

**LAPORAN**  
**MATA KULIAH PEMROGRAMAN MOBILE**  
**JOBSHEET WEEK 6 LAYOUT DAN NAVIGASI**  
**DOSEN PENGAMPU: Rudy Ariyanto, ST., M.Cs.**



**NAMA : NADYA HAPSARI PUTRI**

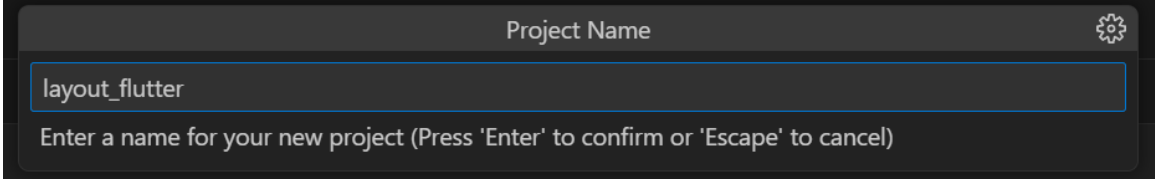
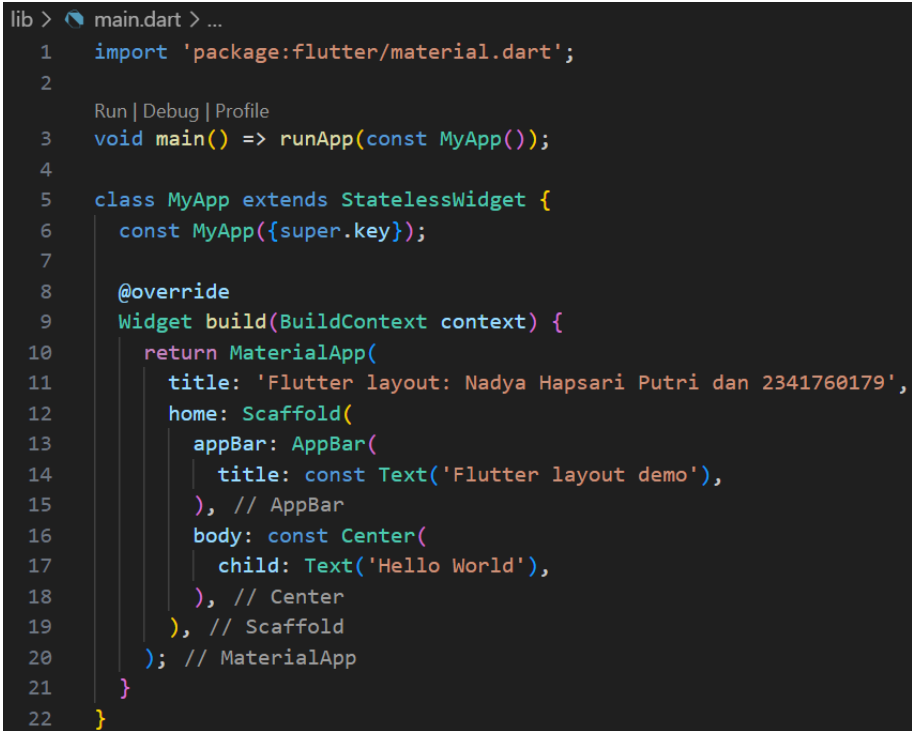
**NIM : 2341760179**

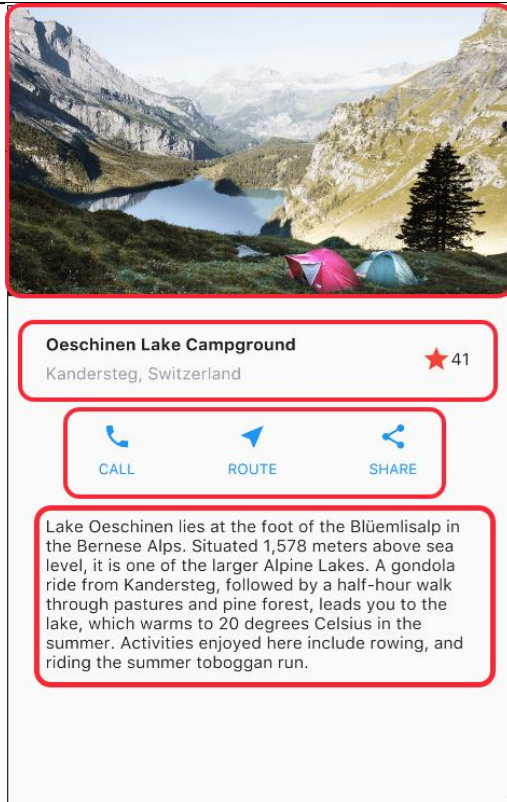
**KELAS : SIB 3D**

**JURUSAN TEKNOLOGI INFORMASI**  
**PROGRAM STUDI D-IV SISTEM INFORMASI BISNIS**  
**POLITEKNIK NEGERI MALANG**

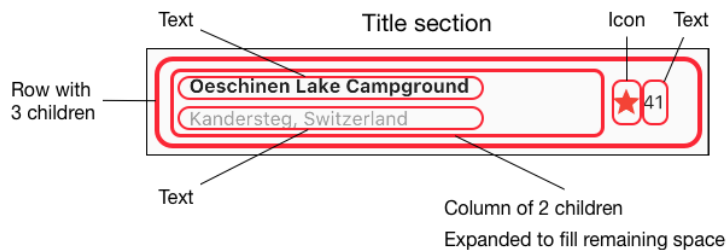
**2025**

## Langkah-Langkah Praktikum

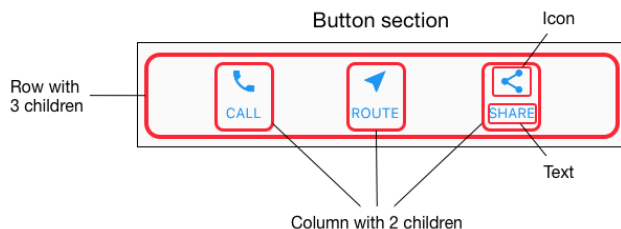
No	Langkah-Langkah
A.	<b>Praktikum 1: Membangun Layout di Flutter</b>
1	<p>Buat Project Baru dengan nama layout_flutter atau sesuaikan dengan laporan praktikum Anda.</p> 
2	<p>Buka file lib/main.dart dan ganti dengan kode awal. Isi nama dan NIM Anda di text title.</p>  <pre>lib &gt; main.dart &gt; ... 1  import 'package:flutter/material.dart'; 2 3  Run   Debug   Profile 4  void main() =&gt; runApp(const MyApp()); 5 6  class MyApp extends StatelessWidget { 7    const MyApp({super.key}); 8 9    @override 10   Widget build(BuildContext context) { 11     return MaterialApp( 12       title: 'Flutter layout: Nadya Hapsari Putri dan 2341760179', 13       home: Scaffold( 14         appBar: AppBar( 15           title: const Text('Flutter layout demo'), 16         ), // AppBar 17         body: const Center( 18           child: Text('Hello World'), 19         ), // Center 20       ), // Scaffold 21     ); // MaterialApp 22   } 23 }</pre>
3	<p>Identifikasi Layout Diagram (memecah layout menjadi elemen dasar seperti baris, kolom, dll.).</p> <p>Langkah pertama adalah memecah tata letak menjadi elemen dasarnya:</p> <ul style="list-style-type: none"><li>• Identifikasi baris dan kolom.</li><li>• Apakah tata letaknya menyertakan kisi-kisi (grid)?</li><li>• Apakah ada elemen yang tumpang tindih?</li><li>• Apakah UI memerlukan tab?</li><li>• Perhatikan area yang memerlukan alignment, padding, atau borders.</li></ul> <p>Pertama, identifikasi elemen yang lebih besar. Dalam contoh ini, empat elemen disusun menjadi sebuah kolom: sebuah gambar, dua baris, dan satu blok teks.</p>



Selanjutnya, buat diagram setiap baris. Baris pertama, disebut bagian Judul, memiliki 3 anak: kolom teks, ikon bintang, dan angka. Anak pertamanya, kolom, berisi 2 baris teks. Kolom pertama itu memakan banyak ruang, sehingga harus dibungkus dengan widget yang Diperluas.



