

TUGAS 08
PRAK09 Task App



Praktikum Pemrograman Berbasis Web - A

Nama :

Adelia Nurlina Putri

4521210059

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS PANCASILA

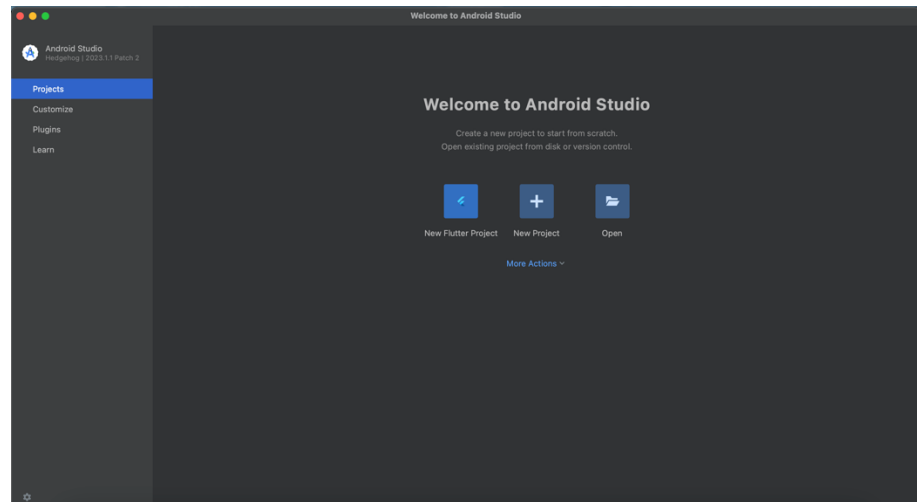
JAKARTA

2024

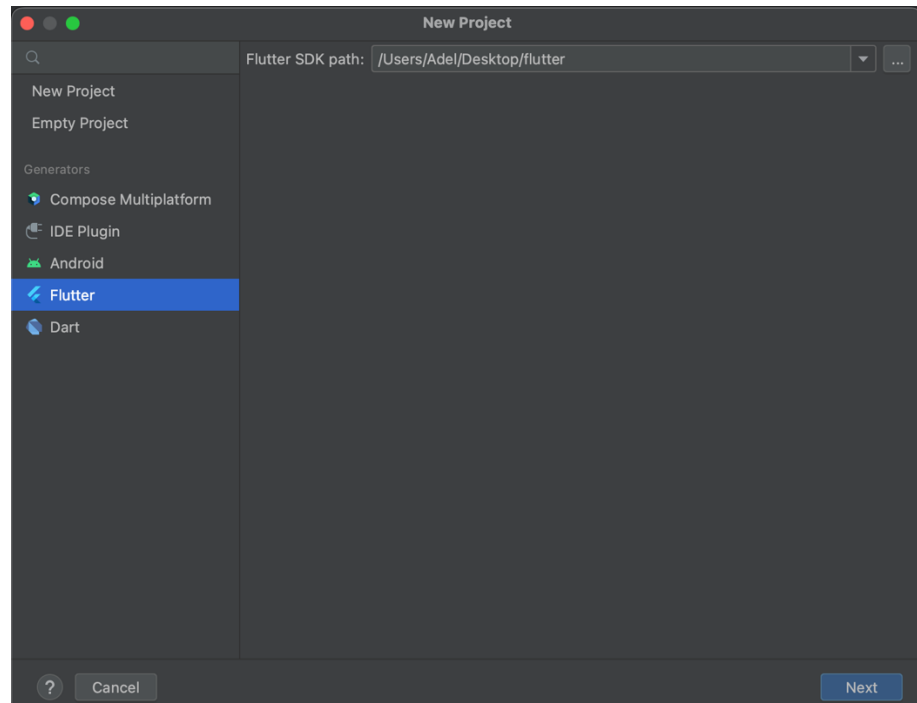
A. Project prak08

a. First Initialization

1. Akses web <https://docs.flutter.dev/ui/layout/tutorial>
2. Buka Android Studio



3. Pilih New Flutter Project -> Flutter -> Next



4. Isi form untuk nama project, tempat project, type, bahasa pemrograman, platforms dan lainnya -> Create

New Project

Project name: task_app

Project location: ~/AndroidStudioProjects/task_app

Description: A new Flutter project.

Project type: Application

Organization: com.example

Android language: ☐ Java ☒ Kotlin

iOS language: ☐ Objective-C ☒ Swift

Platforms: ☒ Android ☒ iOS ☐ Linux ☐ MacOS ☒ Web ☐ Windows

When created, the new project will run on the selected platforms (others can be added later).

