	STMIK Widya Utama	Mobile Programming Lanjut
	Arsitektur Aplikasi Flutter	Job Sheet 04
Semester 6	Dosen : Amin Sholeh, M.Kom.	Hal: 1 dari 8

A. Tujuan

Setelah menyelesaikan job sheet ini, diharapkan mahasiswa mampu:

1. Memahami arsitektur pada aplikasi flutter
2. Menggunakan Widget
3. Menerapkan penggunaan Widget pada project yang dikerjakan

B. Petunjuk

1. Awali setiap aktivitas anda dengan doa, agar anda lancar dalam belajar
2. Kerjakan tugas-tugas praktikum dengan baik, teliti dan sabar
3. Tanyakan kepada dosen apabila ada hal-hal yang kurang jelas

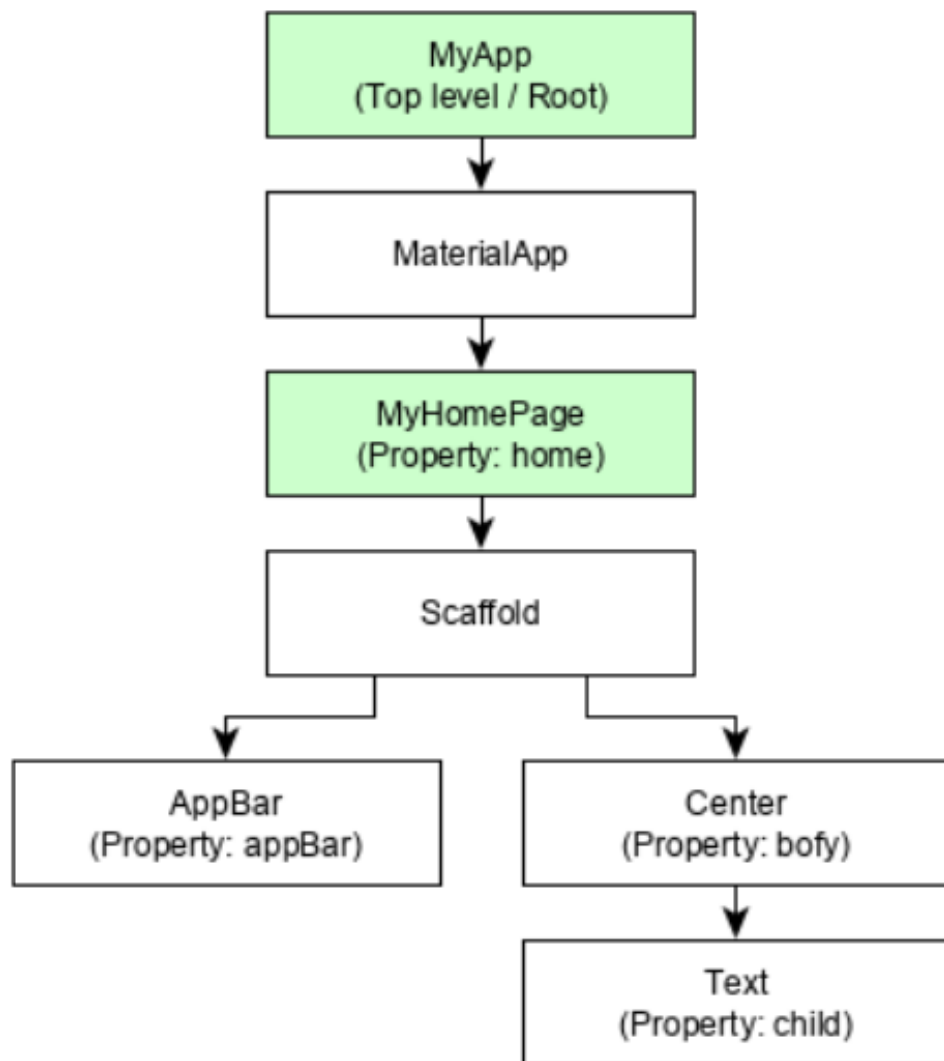
C. Dasar Teori

Konsep inti dari kerangka kerja Flutter adalah Dalam Flutter, Semuanya adalah widget. Widget pada dasarnya adalah komponen antarmuka pengguna yang digunakan untuk membuat antarmuka pengguna aplikasi.

Dalam Flutter, aplikasi itu sendiri adalah sebuah widget. Aplikasi adalah widget tingkat atas dan UI-nya dibangun menggunakan satu atau lebih anak (widget), yang sekali lagi dibangun menggunakan widget anak-anaknya. Fitur composability ini membantu kita untuk membuat antarmuka pengguna dengan kerumitan apapun.


