

**LAPORAN PRAKTIKUM MOBILE PROGRAMMING**  
**MODUL 2**



Nama : Firman Fadilah Noor

NIM : 240605110083

Kelas : B

Tanggal : 10 September 2025

**JURUSAN TEKNIK INFORMATIKA**  
**FAKULTAS SAINS DAN TEKNOLOGI**  
**UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM**  
**MALANG**  
**GANJIL 2025/2026**

## I. Tujuan

1. Memahami konsep widget Row dan Column dalam Flutter.
2. Membuat dan menggunakan widget kustom (custom widget).
3. Mengatur tata letak widget secara horizontal dan vertical.
4. Menampilkan widget menggunakan properti visual seperti warna, border, dan icon.

## II. Langkah Kerja

1. Membuat proyek Flutter baru.
2. Menambahkan kode *row* dan *column*
3. Menjalankan aplikasi pada *emulator* dan *smartphone*
4. Pengujian

## III. Screenshot Hasil

- a. Kode Program

## IV. Kesimpulan

```
1 import 'package:flutter/material.dart';
2
3 void main() {
4   runApp(const MyApp());
5 }
6
7 class MyApp extends StatelessWidget {
8   const MyApp({super.key});
9   @override
10  Widget build(BuildContext context) {
11    return MaterialApp(
12      title: 'Row and Column',
13      home: Scaffold(
14        appBar: AppBar(
15          title: const Text('Row and Column'),
16        ), // AppBar AppBar
17        body: Center(
18          child: Column(
19            mainAxisAlignment: MainAxisAlignment.center,
20            children: [
21              Row(
22                mainAxisAlignment: MainAxisAlignment.center,
23                children: const [
24                  kotakIkon(
25                    warnaKotak: Colors.orange,
26                    icon: Icons.thumb_up,
27                    warnaIcon: Colors.blue,
28                    label: "Like",
29                  ), // kotakIkon kotakIkon
30                  SizedBox(width: 30),
31                  kotakIkon(
32                    warnaKotak: Colors.blue,
33                    icon: Icons.favorite,
34                    warnaIcon: Colors.red,
```

```
35          label: "Love",
36        ), kotakIkon
37      ],
38    ), // Row Row
39    SizedBox(height: 30),
40    Row(
41      mainAxisAlignment: MainAxisAlignment.center,
42      children: const [
43        kotakIkon(
44          warnaKotak: Colors.red,
45          icon: Icons.star,
46          warnaIcon: Colors.yellow,
47          label: "Stars",
48        ), // kotakIkon kotakIkon
49        SizedBox(width: 30),
50        kotakIkon(
51          warnaKotak: Colors.black,
52          icon: Icons.comment,
53          warnaIcon: Colors.white,
54          label: "Comment",
55        ), // kotakIkon kotakIkon
56      ],
57    ) // Row Row
58  ],
59 ), // Column Column
60 ), // Center Center
61 ), // Scaffold Scaffold
62 ); // MaterialApp MaterialApp
63 }
64
65
66 class kotakIkon extends StatelessWidget {
67   final Color warnaKotak;
68   final IconData icon;
```

```

69 final Color warnaIcon;
70 final String label;
71
72 const kotakIcon({
73   super.key,
74   required this.warnaKotak,
75   required this.icon,
76   required this.warnaIcon,
77   required this.label,
78 });
79
80 @override
81 Widget build(BuildContext context) {
82   return Column(
83     children: [
84       Container(
85         width: 100,
86         height: 100,
87         decoration: BoxDecoration(
88           color: warnaKotak,
89           border: Border.all(color: Colors.black, width: 3),
90           borderRadius: BorderRadius.circular(20),
91         ), // BoxDecoration BoxDecoration
92         child: Icon(
93           icon,
94           color: warnaIcon,
95           size: 50,
96         ), // Icon Icon
97       ), // Container Container
98       Text(
99         label,
100         style: const TextStyle(fontWeight: FontWeight.bold),
101       ), // Text Text
102     ],
103   );

