

UNGUIDED MODUL 6



Disusun Oleh :

Nama : Ganes Gemi Putra

Kelas : SE-07-02

NIM : (2311104075)

Dosen : YUDHA ISLAMI SULISTYA

**PROGRAM STUDI S1 SOFTWARE ENGINEERING
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY
PURWOKERTO
2025**

SOAL :**Tugas Mandiri (Unguided)**

1. Modifikasi project Rekomendasi Wisata pada Tugas Unguided 04 modul Antarmuka Pengguna dengan mengimplementasikan widget CustomScrollView, SliverAppBar, dan SliverList untuk merekomendasikan beberapa tempat wisata yang ada di Banyumas disertai foto, nama wisata, dan deskripsi singkat! (buatlah se kreatif mungkin).

1. Pendahuluan**1.1 Latar Belakang**

Proyek ini merupakan pengembangan aplikasi mobile berbasis Flutter yang bertujuan untuk mempromosikan destinasi wisata di Kabupaten Banyumas, Jawa Tengah. Banyumas dikenal sebagai daerah dengan kekayaan alam, budaya, dan kuliner yang belum banyak dieksplorasi oleh wisatawan. Aplikasi ini dirancang untuk memberikan rekomendasi tempat wisata yang akurat, interaktif, dan mudah diakses melalui perangkat mobile. Dengan desain premium dan antarmuka pengguna yang intuitif, aplikasi ini diharapkan dapat meningkatkan minat wisatawan untuk mengunjungi Banyumas.

Proyek ini dikembangkan sebagai bagian dari praktikum pemrograman perangkat bergerak, dengan fokus pada penerapan konsep-konsep dasar Flutter seperti widget, state management, animasi, dan navigasi. Selain itu, proyek ini juga mengintegrasikan elemen lokal seperti bahasa Indonesia untuk meningkatkan keterhubungan dengan pengguna lokal.

1.2 Tujuan Praktikum

- Memahami dan mengimplementasikan konsep dasar pengembangan aplikasi mobile menggunakan Flutter.
- Menerapkan desain antarmuka pengguna (UI/UX) yang responsif dan menarik.
- Mengintegrasikan fitur-fitur interaktif seperti pencarian, filter kategori, dan navigasi halaman.
- Melakukan lokalisasi konten (terjemahan ke bahasa Indonesia) untuk meningkatkan aksesibilitas.
- Menguji dan menganalisis performa aplikasi pada perangkat target.

1.3 Ruang Lingkup

Proyek ini mencakup:

- Pengembangan aplikasi native untuk platform Android dan iOS menggunakan Flutter.
- Implementasi fitur utama: splash screen, daftar wisata, pencarian, filter kategori, dan halaman detail.
- Penggunaan data statis untuk daftar wisata (tidak terhubung ke database eksternal).
- Desain responsif dengan animasi dan efek visual untuk pengalaman pengguna yang premium.

2. Tinjauan Pustaka dan Landasan Teori**2.1 Flutter Framework**

Flutter adalah framework open-source dari Google untuk pengembangan aplikasi mobile lintas platform. Flutter menggunakan bahasa pemrograman Dart dan menyediakan widget-widget siap pakai untuk membangun UI yang konsisten. Keunggulan Flutter meliputi:

- Hot Reload untuk pengembangan cepat.
- Widget tree yang fleksibel untuk desain kustom.
- Dukungan animasi dan transisi yang smooth.

2.2 Konsep UI/UX dalam Aplikasi Mobile

UI/UX (User Interface/User Experience) adalah aspek kunci dalam pengembangan aplikasi. Pada proyek ini, diterapkan prinsip-prinsip seperti:

- **Material Design:** Menggunakan komponen seperti Card, ElevatedButton, dan Scaffold untuk konsistensi.
- **Animasi:** Menggunakan AnimationController dan Tween untuk efek transisi yang menarik.
- **Responsivitas:** Layout yang menyesuaikan dengan berbagai ukuran layar menggunakan MediaQuery dan Flexible widget.

2.3 Lokalisasi dan Internasionalisasi

Lokalisasi dilakukan dengan menerjemahkan teks statis ke bahasa Indonesia, seperti judul header dan deskripsi. Ini meningkatkan relevansi aplikasi untuk pengguna lokal tanpa menggunakan paket eksternal seperti

flutter_localizations.

3. Metodologi

3.1 Alat dan Bahan

- **Perangkat Lunak:** Flutter SDK (versi terbaru), Dart, Android Studio/VS Code.
- **Bahasa Pemrograman:** Dart.
- **Dependensi Utama:** google_fonts untuk font kustom, flutter/material.dart untuk komponen UI.
- **Perangkat Keras:** Komputer dengan spesifikasi minimal (RAM 8GB, prosesor modern), emulator Android atau perangkat fisik.

3.2 Tahapan Pengembangan

1. **Perencanaan:** Analisis kebutuhan, desain wireframe, dan pemilihan fitur.
2. **Implementasi:**
 - Setup proyek Flutter dengan flutter create.
 - Pengembangan widget utama (SplashScreen, WisataList, DetailScreen).
 - Integrasi animasi dan efek visual.
 - Lokalisasi konten (terjemahan judul header ke bahasa Indonesia).
3. **Pengujian:** Build aplikasi dengan flutter pub get, run pada emulator, dan verifikasi fitur.
4. **Dokumentasi:** Penyusunan laporan dan kode sumber.

3.3 Struktur Proyek

- **lib/main.dart:** File utama berisi semua widget dan logika aplikasi.
- **pubspec.yaml:** Konfigurasi dependensi dan aset (gambar wisata).
- **assets/images/:** Folder untuk gambar statis wisata.

4. Implementasi dan Hasil

4.1 Fitur Utama Aplikasi

1. **Splash Screen:** Layar pembuka dengan animasi rotasi logo, fade-in teks, dan transisi ke halaman utama setelah 4 detik.
2. **Daftar Wisata:** Menampilkan kartu wisata dengan gambar, rating, lokasi, dan fasilitas. Mendukung scroll vertikal dan efek shimmer saat loading.
3. **Pencarian dan Filter:** Search bar untuk mencari berdasarkan nama/lokasi, dan chip kategori untuk filter (e.g., Alam & Petualangan, Kuliner & Budaya).
4. **Header Dinamis:** Gambar header yang berganti setiap 5 detik dengan judul dalam bahasa Indonesia (e.g., "Temukan Permata Tersembunyi Banyumas").
5. **Halaman Detail:** Navigasi ke layar detail dengan informasi lengkap wisata, termasuk deskripsi, fasilitas, dan tombol aksi.
6. **Floating Action Button:** Tombol untuk fitur peta (placeholder dengan snackbar).

4.2 Kode Utama dan Penjelasan

- **State Management:** Menggunakan StatefulWidget untuk mengelola state lokal (e.g., _selectedKategori, _searchQuery).
- **Animasi:** AnimationController untuk splash screen dan fade-in list. Timer untuk rotasi header.
- **Lokalisasi:** Judul header diterjemahkan secara manual (e.g., "Discover The Hidden Gems of Banyumas" menjadi "Temukan Permata Tersembunyi Banyumas").
- **Data Wisata:** Daftar objek Wisata dengan atribut seperti name, imageUrl, description, dll.

Contoh kode untuk judul header yang diterjemahkan:

```
String _getTitleForImage(int index) {  
  switch (index) {  
    case 0: return 'Temukan Permata Tersembunyi\nBanyumas';  
    case 1: return 'Jelajahi Petualangan\nPegunungan';  
    case 2: return 'Mengejar Air Terjun\n& Alam';  
    case 3: return 'Budaya Urban &\nKelezatan Lokal';  
    default: return 'Temukan Permata Tersembunyi\nBanyumas';  
  }  
}
```

```
}  
}
```

4.3 Hasil Pengujian

- **Build Sukses:** Aplikasi berhasil dikompilasi dengan flutter pub get tanpa error.
- **Run pada Emulator:** Aplikasi berjalan pada Windows desktop dan browser (Chrome/Edge). Splash screen muncul dengan animasi, list wisata ter-render, dan navigasi berfungsi.
- **Responsivitas:** Layout menyesuaikan dengan ukuran layar, animasi smooth tanpa lag.
- **Lokalisasi:** Judul header dalam bahasa Indonesia meningkatkan keterbacaan untuk pengguna lokal.

