

Praktikum 5
AntarMuka Pengguna 2
Pemrograman Perangkat Bergerak



Disusun Oleh :

Dwi Candra Pratama/2211104035
SE-06-02

Asisten Praktikum :
Muhammad Faza Zulian Gesit Al Barru
Aisyah Hasna Aulia

Dosen Pengampu :
Yudha Islami Sulistya, S.Kom., M.Cs.

PROGRAM STUDI S1 SOFTWARE ENGINEERING
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO 2024

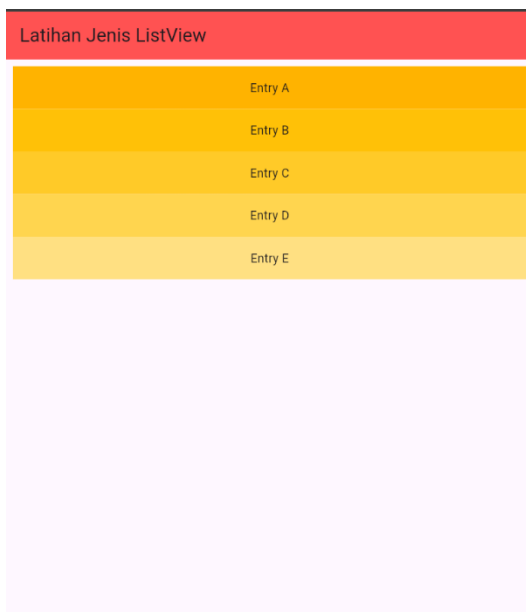
GUIDED

A. Tujuan Praktikum

- Mahasiswa mampu memahami konsep layout pada Flutter
- Mahasiswa dapat mengimplementasikan desain user interface pada Flutter

1. ListView.builder

Widget ListView jenis ini cocok digunakan ketika memiliki data list yang lebih besar. ListView.builder membutuhkan itemBuilder dan itemCount. Parameter itemBuilder merupakan fungsi yang mengembalikan widget untuk ditampilkan. Sedangkan itemCount kita isi dengan jumlah seluruh item yang ingin ditampilkan.



```
jenis_list_view.dart > JenisListView > JenisListView
import 'package:flutter/material.dart';

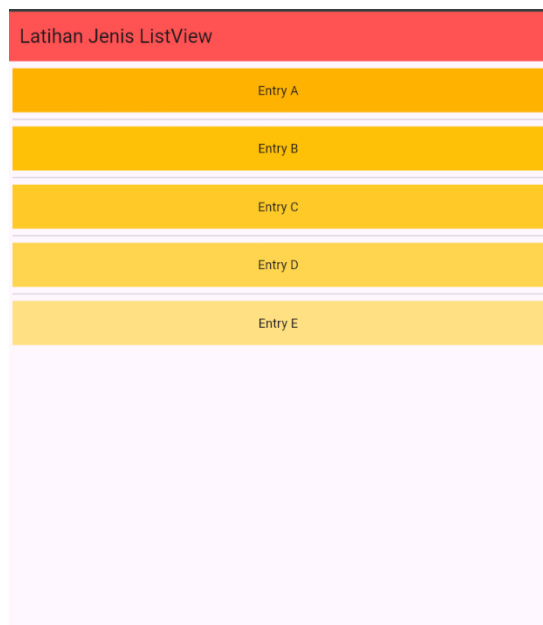
class JenisListView extends StatelessWidget {
  const JenisListView({super.key});

  @override
  Widget build(BuildContext context) {
    final List<String> entries = <String>['A', 'B', 'C', 'D', 'E'];
    final List<int> colorCodes = <int>[600, 500, 400, 300, 200];

    return Scaffold(
      appBar: AppBar(
        title: const Text("Latihan Jenis ListView"),
        backgroundColor: Colors.redAccent,
      ), // AppBar
      body: ListView.separated(
        padding: const EdgeInsets.all(8),
        itemCount: entries.length,
        itemBuilder: (BuildContext context, int index) {
          return Container(
            height: 50,
            color: Colors.amber[colorCodes[index]],
            child: Center(
              child: Text('Entry ${entries[index]}'),
            ), // Center
          ); // Container
        },
        separatorBuilder: (BuildContext context, int index) => const Divider(),
      ), // ListView.separated
    ); // Scaffold
  }
}
```