```

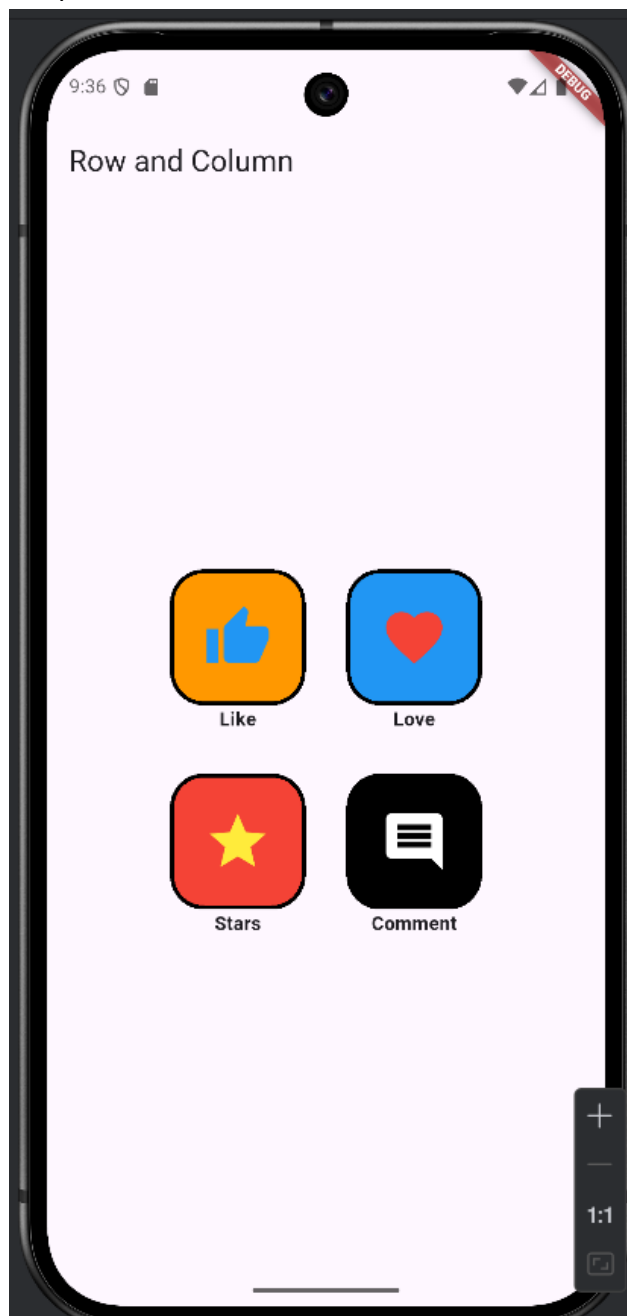
## b. Penjelasan Kode Program

Program Flutter di atas adalah sebuah aplikasi sederhana yang menampilkan kombinasi **Row** dan **Column** untuk mengatur tata letak widget secara terstruktur. Pada fungsi `main()`, aplikasi dijalankan melalui `runApp(MyApp())` yang memanggil widget utama `MyApp` sebagai `StatelessWidget`. Di dalamnya terdapat `MaterialApp` dengan judul "Row and Column", dan halaman utama menggunakan `Scaffold` yang memiliki `AppBar` serta `body`. Bagian `body` memanfaatkan `Center` untuk memposisikan isi ke tengah layar, lalu menggunakan `Column` agar elemen tersusun secara vertikal. Di dalam `Column` terdapat dua `Row`, masing-masing berisi beberapa widget kustom bernama `kotakIcon`.

Widget `kotakIcon` sendiri dibuat sebagai sebuah `StatelessWidget` terpisah, yang menerima parameter warna kotak, ikon, warna ikon, serta label teks. Di dalamnya, widget ini menampilkan `Container` berbentuk kotak dengan ukuran 100x100 piksel, berwarna sesuai properti `warnaKotak`, memiliki border hitam, serta sudut membulat. Di tengah kotak terdapat sebuah ikon (`Icon`) dengan warna dan ukuran tertentu. Di bawah kotak, ditambahkan teks (`Text`) yang menampilkan label dari ikon tersebut. Dengan pendekatan ini, setiap kotak ikon seperti "Like", "Love", "Stars", dan "Comment" dapat dengan mudah dibuat ulang hanya dengan memanggil widget `kotakIcon` dan memberikan nilai parameternya.

Secara keseluruhan, kode ini memperlihatkan cara menggunakan **Row** dan **Column** untuk mengatur layout, sekaligus mencontohkan praktik membuat **widget kustom** yang dapat digunakan berulang agar kode lebih rapi, modular, dan mudah dikembangkan.

c. Output



#### **IV. Kesimpulan**

Berdasarkan hasil praktikum pada modul 2, dapat disimpulkan bahwa penggunaan **Row** dan **Column** dalam Flutter sangat penting untuk mengatur tata letak widget baik secara horizontal maupun vertikal. Melalui implementasi kode program yang disertai dengan pembuatan widget kustom (kotakikon), mahasiswa dapat memahami bagaimana cara menyusun elemen-elemen antarmuka agar lebih terstruktur, rapi, dan mudah digunakan kembali. Selain itu, praktikum ini juga memberikan pemahaman tentang pemanfaatan properti visual seperti warna, border, ikon, dan teks untuk memperindah tampilan aplikasi. Dengan demikian, praktikum ini berhasil memperkuat pemahaman konsep dasar layout Flutter sekaligus melatih keterampilan membuat antarmuka aplikasi yang modular dan efisien.