

# **LAPORAN PROJECT UTS**

## **MOBILE PROGRAMMING LANJUT**

*“FBOX (Platform Musik dengan Flutter) ”*

Dosen Pengampu:  
Ade Kurniawan, S.Pd.T, M.Pd.T



Oleh,  
Muhammad Fatieh Akram Faritshi  
NIM : 22346038

**PROGRAM STUDI INFORMATIKA**  
**DEPARTEMEN TEKNIK ELEKTRONIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS NEGERI PADANG**  
**2025**

# **FBOX**

## **A Music Platform (*Flutter-Based App*)**

### **A. Pendahuluan**

Di era digital saat ini, cara masyarakat menikmati musik telah berubah dari media fisik ke platform digital. Pengguna kini menginginkan akses cepat, praktis, dan fleksibel untuk mendengarkan lagu favorit mereka kapan saja dan di mana saja. Perubahan ini membuka peluang besar untuk menciptakan aplikasi musik yang responsif, menarik, dan mudah digunakan.

FBOX dikembangkan sebagai solusi dari kebutuhan tersebut. Aplikasi ini dirancang untuk memberikan pengalaman menjelajahi musik yang sederhana, modern, dan menyenangkan. Meski belum mendukung pemutaran audio secara langsung, FBOX sudah memiliki tampilan antarmuka dan struktur navigasi yang cocok untuk demo, tugas akhir, atau prototype aplikasi musik digital.

FBOX juga dibuat sebagai proyek pembelajaran dalam pengembangan aplikasi mobile menggunakan Flutter. Proyek ini tidak hanya bertujuan untuk memenuhi tugas UTS, tetapi juga sebagai bentuk penyaluran minat dan hobi dalam dunia musik. Dengan fitur dasar yang sudah tersedia, FBOX dapat menjadi dasar untuk pengembangan aplikasi musik yang lebih lengkap di masa depan.

#### **1. Tujuan Pembuatan**

Tujuan utama dari aplikasi ini adalah untuk:

- Menyediakan platform eksplorasi dan pemutaran lagu dengan antarmuka pengguna (UI) yang menarik dan mudah digunakan.
- Melatih kemampuan dalam mengembangkan aplikasi mobile berbasis Flutter, baik untuk keperluan pembelajaran maupun demonstrasi proyek.
- Memenuhi tugas proyek Ujian Tengah Semester (UTS) sebagai bagian dari penilaian mata kuliah Mobile Programming Lanjut.

- Mewujudkan ide pribadi berdasarkan ketertarikan dan hobi dalam mendengarkan musik, dengan mengembangkan aplikasi yang sesuai minat.

## 2. Target Pengguna

Pengguna muda, yang menginginkan akses cepat dan sederhana terhadap daftar lagu pilihan, dengan desain antarmuka yang kekinian.

## B. FBOX

FBOX adalah aplikasi pemutar musik berbasis mobile dan website yang dikembangkan menggunakan framework Flutter, dengan fokus utama pada pengalaman eksplorasi lagu berdasarkan genre yang sederhana, menarik, dan mudah digunakan. Aplikasi ini dirancang sebagai proyek pembelajaran maupun prototipe awal untuk pengembangan aplikasi streaming musik profesional di masa depan.

Meskipun FBOX belum dilengkapi dengan fitur pemutaran audio secara penuh, aplikasi ini sudah memiliki struktur navigasi yang baik, tampilan antarmuka pengguna (UI) yang intuitif, serta sejumlah fitur penting yang layak untuk kebutuhan demo, presentasi, atau sebagai landasan awal proyek pengembangan lebih lanjut.

Struktur halaman aplikasi terdiri dari:

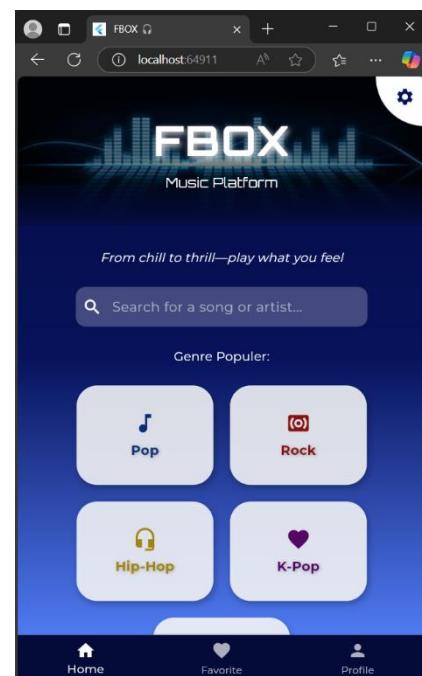
### 1. Halaman Utama (main.dart)

main.dart: Merupakan entry point aplikasi.

Mengatur tema aplikasi, routes, dan menampilkan halaman utama.

Halaman Utama: Halaman ini menampilkan antarmuka utama FBOX, yang menyajikan informasi branding, fitur pencarian lagu, dan akses genre populer.

- a. Tema aplikasi (`ThemeData` dengan font Montserrat dan warna dominan biru tua).
- b. Menampilkan banner besar dengan nama aplikasi dan slogan.
- c. Fitur pencarian lagu (*search bar*).



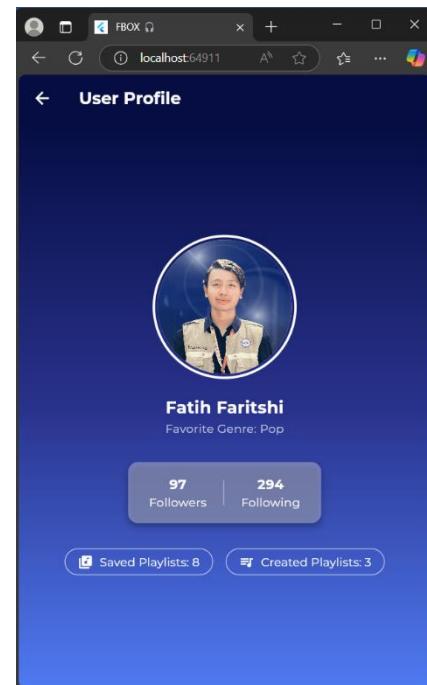
- d. Box daftar genre (Pop, Rock, Hip-Hop, K-Pop, Jazz).
- e. Daftar lagu ditampilkan dalam bentuk list dengan gambar cover, judul, artis.
- f. Tombol icon settings di pojok kanan atas untuk mengakses pengaturan aplikasi.
- g. BottomNavigationBar dengan tiga ikon: Home, Favorite, dan Profile.

## 2. Halaman Profil Pengguna (user\_profile.dart)

Diakses melalui navigasi dengan icon person.

