

LAPORAN PRAKTIKUM MOBILE PROGRAMMING
MODUL 8



Nama : Mauluda Aminaditya

NIM : 240605110132

Kelas : B

Tanggal : 6 Oktober 2025

JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
GANJIL
2025/2026

I. Tujuan

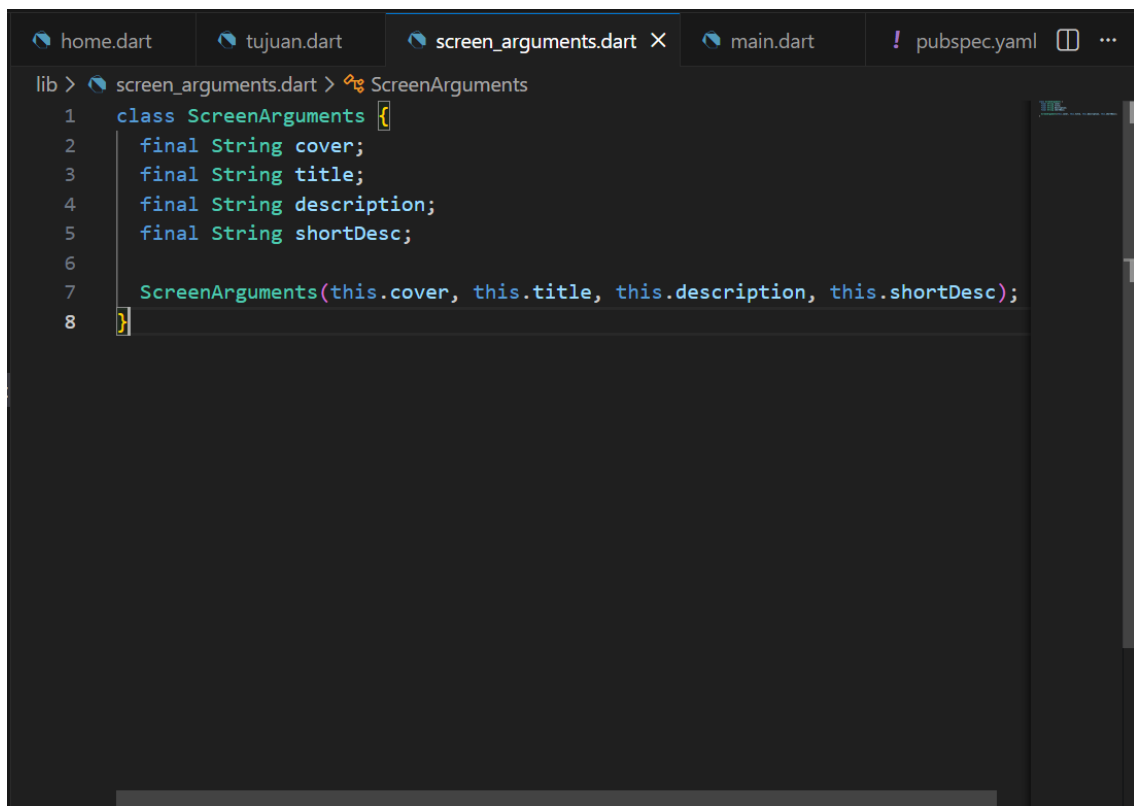
1. Mengimplementasikan navigasi antar halaman menggunakan NamedRoute.
2. Mengirim data dari halaman Home ke halaman Tujuan menggunakan argument.

