

**TUGAS PENDAHULUAN
PEMROGRAMAN PERANGKAT BERGERAK**

**MODUL XII
MAPS & PLACES**



Disusun Oleh :

Dimas Abhipraya Ramansyah / 2311104069

SE-07-02

Asisten Praktikum :

Yoga Eka Pratama

Zulfa Mustafa Akhyar Iswahyudi

Dosen Pengampu :

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

PROGRAM STUDI S1 SOFTWARE ENGINEERING

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2025

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. Menambahkan Google Maps Package

- a. Package yang digunakan untuk mengintegrasikan Google Maps di Flutter adalah `google_maps_flutter` dan berikut langkah-langkahnya :
- Menambahkan dependency `google_maps_flutter` ke file `pubspec.yaml`.
 - Menjalankan perintah `flutter pub get`.
 - Mengaktifkan Google Maps SDK for Android dan iOS di Google Cloud Console.
 - Membuat API Key dari Google Cloud Console.
 - Menambahkan API Key ke konfigurasi Android dan iOS.
- b. API Key diperlukan untuk:
- Mengautentikasi aplikasi ke layanan Google Maps
 - Mengontrol dan membatasi penggunaan API
 - Menjaga keamanan aplikasi
- Pengaturan API Key:
- **Android:** `android/app/src/main/AndroidManifest.xml`
 - **iOS:** `ios/Runner/AppDelegate.swift`

2. Menampilkan Google Maps

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

```
GoogleMap(  
  initialCameraPosition: CameraPosition(  
    target: LatLng(-6.2088, 106.8456),  
    zoom: 14,  
  ),  
)
```

- b. Posisi awal kamera ditentukan menggunakan properti `initialCameraPosition` yang berisi objek `CameraPosition`, dengan parameter:
- `target` untuk menentukan latitude dan longitude
 - `zoom` untuk menentukan tingkat perbesaran peta
- c. **Properti utama widget `GoogleMap`**
- `initialCameraPosition`: menentukan posisi awal kamera

- markers: menampilkan marker pada peta
- mapType: menentukan jenis peta (normal, satellite, hybrid)
- onMapCreated: dipanggil saat peta selesai dibuat
- myLocationEnabled: menampilkan lokasi pengguna

3. Menambahkan Marker

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

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

- infoWindow akan otomatis ditampilkan ketika marker diklik jika properti infoWindow telah didefinisikan pada marker.

4. Menggunakan Place Picker

- Place Picker adalah fitur yang memungkinkan pengguna memilih lokasi melalui peta atau pencarian tempat.

Package yang digunakan untuk mengimplementasikan Place Picker di Flutter adalah: `google_maps_place_picker_mb`

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

```
PlacePicker(
  apiKey: "API_KEY_ANDA",
  initialPosition: LatLng(-6.2088, 106.8456),
  useCurrentLocation: true,
  onPlacePicked: (result) {
    final lat = result.geometry!.location.lat;
    final lng = result.geometry!.location.lng;
```

```
print("Latitude: $lat, Longitude: $lng");  
},  
)
```