

**LAPORAN PRAKTIKUM
PEMROGRAMAN PERANGKAT BERGERAK**

**MODUL IV
ANTARMUKA PENGGUNA**



Disusun Oleh :

Dewi Atika Muthi / 2211104042

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**

BAB I

PENDAHULUAN

A. DASAR TEORI

Flutter adalah framework open-source yang dikembangkan oleh Google untuk membangun aplikasi multi-platform dengan kode tunggal. Dalam pengembangan aplikasi menggunakan Flutter, antarmuka pengguna (UI) dibangun menggunakan widget. Widget adalah elemen dasar dalam Flutter yang merepresentasikan komponen UI seperti tombol, teks, atau bahkan layout keseluruhan aplikasi.

Beberapa konsep penting dalam pengembangan UI Flutter meliputi:

1. **Widget:** Semua elemen UI dalam Flutter adalah widget. Widget dapat berupa elemen visual, kontrol, atau container untuk widget lainnya.
2. **Layout:** Flutter menyediakan berbagai widget layout seperti Container, Row, Column, Stack, dan GridView untuk mengatur tata letak widget.
3. **State Management:** Flutter menggunakan konsep state untuk mengelola data yang dapat berubah dalam aplikasi.

B. MAKSUD DAN TUJUAN

Praktikum ini bertujuan untuk:

1. Memahami konsep layout pada Flutter
2. Mengimplementasikan desain user interface pada Flutter
3. Mempelajari penggunaan berbagai jenis widget layout seperti Container, Row, Column, ListView, GridView, dan Stack
4. Melatih kemampuan dalam merancang dan mengembangkan antarmuka pengguna yang responsif dan menarik menggunakan Flutter

BAB II IMPLEMENTASI

A. PRAKTIKUM

1. GridView

GridView adalah widget yang digunakan untuk menampilkan item dalam bentuk grid atau kisi-kisi garis. Berikut adalah contoh implementasi GridView:

Soal Studi Case

Saya mau membuat program gridview yang menampilkan 6 kotak dengan warna berbeda.

Sourcecode

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

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: GridviewApp(),
    );
  }
}

class GridviewApp extends StatefulWidget {
  const GridviewApp({super.key});

  @override
  State<GridviewApp> createState() => _GridviewAppState();
}

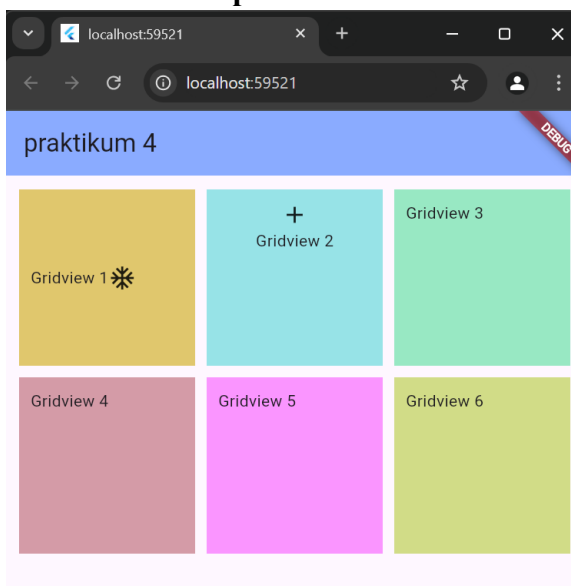
class _GridviewAppState extends State<GridviewApp> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('praktikum 4'),
        backgroundColor: const Color.fromARGB(255, 138, 171,
255),
      ),
      body: GridView.count(
        crossAxisCount: 3,
        mainAxisSpacing: 10,
        crossAxisSpacing: 10,
        padding: EdgeInsets.all(12),
        children: [
          Container(
            padding: EdgeInsets.all(10),
            child: Row(
              children: [
                Text('Gridview 1'),
```

