

MODUL 1

LAYOUT DAN NAVIGATION

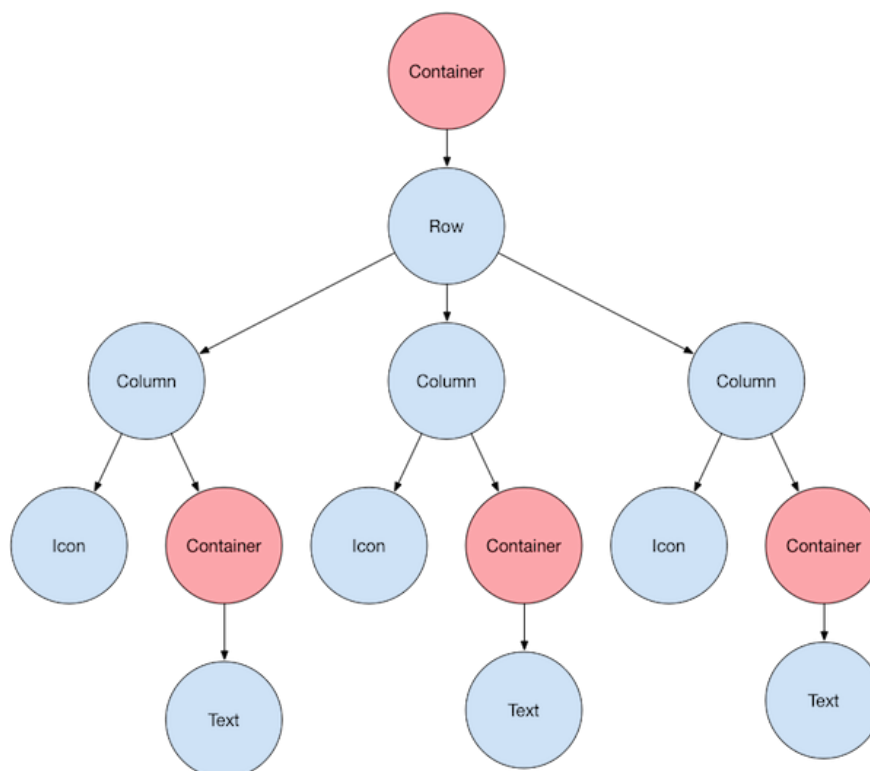
1. Tujuan

- Memahami konsep dasar layout menggunakan *ListView* dan *GridView*.
- Mengimplementasikan tampilan daftar *vertikal*, *horizontal*, dan *grid* dengan efisien.
- Menerapkan navigasi antar halaman menggunakan *Navigator.push* dan *Navigator.pop*.
- Mengirim dan menerima data antar halaman dalam navigasi *Flutter*.

2. Dasar Teori

2.1. Layout

Layout di *Flutter* menggunakan sistem berbasis *widget*. Inti dari mekanisme *layout* *Flutter* adalah *widget*, yang menyusun semua elemen visual dan struktural. Gambar, ikon, dan teks yang ditampilkan dalam aplikasi *Flutter* semuanya merupakan *widget*. Bahkan elemen yang tidak langsung terlihat, seperti *Row*, *Column*, dan *Grid*, juga merupakan *widget* yang bertugas mengatur, menyelaraskan, dan membatasi tampilan elemen-elemen lainnya.



2.1.1. ListView

Menampilkan daftar data adalah pola umum dalam pengembangan *mobile apps*. *Flutter* menyediakan *widget ListView* untuk menyusun dan menampilkan daftar item secara fleksibel dan *scrollable*.

a. *ListView*Sederhana

ListView sederhana digunakan ketika jumlah item tidak terlalu banyak dan diketahui secara statis. Item ditentukan secara langsung dalam daftar *children*.

Contoh Kode :

```
ListView(  
  children: [  
    ListTile(title: Text('Item 1')),  
    ListTile(title: Text('Item 2')),  
    ListTile(title: Text('Item 3')),  
  ],  
)
```

Tambah gambar

b. *ListView.builder*

ListView.builder digunakan untuk menampilkan daftar dengan jumlah item yang dinamis atau sangat banyak. List ini hanya merender elemen yang terlihat di layar, sehingga lebih efisien dalam performa.