5. Analisis dan Pembahasan

5.1 Keunggulan Proyek

- **Desain Premium:** Kombinasi animasi, gradien, dan font Google Fonts menciptakan pengalaman visual yang menarik.
- **Interaktivitas Tinggi:** Fitur pencarian dan filter memudahkan navigasi.
- **Lokalisasi:** Terjemahan konten mendukung inklusivitas budaya.
- **Performa:** Menggunakan widget efisien seperti SliverList untuk scroll yang optimal.

5.2 Kendala dan Solusi

- **Animasi Kompleks:** Sulit mengatur timing animasi; diselesaikan dengan AnimationController dan Interval.
- **Data Statis:** Tidak terhubung database; cocok untuk prototipe, tapi bisa diperluas dengan Firebase.
- **Ukuran Gambar:** Gambar besar memperlambat loading; dioptimalkan dengan Image.asset dan errorBuilder.

5.3 Perbandingan dengan Aplikasi Serupa

Aplikasi ini mirip dengan TripAdvisor atau Google Maps untuk wisata, namun lebih sederhana dan fokus lokal. Keunggulan: UI lebih modern dan bahasa Indonesia.

6. Kesimpulan dan Saran

6.1 Kesimpulan

Proyek ini berhasil mengembangkan aplikasi wisata Banyumas yang fungsional dengan fitur lengkap dan desain premium. Implementasi Flutter memungkinkan pengembangan cepat dan lintas platform. Lokalisasi konten meningkatkan relevansi aplikasi.

Secara khusus, proyek ini merupakan implementasi dari Tugas Unguided 04 modul Antarmuka Pengguna, di mana widget CustomScrollView, SliverAppBar, dan SliverList digunakan untuk merekomendasikan beberapa tempat wisata di Banyumas. Setiap rekomendasi disertai dengan foto, nama wisata, dan deskripsi singkat, yang ditampilkan dalam kartu interaktif dengan efek scroll yang smooth dan responsif.

6.2 Saran untuk Pengembangan Lanjutan

- Integrasi database (e.g., Firebase) untuk data dinamis.
- Tambahkan fitur peta interaktif menggunakan google_maps_flutter.
- Implementasi notifikasi push dan autentikasi pengguna.
- Uji pada perangkat fisik untuk performa yang lebih akurat.

SOURCE CODE :

main.dart

```
import 'package:flutter/material.dart';  
import 'package:google_fonts/google_fonts.dart';  
import 'dart:async';  
import 'dart:math';  
  
void main() {
```

```

runApp(const MyApp());
}

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Wisata Banyumas - Jelajahi Keindahan Alam',
      theme: ThemeData(
        primarySwatch: Colors.green,
        primaryColor: const Color(0xFF2E7D32),
        colorScheme: ColorScheme.fromSwatch(
          primarySwatch: Colors.green,
          backgroundColor: const Color(0xFFF8FDF8),
        ),
        useMaterial3: true,
        appBarTheme: const AppBarTheme(
          backgroundColor: Color(0xFF2E7D32),
          elevation: 0,
          centerTitle: true,
        ),
      ),
      home: const SplashScreen(),
      debugShowCheckedModeBanner: false,
    );
  }
}

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

  @override
  State<SplashScreen> createState() => _SplashScreenState();
}

class _SplashScreenState extends State<SplashScreen>
  with SingleTickerProviderStateMixin {

  late AnimationController _controller;
  late Animation<double> _scaleAnimation;
  late Animation<double> _fadeAnimation;
  late Animation<Offset> _slideAnimation;
  late Animation<Color?> _colorAnimation;
  late Animation<double> _rotationAnimation;

  @override
  void initState() {
    super.initState();

    _controller = AnimationController(
      duration: const Duration(milliseconds: 2500),
      vsync: this,
    );
  }
}

```

```

_scaleAnimation = Tween<double>(  
  begin: 0.0,  
  end: 1.0,  
).animate(CurvedAnimation(  
  parent: _controller,  
  curve: const Interval(0.0, 0.6, curve: Curves.elasticOut),  
));  
  
_fadeAnimation = Tween<double>(  
  begin: 0.0,  
  end: 1.0,  
).animate(CurvedAnimation(  
  parent: _controller,  
  curve: const Interval(0.3, 1.0, curve: Curves.easeIn),  
));  
  
_slideAnimation = Tween<Offset>(  
  begin: const Offset(0, 0.5),  
  end: Offset.zero,  
).animate(CurvedAnimation(  
  parent: _controller,  
  curve: const Interval(0.5, 1.0, curve: Curves.easeOut),  
));  
  
_rotationAnimation = Tween<double>(  
  begin: 0.0,  
  end: 2 * pi,  
).animate(CurvedAnimation(  
  parent: _controller,  
  curve: const Interval(0.0, 1.0, curve: Curves.linear),  
));  
  
_colorAnimation = ColorTween(  
  begin: const Color(0xFF1B5E20),  
  end: const Color(0xFF2E7D32),  
).animate(CurvedAnimation(  
  parent: _controller,  
  curve: Curves.easeInOut,  
));  
  
_controller.forward();  
  
// Navigate to main screen after 4 seconds  
Future.delayed(const Duration(milliseconds: 4000), () {  
  Navigator.pushReplacement(  
    context,  
    PageRouteBuilder(  
      pageBuilder: (context, animation, secondaryAnimation) => const WisataList(),  
      transitionsBuilder: (context, animation, secondaryAnimation, child) {  
        return FadeTransition(  
          opacity: animation,  
          child: child,  
        );  
      },  
      transitionDuration: const Duration(milliseconds: 1200),  
    ),  
  );  
}

```

```

    );
  });
}

@override
void dispose() {
  _controller.dispose();
  super.dispose();
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    body: AnimatedBuilder(
      animation: _controller,
      builder: (context, child) {
        return Container(
          decoration: BoxDecoration(
            gradient: LinearGradient(
              begin: Alignment.topLeft,
              end: Alignment.bottomRight,
              colors: [
                _colorAnimation.value ?? const Color(0xFF1B5E20),
                const Color(0xFF2E7D32),
                const Color(0xFF1B5E20),
              ],
            ),
          ),
          child: Stack(
            children: [
              // Animated background elements
              Positioned(
                top: 100,
                left: -50,
                child: AnimatedContainer(
                  duration: const Duration(milliseconds: 2000),
                  width: 200,
                  height: 200,
                  decoration: BoxDecoration(
                    shape: BoxShape.circle,
                    color: Colors.white.withOpacity(0.05),
                  ),
                ),
              ),
              Positioned(
                bottom: -80,
                right: -80,
                child: AnimatedContainer(
                  duration: const Duration(milliseconds: 2000),
                  width: 300,
                  height: 300,
                  decoration: BoxDecoration(
                    shape: BoxShape.circle,
                    color: Colors.white.withOpacity(0.03),
                  ),
                ),
              ),
            ],
          ),
        );
      },
    ),
  );
}

