

**LAPORAN PRAKTIKUM
PEMROGRAMAN PERANGKAT BERGERAK**

**MODUL 13
NETWORKING**



Disusun Oleh :

Aji Prasetyo Nugroho / 2211104049

S1SE-06-2

Asisten Praktikum :

Muhammad Faza Zulian Gesit Al Barru

Aisyah Hasna Aulia

Dosen Pengampu :

Yudha Islami Sulistya

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

2024

A. GUIDED

- my_home_page.dart

Source code :

```
import 'package:flutter/material.dart';
import 'package:get/get.dart';
import 'package:pertemuan_13/view model/counter_controler.dart';

class MyHomePage extends StatelessWidget {
  MyHomePage({super.key, required this.title});
  final controller = Get.find<CounterController>();

  final String title;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        backgroundColor: Theme.of(context).colorScheme.inversePrimary,
        title: Text(title),
      ),
      body: Center(
        child: Obx(
          () => Column(
            mainAxisAlignment: MainAxisAlignment.center,
            children: <Widget>[
              const Text(
                'You have pushed the button this many times:',
              ),
              Text(
                controller.counter.toString(),
                style: Theme.of(context).textTheme.headlineMedium,
              ),
              ElevatedButton(
                onPressed: () {
                  Get.toNamed('/Detail');
                },
                child: Text('Go to Detail'),
              )
            ],
          ),
        ),
      ),
      floatingActionButton: Row(
        mainAxisAlignment: MainAxisAlignment.spaceEvenly,
        children: [
          FloatingActionButton(
            onPressed: controller.incrementCounter,
            tooltip: 'Increment',
            child: const Icon(Icons.add),
          ),
          FloatingActionButton(
            onPressed: controller.decrementCounter,
            tooltip: 'Decrement',
```

```

        child: const Icon(Icons.remove),
      ),
      FloatingActionButton(
        onPressed: controller.getsnackbar,
        tooltip: 'Snackbar',
        child: const Icon(Icons.chat),
      ),
      FloatingActionButton(
        onPressed: controller.getdialog,
        tooltip: 'dialog',
        child: const Icon(Icons.notifications_active),
      ),
      FloatingActionButton(
        onPressed: controller.getbottomsheet,
        tooltip: 'bottom sheet',
        child: const Icon(Icons.arrow_upward),
      ),
    ],
  ),
);
}
}

```

- detail_page.dart

Source Code :

```

import 'package:flutter/material.dart';

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

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Halaman Detail'),
      ),
      body: const Center(
        child: Text('Halaman Detail'),
      ),
    );
  }
}

```

- main.dart

Source Code :

