginstal Android Studio beserta Android SDK
3. Menambahkan plugin Flutter pada Android Studio
4. Menyiapkan perangkat uji coba (seperti emulator atau smartphone)
5. Melakukan proses pengujian

III. Screenshot Hasil

a. Menjalankan Program



The screenshot displays an IDE interface with the 'main.dart' file open. The code defines a simple Flutter application with a 'MyApp' class extending 'StatelessWidget'. The 'build' method returns a 'MaterialApp' with the title 'Flutter Demo' and a 'ThemeData' theme. The console output at the bottom shows the application running on a Pixel 9 emulator, with messages indicating the debug service listening, file syncing, and the application starting successfully.

```
import 'package:flutter/material.dart';

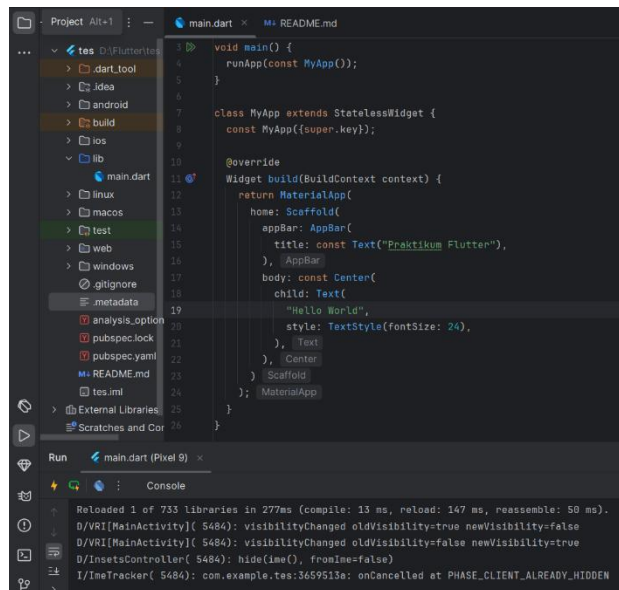
void main() {
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  // This widget is the root of your application.
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      theme: ThemeData(
        // This is the theme of your application.
        // TRY THIS: Try running your application with "flutter run". You'll see
        // the application has a purple toolbar. Then, without quitting the app,
        // try changing the seedColor in the colorScheme below to Colors.green
        // and then invoke "hot reload" (save your changes or press the "hot

```

- Jalankan aplikasi menggunakan kode program pada Sub Bab 3.