```

```

),

Center(
  child: Column(
    mainAxisAlignment: MainAxisAlignment.center,
    children: [
      // Animated Logo with Rotation
      RotationTransition(
        turns: _rotationAnimation,
        child: ScaleTransition(
          scale: _scaleAnimation,
          child: Container(
            padding: const EdgeInsets.all(25),
            decoration: BoxDecoration(
              gradient: const LinearGradient(
                colors: [
                  Colors.white,
                  Color(0xFFE8F5E8),
                ],
                begin: Alignment.topLeft,
                end: Alignment.bottomRight,
              ),
            shape: BoxShape.circle,
            boxShadow: [
              BoxShadow(
                color: Colors.black.withOpacity(0.3),
                blurRadius: 30,
                offset: const Offset(0, 15),
              ),
            ],
          ),
        ),
        child: const Icon(
          Icons.landscape_rounded,
          size: 70,
          color: Color(0xFF2E7D32),
        ),
      ),
    ],
  ),
),

const SizedBox(height: 40),

// Main Title with Fade and Slide
FadeTransition(
  opacity: _fadeAnimation,
  child: SlideTransition(
    position: _slideAnimation,
    child: Column(
      children: [
        Text(
          'Wisata Banyumas',
          style: GoogleFonts.poppins(
            fontSize: 36,
            fontWeight: FontWeight.w800,
            color: Colors.white,
            letterSpacing: 1.5,

```




```

    shadows: [
      const Shadow(
        blurRadius: 15,
        color: Colors.black45,
        offset: Offset(3, 3),
      ),
    ],
  ),
),
const SizedBox(height: 12),
Text(
  'Temukan Keindahan Alam',
  style: GoogleFonts.poppins(
    fontSize: 18,
    color: Colors.white.withOpacity(0.9),
    letterSpacing: 1.2,
    fontWeight: FontWeight.w300,
  ),
),
],
),
),
),
const SizedBox(height: 60),

// Enhanced Loading Indicator
FadeTransition(
  opacity: _fadeAnimation,
  child: Column(
    children: [
      Stack(
        alignment: Alignment.center,
        children: [
          Container(
            width: 60,
            height: 60,
            decoration: BoxDecoration(
              shape: BoxShape.circle,
              border: Border.all(
                color: Colors.white.withOpacity(0.3),
                width: 2,
              ),
            ),
          ),
          SizedBox(
            width: 50,
            height: 50,
            child: CircularProgressIndicator(
              valueColor: AlwaysStoppedAnimation<Color>(
                Colors.white.withOpacity(0.8),
              ),
              strokeWidth: 3,
              backgroundColor: Colors.white.withOpacity(0.2),
            ),
          ),
        ],
      ),
    ],
  ),
),

```

```
class ShimmerEffect extends StatefulWidget {
  final Widget child;
  final bool isLoading;

  const ShimmerEffect({
    super.key,
    required this.child,
    required this.isLoading,
  });

  @override
  State<ShimmerEffect> createState() => _ShimmerEffectState();
}
```

Praktikum PPB

```
_controller = AnimationController(  
  duration: const Duration(milliseconds: 1500),  
  vsync: this,  
)..repeat(reverse: true);  
}  
  
@override  
void dispose() {  
  _controller.dispose();  
  super.dispose();  
}  
  
@override  
Widget build(BuildContext context) {  
  return AnimatedBuilder(  
    animation: _controller,  
    builder: (context, child) {  
      return Opacity(  
        opacity: widget.isLoading ? 0.3 + (_controller.value * 0.4) : 1.0,  
        child: child,  
      );  
    },  
    child: widget.child,  
  );  
}  
}  
  
class Wisata {  
  final String name;  
  final String imageUrl;  
  final String description;  
  final String location;  
  final double rating;  
  final String category;  
  final double distance;  
  final int reviewCount;  
  final List<String> facilities;  
  final double price;  
  final bool isPopular;  
  final bool isFeatured;  
  
  Wisata({  
    required this.name,  
    required this.imageUrl,  
    required this.description,  
    required this.location,  
    required this.rating,  
    required this.category,  
    required this.distance,  
    required this.reviewCount,  
    required this.facilities,  
    required this.price,  
    required this.isPopular,  
    required this.isFeatured,  
  });  
}
```

```

class KategoriFilter extends StatefulWidget {
  final String selectedKategori;
  final ValueChanged<String> onKategoriChanged;

  const KategoriFilter({
    super.key,
    required this.selectedKategori,
    required this.onKategoriChanged,
  });

  @override
  State<KategoriFilter> createState() => _KategoriFilterState();
}

class _KategoriFilterState extends State<KategoriFilter> {
  final List<Map<String, dynamic>> _kategoriList = [
    {'name': 'Semua', 'icon': Icons.all_inclusive, 'color': const Color(0xFF2E7D32)},
    {'name': 'Alam & Petualangan', 'icon': Icons.terrain, 'color': const Color(0xFF4CAF50)},
    {'name': 'Kuliner & Budaya', 'icon': Icons.restaurant, 'color': const Color(0xFF8BC34A)},
    {'name': 'Sejarah & Kuliner', 'icon': Icons.history, 'color': const Color(0xFFCDDC39)},
    {'name': 'Alam & Relaksasi', 'icon': Icons.spa, 'color': const Color(0xFF2196F3)},
    {'name': 'Edukasi & Budaya', 'icon': Icons.school, 'color': const Color(0xFF673AB7)},
    {'name': 'Alam & Edukasi', 'icon': Icons.eco, 'color': const Color(0xFF009688)},
    {'name': 'Alam & Kesehatan', 'icon': Icons.healing, 'color': const Color(0xFFFF9800)},
  ];

  @override
  Widget build(BuildContext context) {
    return Container(
      height: 80,
      decoration: BoxDecoration(
        color: Colors.white,
        boxShadow: [
          BoxShadow(
            color: Colors.black.withOpacity(0.08),
            blurRadius: 15,
            offset: const Offset(0, 5),
          ),
        ],
      ),
      child: ListView.separated(
        scrollDirection: Axis.horizontal,
        padding: const EdgeInsets.symmetric(horizontal: 20),
        itemCount: _kategoriList.length,
        separatorBuilder: (context, index) => const SizedBox(width: 12),
        itemBuilder: (context, index) {
          final kategori = _kategoriList[index];
          final isSelected = widget.selectedKategori == kategori['name'];

          return _buildKategoriChip(kategori, isSelected);
        },
      ),
    );
  }
}

```

```
Widget _buildKategoriChip(Map<String, dynamic> kategori, bool isSelected) {
  return GestureDetector(
    onTap: () => widget.onKategoriChanged(kategori['name']),
    child: AnimatedContainer(
      duration: const Duration(milliseconds: 400),
      curve: Curves.easeInOutQuart,
      padding: const EdgeInsets.symmetric(horizontal: 20, vertical: 12),
      decoration: BoxDecoration(
        gradient: isSelected
          ? LinearGradient(
              colors: [
                kategori['color'] as Color,
                Color.lerp(kategori['color'] as Color, Colors.black, 0.1)!,
              ],
              begin: Alignment.topLeft,
              end: Alignment.bottomRight,
            )
          : null,
        color: isSelected ? null : Colors.grey[50],
        borderRadius: BorderRadius.circular(25),
        border: isSelected
          ? null
          : Border.all(
              color: Colors.grey[200]!,
              width: 1.5,
            ),
        boxShadow: isSelected
          ? [
              BoxShadow(
                color: (kategori['color'] as Color).withOpacity(0.4),
                blurRadius: 15,
                offset: const Offset(0, 8),
              ),
            ]
          : [
              BoxShadow(
                color: Colors.black.withOpacity(0.03),
                blurRadius: 8,
                offset: const Offset(0, 3),
              ),
            ],
      ),
    child: Row(
      mainAxisAlignment: MainAxisAlignment.min,
      children: [
        Icon(
          kategori['icon'] as IconData,
          color: isSelected ? Colors.white : kategori['color'] as Color,
          size: 18,
        ),
        const SizedBox(width: 8),
        Text(
          kategori['name'],
          style: GoogleFonts.poppins(
            fontSize: 14,
            fontWeight: FontWeight.w600,

