

**TUGAS PENDAHULUAN
PEMROGRAMAN PERANGKAT BERGERAK**

**MODUL XII
MAPS AND PLACES**



Disusun Oleh :
Rizky Hanifa Afania / 2211104017
SE-06-01

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

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.

Jawab:

Nama packagenya adalah **google_maps_flutter**. Berikut adalah langkah-langkah untuk menambahkan package **google_maps_flutter**:

- Tambahkan dependensi di file pubspec.yaml

```
dependencies:  
  google_maps_flutter: ^2.10.0
```

Lalu ketik “flutter pub get” di terminal untuk menginstall dependensi

- Tambahkan API Key di dalam file: **android/app/src/main/AndroidManifest.xml** tag <application>

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

- Tambahkan versi minimum SDK di dalam file **android/app/build.gradle**

```
defaultConfig {  
  minSdkVersion 20  
}
```

- Tambahkan widget Google Maps di dalam file dart, misalkan:

```
import  
'package:google_maps_flutter/google_maps_flutter.dart'  
;  
import 'package:flutter/material.dart';  
  
void main() => runApp(MyApp());  
  
class MyApp extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {
```

```

return MaterialApp(
  home: Scaffold(
    appBar: AppBar(title: Text("Google Maps in
Flutter")),
    body: GoogleMap(
      initialCameraPosition: CameraPosition(
        target: LatLng(-6.200000, 106.816666),
        zoom: 10,
      ),
    ),
  ),
);
}
}

```

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

Jawab:

- Alasannya karena API Key dibutuhkan untuk mengautentikasi permintaan aplikasi ke layanan Google Maps Platform. Jika tidak menggunakan API Key maka Google Maps Platform tidak dapat memverifikasi aplikasi dan layanan seperti peta, marker, atau fitur lainnya tidak akan berfungsi.
- Jika untuk media Android, API Key diatur dalam file **AndroidManifest.xml** di dalam tag <application>:

```

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

```

2. Menampilkan Google Maps

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

Jawab:

```

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 StatefulWidget {
  @override
  _GoogleMapScreenState createState() =>
  _GoogleMapScreenState();
}

class _GoogleMapScreenState extends State<GoogleMapScreen> {
  // Inisialisasi posisi awal kamera
  static const CameraPosition _initialCameraPosition =
  CameraPosition(
    target: LatLng(-6.200000, 106.816666), // Koordinat
    Jakarta
    zoom: 10.0,
  );

  late GoogleMapController _mapController;

  @override
  Widget build(BuildContext context) {
    return Scaffold(

```

```

appBar: AppBar(
  title: Text("Google Maps Example"),
),
body: GoogleMap(
  initialCameraPosition: _initialCameraPosition,
  mapType: MapType.normal, // Jenis peta (Normal,
Satellite, Hybrid, Terrain)
  onMapCreated: (controller) {
    _mapController = controller; // Mendapatkan
kontroler Google Maps
  },
  markers: {
    Marker(
      markerId: MarkerId('marker1'),
      position: LatLng(-6.200000, 106.816666), //
Lokasi marker
      infoWindow: InfoWindow(
        title: 'Jakarta',
        snippet: 'Ibu Kota Indonesia',
      ),
    ),
  },
),
);
}
}

```

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

Jawab:

Dengan menggunakan properti `initialCameraPosition` pada widget `GoogleMap`. Lalu posisi kamera ditentukan menggunakan `CameraPosition`, yang memiliki tiga parameter utama, yaitu:

- `target`: Objek `LatLng` yang menentukan koordinat lokasi awal kamera.
- `zoom`: Tingkat zoom peta (nilai default biasanya 10.0).

- tilt (opsional): Sudut kemiringan kamera dalam derajat.
- bearing (opsional): Orientasi kamera dalam derajat (arah utara = 0).