2. ListView.Separated

ListView jenis ini akan menampilkan daftar item yang dipisahkan dengan separator. Penggunaan ListView.separated mirip dengan builder, yang membedakan adalah terdapat satu parameter tambahan wajib yaitu separatorBuilder yang mengembalikan Widget yang akan berperan sebagai separator.



```
jenis_list_view.dart > JenisListView > JenisListView
import 'package:flutter/material.dart';

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

  @override
  Widget build(BuildContext context) {
    final List<String> entries = <String>['A', 'B', 'C', 'D', 'E'];
    final List<int> colorCodes = <int>[600, 500, 400, 300, 200];

    return Scaffold(
      appBar: AppBar(
        title: const Text("Latihan Jenis ListView"),
        backgroundColor: Colors.redAccent,
      ), // AppBar
      body: ListView.separated(
        padding: const EdgeInsets.all(8),
        itemCount: entries.length,
        itemBuilder: (BuildContext context, int index) {
          return Container(
            height: 50,
            color: Colors.amber[colorCodes[index]],
            child: Center(
              child: Text('Entry ${entries[index]}'),
            ), // Center
          ); // Container
        },
        separatorBuilder: (BuildContext context, int index) => const Divider(),
      ), // ListView.separated
    ); // Scaffold
  }
}
```

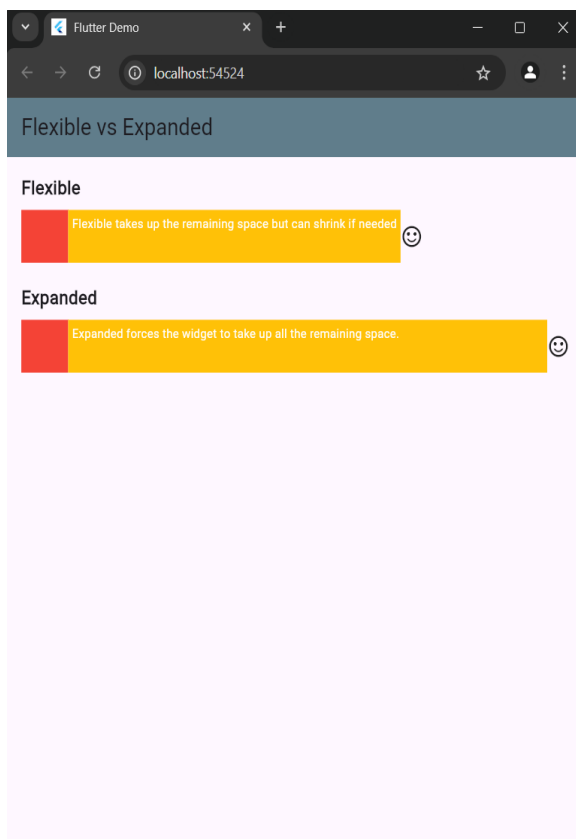
3. Flexible dan Expanded

a) Flexible

Flexible digunakan ketika Anda ingin memberikan ruang fleksibel kepada widget di dalam kolom atau baris. Dengan Flexible, widget dapat mengambil ruang yang tersisa di dalam layout, tetapi tetap memiliki batas maksimal yang disesuaikan dengan kebutuhan ruangnya.