```

```

        color: isSelected ? Colors.white : Colors.grey[700],
        letterSpacing: -0.2,
      ),
    ),
  ],
),
),
);
}
}

class SearchBar extends StatefulWidget {
  final ValueChanged<String> onSearchChanged;
  final VoidCallback onSearchTap;

  const SearchBar({
    super.key,
    required this.onSearchChanged,
    required this.onSearchTap,
  });

  @override
  State<SearchBar> createState() => _SearchBarState();
}

class _SearchBarState extends State<SearchBar> {
  final TextEditingController _controller = TextEditingController();
  final FocusNode _focusNode = FocusNode();

  @override
  void dispose() {
    _controller.dispose();
    _focusNode.dispose();
    super.dispose();
  }

  @override
  Widget build(BuildContext context) {
    return Container(
      margin: const EdgeInsets.symmetric(horizontal: 20, vertical: 16),
      padding: const EdgeInsets.symmetric(horizontal: 16),
      decoration: BoxDecoration(
        color: Colors.white,
        borderRadius: BorderRadius.circular(20),
        boxShadow: [
          BoxShadow(
            color: Colors.black.withOpacity(0.1),
            blurRadius: 15,
            offset: const Offset(0, 5),
          ),
        ],
      ),
      child: Row(
        children: [
          const Icon(Icons.search, color: Color(0xFF2E7D32), size: 24),
          const SizedBox(width: 12),

```

```
Expanded(
  child: TextField(
    controller: _controller,
    focusNode: _focusNode,
    onChanged: widget.onSearchChanged,
    onTap: widget.onSearchTap,
    decoration: InputDecoration(
      hintText: 'Cari wisata, lokasi, atau kategori...',
      hintStyle: GoogleFonts.poppins(
        color: Colors.grey[500],
        fontSize: 16,
      ),
    ),
    border: InputBorder.none,
    contentPadding: const EdgeInsets.symmetric(vertical: 16),
  ),
  style: GoogleFonts.poppins(
    fontSize: 16,
    color: Colors.grey[800],
  ),
),
),
if (_controller.text.isNotEmpty)
  IconButton(
    icon: const Icon(Icons.clear, color: Colors.grey, size: 20),
    onPressed: () {
      _controller.clear();
      widget.onSearchChanged("");
      _focusNode.unfocus();
    },
  ),
],
),
);
}
```

```
class DetailScreen extends StatelessWidget {
  final Wisata wisata;

  const DetailScreen({super.key, required this.wisata});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: const Color(0xFFF8FDF8),
      body: CustomScrollView(
        slivers: [
          SliverAppBar(
            expandedHeight: 350,
            floating: false,
            pinned: true,
            backgroundColor: const Color(0xFF2E7D32),
            flexibleSpace: FlexibleSpaceBar(
              background: Stack(
                fit: StackFit.expand,
                children: [
```

```

Image.asset(
  wisata.imageUrl,
  fit: BoxFit.cover,
  errorBuilder: (context, error, stackTrace) {
    return Container(
      color: const Color(0xFFE8F5E8),
      child: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: [
          const Icon(
            Icons.landscape,
            size: 80,
            color: Color(0xFF2E7D32),
          ),
          const SizedBox(height: 16),
          Text(
            wisata.name,
            style: GoogleFonts.poppins(
              color: const Color(0xFF2E7D32),
              fontWeight: FontWeight.bold,
              fontSize: 20,
            ),
            textAlign: TextAlign.center,
          ),
        ],
      ),
    );
  },
),
Container(
  decoration: BoxDecoration(
    gradient: LinearGradient(
      begin: Alignment.bottomCenter,
      end: Alignment.center,
      colors: [
        Colors.black.withOpacity(0.6),
        Colors.transparent,
      ],
    ),
  ),
),
),
),
],
),
),
leading: IconButton(
  icon: Container(
    padding: const EdgeInsets.all(8),
    decoration: BoxDecoration(
      color: Colors.black.withOpacity(0.5),
      shape: BoxShape.circle,
    ),
    child: const Icon(Icons.arrow_back, color: Colors.white),
  ),
  onPressed: () => Navigator.pop(context),
),
actions: [

```



```

IconButton(
  icon: Container(
    padding: const EdgeInsets.all(8),
    decoration: BoxDecoration(
      color: Colors.black.withOpacity(0.5),
      shape: BoxShape.circle,
    ),
    child: const Icon(Icons.favorite_border, color: Colors.white),
  ),
  onPressed: () {},
),
],
),
SliverToBoxAdapter(
  child: Padding(
    padding: const EdgeInsets.all(24),
    child: Column(
      crossAxisAlignment: CrossAxisAlignment.start,
      children: [
        Row(
          children: [
            Expanded(
              child: Text(
                wisata.name,
                style: GoogleFonts.poppins(
                  fontSize: 28,
                  fontWeight: FontWeight.w800,
                  color: const Color(0xFF1B5E20),
                ),
              ),
            ),
          ],
        ),
        Container(
          padding: const EdgeInsets.symmetric(horizontal: 12, vertical: 6),
          decoration: BoxDecoration(
            color: const Color(0xFF2E7D32),
            borderRadius: BorderRadius.circular(20),
          ),
          child: Row(
            children: [
              const Icon(Icons.star, color: Colors.amber, size: 16),
              const SizedBox(width: 4),
              Text(
                wisata.rating.toString(),
                style: GoogleFonts.poppins(
                  color: Colors.white,
                  fontWeight: FontWeight.w600,
                ),
              ),
            ],
          ),
        ),
        const SizedBox(height: 16),
        Row(
          children: [

```



```

Icon(Icons.location_on, color: Colors.grey[600], size: 20),
const SizedBox(width: 8),
Text(
  wisata.location,
  style: GoogleFonts.poppins(
    fontSize: 16,
    color: Colors.grey[600],
    fontWeight: FontWeight.w500,
  ),
),
const Spacer(),
Text(
  '${wisata.distance} km dari pusat kota',
  style: GoogleFonts.poppins(
    fontSize: 14,
    color: Colors.grey[500],
  ),
),
],
),
const SizedBox(height: 24),
Text(
  'Deskripsi',
  style: GoogleFonts.poppins(
    fontSize: 20,
    fontWeight: FontWeight.w700,
    color: const Color(0xFF2E7D32),
  ),
),
const SizedBox(height: 8),
Text(
  wisata.description,
  style: GoogleFonts.poppins(
    fontSize: 16,
    color: Colors.grey[700],
    height: 1.6,
  ),
),
const SizedBox(height: 24),
Text(
  'Fasilitas',
  style: GoogleFonts.poppins(
    fontSize: 20,
    fontWeight: FontWeight.w700,
    color: const Color(0xFF2E7D32),
  ),
),
const SizedBox(height: 12),
Wrap(
  spacing: 12,
  runSpacing: 12,
  children: wisata.facilities.map((facility) {
    return Container(
      padding: const EdgeInsets.symmetric(horizontal: 16, vertical: 8),
      decoration: BoxDecoration(
        color: const Color(0xFFE8F5E9),

```

```

        borderRadius: BorderRadius.circular(20),
      ),
      child: Text(
        facility,
        style: GoogleFonts.poppins(
          fontSize: 14,
          color: const Color(0xFF2E7D32),
          fontWeight: FontWeight.w500,
        ),
      ),
    ),
  ).toList(),
),
const SizedBox(height: 32),
SizedBox(
  width: double.infinity,
  child: ElevatedButton(
    onPressed: () {},
    style: ElevatedButton.styleFrom(
      backgroundColor: const Color(0xFF2E7D32),
      foregroundColor: Colors.white,
      shape: RoundedRectangleBorder(
        borderRadius: BorderRadius.circular(20),
      ),
      padding: const EdgeInsets.symmetric(vertical: 18),
      elevation: 4,
    ),
    child: Text(
      'Rencanakan Kunjungan',
      style: GoogleFonts.poppins(
        fontSize: 18,
        fontWeight: FontWeight.w600,
      ),
    ),
  ),
),
),
),
],
),
),
),
],
),
);
}
}