Baris kedua, disebut bagian Tombol, juga memiliki 3 anak: setiap anak merupakan kolom yang berisi ikon dan teks.



Setelah tata letak telah dibuat diagramnya, cara termudah adalah dengan menerapkan pendekatan bottom-up. Untuk meminimalkan kebingungan visual dari kode tata letak yang banyak bertumpuk, tempatkan beberapa implementasi dalam variabel dan fungsi.

4

Implementasi Title Row dengan membuat variabel titleSection yang berisi Container dengan Row.

**/\* soal 1 \*/** Letakkan widget Column di dalam widget Expanded agar menyesuaikan ruang yang tersisa di dalam widget Row. Tambahkan properti crossAxisAlignment ke CrossAxisAlignment.start sehingga posisi kolom berada di awal baris.

**/\* soal 2 \*/** Letakkan baris pertama teks di dalam Container sehingga memungkinkan Anda untuk menambahkan padding = 8. Teks 'Batu, Malang, Indonesia' di dalam Column, set warna menjadi abu-abu.

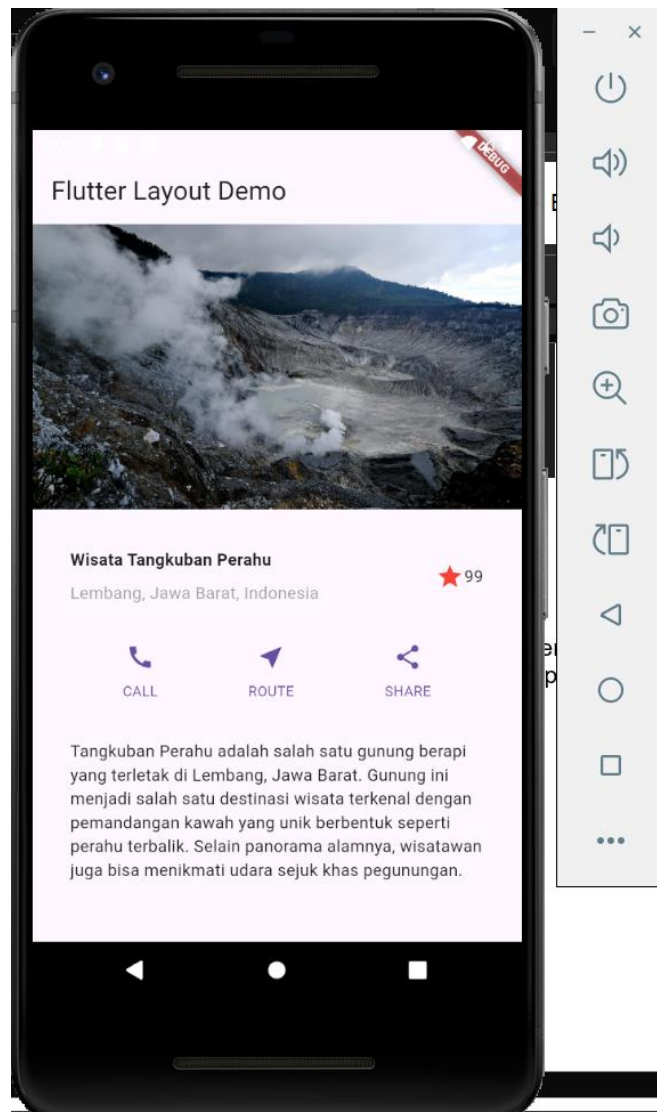
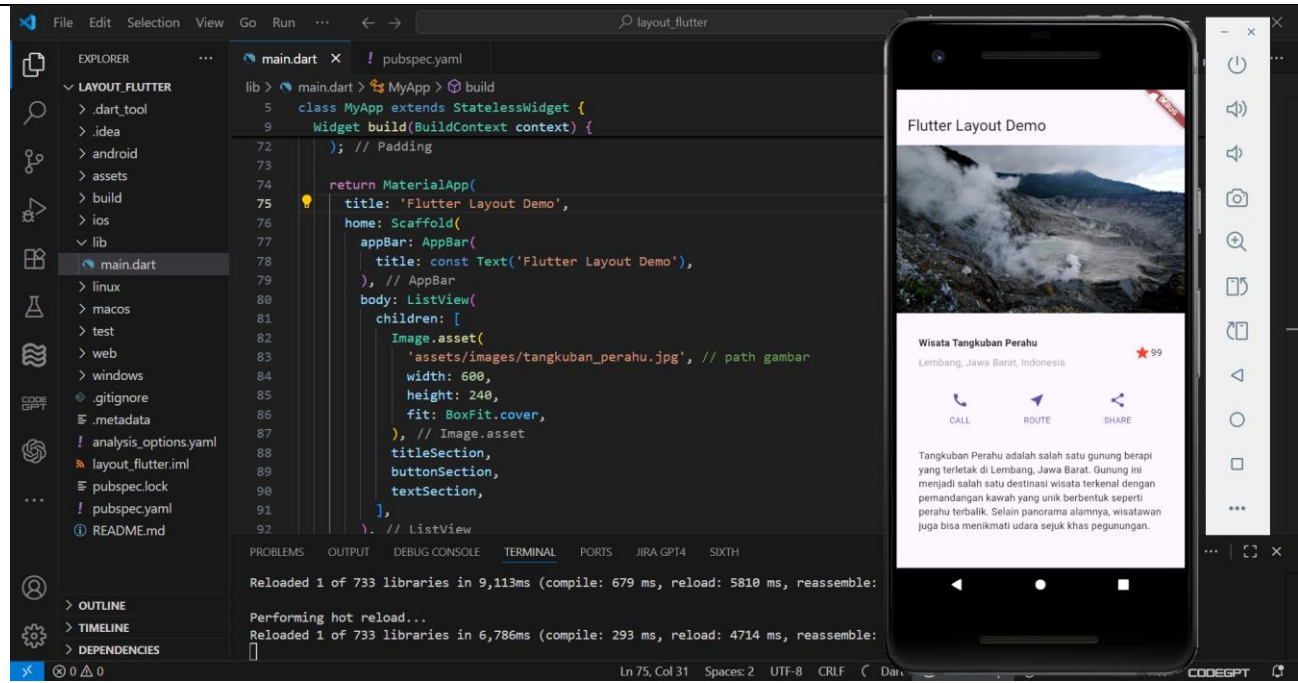
**/\* soal 3 \*/** Dua item terakhir di baris judul adalah ikon bintang, set dengan warna merah, dan teks "41". Seluruh baris ada di dalam Container dan beri padding di sepanjang setiap tepinya sebesar 32 piksel. Kemudian ganti isi body text 'Hello World' dengan variabel titleSection seperti berikut:

```
1  import 'package:flutter/material.dart';
2
3  void main() => runApp(const MyApp());
4
5  class MyApp extends StatelessWidget {
6    const MyApp({super.key});
7
8    @override
9    Widget build(BuildContext context) {
10     Widget titleSection = Container(
11       padding: const EdgeInsets.all(32),
12       child: Row(
13         children: [
14           Expanded(
15             /* soal 1 */
16             child: Column(
17               crossAxisAlignment: CrossAxisAlignment.start,
18               children: [
19                 /* soal 2 */
20                 Container(
21                   padding: const EdgeInsets.only(bottom: 8),
22                   child: const Text(
23                     'Wisata Tangkuban Perahu',
24                     style: TextStyle(
25                       fontWeight: FontWeight.bold,
26                     ),
27                   ),
28                 ),
29                 const Text(
30                   'Lembang, Jawa Barat, Indonesia',
31                   style: TextStyle(
32                     color: Colors.grey,
33                   ),
34                 ),
35               ],
36             ),
37           ),
38           /* soal 3 */
39           const Icon(
40             Icons.star,
41             color: Colors.red,
42           ),
43           const Text('99'),
44         ],
45       ),
46     );
47
```

```

47
48     // Bagian Tombol
49     Color color = Theme.of(context).primaryColor;
50
51     Widget buttonSection = Row(
52       mainAxisAlignment: MainAxisAlignment.spaceEvenly,
53       children: [
54         _buildButtonColumn(color, Icons.call, 'CALL'),
55         _buildButtonColumn(color, Icons.near_me, 'ROUTE'),
56         _buildButtonColumn(color, Icons.share, 'SHARE'),
57       ],
58     );
59
60     // Bagian Deskripsi
61     Widget textSection = const Padding(
62       padding: EdgeInsets.all(32),
63       child: Text(
64         'Tangkuban Perahu adalah salah satu gunung berapi '
65         'yang terletak di Lembang, Jawa Barat. Gunung ini '
66         'menjadi salah satu destinasi wisata terkenal dengan '
67         'pemandangan kawah yang unik berbentuk seperti '
68         'perahu terbalik. Selain panorama alamnya, wisatawan '