II. Langkah Kerja

1. Membuat project Flutter baru
2. Menambahkan dependency http
3. Membuat model ScreenArguments
4. Mengambil data game dari API FreeToGame
5. Menampilkan data pada halaman Home
6. Mengirim data ke halaman Tujuan menggunakan Named Route
7. Menampilkan detail game di halaman Tujuan

III. Screenshot Hasil

a. Code screen_arguments.dart



```
lib > screen_arguments.dart > ScreenArguments
1 class ScreenArguments {
2   final String cover;
3   final String title;
4   final String description;
5   final String shortDesc;
6
7   ScreenArguments(this.cover, this.title, this.description, this.shortDesc);
8 }
```

Code ini untuk membungkus data game agar bisa dikirim antar halaman.

b. Code main.dart

```
lib > home.dart  tujuan.dart  screen_arguments.dart  main.dart x  ! pubspec
lib > main.dart > MyApp > build
1  import 'package:flutter/material.dart';
2  import 'package:flutter/services.dart';
3  import 'home.dart';
4  import 'tujuan.dart';
5
Run | Debug | Profile
6  void main() {
7    runApp(const MyApp());
8  }
9
10 class MyApp extends StatelessWidget {
11   const MyApp({super.key});
12
13   @override
14   Widget build(BuildContext context) {
15     SystemChrome.setSystemUIOverlayStyle(
16       const SystemUiOverlayStyle(statusBarColor: Colors.transparent),
17     );
18
19     return MaterialApp(
20       debugShowCheckedModeBanner: false,
21       initialRoute: '/',
22       routes: {
23         '/': (context) => const Home(),
24         Tujuan.routeName: (context) => const Tujuan(),
25       },
26     ); // MaterialApp
27   }
28 }
```

c. Code home.dart

```
home.dart X  tujuan.dart  screen_arguments.dart  main.dart  ! pubspec.yaml  ...

lib > home.dart > _HomeState > build
1  import 'package:flutter/material.dart';
2  import 'dart:convert';
3  import 'package:http/http.dart' as http;
4  import 'screen_arguments.dart';
5  import 'tujuan.dart';
6
7  class Home extends StatefulWidget {
8    const Home({super.key});
9
10   @override
11   State<Home> createState() => _HomeState();
12 }
13
14 class _HomeState extends State<Home> {
15   var title,
16       thumbnail,
17       short_description,
18       description,
19       genre,
20       platform,
21       release,
22       cover,
23       gameid,
24       publisher;
25
26   Future<void> getGame(String gameid) async {
27     http.Response response = await http.get(
28       Uri.parse('https://www.freetogame.com/api/game?id=$gameid'),
29     );
30
31     var results = jsonDecode(response.body);
32
33     setState(() {
```

```

34     this.gameid = gameid;
35     title = results['title'];
36     thumbnail = results['thumbnail'];
37     short_description = results['short_description'];
38     description = results['description'];
39     genre = results['genre'];
40     platform = results['platform'];
41     publisher = results['publisher'];
42     release = results['release_date'];
43     cover = results['screenshots'][0]['image'];
44   });
45 }
46
47 @override
48 void initState() {
49   super.initState();
50   getGame('475'); // game default
51 }
52
53 @override
54 Widget build(BuildContext context) {
55   return Scaffold(
56     backgroundColor: const Color(0xFF0081c9),
57     body: SafeArea(
58       child: Center(
59         child: gameid == null
60           ? const CircularProgressIndicator(
61             color: Colors.white,
62           ) // CircularProgressIndicator
63           : GestureDetector(
64             onTap: () {

```

```

65       Navigator.pushNamed(
66         context,
67         Tujuan.routeName,
68         arguments: ScreenArguments(
69           cover,
70           title,
71           description,
72           short_description,
73         ), // ScreenArguments
74       );
75     },
76     child: Container(
77       padding: const EdgeInsets.all(15),
78       margin: const EdgeInsets.all(15),
79       decoration: BoxDecoration(
80         color: Colors.white,
81         borderRadius: BorderRadius.circular(15),
82       ), // BoxDecoration
83       child: Column(
84         mainAxisAlignment: MainAxisAlignment.min,
85         children: [
86           Image.network(thumbnail),
87           const SizedBox(height: 15),
88           Text(
89             title,
90             style: const TextStyle(
91               fontSize: 24,
92               fontWeight: FontWeight.bold,
93             ), // TextStyle
94             textAlign: TextAlign.center,

