

**LAPORAN PRAKTIKUM**  
**Modul 05**  
**“ANTARMUKA PENGGUNA LANJUTAN”**



**Disusun Oleh:**  
**Ganesha Rahman Gibran -2211104058**  
**Kelas S1SE-06-02**

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

### Tujuan

1. Mahasiswa mampu memahami konsep layout pada Flutter.
2. Mahasiswa dapat mengimplementasikan desain user interface pada Flutter.

### Landasan Teori

#### **ListView.builder**

Widget ini cocok digunakan saat memiliki daftar data yang besar. `ListView.builder` memerlukan dua parameter penting: `itemBuilder` untuk membangun widget yang ditampilkan, dan `itemCount` untuk menentukan jumlah item yang ingin ditampilkan.

Implementasi `ListView.builder`:

```
final List<String> entries = <String>['A', 'B', 'C'];
final List<int> colorCodes = <int>[600, 500, 100];

ListView.builder(
  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]}'),
      ),
    );
  });
```

#### **ListView.separated**

`ListView` jenis ini akan menampilkan daftar item dengan pemisah (separator) di antara setiap item. Selain `itemBuilder` dan `itemCount`, widget ini juga memerlukan parameter `separatorBuilder` untuk menentukan widget yang digunakan sebagai pemisah.

Implementasi `ListView.separated`:

```
final List<String> entries = <String>['A', 'B', 'C'];
final List<int> colorCodes = <int>[600, 500, 100];

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]}')),  
  );  
},  
separatorBuilder: (BuildContext context, int index) => const  
Divider(),  
);
```

### Flexible dan Expanded

- **Flexible**

Flexible memungkinkan widget mengambil ruang sisa di dalam layout tanpa melebihi batas maksimal yang diperlukan.

- **Expanded**

Expanded adalah turunan dari Flexible yang akan mengisi semua ruang yang tersisa dalam kolom atau baris, tanpa memperhatikan kebutuhan ruang minimum.

### Implementasi Flexible dan Expanded

```
Column(  
  children: [  
    // Flexible  
    Row(  
      children: <Widget>[  
        Container(  
          width: 50,  
          height: 100,  
          color: Colors.red,  
        ),  
        Flexible(  
          child: Container(  
            height: 100,  
            color: Colors.green,  
            child: const Text(  
              "Flexible takes up the remaining space but can  
shrink if needed.",  
            ),  
          ),  
        ),  
        const Icon(Icons.sentiment_very_satisfied),  
      ],  
    ),  
  ],  
)
```

```
    ],  
  ),  
  SizedBox(height: 20),  
  
  // Expanded  
  Row(  
    children: <Widget>[  
      Container(  
        width: 50,  
        height: 100,  
        color: Colors.red,  
      ),  
      Expanded(  
        child: Container(  
          height: 100,  
          color: Colors.green,  
          child: const Text(  
            "Expanded forces the widget to take up all the  
remaining space.",  
          ),  
        ),  
      ),  
      const Icon(Icons.sentiment_very_satisfied),  
    ],  
  ),  
],  
);
```

### CustomScrollView

Widget ini memungkinkan pengembang untuk membuat tata letak dengan efek scroll yang melibatkan elemen seperti SliverAppBar, SliverList, dan SliverGrid.

#### Implementasi CustomScrollView

```
CustomScrollView(  
  slivers: <Widget>[  
    // Sliver AppBar  
    const SliverAppBar(  
      pinned: true,  
      expandedHeight: 250.0,  
      flexibleSpace: FlexibleSpaceBar(  
        title: Text('Demo'),  
      ),  
    ),  
  ],  
);
```

```
    ),  
  ),  
  // Sliver Grid  
  SliverGrid(  
    gridDelegate: const  
SliverGridDelegateWithMaxCrossAxisExtent(  
      maxCrossAxisExtent: 200.0,  
      mainAxisSpacing: 10.0,  
      crossAxisSpacing: 10.0,  
      childAspectRatio: 4.0,  
    ),  
    delegate: SliverChildBuilderDelegate(  
      (BuildContext context, int index) {  
        return Container(  
          alignment: Alignment.center,  
          color: Colors.teal[100 * (index % 9)],  
          child: Text('Grid Item $index'),  
        );  
      },  
      childCount: 20,  
    ),  
  ),  
  // Sliver Fixed List  
  SliverFixedExtentList(  
    itemExtent: 50.0,  
    delegate: SliverChildBuilderDelegate(  
      (BuildContext context, int index) {  
        return Container(  
          alignment: Alignment.center,  
          color: Colors.lightBlue[100 * (index % 9)],  
          child: Text('List Item $index'),  
        );  
      },  
    ),  
  ),  
],  
);
```

