

**LAPORAN PRAKTIKUM
PEMROGRAMAN PERANGKAT BERGERAK**

**MODUL VII
NAVIGASI DAN NOTIFIKASI**



Disusun Oleh :

Dewi Atika Muthi / 2211104042

SE-06-02

Asisten Praktikum :

Muhammad Faza Zulian Gesit Al Barru

Aisyah Hasna Aulia

Dosen Pengampu :

Yudha Islami Sulistya, S.Kom., M.Cs

PROGRAM STUDI S1 SOFTWARE ENGINEERING

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

BAB I

PENDAHULUAN

A. DASAR TEORI

Dalam pengembangan aplikasi mobile menggunakan Flutter, navigasi dan notifikasi merupakan fitur yang sangat penting. Navigasi memungkinkan aplikasi untuk berpindah antar halaman atau tampilan secara dinamis. Flutter menggunakan `Navigator` sebagai widget utama untuk mengelola stack dari halaman-halaman (route). `Navigator.push` memungkinkan pengguna menuju halaman baru, sedangkan `Navigator.pop` memungkinkan kembali ke halaman sebelumnya. Selain itu, aplikasi sering membutuhkan fitur notifikasi untuk memberikan informasi atau mengingatkan pengguna terhadap suatu kejadian.

Package `flutter_local_notifications` digunakan untuk mengimplementasikan notifikasi lokal pada aplikasi Flutter, baik di Android maupun iOS. Dengan notifikasi, aplikasi dapat berinteraksi lebih aktif dengan pengguna, seperti mengirimkan peringatan atau mengundang pengguna kembali ke aplikasi.

B. MAKSUD DAN TUJUAN

Tujuan dari praktikum ini adalah:

1. Memahami konsep layout pada Flutter.
2. Mampu mengimplementasikan desain antarmuka pengguna (UI) pada Flutter.
3. Mengetahui cara membuat model data dalam aplikasi.
4. Mampu mengimplementasikan navigasi antar halaman beserta pengiriman data antar halaman.
5. Mampu mengimplementasikan fitur notifikasi lokal pada aplikasi Flutter

BAB II IMPLEMENTASI

A. PRAKTIKUM

1. Membuat Model Class

Soal Studi Case

Model class merupakan representasi struktur data yang akan digunakan dalam aplikasi. Misalnya, model Album yang menyimpan informasi seperti `userId`, `id`, dan `title`.

Sourcecode

```
class Album {
    final int userId;
    final int id;
    final String title;

    const Album({
        required this.userId,
        required this.id,
        required this.title,
    });

    factory Album.fromJson(Map<String, dynamic> json) {
        return Album(
            userId: json['user_id'],
            id: json['id'],
            title: json['title'],
        );
    }
}
```

Deskripsi Program

Kelas Album berfungsi sebagai model data untuk menampung informasi album yang diterima dari sumber data seperti API dalam format JSON

2. Navigation Pindah Halaman

Soal Studi Case

Dalam Flutter, navigasi antar halaman dapat dilakukan dengan memanggil `Navigator.push()` untuk berpindah ke halaman baru, sedangkan `Navigator.pop()` untuk kembali ke halaman sebelumnya.

Sourcecode

```
Navigator.push(
    context,
    MaterialPageRoute(builder: (context) => SecondRoute()),
);
```

Deskripsi Program

Kode di atas melakukan navigasi dari halaman saat ini menuju halaman `SecondRoute`. Hal ini memudahkan pengguna untuk berpindah dari satu halaman ke halaman lain sesuai dengan alur aplikasi

3. Navigation Mengirim Data

Soal Studi Case

Navigasi pada Flutter juga dapat digunakan untuk mengirimkan data dari satu halaman ke halaman lainnya dengan menambahkan parameter pada konstruktor **halaman tujuan**.