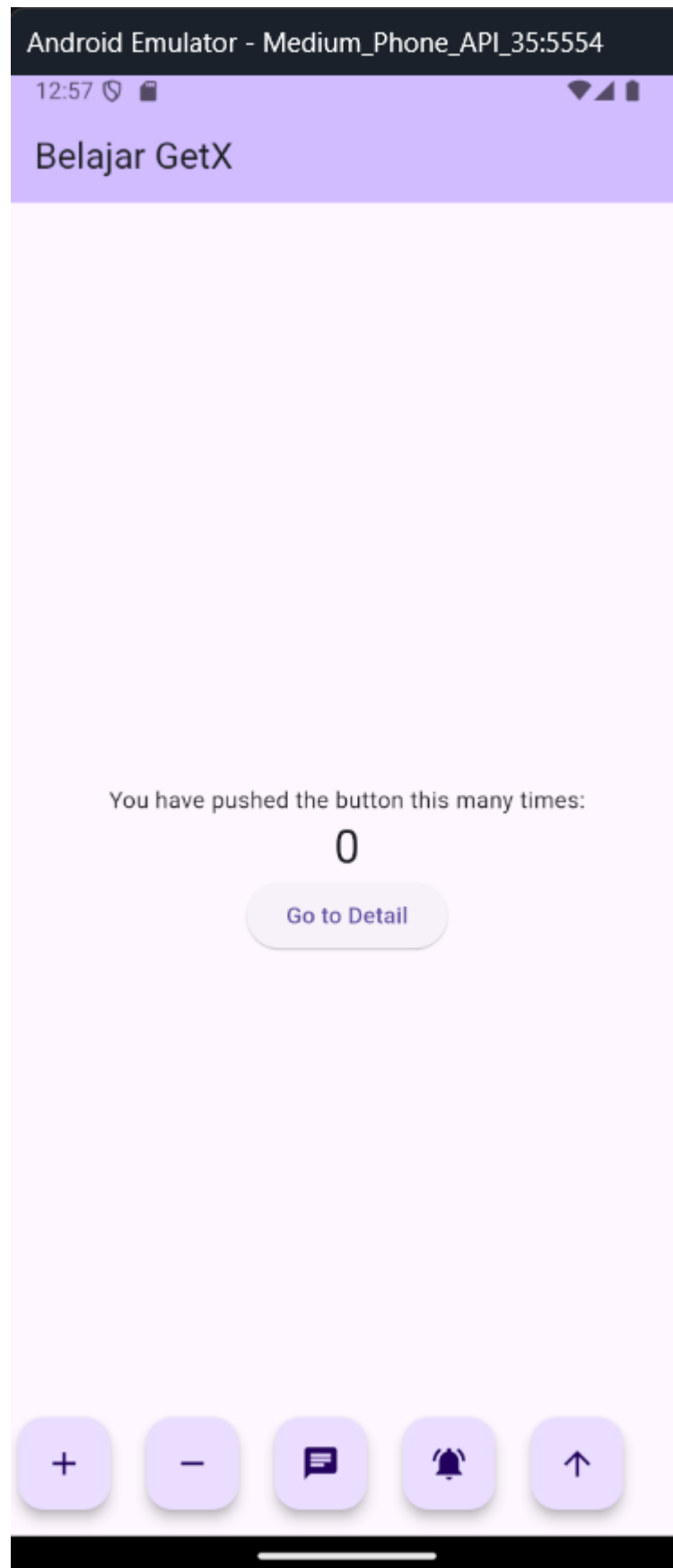
```
import 'package:flutter/material.dart';
import 'package:get/get.dart';
import 'package:pertemuan_13/view_model/counter_controler.dart';
import 'package:pertemuan_13/view/detail_page.dart';
import 'package:pertemuan_13/view/my_home_page.dart';

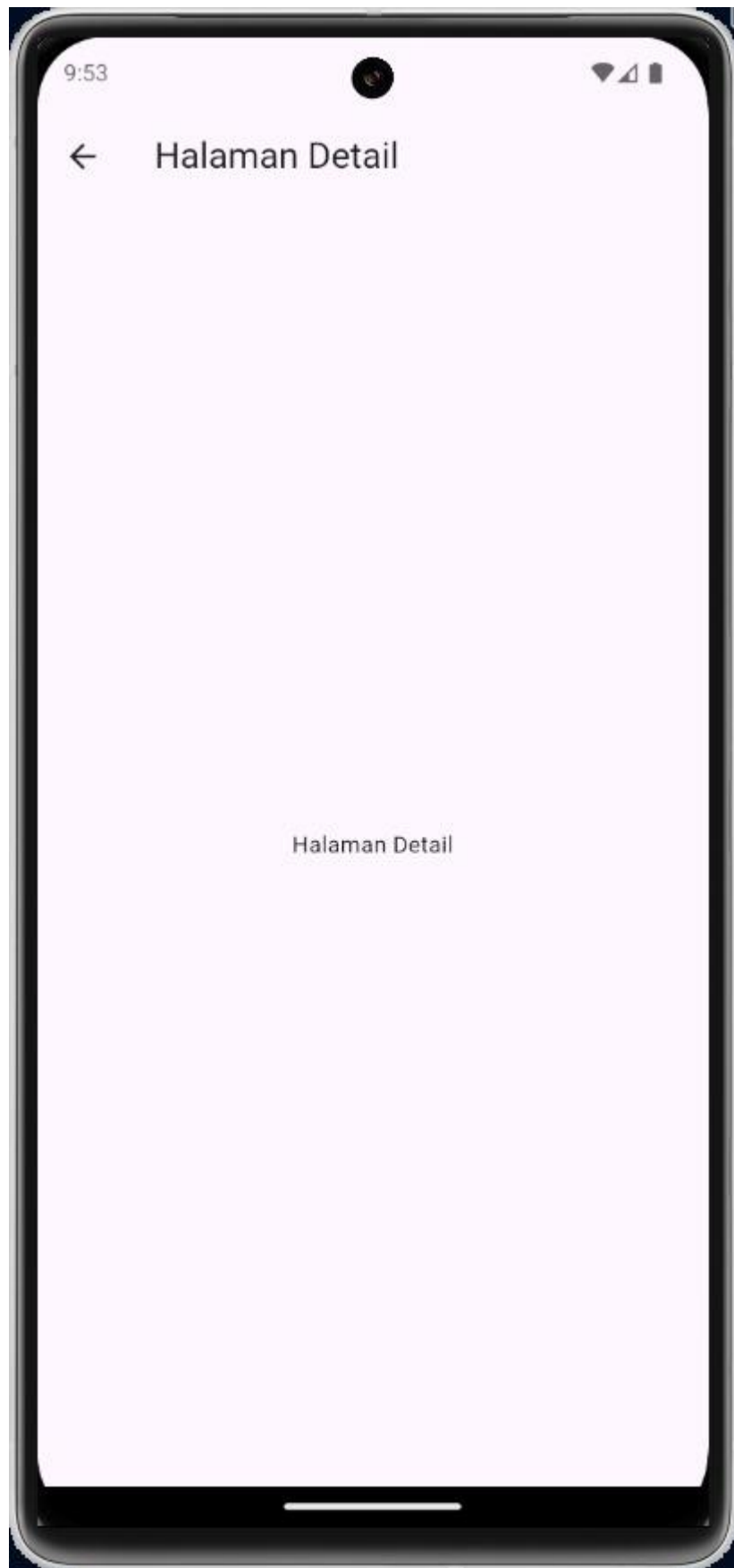
void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  MyApp({super.key});
  final CounterController controller = Get.put(CounterController());

  @override
  Widget build(BuildContext context) {
    return GetMaterialApp(
      debugShowCheckedModeBanner: false,
      initialRoute: '/',
      getPages: [
        GetPage(
          name: '/',
          page: () => MyHomePage(title: 'Belajar GetX'),
        ),
        GetPage(
          name: '/Detail',
          page: () => DetailPage(),
        ),
      ],
    );
  }
}
```