## Guided

### Main.dart

```
import 'package:flutter/material.dart';
import 'package:guidodi/custom.dart';
import 'package:guidodi/flexible_expanded.dart'; // Unused import: 'package:guidodi/flexible_expanded.dart'
import 'package:guidodi/jenis_list_view.dart'; // Unused import: 'package:guidodi/jenis_list_view.dart'

Run | Debug | Profile
void main() {
  runApp(const MyApp());
}

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      debugShowCheckedModeBanner: false,
      theme: ThemeData(
        colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),
        useMaterial3: true,
      ), // ThemeData
      home:
        Custom(), // Use 'const' with the constructor to improve performance.
        // FlexibleExpandedScreen(),
        // JenisListView(),
    ); // MaterialApp
  }
}

class MyHomePage extends StatefulWidget {
  const MyHomePage({super.key, required this.title});
  final String title;

  @override
  State<MyHomePage> createState() => _MyHomePageState();
}

class _MyHomePageState extends State<MyHomePage> {
  int _counter = 0;

  void _incrementCounter() {
    setState(() {
      _counter++;
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        backgroundColor: Theme.of(context).colorScheme.inversePrimary,
        title: Text(widget.title),
      ), // AppBar
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: <Widget>[
            const Text(
              'You have pushed the button this many times:',
            ), // Text
            Text(
              '$_counter',
              style: Theme.of(context).textTheme.headlineMedium,
            ), // Text
          ], // <Widget>[]
        ), // Column
      ), // Center
      floatingActionButton: FloatingActionButton(
        onPressed: _incrementCounter,
        tooltip: 'Increment',
        child: const Icon(Icons.add),
      ), // FloatingActionButton
    ); // Scaffold
  }
}
```

Input :

- jenis\_list\_view.dart

```
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 List View"),
        backgroundColor: Colors.amber,
      ), // AppBar
      body: ListView.separated(
        itemBuilder: (BuildContext context, int index) {
          return Container(
            height: 50,
            color: Colors.amber[colorCodes[index]],
            child: Center(child: Text("Entry ${entries[index]}")),
          ); // Container
        },
        itemCount: entries.length,
        separatorBuilder: (BuildContext context, int index) {
          return Container(
            height: 10,
            color: Colors.red,
          ); // Container
        },
      ), // ListView.separated
    ); // Scaffold
  }
}
```