Halaman ini menampilkan informasi profil pengguna, seperti :

- a. Foto Pengguna
- b. Nama Pengguna
- c. Genre Favorit
- d. Jumlah Followers dan Following
- e. Daftar playlist yang disukai dan dibuat



## 3. Halaman Lagu Favorit (favorite\_page.dart)

Halaman ini menampilkan daftar lagu favorit pengguna. Diakses melalui navigasi bawah dengan ikon favorite/love, halaman ini berfungsi sebagai tempat pengguna menyimpan dan melihat lagu-lagu yang mereka sukai. Halaman ini terdapat:

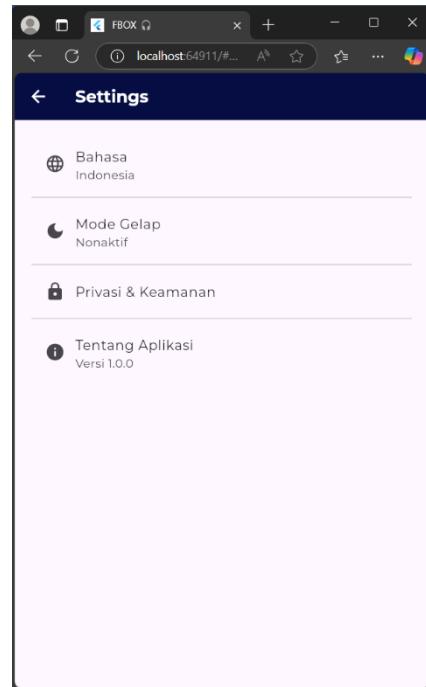
- a. AppBar dengan judul "My Favorite".
- b. Tombol icon settings di pojok kanan atas untuk mengakses pengaturan aplikasi.
- c. Bagian utama (body) menampilkan pesan teks yang menyatakan bahwa belum ada lagu favorit, dengan ikon ekspresi lucu ("lol") untuk memberi kesan santai.
- d. BottomNavigationBar dengan tiga ikon: Home, Favorite, dan Profile.



#### 4. Halaman Pengaturan (settings\_page.dart)

Halaman ini menyediakan berbagai opsi pengaturan untuk aplikasi, memungkinkan pengguna menyesuaikan pengalaman penggunaan sesuai preferensi mereka. Halaman ini dapat diakses melalui ikon settings yang tersedia di pojok kanan atas halaman lain seperti Home dan Favorit.

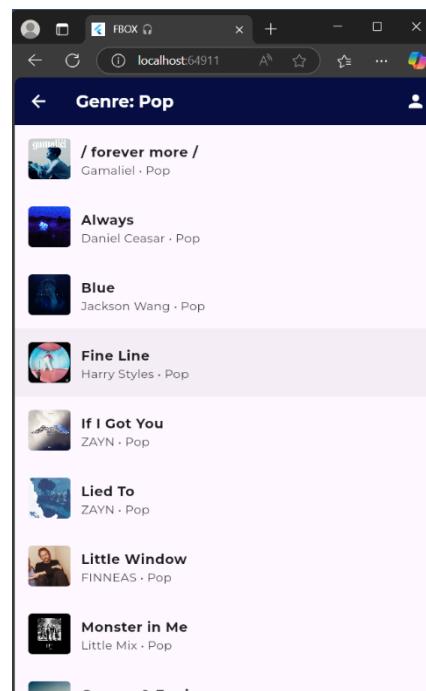
- a. AppBar dengan judul "Pengaturan".
- b. Konten ditampilkan dalam bentuk ListView, sehingga pengguna dapat menggulir opsi pengaturan dengan mudah.
- c. Beberapa pengaturan yang tersedia, namun tidak berfungsi:
  - Bahasa: Menampilkan bahasa yang digunakan dalam aplikasi (Indonesia).
  - Mode Gelap: Informasi apakah mode gelap aktif atau tidak (saat ini Nonaktif).
  - Privasi & Keamanan: Akses ke pengaturan yang berkaitan dengan keamanan akun pengguna.
  - Tentang Aplikasi: Menampilkan informasi versi aplikasi (Versi 1.0.0).



#### 5. Halaman Daftar Lagu (song\_list.dart)

Halaman ini menampilkan daftar lagu berdasarkan genre yang dipilih pengguna. Diakses melalui box genre yang ada di halaman utama, dan memberikan pengguna akses lebih dalam untuk menjelajahi lagu-lagu sesuai minat mereka.

- a. AppBar dengan judul dinamis yang menampilkan genre yang sedang dibuka, misalnya "Genre: Pop".
- b. Tombol ikon profil di kanan atas yang mengarahkan pengguna ke halaman profil mereka.



c. Bagian utama halaman menampilkan daftar lagu dalam bentuk list. Setiap item berisi:

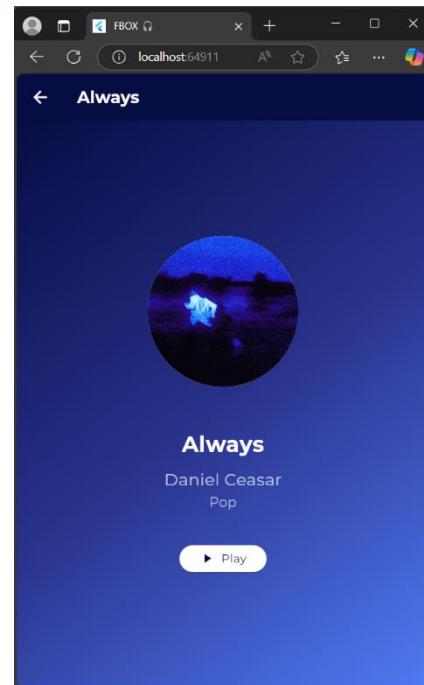
- Gambar cover lagu
- Judul lagu
- Nama artis dan genre

Pengguna dapat mengetuk salah satu lagu untuk melihat detailnya di halaman SongDetailPage.

## 6. Halaman Detail Lagu (song\_detail.dart)

Halaman ini menampilkan detail lengkap dari lagu yang dipilih pengguna, termasuk visual animasi yang interaktif. Ini adalah halaman lanjutan yang dibuka saat pengguna mengetuk salah satu lagu dari daftar genre.

- a. AppBar menampilkan judul lagu di bagian atas dengan warna latar biru gelap yang konsisten.
- b. Tampilan visual lagu yang mencolok:
- c. Gambar cover lagu ditampilkan dalam bentuk lingkaran (CircleAvatar)
  - Gambar akan berputar dan membesar secara animatif saat tombol Play ditekan, menggunakan RotationTransition dan ScaleTransition yang dikontrol oleh AnimationController.
  - Informasi lagu seperti judul, artis, dan genre ditampilkan di bawah gambar dengan tipografi yang elegan dan warna kontras.
  - Tombol Play/Pause di bagian bawah memungkinkan pengguna memicu animasi seolah lagu sedang diputar. Tombol ini tidak memutar audio sungguhan, tetapi menampilkan efek visual sebagai simulasi player.