Contoh kode :

```
ListView.builder(  
  itemCount: 10,  
  itemBuilder: (context, index) {  
    return ListTile(  
      title: Text('Item \'$index\''),  
    );  
  },  
)
```

c. *Horizontal ListView*

Horizontal ListView adalah varian dari *ListView* di mana *scroll* dilakukan secara *horizontal*. Ini cocok untuk menampilkan elemen-elemen seperti gambar, kategori, atau produk.

Contoh kode :

```
ListView.builder(  
  scrollDirection: Axis.horizontal,  
  itemCount: 5,  
  itemBuilder: (context, index) {
```

```

return Container(
  width: 100,
  margin: EdgeInsets.all(8),
  color: Colors.blue,
  child: Center(child: Text('Item \${index}')),
);
},
)

```

2.1.2. *GridView*

GridView adalah *widget layout* yang digunakan untuk menampilkan elemen-elemen dalam bentuk grid atau kotak-kotak. *GridView* cocok digunakan untuk menampilkan gambar, produk, menu ikon, dan konten visual lainnya yang membutuhkan penyusunan dalam baris dan kolom.

Kasus penggunaan umum *GridView*:

- Galeri foto
- Menu aplikasi dengan ikon
- Tampilan produk dalam *e-commerce*

Contoh kode menggunakan *GridView.Count()*:

```

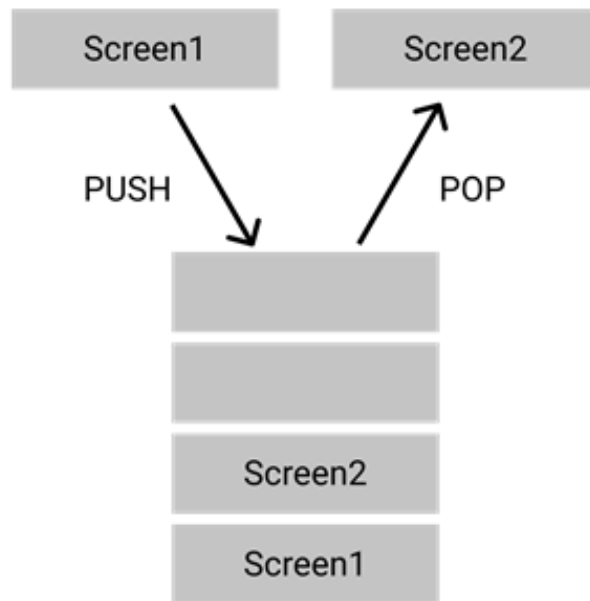
GridView.count(
  crossAxisCount: 2,
  children: List.generate(6, (index) {
    return Container(
      margin: EdgeInsets.all(8),
      color: Colors.teal,
      child: Center(child: Text('Item \${index}')),
    );
  })),
)

```

2.2. *Navigation*

Navigasi dalam Flutter adalah proses berpindah dari satu halaman (*screen*) ke halaman lainnya. Dalam Flutter, halaman biasanya diwakili oleh *Widget*, dan navigasi dikelola oleh *Navigator*.

Navigator bekerja seperti tumpukan (*stack*). Setiap kali kita menavigasi ke halaman baru, halaman tersebut ditumpuk (*push*) ke atas *stack*. Ketika kembali ke halaman sebelumnya, halaman teratas diambil keluar (*pop*) dari *stack*. Hal ini cocok dengan pola navigasi pada aplikasi mobile modern.



a. *Navigator.push()*

Digunakan untuk berpindah ke halaman baru. Halaman baru akan ditambahkan ke atas stack, dan halaman sebelumnya tetap ada di bawahnya.

Contoh Kode :

```
Navigator.push(  
  context,  
  MaterialPageRoute(builder: (context) => SecondPage() ),  
);
```

b. *Navigator.pop()*

Digunakan untuk kembali ke halaman sebelumnya dengan cara menghapus halaman saat ini dari atas stack.

Contoh Kode :

```
Navigator.pop(context);
```

3. Latihan

Ikuti langkah-langkah berikut!

- 1) Clone repository pada link berikut <https://bit.ly/latprak-modul-1>
- 2) Lengkapi kode pada penggalan yang telah tersedia di file latihan sesuai dengan ketentuan yang terdapat pada file README.md!

3) Dikumpulkan 2x24 jam di github! (*optional*)

4. Tugas Praktikum

Buatlah *mobile apps* menggunakan *layout* dan *navigation* seperti *design* figma (*high fidelity*) pada *link* berikut : <https://bit.ly/tuprak-modul-1>