```

```

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

```

```

  @override
  State<WisataList> createState() => _WisataListState();
}

```

```

class _WisataListState extends State<WisataList> with TickerProviderStateMixin {
  late AnimationController _fadeController;
  late Animation<double> _fadeAnimation;

```

```
String _selectedKategori = 'Semua';
bool _isLoading = true;
final ScrollController _scrollController = ScrollController();
double _scrollOffset = 0.0;
String _searchQuery = '';
bool _isSearching = false;
int _currentHeaderImage = 0;
late Timer _headerTimer;

// Daftar gambar header yang akan berganti-ganti
final List<String> _headerImages = [
  'assets/images/wisataareabanyumas.png',
  'assets/images/baturraden.png',
  'assets/images/curugcipendok.png',
  'assets/images/alunalunpurwokerto.png',
];

List<Wisata> get _wisatas => [
  Wisata(
    name: 'Baturraden Adventure Forest',
    imageUrl: 'assets/images/baturraden.png',
    description: 'Surga wisata alam di lereng Gunung Slamet dengan pemandangan hijau pegunungan yang memukau. Menawarkan berbagai wahana outbound, pemandian air panas, dan tracking menuju puncak. Cocok untuk keluarga dan petualang.',
    location: 'Kec. Baturraden',
    rating: 4.8,
    category: 'Alam & Petualangan',
    distance: 12.5,
    reviewCount: 1247,
    facilities: ['Parkir Luas', 'Toilet', 'Restoran', 'Penginapan', 'Wahana Outbound'],
    price: 25000,
    isPopular: true,
    isFeatured: true,
  ),
  Wisata(
    name: 'Curug Cipendok',
    imageUrl: 'assets/images/curugcipendok.png',
    description: 'Air terjun spektakuler setinggi 92 meter yang dikelilingi hutan tropis lebat. Suara gemuruh air yang jatuh menciptakan atmosfer magis dan menenangkan. Perfect untuk fotografi dan meditasi.',
    location: 'Kec. Cilongok',
    rating: 4.6,
    category: 'Alam & Petualangan',
    distance: 8.2,
    reviewCount: 892,
    facilities: ['Parkir', 'Warung Makan', 'Area Camping', 'Spot Foto'],
    price: 15000,
    isPopular: true,
    isFeatured: false,
  ),
  Wisata(
    name: 'Alun-Alun Purwokerto',
    imageUrl: 'assets/images/alunalunpurwokerto.png',
    description: 'Jantung kota Purwokerto yang selalu hidup dengan berbagai aktivitas. Tempat perfect untuk menikmati kuliner khas Banyumas sambil merasakan denyut nadi kehidupan lokal dengan suasana yang asri dan nyaman.',
    location: 'Purwokerto Kota',
```

```

rating: 4.5,
category: 'Kuliner & Budaya',
distance: 0.5,
reviewCount: 1563,
facilities: ['Food Court', 'Area Bermain', 'Mushola', 'Toilet', 'Free WiFi', 'Taman'],
price: 0,
isPopular: true,
isFeatured: true,
),
Wisata(
  name: 'Kawasan Heritage Sokaraja',
  imageUrl: 'assets/images/GowaLawa.png',
  description: 'Jelajahi pesona masa lalu dengan arsitektur kolonial Belanda yang masih terpelihara. Jangan
lewatkan sate khas Sokaraja yang legendaris di pasar tradisionalnya.',
  location: 'Kec. Sokaraja',
  rating: 4.4,
  category: 'Sejarah & Kuliner',
  distance: 15.8,
  reviewCount: 734,
  facilities: ['Museum', 'Pasar Tradisional', 'Kuliner', 'Spot Sejarah'],
  price: 10000,
  isPopular: false,
  isFeatured: true,
),
Wisata(
  name: 'Telaga Sunyi',
  imageUrl: 'assets/images/telagasunyi.png',
  description: 'Danau permai yang tersembunyi di antara perbukitan, menawarkan ketenangan dan
kedamaian. Spot favorit untuk camping, fishing, dan menikmati sunrise yang memesona.',
  location: 'Kec. Kedungbanteng',
  rating: 4.7,
  category: 'Alam & Relaksasi',
  distance: 20.3,
  reviewCount: 567,
  facilities: ['Area Camping', 'Fishing', 'Warung', 'Toilet', 'Gazebo'],
  price: 12000,
  isPopular: true,
  isFeatured: false,
),
Wisata(
  name: 'Museum Wayang Banyumas',
  imageUrl: 'assets/images/musiumwayangbbanyumas.png',
  description: 'Menyimpan khazanah budaya Banyumas yang kaya, mulai dari koleksi wayang, batik khas,
hingga artefak sejarah. Pengalaman edukatif yang menyenangkan untuk semua usia.',
  location: 'Purwokerto Selatan',
  rating: 4.3,
  category: 'Edukasi & Budaya',
  distance: 2.1,
  reviewCount: 423,
  facilities: ['Parkir', 'Toilet', 'Guide Tour', 'Souvenir', 'Perpustakaan'],
  price: 5000,
  isPopular: false,
  isFeatured: false,
),
Wisata(
  name: 'Kebun Raya Baturraden',

```

```

    imageUrl: 'assets/images/kebunrayabatturaden.png',
    description: 'Surga bagi pecinta flora dengan koleksi tanaman langka dari seluruh Indonesia. Dilengkapi sky
    walk yang menawarkan pemandangan Gunung Slamet yang epik.',
    location: 'Kec. Baturraden',
    rating: 4.6,
    category: 'Alam & Edukasi',
    distance: 13.7,
    reviewCount: 678,
    facilities: ['Sky Walk', 'Rest Area', 'Toilet', 'Parkir', 'Kafe'],
    price: 20000,
    isPopular: true,
    isFeatured: true,
  ),
  Wisata(
    name: 'Pancuran Pitu',
    imageUrl: 'assets/images/pancuranpitu.png',
    description: 'Pemandian air panas alami dengan tujuh pancuran yang dipercaya memiliki khasiat untuk
    kesehatan. Suasana mistis dan alam yang masih asri membuatnya semakin menarik.',
    location: 'Kec. Baturraden',
    rating: 4.5,
    category: 'Alam & Kesehatan',
    distance: 14.2,
    reviewCount: 512,
    facilities: ['Kolam Renang', 'Pemandian Air Panas', 'Warung', 'Parkir', 'Ruang Ganti'],
    price: 18000,
    isPopular: false,
    isFeatured: true,
  ),
];

List<Wisata> get_filteredWisatas {
  var filtered = _wisatas;

  // Filter by category
  if (_selectedKategori != 'Semua') {
    filtered = filtered.where((w) => w.category == _selectedKategori).toList();
  }

  // Filter by search query
  if (_searchQuery.isNotEmpty) {
    filtered = filtered.where((w) =>
      w.name.toLowerCase().contains(_searchQuery.toLowerCase()) ||
      w.location.toLowerCase().contains(_searchQuery.toLowerCase()) ||
      w.category.toLowerCase().contains(_searchQuery.toLowerCase()) ||
      w.description.toLowerCase().contains(_searchQuery.toLowerCase())
    ).toList();
  }

  return filtered;
}

@override
void initState() {
  super.initState();

  _fadeController = AnimationController(
    duration: const Duration(milliseconds: 1200),