Output :





Belajar GetX

You have pushed the button this many times:

7

[Go to Detail](#)



12:58



GetX

ini snackbar

You have pushed the button this many times:

7

Go to Detail



Belajar GetX

Get X Dialog

ini dialog

Go to Detail



12:58



Belajar GetX

You have pushed the button this many times:

7

Go to Detail

ni bottomsheet

Deskripsi Program

Program tersebut adalah sebuah aplikasi Flutter sederhana yang menggunakan GetX untuk manajemen state dan navigasi. Program ini terdiri dari dua halaman: MyHomePage sebagai halaman utama dan DetailPage sebagai halaman detail. Pada halaman utama, terdapat teks yang menunjukkan jumlah penghitungan tombol yang ditekan, tombol-tombol aksi untuk menambah dan mengurangi nilai, serta tombol navigasi ke halaman detail. Komponen Obx digunakan untuk memperbarui UI secara otomatis berdasarkan perubahan pada nilai counter di dalam controller. Fungsi seperti incrementCounter, decrementCounter, getsnackbar, getdialog, dan getbottomsheet adalah metode dalam controller yang mempermudah pengelolaan aksi pengguna.

Halaman DetailPage hanya berfungsi sebagai placeholder sederhana yang menampilkan teks "Halaman Detail." Navigasi ke halaman ini dilakukan melalui Get.toNamed('/Detail'). Program ini menggunakan FloatingActionButton untuk memberikan beberapa aksi interaktif pada pengguna, seperti menampilkan snackbar, dialog, dan bottom sheet. Dengan pendekatan berbasis GetX, program ini menjadi lebih sederhana dan terorganisir karena manajemen state, dependensi, dan navigasi dikelola secara efisien tanpa boilerplate code tambahan.

B. Unguided

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.