```

        Icon(Icons.ac_unit),
      ],
    ),
    color: const Color.fromARGB(255, 224, 199, 109),
  ),
  Container(
    padding: EdgeInsets.all(10),
    child: Column(
      children: [
        Icon(Icons.add),
        Text('Gridview 2'),
      ],
    ),
    color: const Color.fromARGB(255, 151, 227, 231),
  ),
  Container(
    padding: EdgeInsets.all(10),
    child: Text('Gridview 3'),
    color: const Color.fromARGB(255, 152, 232, 195),
  ),
  Container(
    padding: EdgeInsets.all(10),
    child: Text('Gridview 4'),
    color: const Color.fromARGB(255, 212, 155, 167),
  ),
  Container(
    padding: EdgeInsets.all(10),
    child: Text('Gridview 5'),
    color: const Color.fromARGB(255, 250, 149, 255),
  ),
  Container(
    padding: EdgeInsets.all(10),
    child: Text('Gridview 6'),
    color: const Color.fromARGB(255, 209, 220, 135),
  ),
],
));
}
}

```

Screenshot Output



Deskripsi Program

Kode ini membuat grid dengan jumlah kolom yang tetap. Parameter `crossAxisCount: 3` menunjukkan bahwa grid akan memiliki 3 kolom.

1. MainAxisSpacing dan CrossAxisSpacing

- `mainAxisSpacing`: Memberikan jarak vertikal antar item di grid.
- `crossAxisSpacing`: Memberikan jarak horizontal antar item di grid.

2. Padding

Digunakan untuk memberi jarak antara konten grid dan tepi layer, seluruh grid diberi padding 12 unit dari tepi layar.

3. Children

GridView diisi dengan beberapa `Container` sebagai **children**, di mana setiap container berisi widget lain, seperti:

- `Text`: Untuk menampilkan teks seperti "Gridview 1".
- `Icon`: Untuk menampilkan ikon, seperti `Icons.ac_unit` atau `Icons.add`.
- Penataan widget dalam setiap container dapat menggunakan `Row` (untuk horizontal) atau `Column` (untuk vertikal).

5. Desain Tampilan

- Setiap `Container` memiliki warna background berbeda, menggunakan `color: Color.fromARGB(...)`, sehingga tampilan grid lebih bervariasi.

2. ListView

ListView adalah widget yang digunakan untuk menampilkan daftar item yang dapat di-scroll. Berikut contoh implementasi ListView:

Soal Studi Case

Membuat program ListView:

Sourcecode

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

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: const ListViewApp(),
    );
  }
}
```

```

    );
  }
}

class ListViewApp extends StatefulWidget {
  const ListViewApp({super.key});

  @override
  State<ListViewApp> createState() => _ListViewAppState();
}

class _ListViewAppState extends State<ListViewApp> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Latihan ListView'),
      ),
      body: ListView(
        children: [
          Container(
            height: 200,
            width: 200,
            color: const Color.fromARGB(255, 255, 131, 122),
            child: Center(
              child: Text("Type A"),
            ),
          ),
          Container(
            height: 200,
            width: 200,
            color: const Color.fromARGB(255, 164, 184, 255),
            child: Center(
              child: Text("Type A"),
            ),
          ),
          Container(
            height: 200,
            width: 200,
            color: const Color.fromARGB(255, 240, 156, 255),
            child: Center(
              child: Text("Type A"),
            ),
          ),
          Container(
            height: 200,
            width: 200,
            color: const Color.fromARGB(255, 149, 255, 188),
            child: Center(
              child: Text("Type A"),
            ),
          ),
        ],
      ),
    );
  }
}

```

Screenshoot Output



Deskripsi Program:

ListView digunakan sebagai body dari **Scaffold** untuk menampilkan daftar konten secara vertikal. Setiap elemen daftar di dalam ListView adalah sebuah **Container**.