Sourcecode

```
Navigator.push(
  context,
  MaterialPageRoute(builder: (context) => DetailScreen(title: "Detail
    User")),
);

class DetailScreen extends StatelessWidget {
  final String title;

  const DetailScreen({Key? key, required this.title}) : super(key:
    key);

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text(title),
      ),
    );
  }
}
```

Deskripsi Program

Kode ini mengarahkan pengguna ke halaman `DetailScreen` dan mengirimkan data berupa `title`. Di halaman `DetailScreen`, data `title` digunakan untuk menampilkan judul pada `AppBar`, sehingga pengguna mendapatkan informasi kontekstual yang jelas di halaman baru

4. Notification Menggunakan flutter_local_notifications

Soal Studi Case

Deskripsi Materi: Notifikasi lokal pada Flutter diimplementasikan dengan package `flutter_local_notifications`. Untuk menggunakan package ini, aplikasi harus melakukan konfigurasi pada `pubspec.yaml`, `AndroidManifest.xml`, serta izin notifikasi untuk platform Android dan iOS.

Sourcecode (pubspec.yaml)

```
..
dependencies:
  permission_handler: ^10.2.0
```

```

        flutter:
          sdk: flutter

        flutter_local_notifications: ^17.2.4
    ..

```

Sourcecode (AndroidManifest.xml)

```

    ..
    <uses-permission android:name="android.permission.VIBRATE" />
    <uses-permission
    android:name="android.permission.RECEIVE_BOOT_COMPLETED"/>
    <uses-permission
    android:name="android.permission.POST_NOTIFICATIONS" />
    ..

```

Sourcecode

```

FlutterLocalNotificationsPlugin flutterLocalNotificationsPlugin
= FlutterLocalNotificationsPlugin();

void showNotification() async {
    var android = AndroidNotificationDetails(
        'id', 'channel', 'description',
        priority: Priority.high, importance: Importance.max);
    var iOS = IOSNotificationDetails();
    var platform = NotificationDetails(android: android, iOS:
iOS);
    await flutterLocalNotificationsPlugin.show(
        0, 'Flutter Notification', 'This is a test notification',
        platform,
        payload: 'Welcome to Flutter');
}

```

Deskripsi Program

Dengan `showNotification()`, aplikasi dapat menampilkan notifikasi sederhana pada perangkat Android dan iOS. Notifikasi ini akan muncul dengan judul "Flutter Notification" dan pesan "This is a test notification". Package ini memungkinkan pengaturan prioritas, jenis notifikasi, dan tampilan notifikasi sesuai kebutuhan platform

5. CODE PROGRAM LENGKAP

Soal Studi Case

Membuat program untuk menampilkan list barang toko perlengkapan gaming, yang saat diklik akan muncul notifikasi.

Sourcecode (lib/models/product.dart)

```

class Product {
    final int id;
    final String nama;
    final double harga;
    final String gambarUrl;
    final String deskripsi;

    // Constructor
    Product({
        required this.id,
        required this.nama,

```

```

        required this.harga,
        required this.gambarUrl,
        required this.deskripsi,
    });

    // Method untuk mengonversi Json => Object Product
    factory Product.fromJson(Map<String, dynamic> json) {
        return Product(
            id: json['id'],
            nama: json['nama'],
            harga: json['harga'],
            gambarUrl: json['gambarUrl'],
            deskripsi: json['deskripsi'],
        );
    }

    // Method untuk mengonversi Object Product => Json
    Map<String, dynamic> toJson() {
        return {
            "id": id,
            "nama": nama,
            "harga": harga,
            "gambarUrl": gambarUrl,
            "deskripsi": deskripsi,
        };
    }
}

```

Deskripsi program:

File ini mendefinisikan kelas `Product` untuk merepresentasikan data produk dengan atribut:

- `id`: ID unik produk
- `nama`: Nama produk
- `harga`: Harga produk
- `gambarUrl`: URL gambar produk
- `deskripsi`: Deskripsi produk

Kelas ini juga memiliki metode `fromJson` untuk konversi dari JSON ke objek `Product` dan `toJson` untuk konversi sebaliknya.

Sourcecode (lib/pages/detailpagi.dart)

```

import 'package:flutter/material.dart';

class DetailPage extends StatelessWidget {
    const DetailPage({super.key, required this.data});

    final Widget data;

    @override
    Widget build(BuildContext context) {
        return Scaffold(
            appBar: AppBar(
                title: Text('Detail Page'),
                centerTitle: true,
                backgroundColor: Colors.pink[200],
            ),
            body: Center(
                child: data,
            ),
        );
    }
}

```

Deskripsi Program:

DetailPage adalah halaman yang **menampilkan detail produk tertentu** ketika **produk diklik**. Halaman ini menampilkan nama produk sebagai contoh informasi yang ditampilkan di detail.