69         'juga bisa menikmati udara sejuk khas pegunungan.',
70         softWrap: true,
71       ),
72     );
73
74     return MaterialApp(
75       title: 'Flutter layout demo',
76       home: Scaffold(
77         appBar: AppBar(
78           title: const Text('Flutter layout demo'),
79         ),
80         body: ListView(
81           children: [
82             Image.asset(
83               'assets/images/tangkuban_perahu.jpg', // path gambar
84               width: 600,
85               height: 240,
86               fit: BoxFit.cover,
87             ),
88             titleSection,
89             buttonSection,
90             textSection,
91           ],
92         ),
93       ),
94     );
95   }
96   // Fungsi untuk membuat tombol (CALL, ROUTE, SHARE)
97   Column _buildButtonColumn(Color color, IconData icon, String label)
98   {
99     return Column(
100       mainAxisAlignment: MainAxisAlignment.min,
101       mainAxisAlignment: MainAxisAlignment.center,
102       children: [
103         Icon(icon, color: color),
104         Container(
105           margin: const EdgeInsets.only(top: 8),
106           child: Text(
107             label,
108             style: TextStyle(
109               fontSize: 12,
110               fontWeight: FontWeight.w400,
111               color: color,
112             ),
113           ),
114         ],
115       );
116   }
117 }

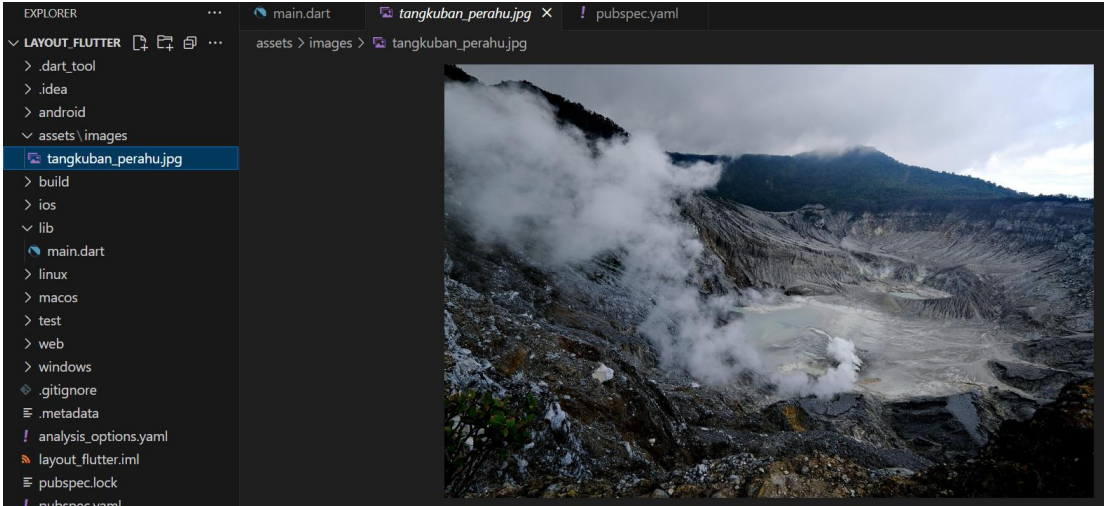
```



## B. Praktikum 2: Implementasi Button Row

1	<p>Buat method <code>buildButtonColumn(Color color, IconData icon, String label)</code> yang mengembalikan widget <code>Column</code> (berisi <code>Icon</code> dan <code>Text</code>).</p> <pre> 97   Column _buildButtonColumn(Color color, IconData icon, String label) { 98     return Column( 99       mainAxisAlignment: MainAxisAlignment.min, 100      mainAxisAlignment: MainAxisAlignment.center, 101      children: [ 102        Icon(icon, color: color), 103        Container( 104          margin: const EdgeInsets.only(top: 8), 105          child: Text( 106            label, 107            style: TextStyle( 108              fontSize: 12, 109              fontWeight: FontWeight.w400, 110              color: color, 111            ), // TextStyle 112          ), // Text 113        ), // Container 114      ], 115    ); // Column 116  } 117 }</pre>
2	<p>Buat widget <code>buttonSection</code> yang merupakan <code>Row</code>. Set <code>MainAxisAlignment.spaceEvenly</code>. Panggil <code>buildButtonColumn()</code> tiga kali untuk membuat kolom tombol 'CALL', 'ROUTE', dan 'SHARE'.</p> <pre> 48   // Bagian Tombol 49   Color color = Theme.of(context).primaryColor; 50 51   Widget buttonSection = Row( 52     mainAxisAlignment: MainAxisAlignment.spaceEvenly, 53     children: [ 54       _buildButtonColumn(color, Icons.call, 'CALL'), 55       _buildButtonColumn(color, Icons.near_me, 'ROUTE'), 56       _buildButtonColumn(color, Icons.share, 'SHARE'), 57     ], 58   );</pre>
3	<p>Tambah <code>buttonSection</code> ke <code>body</code> di dalam <code>Column</code> (di bawah <code>titleSection</code>).</p> <pre> 80   body: ListView( 81     children: [ 82       Image.asset( 83         'assets/images/tangkuban_perahu.jpg', // path gambar 84         width: 600, 85         height: 240, 86         fit: BoxFit.cover, 87       ), // Image.asset 88       titleSection, 89       buttonSection, 90       textSection,</pre>
C.	<p><b>Praktikum 3: Implementasi Text Section</b></p>



