

**LAPORAN PRAKTIKUM MOBILE PROGRAMMING**  
**MODUL 4**



Nama : Mauluda Aminaditya

NIM : 240605110132

Kelas : B

Tanggal : 15 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 Flexible dan Expanded dalam Flutter.
2. Membedakan perilaku flex dan fit pada widget Flexible.
3. Membandingkan perbedaan widget Flexible dan Expanded dalam mengisi ruang kosong pada tata letak.

## **II. Langkah Kerja**

1. Membuat proyek Flutter tema gelap.
2. Menambahkan struktur utama aplikasi menggunakan Scaffold.
3. Menambahkan control bar bagian bawah layar.
4. Membuat barisan tombol kontrol musik bentuk row.
5. Menyusun 5 ikon kontrol (Shuffle, Previous, Play, Next, dan Repcat).
6. Mengatur 4 ikon samping agar ukurannya seimbang.
7. Membuat ikon play yang lebih besar.
8. Memberikan warna putih pada semua ikon agar kontras.

## **III. Screenshot Hasil**

- a. Hasil di Samrtphone

The screenshot shows a code editor window for a Flutter application named 'main.dart'. The code is written in Dart and defines a dark-themed Material App with a Music Player UI.

```
alignment > lib > main.dart > ...
1 import 'package:flutter/material.dart';
2
3 Run | Debug | Profile
4 void main() {
5   runApp(const MyApp());
6 }
7
8 class MyApp extends StatelessWidget {
9   const MyApp({super.key});
10
11   @override
12   Widget build(BuildContext context) {
13     return MaterialApp(
14       theme: ThemeData.dark(),
15       home: const MusicPlayerUI(),
16     ); // MaterialApp
17 }
18
19 class MusicPlayerUI extends StatelessWidget {
20   const MusicPlayerUI({super.key});
21
22   @override
23   Widget build(BuildContext context) {
24     return Scaffold(
25       appBar: AppBar(
26         title: const Text("Music Control Bar"),
27       ), // AppBar
28
29       // ===== CONTROL BAR ADA DI BAWAH LAYAR =====
30       bottomNavigationBar: Container(
31         padding: const EdgeInsets.symmetric(horizontal: 8, vertical: 10),
32         color: Colors.black54,
33
34         child: Row(
35           children: [
36             // 1. Shuffle
37             const Expanded(
38               child: Icon(Icons.shuffle, color: Colors.white, size: 30),
39             ), // Expanded
40           ],
41         ),
42       ),
43     );
44   }
45 }
```

```
41     // 2. Previous
42     const Expanded(
43       child: Icon(Icons.skip_previous, color: Colors.white, size: 30),
44     ), // Expanded
45
46     // 3. Play (lebih besar)
47     Flexible(
48       flex: 2,
49       fit: FlexFit.tight,
50       child: const Icon(
51         Icons.play_circle_fill,
52         color: Colors.white,
53         size: 45,
54       ), // Icon
55     ), // Flexible
56
57     // 4. Next
58     const Expanded(
59       child: Icon(Icons.skip_next, color: Colors.white, size: 30),
60     ), // Expanded
61
62     // 5. Repeat
63     const Expanded(
64       child: Icon(Icons.repeat, color: Colors.white, size: 30),
65     ), // Expanded
66   ],
67   ), // Row
68 ), // Container
69
70 body: const Center(
71   child: Text(
72     "Music Player UI",
73     style: TextStyle(fontsize: 20),
74   ), // Text
75 ), // Center
76 ); // Scaffold
77
78 }
79 }
```



Pada layar utama aplikasi, terlihat teks “Pemutar Musik” yang berada tepat di bagian tengah sebagai fokus awal pengguna. Di bagian bawah, terdapat sebuah control bar dengan latar gelap yang memuat lima tombol pengendali musik: Shuffle, Previous, Play, Next, dan Repeat. Tombol Play ditempatkan di

tengah dan dibuat dengan ukuran lebih besar sehingga tampak lebih menonjol dibanding empat ikon lainnya yang menggunakan ukuran seragam. Seluruh ikon diberi warna putih sehingga kontras dengan background gelap dan mudah dilihat oleh pengguna.

#### **IV. Kesimpulan**

Dari pengerjaan tugas ini dapat disimpulkan bahwa penggunaan widget Expanded dan Flexible pada Flutter sangat efektif untuk mengatur elemen UI agar terlihat seimbang dan terstruktur. Seluruh tombol kontrol musik dapat tersusun secara proporsional, sementara tombol Play dapat dibuat lebih dominan sesuai fungsinya. Melalui praktikum ini diperoleh pemahaman bahwa pemilihan widget tata letak yang tepat berpengaruh besar terhadap kenyamanan tampilan aplikasi, serta pentingnya memahami peran flex dan fit dalam pengaturan distribusi ruang.