TUGAS PENDAHULUAN / TUGAS UNGUIDED
PEMROGRAMAN PERANGKAT BERGERAK

MODUL XII

MAPS AND PLACES

Disusun Oleh :

Farhan Kurniawan / 2311104073

SE-07-02

Asisten Praktikum :

Zulfa Mustafa Akhyar Iswahyudi
Yoga Eka Pratama

Dosen Pengampu :

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

PROGRAM STUDI S1 SOFTWARE ENGINEERING

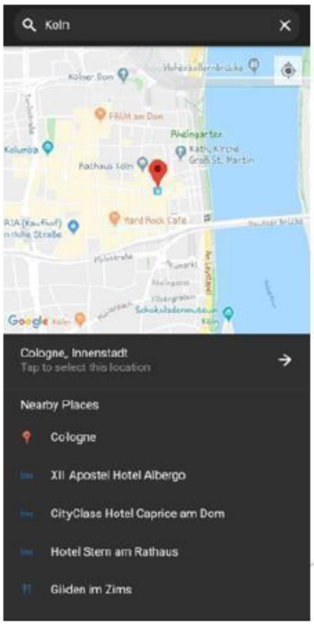
FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2025

TUGAS PENDAHULUAN / TUGAS UNGUIDED

A. SOAL



Cologne, Innenstadt
Tap to select this location

Nearby Places

Cologne

XII Apostel Hotel Albergo

CityClass Hotel Caprice am Dom

Hotel Stern am Rathaus

Gilden im Zims

Tugas Mandiri (Unguided)

Dari tugas guided yang telah dikerjakan, lanjutkan hingga ke bagian place picker untuk memberikan informasi mengenai lokasi yang ditunjuk di peta.

Note: Jangan lupa sertakan source code, screenshoot output, dan deskripsi program. Kreatifitas menjadi nilai tambah.

B. JAWABAN

Source Code pada Main Dart

```
//render_maps.dart
import 'package:flutter/material.dart';
import 'package:google_maps_flutter/google_maps_flutter.dart';
import 'package:place_picker_google/place_picker_google.dart';
import 'place_picker.dart';

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

  @override
  State<RenderMaps> createState() => _RenderMapsState();
}

class _RenderMapsState extends State<RenderMaps> {
  static final LatLng _defaultLocation = const LatLng(
    -7.421825325207339,
    109.24125221012827,
  );
  static final CameraPosition _cameraPosition = CameraPosition(
    target: _defaultLocation,
    zoom: 14,
  );

  LocationResult? _picked;
  final Set<Marker> _markers = {};

  void _openPlacePicker() async {
    final LocationResult? result = await Navigator.push(
      context,
      MaterialPageRoute(builder: (_) => const PlacePickerScreen()),
    );
    if (result != null) {
      setState(() {
```

```
      _picked = result;
      final LatLng? pos = result.latLng;
      _markers.clear();
      if (pos != null) {
        _markers.add(
          Marker(markerId: const MarkerId('picked'), position: pos),
        );
      }
    });
  }
}

Widget _buildInfoCard() {
  if (_picked == null) return const SizedBox.shrink();
  final LatLng? pos = _picked!.latLng;
  final String title =
      _picked!.name ?? _picked!.formattedAddress ?? 'Lokasi Terpilih';
  return Container(
    margin: const EdgeInsets.all(16),
    padding: const EdgeInsets.all(16),
    decoration: BoxDecoration(
      color: Colors.white,
      borderRadius: BorderRadius.circular(12),
      boxShadow: const [BoxShadow(color: Colors.black12, blurRadius: 8)],
    ),
    child: Column(
      crossAxisAlignment: CrossAxisAlignment.start,
      mainAxisSize: MainAxisSize.min,
      children: [
        Text(
          title,
          style: const TextStyle(fontSize: 16, fontWeight: FontWeight.bold),
        ),
        const SizedBox(height: 8),
        if (_picked!.formattedAddress != null)
```

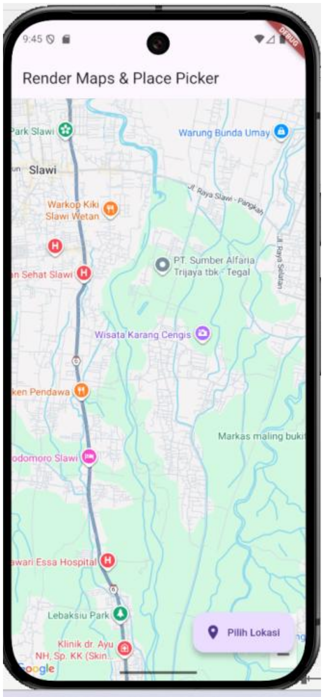
```
            Text(_picked!.formattedAddress!),
          if (pos != null) ...[
            const SizedBox(height: 8),
            Text(
              'Koordinat: ${pos.latitude.toStringAsFixed(6)}, '
              '${pos.longitude.toStringAsFixed(6)}',
            ),
          ],
        ],
      ),
    );
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: const Text('Render Maps & Place Picker')),
      body: Stack(
        children: [
          GoogleMap(
            initialCameraPosition: _cameraPosition,
            markers: _markers,
            myLocationButtonEnabled: true,
          ),
          Positioned(left: 0, right: 0, bottom: 0, child: _buildInfoCard()),
        ],
      ),
      floatingActionButton: FloatingActionButton.extended(
        onPressed: _openPlacePicker,
        icon: const Icon(Icons.place),
        label: const Text('Pilih Lokasi'),
      ),
    );
  }
}
```