Sebagai contoh, hirarki widget dari aplikasi hello world (yang dibuat pada bab sebelumnya) adalah seperti yang ditunjukkan pada diagram berikut:

	STMIK Widya Utama	Mobile Programming Lanjut
	Arsitektur Aplikasi Flutter	Job Sheet 04
Semester 6	Dosen : Amin Sholeh, M.Kom.	Hal: 2 dari 8



Berikut ini poin-poin patut diperhatikan:

- **MyApp** adalah widget yang dibuat oleh pengguna dan dibuat menggunakan widget asli Flutter, MaterialApp.
- **MaterialApp** memiliki properti home untuk menentukan antarmuka pengguna dari halaman beranda, yang sekali lagi merupakan widget buatan pengguna, MyHomePage.
- **MyHomePage** dibuat dengan menggunakan widget native Flutter lainnya, Scaffold.

	STMIK Widya Utama	Mobile Programming Lanjut
	Arsitektur Aplikasi Flutter	Job Sheet 04
Semester 6	Dosen : Amin Sholeh, M.Kom.	Hal: 3 dari 8

- **Scaffold** memiliki dua properti - body dan appBar.
- **body** digunakan untuk menentukan antarmuka pengguna utama dan appBar digunakan untuk menentukan antarmuka pengguna header.
- **Header UI** dibuat menggunakan widget native flutter, appBar dan Body UI dibuat menggunakan widget Center.
- **Widget Center** memiliki properti, Child, yang merujuk pada konten yang sebenarnya dan dibuat menggunakan widget Text.

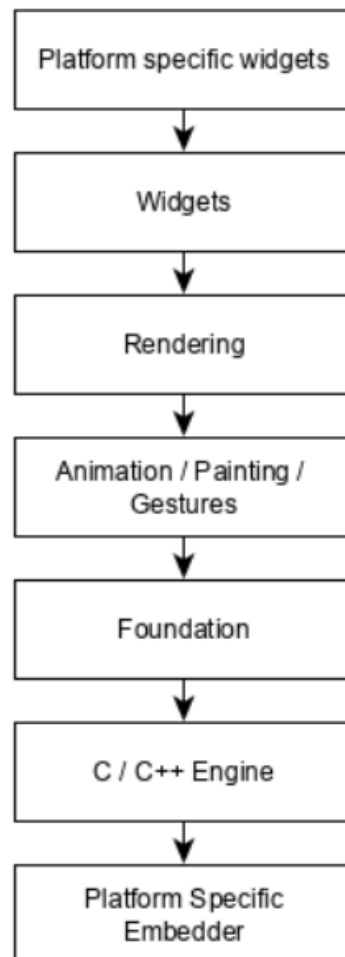
Widget Flutter mendukung interaksi melalui widget khusus, GestureDetector. **GestureDetector** adalah widget tak terlihat yang memiliki kemampuan untuk menangkap interaksi pengguna seperti mengetuk, menyeret, dll., dari widget turunannya. Banyak widget asli Flutter yang mendukung interaksi melalui penggunaan GestureDetector. Kita juga dapat memasukkan fitur interaktif ke dalam widget yang sudah ada dengan mengkomposisikannya dengan widget GestureDetector. Kita akan mempelajari gerakan secara terpisah dalam bab-bab mendatang.

Widget Flutter mendukung pemeliharaan Status dengan menyediakan widget khusus, **StatefulWidget**. Widget harus diturunkan dari widget StatefulWidget untuk mendukung pemeliharaan state dan semua widget lainnya harus diturunkan dari StatefulWidget. Widget Flutter bersifat reaktif secara native. Hal ini mirip dengan reactjs dan StatefulWidget akan di-render ulang secara otomatis setiap kali state internalnya berubah. Perenderan ulang dioptimalkan dengan menemukan perbedaan antara UI widget lama dan baru dan hanya merender perubahan yang diperlukan.

Konsep yang paling penting dari framework Flutter adalah bahwa kerangka kerja ini dikelompokkan ke dalam beberapa kategori dalam hal kompleksitas dan disusun dengan jelas dalam lapisan-lapisan (layer-layer) yang semakin lama semakin berkurang kompleksitasnya. Sebuah lapisan dibangun menggunakan lapisan di bawahnya. Lapisan paling atas adalah widget khusus untuk Android dan iOS. Lapisan berikutnya memiliki semua widget asli flutter. Lapisan berikutnya adalah lapisan Rendering, yang merupakan komponen renderer tingkat rendah dan merender semua yang ada di aplikasi flutter. Lapisan turun ke kode spesifik platform inti.

Gambaran umum lapisan dalam Flutter ditentukan dalam diagram di bawah ini:

	STMIK Widya Utama	Mobile Programming Lanjut
	Arsitektur Aplikasi Flutter	Job Sheet 04
Semester 6	Dosen : Amin Sholeh, M.Kom.	Hal: 4 dari 8



Poin-poin berikut ini merangkum arsitektur Flutter:

- Dalam Flutter, semuanya adalah widget dan widget yang kompleks terdiri dari widget yang sudah ada.
- Fitur interaktif dapat dimasukkan kapan pun diperlukan menggunakan widget GestureDetector.
- Status widget dapat dipertahankan kapan pun diperlukan menggunakan widget StatefulWidget.
- Flutter menawarkan desain berlapis sehingga setiap lapisan dapat diprogram tergantung pada kompleksitas tugas.

