

**TUGAS PENDAHULUAN
PEMROGRAMAN PERANGKAT BERGERAK**

**MODUL XII
MAPS & PLACES**



Disusun Oleh :

Zivana Afra Yulianto / 2211104039

S1SE-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

TUGAS PENDAHULUAN

SOAL

1. Menambahkan Google Maps Package

- a. Apa nama package yang digunakan untuk mengintegrasikan Google Maps di Flutter dan sebutkan langkah-langkah yang diperlukan untuk menambahkan package Google Maps ke dalam proyek Flutter.

- Nama package: Package yang digunakan untuk mengintegrasikan Google Maps di Flutter adalah `google_maps_flutter`.

- Langkah-langkah menambahkan package:

1. Tambahkan dependensi `google_maps_flutter` ke dalam file `pubspec.yaml`:

```
dependencies:  
  google_maps_flutter: ^latest_version
```

2. Jalankan perintah:

```
bash  
flutter pub get
```

3. Pastikan Anda telah menambahkan konfigurasi API Key untuk platform Android dan iOS:

- o Android: Tambahkan API Key di

```
android/app/src/main/AndroidManifest.xml:
```

```
<meta-data  
  android:name="com.google.android.geo.API_KEY"  
  android:value="YOUR_API_KEY"/>
```

- o iOS: Tambahkan API Key di file

```
ios/Runner/AppDelegate.swift atau di file AppDelegate.m.
```

- b. Mengapa kita perlu menambahkan API Key, dan di mana API Key tersebut diatur dalam aplikasi Flutter?

- Mengapa perlu API Key? API Key diperlukan untuk memverifikasi aplikasi Anda ke layanan Google Maps, sehingga Google dapat memberikan akses ke fitur-fitur peta berdasarkan kuota penggunaan.

- Tempat pengaturan API Key: API Key diatur di konsol Google Cloud Platform pada bagian Credentials.

2. Menampilkan Google Maps

- a. Tuliskan kode untuk menampilkan Google Map di Flutter menggunakan widget **GoogleMap**.

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

void main() => runApp(MyApp());

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        appBar: AppBar(title: Text('Google Maps Example')),
        body: GoogleMap(
          initialCameraPosition: CameraPosition(
            target: LatLng(-6.2088, 106.8456), // Posisi awal di Jakarta
            zoom: 14.0,
          ),
        ),
      ),
    );
  }
}
```

- b. Bagaimana cara menentukan posisi awal kamera (camera position) pada Google Maps di Flutter?

Menentukan posisi awal kamera (camera position): Posisi awal kamera ditentukan dengan properti `initialCameraPosition` yang diisi oleh objek `CameraPosition` (menggunakan koordinat latitude dan longitude).

- c. Sebutkan properti utama dari widget **GoogleMap** dan fungsinya.

Properti utama dari widget **GoogleMap**:

1. `initialCameraPosition`: Menentukan lokasi awal kamera.
2. `markers`: Untuk menambahkan penanda (marker) pada peta.
3. `onMapCreated`: Callback yang dipanggil saat peta berhasil diinisialisasi.

3. Menambahkan Marker

- a. Tuliskan kode untuk menambahkan marker di posisi tertentu (latitude: -6.2088, longitude: 106.8456) pada Google Maps.

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

class MyMapApp extends StatefulWidget {
  @override
  _MyMapAppState createState() => _MyMapAppState();
}

class _MyMapAppState extends State<MyMapApp> {
  late GoogleMapController mapController;

  final LatLng _markerPosition = LatLng(-6.2088, 106.8456);

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: Text('Marker Example')),
      body: GoogleMap(
        initialCameraPosition: CameraPosition(
          target: _markerPosition,
          zoom: 14.0,
        ),
        markers: {
          Marker(
            markerId: MarkerId('marker_1'),
            position: _markerPosition,
            infoWindow: InfoWindow(
              title: 'Jakarta',
              snippet: 'Ibukota Indonesia',
            ),
          ),
        },
      ),
    );
  }
}

void main() => runApp(MaterialApp(home: MyMapApp()));
```

- b. Bagaimana cara menampilkan info window saat marker diklik?

Cara menampilkan info window saat marker diklik: Info window dapat ditampilkan menggunakan properti infoWindow pada objek Marker.

4. Menggunakan Place Picker

- a. Apa itu Place Picker, dan bagaimana cara kerjanya di Flutter dan sebutkan nama package yang digunakan untuk implementasi Place Picker di Flutter.

- Apa itu Place Picker? Place Picker adalah fitur untuk memilih lokasi tertentu pada peta yang mengembalikan koordinat (latitude dan longitude). Di Flutter, ini dapat diimplementasikan menggunakan package seperti

`flutter_google_places`.

- Langkah implementasi Place Picker: Tambahkan package ke `pubspec.yaml`:

```
dependencies:
  flutter_google_places: ^latest_version
```

- b. Tuliskan kode untuk menampilkan Place Picker, lalu kembalikan lokasi yang dipilih oleh pengguna dalam bentuk latitude dan longitude.

```
import 'package:flutter/material.dart';
import 'package:google_maps_flutter/google_maps_flutter.dart';
import 'package:flutter_google_places/flutter_google_places.dart';
import 'package:google_maps_webservice/places.dart';

const String kGoogleApiKey = "YOUR_API_KEY";

void main() => runApp(MyApp());

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: PlacePickerExample(),
    );
  }
}

class PlacePickerExample extends StatefulWidget {
  @override
  _PlacePickerExampleState createState() => _PlacePickerExampleState();
}

class _PlacePickerExampleState extends State<PlacePickerExample> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: Text('Place Picker Example')),
      body: Center(
        child: ElevatedButton(
          onPressed: () async {
            Prediction? p = await PlacesAutocomplete.show(
              context: context,
              apiKey: kGoogleApiKey,
              mode: Mode.overlay, // Fullscreen mode
            );

            if (p != null) {
              final places = GoogleMapsPlaces(apiKey: kGoogleApiKey);
              PlacesDetailsResponse detail =
                await places.getDetailsByPlaceId(p.placeId!);
              final lat = detail.result.geometry!.location.lat;
              final lng = detail.result.geometry!.location.lng;

              print("Latitude: $lat, Longitude: $lng");
            }
          },
          child: Text("Pick a Place"),
        ),
      ),
    );
  }
}
```