Pada halaman ini, AppBar diberi judul "Detail Page" dan warna latar belakang AppBar diatur menjadi pink muda.

Sourcecode (lib/pages/mypage.dart)

```
import 'package:flutter/material.dart';
import 'package:md7/models/product.dart';
import 'package:md7/pages/detailpage.dart';
import 'package:md7/main.dart'; // Import main.dart untuk akses
                                showNotification

class MyPage extends StatelessWidget {
  MyPage({super.key});

  final List<Product> products = [
    Product(
      id: 1,
      nama: "Mouse",
      harga: 450000.00,
      gambarUrl:

        'https://resource.logitechg.com/w_386,ar_1.0,c_limit,f_auto,q_au
to,dpr_2.0/d_transparent.gif/content/dam/gaming/en/products/g502
x-plus/gallery/g502x-plus-gallery-1-black.png?v=1',
      deskripsi: 'Mouse Gaming yang lagi tren bost',
    ),
    Product(
      id: 2,
      nama: "Keyboard Mechanical",
      harga: 560000.00,
      gambarUrl:

        'https://resource.logitech.com/w_1600,c_limit,q_auto,f_auto,dpr_
1.0/d_transparent.gif/content/dam/logitech/en/products/keyboards
/mx-mechanical/gallery/mx-mechanical-keyboard-top-view-graphite-
us.png?v=1',
      deskripsi: 'Keyboard mechanical kece lu pasti demen',
    ),
    Product(
      id: 3,
      nama: "Headphone Gaming",
      harga: 890000.00,
      gambarUrl:
        'https://m.media-
amazon.com/images/I/61CGHv6kmWL.AC_UF894,1000_QL80.jpg',
      deskripsi: 'Headphone gaming kece temen lu pasti iri',
    ),
  ];

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text("Class Model"),
        centerTitle: true,
        backgroundColor: Colors.pink[100],
      ),
      body: ListView.builder(
        itemCount: products.length,
```

```

        itemBuilder: (context, index) {
          final product = products[index];
          return ListTile(
            leading: Image.network(
              product.gambarUrl,
              width: 70,
              height: 70,
            ),
            title: Text(product.nama),
            subtitle: Column(
              crossAxisAlignment: CrossAxisAlignment.start,
              children: [
                Text('Rp${product.harga}'),
                Text('Deskripsi: ${product.deskripsi}'),
              ],
            ),
            onTap: () {
              // Memanggil notifikasi saat item diklik
              showNotification(product.nama, "Harga:
              Rp${product.harga}");

              // Navigasi ke DetailPage
              Navigator.push(
                context,
                MaterialPageRoute(
                  builder: (_) => DetailPage(
                    data: Text(product.nama),
                  ),
                ),
              );
            },
          );
        },
      ),
    );
  },
);
}

```

Deskripsi program:

Halaman ini menampilkan daftar produk menggunakan `ListView.builder`. Untuk setiap produk, ditampilkan gambar, nama, harga, dan deskripsi produk.

Ketika pengguna mengklik produk tertentu, program melakukan dua hal:

- Memanggil fungsi `showNotification` untuk menampilkan notifikasi dengan judul nama produk dan isi berupa harga produk.
- Mengarahkan pengguna ke halaman `DetailPage` yang menampilkan nama produk.

Sourccode (lib/main.dart)

```

import 'package:flutter/material.dart';
import 'package:flutter_local_notifications/flutter_local_notifications.dart';
import 'package:md7/pages/mypage.dart';
import 'package:permission_handler/permission_handler.dart';

final FlutterLocalNotificationsPlugin flutterLocalNotificationsPlugin =
  FlutterLocalNotificationsPlugin();

// Pindahkan fungsi showNotification ke luar main agar dapat diakses di seluruh
// aplikasi
Future<void> showNotification(String title, String body) async {
  const AndroidNotificationDetails androidPlatformChannelSpecifics =
    AndroidNotificationDetails(
      'yovvv', // ID unik untuk channel
      'your_channel_name', // Nama channel
    );
}