## C. Implementasi Kode

### 1. main.dart :

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

```
import 'package:flutter/animation.dart';
import 'song_list.dart';
import 'user_profile.dart';
import 'favorite_page.dart';
import 'settings_page.dart';

void main() {
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      title: 'FBOX 🎵',
      theme: ThemeData(
        fontFamily: 'Montserrat',
        primaryColor: const Color.fromARGB(255, 5, 13, 67),
        appBarTheme: const AppBarTheme(
          backgroundColor: Color.fromARGB(255, 5, 13, 67),
          titleTextStyle: TextStyle(
            fontFamily: 'Montserrat',
            color: Colors.white,
            fontSize: 20,
            fontWeight: FontWeight.bold,
          ),
          iconTheme: IconThemeData(color: Colors.white),
        ),
        textTheme: const TextTheme(
          bodyMedium: TextStyle(fontFamily: 'Montserrat'),
        ),
      ),
      home: const HomePage(),
      routes: {'/settings': (context) => const SettingsPage()},
    );
  }
}

class HomePage extends StatefulWidget {
  const HomePage({super.key});

  @override
  State<HomePage> createState() => HomePageState();
}
```

```
class _HomePageState extends State<HomePage> {
  final TextEditingController _searchController = TextEditingController();
  List<Map<String, String>> filteredSongs = [];

  int _selectedIndex = 0;

  final List<Map<String, String>> songs = const [
    {
      "title": "/ forever more /",
      "artist": "Gamaliel",
      "genre": "Pop",
      "image": "assets/q1.png",
    },
    {
      "title": "Always",
      "artist": "Daniel Ceasar",
      "genre": "Pop",
      "image": "assets/always.png",
    },
    {
      "title": "Blue",
      "artist": "Jackson Wang",
      "genre": "Pop",
      "image": "assets/blue.jpeg",
    },
    {
      "title": "Ditto",
      "artist": "NewJeans",
      "genre": "K-Pop",
      "image": "assets/ditto.jpg",
    },
    {
      "title": "ETA",
      "artist": "NewJeans",
      "genre": "K-Pop",
      "image": "assets/eta.jpg",
    },
    {
      "title": "Fine Line",
      "artist": "Harry Styles",
      "genre": "Pop",
      "image": "assets/fine-line.jpeg",
    },
    {
      "title": "If I Got You",
      "artist": "ZAYN",
      "genre": "Pop",
      "image": "assets/if-got-you.jpeg",
    },
  ];
}
```

```
        },
        {
            "title": "her",
            "artist": "GOT7",
            "genre": "K-Pop",
            "image": "assets/got7.jpeg",
        },
        {
            "title": "Lied To",
            "artist": "ZAYN",
            "genre": "Pop",
            "image": "assets/liedto.jpeg",
        },
        {
            "title": "Little Window",
            "artist": "FINNEAS",
            "genre": "Pop",
            "image": "assets/window.png",
        },
        {
            "title": "Love Wins All",
            "artist": "IU",
            "genre": "K-Pop",
            "image": "assets/iu.png",
        },
        {
            "title": "Monster in Me",
            "artist": "Little Mix",
            "genre": "Pop",
            "image": "assets/lm5.jpeg",
        },
        {
            "title": "Oceans & Engines",
            "artist": "NIKI",
            "genre": "Pop",
            "image": "assets/oceans.jpg",
        },
        {
            "title": "Shape of You",
            "artist": "Ed Sheeran",
            "genre": "Pop",
            "image": "assets/ed.png",
        },
        {
            "title": "So Long, London",
            "artist": "Taylor Swift",
            "genre": "Pop",
            "image": "assets/london.png",
        }
```



```
        song['artist']!.toLowerCase().contains(query.toLowerCase()),
    se()),
    )
    .toList();
})
}

void _onBottomNavTapped(int index) {
    setState(() {
        selectedIndex = index;
    });

    if (index == 1) {
        Navigator.push(
            context,
            MaterialPageRoute(builder: (_) => const FavoritePage()),
        );
    } else if (index == 2) {
        Navigator.push(
            context,
            MaterialPageRoute(builder: (_) => const UserProfilePage()),
        );
    }
}

@Override
Widget build(BuildContext context) {
    return Scaffold(
        body: Container(
            decoration: const BoxDecoration(
                gradient: LinearGradient(
                    begin: Alignment.topCenter,
                    end: Alignment.bottomCenter,
                    colors: [
                        Color.fromARGB(255, 5, 13, 67),
                        Color.fromARGB(255, 7, 18, 90),
                        Color.fromARGB(255, 80, 123, 243),
                    ],
            ),
        ),
        child: Column(
            children: [
                SizedBox(
                    width: double.infinity,
                    height: 180,
                    child: Stack(
                        fit: StackFit.expand,
                        children: [

```

```
Image.asset('assets/banner.jpg', fit: BoxFit.cover),

// Overlay hitam semi-transparan seluruh banner
Container(color: Colors.black.withOpacity(0.5)),

// Efek gradasi pudar di bawah banner
Positioned(
    bottom: 0,
    left: 0,
    right: 0,
    height: 50,
    child: Container(
        decoration: const BoxDecoration(
            gradient: LinearGradient(
                begin: Alignment.topCenter,
                end: Alignment.bottomCenter,
                colors: [
                    Color.fromARGB(34, 6, 15, 75),
                    Color.fromARGB(255, 5, 13, 67), // Atau
samakan dengan background kamu
                ],
            ),
        ),
    ),
),
),

// Background transparan belakang icon settings
Positioned(
    top: 0,
    right: 0,
    child: Container(
        width: 60,
        height: 60,
        decoration: BoxDecoration(
            color: Colors.white,
            borderRadius: const BorderRadius.only(
                bottomLeft: Radius.circular(100),
            ),
        ),
    ),
),
),

// Tombol settings
Positioned(
    top: 5,
    right: 5,
    child: IconButton(
```



```

        ],
      ),
    ),

  const SizedBox(height: 28),
  const Text(
    'From chill to thrill—play what you feel',
    style: TextStyle(
      fontSize: 15,
      color: Colors.white,
      fontStyle: FontStyle.italic,
    ),
  ),

  const SizedBox(height: 25),
  Padding(
    padding: const EdgeInsets.symmetric(horizontal: 70),
    child: TextField(
      controller: _searchController,
      onChanged: _filterSongs,
      style: const TextStyle(color: Colors.white),
      decoration: InputDecoration(
        hintText: 'Search for a song or artist...',
        hintStyle: const TextStyle(color: Colors.white54),
        filled: true,
        fillColor: Colors.white24,
        border: OutlineInputBorder(
          borderRadius: BorderRadius.circular(12),
          borderSide: BorderSide.none,
        ),
        prefixIcon: const Icon(Icons.search, color:
        Colors.white),
      ),
    ),
  ),
),

const SizedBox(height: 25),
const Text(
  'Genre Populer:',
  style: TextStyle(fontSize: 15, color: Colors.white),
),
const SizedBox(height: 15),
if (_searchController.text.isNotEmpty)
  Expanded(
    child: ListView.builder(
      itemCount: filteredSongs.length,
      itemBuilder: (context, index) {
        final song = filteredSongs[index];

