

Materi: Widget Dasar Flutter

Tujuan Pembelajaran

Setelah mempelajari materi ini, mahasiswa mampu:

1. Memahami konsep dasar widget di Flutter.
 2. Menggunakan widget utama seperti **Text**, **Image**, **Icon**, **Container**, **Scaffold**, dan **AppBar**.
 3. Membuat tampilan aplikasi sederhana dengan kombinasi widget dasar.
-

1. Konsep Dasar Widget di Flutter

Di Flutter, **semua elemen tampilan adalah widget** — mulai dari teks, gambar, tombol, hingga layout.

Widget dibagi menjadi dua jenis utama:

1. **Stateless Widget** → Tampilan statis, tidak berubah selama aplikasi berjalan.
2. **Stateful Widget** → Tampilan yang bisa berubah (dinamis) saat berinteraksi dengan user.

Contoh:

- Text, Icon, Image → widget statis
 - Checkbox, Slider, Form → widget dinamis (stateful)
-

2. Widget Dasar

Mari kita pelajari satu per satu 📌

Widget Text

Menampilkan teks di layar.

```
Text(  
  'Halo Flutter!',  
  style: TextStyle(  
    fontSize: 24,  
    fontWeight: FontWeight.bold,  
    color: Colors.blue,  
  ),  
)
```

)

Keterangan:

- `fontSize` → ukuran huruf
 - `fontWeight` → ketebalan (bold, normal)
 - `color` → warna teks
-

Widget Image

Menampilkan gambar dari internet atau dari file lokal (assets).

```
Image.network(  
  'https://flutter.github.io/assets-for-api-docs/assets/widgets/owl.jpg',  
  width: 150,  
  height: 150,  
  fit: BoxFit.cover,  
)
```

Untuk gambar lokal:

- Simpan di folder `/assets/images/`
- Tambahkan ke `pubspec.yaml`:

```
flutter:  
  assets:  
    - assets/images/
```

Kemudian gunakan:

```
Image.asset('assets/images/logo.png')
```

Widget Icon

Menampilkan ikon bawaan Flutter (Material Icons).

```
Icon(  
  Icons.favorite,  
  color: Colors.red,  
  size: 40,  
)
```

Paket ikon sudah bawaan Flutter, tidak perlu import tambahan.

Widget Container

Digunakan untuk membungkus widget lain dan memberikan **padding, margin, border, atau warna latar**.

```
Container(  
  padding: EdgeInsets.all(16),  
  margin: EdgeInsets.all(12),  
  decoration: BoxDecoration(  
    color: Colors.blue.shade100,  
    borderRadius: BorderRadius.circular(12),  
    border: Border.all(color: Colors.blue, width: 2),  
  ),  
  child: Text(  
    'Ini di dalam Container',  
    style: TextStyle(fontSize: 18),  
  ),  
)
```

Widget Scaffold dan AppBar

Scaffold adalah **kerangka dasar halaman** dalam aplikasi Flutter. Biasanya berisi:

- AppBar (judul di atas)
- body (isi halaman)
- floatingActionButton, drawer, bottomNavigationBar, dll.

```
Scaffold(  
  appBar: AppBar(  
    title: Text('Belajar Widget Dasar'),  
    backgroundColor: Colors.blue,  
  ),  
  body: Center(  
    child: Text('Halo Dunia Flutter!'),  
  ),  
)
```

3. Contoh Kasus Program Sederhana

Aplikasi Profil Sederhana

Nama file: main.dart

```
import 'package:flutter/material.dart';  
  
void main() {  
  runApp(const MyApp());  
}
```

```

}

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Aplikasi Profil Flutter',
      debugShowCheckedModeBanner: false,
      home: const HomePage(),
    );
  }
}

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

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Profil Saya'),
        backgroundColor: Colors.teal,
        centerTitle: true,
      ),
      body: Center(
        child: Container(
          padding: const EdgeInsets.all(20),
          margin: const EdgeInsets.all(20),
          decoration: BoxDecoration(
            color: Colors.teal.shade50,
            borderRadius: BorderRadius.circular(20),
            boxShadow: [
              BoxShadow(
                color: Colors.grey.withOpacity(0.5),
                spreadRadius: 4,
                blurRadius: 7,
                offset: const Offset(0, 3),
              ),
            ],
          ),
        child: Column(
          mainAxisAlignment: MainAxisAlignment.min,
          children: const [
            CircleAvatar(
              radius: 50,
              backgroundImage: NetworkImage(
                'https://flutter.github.io/assets-for-api-
docs/assets/widgets/owl.jpg',
              ),
            ),
            SizedBox(height: 15),
            Text(
              'Fajar Budi Hartono',
              style: TextStyle(
                fontSize: 22,

