

**UNGUIDED**  
**PEMROGRAMAN PERANGKAT BERGERAK**

**MODUL XIII**  
**NETWORKING**



**Disusun Oleh :**  
**Ilham Lii Assidaq**  
**2311104068**  
**Asisten Praktikum :**  
**Yoga Eka Pratama**  
**Zulfa Mustafa Akhyar Iswahyudi**  
**Dosen Pengampu :**  
**Yudha Islami Sulistya, S.Kom., M.Cs.**

**PROGRAM STUDI S1 SOFTWARE ENGINEERING**  
**FAKULTAS INFORMATIKA**  
**TELKOM UNIVERSITY PURWOKERTO**  
**2024**

**TUGAS PENDAHULUAN**

## 1. SOAL

Buatlah Aplikasi catatan sederhana menggunakan GetX

a. Source Code

Note\_controller

```
import 'package:get/get.dart';
import '../models/note_model.dart';
```

```
class NoteController extends GetxController {
```

```
    var notes = <Note>[].obs;
```

```
    var nextId = 1.obs;
```

```
    void addNote(String title, String description) {
```

```
        notes.add(Note(id: nextId.value, title: title, description: description));
```

```
        nextId++;
    
```

```
    Get.back();
    Get.snackbar('Berhasil', 'Catatan telah ditambahkan');
```

```
}
```

```
    void removeNote(int id) {
```

```
        notes.removeWhere((note) => note.id == id);
    
```

```
    Get.snackbar('Berhasil', 'Catatan telah dihapus');
```

```
}
```

```
}
```

Note\_model

```
class Note {
```

```
    int id;
```

```
    String title;
```

```
    String description;
```

```
    Note({
```

```
required this.id,  
required this.title,  
required this.description,  
});  
}  
  
Add_note_page.dart  
import 'package:flutter/material.dart';  
import 'package:get/get.dart';  
import './controllers/note_controller.dart';  
  
class AddNotePage extends StatelessWidget {  
  final NoteController noteController = Get.find();  
  final TextEditingController titleController = TextEditingController();  
  final TextEditingController descController = TextEditingController();  
  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text('Tambah Catatan Baru'),  
        backgroundColor: Colors.blue[700],  
        leading: IconButton(  
          icon: Icon(Icons.arrow_back),  
          onPressed: () => Get.back(),  
        ),  
      ),  
      body: Padding(  
        padding: const EdgeInsets.all(16.0),  
        child: Column(  
          children: [  
            TextField(  
              controller: titleController,  
              decoration: InputDecoration(  
                
```

```
        labelText: 'Judul',
        border: OutlineInputBorder(),
        prefixIcon: Icon(Icons.title),
      ),
    ),
    SizedBox(height: 16),
    TextField(
      controller: descController,
      decoration: InputDecoration(
        labelText: 'Deskripsi',
        border: OutlineInputBorder(),
        prefixIcon: Icon(Icons.description),
      ),
      maxLines: 5,
    ),
    SizedBox(height: 24),
    ElevatedButton.icon(
      onPressed: () {
        if (titleController.text.isNotEmpty &&
            descController.text.isNotEmpty) {
          noteController.addNote(
            titleController.text, descController.text);
        } else {
          Get.snackbar('Peringatan', 'Judul dan deskripsi wajib diisi');
        }
      },
      icon: Icon(Icons.save),
      label: Text('Simpan Catatan'),
      style: ElevatedButton.styleFrom(
        padding: EdgeInsets.symmetric(horizontal: 24, vertical: 12),
      ),
    ),
  ],
),
```

```
    ),  
    );  
}  
}
```

### Home\_page.dart

```
import 'package:flutter/material.dart';  
import 'package:get/get.dart';  
import './controllers/note_controller.dart';  
import './models/note_model.dart';  
  
class HomePage extends StatelessWidget {  
  final NoteController noteController = Get.put(NoteController());  
  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text('Catatan Saya'),  
        backgroundColor: Colors.blue[700],  
      ),  
      body: Obx(() => noteController.notes.isEmpty  
        ? Center(  
          child: Text(  
            'Belum ada catatan.',  
            style: TextStyle(fontSize: 18, color: Colors.grey),  
          ),  
        )  
      : ListView.builder(  
        itemCount: noteController.notes.length,  
        itemBuilder: (context, index) {  
          Note note = noteController.notes[index];  
          return Card(  
            margin: EdgeInsets.symmetric(horizontal: 12, vertical: 6),
```

```
        child: ListTile(
            title: Text(
                note.title,
                style: TextStyle(fontWeight: FontWeight.bold),
            ),
            subtitle: Text(note.description),
            trailing: IconButton(
                icon: Icon(Icons.delete, color: Colors.red),
                onPressed: () => noteController.removeNote(note.id),
            ),
        ),
    );
},
)),
floatingActionButton: FloatingActionButton(
    onPressed: () => Get.toNamed('/addNote'),
    child: Icon(Icons.add),
    backgroundColor: Colors.blue[700],
),
);
}
}
```

## Main.dart

```
import 'package:flutter/material.dart';
import 'package:get/get.dart';
import 'views/home_page.dart';
import 'views/add_note_page.dart';
```

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

```

class MyApp extends StatelessWidget {
    @override
    Widget build(BuildContext context) {
        return GetMaterialApp(
            title: 'Simple Notes App',
            initialRoute: '/',
            getPages: [
                GetPage(name: '/', page: () => HomePage()),
                GetPage(name: '/addNote', page: () => AddNotePage()),
            ],
            theme: ThemeData(primarySwatch: Colors.blue),
            debugShowCheckedModeBanner: false,
        );
    }
}

```

The screenshot shows a development setup for a Flutter application named "Simple Notes App".