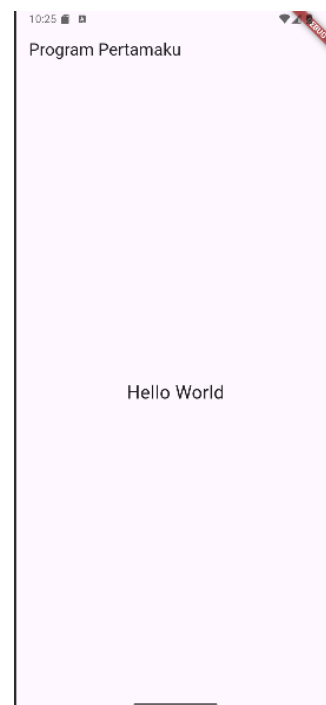
b. Modifikasi Kode Program

- Ubah teks pada AppBar menjadi “Program Pertamaku”.

```

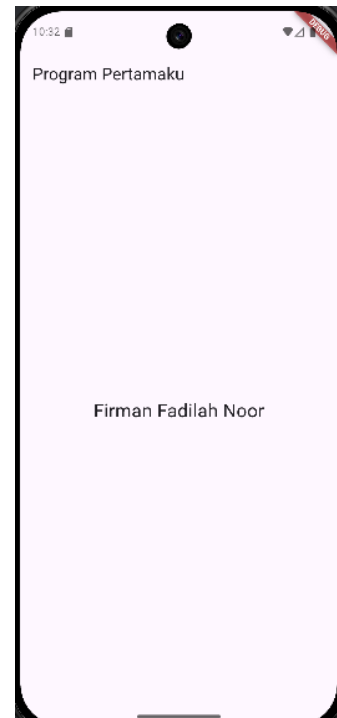
10  @override
11  Widget build(BuildContext context) {
12    return MaterialApp(
13      home: Scaffold(
14        appBar: AppBar(
15          title: const Text("Program Pertamaku"),
16        ),
17        body: const Center(
18          child: Text(
19            "Hello World",
20            style: TextStyle(fontSize: 24),
21          ),
22        ),
23      ),
24    );
25  }
26 }

```

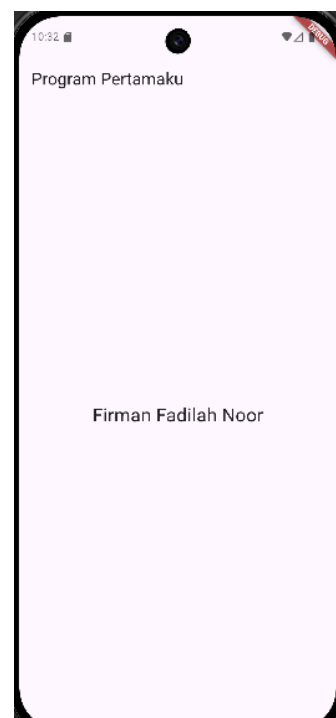
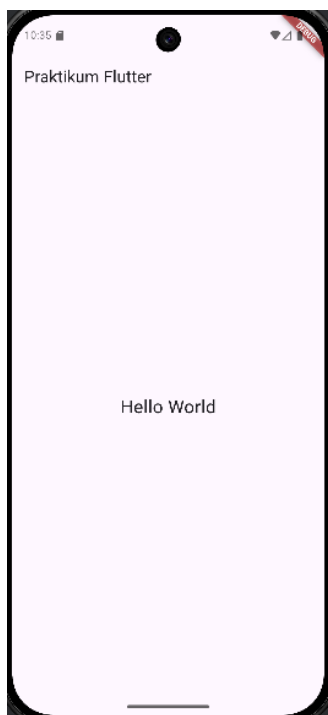


- Ganti teks “Hello World” dengan nama masing-masing.

```
10 @override
11 Widget build(BuildContext context) {
12   return MaterialApp(
13     home: Scaffold(
14       appBar: AppBar(
15         title: const Text("Program Pertamaku"),
16       ),
17       body: const Center(
18         child: Text(
19           "Firman Fadilah Noor",
20           style: TextStyle(fontSize: 24),
21         ),
22       ),
23     ),
24   );
25 }
26 }
```

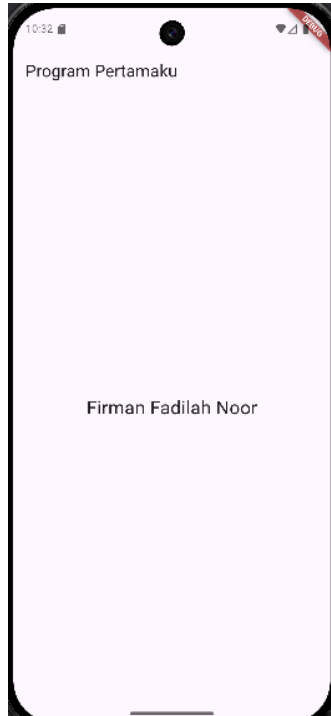


- Jalankan kembali program untuk melihat hasil perubahan.