1	<p>Buat widget textSection yang merupakan Container. Masukkan teks ke dalamnya (termasuk nama dan NIM) dengan padding 32 dan properti softWrap: true.</p> <pre> 60      // Bagian Deskripsi 61      Widget textSection = Container( 62        padding: const EdgeInsets.all(32), 63        child: const Text( 64          'Tangkuban Perahu adalah salah satu gunung berapi ' 65          'yang terletak di Lembang, Jawa Barat. Gunung ini ' 66          'menjadi salah satu destinasi wisata terkenal dengan ' 67          'pemandangan kawah yang unik berbentuk seperti ' 68          'perahu terbalik. Selain panorama alamnya, wisatawan ' 69          'juga bisa menikmati udara sejuk khas pegunungan.', 70          softWrap: true, 71        ), // Text 72      ); // Container </pre>
2	<p>Tambahkan variabel textSection ke body di dalam Column (di bawah buttonSection).</p> <pre> 80      body: ListView( 81        children: [ 82          Image.asset( 83            'assets/images/tangkuban_perahu.jpg', // path gambar 84            width: 600, 85            height: 240, 86            fit: BoxFit.cover, 87          ), // Image.asset 88          titleSection, 89          buttonSection, 90          textSection, </pre>
D.	<p><b>Praktikum 4: Implementasi Image Section</b></p>
1	<p>Siapkan aset gambar: Cari gambar, buat folder images di root project, masukkan file gambar (lake.jpg contoh), lalu set aset di file pubspec.yaml.</p> 



	<pre>! pubspec.yaml 51 52 # The following section is specific to Flutter packages. 53 flutter: 54   assets: 55     - assets/images/tangkuban_perahu.jpg 56</pre>	
2	<p>Tambahkan gambar ke body dengan menggunakan Image.asset('images/lake.jpg') di bagian atas children dari Column (di atas titleSection). Set width: 600, height: 240, dan fit: BoxFit.cover.</p> <pre>80 body: ListView( 81   children: [ 82     Image.asset( 83       'assets/images/tangkuban_perahu.jpg', // path gambar 84       width: 600, 85       height: 240, 86       fit: BoxFit.cover, 87     ), // Image.asset 88     titleSection, 89     buttonSection, 90     textSection,</pre>	
3	<p>Ubah Column menjadi ListView agar mendukung scroll pada perangkat beresolusi kecil.</p> <pre>80 body: ListView( 81   children: [ 82     Image.asset( 83       'assets/images/tangkuban_perahu.jpg', // path gambar 84       width: 600, 85       height: 240, 86       fit: BoxFit.cover, 87     ), // Image.asset 88     titleSection, 89     buttonSection, 90     textSection,</pre>	
<b>E.</b>	<b>Tugas Praktikum 1</b>	
1	<p>Selesaikan Praktikum 1 sampai 4, lalu dokumentasikan dan push ke repository Anda berupa screenshot hasil pekerjaan beserta penjelasannya di file README.md.</p>	

# layout\_flutter - Jobsheet Layout dan Navigasi

A new Flutter project.

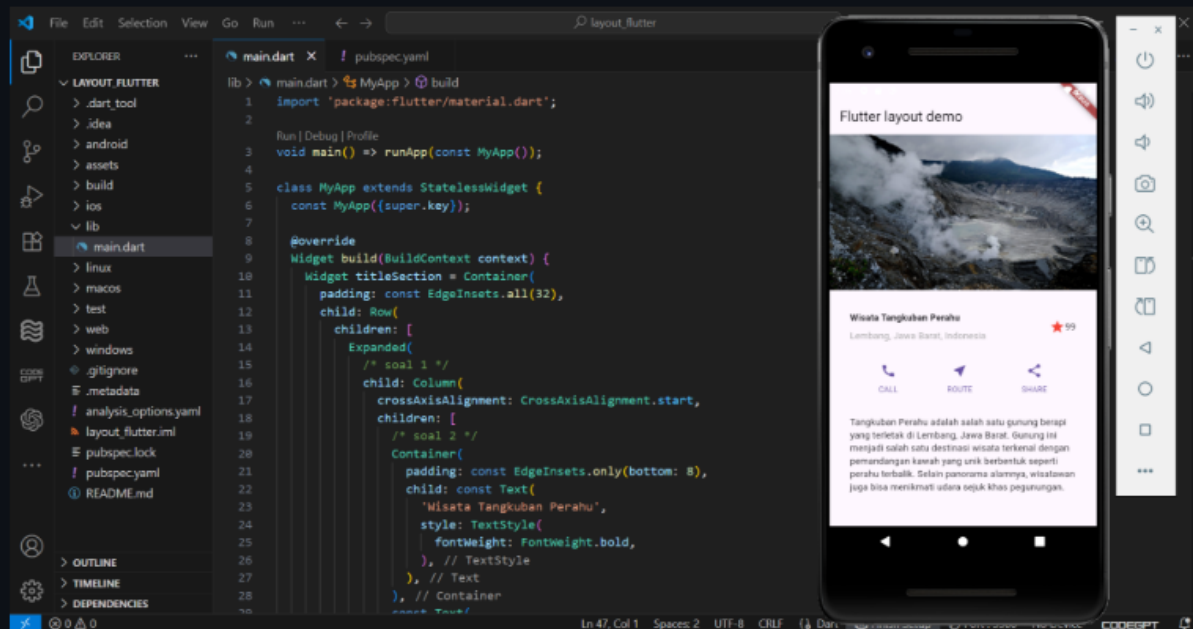
Nama: [Nadya Hapsari Putri]

NIM: [2341760179]

Kelas: [SIB 3D]

## Hasil Akhir Praktikum 1-4

Berikut adalah hasil akhir dari Praktikum 1-4 yang telah saya kerjakan.



### Penjelasan Singkat:

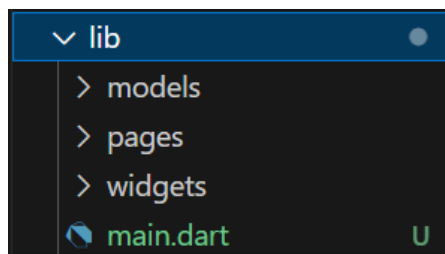
- Praktikum 1: Membuat layout dasar (Image, Title Section, Button Section, Text Section).
- Praktikum 2: Menambahkan widget tambahan sesuai instruksi.
- Praktikum 3: Menyusun tata letak lebih kompleks sesuai jobsheet.
- Praktikum 4: Implementasi navigasi dasar.
- Hasil akhirnya adalah aplikasi Flutter dengan layout lengkap dan navigasi sesuai arahan jobsheet.