b) Expanded

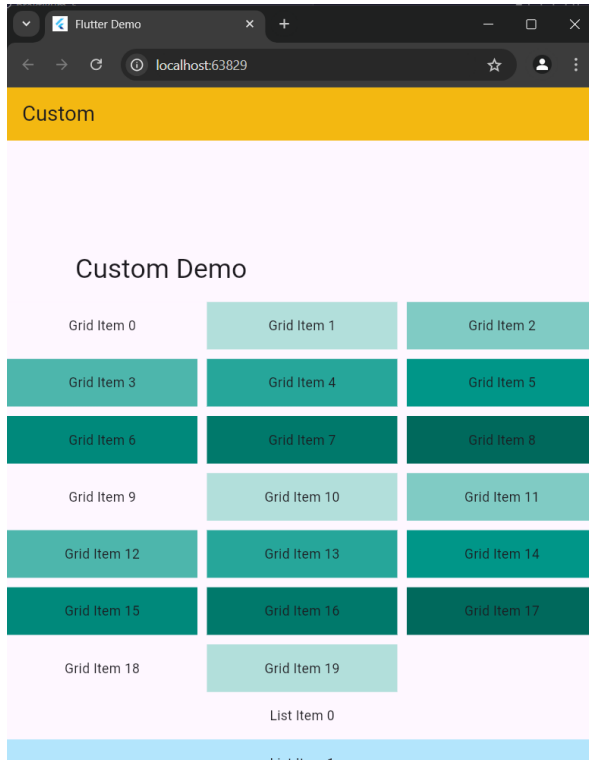
Expanded adalah turunan dari Flexible yang secara otomatis mengisi semua ruang yang tersisa di dalam kolom atau baris. Ketika Anda menggunakan Expanded, widget tersebut akan mengambil ruang sebanyak mungkin tanpa memperhatikan kebutuhan ruang minimum.



```
flexible_expanded.dart > FlexibleExpanded > _buildGreenContainer
1 import 'package:flutter/material.dart';
2
3 class FlexibleExpanded extends StatelessWidget {
4   const FlexibleExpanded({super.key});
5
6   @override
7   Widget build(BuildContext context) {
8     return Scaffold(
9       appBar: AppBar(
10        title: const Text("Flexible vs Expanded"),
11        backgroundColor: Colors.blueGrey,
12      ), // AppBar
13      body: Padding(
14        padding: const EdgeInsets.all(16.0),
15        child: Column(
16          children: [
17            _buildExampleRow(
18              title: "Flexible",
19              description: "Flexible takes up the remaining space but can shrink if needed",
20              isFlexible: true,
21            ),
22            const SizedBox(height: 20),
23            _buildExampleRow(
24              title: "Expanded",
25              description: "Expanded forces the widget to take up all the remaining space.",
26              isFlexible: false,
27            ),
28          ],
29        ), // Column
30      ), // Padding
31    ); // Scaffold
32  }
33
34  Widget _buildExampleRow(
35    required String title,
36    required String description,
37    required bool isFlexible,
38  ) {
39    return Column(
40      crossAxisAlignment: CrossAxisAlignment.start,
41      children: [
42        Text(
43          title,
44          style: const TextStyle(fontSize: 18, fontWeight: FontWeight.bold),
45        ), // Text
46        const SizedBox(height: 8),
47        Row(
48          children: [
49            Container(
50              width: 50,
51              height: 50,
52              color: Colors.red,
53            ), // Container
54            if (isFlexible)
55              Flexible(child: _buildGreenContainer(description))
56            else
57              Expanded(child: _buildGreenContainer(description)),
58            const Icon(Icons.sentiment_satisfied_alt),
59          ],
60        ), // Row
61      ],
62    ); // Column
63  }
64
65  Widget _buildGreenContainer(String text) {
66    return Container(
67      height: 50,
68      color: Colors.amber,
69      padding: const EdgeInsets.all(4),
70      child: Text(
71        text,
72        style: const TextStyle(fontSize: 12, color: Colors.white),
73      ), // Text
74    ); // Container
75  }
76 }
```

4. CustomScrollView

Widget ini memungkinkan membuat efek pada list, grid, maupun header yang lebar. Misalnya, ketika ingin membuat scroll view yang berisi app bar yang lebar yang meliputi list dan grid secara bersamaan, maka bisa menggunakan 3 widget sliver, yaitu SliverAppBar, SliverList, dan SliverGrid.



```
1 import 'package:flutter/material.dart';
2
3 class Custom extends StatelessWidget {
4   const Custom({super.key});
5
6   @override
7   Widget build(BuildContext context) {
8     return Scaffold(
9       appBar: AppBar(
10        title: const Text("Custom"),
11        backgroundColor: Colors.amber,
12      ), // AppBar
13       body: CustomScrollView(
14         slivers: <Widget>[
15           // Sliver App Bar
16           const SliverAppBar(
17             pinned: true,
18             expandedHeight: 250,
19             flexibleSpace: FlexibleSpaceBar(
20               title: Text('Custom Demo'),
21             ), // FlexibleSpaceBar
22           ), // SliverAppBar
23           // Sliver Grid
24           SliverGrid(
25             gridDelegate: const SliverGridDelegateWithMaxCrossAxisExtent(
26               maxCrossAxisExtent: 200,
27               mainAxisSpacing: 10,
28               crossAxisSpacing: 10,
29               childAspectRatio: 4,
30             ), // SliverGridDelegateWithMaxCrossAxisExtent
31             delegate: SliverChildBuilderDelegate(
32               (BuildContext context, int index) {
33                 return Container(
34                   alignment: Alignment.center,
35                   color: Colors.teal[100 * (index % 9)],
36                   child: Text('Grid Item $index'),
37                 ); // Container
38               },
39               childCount: 20,
40             ), // SliverChildBuilderDelegate
41           ), // SliverGrid
42           // Sliver Fixed List
43           SliverFixedExtentList(
44             itemExtent: 50,
45             delegate: SliverChildBuilderDelegate(
46               (BuildContext context, int index) {
47                 return Container(
48                   alignment: Alignment.center,
49                   color: Colors.lightBlue[100 * (index % 9)],
50                   child: Text('List Item $index'),
51                 ); // Container
52               },
53               childCount: 20,
54             ), // SliverChildBuilderDelegate
55           ), // SliverFixedExtentList
56         ], // <Widget>[]
57       ), // CustomScrollView
58     ); // Scaffold
59   }
60 }
```