```

```

        return ListTile(
            Leading: Image.asset(
                song['image']!,
                width: 50,
                height: 50,
                fit: BoxFit.cover,
            ),
            title: Text(
                song['title']!,
                style: const TextStyle(color: Colors.white),
            ),
            subtitle: Text(
                song['artist']!,
                style: const TextStyle(color: Colors.white70),
            ),
        );
    },
),
),
),
),
),
else
    const Expanded(child: _GenreGrid()),
),
),
),
bottomNavigationBar: BottomNavigationBar(
    backgroundColor: const Color.fromARGB(255, 4, 11, 56),
    selectedItemColor: Colors.white,
    unselectedItemColor: const Color.fromARGB(171, 255, 255, 255),
    currentIndex: _selectedIndex,
    onTap: _onBottomNavTapped,
    items: const [
        BottomNavigationBarItem(icon: Icon(Icons.home), label: 'Home'),
        BottomNavigationBarItem(
            icon: Icon(Icons.favorite),
            label: 'Favorite',
        ),
        BottomNavigationBarItem(icon: Icon(Icons.person), label:
'Profile'),
    ],
),
);
}
}

class _GenreGrid extends StatefulWidget {
const _GenreGrid({super.key});

@Override

```

```
State<GenreGrid> createState() => GenreGridState();
}

class GenreGridState extends State<GenreGrid> {
  static const List<String> genres = [
    'Pop',
    'Rock',
    'Hip-Hop',
    'K-Pop',
    'Jazz',
  ];

  static const Map<String, Map<String, dynamic>> genreStyles = {
    'Pop': {
      'icon': Icons.music_note,
      'textColor': Color(0xFF062F75),
      'iconColor': Color(0xFF062F75),
    },
    'Rock': {
      'icon': Icons.surround_sound,
      'textColor': Color(0xFF89100E),
      'iconColor': Color(0xFF89100E),
    },
    'Hip-Hop': {
      'icon': Icons.headset_mic,
      'textColor': Color(0xFFA2870D),
      'iconColor': Color(0xFFA2870D),
    },
    'K-Pop': {
      'icon': Icons.favorite,
      'textColor': Color(0xFF550863),
      'iconColor': Color(0xFF550863),
    },
    'Jazz': {
      'icon': Icons.audiotrack,
      'textColor': Color(0xFF08660D),
      'iconColor': Color(0xFF08660D),
    },
  };
}

final Map<String, bool> _isHovered = {};

Widget _buildGenreBox(BuildContext context, String genre) {
  final style = genreStyles[genre]!;

  _isHovered.putIfAbsent(genre, () => false);

  return MouseRegion(
```

```
    onEnter: (_) => setState(() => _isHovered[genre] = true),
    onExit: (_) => setState(() => _isHovered[genre] = false),
    child: AnimatedScale(
      scale: _isHovered[genre]! ? 0.95 : 1.0,
      duration: const Duration(milliseconds: 200),
      curve: Curves.easeInOut,
      child: Container(
        margin: const EdgeInsets.all(10),
        width: 165,
        height: 120,
        decoration: BoxDecoration(
          color: Colors.white.withOpacity(0.85),
          borderRadius: BorderRadius.circular(25),
          boxShadow: [
            BoxShadow(
              color: Colors.black.withOpacity(0.25),
              blurRadius: 6,
              offset: const Offset(2, 4),
            ),
          ],
        ),
        child: InkWell(
          borderRadius: BorderRadius.circular(25),
          onTap: () {
            Navigator.push(
              context,
              MaterialPageRoute(builder: (_) => SongListPage(genre: genre)),
            );
          },
        ),
        child: Center(
          child: Column(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
              Icon(style['icon'], size: 30, color: style['iconColor']),
              const SizedBox(height: 6),
              Text(
                genre,
                style: TextStyle(
                  fontSize: 16,
                  fontWeight: FontWeight.bold,
                  fontFamily: 'Montserrat',
                  color: style['textColor'],
                  shadows: const [
                    Shadow(
                      color: Colors.black26,
                      offset: Offset(1, 1),
                    )
                  ],
                ),
              ),
            ],
          ),
        ),
      ),
    ),
  ),
);
```

```

        blurRadius: 2,
        ),
        ],
        ),
        textAlign: TextAlign.center,
        ),
        ],
        ),
        ),
        ),
        ),
        ),
        );
    }

@Override
Widget build(BuildContext context) {
    return Center(
        child: SingleChildScrollView(
            child: Column(
                children: [
                    Row(
                        mainAxisAlignment: MainAxisAlignment.center,
                        children: [
                            _buildGenreBox(context, genres[0]),
                            _buildGenreBox(context, genres[1]),
                        ],
                    ),
                    Row(
                        mainAxisAlignment: MainAxisAlignment.center,
                        children: [
                            _buildGenreBox(context, genres[2]),
                            _buildGenreBox(context, genres[3]),
                        ],
                    ),
                    Row(
                        mainAxisAlignment: MainAxisAlignment.center,
                        children: [
                            _buildGenreBox(context, genres[4]),
                        ],
                    ),
                ],
            );
}
}

```

## 2. user\_profile.dart:

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

class UserProfilePage extends StatelessWidget {
  const UserProfilePage({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('User Profile'),
        backgroundColor: const Color.fromARGB(255, 5, 13, 67),
      ),
      body: Container(
        width: double.infinity,
        height: double.infinity,
        decoration: const BoxDecoration(
          gradient: LinearGradient(
            begin: Alignment.topCenter,
            end: Alignment.bottomCenter,
            colors: [
              Color.fromARGB(255, 5, 13, 67),
              Color.fromARGB(255, 41, 50, 139),
              Color.fromARGB(255, 80, 123, 243),
            ],
          ),
        ),
        child: Center(
          child: Column(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
              // Profile picture
              Container(
                padding: const EdgeInsets.all(3), // Ketebalan outline
                decoration: BoxDecoration(
                  shape: BoxShape.circle,
                  border: Border.all(color: Colors.white, width: 3),
                ),
                child: const CircleAvatar(
                  radius: 80,
                  backgroundImage: AssetImage('assets/fatih.png'),
                ),
              ),
              const SizedBox(height: 20),
              const Text(
                'Fatih Faritshi',
                style: TextStyle(
                  fontSize: 20,
                ),
              ),
            ],
          ),
        ),
      ),
    );
  }
}
```

