

LAPORAN PRAKTIKUM PEMROGRAMAN MOBILE

(Widget SizedBox, Spacer, dan Card)



Dosen Pembimbing:

A'LA SYAUQI,M.Kom

Disusun oleh¹:

Nama : Muhammad Salman Alfarisi

NIM : 230605110060

Kelas : C

Tanggal : Rabu, 17 September 2025

FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM

MALANG 2025/2026

¹ Muhammad Salman Alfarisi (230605110060)

I. Tujuan

Tujuan dari praktikum ini adalah untuk menerapkan widget **Card** dalam menampilkan informasi musik yang sedang diputar dengan tampilan rapi dan terstruktur. Selain itu, praktikum ini bertujuan agar mahasiswa dapat memahami penggunaan **SizedBox** sebagai pengatur jarak tetap antar elemen, serta penggunaan **Spacer** untuk mengatur jarak proporsional sehingga antarmuka menjadi lebih responsif. Dengan menggunakan **ThemeData.dark()**, aplikasi diharapkan memiliki tampilan yang elegan dan modern sesuai kebutuhan desain UI pada Flutter.

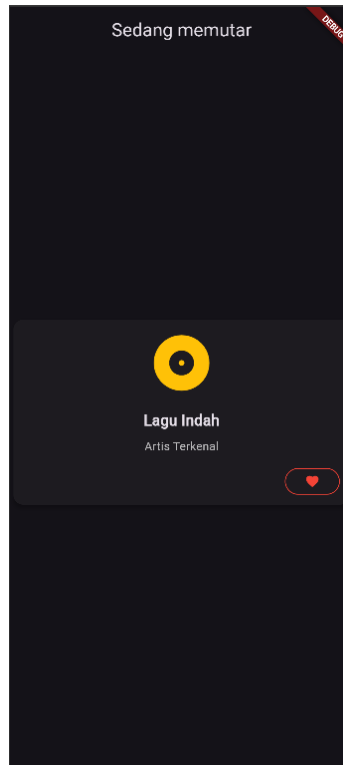
II. Langkah Kerja

1. Buat proyek Flutter baru dengan nama `music_card`.
 - Jalankan perintah `flutter create music_card`. Buka project di Android Studio atau VS Code.
2. Atur tema aplikasi menjadi gelap.
 - Ubah `MaterialApp` agar menggunakan `ThemeData.dark()` supaya tampilan elegan dengan background gelap.
3. Buat `AppBar` aplikasi.
 - Tambahkan `AppBar` dengan judul “Sedang memutar”. Gunakan properti `centerTitle: true` agar judul berada di tengah.
4. Tempatkan `Card` di tengah layar.
 - Gunakan widget `Center` di dalam body `Scaffold` untuk memposisikan `Card` tepat di tengah layar.
5. Atur properti `Card`.
 - Beri `elevation: 4` agar ada efek bayangan. Gunakan `RoundedRectangleBorder` dengan `borderRadius.circular(12)` agar sudut `Card` membulat. Tambahkan `Padding(EdgeInsets.all(12))` agar isi `Card` tidak terlalu mepet ke tepi.
6. Tambahkan isi `Card` berupa informasi musik.
 - Cover album menggunakan ikon (misalnya `Icons.album`). Judul lagu ditampilkan dengan teks tebal. Nama artis ditampilkan dengan teks lebih kecil dan warna abu-abu.
7. Gunakan `SizedBox` untuk memberi jarak tetap antar elemen.
 - Contoh: jarak antara ikon cover dan judul lagu, serta jarak antara judul lagu dan nama artis.
8. Tambahkan tombol aksi (Like).
 - Gunakan ikon hati (`Icons.favorite`). Bungkus dalam `Row` agar posisinya bisa diatur.
9. Gunakan `Spacer` untuk pengaturan jarak proporsional.
 - Letakkan `Spacer` di kiri dan kanan ikon hati agar posisinya tepat di tengah.
10. Jalankan aplikasi di emulator atau perangkat.

- Pastikan Card tampil di tengah layar. Hasil akhir: tampilan kartu musik dengan cover album, judul lagu, nama artis, serta tombol Like yang rapi.

III. Screenshot Hasil

1. Tampilan Aplikasi:



2. Source code:

```

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
10   @override
11   Widget build(BuildContext context) {
12     return MaterialApp(
13       title: 'Music Card',
14       theme: ThemeData.dark(), // tema gelap
15       home: Scaffold(
16         appBar: AppBar(
17           title: const Text('Sedang memutar'),
18           centerTitle: true,
19         ),
20         body: Center(
21           child: Card(
22             elevation: 4,
23             shape: RoundedRectangleBorder(
24               borderRadius: BorderRadius.circular(12),
25             ),
26             child: Padding(
27               padding: const EdgeInsets.all(12),
28               child: Column(
29                 mainAxisAlignment: MainAxisAlignment.min,
30                 children: [
31                   const Icon(
32                     Icons.album,

```

```

33             size: 80,
34             color: Colors.amber,
35           ),
36           const SizedBox(height: 16),
37           const Text(
38             "Lagu Indah",
39             style: TextStyle(
40               fontSize: 18, fontWeight: FontWeight.bold),
41           ),
42           const SizedBox(height: 8),
43           const Text(
44             "Artis Terkenal",
45             style: TextStyle(
46               fontSize: 14, color: Colors.white70),
47           ),
48           const SizedBox(height: 16),
49           Row(
50             children: [
51               const Spacer(),
52               OutlinedButton(
53                 onPressed: () {},
54                 style: OutlinedButton.styleFrom(
55                   side: const BorderSide(color: Colors.red),
56                 ),
57               ),
58               OutlinedButton(
59                 child: const Icon(Icons.favorite,
60                   color: Colors.red),
61               ),
62             ],
63           ),
64           const Column(
65             children: [
66               const Padding(

```

IV. Kesimpulan

Dari praktikum yang telah dilakukan dapat disimpulkan bahwa widget **Card** sangat efektif untuk menampilkan informasi musik secara rapi dalam bentuk kartu interaktif. Penggunaan **SizedBox** membantu memberikan jarak tetap antar elemen sehingga tata letak lebih tertata, sementara **Spacer** berperan dalam mengatur jarak proporsional agar tampilan menyesuaikan dengan ukuran layar yang berbeda. Dengan mengombinasikan Card, SizedBox, dan Spacer, diperoleh desain antarmuka sederhana namun tetap elegan dan informatif sesuai spesifikasi tugas.