3 Kumpulkan link commit repository GitHub kepada dosen yang telah disepakati.

<https://github.com/hapsari putri14/pemograman-mobile.git>

## F. Praktikum 5: Membangun Navigasi di Flutter

1 Siapkan project baru dengan nama **belanja** dan susunan folder: lib (models, pages, widgets, main.dart).



2	<p>Mendefinisikan Route: Buat dua file dart <code>home_page.dart</code> dan <code>item_page.dart</code> di folder <code>pages</code>. Deklarasikan class <code>HomePage</code> dan <code>ItemPage</code> yang diturunkan dari <code>StatelessWidget</code>.</p> <div style="display: flex; justify-content: space-between;"> <pre style="background-color: #2e3436; color: #eeeeec; padding: 10px; width: 48%;"> lib &gt; pages &gt; home_page.dart &gt; ... 1 import 'package:flutter/material.dart'; 2 3 class HomePage extends StatelessWidget { 4   const HomePage({super.key}); 5 6   @override 7   Widget build(BuildContext context) { 8     return Scaffold( 9       appBar: AppBar( 10        title: const Text("Home Page"), 11      ), // AppBar 12      body: const Center( 13        child: Text("Ini adalah halaman Home"), 14      ), // Center 15    ); // Scaffold 16  } 17 }</pre> <pre style="background-color: #2e3436; color: #eeeeec; padding: 10px; width: 48%;"> lib &gt; pages &gt; item_page.dart &gt; ... 1 import 'package:flutter/material.dart'; 2 3 class ItemPage extends StatelessWidget { 4   const ItemPage({super.key}); 5 6   @override 7   Widget build(BuildContext context) { 8     return Scaffold( 9       appBar: AppBar( 10        title: const Text("Item Page"), 11      ), // AppBar 12      body: const Center( 13        child: Text("Ini adalah halaman Item"), 14      ), // Center 15    ); // Scaffold 16  } 17 }</pre> </div>
3	<p>Lengkapi kode di <code>main.dart</code>: Definisikan route unik. <code>HomePage</code> sebagai <code>/</code> dan <code>ItemPage</code> sebagai <code>/item</code>. Gunakan <code>initialRoute: '/'</code> dan set routes di <code>MaterialApp</code>.</p> <pre style="background-color: #2e3436; color: #eeeeec; padding: 10px;"> lib &gt; main.dart &gt; ... 1 import 'package:flutter/material.dart'; 2 import 'pages/home_page.dart'; 3 import 'pages/item_page.dart'; 4 5 void main() { 6   runApp( 7     MaterialApp( 8       debugShowCheckedModeBanner: false, 9       initialRoute: '/', // Halaman pertama 10      routes: { 11        '/': (context) =&gt; HomePage(), 12        '/item': (context) =&gt; ItemPage(), 13      }, 14    ), // MaterialApp 15  ); 16 }</pre>
4	<p>Membuat data model: Buat file <code>item.dart</code> di folder <code>models</code>. Definisikan class <code>Item</code> dengan properti <code>String name</code> dan <code>int price</code>.</p> <pre style="background-color: #2e3436; color: #eeeeec; padding: 10px;"> lib &gt; models &gt; item.dart &gt; ... 1 class Item { 2   String name; 3   int price; 4 5   // Constructor dengan named parameter 6   Item({required this.name, required this.price}); 7 }</pre>

- 5 Lengkapi kode di class HomePage: Definiskan final List<Item> items yang berisi data objek Item (contoh: 'Sugar': 5000, 'Salt': 2000).

```
lib > pages > home_page.dart > HomePage > build
1  import 'package:flutter/material.dart';
2  import '../models/item.dart';
3
4  class HomePage extends StatelessWidget {
5    HomePage({super.key});
6
7    // Tambahan dari langkah 5: definisi list items
8    final List<Item> items = [
9      Item(name: 'Sugar', price: 5000),
10     Item(name: 'Salt', price: 2000),
11   ];
12
13   @override
14   Widget build(BuildContext context) {
15     return Scaffold(
16       appBar: AppBar(
17         title: const Text("Home Page"),
```

- 6 Membuat ListView dan itemBuilder: Gunakan ListView.builder di body HomePage untuk menampilkan data dari items menggunakan widget Card untuk batas datanya.

```
21   body: Container(
22     margin: const EdgeInsets.all(8),
23     child: ListView.builder(
24       itemCount: items.length,
25       itemBuilder: (context, index) {
26         final item = items[index];
27         return InkWell(
28           onTap: () {
29             // Navigasi ke ItemPage + kirim data item
30             Navigator.pushNamed(context, '/item', arguments: item,);
31           },
32           child: Card(
33             margin: const EdgeInsets.symmetric(vertical: 6, horizontal: 4),
34             child: Container(
35               padding: const EdgeInsets.all(12),
36               child: Row(
37                 mainAxisAlignment: MainAxisAlignment.spaceBetween,
38                 children: [
39                   Text(item.name),
40                   Text(item.price.toString()),
41                 ],
42               ), // ROW
43             ), // Container
44           ), // Card
45         ); // InkWell
46       },
47     ), // ListView.builder
48   ), // Container
49 ); // Scaffold
50 }
51 }
```

