

**LAPORAN TUGAS GUIDED  
PEMROGRAMAN PERANGKAT BERGERAK**

**MODUL 4 ANTAR  
MUKA PENGGUNA**



**Disusun Oleh :**

**Ade Fatkhul Anam / 22111040451**

**SE-06-02**

**Asisten Praktikum :**

**Muhammad Faza Zulian Gesit Al Barru**

**Aisyah Hasna Aulia**

**Dosen Pengampu :**

**Yudha Islami Sulistya**

**PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2024**

## MODUL 4

### ANTARMUKA PENGGUNA

#### A. Tujuan Praktikum

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

#### B. Pengenalan Widget

Flutter menggunakan Widget sebagai komponen dasar untuk semua tampilan. Setiap widget dipecah menjadi komponen-komponen kecil yang memiliki state dan konfigurasi masing-masing. Ketika ada perubahan pada state, widget akan membuat ulang dirinya sendiri agar tampilan selalu sesuai dengan perubahan tersebut.

#### C. GridView

GridView adalah widget dalam Flutter yang mirip dengan Array 2D pada berbagai bahasa pemrograman. Widget ini digunakan untuk menampilkan elemen-elemen dalam format grid, seperti gambar, teks, ikon, dan sebagainya. Berikut adalah contoh penggunaan GridView:

Source Code:

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

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

  @override
  State<GridViewApp> createState() => _GridViewState();
}

class _GridViewState extends State<GridViewApp> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Praktikum 4 Ade Fatkhul Anam'),
        backgroundColor: Colors.blueAccent[700],
      ),
      body: GridView.count(
        crossAxisCount: 2,
        mainAxisSpacing: 10,
        crossAxisSpacing: 10,
        padding: EdgeInsets.all(12),
        children: [
          Container(
            padding: EdgeInsets.all(10),
            child: Text('GridView1'),
            color: Colors.amberAccent,
          ),
          Container(
            padding: EdgeInsets.all(10),
            child: Text('GridView2'),
            color: Colors.red,
```

```

    ),
    Container(
      padding: EdgeInsets.all(10),
      child: Text('GridView3'),
      color: Colors.green,
    ),
    Container(
      padding: EdgeInsets.all(10),
      child: Text('GridView4'),
      color: Colors.yellow,
    ),
    Container(
      padding: EdgeInsets.all(10),
      child: Text('GridView5'),
      color: Colors.red,
    ),
    Container(
      padding: EdgeInsets.all(10),
      child: Text('GridView5'),
      color: Colors.red,
    ),
  ],
),
);
}
}

```

Output:



#### D. ListView

ListView adalah widget scroll yang paling sering digunakan di Flutter. Widget ini memungkinkan menampilkan beberapa komponen atau widget melalui variabel children. Dalam contoh kali ini, akan digunakan ListView default dengan variabel children berupa `List<Widget>`. Cara menggunakannya adalah dengan menambahkan widget yang ingin diatur sebagai children dari ListView.

Source Code:

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

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

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Latihan List View Ade Fatkhul Anam'),
        backgroundColor: Colors.blueAccent[700],
      ),
      body: ListView(
        scrollDirection: Axis.horizontal,
        children: [
          Container(
            height: 100,
            width: 300,
            color: Colors.red,
            child: Center(child: Text("Type A")),
          ),
          Container(
            height: 100,
            width: 300,
            color: Colors.yellow,
            child: Center(child: Text("Type B")),
          ),
          Container(
            height: 100,
            width: 300,
            color: Colors.green,
            child: Center(child: Text("Type C")),
          ),
          Container(
            height: 100,
            width: 300,
            color: Colors.blue,
            child: Center(child: Text("Type D")),
          ),
        ],
      ),
    );
  }
}
```

Output :



## E. Stack

Widget ini memungkinkan tumpang tindih antara satu widget dengan widget lainnya. Misalnya, gambar dan teks yang ditumpuk bersama atau overlay yang mengandung tombol serta widget lainnya. Dengan **Stack**, kita bisa mengatur posisi widget secara bertumpuk di atas satu sama lain.

Source code:

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

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

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Latihan Stack'),
        backgroundColor: Colors.blueAccent[700],
      ),
      body: Stack(
        children: [
          Container(
            margin: EdgeInsets.all(5),
            height: 200,
            width: 200,
            color: Colors.red,
          ),
        ],
      ),
    );
  }
}
```

```
        Container(  
          margin: EdgeInsets.all(10),  
          height: 200,  
          width: 200,  
          color: Colors.yellow,  
        ),  
        Container(  
          margin: EdgeInsets.all(20),  
          height: 200,  
          width: 200,  
          color: Colors.green,  
        ),  
      ],  
    ),  
  );  
}
```

Output:



## F. Kesimpulan

Pada praktikum Modul 4 mengenai Antarmuka Pengguna dengan Flutter, mahasiswa diajarkan tentang konsep dasar pembuatan antarmuka menggunakan widget-widget Flutter. Melalui pemahaman dan implementasi widget seperti **GridView**, **ListView**, dan **Stack**, mahasiswa mampu membangun tampilan yang dinamis dan interaktif.

- **GridView** memungkinkan penataan elemen dalam format grid, cocok untuk menampilkan banyak item secara efisien.
- **ListView** digunakan untuk membuat daftar komponen yang dapat digulirkan, dengan opsi pengaturan arah scroll secara horizontal maupun vertikal.
- **Stack** memungkinkan penumpukan widget secara berlapis, memberikan fleksibilitas dalam penataan komponen secara visual.

Melalui berbagai contoh kode yang telah diterapkan, mahasiswa dapat lebih memahami cara kerja layout dan manipulasi widget di Flutter untuk membangun antarmuka pengguna yang menarik dan responsif.