

**TUGAS PENDAHULUAN
PEMROGRAMAN PERANGKAT BERGERAK**

**MODUL 13
NETWORKING**



Disusun Oleh :

Fahmi Hasan Asagaf/ 2311104074

S1SE-07-02

Dosen Pengampu :

Yudha Islami Sulistya, S.Kom., M.Cs.

PROGRAM STUDI S1 SOFTWARE ENGINEERING

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2025

TUGAS UNGUIDED 11

A. SOAL

SOAL

Buatlah Aplikasi Catatan Sederhana menggunakan GetX, dengan ketentuan sebagai berikut :

1. Halaman utama atau Homepage untuk menampilkan daftar catatan yang telah ditambahkan. Setiap catatan terdiri dari judul dan deskripsi singkat, serta terdapat tombol untuk menghapus catatan dari daftar.
2. Halaman kedua untuk menambah catatan baru, berisi : form untuk memasukkan judul dan deskripsi catatan, serta tombol untuk menyimpan catatan ke daftar (Homepage).
3. Menggunakan getx controller.
4. Menggunakan getx routing untuk navigasi halaman.

JAWABAN

Source Code:

```
1.CLASS NOTE
class Note {
  String title;
  String description;

  Note({required this.title, required this.description});
}

1 Controller (GetX)
import 'package:get/get.dart';
import 'note.dart';

class NoteController extends GetxController {
  var notes = <Note>[].obs;

  void addNote(String title, String desc) {
    notes.add(Note(title: title, description: desc));
  }
}
```

```
void deleteNote(int index) {  
  notes.removeAt(index);  
}  
}
```

3.Main & Routing (GetMaterialApp)

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

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

```
class MyApp extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return GetMaterialApp(  
      debugShowCheckedModeBanner: false,  
      title: 'Catatan GetX',  
      initialRoute: '/',  
      getPages: [  
        GetPage(name: '/', page: () => HomePage()),  
        GetPage(name: '/add', page: () => AddNotePage()),  
      ],  
    );  
  }  
}
```

4.Homepage (Daftar Catatan)

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

class HomePage extends StatelessWidget {
  final NoteController controller = Get.put(NoteController());

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: Text("Daftar Catatan")),
      body: Obx() {
        if (controller.notes.isEmpty) {
          return Center(child: Text("Belum ada catatan"));
        }
        return ListView.builder(
          itemCount: controller.notes.length,
          itemBuilder: (context, index) {
            return Card(
              child: ListTile(
                title: Text(controller.notes[index].title),
                subtitle: Text(controller.notes[index].description),
                trailing: IconButton(
                  icon: Icon(Icons.delete, color: Colors.red),
                  onPressed: () {
                    controller.deleteNote(index);
                  },
                ),
              ),
            );
          },
        );
      },
      floatingActionButton: FloatingActionButton(
```

```

        onPressed: () {
          Get.toNamed('/add');
        },
        child: Icon(Icons.add),
      ),
    );
  }
}

```

Halaman Tambah Catatan

```

import 'package:flutter/material.dart';
import 'package:get/get.dart';
import 'note_controller.dart';

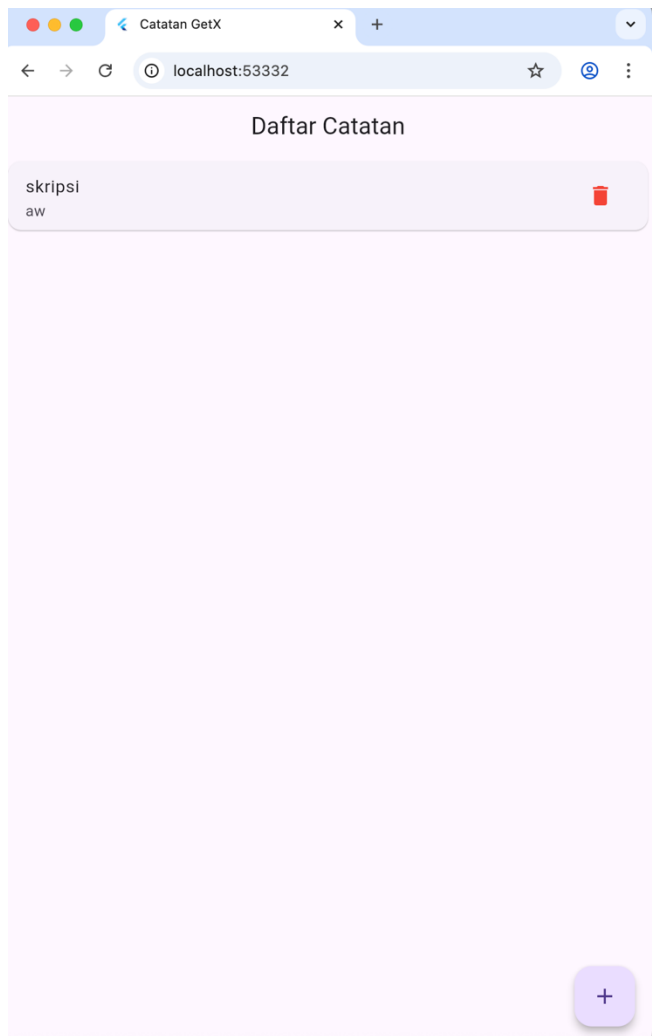
class AddNotePage extends StatelessWidget {
  final NoteController controller = Get.find();
  final TextEditingController titleC = TextEditingController();
  final TextEditingController descC = TextEditingController();

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: Text("Tambah Catatan")),
      body: Padding(
        padding: EdgeInsets.all(16),
        child: Column(
          children: [
            TextField(
              controller: titleC,
              decoration: InputDecoration(labelText: "Judul"),
            ),
            TextField(
              controller: descC,
              decoration: InputDecoration(labelText: "Deskripsi"),
            ),
            SizedBox(height: 20),
          ],
        ),
      ),
    );
  }
}

```

```
ElevatedButton(  
  onPressed: () {  
    controller.addNote(titleC.text, descC.text);  
    Get.back();  
  },  
  child: Text("Simpan"),  
)  
],  
)  
,  
);  
}  
}
```

Output



Penjelasan Program

Aplikasi ini menerapkan state management GetX untuk mengelola daftar catatan secara reaktif. Data catatan disimpan dalam variabel observable (RxList) sehingga UI otomatis diperbarui ketika data berubah. Navigasi antar halaman dilakukan menggunakan GetX Routing tanpa memerlukan BuildContext. Dengan pendekatan ini, kode menjadi lebih ringkas, terstruktur, dan mudah dipelihara