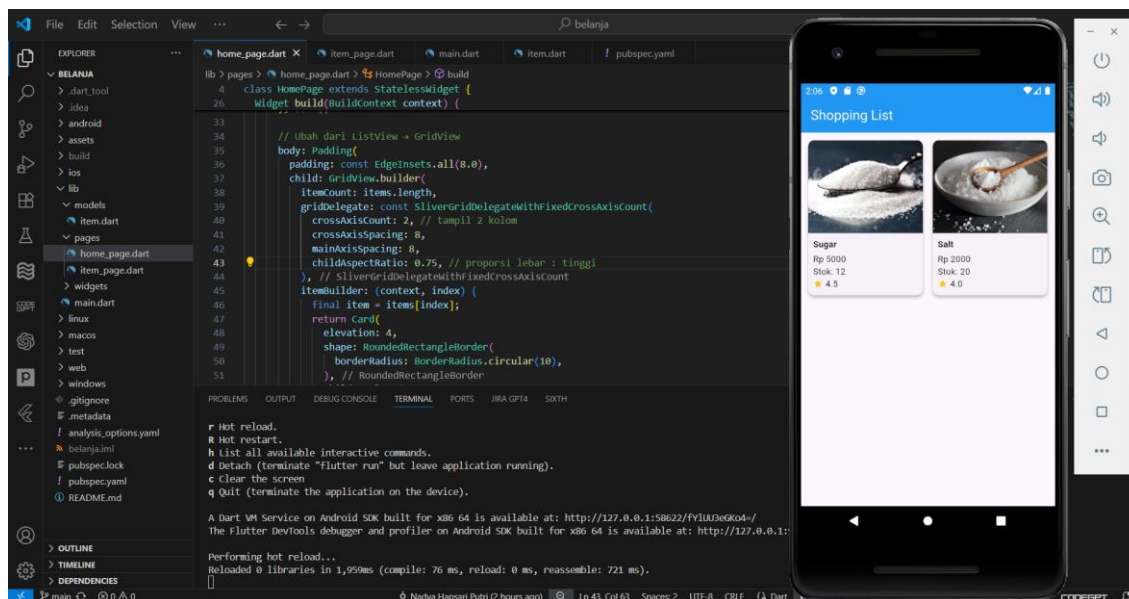
7	<p>Menambahkan aksi pada ListView: Bungkus widget Card dengan widget InkWell. Tambahkan onTap untuk berpindah ke halaman ItemPage menggunakan Navigator.pushNamed(context, '/item').</p> <pre> 27         return InkWell( 28             onTap: () { 29                 // Navigasi ke ItemPage + kirim data item 30                 Navigator.pushNamed(context, '/item', arguments: item,); 31             }, </pre>
G.	<p><b>Tugas Praktikum 2</b></p>
1	<p>Pengiriman data ke halaman berikutnya: Perbarui kode Navigator.pushNamed dengan menambahkan informasi arguments: item (item adalah objek dari data yang diklik).</p> <pre> 27         return InkWell( 28             onTap: () { 29                 // Navigasi ke ItemPage + kirim data item 30                 Navigator.pushNamed(context, '/item', arguments: item,); 31             }, </pre>
2	<p>Nilai dari halaman sebelumnya dapat dibaca dengan ModalRoute di fungsi build pada ItemPage, lalu digunakan seperti variabel biasa.</p> <pre> 7         @override 8         Widget build(BuildContext context) { 9             // Ambil data dari arguments 10            final Item item = ModalRoute.of(context)!.settings.arguments as Item; </pre>
3	<p>Tambahkan atribut foto produk, stok, dan rating pada hasil akhir aplikasi belanja. Ubah tampilan menjadi GridView seperti di aplikasi marketplace pada umumnya.</p> <div style="display: flex; justify-content: space-between;"> <pre> lib &gt; models &gt; item.dart &gt; ... 1 class Item { 2     String name; 3     int price; 4     String image; 5     int stock; 6     double rating; 7 8     // Constructor dengan named parameter 9     Item({ 10         required this.name, 11         required this.price, 12         required this.image, 13         required this.stock, 14         required this.rating, 15     }); 16 } </pre> <pre> ! pubspec.yaml 52 # The following section is specific to Flutter 53 flutter: 54     assets: 55         - assets/images/sugar.jpg 56         - assets/images/salt.jpg 57 </pre> <pre> lib &gt; pages &gt; home_page.dart &gt; HomePage &gt; build 1 import 'package:flutter/material.dart'; 2 import '../models/item.dart'; 3 4 class HomePage extends StatelessWidget { 5     HomePage({super.key}); 6 7     // List data Item 8     final List&lt;Item&gt; items = [ 9         Item( 10             name: 'Sugar', 11             price: 5000, 12             image: 'assets/images/sugar.jpg', 13             stock: 12, 14             rating: 4.5, 15         ), 16         Item( 17             name: 'Salt', 18             price: 2000, 19             image: 'assets/images/salt.jpg', 20             stock: 20, 21             rating: 4.0, 22         ), 23     ]; </pre> </div>

```

lib > pages > home_page.dart > ...
4   class HomePage extends StatelessWidget {
26   Widget build(BuildContext context) {

34   // Ubah dari ListView → GridView
35   body: Padding(
36     padding: const EdgeInsets.all(8.0),
37     child: GridView.builder(
38       itemCount: items.length,
39       gridDelegate: const SliverGridDelegateWithFixedCrossAxisCount(
40         crossAxisCount: 2, // tampil 2 kolom
41         crossAxisSpacing: 8,
42         mainAxisSpacing: 8,
43         childAspectRatio: 0.75, // proporsi lebar : tinggi
44       ), // SliverGridDelegateWithFixedCrossAxisCount
45       itemBuilder: (context, index) {
46         final item = items[index];
47         return Card(
48           elevation: 4,
49           shape: RoundedRectangleBorder(
50             borderRadius: BorderRadius.circular(10),
51           ), // RoundedRectangleBorder
52           child: Column(
53             crossAxisAlignment: CrossAxisAlignment.start,
54             children: [
55               // Gambar produk
56               Expanded(
57                 child: ClipRRect(
58                   borderRadius: const BorderRadius.vertical(top: Radius.circular(10)),
59                   child: Image.asset(
60                     item.image,
61                     fit: BoxFit.cover,
62                     width: double.infinity,
63                   ), // Image.asset
64                 ), // ClipRRect
65               ), // Expanded
66               // Informasi produk
67               Padding(
68                 padding: const EdgeInsets.all(8.0),
69                 child: Column(
70                   crossAxisAlignment: CrossAxisAlignment.start,
71                   children: [
72                     Text(item.name,
73                       style: const TextStyle(fontWeight: FontWeight.bold)), // Text
74                     const SizedBox(height: 4),

```

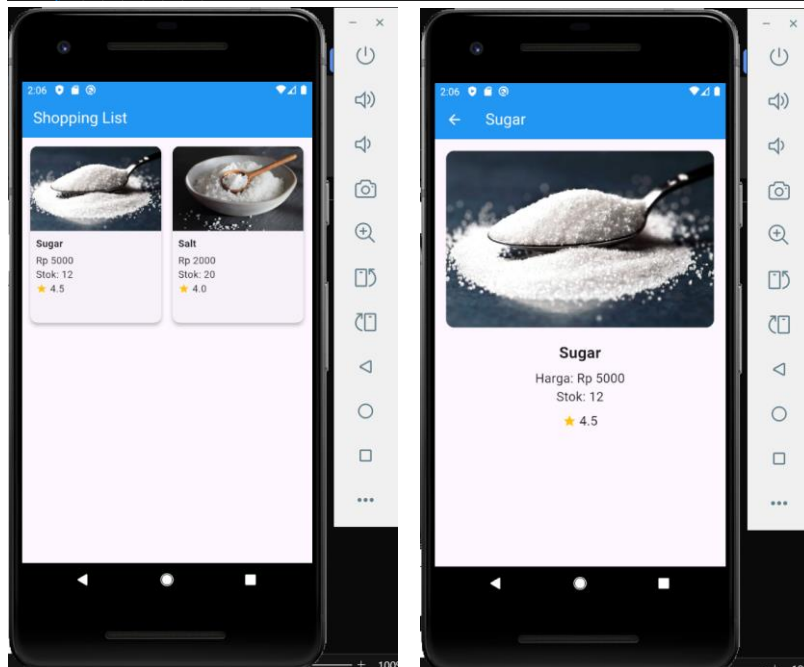




#### 4 Implementasikan Hero widget pada aplikasi belanja Anda.

```
lib > pages > home_page.dart > HomePage > build
4 class HomePage extends StatelessWidget {
26   widget build(BuildContext context) {
56     child: Card(
57       elevation: 4,
58       shape: RoundedRectangleBorder(
59         borderRadius: BorderRadius.circular(10),
60       ), // RoundedRectangleBorder
61       child: Column(
62         crossAxisAlignment: CrossAxisAlignment.start,
63         children: [
64           // Tambah Hero di sekitar gambar
65           Hero(
66             tag: item.name, // tag unik agar animasi tersambung
67             child: ClipRRect(
68               borderRadius: const BorderRadius.vertical(
69                 top: Radius.circular(10),
70               ), // BorderRadius.vertical
71               child: Image.asset(
72                 item.image,
73                 fit: BoxFit.cover,
74                 width: double.infinity,
75                 height: 120,
76               ), // Image.asset
77             ), // ClipRRect
78           ), // Hero
```

```
lib > pages > item_page.dart > ItemPage > build
4 class ItemPage extends StatelessWidget {
8   widget build(BuildContext context) {
21     child: Column(
22       crossAxisAlignment: CrossAxisAlignment.center,
23       children: [
24         // Hero widget agar gambar berpindah dengan animasi dari HomePage
25         Hero(
26           tag: item.name, // tag harus sama seperti di HomePage
27           child: ClipRRect(
28             borderRadius: BorderRadius.circular(12),
29             child: Image.asset(
30               item.image,
31               fit: BoxFit.cover,
32               width: double.infinity,
33               height: 250,
34             ), // Image.asset
35           ), // ClipRRect
36         ), // Hero
```