ListView memungkinkan pengguna untuk menggulirkan konten yang panjang dalam satu arah (dalam dalam program ini digunakan secara vertikal).

p/s: Untuk menampilkan konten dalam **orientasi horizontal**, Anda hanya perlu menambahkan properti **scrollDirection** dengan nilai **Axis.horizontal** pada widget **ListView**.

Setiap item dalam ListView adalah **Container** dengan properti **height** dan **width** yang sama (**200 unit**).

Setiap **container** memiliki warna background yang berbeda, yang ditentukan oleh kode warna menggunakan **Color.fromARGB(...)** yang setiap konten container berupa teks "Type A" etc, ditampilkan di tengah dengan widget **Center**.

3. Stack

Stack adalah widget yang memungkinkan penumpukan widget satu di atas yang lain. Berikut contoh implementasi Stack:

Soal Studi Case

Membuat 2 program stack: tumpukan warna, dan gradiasi warna dan teks di atasnya.

Sourcecode

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

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: const StackApp(),
    );
  }
}

class StackApp extends StatefulWidget {
  const StackApp({super.key});

  @override
  State<StackApp> createState() => _StackAppState();
}

class _StackAppState extends State<StackApp> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Latihan Stack'),
        backgroundColor: const Color.fromARGB(255, 255, 233,
154),
      ),
      body: Column(
        children: [
          Stack(
            children: <Widget>[
              Container(
                margin: EdgeInsets.all(5),
                color: const Color.fromARGB(255, 86, 165, 229),
                height: 100,
                width: 100,
              ),
            ],
          ),
        ],
      ),
    );
  }
}
```

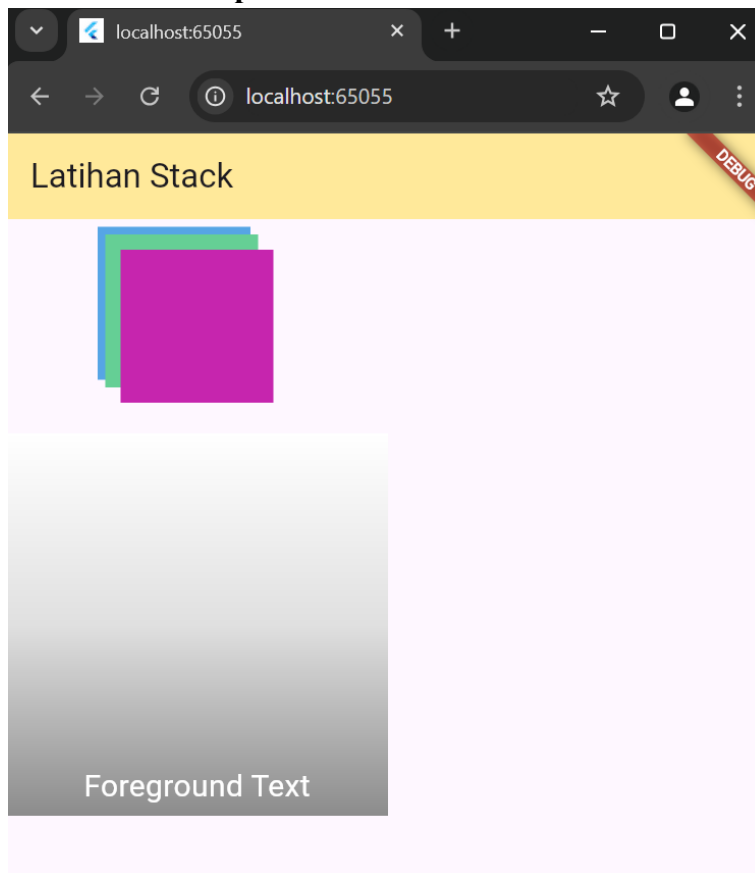