```

```

        fontWeight: FontWeight.bold,
        color: Colors.teal,
      ),
    ),
    SizedBox(height: 8),
    Text(
      'Mobile Developer | Flutter Enthusiast',
      style: TextStyle(fontSize: 16, color: Colors.black54),
    ),
    SizedBox(height: 15),
    Row(
      mainAxisAlignment: MainAxisAlignment.center,
      children: [
        Icon(Icons.phone, color: Colors.green),
        SizedBox(width: 5),
        Text('0812-3456-7890'),
      ],
    ),
    SizedBox(height: 10),
    Row(
      mainAxisAlignment: MainAxisAlignment.center,
      children: [
        Icon(Icons.email, color: Colors.red),
        SizedBox(width: 5),
        Text('fajar@example.com'),
      ],
    ),
  ],
),
),
);
}
}

```

Hasil Tampilan (Konsep Visual)

```

+-----+
| AppBar: Profil Saya |
+-----+
|          (Foto)          |
|      Fajar Budi Hartono  |
| Mobile Developer | Flutter Enthusiast |
| 📞 0812-3456-7890 |
| ✉️ fajar@example.com |
+-----+

```

Latihan untuk Mahasiswa

1. Ubah warna tema aplikasi (AppBar, background).
2. Tambahkan `Icon` media sosial (Facebook, Instagram).

3. Ganti `Image.network` menjadi `Image.asset` (gunakan gambar lokal).
4. Buat versi kedua dengan background gradient.

Dalam **Flutter**, **widget** adalah **komponen dasar** yang digunakan untuk membangun tampilan (UI) sebuah aplikasi.

Bisa dibilang, **semua yang kamu lihat di layar Flutter adalah widget** — mulai dari teks, tombol, gambar, hingga layout dan struktur halaman itu sendiri.

Pengertian Widget

Secara sederhana:

Widget adalah elemen antarmuka pengguna (UI element) dalam Flutter yang mendeskripsikan bagian dari tampilan aplikasi.

Flutter menggunakan **arsitektur deklaratif**, artinya kamu *mendeskripsikan seperti apa tampilan yang diinginkan*, dan Flutter yang akan menampilkan hasilnya.

Contoh:

```
Text('Halo Dunia')
```

Widget ini memberitahu Flutter untuk menampilkan tulisan **Halo Dunia** di layar.

Jenis-jenis Widget

Secara umum, widget di Flutter terbagi menjadi dua jenis utama:

1. StatelessWidget

- Tidak memiliki data yang berubah (state tetap).
- Contoh: Text, Icon, Container, AppBar.

Contoh kode:

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

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

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        appBar: AppBar(title: Text('Contoh StatelessWidget')),
        body: Center(
          child: Text('Halo Dunia!'),
        ),
      ),
    );
  }
}
```

```

    ),
  );
}
}

```

2. StatefulWidget

- Memiliki data yang bisa berubah saat aplikasi berjalan (memiliki *state*).
- Contoh: Checkbox, TextField, Slider, Switch.

Contoh kode:

```

import 'package:flutter/material.dart';

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

class MyApp extends StatefulWidget {
  @override
  State<MyApp> createState() => _MyAppState();
}

class _MyAppState extends State<MyApp> {
  int counter = 0;

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        appBar: AppBar(title: Text('Contoh StatefulWidget')),
        body: Center(
          child: Column(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
              Text('Anda menekan tombol sebanyak: $counter kali'),
              SizedBox(height: 10),
              ElevatedButton(
                onPressed: () {
                  setState(() {
                    counter++;
                  });
                },
                child: Text('Tekan Saya'),
              ),
            ],
          ),
        ),
      ),
    );
  }
}

```

Kesimpulan

Aspek	StatelessWidget	StatefulWidget
Perubahan Data	Tidak bisa berubah	Bisa berubah
Contoh	Text, Icon, Container	Checkbox, Slider, Form
Kapan Digunakan	Tampilan statis	Tampilan yang interaktif / berubah