```

```
vsync: this,
);

_fadeAnimation = Tween<double>(begin: 0.0, end: 1.0).animate(
  CurvedAnimation(parent: _fadeController, curve: Curves.easeInOut),
);

_scrollController.addListener(() {
  setState() {
    _scrollOffset = _scrollController.offset;
  });
});

// Timer untuk mengganti gambar header setiap 5 detik
_headerTimer = Timer.periodic(const Duration(seconds: 5), (timer) {
  setState() {
    _currentHeaderImage = (_currentHeaderImage + 1) % _headerImages.length;
  });
});

// Simulate data loading
Future.delayed(const Duration(milliseconds: 1500), () {
  setState() {
    _isLoading = false;
  });
  _fadeController.forward();
});
}

@override
void dispose() {
  _fadeController.dispose();
  _scrollController.dispose();
  _headerTimer.cancel();
  super.dispose();
}

void _onKategoriChanged(String kategori) {
  setState() {
    _selectedKategori = kategori;
  });
}

void _onSearchChanged(String query) {
  setState() {
    _searchQuery = query;
  });
}

void _onSearchTap() {
  setState() {
    _isSearching = true;
  });
}

void _showDetail(Wisata wisata) {
```

```

Navigator.push(
  context,
  MaterialPageRoute(
    pageBuilder: (context, animation, secondaryAnimation) => DetailScreen(wisata: wisata),
    transitionsBuilder: (context, animation, secondaryAnimation, child) {
      return FadeTransition(
        opacity: animation,
        child: child,
      );
    },
    transitionDuration: const Duration(milliseconds: 600),
  ),
);
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    backgroundColor: const Color(0xFFFF8FDF8),
    body: _isLoading
      ? _buildLoadingScreen()
      : FadeTransition(
        opacity: _fadeAnimation,
        child: CustomScrollView(
          controller: _scrollController,
          physics: const BouncingScrollPhysics(),
          slivers: [
            _buildAppBar(),
            if (!_isSearching) _buildKategoriSection(),
            _buildSearchSection(),
            _buildHeaderSection(),
            _buildWisataList(),
          ],
        ),
      ),
    floatingActionButton: _buildFloatingActionButton(),
  );
}

Widget _buildLoadingScreen() {
  return Scaffold(
    backgroundColor: const Color(0xFFFF8FDF8),
    body: Center(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: [
          ShimmerEffect(
            isLoading: true,
            child: Container(
              width: 100,
              height: 100,
              decoration: BoxDecoration(
                gradient: const LinearGradient(
                  colors: [Color(0xFF4CAF50), Color(0xFF2E7D32)],
                  begin: Alignment.topLeft,
                  end: Alignment.bottomRight,
                ),
              ),
            ),
          ),
        ],
      ),
    ),
  );
}

```



```

    ),
    borderRadius: BorderRadius.circular(25),
    boxShadow: [
      BoxShadow(
        color: Colors.green.withOpacity(0.3),
        blurRadius: 15,
        offset: const Offset(0, 8),
      ),
    ],
  ),
  child: const Icon(
    Icons.landscape,
    size: 50,
    color: Colors.white,
  ),
),
const SizedBox(height: 30),
Text(
  'Memuat Keindahan Banyumas...',
  style: GoogleFonts.poppins(
    fontSize: 18,
    color: const Color(0xFF2E7D32),
    fontWeight: FontWeight.w600,
  ),
),
const SizedBox(height: 20),
SizedBox(
  width: 200,
  child: LinearProgressIndicator(
    backgroundColor: Colors.grey[200],
    valueColor: const AlwaysStoppedAnimation<Color>(Color(0xFF2E7D32)),
    borderRadius: BorderRadius.circular(10),
    minHeight: 6,
  ),
),
const SizedBox(height: 10),
Text(
  'Menyiapkan pengalaman terbaik untuk Anda',
  style: GoogleFonts.poppins(
    fontSize: 12,
    color: Colors.grey[600],
  ),
),
],
),
),
);
}

SliverAppBar _buildAppBar() {
  return SliverAppBar(
    expandedHeight: 350.0,
    floating: false,
    pinned: true,
    snap: false,
  );
}

```

```

stretch: true,
backgroundColor: const Color(0xFF2E7D32),
flexibleSpace: FlexibleSpaceBar(
  title: AnimatedOpacity(
    opacity: _scrollOffset > 100 ? 1.0 : 0.0,
    duration: const Duration(milliseconds: 300),
    child: Text(
      'Wisata Banyumas',
      style: GoogleFonts.poppins(
        color: Colors.white,
        fontWeight: FontWeight.bold,
        fontSize: 18,
      ),
    ),
  ),
),
background: AnimatedSwitcher(
  duration: const Duration(milliseconds: 1000),
  child: Stack(
    key: ValueKey(_currentHeaderImage),
    fit: StackFit.expand,
    children: [
      // Background Image dengan fade transition
      Image.asset(
        _headerImages[_currentHeaderImage],
        fit: BoxFit.cover,
        errorBuilder: (context, error, stackTrace) {
          return Container(
            color: const Color(0xFFE8F5E8),
            child: const Icon(
              Icons.landscape,
              size: 100,
              color: Color(0xFF2E7D32),
            ),
          );
        },
      ),
      // Gradient Overlay
      Container(
        decoration: BoxDecoration(
          gradient: LinearGradient(
            begin: Alignment.bottomCenter,
            end: Alignment.topCenter,
            colors: [
              const Color(0xFF2E7D32).withOpacity(0.95),
              const Color(0xFF2E7D32).withOpacity(0.3),
              Colors.transparent,
            ],
          ),
        ),
      ),
      // Content Overlay
      Positioned(
        bottom: 80,
        left: 20,

```

```

right: 20,
child: Column(
  crossAxisAlignment: CrossAxisAlignment.start,
  children: [
    // Animated Tagline
    AnimatedContainer(
      duration: const Duration(milliseconds: 500),
      padding: const EdgeInsets.symmetric(horizontal: 16, vertical: 8),
      decoration: BoxDecoration(
        color: Colors.white.withOpacity(0.2),
        borderRadius: BorderRadius.circular(20),
      ),
    ),
    child: Row(
      mainAxisAlignment: MainAxisAlignment.min,
      children: [
        const Icon(Icons.eco, color: Colors.white, size: 16),
        const SizedBox(width: 8),
        Text(
          _getTaglineForImage(_currentHeaderImage),
          style: GoogleFonts.poppins(
            color: Colors.white,
            fontSize: 14,
            fontWeight: FontWeight.w500,
          ),
        ),
      ],
    ),
    const SizedBox(height: 16),
    // Main Title
    Text(
      _getTitleForImage(_currentHeaderImage),
      style: GoogleFonts.poppins(
        color: Colors.white,
        fontSize: 32,
        fontWeight: FontWeight.w800,
        height: 1.2,
        shadows: [
          const Shadow(
            blurRadius: 10,
            color: Colors.black45,
            offset: Offset(2, 2),
          ),
        ],
      ),
    ),
    // Image Indicator
    Positioned(
      bottom: 20,
      left: 0,
      right: 0,
      child: Row(

```

```

    mainAxisAlignment: MainAxisAlignment.center,
    children: List.generate(_headerImages.length, (index) {
      return Container(
        width: 8,
        height: 8,
        margin: const EdgeInsets.symmetric(horizontal: 4),
        decoration: BoxDecoration(
          shape: BoxShape.circle,
          color: _currentHeaderImage == index
            ? Colors.white
            : Colors.white.withOpacity(0.5),
        ),
      );
    }),
  ),
],
),
stretchModes: const [
  StretchMode.zoomBackground,
  StretchMode.fadeTitle,
],
),
actions: [
  IconButton(
    icon: Container(
      padding: const EdgeInsets.all(8),
      decoration: BoxDecoration(
        color: Colors.white.withOpacity(0.2),
        shape: BoxShape.circle,
      ),
    ),
    child: const Icon(Icons.search, color: Colors.white, size: 22),
  ),
  onPressed: _onSearchTap,
),
  IconButton(
    icon: Container(
      padding: const EdgeInsets.all(8),
      decoration: BoxDecoration(
        color: Colors.white.withOpacity(0.2),
        shape: BoxShape.circle,
      ),
    ),
    child: const Icon(Icons.favorite_border, color: Colors.white, size: 22),
  ),
  onPressed: () {},
),
],
);
}

String _getTaglineForImage(int index) {
  switch (index) {
    case 0: return 'Alam • Petualangan • Budaya';
    case 1: return 'Gunung • Pemandian Air Panas • Hutan';
    case 2: return 'Air Terjun • Alam • Pendakian';
  }
}