```
        fontWeight: FontWeight.bold,
        color: Colors.white,
    ),
),
const Text(
    'Favorite Genre: Pop',
    style: TextStyle(color: Colors.white70),
),
const SizedBox(height: 30),

// Followers & Following
Container(
    padding: const EdgeInsets.symmetric(
        vertical: 15,
        horizontal: 25,
    ),
    decoration: BoxDecoration(
        color: Colors.white.withOpacity(0.35),
        borderRadius: BorderRadius.circular(12),
        boxShadow: [
            BoxShadow(
                color: Colors.black.withOpacity(0.3),
                blurRadius: 6,
                offset: const Offset(0, 4),
            ),
        ],
    ),
),
child: Row(
    mainAxisAlignment: MainAxisAlignment.min,
    children: [
        Column(
            children: const [
                Text(
                    '97',
                    style: TextStyle(
                        fontSize: 16,
                        fontWeight: FontWeight.bold,
                        color: Colors.white,
                    ),
                ),
                Text(
                    'Followers',
                    style: TextStyle(color: Colors.white),
                ),
            ],
        ),
    ],
),
),
// Garis pemisah
```

```
Container(
  height: 30,
  width: 1,
  color: Colors.white54,
  margin: const EdgeInsets.symmetric(horizontal: 20),
),
Column(
  children: const [
    Text(
      '294',
      style: TextStyle(
        fontSize: 16,
        fontWeight: FontWeight.bold,
        color: Colors.white,
      ),
    ),
    Text(
      'Following',
      style: TextStyle(color: Colors.white),
    ),
  ],
),
),
),
),
),

const SizedBox(height: 30),


// Playlist Info
Row(
  mainAxisAlignment: MainAxisAlignment.center,
  children: [
    OutlinedButton.icon(
      onPressed: () {},
      icon: const Icon(Icons.library_music, color:
Colors.white),
      label: const Text(
        'Saved Playlists: 8',
        style: TextStyle(color: Colors.white),
      ),
      style: OutlinedButton.styleFrom(
        side: const BorderSide(color: Colors.white70),
        shape: RoundedRectangleBorder(
          borderRadius: BorderRadius.circular(20),
        ),
        padding: const EdgeInsets.symmetric(
          horizontal: 16,
          vertical: 12,
        ),
      ),
    ),
  ],
);
```

```

        ),
        ),
        ),
        const SizedBox(width: 15),
        OutlinedButton.icon(
            onPressed: () {},
            icon: const Icon(Icons.queue_music, color:
Colors.white),
            label: const Text(
                'Created Playlists: 3',
                style: TextStyle(color: Colors.white),
            ),
            style: OutlinedButton.styleFrom(
                side: const BorderSide(color: Colors.white70),
                shape: RoundedRectangleBorder(
                    borderRadius: BorderRadius.circular(20),
                ),
                padding: const EdgeInsets.symmetric(
                    horizontal: 16,
                    vertical: 12,
                ),
            ),
            ),
            ],
            ],
            ),
            ),
            );
        }
    }
}

```

### 3. favorite\_page.dart:

```

import 'package:flutter/material.dart';
import 'user_profile.dart';
import 'main.dart';

class FavoritePage extends StatelessWidget {
    const FavoritePage({super.key});

    @override
    Widget build(BuildContext context) {
        return Scaffold(
            appBar: AppBar(
                title: const Text('My Favorite'),

```

```
backgroundColor: const Color.fromARGB(255, 5, 13, 67),
actions: [
  IconButton(
    icon: const Icon(Icons.settings),
    onPressed: () {
      Navigator.pushNamed(context, '/settings');
    },
  ),
],
body: const Center(
  child: Text(
    'Belum ada lagu favorit 🎵',
    style: TextStyle(
      fontSize: 18,
      color: Colors.black54,
      fontFamily: 'Montserrat',
    ),
  ),
),
bottomNavigationBar: BottomNavigationBar(
  backgroundColor: const Color.fromARGB(255, 5, 13, 67),
  selectedItemColor: Colors.white,
  unselectedItemColor: Color.fromARGB(171, 255, 255, 255),
  currentIndex: 1, // Halaman favorit aktif
  onTap: (index) {
    if (index == 0) {
      Navigator.pushReplacement(
        context,
        MaterialPageRoute(builder: (_) => const HomePage()),
      );
    } else if (index == 2) {
      Navigator.pushReplacement(
        context,
        MaterialPageRoute(builder: (_) => const UserProfilePage()),
      );
    }
  },
  items: const [
    BottomNavigationBarItem(
      icon: Icon(Icons.home),
      label: 'Home',
    ),
    BottomNavigationBarItem(
      icon: Icon(Icons.favorite),
      label: 'Favorite',
    ),
    BottomNavigationBarItem(

```

```
        icon: Icon(Icons.person),
        label: 'Profile',
      ),
    ],
  ),
);
}
}
```

#### 4. settings\_page.dart:

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

class SettingsPage extends StatelessWidget {
  const SettingsPage({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Settings'),
        backgroundColor: const Color.fromARGB(255, 5, 13, 67),
      ),
      body: ListView(
        padding: const EdgeInsets.all(20),
        children: const [
          ListTile(
            leading: Icon(Icons.language),
            title: Text('Bahasa'),
            subtitle: Text('Indonesia'),
          ),
          Divider(),
          ListTile(
            leading: Icon(Icons.dark_mode),
            title: Text('Mode Gelap'),
            subtitle: Text('Nonaktif'),
          ),
          Divider(),
          ListTile(
            leading: Icon(Icons.lock),
            title: Text('Privasi & Keamanan'),
          ),
          Divider(),
          ListTile(
            leading: Icon(Icons.info),
            title: Text('Tentang Aplikasi'),
            subtitle: Text('Versi 1.0.0'),
          ),
        ],
      ),
    );
  }
}
```

```
        ),
      ],
    ),
  );
}
```

## 5. song\_list.dart:

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

class SongListPage extends StatelessWidget {
  final String genre;

  const SongListPage({super.key, required this.genre});

  final List<Map<String, String>> songs = const [
    {
      "title": "/ forever more /",
      "artist": "Gamaliel",
      "genre": "Pop",
      "image": "assets/q1.png",
    },
    {
      "title": "Always",
      "artist": "Daniel Ceasar",
      "genre": "Pop",
      "image": "assets/always.png",
    },
    {
      "title": "Blue",
      "artist": "Jackson Wang",
      "genre": "Pop",
      "image": "assets/blue.jpeg",
    },
    {
      "title": "Ditto",
      "artist": "NewJeans",
      "genre": "K-Pop",
      "image": "assets/ditto.jpg",
    },
    {
      "title": "ETA",
      "artist": "NewJeans",
      "genre": "K-Pop",
      "image": "assets/eta.jpg",
    },
  ];
}
```