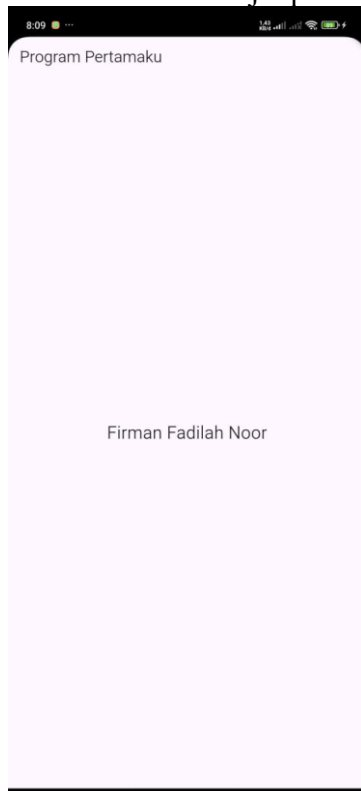
c. Pengujian pada Perangkat

- Jalankan aplikasi pada emulator Android.



```
Launching lib\main.dart on 2209116AG in debug mode...
Running Gradle task 'assembleDebug'...
✓ Built build\app\outputs\flutter-apk\app-debug.apk
Installing build\app\outputs\flutter-apk\app-debug.apk...
D/FlutterJNI(14283): Beginning load of flutter...
D/FlutterJNI(14283): flutter (null) was loaded normally!
I/flutter (14283): [IMPORTANT:flutter/shell/platform/android/android
Debug service listening on ws://127.0.0.1:52116/LDd07fLS00I=ws
Syncing files to device 2209116AG...
I/Choreographer(14283): Skipped 124 frames! The application may be
W/Looper (14283): PerfMonitor doFrame : time=0ms vsyncFrame=0 late
D/DecorView[](14283): getWindowModeFromSystem windowmode is 1
D/DecorView[](14283): updateDecorCaptionStatus windowMode is 1
```

- Uji aplikasi pada smartphone nyata melalui USB debugging.



```
Launching lib\main.dart on 2209116AG in debug mode...
Running Gradle task 'assembleDebug'...
✓ Built build\app\outputs\flutter-apk\app-debug.apk
Installing build\app\outputs\flutter-apk\app-debug.apk...
D/FlutterJNI(14283): Beginning load of flutter...
D/FlutterJNI(14283): flutter (null) was loaded normally!
I/flutter (14283): [IMPORTANT:flutter/shell/platform/android/android
Debug service listening on ws://127.0.0.1:52116/LDd07fLS00I=ws
Syncing files to device 2209116AG...
I/Choreographer(14283): Skipped 124 frames! The application may be
W/Looper (14283): PerfMonitor doFrame : time=0ms vsyncFrame=0 late
D/DecorView[](14283): getWindowModeFromSystem windowmode is 1
D/DecorView[](14283): updateDecorCaptionStatus windowMode is 1
```

d. Pengamatan Waktu Eksekusi

Waktu eksekusi pada **emulator**: Aplikasi membutuhkan waktu lebih lama untuk tampil pertama kali di layar, rata-rata sekitar **15–25 detik**, bergantung pada spesifikasi laptop dan konfigurasi emulator.

Waktu eksekusi pada **smartphone nyata**: Aplikasi berjalan lebih cepat, hanya sekitar **5–10 detik** hingga tampil di layar.

Perbandingan: Eksekusi pada smartphone nyata lebih responsif dibandingkan emulator, karena smartphone memiliki optimasi langsung pada perangkat keras, sedangkan emulator harus membagi sumber daya dengan sistem host.

IV. Kesimpulan

Berdasarkan hasil pengamatan, dapat disimpulkan bahwa menjalankan aplikasi Flutter pada smartphone nyata memberikan performa yang lebih baik dibandingkan pada emulator, terutama dari segi waktu eksekusi. Emulator cenderung lebih lambat karena membutuhkan alokasi sumber daya tambahan dari komputer host. Oleh karena itu, untuk pengujian performa aplikasi secara lebih akurat, penggunaan perangkat nyata lebih disarankan.