- flexible\_expanded.dart

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

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

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text("Flexible vs Expanded"), // Use 'const' with the constructor to improve performance. Try adding the 'const' keyword to the Text widget.
        backgroundColor: Colors.amber,
      ), // AppBar
      body: Column(
        children: [
          Text( // Use 'const' with the constructor to improve performance. Try adding the 'const' keyword to the Text widget.
            "Flexible",
            style: TextStyle(fontSize: 18), // Use 'const' with the constructor to improve performance. Try adding the 'const' keyword to the TextStyle widget.
          ), // Text
          SizedBox( // Use 'const' with the constructor to improve performance. Try adding the 'const' keyword to the SizedBox widget.
            height: 8,
          ), // SizedBox
          Row(
            children: [
              Container(
                width: 50,
                height: 100,
                color: Colors.red,
              ), // Container
              Flexible(
                child: Container(
                  height: 100,
                  color: Colors.amber,
                  child:
                    const Text("This is Flexible widget, it takes up remaining space but can shrink if needed"),
                ), // Container
              ), // Flexible
              const Icon(Icons.sentiment_satisfied_alt_rounded)
            ],
          ), // Row
          SizedBox( // Use 'const' with the constructor to improve performance. Try adding the 'const' keyword to the SizedBox widget.
            height: 10,
          ), // SizedBox
          Text( // Use 'const' with the constructor to improve performance. Try adding the 'const' keyword to the Text widget.
            "Expanded",
            style: TextStyle(fontSize: 18), // Use 'const' with the constructor to improve performance. Try adding the 'const' keyword to the TextStyle widget.
          ), // Text
          const SizedBox(
            height: 8,
          ), // SizedBox
          Row(
            children: [
              Container(
                width: 50,
                height: 100,
                color: Colors.red,
              ), // Container
              Expanded(
                child: Container(
                  height: 100,
                  color: Colors.amber,
                  child:
                    const Text("This is Expanded widget, it forces the widget to take remaining spaces"),
                ), // Container
              ), // Expanded
              const Icon(Icons.sentiment_satisfied_alt_rounded)
            ],
          ), // Row
        ],
      ), // Column
    ); // Scaffold
  }
}
```



- custom.dart

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

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

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: CustomScrollView(
        slivers: <Widget>[
          const SliverAppBar(
            pinned: true,
            expandedHeight: 200,
            flexibleSpace: FlexibleSpaceBar(
              title: Text("Custom Demo"),
            ), // FlexibleSpaceBar
          ), // SliverAppBar

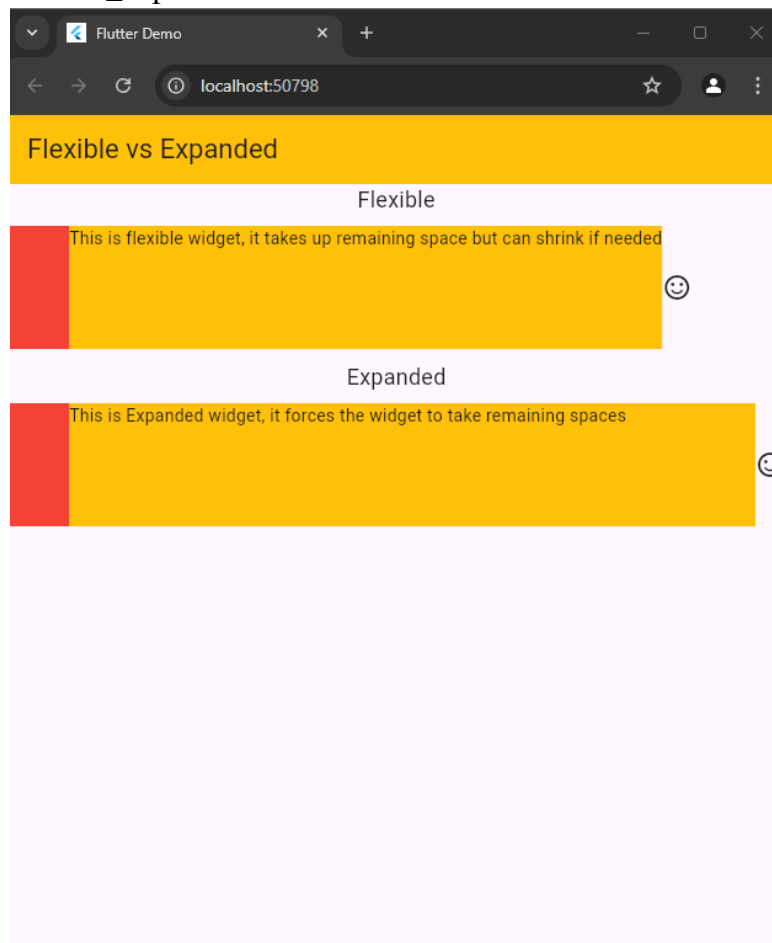
          // SliverGrid -> tampilan grid
          SliverGrid(
            delegate: SliverChildBuilderDelegate(
              (BuildContext context, int index) {
                return Container(
                  height: 100,
                  alignment: Alignment.center,
                  color: Colors.amber[100 * (index % 9)],
                  child: Column(
                    children: [
                      Icon(Icons.access_alarm, size: 15,),
                      Text('Grid Item $index'),
                    ],
                  ), // Column
                ); // Container
              },
            childCount: 20,
          ), // SliverChildBuilderDelegate
          gridDelegate: const SliverGridDelegateWithMaxCrossAxisExtent(
            maxCrossAxisExtent: 200,
            mainAxisSpacing: 10,
            crossAxisSpacing: 10,
            childAspectRatio: 4,
          ), // SliverGridDelegateWithMaxCrossAxisExtent
        ], // SliverGrid
      ), // <Widget>[]
    ), // CustomScrollView
  ); // Scaffold
}
```

Output :