Source Code :

- a. notes_controller.dart

Source code :

```
import 'package:get/get.dart';
import 'package:pertemuan_13/models/note_model.dart';
import 'package:uuid/uuid.dart';

class NotesController extends GetxController {
  RxList<Note> notes = <Note>[].obs;
  final uuid = Uuid();

  void addNote(String title, String description) {
    final newNote = Note(id: uuid.v4(), title: title, description: description);
    notes.add(newNote);
    notes.sort((a, b) => b.createdAt.compareTo(a.createdAt));
  }

  void deleteNote(String id) {
    notes.removeWhere((note) => note.id == id);
  }

  Note? getNoteById(String id) {
    return notes.firstWhereOrNull((note) => note.id == id);
  }
}
```

- b. note_model.dart

Source code :

```
class Note {
  String id;
  String title;
  String description;
  DateTime createdAt;

  Note({
    required this.id,
    required this.title,
```

```

        required this.description,
        DateTime? createdAt,
    }) : createdAt = createdAt ?? DateTime.now();
}

```

c. app_pages.dart

Source code :

```

import 'package:get/get.dart';
import 'package:pertemuan_13/routes/app_routes.dart';
import 'package:pertemuan_13/views/home_page.dart';
import 'package:pertemuan_13/views/add_note_page.dart';

class AppPages {
  static final pages = [
    GetPage(
      name: AppRoutes.HOME,
      page: () => HomePage(),
      transition: Transition.fadeIn),
    GetPage(
      name: AppRoutes.ADD_NOTE,
      page: () => AddNotePage(),
      transition: Transition.rightToLeft),
  ];
}

```

d. app_routes.dart

Source code :

```

class AppRoutes {
  static const HOME = '/';
  static const ADD_NOTE = '/add-note';
}

```

e. add_note.dart

Sorce code :

```

import 'package:flutter/material.dart';
import 'package:get/get.dart';
import 'package:pertemuan_13/controllers/notes_controller.dart';

class AddNotePage extends StatelessWidget {
  final NotesController _controller = Get.find();
  final TextEditingController _titleController = TextEditingController();
  final TextEditingController _descriptionController = TextEditingController();
  final _formKey = GlobalKey<FormState>();

  AddNotePage({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Tambah Catatan Baru'),
        centerTitle: true,
      ),
      body: Form(
        key: _formKey,

```

```

child: ListView(
  padding: EdgeInsets.all(16),
  children: [
    TextFormField(
      controller: _titleController,
      decoration: InputDecoration(
        labelText: 'Judul Catatan',
        prefixIcon: Icon(Icons.title),
        border:
          OutlineInputBorder(borderRadius: BorderRadius.circular(12)),
      ),
      validator: (value) {
        if (value == null || value.isEmpty) {
          return 'Judul tidak boleh kosong';
        }
        return null;
      },
    ),
    SizedBox(height: 16),
    TextFormField(
      controller: _descriptionController,
      maxLines: 5,
      decoration: InputDecoration(
        labelText: 'Deskripsi Catatan',
        prefixIcon: Icon(Icons.description),
        border:
          OutlineInputBorder(borderRadius: BorderRadius.circular(12)),
      ),
      validator: (value) {
        if (value == null || value.isEmpty) {
          return 'Deskripsi tidak boleh kosong';
        }
        return null;
      },
    ),
    SizedBox(height: 24),
    ElevatedButton.icon(
      onPressed: _saveNote,
      icon: Icon(Icons.save),
      label: Text('Simpan Catatan'),
      style: ElevatedButton.styleFrom(
        padding: EdgeInsets.symmetric(vertical: 16),
        shape: RoundedRectangleBorder(
          borderRadius: BorderRadius.circular(12)),
      ),
    ),
  ],
),
);
}

void _saveNote() {
  if (_formKey.currentState!.validate()) {

```

```

        _controller.addNote(
            _titleController.text.trim(), _descriptionController.text.trim());
    Get.back();
    Get.snackbar(
        'Berhasil',
        'Catatan baru telah disimpan',
        snackPosition: SnackPosition.BOTTOM,
        backgroundColor: Colors.green.shade500,
        colorText: Colors.white,
    );
}
}
}