```

```

95     ), // Text
96     const SizedBox(height: 10),
97     Row(
98       mainAxisAlignment: MainAxisAlignment.spaceAround,
99       children: [
100         Column(
101           crossAxisAlignment: CrossAxisAlignment.start,
102           children: [
103             Text("Genre: $genre"),
104             Text("Platform: $platform"),
105           ],
106         ), // Column
107         Column(
108           crossAxisAlignment: CrossAxisAlignment.start,
109           children: [
110             Text("Publisher: $publisher"),
111             Text("Release: $release"),
112           ],
113         ), // Column
114       ],
115     ), // Row
116     const SizedBox(height: 10),
117     Text(
118       short_description,
119       textAlign: TextAlign.center,
120     ), // Text
121   ],
122 ), // Column
123 ), // Container
124 ), // GestureDetector

```

```

125   ), // Center
126 ), // SafeArea
127 ); // Scaffold
128 }
129 }

```

d. Code tujuan_dart

```
lib > tujuan.dart > Tujuan
1 import 'package:flutter/material.dart';
2 import 'package:flutter/services.dart';
3 import 'screen_arguments.dart';
4
5 class Tujuan extends StatelessWidget {
6   const Tujuan({super.key});
7   static const routeName = '/extractArguments';
8
9   @override
10  Widget build(BuildContext context) {
11    SystemChrome.setSystemUIOverlayStyle(
12      const SystemUiOverlayStyle(statusBarColor: Colors.transparent),
13    );
14
15    final args =
16      ModalRoute.of(context)!.settings.arguments as ScreenArguments;
17
18    return Scaffold(
19      backgroundColor: const Color(0xFF94A29),
20      body: SafeArea(
21        child: SingleChildScrollView(
22          child: Container(
23            margin: const EdgeInsets.all(15),
24            child: Column(
25              children: [
26                Image.network(args.cover),
27                const SizedBox(height: 15),
28                Text(
29                  args.title,
30                  style: const TextStyle(
31                    fontSize: 20,
32                    fontWeight: FontWeight.bold,
33                  ), // TextStyle
34                ), // Text
35                const SizedBox(height: 10),
36                Text(args.description),
37                const SizedBox(height: 20),
38                ElevatedButton(
39                  onPressed: () {
40                    Navigator.pop(context);
41                  },
42                  child: const Text('Kembali'),
43                ), // ElevatedButton
44              ],
45            ), // Column
46          ), // Container
47        ), // SingleChildScrollView
48      ), // SafeArea
49    ); // Scaffold
50  }
51 }
```

Program ini merupakan aplikasi Flutter yang menampilkan informasi game dengan mengambil data secara real-time dari API FreeToGame. Data game diambil menggunakan paket http dan diolah dalam format JSON, kemudian ditampilkan pada halaman Home. Selama proses pengambilan data

berlangsung, aplikasi menampilkan indikator loading.

Navigasi antar halaman dilakukan menggunakan Named Route, di mana data game dikirim dari halaman Home ke halaman Tujuan melalui objek ScreenArguments. Pada halaman Tujuan, data tersebut diterima menggunakan ModalRoute dan ditampilkan sebagai detail informasi game. Tombol kembali disediakan untuk kembali ke halaman Home.

Aplikasi ini menunjukkan penerapan konsep REST API, Named Route, dan pengiriman data menggunakan argument dalam pengembangan aplikasi Flutter.

IV. Kesimpulan

Praktikum ini menunjukkan bahwa navigasi antar halaman pada Flutter dapat dilakukan dengan mudah menggunakan Named Route. Pengiriman data antar halaman dapat dilakukan secara efisien menggunakan arguments yang dikemas dalam sebuah objek. Selain itu, penggunaan REST API FreeToGame memungkinkan aplikasi menampilkan data game secara real-time dan membuat aplikasi lebih dinamis.