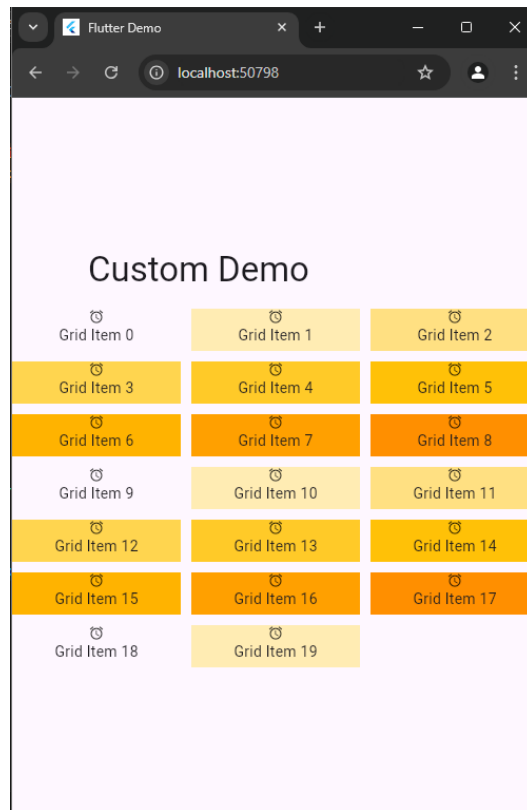
- `jenis_list_view.dart`



- `flexible_expanded.dart`



- custom.dart



### 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).

Input :

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

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

class MyApp extends StatelessWidget {
  const MyApp({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Rekomendasi Wisata',
      theme: ThemeData(
        primarySwatch: Colors.blue,
        scaffoldBackgroundColor: Colors.white,
      ),
    );
  }
}
```

```

        home: const TouristDestinations(),
        debugShowCheckedModeBanner: false,
    );
}
}

class TouristDestinations extends StatelessWidget {
  const TouristDestinations({Key? key}) : super(key: key);

  final List<Map<String, String>> destinations = const [
    {
      "name": "Baturraden",
      "description":
        "Baturraden adalah sebuah kawasan wisata yang terletak sekitar 15 km
        dari pusat kota Purwokerto di lereng selatan Gunung Slamet",
      "image":
        "https://asset.kompas.com/crops/0nZi0zF4l1Kf6pwzhfeQ9Fa7Xs=/4x0:997x66
        2/750x500/data/photo/2022/06/06/629db5191baf7.jpg"
    },
    {
      "name": "Telaga Sunyi",
      "description":
        "Telaga Sunyi terletak di Kabupaten Banyumas, Jawa Tengah, dan berada
        pada ketinggian sekitar 700 meter.",
      "image":
        "https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQWxlXQHnRdnq-
        J6jNkUr_21ervTOcmunPffw&s"
    },
    {
      "name": "Curug Telu",
      "description":
        "Curug Telu terletak di Desa Karangsalam, Kecamatan Baturraden,
        Kabupaten Banyumas.",
      "image":
        "https://www.banyumasekspres.id/wp-content/uploads/2022/12/curug-
        telu.jpg"
    },
  ];

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: CustomScrollView(
        slivers: [
          SliverAppBar(
            expandedHeight: 250.0,
            floating: false,
            backgroundColor: const Color.fromARGB(255, 132, 255, 16),
            pinned: true,
            flexibleSpace: FlexibleSpaceBar(
              title: const Text(
                "Wisata Banyumas",
                style: TextStyle(
                  color : Colors.white,
                ),
              ),
            ),
          ),
        ],
      ),
    );
  }
}

```

```

    ),
    background: Image.network(
      'https://backpackerjakarta.com/wp-content/uploads/2017/07/gunung-
      slamet-jawa-tengah-1024x797.jpg',
      fit: BoxFit.cover,
      loadingBuilder: (context, child, loadingProgress) {
        if (loadingProgress == null) return child;
        return Center(
          child: CircularProgressIndicator(),
        );
      },
      errorBuilder: (context, error, stackTrace) {
        return Container(
          color: Colors.grey[300],
          child: const Center(
            child: Icon(Icons.error, color: Colors.red),
          ),
        );
      },
    ),
  ),
  actions: [
    IconButton(
      icon: const Icon(Icons.search),
      onPressed: () {
        // Add search functionality
      },
    ),
  ],
),
SliverToBoxAdapter(
  child: _buildWelcomeBanner(),
),
SliverList(
  delegate: SliverChildBuilderDelegate(
    (context, index) {
      return _buildDestinationCard(context, destinations[index]);
    },
    childCount: destinations.length,
  ),
),
],
),
);
}