```

```

    case 3: return 'Kota • Budaya • Kuliner';
    default: return 'Alam • Petualangan • Budaya';
  }
}

String _getTitleForImage(int index) {
  switch (index) {
    case 0: return 'Temukan Permata Tersembunyi\nBanyumas';
    case 1: return 'Jelajahi Petualangan\nPegunungan';
    case 2: return 'Mengejar Air Terjun\n& Alam';
    case 3: return 'Budaya Urban &\nKelezatan Lokal';
    default: return 'Temukan Permata Tersembunyi\nBanyumas';
  }
}

SliverToBoxAdapter _buildKategoriSection() {
  return SliverToBoxAdapter(
    child: KategoriFilter(
      selectedKategori: _selectedKategori,
      onKategoriChanged: _onKategoriChanged,
    ),
  );
}

SliverToBoxAdapter _buildSearchSection() {
  return SliverToBoxAdapter(
    child: SearchBar(
      onSearchChanged: _onSearchChanged,
      onSearchTap: _onSearchTap,
    ),
  );
}

SliverToBoxAdapter _buildHeaderSection() {
  final filteredCount = _filteredWisatas.length;
  final totalCount = _wisatas.length;

  return SliverToBoxAdapter(
    child: Padding(
      padding: const EdgeInsets.all(24.0),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [
          Row(
            children: [
              Container(
                padding: const EdgeInsets.all(12),
                decoration: BoxDecoration(
                  gradient: const LinearGradient(
                    colors: [Color(0xFF4CAF50), Color(0xFF2E7D32)],
                    begin: Alignment.topLeft,
                    end: Alignment.bottomRight,
                  ),
                borderRadius: BorderRadius.circular(16),
                boxShadow: [
                  BoxShadow(

```

```

        color: Colors.green.withOpacity(0.3),
        blurRadius: 10,
        offset: const Offset(0, 4),
      ),
    ],
  ),
  child: const Icon(Icons.auto_awesome, color: Colors.white, size: 28),
),
const SizedBox(width: 16),
Expanded(
  child: Column(
    crossAxisAlignment: CrossAxisAlignment.start,
    children: [
      Text(
        _searchQuery.isEmpty ? 'Rekomendasi Teratas' : 'Hasil Pencarian',
        style: GoogleFonts.poppins(
          fontSize: 24,
          fontWeight: FontWeight.w800,
          color: Colors.grey[800],
          height: 1.2,
        ),
      ),
    ],
  ),
  const SizedBox(height: 4),
  Text(
    _searchQuery.isEmpty
      ? 'Seleksi pilihan tempat wisata wajib kunjung di Banyumas'
      : 'Ditemukan $filteredCount dari $totalCount wisata',
    style: GoogleFonts.poppins(
      fontSize: 14,
      color: Colors.grey[600],
    ),
  ),
],
),
const SizedBox(height: 8),
],
),
);
}

SliverList _buildWisataList() {
  final filteredWisatas = _filteredWisatas;

  if (filteredWisatas.isEmpty) {
    return SliverList(
      delegate: SliverChildListDelegate([
        Padding(
          padding: const EdgeInsets.all(40),
          child: Column(
            children: [
              Icon(
                Icons.search_off,

```

```

        size: 80,
        color: Colors.grey[400],
      ),
      const SizedBox(height: 16),
      Text(
        'Tidak ada hasil ditemukan',
        style: GoogleFonts.poppins(
          fontSize: 18,
          fontWeight: FontWeight.w600,
          color: Colors.grey[600],
        ),
      ),
      const SizedBox(height: 8),
      Text(
        'Coba kata kunci lain atau kategori berbeda',
        style: GoogleFonts.poppins(
          fontSize: 14,
          color: Colors.grey[500],
        ),
        textAlign: TextAlign.center,
      ),
    ],
  ),
),
],
);
}

return SliverList(
  delegate: SliverChildBuilderDelegate(
    (context, index) {
      final wisata = filteredWisatas[index];
      return _buildWisataCard(wisata, index);
    },
    childCount: filteredWisatas.length,
  ),
);
}

Widget _buildWisataCard(Wisata wisata, int index) {
  return Container(
    margin: const EdgeInsets.fromLTRB(20, 8, 20, 16),
    child: Card(
      elevation: 8,
      shape: RoundedRectangleBorder(
        borderRadius: BorderRadius.circular(24),
      ),
      shadowColor: Colors.black.withOpacity(0.2),
      child: InkWell(
        onTap: () => _showDetail(wisata),
        borderRadius: BorderRadius.circular(24),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            Stack(
              children: [

```

```
// Main Image with Shimmer
ClipRRect(
  borderRadius: const BorderRadius.only(
    topLeft: Radius.circular(24),
    topRight: Radius.circular(24),
  ),
  child: ShimmerEffect(
    isLoading: _isLoading,
    child: SizedBox(
      height: 220,
      width: double.infinity,
      child: Image.asset(
        wisata.imageUrl,
        fit: BoxFit.cover,
        errorBuilder: (context, error, stackTrace) {
          return Container(
            height: 220,
            decoration: BoxDecoration(
              gradient: const LinearGradient(
                colors: [Color(0xFFE8F5E8), Color(0xFFC8E6C9)],
                begin: Alignment.topLeft,
                end: Alignment.bottomRight,
              ),
            ),
          ),
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            const Icon(
              Icons.landscape,
              size: 50,
              color: Color(0xFF2E7D32),
            ),
            const SizedBox(height: 8),
            Text(
              wisata.name,
              style: GoogleFonts.poppins(
                color: const Color(0xFF2E7D32),
                fontWeight: FontWeight.bold,
                fontSize: 16,
              ),
              textAlign: TextAlign.center,
            ),
          ],
        ),
      ),
    ),
  ),
),
);

// Gradient Overlay
Container(
  height: 220,
  decoration: BoxDecoration(
    borderRadius: const BorderRadius.only(
```



```

    topLeft: Radius.circular(24),
    topRight: Radius.circular(24),
  ),
  gradient: LinearGradient(
    begin: Alignment.bottomCenter,
    end: Alignment.topCenter,
    colors: [
      Colors.black.withOpacity(0.4),
      Colors.transparent,
    ],
  ),
),
),
),
),

// Rating Badge
Positioned(
  top: 16,
  right: 16,
  child: Container(
    padding: const EdgeInsets.symmetric(horizontal: 12, vertical: 6),
    decoration: BoxDecoration(
      color: Colors.white.withOpacity(0.95),
      borderRadius: BorderRadius.circular(20),
      boxShadow: [
        BoxShadow(
          color: Colors.black.withOpacity(0.1),
          blurRadius: 10,
          offset: const Offset(0, 4),
        ),
      ],
    ),
  ),
  child: Row(
    mainAxisAlignment: MainAxisAlignment.min,
    children: [
      const Icon(Icons.star, color: Colors.amber, size: 16),
      const SizedBox(width: 4),
      Text(
        wisata.rating.toString(),
        style: GoogleFonts.poppins(
          fontWeight: FontWeight.bold,
          fontSize: 14,
        ),
      ),
    ],
  ),
),
),
),

// Distance Badge
Positioned(
  top: 16,
  left: 16,
  child: Container(
    padding: const EdgeInsets.symmetric(horizontal: 12, vertical: 6),
    decoration: BoxDecoration(
      color: const Color(0xFF2E7D32).withOpacity(0.9),