☐ Create project offline

▼ More Settings

Module name: task_app

Content root: /Users/Adel/AndroidStudioProjects/task_app

Module file location: /Users/Adel/AndroidStudioProjects/task_app

Project format: .idea (directory-based)

? Cancel Previous Create

b. Membuat Ap

1. Add Package Provider

```
$ flutter pub add provider
```

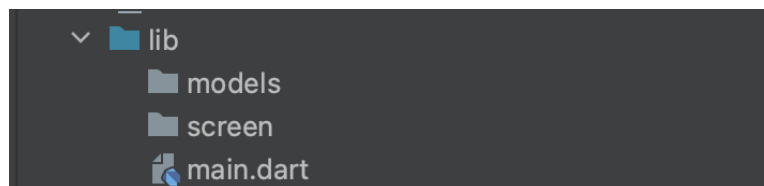
```
Terminal: Local x + v
Adel@anps task_app % flutter pub add provider
Resolving dependencies...
  flutter_lints 3.0.2 (4.0.0 available)
  leak_tracker 10.0.0 (10.0.5 available)
  leak_tracker_flutter_testing 2.0.1 (3.0.5 available)
  leak_tracker_testing 2.0.1 (3.0.1 available)
  lints 3.0.0 (4.0.0 available)
  material_color_utilities 0.8.0 (0.11.1 available)
  meta 1.11.0 (1.15.0 available)
+ nested 1.0.0
+ provider 6.1.2
  test_api 0.6.1 (0.7.1 available)
  vm_service 13.0.0 (14.2.2 available)
Changed 2 dependencies!
9 packages have newer versions incompatible with dependency constraints.
Try `flutter pub outdated` for more information.
Adel@anps task_app %
```

```

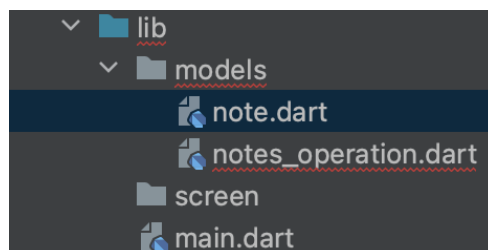
32  sdk: flutter
33
34
35  # The following adds the CupertinoIcons class to the Flutter SDK's default
36  # Use with the CupertinoIcons class for iOS-style icons
37  cupertino_icons: ^1.0.6
38  provider: ^6.1.2
39
40  dev_dependencies:
41    flutter_test:
42      sdk: flutter

```

2. Membuat folder models untuk memasukan kela Note dan NotesOperation yang berfungsi untuk membuat String dan membuat object note serta list note
3. Membuat folder screen untuk membuat tampilan notes dan tampilan notes ketika akan menambahkan note atau task baru.



4. File file notes_operation.dart dan note.dart pada folder models



5. File notes_operation.dart

```

import 'package:flutter/cupertino.dart';
import 'package:task_app/models/note.dart';
class NotesOperation with ChangeNotifier {
  //List of note
  List<Note> _notes = [];
  List<Note> get getNotes {
    return _notes;
  }
  NotesOperation() {
    addNewNote('First Note', 'First Note
Description');
  }
}

```

```
void addNewNote(String title, String description) {
  Note note = Note(title, description);
  _notes.add(note);
  notifyListeners();
}
}
```

6. File note.dart

```
class Note {
  String title;
  String description;

  Note(this.title, this.description);
}
```

7. Membuat file home_screen.dart dan add_screen.dart

8. File home_screen.dart

```
import 'package:flutter/material.dart';
import 'package:provider/provider.dart';
import 'package:task_app/models/note.dart';
import 'package:task_app/models/notes_operation.dart';
import 'package:task_app/screen/add_screen.dart';

class HomeScreen extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.blueGrey,
      floatingActionButton: FloatingActionButton(
        onPressed: () {
          Navigator.push(
            context,
            MaterialPageRoute(
              builder: (context) => AddScreen(),
            ),
          );
        },
        child: Icon(Icons.add, size: 30, color:
Colors.blueGrey),
        backgroundColor: Colors.white,
      ),
      appBar: AppBar(
        title: Text(
          'Task Manager',
          style: TextStyle(
            fontSize: 25,
            fontWeight: FontWeight.bold,
          ),
        ),
        centerTitle: true,
        elevation: 0,
      ),
    );
  }
}
```

```

        backgroundColor: Colors.transparent,
      ),
      body: Consumer<NotesOperation>(
        builder: (context, NotesOperation data, child)
      {
        return ListView.builder(
          itemCount: data.getNotes.length,
          itemBuilder: (context, index) {
            return NotesCard(data.getNotes[index]);
          },
        );
      },
    ),
  );
}

class NotesCard extends StatelessWidget {
  final Note note;

  NotesCard(this.note);

  @override
  Widget build(BuildContext context) {
    return Container(
      margin: EdgeInsets.all(15),
      padding: EdgeInsets.all(15),
      height: 150,
      decoration: BoxDecoration(
        color: Colors.white,
        borderRadius: BorderRadius.circular(15),
      ),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [
          Text(
            note.title,
            style: TextStyle(fontSize: 20, fontWeight:
FontWeight.bold),
          ),
          SizedBox(height: 5),
          Text(
            note.description,
            style: TextStyle(fontSize: 17),
          ),
        ],
      ),
    );
  }
}