```



```

        channelDescription: 'your_channel_description', // Deskripsi channel
        importance: Importance.max,
        priority: Priority.high,
        showWhen: true,
    );

    const NotificationDetails platformChannelSpecifics = NotificationDetails(
      android: androidPlatformChannelSpecifics,
    );

    await flutterLocalNotificationsPlugin.show(
      0, // ID notifikasi
      title, // Judul notifikasi
      body, // Isi notifikasi
      platformChannelSpecifics,
    );
  }

  void main() async {
    WidgetsFlutterBinding.ensureInitialized();
    // Minta izin notifikasi untuk Android 13 dan lebih tinggi
    if (await Permission.notification.isDenied) {
      await Permission.notification.request();
    }

    // Inisialisasi notifikasi lokal
    await InitializationSettings();

    const AndroidInitializationSettings initializationSettingsAndroid =
      AndroidInitializationSettings('@mipmap/ic_launcher');

    const DarwinInitializationSettings initializationSettingsIOS =
      DarwinInitializationSettings();

    const InitializationSettings initializationSettings = InitializationSettings(
      android: initializationSettingsAndroid,
      iOS: initializationSettingsIOS,
    );

    await flutterLocalNotificationsPlugin.initialize(initializationSettings);

    runApp(MyApp());
  }

  class MyApp extends StatelessWidget {
    @override
    Widget build(BuildContext context) {
      return MaterialApp(
        title: 'Flutter Local Notifications',
        theme: ThemeData(primarySwatch: Colors.blue),
        home: MyPage(),
      );
    }
  }

```

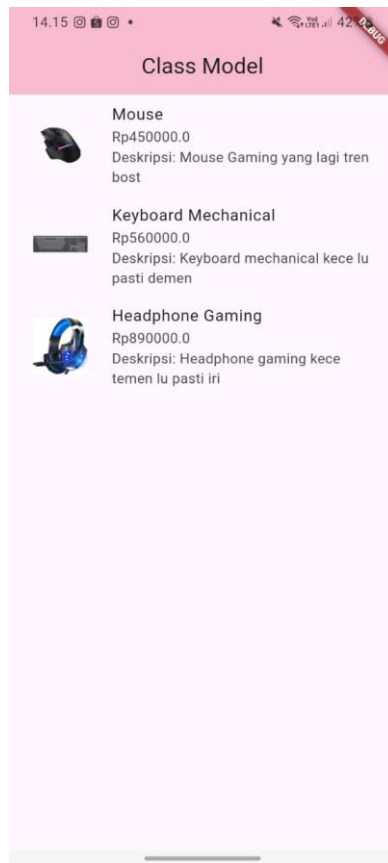
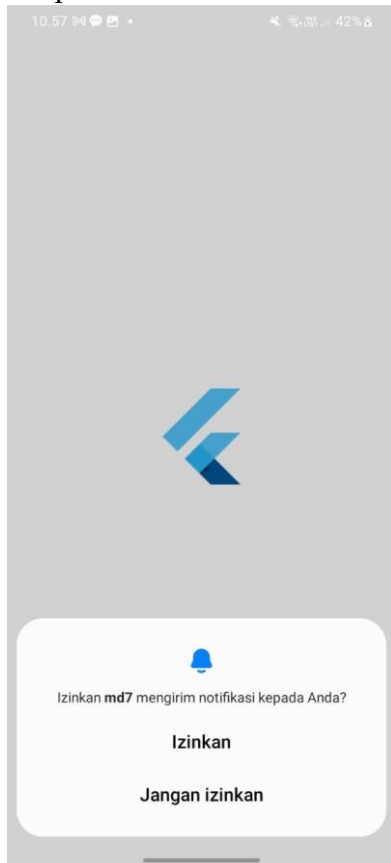
Deskripsi Program

Program menggunakan `flutter_local_notifications` untuk menampilkan notifikasi ketika pengguna memilih produk.

Fungsi `showNotification` digunakan untuk menampilkan notifikasi dengan konfigurasi saluran notifikasi, berisi judul (nama produk) dan pesan (harga produk).