	STMIK Widya Utama	Mobile Programming Lanjut
	Arsitektur Aplikasi Flutter	Job Sheet 04
Semester 6	Dosen : Amin Sholeh, M.Kom.	Hal: 5 dari 8

D. Latihan

StatelessWidget adalah class widget yang propertinya immutable, artinya nilainya tidak bisa diubah. Cara membuat StatelessWidget adalah dengan membuat class turunan (extends) dari class StatelessWidget.

Contoh:

```
class MyApp extends StatelessWidget { }
```

Mari kita coba di dalam aplikasi.

Silahkan buka file **lib/main.dart** kemudian hapus semua isinya dan ganti dengan kode ini:


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

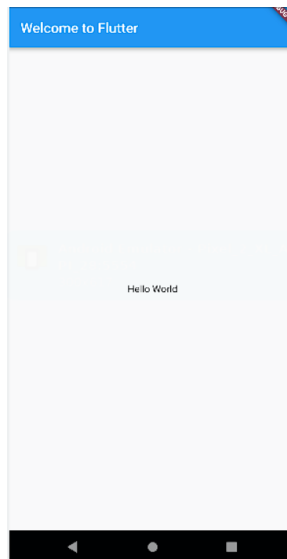
void main() => runApp(MyApp());

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'First App',
      home: Scaffold(
        appBar: AppBar(
          title: Text('Welcome to Flutter'),
        ),
        body: Center(
          child: Text('Hello World'),
        ),
      ),
    );
  }
}
```

Setelah itu, coba jalankan aplikasi di emulator.

Maka hasilnya:

	STMIK Widya Utama	Mobile Programming Lanjut
	Arsitektur Aplikasi Flutter	Job Sheet 04
Semester 6	Dosen : Amin Sholeh, M.Kom.	Hal: 6 dari 8



Penjelasan detailnya seperti berikut :

```

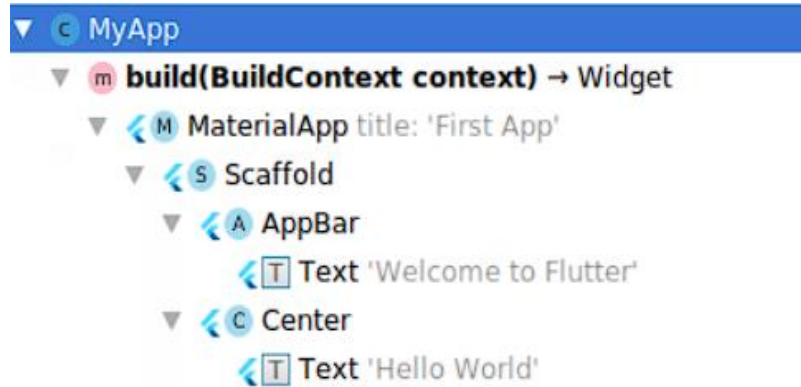
1  import 'package:flutter/material.dart'; ← bagian import
2
3  void main() => runApp(MyApp()); ← bagian main
4
5  class MyApp extends StatelessWidget {
6    @override
7    Widget build(BuildContext context) {
8      return MaterialApp( ← judul aplikasi untuk recent app
9        title: 'First App',
10       home: Scaffold(
11         appBar: AppBar( ← judul untuk ditampilkan di AppBar
12           title: Text('Welcome to Flutter'),
13         ), // AppBar
14         body: Center(
15           child: Text('Hello World'), ← Teks untuk ditampilkan pada body
16         ), // Center
17       ), // Scaffold
18     ); // MaterialApp
19   }
20 }

```

	STMIK Widya Utama	Mobile Programming Lanjut
	Arsitektur Aplikasi Flutter	Job Sheet 04
Semester 6	Dosen : Amin Sholeh, M.Kom.	Hal: 7 dari 8

Pada aplikasi di atas, kita membuat StatelessWidget yang berisi widget **MaterialApp()**. Kemudian di dalam **MaterialApp()** berisi widget lagi: **Scaffold**, **AppBar**, **Center**, dan **Text**.

Sehingga strukturnya akan terlihat seperti ini:



Penjelasan:

- **MyApp** adalah StatelessWidget, merupakan widget induk;
- **MaterialApp** adalah widget yang membungkus beberapa widget yang menggunakan tema material design 1;
- **Scaffold** adalah widget untuk struktur dasar material design;
- **AppBar** adalah widget untuk membuat AppBar;
- **Center** adalah Widget layout untuk membuat widget ke tengah;
- **Text** adalah widget untuk membuat teks.

	STMIK Widya Utama	Mobile Programming Lanjut
	Arsitektur Aplikasi Flutter	Job Sheet 04
Semester 6	Dosen : Amin Sholeh, M.Kom.	Hal: 8 dari 8

E. Tugas Praktikum

1. Buatlah sebuah project flutter untuk mengubah warna background dan teks seperti dibawah ini!
2. Unggah project tersebut di github!



Kirimkan link repository tugas ke tautan berikut: <https://s.id/kelasflutterswu>