```

        Container(
          margin: EdgeInsets.all(10),
          color: const Color.fromARGB(255, 101, 207, 149),
          height: 100,
          width: 100,
        ),
        Container(
          margin: EdgeInsets.all(20),
          color: const Color.fromARGB(255, 198, 37, 174),
          height: 100,
          width: 100,
        ),
      ],
    ),
    SizedBox(
      width: 250,
      height: 250,
      child: Stack(
        children: <Widget>[
          Container(
            width: 250,
            height: 250,
            color: Colors.white,
          ),
          Container(
            padding: const EdgeInsets.all(5.0),
            alignment: Alignment.bottomCenter,
            decoration: BoxDecoration(
              gradient: LinearGradient(
                begin: Alignment.topCenter,
                end: Alignment.bottomCenter,
                colors: <Color>[
                  Colors.black.withAlpha(0),
                  Colors.black12,
                  Colors.black45
                ],
              ),
            ),
            child: const Text(
              'Foreground Text',
              style: TextStyle(color: Colors.white,
fontSize: 20.0),
            ),
          ),
        ],
      ),
    ),
  );
}

```

Screenshoot Output



Deskripsi Program:

1. Tumpukan Warna:

Terdapat tiga **Container** berukuran 100x100 dengan warna berbeda: biru muda, hijau muda, dan ungu. Masing-masing memiliki margin yang bertambah (5, 10, dan 20), sehingga terlihat seperti lapisan yang bertumpuk. **Stack** digunakan untuk menempatkan kontainer ini di atas satu sama lain dengan margin yang menggeser posisi masing-masing.

2. Gradient Hitam-Putih dengan Teks:

Stack berukuran 250x250 ini memiliki dua lapisan. Lapisan pertama adalah **Container** berwarna putih yang menjadi latar belakang. Lapisan kedua adalah **Container** dengan **gradient** hitam transparan di bagian atas hingga hitam pekat di bagian bawah, dan di atasnya terdapat teks "Foreground Text" berwarna putih di bagian bawah stack.

B. LATIHAN

1. Modifikasi project TP 04 (Tugas Pendahuluan) untuk Antarmuka Pengguna, yang mana di dalamnya terdapat ListView, untuk merekomendasikan beberapa tempat wisata yang ada di Banyumas disertai foto, nama wisata, dan deskripsi singkat!

(p/s: Saya izin mengganti wisata Baturraden **menjadi wisata Malang**, karena mendapatkan foto yang sulit izin aksesnya, jadi gambar tidak dapat muncul)

Source Code:

```
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: 'Wisata Malang',
      theme: ThemeData(
        primarySwatch: Colors.green,
        fontFamily: 'Roboto',
      ),
      home: const WisataMalangPage(),
    );
  }
}

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

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Wisata Malang'),
        backgroundColor: const Color.fromARGB(255, 167, 255, 160),
      ),
      body: ListView(
        children: const [
          WisataItem(
            image:
              'https://upload.wikimedia.org/wikipedia/commons/e/ec/Mount_Bromo_panorama.jpg',
            name: 'Gunung Bromo',
            description:
              'Gunung berapi aktif dengan pemandangan matahari terbit yang menakjubkan.',
            color: Colors.orange,
            icon: Icons.landscape,
          ),
          WisataItem(
            image:
```