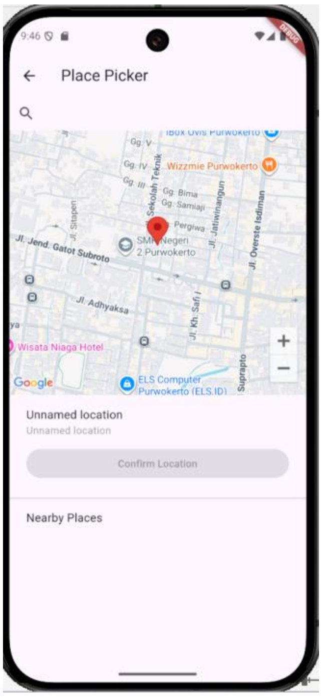
```
// place_picker.dart
import 'package:flutter/material.dart';
import 'package:google_maps_flutter/google_maps_flutter.dart';
import 'package:place_picker_google/place_picker_google.dart';

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

  @override
  State<PlacePickerScreen> createState() => _PlacePickerScreenState();
}

class _PlacePickerScreenState extends State<PlacePickerScreen> {
  static final LatLng _defaultLocation = const LatLng(
    -7.421825325207339,
    109.24125221012827,
  );



  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: const Text('Place Picker')),
      body: PlacePicker(
        apiKey: "AIzaSyAhLKzV0RnTmIyg1LM3Gbo62tucGfuZTn8",
        initialLocation: _defaultLocation,
        onPlacePicked: (LocationResult result) {
          Navigator.pop(context, result);
        },
        enableNearbyPlaces: true,
      ),
    );
  }
}
```

```
//connection.dart
import 'package:supabase_flutter/supabase_flutter.dart';

void Connection() async {
  await Supabase.initialize(
    url: 'https://uoxncoqzdprtuhtozokv.supabase.co',
    anonKey:
        'eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJzdXBhYmFzZSIsI
nJlZiI6InVveG5jb3F6ZHBydHVodG96b2t2Iiwicm9sZSI6ImFub24iLCJpYXQiO
jE3NjM0NTQxNTQsImV4cCI6MjA3OTAzMDE1NH0.8HuA5Kx3uiQSy8iowgM
VFeTRzkNFQGH8_4AtN6dGdRU',
  );
}

//insert data
final db = Supabase.instance.client;
String table = 'perpustakaan';

Future<void> insertData(Map<String, dynamic> data) async {
  try {
    await db.from(table).insert(data);
    print('Data inserted successfully');
  } catch (e) {
    print('Error inserting data: $e');
  }
}

//read data
Future<List> getData() async {
  final response = await db.from(table).select();
  print(response.toList().toString());
  return response;
}

//update data
```

```
Future<void> updateData(Map<String, dynamic> row) async {
  try {
    await db.from(table).update(row).eq('id', row['id']);
    print('Data updated successfully');
  } catch (e) {
    print('Error updating data: $e');
  }
}

//delete data
Future<void> deleteData(String id) async {
  try {
    await db.from(table).delete().eq('id', id);
    print('Data deleted successfully');
  } catch (e) {
    print('Error deleting data: $e');
  }
}

//main.dart
import 'package:flutter/material.dart';
import
'package:google_maps_flutter_android/google_maps_flutter_android.dart';
import
'package:google_maps_flutter_platform_interface/google_maps_flutter_platform
_interface.dart';
import 'render_maps.dart';

void main() {
  final GoogleMapsFlutterPlatform mapsImplementation =
      GoogleMapsFlutterPlatform.instance;
  if (mapsImplementation is GoogleMapsFlutterAndroid) {
    mapsImplementation.useAndroidViewSurface = true;
  }
  runApp(const MyApp());
}

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Unguided Maps & Place Picker',
      theme: ThemeData(primarySwatch: Colors.blue),
      home: const RenderMaps(),
    );
  }
}
```

Screenshot Output:





Deskripsi Program

Program ini merupakan aplikasi Flutter yang mengintegrasikan **Google Maps** dan **Place Picker** untuk memilih lokasi secara interaktif. Pada halaman utama (RenderMaps), aplikasi menampilkan peta Google Maps dengan posisi awal pada koordinat tertentu. Pengguna dapat menekan tombol *Floating Action Button* “Pilih Lokasi” untuk membuka halaman *Place Picker*, lalu memilih lokasi yang diinginkan. Setelah lokasi dipilih, aplikasi akan kembali ke halaman utama, menampilkan *marker* pada titik lokasi tersebut, serta menampilkan informasi lokasi berupa nama tempat, alamat (jika tersedia), dan koordinat lintang serta bujur dalam sebuah *info card* di bagian bawah layar.

Selain fitur peta, program ini juga terhubung dengan **Supabase** sebagai *backend* untuk pengelolaan data. File connection.dart berfungsi untuk melakukan inisialisasi koneksi Supabase, sedangkan fungsi insertData, getData, updateData, dan deleteData digunakan untuk melakukan operasi **CRUD (Create, Read, Update, Delete)** pada tabel perpustakaan. Dengan kombinasi Google Maps untuk pemilihan lokasi dan Supabase untuk penyimpanan data, aplikasi ini dapat digunakan sebagai dasar sistem pencatatan lokasi (misalnya lokasi perpustakaan atau fasilitas umum) yang bersifat interaktif dan terintegrasi dengan basis data berbasis cloud.