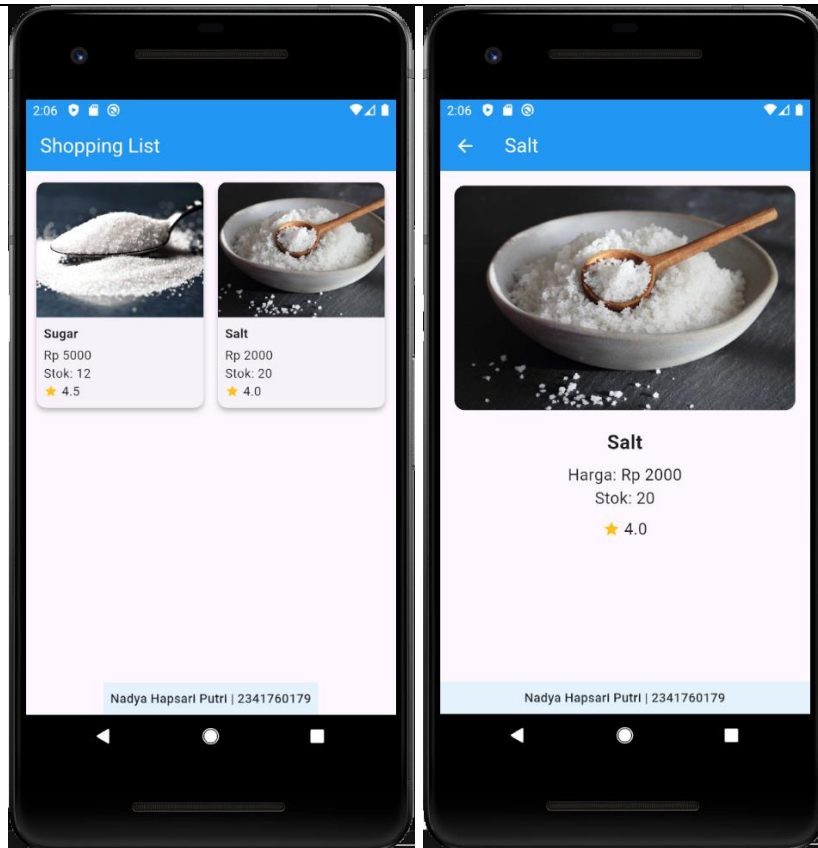
- 5 Sesuaikan dan modifikasi tampilan sehingga menjadi aplikasi menarik. Pecah widget menjadi kode yang lebih kecil. Tambahkan Nama dan NIM di footer aplikasi belanja Anda.

Membuat file baru lib/widgets/item\_card.dart

```
1 import 'package:flutter/material.dart';
2 import '../models/item.dart';
3
4 class ItemCard extends StatelessWidget {
5   final Item item;
6   final VoidCallback onTap;
7
8   const ItemCard({super.key, required this.item, required this.onTap});
9
10  @override
11  Widget build(BuildContext context) {
12    return InkWell(
13      onTap: onTap,
14      child: Card(
15        elevation: 4,
16        shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(10)),
17        child: Column(
18          crossAxisAlignment: CrossAxisAlignment.start,
19          children: [
20            Expanded(
21              child: ClipRRect(
22                borderRadius: const BorderRadius.vertical(top: Radius.circular(10)),
23                child: Hero(
24                  tag: item.name,
25                  child: Image.asset(
26                    item.image,
27                    fit: BoxFit.cover,
28                    width: double.infinity,
29                  ),
30                ),
31            ),
32          ],
33        ),
34        Padding(
35          padding: const EdgeInsets.all(8),
36          child: Column(
37            crossAxisAlignment: CrossAxisAlignment.start,
38            children: [
39              Text(item.name, style: const TextStyle(fontWeight: FontWeight.bold)),
40              const SizedBox(height: 4),
41              Text("Rp ${item.price}"),
42              Text("Stok: ${item.stock}"),
43              Row(
44                children: [
45                  const Icon(Icons.star, color: Colors.amber, size: 16),
46                  const SizedBox(width: 4),
47                  Text(item.rating.toString()),
48                ],
49              ),
50            ],
51          ),
52        ),
53      ),
54    );
55  }
56 }
57 }
```

```
lib > pages > item_page.dart > ...
4 class ItemPage extends StatelessWidget {
8   Widget build(BuildContext context) {
10
11     // Menambahkan footer Nama & NIM
12     bottomNavigationBar: Container(
13       color: Colors.blue.shade50,
14       padding: const EdgeInsets.all(8),
15       child: const Text(
16         'Nadya Hapsari Putri | 2341760179',
17         textAlign: TextAlign.center,
18         style: TextStyle(
19           fontSize: 14,
20           fontWeight: FontWeight.w500,
21           color: Colors.black87,
22         ), // TextStyle
23       ), // Text
24     ), // Container
25   ); // Scaffold
26 }
27 }
```

```
lib > pages > home_page.dart > HomePage > build
5 class HomePage extends StatelessWidget {
26   Widget build(BuildContext context) {
28
29     // Footer dengan nama dan NIM
30     Container(
31       padding: const EdgeInsets.all(8),
32       color: Colors.blue.shade50,
33       child: const Text(
34         'Nadya Hapsari Putri | 2341760179',
35         style: TextStyle(fontSize: 14, fontWeight: FontWeight.w500),
36       ), // Text
37     ), // Container
38   ), // Column
39 ); // Scaffold
40 }
41 }
```