**VS Code (Left):**

- File Explorer:** Shows the project structure with files like `note\_controller.dart`, `main.dart`, `pubspec.yaml`, and `note\_model.dart`.
- Terminal:** Displays the command-line interface with the following output:

```

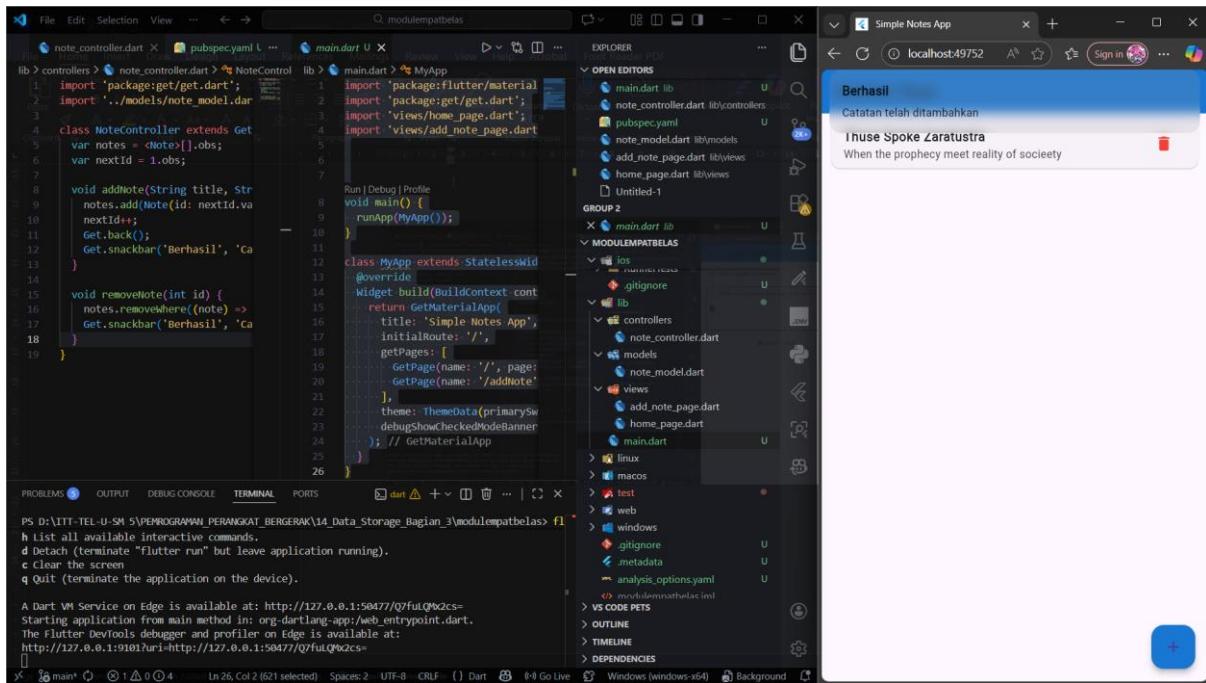
PS D:\ITTEL-U-SM\5\PROGRAMAN_PERANGKAT_BERGERAK\14_Data_Storage_Bagian_3\modulempatbelas> flutter run
[✓] Flutter (Channel stable, v2.5.3, on Windows 10, locale id-ID)
[✗] Android toolchain - develop for Android devices
    ✗ Android Studio not installed. To download it from the Play Store, open https://flutter.dev/docs/get-started/install/macos#android-setup.
[✗] Xcode - develop for iOS and macOS
    ✗ Xcode not installed. To download it from the Mac App Store, open https://flutter.dev/docs/get-started/install/macos#ios-setup.
[✗] Connected device
    ! No devices are connected.

A Dart VM Service on Edge is available at: http://127.0.0.1:50477/07fulQ%2cs=
Starting application from main method in: org-dartlang-app/web_entrypoint.dart.
The Flutter DevTools debugger and profiler on Edge is available at:
http://127.0.0.1:9101?url=http://127.0.0.1:50477/07fulQ%2cs=

```

**Browser (Right):**

The application is running at [localhost:49752/](http://localhost:49752/). The screen shows a form titled "Tambah Catatan Baru" (Add New Note) with fields for "Judul" (Title) containing "Thuse Spoke Zarathustra" and "Deskripsi" (Description) containing "When the prophecy meet reality of society". A "Simpan Catatan" (Save Note) button is visible.



Aplikasi ini merupakan aplikasi catatan sederhana berbasis Flutter yang menggunakan GetX sebagai state management dan navigasi. Pengguna dapat menambahkan, menampilkan, dan menghapus catatan yang terdiri dari judul dan deskripsi. Data catatan dikelola secara reaktif melalui NoteController, sehingga perubahan data langsung memperbarui tampilan tanpa perlu setState. Halaman utama menampilkan daftar catatan, sedangkan halaman tambah catatan menyediakan form input. GetX juga dimanfaatkan untuk perpindahan halaman (routing) serta menampilkan notifikasi (snackbar) sebagai umpan balik atas aksi pengguna.