UNGUIDED

1. Modifikasi project Rekomendasi Wisata pada Tugas Unguided 04 modul Antarmuka Pengguna dengan mengimplementasikan widget CustomScrollView, SliverAppBar, dan SliverList untuk merekomendasikan beberapa tempat wisata yang ada di Banyumas disertai foto, nama wisata, dan deskripsi singkat! (buatlah se kreatif mungkin).

```

import 'package:flutter/material.dart';

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Wisata Banyumas',
      debugShowCheckedModeBanner: false,
      theme: ThemeData(
        primarySwatch: Colors.orange,
        scaffoldBackgroundColor: Colors.orange[50],
      ),
      home: WisataListPage(),
    );
  }
}

class WisataListPage extends StatelessWidget {
  final List<Map<String, String>> wisataList = [
    {
      'nama': 'Baturraden',
      'deskripsi':
        'Objek wisata alam di lereng Gunung Slamet dengan pemandangan indah, air terjun,
dan berbagai wahana rekreasi.',
      'imageUrl':
        'https://1.bp.blogspot.com/-
_zqY3NnvYRM/ViI8Fpkw9XI/AAAAAAAAHDw/ONooPXfN15s/s1600/pintu%2Bmasu
k%2BBlokawsiata%2Bbaturaden%2Brawalo-ku.bogspot.com.jpg',
    },
    {
      'nama': 'Curug Cipendok',
      'deskripsi':
        'Air terjun spektakuler setinggi 92 meter yang dikelilingi hutan tropis yang rimbun.',
      'imageUrl':
        'https://i0.wp.com/upload.wikimedia.org/wikipedia/commons/c/cb/Curug_Cipendok
%2C_Purwokerto%2C_2015-03-22.jpg',
    },
    {
      'nama': 'Telaga Sunyi',
      'deskripsi':
        'Danau tersembunyi di tengah hutan dengan air yang tenang dan pemandangan yang
menenangkan.',
      'imageUrl':
        'https://th.bing.com/th/id/OIP.PqBOPdD4wwDRA3lw3NbE9gHaE8?rs=1&pid=Img
DetMain',
    },
  ],