6 Selesaikan Praktikum 5: Navigasi dan Rute tersebut. Cobalah modifikasi menggunakan plugin `go_router`.

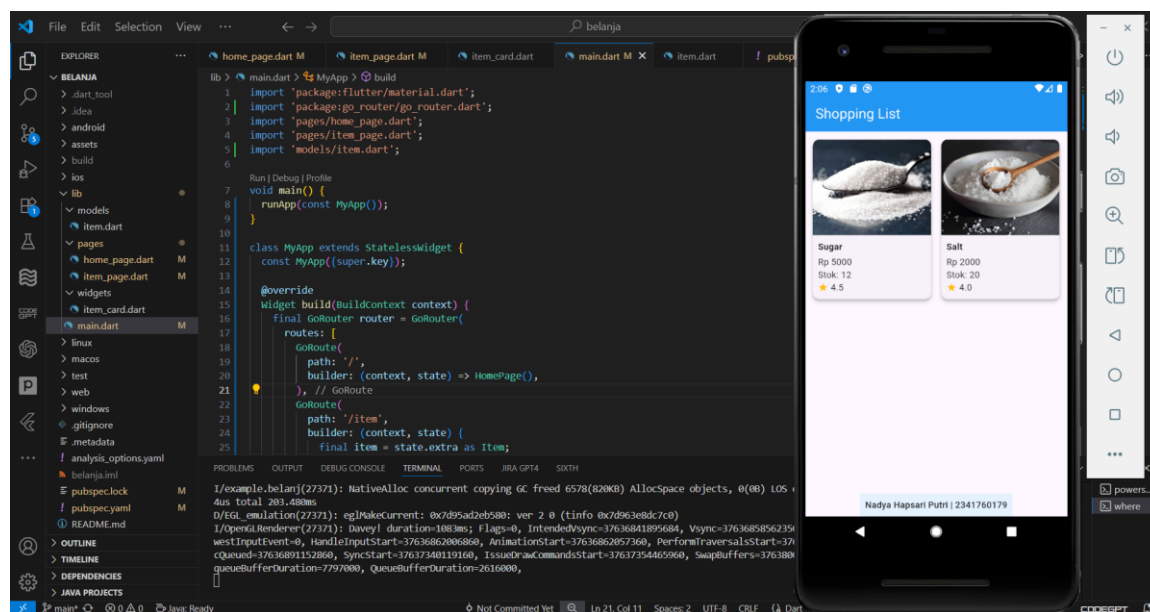
```
! pubspec.yaml
26 # consider running `flutter pub upgrade --major-versions`. Alternat
27 # dependencies can be manually updated by changing the version numb
28 # the latest version available on pub.dev. To see which dependencie
29 # versions available, run `flutter pub outdated`.
30 dependencies:
31   flutter:
32     sdk: flutter
33   go_router: ^14.0.1
34
35 # The following adds the Cupertino Icons font to your application

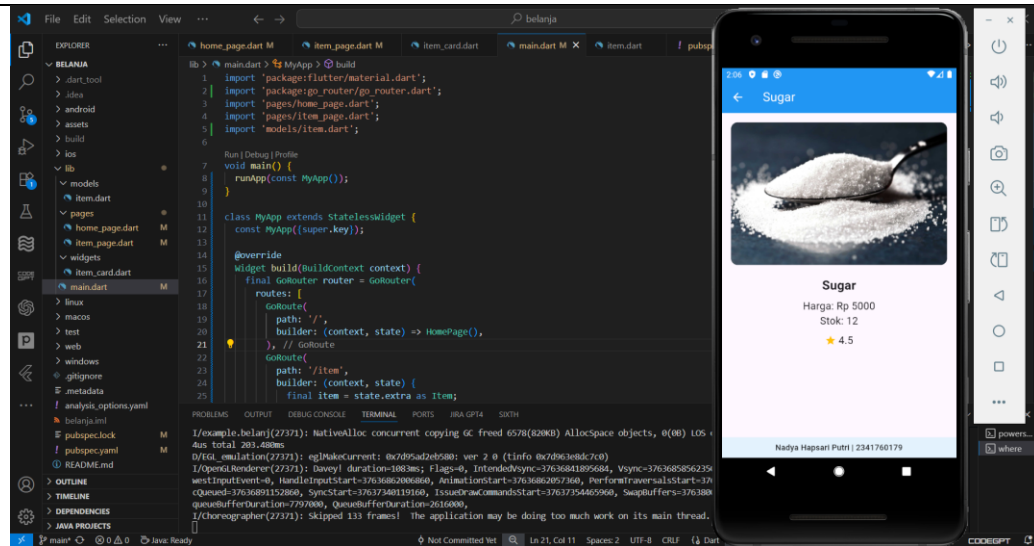
[belanja] flutter pub get --no-example
Resolving dependencies...
Downloading packages...
  characters 1.4.0 (1.4.1 available)
  flutter_lints 5.0.0 (6.0.0 available)
+ flutter_web_plugins 0.0.0 from sdk flutter
+ go_router 14.8.1 (16.2.4 available)
  lints 5.1.1 (6.0.0 available)
+ logging 1.3.0
  material_color_utilities 0.11.1 (0.13.0 available)
  meta 1.16.0 (1.17.0 available)
  test_api 0.7.6 (0.7.7 available)
Changed 3 dependencies!
7 packages have newer versions incompatible with dependency constraints.
Try `flutter pub outdated` for more information.
exit code 0
```

```
lib > pages > home_page.dart > HomePage > items
6 class HomePage extends StatelessWidget {
27   Widget build(BuildContext context) {
46     // SliverGridDelegateWithFixedCrossAxisCount
47     itemBuilder: (context, index) {
48       final item = items[index];
49       return ItemCard(
50         item: item,
51         onTap: () {
52           context.push('/item', extra: item);
53         },
54       ); // ItemCard
55     }, // GridView.builder
56   ), // Padding
57 ), // Expanded
58 }
```

```
lib > pages > item_page.dart > ItemPage > build
1 import 'package:flutter/material.dart';
2 import '../models/item.dart';
3
4 class ItemPage extends StatelessWidget {
5   final Item item;
6
7   const ItemPage({super.key, required this.item});
```

```
lib > main.dart > MyApp
1 import 'package:flutter/material.dart';
2 import 'package:go_router/go_router.dart';
3 import 'pages/home_page.dart';
4 import 'pages/item_page.dart';
5 import 'models/item.dart';
6
7 Run | Debug | Profile
8 void main() {
9   runApp(const MyApp());
10 }
11
12 class MyApp extends StatelessWidget {
13   const MyApp({super.key});
14
15   @override
16   Widget build(BuildContext context) {
17     final GoRouter router = GoRouter(
18       routes: [
19         GoRoute(
20           path: '/',
21           builder: (context, state) => HomePage(),
22         ), // GoRoute
23         GoRoute(
24           path: '/item',
25           builder: (context, state) {
26             final item = state.extra as Item;
27             return ItemPage(item: item);
28           }, // GoRoute
29       ], // GoRouter
30     ); // GoRouter
31
32     return MaterialApp.router(
33       debugShowCheckedModeBanner: false,
34       routerConfig: router,
35     );
36   }
37 }
```





Dokumentasikan dan push ke repository Anda berupa screenshot setiap hasil pekerjaan beserta penjelasannya di file README.md. Kumpulkan link commit repository GitHub kepada dosen yang telah disepakati.

## belanja - Jobsheet Week 6: Layout dan Navigasi

A new Flutter project.

Nama: Nadya Hapsari Putri

NIM: 2341760179

Kelas: SIB 3D

### Praktikum 5

#### Penjelasan Singkat

- Membuat dua halaman, yaitu `HomePage` dan `ItemPage`.
- Menambahkan navigasi menggunakan `Navigator.pushNamed` dan `ModalRoute` untuk mengirim data antar halaman.
- Menampilkan daftar produk menggunakan `GridView` dan `Card`.
- Setiap item bisa ditekan dan menampilkan detail produk di halaman berikutnya.

### Hasil Akhir Praktikum 5

#### Halaman Home Page