Program juga meminta izin notifikasi saat pertama kali dijalankan, terutama untuk perangkat dengan Android 13 atau yang lebih tinggi.

Output:



B. UNGUIDED

1. Buatlah satu project untuk menampilkan beberapa produk dan halaman e-commerce dengan menerapkan class model serta navigasi halaman.

Soal Studi Case

Membuat project e-commerce kosmetik dengan Flutter.

Sourcecode (models/product.dart)

```
class Product {
  final int id;
  final String name;
  final double price;
  final String imageUrl;
  final String description;
  final String category;
  final double rating;

  Product({
    required this.id,
    required this.name,
    required this.price,
    required this.imageUrl,
    required this.description,
    required this.category,
    required this.rating,
  });
}
```

Sourcecode (pages/cart_page.dart)

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

class CartPage extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Shopping Cart'),
        backgroundColor: Colors.pink[100],
      ),
      body: Center(
        child: Text('Cart is empty'),
      ),
    );
  }
}
```

Sourcecode (pages/detail_page.dart)

```
import 'package:flutter/material.dart';
import 'package:unmd7/models/product.dart';
import 'package:unmd7/main.dart';

class DetailPage extends StatelessWidget {
  final Product product;

  const DetailPage({required this.product});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Product Detail'),
```

```

        backgroundColor: Colors.pink[100],
      ),
      body: SingleChildScrollView(
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            Image.network(
              product.imageUrl,
              height: 300,
              width: double.infinity,
              fit: BoxFit.cover,
            ),
            Padding(
              padding: EdgeInsets.all(16),
              child: Column(
                crossAxisAlignment: CrossAxisAlignment.start,
                children: [
                  Text(
                    product.name,
                    style: TextStyle(
                      fontSize: 24,
                      fontWeight: FontWeight.bold,
                    ),
                  ),
                  SizedBox(height: 8),
                  Text(
                    'Rp ${product.price.toStringAsFixed(0)}',
                    style: TextStyle(
                      fontSize: 20,
                      color: Colors.pink,
                      fontWeight: FontWeight.bold,
                    ),
                  ),
                  SizedBox(height: 16),
                  Row(
                    children: [
                      Icon(Icons.star, color: Colors.amber),
                      SizedBox(width: 4),
                      Text(
                        '${product.rating}',
                        style: TextStyle(fontSize: 16),
                      ),
                    ],
                  ),
                  SizedBox(height: 16),
                  Text(
                    'Description',
                    style: TextStyle(
                      fontSize: 18,
                      fontWeight: FontWeight.bold,
                    ),
                  ),
                  SizedBox(height: 8),
                  Text(
                    product.description,
                    style: TextStyle(fontSize: 16),
                  ),
                ],
              ),
            ),
          ],
        ),
      ),
    ),
  ),
  bottomNavigationBar: Padding(
    padding: EdgeInsets.all(16),
    child: ElevatedButton(
      onPressed: () {

```

```

// Tampilkan notifikasi ketika produk ditambahkan ke
keranjang
    showNotification(
        "Added to Cart",
        "${product.name} has been added to your cart",
    );
},
child: Text(
    'Add to Cart',
    style: TextStyle(fontSize: 16),
),
style: ElevatedButton.styleFrom(
    backgroundColor: Colors.pink[300],
    padding: EdgeInsets.symmetric(vertical: 16),
),
),
),
);
}
}

```

Sourcecode (pages/home_page.dart)

```

import 'package:flutter/material.dart';
import 'package:unmd7/models/product.dart';
import 'package:unmd7/pages/detail_page.dart';
import 'package:unmd7/pages/cart_page.dart';

