

## MINGGU 2: STYLE, TEXT, ICON, APPBAR, DAN LAYOUT

<b>Pokok Bahasan</b>	<b>: Style, Text, Icon, AppBar, dan Layout</b>
<b>Minggu</b>	<b>: 2</b>
<b>Tempat</b>	<b>: Daring / Luring Politeknik Negeri Jember</b>
<b>Alokasi Waktu</b>	<b>: 100 menit x 4</b>

### **a. Capaian Pembelajaran Mata Kuliah (CPMK)**

1. Mahasiswa mampu memahami konsep Style, Text, Icon, AppBar, dan Layout dalam Flutter
2. Mahasiswa mampu menerapkan Style, Text, Icon, AppBar, dan Layout dalam Flutter

### **b. Indikator**

Kemampuan mahasiswa dalam memahami dan menerapkan Style, Text, Icon, AppBar, dan Layout dalam Flutter

### **c. Dasar Teori**

Flutter widget digunakan dalam pengaturan style, text, dan icon. Flutter widget dibuat menggunakan framework modern yang mengambil inspirasi dari **React**. Ide utamanya adalah membangun User Interface dari widget. Widget menjelaskan seperti apa tampilan jika diberikan konfigurasi dan status saat ini. Saat status widget berubah, widget akan membuat ulang deskripsinya, yang mana kerangka kerja berbeda dengan deskripsi sebelumnya untuk menentukan perubahan minimal yang diperlukan dalam pohon render yang mendasari untuk transisi dari satu status ke status berikutnya. Flutter hadir dengan rangkaian widget dasar yang powerful, dimana widget dasar yang biasanya digunakan sebagai berikut:

1. Text. Widget untuk membuat serangkaian teks bergaya dalam aplikasi Anda.
2. Column, Row. Widget fleksibel ini untuk membuat tata letak yang fleksibel baik dalam arah horizontal (Baris) dan vertikal (Kolom). Desain objek ini didasarkan pada model tata letak flexbox web.
3. Stack. Daripada menggunakan orientasi linier (baik horizontal atau vertikal), widget Stack dapat digunakan untuk menempatkan widget di atas satu sama lain dalam urutan cat. kemudian dapat menggunakan widget diposisikan pada children dari Stack untuk

memposisikannya relatif terhadap tepi atas, kanan, bawah, atau kiri stack. Stack didasarkan pada model tata letak pemosisian absolut web.

4. Container. Widget ini untuk membuat elemen visual persegi panjang. Container dapat didekorasi dengan BoxDecoration, seperti background, border, atau shadow. Container juga dapat memiliki margin, padding, dan constraint yang diterapkan dengan ukuran tertentu. Selain itu, container dapat juga ditransformasikan dalam ruang tiga dimensi seperti menggunakan matriks.

#### d. Alat dan Bahan

1. Java Development Kit (JDK)
2. Android Studio;
3. Android SDK;
4. Flutter SDK;
5. Teks Editor (atau bisa juga pakai Android Studio atau Visual Studio Code).

#### e. Prosedur Kerja

1. Pada IDE **Visual Studio Code** atau **Android Studio**, buatlah project baru dengan nama **'style\_appbar\_layout'**.
2. Hapus semua code default yang ada pada file **lib/main.dart**, dan ketikan code di bawah ini

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

void main() {
  runApp(MaterialApp(
    home: firstPage(),
  ));
}

class firstPage extends StatelessWidget{
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: Container(
        child: Center(
          child: Text('Belajar Flutter'),
        ),
      ),
    );
    throw UnimplementedError();
  }
}
```

3. Hasil running dari code tersebut hanya berupa text “Belajar Flutter” yang terletak di tengah layar/screen. Tambahkan code berikut di dalam **body: Container** untuk menambahkan style seperti background color, width, dan height.

```
color: Colors.blue[900],
width: 200.0,
height: 100.0,
```

4. Sedangkan untuk menambahkan style untuk text, ubah baris code untuk **child: Text** menjadi seperti berikut ini.

```
child: Text('Belajar Flutter', style: TextStyle(color: Colors.black,
fontFamily: "Serif", fontSize: 24.0),
```

5. Selain text, pada flutter juga bisa dengan simple menambahkan icon dalam layout atau tampilannya. Untuk menampilkan icon ke dalam layout, ganti **child: Text** menjadi **child: Icon** kemudian tambahkan property seperti pada code berikut ini.

```
child: Icon(Icons.android, color: Colors.green, size: 70.0),
```

6. AppBar dibuat dengan menggunakan widget appBar. Tambahkan baris code berikut ke dalam scaffold.

```
appBar: AppBar(
  backgroundColor: Colors.blue[500],
  leading: Icon(Icons.home_filled),
  title: Text("ini AppBar"),
),
```

7. Di dalam AppBar juga bisa ditambahkan Action, sebagai contoh tambahkan code berikut di dalam widget AppBar.

```
actions: <Widget>[Icon(Icons.search)],
```

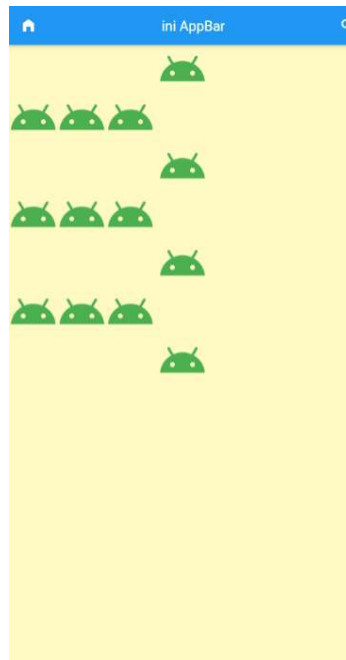
8. Untuk pengaturan layout dapat menggunakan widget Row dan Colomn. Tambahkan code berikut ke dalam **body: Container** untuk menggunakan widget Row. Dimana di dalam widget Row terdapat children untuk widget Icon sebelumnya.

```
child: Row(children: <Widget>[
  Icon(
    Icons.android,
    color: Colors.green,
    size: 70.0,
  )
```

9. Widget Icon yang menjadi children di dalam widget Row bisa di copy-paste lagi 2 kali di bawah sehingga terdapat 3 icon android yang lebih dari satu dalam satu barisnya.
10. Gantilah widget Row sebelumnya menjadi widget Colomn.
11. Perhatikan setiap perubahan hasil running dari masing-masing langkah prosedur kerja.

## f. Hasil dan Pembahasan

1. Buatlah hasil dari prosedur kerja selama praktikum menjadi seperti gambar berikut



2. Dokumentasi tugas berupa laporan (penjelasan mengenai potongan source-code Style, Text, Icon, AppBar, dan Layout).
3. Dokumentasi tugas berupa file pdf.

**g. Kesimpulan**

Mahasiswa memahami mengenai penerapan Style, Text, Icon, AppBar, dan Layout dalam flutter.

**h. Rubrik Penilaian**

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menjelaskan dari tugas ditunjang dengan bukti referensi	1	2	3	④
2	Ketepatan waktu dan ketepatan dalam menjelaskan dari tugas	1	2	③	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan tugas	1	②	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan tugas	①	2	3	4
Jumlah skor					