```

Praktikum PPB

```
// Featured Badge
if (wisata.isFeatured)
  Positioned(
    top: 50,
    right: 16,
    child: Container(
      padding: const EdgeInsets.symmetric(horizontal: 10, vertical: 4),
      decoration: BoxDecoration(
        color: const Color(0xFFFFF6D00),
        borderRadius: BorderRadius.circular(12),
        boxShadow: [
          BoxShadow(
            color: const Color(0xFFFFF6D00).withOpacity(0.3),
            blurRadius: 8,
            offset: const Offset(0, 2),
          ),
        ],
      ),
    child: Row(
      mainAxisAlignment: MainAxisAlignment.min,
      children: [
        const Icon(Icons.featured_play_list, color: Colors.white, size: 12),
        const SizedBox(width: 4),
        Text(
          'FEATURED',
          style: GoogleFonts.poppins(
            color: Colors.white,
            fontSize: 10,
            fontWeight: FontWeight.w700,
          ),
        ),
      ],
    ),
  ),

// Category Badge
Positioned(
  bottom: 16,
  left: 16,
  child: Container(
    padding: const EdgeInsets.symmetric(horizontal: 12, vertical: 6),
    decoration: BoxDecoration(
      color: Colors.black.withOpacity(0.7),
      borderRadius: BorderRadius.circular(12),
    ),
    child: Text(
      wisata.category,
      style: GoogleFonts.poppins(
        color: Colors.white,
        fontSize: 12,
        fontWeight: FontWeight.w500,
      ),
    ),
  ),
),
```



```

    ],
  ),

  // Content
  Padding(
    padding: const EdgeInsets.all(20),
    child: Column(
      crossAxisAlignment: CrossAxisAlignment.start,
      children: [
        Text(
          wisata.name,
          style: GoogleFonts.poppins(
            fontSize: 22,
            fontWeight: FontWeight.w800,
            color: const Color(0xFF1B5E20),
            height: 1.2,
          ),
        ),
      ],
    ),

    const SizedBox(height: 8),

    Row(
      children: [
        Icon(Icons.location_on_outlined, color: Colors.grey[600], size: 16),
        const SizedBox(width: 6),
        Expanded(
          child: Text(
            wisata.location,
            style: GoogleFonts.poppins(
              fontSize: 14,
              color: Colors.grey[600],
              fontWeight: FontWeight.w500,
            ),
          ),
        ),
        const Spacer(),
        Text(
          '${wisata.reviewCount} reviews',
          style: GoogleFonts.poppins(
            fontSize: 12,
            color: Colors.grey[500],
          ),
        ),
      ],
    ),

    const SizedBox(height: 12),

    Text(
      wisata.description,
      style: GoogleFonts.poppins(
        fontSize: 14,
        color: Colors.grey[700],
        height: 1.5,
      ),
    ),
    maxLines: 2,

```

```

    overflow: TextOverflow.ellipsis,
  ),

  const SizedBox(height: 16),

  // Facilities
  Wrap(
    spacing: 8,
    runSpacing: 8,
    children: wisata.facilities.take(3).map((facility) {
      return Container(
        padding: const EdgeInsets.symmetric(horizontal: 10, vertical: 6),
        decoration: BoxDecoration(
          color: const Color(0xFFE8F5E8),
          borderRadius: BorderRadius.circular(12),
          boxShadow: [
            BoxShadow(
              color: Colors.black.withOpacity(0.05),
              blurRadius: 4,
              offset: const Offset(0, 2),
            ),
          ],
        ),
        child: Text(
          facility,
          style: GoogleFonts.poppins(
            fontSize: 12,
            color: const Color(0xFF2E7D32),
            fontWeight: FontWeight.w500,
          ),
        ),
      );
    }).toList(),
  ),

  const SizedBox(height: 16),

  // Price and Action Button
  Row(
    children: [
      if (wisata.price > 0)
        Container(
          padding: const EdgeInsets.symmetric(horizontal: 12, vertical: 6),
          decoration: BoxDecoration(
            color: const Color(0xFFFFF3E0),
            borderRadius: BorderRadius.circular(12),
            boxShadow: [
              BoxShadow(
                color: Colors.orange.withOpacity(0.1),
                blurRadius: 4,
                offset: const Offset(0, 2),
              ),
            ],
          ),
          child: Text(
            'Rp ${wisata.price.toInt()}',

```

Praktikum PPB

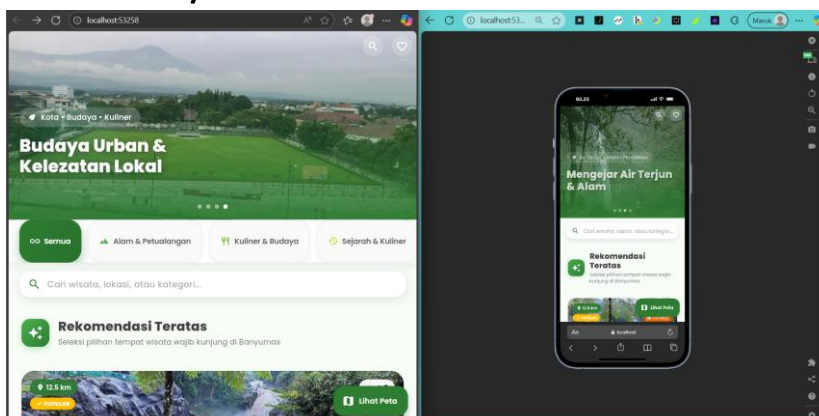
```

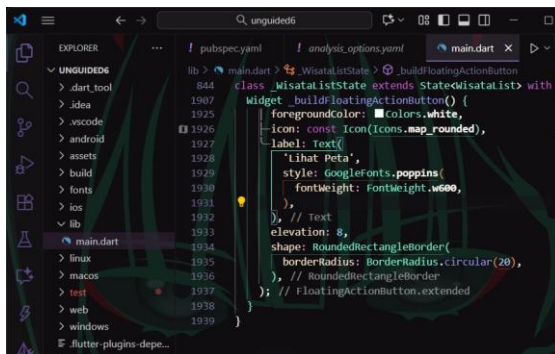
    ),
  ),
),
);
}

Widget _buildFloatingActionButton() {
  return FloatingActionButton.extended(
    onPressed: () {
      ScaffoldMessenger.of(context).showSnackBar(
        SnackBar(
          content: Text(
            'Fitur peta akan segera hadir!',
            style: GoogleFonts.poppins(),
          ),
          backgroundColor: const Color(0xFF2E7D32),
          behavior: SnackBarBehavior.floating,
          shape: RoundedRectangleBorder(
            borderRadius: BorderRadius.circular(12),
          ),
        ),
      );
    },
    backgroundColor: const Color(0xFF2E7D32),
    foregroundColor: Colors.white,
    icon: const Icon(Icons.map_rounded),
    label: Text(
      'Lihat Peta',
      style: GoogleFonts.poppins(
        fontWeight: FontWeight.w600,
      ),
    ),
    elevation: 8,
    shape: RoundedRectangleBorder(
      borderRadius: BorderRadius.circular(20),
    ),
  );
}
}

```

OUTPUT/SCREENSHOOT :





```
class WisatalistState extends State<Wisatalist> with
Widget _buildFloatingActionButton() {
  foregroundColor: Colors.white,
  icon: const Icon(Icons.map_rounded),
  label: Text(
    'Lihat Peta',
    style: GoogleFonts.poppins(
      fontWeight: FontWeight.w600,
    ), // Text
    elevation: 8,
    shape: RoundedRectangleBorder(
      borderRadius: BorderRadius.circular(20),
    ), // RoundedRectangleBorder
  ); // FloatingActionButton.extended
}
```

```
PS D:\PPB_2311104075_GANES_GEMI_PUTRA\06_Interaksi_Pengguna\UNGUIDED\unguided6> flutter run
device).
```

```
A Dart VM Service on Edge is available at:
http://127.0.0.1:59172/cDEl8Ci1EU4=
Starting application from main method in:
org-dartlang-app:/web_entrypoint.dart.
The Flutter DevTools debugger and profiler on
Edge is available at:
http://127.0.0.1:9102?uri=http://127.0.0.1:59
172/cDEl8Ci1EU4=
```