```
        },
        {
            "title": "Fine Line",
            "artist": "Harry Styles",
            "genre": "Pop",
            "image": "assets/fineline.jpeg",
        },
        {
            "title": "If I Got You",
            "artist": "ZAYN",
            "genre": "Pop",
            "image": "assets/ifgot.jpeg",
        },
        {
            "title": "her",
            "artist": "GOT7",
            "genre": "K-Pop",
            "image": "assets/got7.jpeg",
        },
        {
            "title": "Lied To",
            "artist": "ZAYN",
            "genre": "Pop",
            "image": "assets/liedto.jpeg",
        },
        {
            "title": "Little Window",
            "artist": "FINNEAS",
            "genre": "Pop",
            "image": "assets/window.png",
        },
        {
            "title": "Love Wins All",
            "artist": "IU",
            "genre": "K-Pop",
            "image": "assets/iu.png",
        },
        {
            "title": "Monster in Me",
            "artist": "Little Mix",
            "genre": "Pop",
            "image": "assets/lm5.jpeg",
        },
        {
            "title": "Oceans & Engines",
            "artist": "NIKI",
            "genre": "Pop",
            "image": "assets/oceans.jpg",
```

```
},
{
  "title": "Shape of You",
  "artist": "Ed Sheeran",
  "genre": "Pop",
  "image": "assets/ed.png",
},
{
  "title": "So Long, London",
  "artist": "Taylor Swift",
  "genre": "Pop",
  "image": "assets/london.png",
},
{
  "title": "Star",
  "artist": "Colde",
  "genre": "K-Pop",
  "image": "assets/star.jpg",
},
{
  "title": "stay a little longer",
  "artist": "ROSÉ",
  "genre": "K-Pop",
  "image": "assets/staylonger.jpeg",
},
{
  "title": "Thank You",
  "artist": "GOT7",
  "genre": "K-Pop",
  "image": "assets/thankyou.jpg",
},
{
  "title": "TOXIC",
  "artist": "MEOVV",
  "genre": "K-Pop",
  "image": "assets/toxic.jpg",
},
{
  "title": "WILDFLOWER",
  "artist": "Billie Eilish",
  "genre": "Pop",
  "image": "assets/wild.png",
},
{
  "title": "Woojooin",
  "artist": "Soe Young Joo (OST Melo Movie)",
  "genre": "K-Pop",
  "image": "assets/woojooin.png",
```

```
        },
    ];

    @override
    Widget build(BuildContext context) {
        final filteredSongs =
            songs.where((song) => song['genre'] == genre).toList();

        return Scaffold(
            appBar: AppBar(
                title: Text(
                    'Genre: $genre',
                    style: const TextStyle(
                        fontFamily: 'Montserrat',
                        fontWeight: FontWeight.bold,
                    ),
                ),
                actions: [
                    IconButton(
                        icon: const Icon(Icons.person),
                        onPressed: () {
                            Navigator.pushNamed(context, '/profile');
                        },
                    ),
                ],
            ),
            body: ListView.builder(
                itemCount: filteredSongs.length,
                itemBuilder: (context, index) {
                    final song = filteredSongs[index];
                    return ListTile(
                        contentPadding: const EdgeInsets.symmetric(
                            horizontal: 16,
                            vertical: 8,
                        ),
                        leading: ClipRRect(
                            borderRadius: BorderRadius.circular(5), // Sudut lengkung
                            child: Image.asset(
                                song['image']!,
                                width: 50,
                                height: 70,
                                fit: BoxFit.cover, // Pastikan rasio 1:1
                            ),
                        ),
                        title: Text(
                            song['title']!,
                            style: const TextStyle(
                                fontFamily: 'Montserrat',
                                color: Colors.purple,
                            ),
                        ),
                    );
                },
            ),
        );
    }
}
```