```

f. note_card.dart

Source code :

```

import 'package:flutter/material.dart';
import 'package:get/get.dart';
import 'package:intl/intl.dart';
import 'package:pertemuan_13/controllers/notes_controller.dart';
import 'package:pertemuan_13/models/note_model.dart';

class NoteCard extends StatelessWidget {
    final Note note;
    final NotesController _controller = Get.find();

    NoteCard({super.key, required this.note});

    @override
    Widget build(BuildContext context) {
        return Card(
            elevation: 4,
            margin: EdgeInsets.symmetric(vertical: 8),
            shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(12)),
            child: Dismissible(
                key: Key(note.id),
                direction: DismissDirection.endToStart,
                background: _buildDeleteBackground(),
                confirmDismiss: (direction) async {
                    return await _showDeleteConfirmation(context);
                },
                onDismissed: (direction) {
                    _controller.deleteNote(note.id);
                },
                child: ListTile(
                    title: Text(
                        note.title,
                        style: Theme.of(context).textTheme.titleMedium,
                        maxLines: 1,
                        overflow: TextOverflow.ellipsis,
                    ),
                    subtitle: Column(
                        crossAxisAlignment: CrossAxisAlignment.start,
                        children: [

```

```

        Text(
          note.description,
          maxlines: 2,
          overflow: TextOverflow.ellipsis,
        ),
        SizedBox(height: 4),
        Text(
          DateFormat('dd MMM yyyy HH:mm').format(note.createdAt),
          style: Theme.of(context).textTheme.bodySmall,
        )
      ],
    ),
    trailing: IconButton(
      icon: Icon(Icons.delete, color: Colors.red),
      onPressed: () => _confirmDelete(context),
    ),
  ),
);
}

Widget _buildDeleteBackground() {
  return Container(
    color: Colors.red,
    alignment: Alignment.centerRight,
    padding: EdgeInsets.only(right: 20),
    child: Icon(Icons.delete, color: Colors.white),
  );
}

Future<bool?> _showDeleteConfirmation(BuildContext context) {
  return showDialog<bool>({
    context: context,
    builder: (context) => AlertDialog(
      title: Text('Hapus Catatan'),
      content: Text('Apakah Anda yakin ingin menghapus catatan ini?'),
      actions: [
        TextButton(
          onPressed: () => Navigator.of(context).pop(false),
          child: Text('Batal'),
        ),
        ElevatedButton(
          onPressed: () => Navigator.of(context).pop(true),
          style: ElevatedButton.styleFrom(backgroundColor: Colors.red),
          child: Text('Hapus'),
        ),
      ],
    ),
  );
}

void _confirmDelete(BuildContext context) {
  showDialog(

```



```

        context: context,
        builder: (context) => AlertDialog(
          title: Text('Hapus Catatan'),
          content: Text('Apakah Anda yakin ingin menghapus catatan ini?'),
          actions: [
            TextButton(
              onPressed: () => Navigator.of(context).pop(),
              child: Text('Batal'),
            ),
            ElevatedButton(
              onPressed: () {
                _controller.deleteNote(note.id);
                Navigator.of(context).pop();
              },
              style: ElevatedButton.styleFrom(backgroundColor: Colors.red),
              child: Text('Hapus'),
            ),
          ],
        ),
      ),
    );
  }
}

```

g. home_page.dart

Source code :

```

import 'package:flutter/material.dart';
import 'package:get/get.dart';
import 'package:pertemuan_13/controllers/notes_controller.dart';
import 'package:pertemuan_13/routes/app_routes.dart';
import 'package:pertemuan_13/views/widgets/note_card.dart';

class HomePage extends StatelessWidget {
  final NotesController _controller = Get.put(NotesController());

  HomePage({super.key});

  @override

  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text(
          'Catatan Saya',
          style: Theme.of(context).textTheme.titleLarge,
        ),
        centerTitle: true,
        actions: [
          IconButton(
            icon: Icon(Icons.search),
            onPressed: () {
              // Implementasi fitur pencarian
            },
          ),
        ],
      ),
    );
  }
}