Widget _buildWelcomeBanner() {
  return Container(
    padding: const EdgeInsets.all(16.0),
    color: Colors.white,
    child: Column(
      crossAxisAlignment: CrossAxisAlignment.start,
      children: const [
        Text(
          'Selamat Datang di Wisata Banyumas!',

```

```

        style: TextStyle(
          fontSize: 24,
          fontWeight: FontWeight.bold,
          color: const Color.fromARGB(255, 132, 255, 16),
        ),
      ),
      SizedBox(height: 5),
      Text(
        'Temukan berbagai objek wisata menarik di Banyumas.',
        style: TextStyle(fontSize: 16, color: Colors.black54),
      ),
    ],
  ),
);
}

Widget _buildDestinationCard(
  BuildContext context, Map<String, String> destination) {
  return Card(
    elevation: 8,
    margin: const EdgeInsets.symmetric(vertical: 10, horizontal: 15),
    shape: RoundedRectangleBorder(
      borderRadius: BorderRadius.circular(20),
    ),
    child: Stack(
      children: [
        ClipRRect(
          borderRadius: const BorderRadius.vertical(top: Radius.circular(20)),
          child: Image.network(
            destination['image']!,
            fit: BoxFit.cover,
            width: double.infinity,
            height: 200,
            loadingBuilder: (context, child, loadingProgress) {
              if (loadingProgress == null) return child;
              return Center(
                child: CircularProgressIndicator(
                  value: loadingProgress.expectedTotalBytes != null
                    ? loadingProgress.cumulativeBytesLoaded /
                      (loadingProgress.expectedTotalBytes ?? 1)
                    : null,
                ),
              ),
            ),
          ),
        ],
      ),
      errorBuilder: (context, error, stackTrace) {
        return Container(
          color: Colors.grey[300],
          height: 200,
          child: const Center(
            child: Icon(Icons.error, color: Colors.red),
          ),
        );
      },
    ),
  ),
);

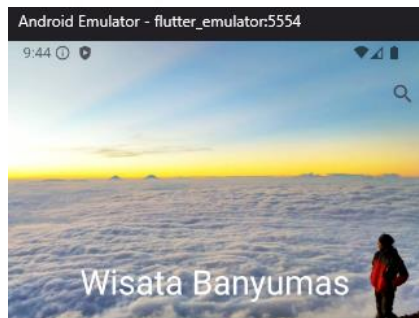
```

```

    Positioned(
      bottom: 10,
      left: 10,
      right: 10,
      child: Container(
        padding: const EdgeInsets.all(10),
        decoration: BoxDecoration(
          color: Colors.black54,
          borderRadius: BorderRadius.circular(10),
        ),
      ),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [
          Text(
            destination['name']!,
            style: const TextStyle(
              fontSize: 24,
              fontWeight: FontWeight.bold,
              color: Colors.white,
            ),
          ),
          const SizedBox(height: 5),
          Container(
            constraints: const BoxConstraints(
              maxHeight: 50,
            ),
            child: Text(
              destination['description']!,
              style: const TextStyle(
                fontSize: 16,
                color: Colors.white70,
              ),
            ),
            overflow: TextOverflow.ellipsis,
            maxLines: 2,
          ),
        ],
      ),
    ],
  ),
),
];
);
}
}

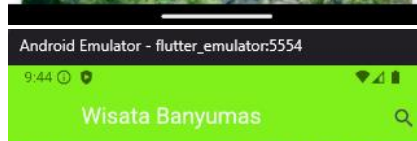
```

Output :



### Selamat Datang di Wisata Banyumas!

Temukan berbagai objek wisata menarik di  
Banyumas.



## Kesimpulan

Flutter adalah framework yang memungkinkan pengembang membangun aplikasi mobile dengan antarmuka pengguna yang responsif dan dinamis, di mana elemen tampilan disusun dari berbagai jenis widget. Widget seperti “ListView.builder”,



“ListView.separated”, “Flexible”, “Expanded”, dan “CustomScrollView” memiliki peran penting dalam mengelola tata letak dan tampilan data. “ListView.builder” cocok untuk daftar data yang besar, “ListView.separated” menambah pemisah antar item, sementara “Flexible” dan “Expanded” membantu mengatur penggunaan ruang dalam layout. Dengan memahami berbagai widget ini, pengembang dapat menyusun tata letak yang fleksibel dan efisien, menciptakan pengalaman pengguna yang optimal dalam aplikasi.