class HomePage extends StatelessWidget {
  final List<Product> products = [
    Product(
      id: 1,
      name: "Lipstick Matte Rose",
      price: 129000,
      imageUrl:
        "https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcTazzqNSr7CLOdFn0MB5MVoeMquNYRM9aF
gQ&s",
      description:
        "Lipstick matte dengan formula tahan lama, mengandung
vitamin E untuk melembabkan bibir.",
      category: "Lips",
      rating: 4.8,
    ),
    Product(
      id: 2,
      name: "Foundation Natural Glow",
      price: 289000,
      imageUrl:
        "https://lumene.com/cdn/shop/files/297_35aa0be2-1dee-443f-
8d19-eaf42b62845a.jpg?v=1704168239&width=3000",
      description:
        "Foundation dengan coverage medium to full, memberikan hasil
akhir natural glowing.",
      category: "Face",
      rating: 4.7,
    ),
    Product(
      id: 3,
      name: "Mascara Waterproof",
      price: 159000,
      imageUrl:
        "https://www.static-
src.com/wcsstore/Indraprastha/images/catalog/full/catalog-image/MTA-
109085722/no_brand_esenses_mascara_waterproof_volume_8_ml_full101_p0ta0
9pl.jpg",
      description:

```

```

        "Mascara waterproof yang membuat bulu mata lebih tebal dan
panjang.",
        category: "Eyes",
        rating: 4.6,
    ),
    Product(
        id: 4,
        name: "Matte Cushion",
        price: 149000,
        imageUrl:
            "https://img.ws.mms.shopee.co.id/id-11134207-7r98w-
lwwzruhwlkrf4f",
        description:
            "Matte cushion yang cocok untuk kulit oily kamu dan bisa
menahan kulit berminyakmu 24 jam!",
        category: "Eyes",
        rating: 4.6,
    ),
];

@override
Widget build(BuildContext context) {
    return Scaffold(
        appBar: AppBar(
            title: Text('Beauty Shop'),
            backgroundColor: Colors.pink[100],
            actions: [
                IconButton(
                    icon: Icon(Icons.shopping_cart),
                    onPressed: () {
                        Navigator.push(
                            context,
                            MaterialPageRoute(builder: (context) => CartPage()),
                        );
                    },
                ),
            ],
        ),
        body: GridView.builder(
            padding: EdgeInsets.all(16),
            gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(
                crossAxisCount: 2,
                childAspectRatio: 0.75,
                crossAxisSpacing: 16,
                mainAxisSpacing: 16,
            ),
            itemCount: products.length,
            itemBuilder: (context, index) {
                return ProductCard(product: products[index]);
            },
        ),
    );
}

class ProductCard extends StatelessWidget {
    final Product product;

    const ProductCard({required this.product});

    @override
    Widget build(BuildContext context) {
        return GestureDetector(
            onTap: () {
                Navigator.push(
                    context,
                    MaterialPageRoute(
                        builder: (context) => DetailPage(product: product),
                    ),
                );
            },
        );
    }
}

```

```

    ),
  );
},
child: Card(
  elevation: 4,
  shape: RoundedRectangleBorder(
    borderRadius: BorderRadius.circular(12),
  ),
  child: Column(
    crossAxisAlignment: CrossAxisAlignment.start,
    children: [
      Expanded(
        child: ClipRRect(
          borderRadius: BorderRadius.vertical(top:
Radius.circular(12)),
          child: Image.network(
            product.imageUrl,
            fit: BoxFit.cover,
            width: double.infinity,
          ),
        ),
      ),
    ],
  ),
  Padding(
    padding: EdgeInsets.all(8),
    child: Column(
      crossAxisAlignment: CrossAxisAlignment.start,
      children: [
        Text(
          product.name,
          style: TextStyle(
            fontWeight: FontWeight.bold,
            fontSize: 14,
          ),
          maxLines: 2,
          overflow: TextOverflow.ellipsis,
        ),
        SizedBox(height: 4),
        Text(
          'Rp ${product.price.toStringAsFixed(0)}',
          style: TextStyle(
            color: Colors.pink,
            fontWeight: FontWeight.bold,
          ),
        ),
      ],
    ),
  ),
],
),
),
),
),
),
);
}
}

```

Sourcecode (main.dart)

```

import 'package:flutter/material.dart';
import
'package:flutter_local_notifications/flutter_local_notifications.dart'
;
import 'package:unmd7/pages/home_page.dart';
import 'package:permission_handler/permission_handler.dart';

final FlutterLocalNotificationsPlugin flutterLocalNotificationsPlugin
=
FlutterLocalNotificationsPlugin();