```

                'https://img.inews.co.id/media/1200/files/inews_new/2
021/08/09/jatim_park.jpg',
                name: 'Jatim Park',
                description:
                    'Taman hiburan keluarga dengan berbagai wahana dan
atraksi menarik.',
                color: Colors.blue,
                icon: Icons.attractions,
            ),
            WisataItem(
                image:
                    'https://static.promediateknologi.id/crop/0x0:0x0/0x0
/webp/photo/p2/231/2024/09/11/Segarnya-suasana-kawasan-Air-Terjun-
Coban-Rondo-Gambar-dari-Dhiky-Aditya-Shutterstock-1200347082.png',
                name: 'Air Terjun Coban Rondo',
                description:
                    'Air terjun indah dengan ketinggian 84 meter,
dikelilingi hutan pinus.',
                color: Colors.teal,
                icon: Icons.water_drop,
            ),
            WisataItem(
                image:
                    'https://upload.wikimedia.org/wikipedia/commons/thumb
/f/fb/Batu_Secret_Zoo.jpg/300px-Batu_Secret_Zoo.jpg',
                name: 'Batu Secret Zoo',
                description:
                    'Kebun binatang modern dengan koleksi hewan dari
berbagai belahan dunia.',
                color: Colors.green,
                icon: Icons.pets,
            ),
            WisataItem(
                image:
                    'https://static.promediateknologi.id/crop/0x0:0x0/0x0
/webp/photo/p2/74/2024/03/27/Pantai-Balekambang-1601454064.png',
                name: 'Pantai Balekambang',
                description:
                    'Pantai eksotis dengan Pura Amerta Jati yang
menyerupai Tanah Lot di Bali.',
                color: Colors.blue,
                icon: Icons.beach_access,
            ),
        ],
    ),
);
}

```

```

class WisataItem extends StatelessWidget {
  final String image;
  final String name;
  final String description;
  final Color color;
  final IconData icon;

  const WisataItem({
    Key? key,
    required this.image,
    required this.name,
    required this.description,

```

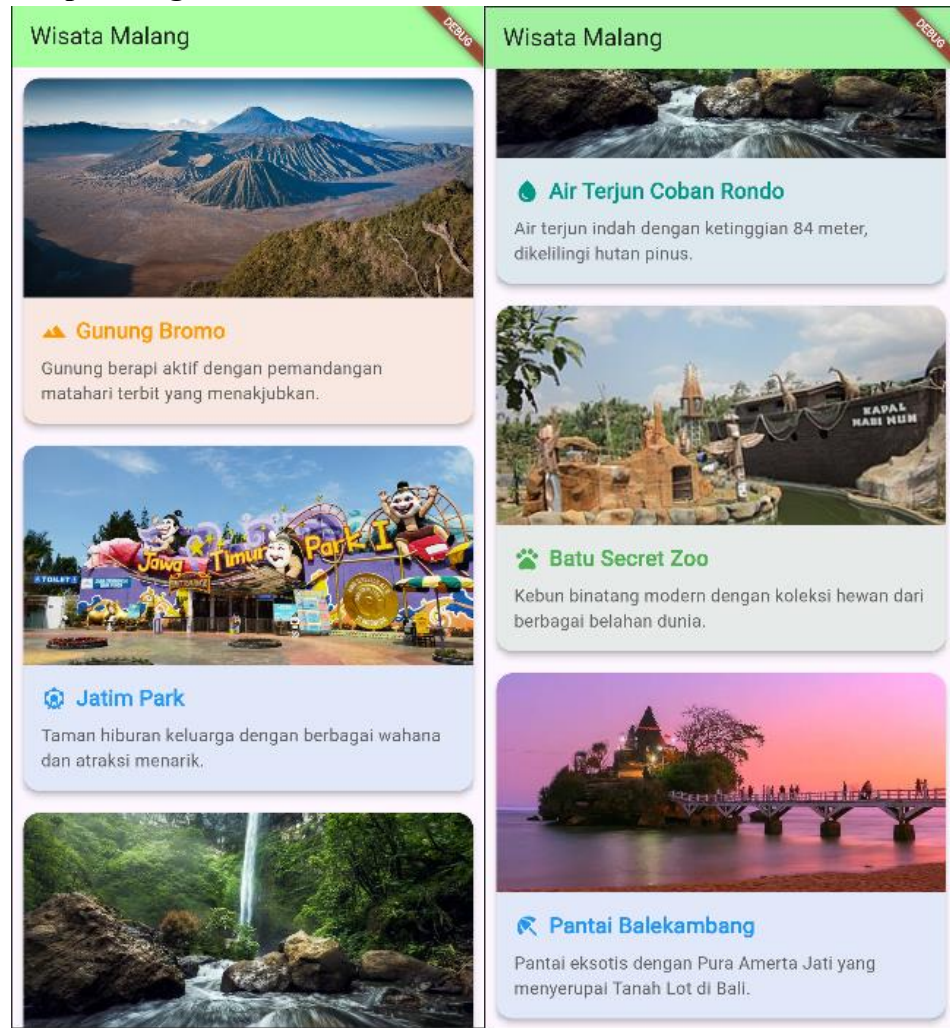
```

        required this.color,
        required this.icon,
    }) : super(key: key);