```

```

    ),
    body: Obx(() => _controller.notes.isEmpty
      ? _buildEmptyState(context)
      : _buildNotesList(context)),
    floatingActionButton: FloatingActionButton.extended(
      onPressed: () => Get.toNamed(AppRoutes.ADD_NOTE),
      icon: Icon(Icons.add),
      label: Text('Tambah Catatan'),
    ),
  ),
);
}

Widget _buildEmptyState(BuildContext context) {
  return Center(
    child: Column(
      mainAxisAlignment: MainAxisAlignment.center,
      children: [
        Icon(
          Icons.note_add,
          size: 100,
          color: Theme.of(context).colorScheme.secondary,
        ),
        SizedBox(height: 16),
        Text(
          'Belum Ada Catatan',
          style: Theme.of(context).textTheme.titleLarge,
        ),
        SizedBox(height: 8),
        Text(
          'Tekan tombol + untuk membuat catatan baru',
          style: Theme.of(context).textTheme.bodyMedium,
        ),
      ],
    ),
  );
}

Widget _buildNotesList(BuildContext context) {
  return ListView.builder(
    padding: EdgeInsets.all(8),
    itemCount: _controller.notes.length,
    itemBuilder: (context, index) {
      final note = _controller.notes[index];
      return NoteCard(note: note);
    },
  );
}
}

```

h. main.dart

Source code :

```

import 'package:flutter/material.dart';
import 'package:get/get.dart';
import 'package:pertemuan_13/routes/app_pages.dart';

```

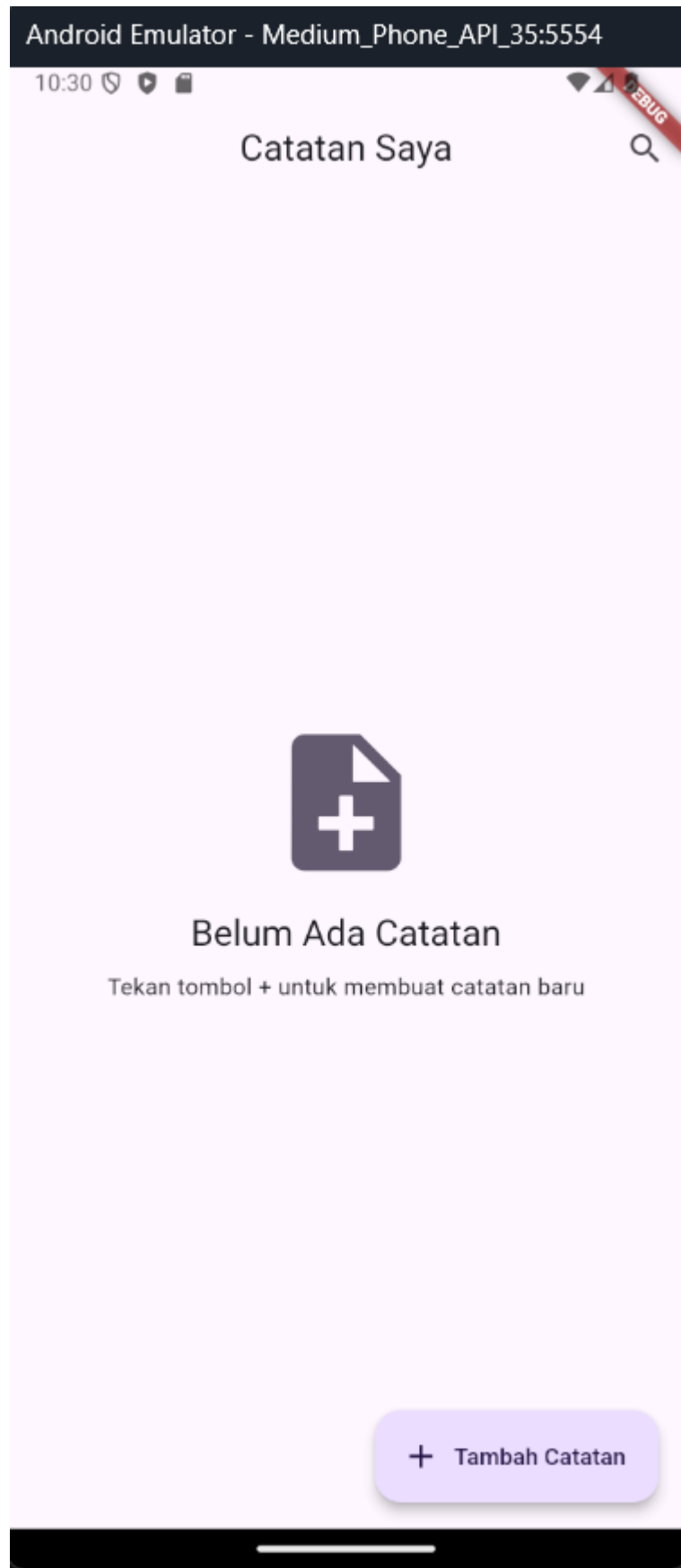
```
import 'package:pertemuan_13/routes/app_routes.dart';

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

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

  @override
  Widget build(BuildContext context) {
    return GetMaterialApp(
      title: 'Catatan Sederhana',
      theme: ThemeData(
        colorScheme: ColorScheme.fromSeed(
          seedColor: Colors.deepPurple,
          brightness: Brightness.light,
        ),
        useMaterial3: true,
      ),
      darkTheme: ThemeData(
        colorScheme: ColorScheme.fromSeed(
          seedColor: Colors.deepPurple,
          brightness: Brightness.dark,
        ),
        useMaterial3: true,
      ),
      initialRoute: AppRoutes.HOME,
      getPages: AppPages.pages,
    );
  }
}
```

