



BOILERPLATE PADA FLUTTER

yunia ikawati



OVERVIEW

01

BOILERPLATE

02

Fitur dan Jenis
BOILERPLATE

03

Penggunaan
BOILERPLATE

BOILERPLATE

- **Boilerplate** dalam Flutter adalah kode awal atau template dasar yang sudah memiliki struktur dan konfigurasi yang umum digunakan, sehingga memudahkan pengembang untuk memulai tanpa harus membangun semuanya dari nol.
- Kode ini menjadi **pondasi utama** sebelum kita mulai menambahkan fitur-fitur lain pada aplikasi.
- Ini biasanya mencakup pengaturan **arsitektur, manajemen state, dependensi, dan fitur dasar seperti autentikasi atau navigasi.**

Contoh Struktur Pada Flutter

```
lib/  
  └─ main.dart  
test/  
  └─ widget_test.dart  
pubspec.yaml  
android/  
ios/  
web/
```

- `lib/` → kode utama aplikasi
- `main.dart` → file utama (entry point)
- `pubspec.yaml` → konfigurasi dependencies
- `android/`, `ios/`, `web/` → platform target
- `test/` → untuk unit testing

Isi default kode pada main.dart

```
void main() {  
  runApp(const MyApp());  
}  
  
class MyApp extends StatelessWidget {  
  const MyApp({super.key});  
  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      title: 'Flutter Demo',  
      home: const MyHomePage(),  
    );  
  }  
}
```

Fitur-fitur pada Boilerplate Zubairehman

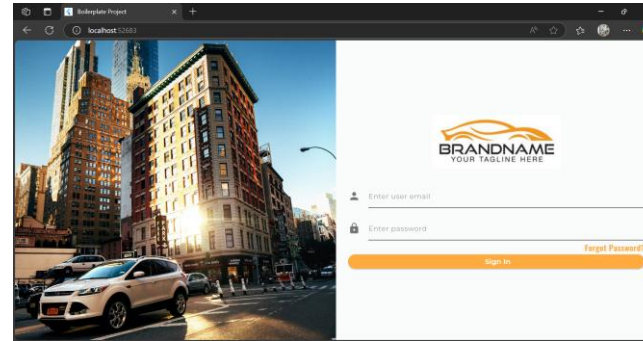
1. **Splash** ⇒ Layar pertama saat aplikasi diluncurkan.
2. **Login** ⇒ Layar login dengan tata letak UI yang sudah jadi
3. **Home** ⇒ Layar pertama untuk dinavigasi setelah login
4. **Routing** ⇒ Halaman koleksi untuk menyimpan semua layar perutean agar Navigasi lebih mudah
5. **Thema** ⇒ Tema Gelap dan Terang tersedia untuk digunakan
6. **Dio** ⇒ Paket permintaan HTTP
7. **Database** ⇒ Sembast
8. **MobX** ⇒ Manajemen Status
9. **Provider** ⇒ Manajemen Status
10. **Encryption** ⇒ enkripsi dan dekripsi untuk String/Uint8List
11. **Validation** ⇒ Validasi dan sanitasi string untuk Dart
12. **Code Generation** ⇒ Membuat kode yang diperlukan dengan build_runner
13. **User Notification** ⇒ Menampilkan pemberitahuan, kesalahan, masalah sebagai Flushbar
14. **Logging** ⇒ Flogs
15. **Dependency Injection** ⇒ Injeksi ketergantungan waktu kompilasi
16. **Dark Theme Support**
17. **Multilingual Support** ⇒ Dukungan pelokalan untuk bahasa pilihan Anda

Contoh Jenis-jenis Boilerplate

Flutter Boilerplate Project	Menggunakan MobX dan Provider untuk manajemen state.
Flutter Template	Template dengan fitur CRUD dan berbagai pola pengembangan.
Ship Flutter Template	Template sederhana untuk proyek Flutter.
Flutter Boilerplate Modular	Boilerplate dengan pendekatan modular.

Link Contoh Jenis-jenis Boilerplate

1. [zubairehman/flutter_boilerplate](#) project: A boilerplate project created in flutter using MobX and Provider.



2. <https://flutterawesome.com/a-template-for-flutter-projects-with-crud/>
3. <https://github.com/marcelpinto/ship-flutter-template?tab=readme-ov-file>
4. <https://github.com/brsHunterX/flutter-boilerplate-modular>

LATIHAN

IMPLEMENTASIKAN SALAH SATU REFERENSI YANG SUDAH
DISEMATKAN SEBELUMNYA?

Referensi

1. <https://flutterawesome.com/a-boilerplate-project-which-adopts-the-concept-of-clean-architecture-and-modularization/>
2. <https://github.com/humphreydaniel/flutter-clean-architecture-boilerplate?tab=readme-ov-file>
3. <https://codereis.com/posts/new-flutter-project-with-a-boilerplate/>
4. <https://github.com/zubairehman>