```
        fontWeight: FontWeight.bold,
        fontSize: 16,
    ),
),
subtitle: Text(
    '${song['artist']} • ${song['genre']}',
    style: const TextStyle(fontFamily: 'Montserrat', fontSize:
14),
),
onTap: () {
    Navigator.push(
        context,
        MaterialPageRoute(
            builder: (context) => SongDetailPage(song: song),
        ),
    );
},
);
},
),
);
}
}
```

## 6. song\_detail.dart:

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

class SongDetailPage extends StatefulWidget {
  final Map<String, String> song;

  const SongDetailPage({required this.song, super.key});

  @override
  State<SongDetailPage> createState() => _SongDetailPageState();
}

class _SongDetailPageState extends State<SongDetailPage>
    with TickerProviderStateMixin {
  bool isPlaying = false;

  late AnimationController _scaleController;
  late AnimationController _rotationController;
  late Animation<double> _scaleAnimation;
  late Animation<double> _rotationAnimation;

  @override
```

```
void initState() {
  super.initState();

  _scaleController = AnimationController(
    vsync: this,
    duration: const Duration(seconds: 2),
  );

  _rotationController = AnimationController(
    vsync: this,
    duration: const Duration(seconds: 5),
  );

  _scaleAnimation = Tween<double>(begin: 1.0, end: 1.2).animate(
    CurvedAnimation(parent: _scaleController, curve: Curves.easeInOut),
  );

  _rotationAnimation = Tween<double>(begin: 0.0, end: 1.0).animate(
    CurvedAnimation(parent: _rotationController, curve: Curves.linear),
  );
}

void togglePlay() {
  setState(() {
    isPlaying = !isPlaying;
    if (isPlaying) {
      _scaleController.repeat(reverse: true);
      _rotationController.repeat();
    } else {
      _scaleController.stop();
      _rotationController.stop();
      _scaleController.reset();
      _rotationController.reset();
    }
  });
}

@Override
void dispose() {
  _scaleController.dispose();
  _rotationController.dispose();
  super.dispose();
}

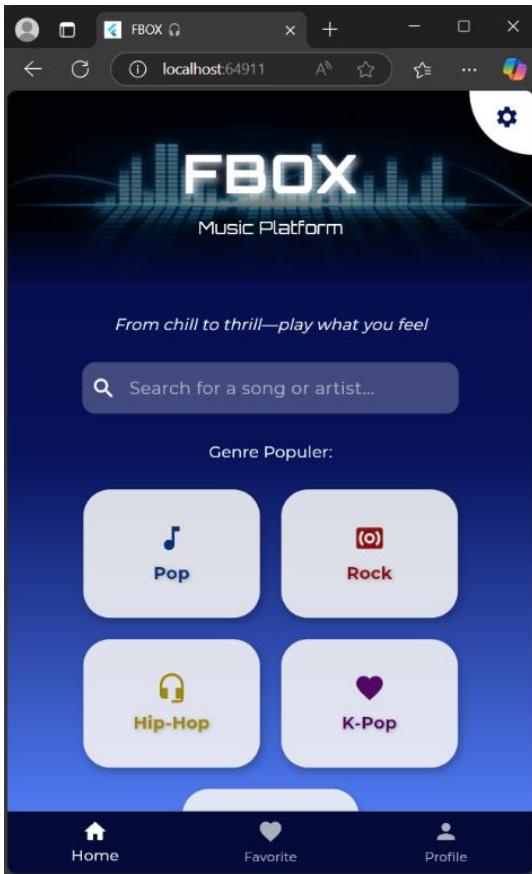
@Override
Widget build(BuildContext context) {
  final song = widget.song;
```

```
return Scaffold(
  appBar: AppBar(
    title: Text(song['title']!),
    backgroundColor: const Color.fromARGB(255, 5, 13, 67),
  ),
  body: Container(
    width: double.infinity,
    decoration: const BoxDecoration(
      gradient: LinearGradient(
        colors: [
          Color.fromARGB(255, 5, 13, 67),
          Color.fromARGB(255, 41, 50, 139),
          Color.fromARGB(255, 80, 123, 243),
        ],
        begin: Alignment.topLeft,
        end: Alignment.bottomRight,
      ),
    ),
    padding: const EdgeInsets.all(20),
    child: Center(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: [
          RotationTransition(
            turns: _rotationAnimation,
            child: ScaleTransition(
              scale: _scaleAnimation,
              child: CircleAvatar(
                radius: 90,
                backgroundImage: AssetImage(
                  song['image'] ?? 'assets/music_placeholder.jpg',
                ),
              ),
            ),
          ),
          const SizedBox(height: 50),
          Text(
            song['title']!,
            style: const TextStyle(
              fontSize: 26,
              fontWeight: FontWeight.bold,
              color: Colors.white,
            ),
          ),
          const SizedBox(height: 10),
          Text(
            song['artist']!,

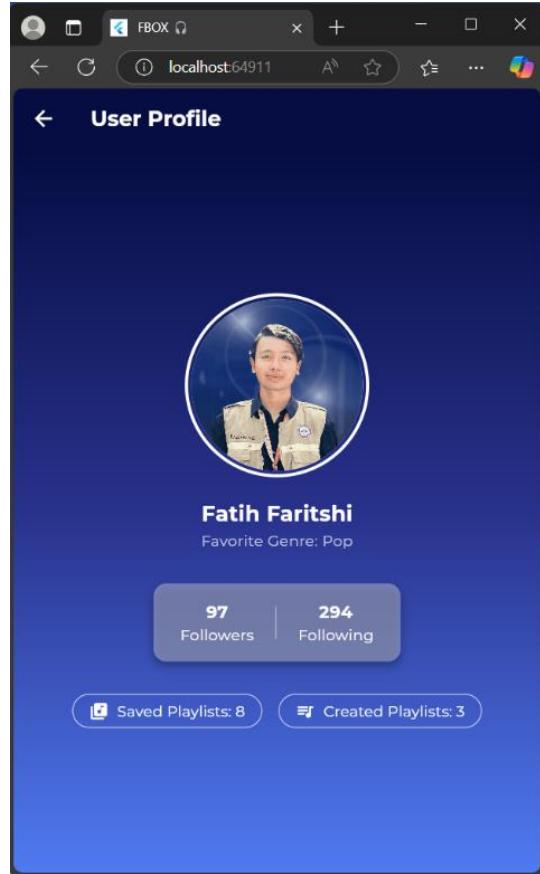
```

```
        style: const TextStyle(fontSize: 20, color:  
Colors.white70),  
    ),  
    Text(  
        song['genre']!,  
        style: const TextStyle(fontSize: 16, color:  
Colors.white60),  
    ),  
    const SizedBox(height: 40),  
    ElevatedButton.icon(  
        onPressed: togglePlay,  
        icon: Icon(isPlaying ? Icons.pause : Icons.play_arrow),  
        label: Text(isPlaying ? 'Pause' : 'Play'),  
        style: ElevatedButton.styleFrom(  
            backgroundColor: Colors.white,  
            foregroundColor: const Color.fromARGB(255, 5, 13, 67),  
            padding: const EdgeInsets.symmetric(  
                horizontal: 24,  
                vertical: 12,  
            ),  
            ),  
            ),  
            ],  
            ),  
            ),  
            );  
        }  
    }  
}
```

## D. Output



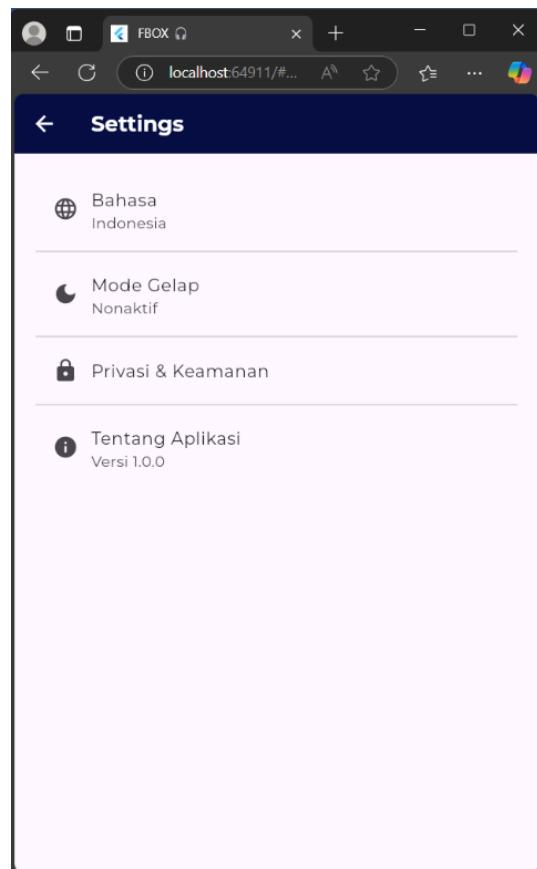
(Halaman Utama)