Output :



10:30



Tambah Catatan Baru


Judul Catatan

T Laundry

Deskripsi Catatan

Baju = 5

Celana = 3

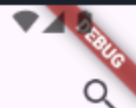
 Flannel = 3



Simpan Catatan



10:31



Catatan Saya



Prak PPB

Aji Prasetyo

Nugroho_2211104049_S1SE-06-2

13 Dec 2024 22:31



Laundry

Baju = 5

Celana = 3

13 Dec 2024 22:30



Berhasil

Catatan baru telah disimpan

10:32

DEBUG

Catatan Saya



Prak PPB

Aji Prasetyo

Nugroho_2211104049_S1SE-06-2

13 Dec 2024 22:31



Laundry

Baju = 5

Celana = 3

13 Dec 2024 22:30



Hapus Catatan

Apakah Anda yakin ingin menghapus catatan ini?

Batal

Hapus

+ Tambah Catatan

10:32



Catatan Saya



Prak PPB

Aji Prasetyo

Nugroho_2211104049_S1SE-06-2

13 Dec 2024 22:31



+ Tambah Catatan

Penjelasan Program

Program di atas adalah aplikasi Flutter berbasis GetX untuk mencatat dan mengelola catatan pribadi. Aplikasi ini memiliki beberapa fitur utama, seperti menambah catatan baru, menampilkan daftar catatan yang sudah dibuat, dan menghapus catatan. Komponen utama aplikasi terdiri dari tiga bagian: halaman utama (HomePage), halaman tambah catatan (AddNotePage), dan widget kartu catatan (NoteCard). Pada halaman utama, catatan ditampilkan dalam bentuk daftar menggunakan widget ListView. Jika tidak ada catatan, akan ditampilkan pesan kosong dengan ikon sebagai indikator. Halaman ini juga menyediakan tombol untuk menambah catatan baru dengan navigasi ke halaman AddNotePage menggunakan routing yang didefinisikan dengan GetX.

Halaman AddNotePage memungkinkan pengguna menambahkan catatan baru dengan mengisi formulir yang terdiri dari judul dan deskripsi. Validasi diterapkan untuk memastikan bahwa kedua input tidak kosong. Setelah catatan disimpan, pengguna diarahkan kembali ke halaman utama dan notifikasi menggunakan Get.snackbar akan muncul sebagai konfirmasi. Selain itu, setiap catatan ditampilkan menggunakan widget NoteCard, yang mendukung penghapusan catatan melalui fitur geser (Dismissible) atau tombol hapus. Konfirmasi dialog muncul sebelum catatan dihapus untuk memastikan keamanan data pengguna. Secara keseluruhan, aplikasi ini dirancang sederhana dengan fokus pada pengalaman pengguna yang efisien untuk mengelola catatan sehari-hari.