```

```

{
  'nama': 'Small World',
  'deskripsi':
    'Taman miniatur dengan replika bangunan terkenal dari seluruh dunia, cocok untuk
    edukasi dan foto-foto.',
  'imageUrl':
    'https://th.bing.com/th/id/OIP.F7hjRE6W5nWEFDl_LH98MgHaEn?rs=1&pid=Img
    DetMain',
  },
  {
    'nama': 'Taman Andhang Pangrenan',
    'deskripsi':
      'Taman kota yang asri dengan berbagai fasilitas olahraga dan area bermain anak.',
    'imageUrl':
      'https://3.bp.blogspot.com/-
      KfqpvWH1anI/VMWbGUTMGxI/AAAAAAAAA6o/D1QFZmS0HYE/s1600/taman%2B
      andhang-pangrenan-purwokerto%2B(2).jpg',
    },
  ];

  WisataListPage( {super.key} );

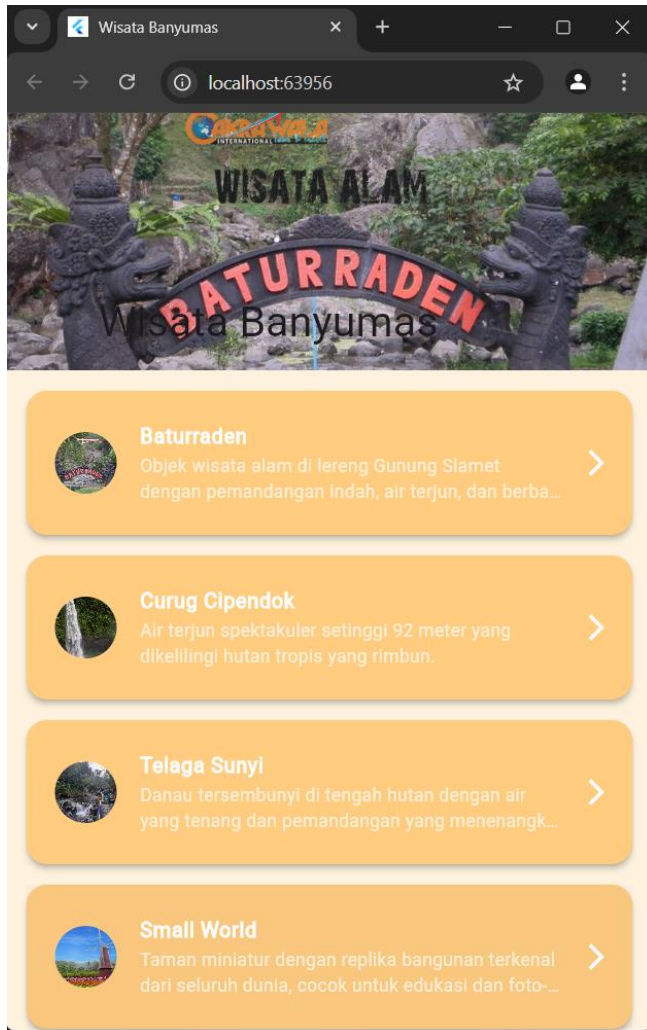
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: CustomScrollView(
        slivers: [
          SliverAppBar(
            expandedHeight: 200.0,
            floating: false,
            pinned: false,
            flexibleSpace: FlexibleSpaceBar(
              title: const Text('Wisata Banyumas'),
              background: Image.network(
                'https://1.bp.blogspot.com/-
                qU2nBUTNV9E/WNt0uHkBV2I/AAAAAAAAAQg/8xid6JqNX9UjZMXKDSmibzG-
                jMfwkaMfACLeB/s1600/Obyek-Wisata-Baturaden%2Bcopy.jpg',
                fit: BoxFit.cover,
              ),
              collapseMode: CollapseMode.parallax,
              stretchModes: const [StretchMode.fadeTitle],
            ),
            shape: const RoundedRectangleBorder(
              borderRadius: BorderRadius.only(
                bottomLeft: Radius.circular(20),
                bottomRight: Radius.circular(20),
              ),
            ),
            backgroundColor: Colors.orange,
          ),
        ],
      ),
    );
  }

```

```

SliverPadding(
padding: const EdgeInsets.symmetric(vertical: 8),
sliver: SliverList(
delegate: SliverChildBuilderDelegate(
(context, index) {
final wisata = wisataList[index];
return Card(
margin:
const EdgeInsets.symmetric(horizontal: 16, vertical: 8),
color: Colors.orange[200],
elevation: 4,
shape: RoundedRectangleBorder(
borderRadius: BorderRadius.circular(16),
),
child: ListTile(
contentPadding: const EdgeInsets.all(16),
leading: CircleAvatar(
radius: 30,
backgroundImage: NetworkImage(wisata['imageUrl']!),
backgroundColor: Colors.orange[100],
),
title: Text(
wisata['nama']!,
style: const TextStyle(
fontWeight: FontWeight.bold, color: Colors.white),
),
subtitle: Text(
wisata['deskripsi']!,
style: const TextStyle(color: Colors.white70),
maxLines: 2,
overflow: TextOverflow.ellipsis,
),
trailing: const Icon(Icons.arrow_forward_ios,
color: Colors.white),
onTap: () {
// Implementasi untuk menampilkan detail wisata
},
),
);
},
),
childCount: wisataList.length,
),
),
],
),
);
}
}

```



Program Flutter ini menampilkan rekomendasi wisata di Banyumas menggunakan widget seperti CustomScrollView, SliverAppBar, dan SliverList untuk antarmuka dinamis. SliverAppBar di bagian atas berisi gambar latar dan judul "Wisata Banyumas" yang hilang saat di-scroll. Daftar tempat wisata, seperti Baturraden, Curug Cipendok, dan Telaga Sunyi, ditampilkan dalam kartu dengan foto dan deskripsi singkat. Desain menggunakan dominasi warna oranye, dengan ListTile dan CircleAvatar untuk foto. Efek parallax pada SliverAppBar menambah kesan profesional, dan scroll responsif di berbagai ukuran layar.