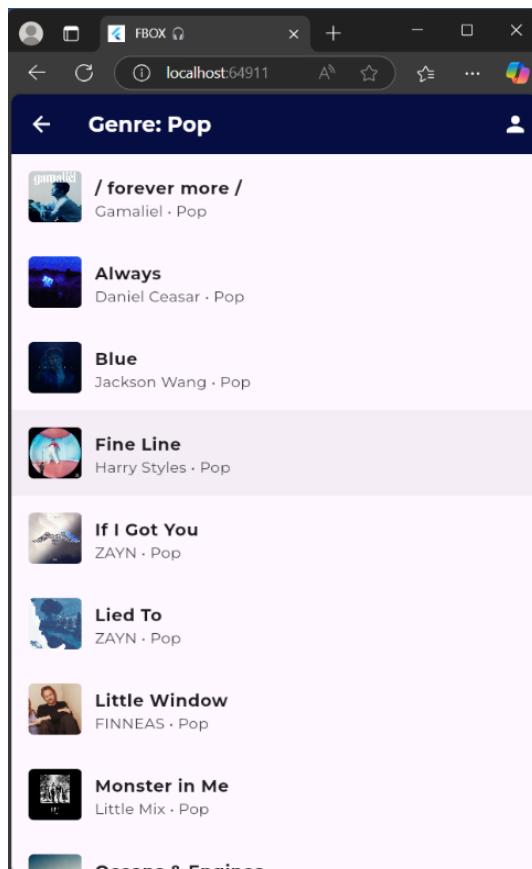
(Halaman Profil Pengguna)



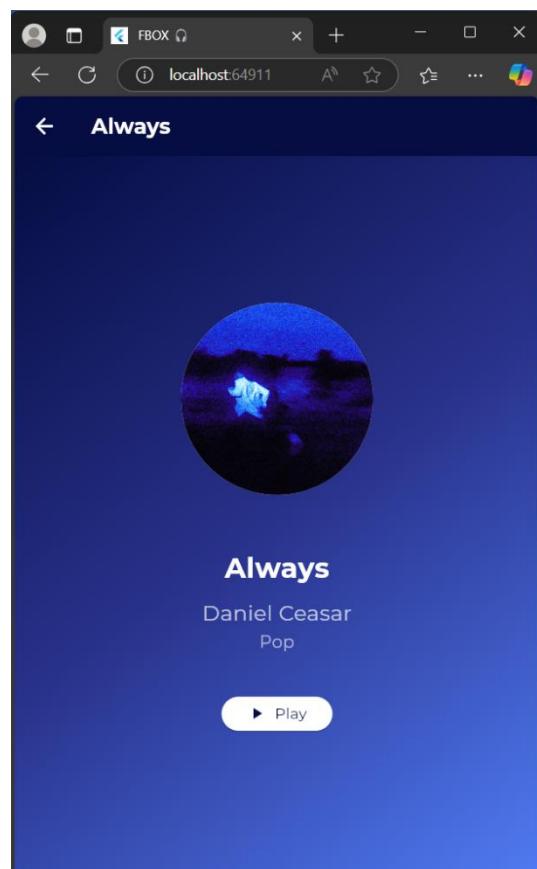
(Halaman Lagu Favorit)



(Halaman Pengaturan)



(Halaman Daftar Lagu)



(Halaman Detail Lagu)

## E. Penggunaan 4 Elemen JobSheet

### 1. Widget

Widget adalah komponen dasar dari semua aplikasi Flutter. Setiap elemen tampilan di layar baik itu teks, gambar, tombol, hingga struktur layout seperti kolom dan baris. Aplikasi FBOX menggunakan berbagai jenis widget, baik stateless maupun stateful, dengan kombinasi UI dan animasi.

| Jenis Widget   | Penggunaan   |
|--|--|
| StatelessWidget  | FavoritePage, SettingsPage, SongListPage, MyApp                          |
| StatefulWidget   | HomePage, SongDetailPage   |
| Scaffold   | Digunakan di semua halaman untuk struktur dasar UI                       |
| AppBar   | Menampilkan judul halaman dan tombol navigasi (seperti settings/profile) |
| ListView.builder   | Menampilkan daftar lagu (di HomePage, SongListPage)                      |
| ListTile   | Untuk menampilkan lagu dalam list item lengkap dengan gambar, judul, dll |
| TextField  | Sebagai kolom pencarian lagu   |
| Text, Icon, Image.asset, CircleAvatar, Container, Column, Row, Padding, SizedBox | Dipakai dalam seluruh layout halaman                                     |
| BottomNavigationBar  | Navigasi di bagian bawah halaman   |
| AnimatedScale, RotationTransition, ScaleTransition                               | Memberi efek animasi di SongDetailPage                                   |

### 2. Button

Button adalah widget yang memungkinkan pengguna **berinteraksi langsung** dengan aplikasi. Saat ditekan, button dapat mengeksekusi suatu aksi seperti navigasi ke halaman lain, mengubah data, atau memicu animasi. Penggunaan tombol ada di setiap

halaman FBOX, mulai dari ikon, tombol aksi utama (Play/Pause), hingga genre navigation.

| Tipe Button             | Halaman                 | Fungsi  |
|-------------------------|-------------------------|---|
| IconButton              | AppBar di semua halaman | Navigasi ke settings atau profile                     |
| ElevatedButton.icon     | SongDetailPage          | Tombol Play/Pause musik dengan ikon                   |
| InkWell                 | HomePage (Genre Grid)   | Menavigasi pengguna ke halaman SongListPage           |
| BottomNavigationBarItem | Semua halaman utama     | Mengaktifkan navigasi bawah (Home, Favorite, Profile) |

### 3. Layout

Layout di Flutter digunakan untuk mengatur posisi dan ukuran widget di layar.

| Widget Layout               | Fungsi/Contoh Penggunaan  |
|-----------------------------|---|
| Column, Row                 | Menyusun elemen vertikal & horizontal (HomePage, SongDetailPage, dsb.)    |
| Expanded, SizedBox          | Menyesuaikan ukuran elemen, membuat ruang antar komponen                  |
| Stack, Positioned           | Menyusun elemen pada banner (HomePage) dengan layer (teks di atas gambar) |
| ListView / ListView.builder | Menampilkan list lagu dengan scrollable layout                            |
| Padding, Margin, Container  | Untuk pengaturan jarak, ukuran dan styling box seperti genre & list lagu  |
| SingleChildScrollView       | Memungkinkan scroll seluruh elemen (GenreGrid, dll)                       |

### 4. Navigasi

Navigasi adalah cara pengguna berpindah antar halaman (screen) dalam aplikasi Flutter. FBOX menggunakan navigasi untuk ke halaman Utama, Favorit, Profil, Daftar Lagu, Detail Lagu, dan Pengaturan.

| Jenis Navigasi  | Contoh Penggunaan  |
|---|--|
| <code>Navigator.push()</code>                         | Menuju ke halaman baru (Profile, Settings, Song Detail)          |
| <code>Navigator.pushReplacement()</code>              | Untuk mengganti halaman aktif saat tap BottomNavigationBar       |
| <code>routes pada MaterialApp</code>                  | Routing global seperti ke /settings                              |
| <code>MaterialPageRoute</code>                        | Navigasi manual antar halaman, dengan data (e.g., genre)         |
| <code>Navigasi Berdasarkan BottomNavigationBar</code> | Navigasi utama antar HomePage, FavoritePage, dan UserProfilePage |