

**TUGAS PENDAHULUAN
PEMROGRAMAN PERANGKAT BERGERAK**

**MODUL XII
MAPS & PLACES**



Disusun Oleh :

**Salman Alfa Rizzi
2211104056**

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

2. Menampilkan Google Maps

- a. Tuliskan kode untuk menampilkan Google Map di Flutter menggunakan widget GoogleMap.
- b. Bagaimana cara menentukan posisi awal kamera (camera position) pada Google Maps di Flutter?
- c. Sebutkan properti utama dari widget GoogleMap dan fungsinya.

3. Menambahkan Marker

- a. Tuliskan kode untuk menambahkan marker di posisi tertentu (latitude: -6.2088, longitude: 106.8456) pada Google Maps.
- b. Bagaimana cara menampilkan info window saat marker diklik?

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

JAWABAN:

1. A. Download package di flutter pub dev yang bernama `google_maps_flutter`.

Tambahkan dependencies:

```
google_maps_flutter: ^2.2.0
```

pada `pubspec.yaml` lalu ketik `flutter pub get` pada terminal

B. API Key diperlukan untuk mengidentifikasi aplikasi Anda agar dapat mengakses layanan Google Maps API. Tanpa API Key, akses ke layanan seperti Maps, Geocoding, atau Directions tidak diizinkan.

Di mana API Key diatur?

- API Key diatur dalam file konfigurasi aplikasi, seperti `AndroidManifest.xml` (Android) atau `AppDelegate.swift` (iOS).

Untuk Android: Tambahkan API Key

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

```
xml
```

```
<meta-data
```

```
android:name="com.google.android.geo.API_KEY"
```

```
android:value="YOUR_API_KEY"/>
```

IOS Tambahkan API Key di `ios/Runner/AppDelegate.swift`:

```
GMSServices.provideAPIKey("YOUR_API_KEY")
```

2. A. Kode google maps

```
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: GoogleMapScreen(),
```

```
    );
```

```
  }
```

```
}
```

```
class GoogleMapScreen extends StatelessWidget {
```

```

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(title: Text("Google Maps")),
    body: GoogleMap(
      initialCameraPosition: CameraPosition(
        target: LatLng(-6.2088, 106.8456), // Jakarta
        zoom: 14,
      ),
    ),
  );
}

```

B. Cara Menentukan Posisi Awal Kamera

Gunakan properti `initialCameraPosition` dalam widget `GoogleMap`, seperti: Dart.

```

initialCameraPosition: CameraPosition(
  target: LatLng(latitude, longitude), // Latitude dan Longitude
  zoom: 14, // Tingkat Zoom
)

```

C. Properti Utama dari GoogleMap

Properti	Fungsi
<code>initialCameraPosition</code>	Menentukan posisi awal kamera pada peta.
<code>markers</code>	Menampilkan marker pada peta.
<code>mapType</code>	Mengatur jenis peta (normal, satellite, hybrid, atau terrain).
<code>onMapCreated</code>	Callback yang dipanggil saat peta telah dibuat.
<code>zoomControlsEnabled</code>	Menentukan apakah kontrol zoom ditampilkan.

3. A. Kode untuk menambahkan marker

```

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

```

```

class GoogleMapMarkerScreen extends StatelessWidget {
  @override

```

```

Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(title: Text("Google Maps with Marker")),
    body: GoogleMap(
      initialCameraPosition: CameraPosition(
        target: LatLng(-6.2088, 106.8456), // Jakarta
        zoom: 14,
      ),
      markers: {
        Marker(
          markerId: MarkerId("jakarta"),
          position: LatLng(-6.2088, 106.8456),
          infoWindow: InfoWindow(title: "Jakarta", snippet: "Ibu Kota Indonesia"),
        ),
      },
    ),
  );
}

```

B. Menampilkan Info Window saat Marker Diklik

Gunakan properti infoWindow pada Marker. Contoh:

```

Marker(
  markerId: MarkerId("jakarta"),
  position: LatLng(-6.2088, 106.8456),
  infoWindow: InfoWindow(
    title: "Jakarta",
    snippet: "Ibu Kota Indonesia",
  ),
)

```

4. A. Place Picker adalah fitur untuk memilih lokasi tertentu di Google Maps.

Package yang umum digunakan untuk implementasi ini adalah flutter_google_places.

b. Kode untuk Menampilkan Place Picker

```

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

```

```

class PlacePickerScreen extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: Text("Place Picker")),
      body: Center(
        child: ElevatedButton(
          onPressed: () async {
            // Simulasikan pemilihan lokasi
            LatLng selectedLocation = await _pickPlace(context);
            ScaffoldMessenger.of(context).showSnackBar(
              SnackBar(
                content: Text("Lat: ${selectedLocation.latitude}, Lng:
                ${selectedLocation.longitude}"),
              ),
            );
          },
          child: Text("Pilih Lokasi"),
        ),
      ),
    );
  }

  Future<LatLng> _pickPlace(BuildContext context) async {
    // Logika untuk memilih tempat menggunakan Place Picker API
    // Simulasi saja untuk saat ini
    return LatLng(-6.2088, 106.8456); // Jakarta
  }
}

```