Future<void> showNotification(String title, String body) async {

```

```

const AndroidNotificationDetails androidPlatformChannelSpecifics =
  AndroidNotificationDetails(
    'beauty_shop',
    'Beauty Shop Notifications',
    channelDescription: 'Notifications from Beauty Shop',
    importance: Importance.max,
    priority: Priority.high,
    showWhen: true,
  );
const NotificationDetails platformChannelSpecifics =
  NotificationDetails(android: androidPlatformChannelSpecifics);

await flutterLocalNotificationsPlugin.show(
  0,
  title,
  body,
  platformChannelSpecifics,
);
}

void main() async {
  WidgetsFlutterBinding.ensureInitialized();

  if (await Permission.notification.isDenied) {
    await Permission.notification.request();
  }

  const AndroidInitializationSettings initializationSettingsAndroid =
    AndroidInitializationSettings('@mipmap/ic_launcher');
  const DarwinInitializationSettings initializationSettingsIOS =
    DarwinInitializationSettings();
  const InitializationSettings initializationSettings =
  InitializationSettings(
    android: initializationSettingsAndroid,
    iOS: initializationSettingsIOS,
  );

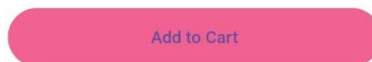
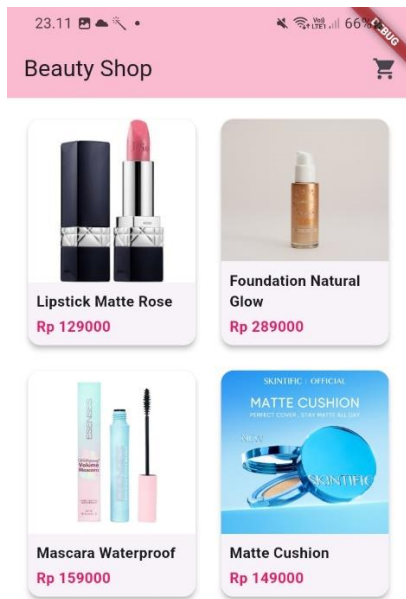
  await
flutterLocalNotificationsPlugin.initialize(initializationSettings);

  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Beauty Shop',
      theme: ThemeData(
        primarySwatch: Colors.pink,
        scaffoldBackgroundColor: Colors.white,
      ),
      home: HomePage(),
    );
  }
}

```


Screenshot Output



Deskripsi Program

Proyek ini memiliki fitur-fitur berikut:

Halaman Utama (`HomePage`):

- Menampilkan grid produk kosmetik
- Setiap produk ditampilkan dalam card dengan gambar dan informasi dasar
- Tombol keranjang di `AppBar`

Halaman Detail (`DetailPage`):

- Menampilkan gambar produk lebih besar
- Informasi lengkap produk (nama, harga, rating, deskripsi)
- Tombol "Add to Cart" yang memunculkan notifikasi

Halaman Keranjang (`CartPage`):

- Halaman basic untuk pengembangan fitur keranjang, belum memiliki back-end

Fitur Notifikasi:

- Muncul ketika produk ditambahkan ke keranjang
- Meminta izin notifikasi saat aplikasi pertama kali dibuka

Model Data:

- Class `Product` dengan properti lengkap untuk setiap produk

BAB III

KESIMPULAN DAN SARAN

A. KESIMPULAN

Dari praktikum ini, kami mempelajari konsep dasar navigasi dan notifikasi pada Flutter. Navigasi membantu aplikasi untuk berpindah antar halaman dan mengirim data antar halaman. Notifikasi bisa membuat aplikasi untuk berinteraksi dengan pengguna di luar aplikasi. Dengan menggunakan `flutter_local_notifications`, notifikasi dapat ditampilkan pada Android dan iOS, memberikan fleksibilitas dalam penyampaian informasi kepada pengguna. Praktikum ini juga menekankan pentingnya pengelolaan data melalui model yang terstruktur, seperti Album dan Product, untuk memudahkan pengelolaan dan pemrosesan data dalam aplikasi. Dengan memahami dan menerapkan konsep-konsep ini, aplikasi yang dihasilkan menjadi lebih interaktif, informatif, dan user-friendly.

B. REFERENSI

Dokumentasi Flutter. <https://flutter.dev/docs>

Package Flutter Local Notifications. https://pub.dev/packages/flutter_local_notifications