    @override
    Widget build(BuildContext context) {
        return Card(
            margin: const EdgeInsets.all(10.0),
            elevation: 5.0,
            shape: RoundedRectangleBorder(
                borderRadius: BorderRadius.circular(15.0),
            ),
            child: Column(
                children: [
                    ClipRRect(
                        borderRadius:
                            const BorderRadius.vertical(top:
                                Radius.circular(15.0)),
                        child: Image.network(
                            image,
                            height: 200,
                            width: double.infinity,
                            fit: BoxFit.cover,
                        ),
                    ),
                    Container(
                        color: color.withOpacity(0.1),
                        padding: const EdgeInsets.all(16.0),
                        child: Column(
                            crossAxisAlignment: CrossAxisAlignment.start,
                            children: [
                                Row(
                                    children: [
                                        Icon(icon, color: color),
                                        const SizedBox(width: 8),
                                        Text(
                                            name,
                                            style: TextStyle(
                                                fontSize: 20,
                                                fontWeight: FontWeight.bold,
                                                color: color,
                                            ),
                                        ),
                                    ],
                                ),
                                const SizedBox(height: 8),
                                Text(
                                    description,
                                    style: TextStyle(
                                        fontSize: 16,
                                        color: Colors.grey[700],
                                    ),
                                ),
                            ],
                        ),
                    ),
                ],
            ),
        );
    }
}

```

Output Program:



Deskripsi Program:

1. Implementasi ListView:

- ListView digunakan sebagai widget utama dalam body Scaffold.
- ListView.builder mungkin digunakan untuk efisiensi jika daftar tempat wisata panjang atau dinamis.

1. Item ListView (WisataItem):

- Setiap item dalam ListView adalah instance dari widget kustom WisataItem.
- WisataItem mewakili satu tempat wisata, menampilkan gambar, nama, deskripsi, ikon, dan warna tema.

3. Komponen WisataItem:

- Image.network: Untuk menampilkan gambar tempat wisata dari URL.
- Text widgets: Untuk nama dan deskripsi tempat wisata.
- Icon: Menampilkan ikon yang relevan dengan jenis tempat wisata.
- Container dengan warna latar: Memberikan tema visual untuk setiap item.

BAB III

KESIMPULAN DAN SARAN

A. KESIMPULAN

Praktikum ini memberikan pemahaman mendalam tentang pengembangan antarmuka pengguna menggunakan Flutter. Melalui implementasi berbagai widget layout seperti GridView, ListView, dan Stack, kita dapat melihat fleksibilitas Flutter dalam menciptakan UI yang kompleks dan responsif. GridView memungkinkan penyusunan elemen dalam format grid yang teratur, sangat berguna untuk menampilkan koleksi item seperti galeri foto atau menu aplikasi. ListView bisa digunakan untuk menampilkan daftar item yang dapat di-scroll, ideal untuk daftar kontak atau feed berita. Stack membantu menciptakan UI yang lebih dinamis dengan menumpuk widget, berguna untuk overlay atau efek visual yang kompleks. Keberagaman widget dari Flutter sangat membantu pengembangan aplikasi dengan UI yang kaya dan interaktif.

B. REFERENSI

Flutter Documentation. (n.d.). Retrieved from <https://flutter.dev/docs>
Dart Documentation. (n.d.). Retrieved from <https://dart.dev/guides>
Flutter Widget Catalog. (n.d.). Retrieved from <https://flutter.dev/docs/development/ui/widgets>