```

9. File add_screen.dart

```

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

```

```

import 'package:task_app/models/notes_operation.dart';

class AddScreen extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    String titleText = '';
    String descriptionText = '';

    return Scaffold (
      backgroundColor: Colors.blueGrey,
      appBar: AppBar (
        title: Text('Task Manager'),
        centerTitle: true,
        elevation: 0,
        backgroundColor: Colors.transparent,
      ),
      body: Padding (
        padding: EdgeInsets.only(
          top: 15,
          left: 15,
          right: 15,
          bottom: 80,
        ),
        child: Column (
          children: [
            TextField(
              decoration: InputDecoration(
                border: InputBorder.none,
                hintText: 'Title',
                hintStyle: TextStyle(
                  fontSize: 20,
                  fontWeight: FontWeight.bold,
                  color: Colors.white,
                ),
              ),
              style: TextStyle(
                fontSize: 20,
                fontWeight: FontWeight.bold,
                color: Colors.white,
              ),
              onChanged: (value) {
                titleText = value;
              },
            ),
            Expanded (
              child: TextField(
                decoration: InputDecoration(
                  border: InputBorder.none,
                  hintText: 'Enter Description',
                  hintStyle: TextStyle (
                    fontSize: 18,
                    color: Colors.white,
                  ),
                ),
                style: TextStyle(
                  fontSize: 18,

```

```

        color: Colors.white,
      ),
      onChanged: (value) {
        descriptionText = value;
      },
    ),
  ),
  TextButton (
    onPressed: () {
      Provider.of<NotesOperation>(context, listen:
false)
        .addNewNote(titleText, descriptionText);
      Navigator.pop(context);
    },
    style: TextButton.styleFrom(
      padding:
        EdgeInsets.only(left: 30, right: 30, top: 10,
bottom: 10),
      backgroundColor: Colors.white,
    ),
    child: Text ('Add Note',
      style: TextStyle(
        fontSize: 20,
        fontWeight: FontWeight.bold,
        color: Colors.blueGrey,
      )),
  ),
],
),
),
);
}
}

```

10. File main.dart

```

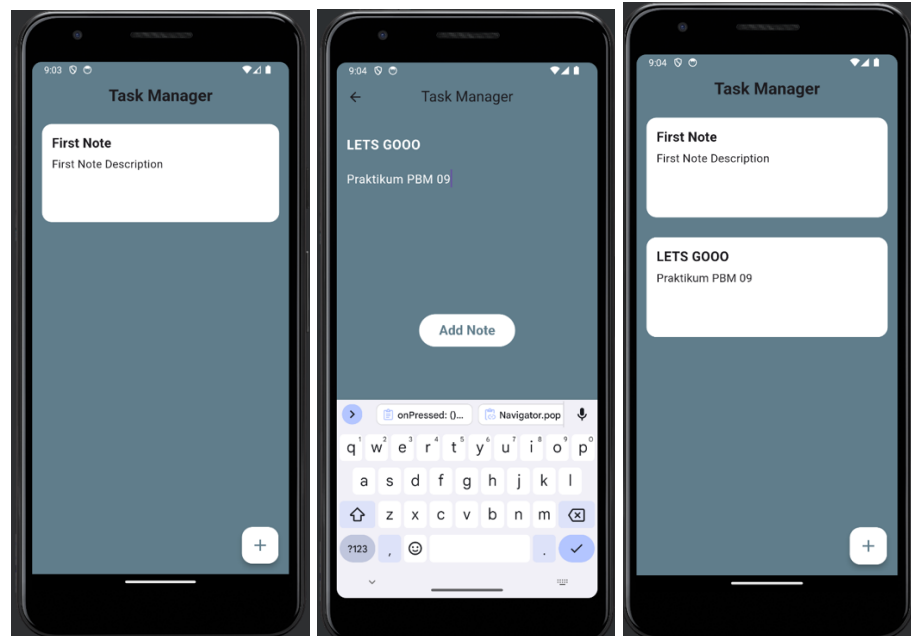
import 'package:flutter/material.dart';
import 'package:provider/provider.dart';
import 'package:task_app/models/notes_operation.dart';
import 'package:task_app/screen/home_screen.dart';

void main () => runApp(MyApp ());

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return ChangeNotifierProvider<NotesOperation>(
      create: (context) => NotesOperation(),
      child: MaterialApp(
        debugShowCheckedModeBanner: false,
        home: HomeScreen(),
      ),
    );
  }
}

```


Kode di atas akan membuat aplikasi Flutter dengan state management provider untuk mengelola tugas aplikasi



c. **Push Code to GitHub** https://github.com/adelianurlinap/Prak_PBM.git

1. Clone Repository yang telah dibuat :

```
git clone https://github.com/adelianurlinap/Prak_PBM.git
```

2. Add dan commit

```
git add .  
git commit -m "Tugas8"
```

3. Push code

```
git push
```

B. Kesimpulan

Pada tugas 8 ini, telah dilakukan praktek untuk membuat aplikasi dengan state management provider

C. Referensi

modul prak09