Contoh penggunaan:

```
CameraPosition initialPosition = CameraPosition(
  target: LatLng(-6.200000, 106.816666), // Jakarta
  zoom: 12.0, // Zoom level
  tilt: 30.0, // Kemiringan kamera
  bearing: 90.0, // Kamera menghadap timur
);
```

c. Sebutkan properti utama dari widget GoogleMap dan fungsinya.

Jawab:

- initialCameraPosition: untuk menentukan posisi awal kamera saat peta pertama kali dimuat.
- mapType: untuk menentukan jenis peta yang ditampilkan.
- onMapCreated: Callback yang dipanggil ketika peta selesai dimuat. Memberikan kontroler GoogleMapController.
- markers: untuk menambahkan titik marker ke peta.
- polylines: untuk menambahkan garis lintasan (path) pada peta.
- zoomControlsEnabled untuk menentukan apakah tombol kontrol zoom diaktifkan.

3. Menambahkan Marker

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

Jawab:

```
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 StatefulWidget {
  @override
  _GoogleMapScreenState createState() =>
    _GoogleMapScreenState();
}

class _GoogleMapScreenState extends State<GoogleMapScreen> {
  static const CameraPosition _initialCameraPosition =
    CameraPosition(
      target: LatLng(-6.2088, 106.8456), // Jakarta (Monas)
      zoom: 15.0,
    );

  late GoogleMapController _mapController;

  // Set untuk menyimpan marker
  final Set<Marker> _markers = {};

  @override
  void initState() {
    super.initState();
    // Tambahkan marker di posisi tertentu
    _addMarker(
      LatLng(-6.2088, 106.8456),
      "Monas",
      "Monumen Nasional, Jakarta",
    );
  }
}

```

```

    }

    void _addMarker(LatLng position, String title, String
snippet) {
        setState(() {
            _markers.add(
                Marker(
                    markerId: MarkerId(position.toString()),
                    position: position,
                    infoWindow: InfoWindow(title: title, snippet:
snippet),
                    onTap: () {
                        print("Marker tapped: $title");
                    },
                ),
            );
        });
    }

    @override
    Widget build(BuildContext context) {
        return Scaffold(
            appBar: AppBar(
                title: Text("Google Maps with Marker"),
            ),
            body: GoogleMap(
                initialCameraPosition: _initialCameraPosition,
                mapType: MapType.normal,
                markers: _markers,
                onMapCreated: (controller) {
                    _mapController = controller;
                },
            ),
        );
    }

```



```
}  
}
```

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

Jawab:

Info window akan otomatis ditampilkan saat terdapat properti infoWindow di marker. Jika ingin memastikan info window ditampilkan setiap kali marker diklik, bisa dengan menambahkan kode ini

```
infoWindow: InfoWindow(title: "Monas", snippet: "Monumen  
Nasional, Jakarta"),
```

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.

Jawab:

- Place Picker adalah fitur agar pengguna bisa memilih lokasi dari peta menggunakan antarmuka yang interaktif. Fitur ini biasanya ditemukan pada aplikasi seperti pengiriman makanan, pemesanan transportasi, atau aplikasi berbasis lokasi lainnya.
- Cara kerja dari Place Picker yaitu menggunakan layanan Google Maps API untuk menampilkan peta dan memfasilitasi pemilihan lokasi. Saat pengguna memilih lokasi, latitude dan longitude dari lokasi tersebut akan diambil dan dapat digunakan dalam aplikasi.
- Nama package: **google_maps_flutter** untuk peta dasar, sedangkan **flutter_google_places** atau **google_maps_place_picker** untuk pencarian dan pemilihan lokasi

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

Jawab:

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

```

import
'package:google_maps_place_picker_mb/google_maps_place_picker_mb.dart';

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

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

class PlacePickerScreen extends StatelessWidget {
  final String googleMapsApiKey =
"YOUR_GOOGLE_MAPS_API_KEY"; // Ganti dengan API Key Anda

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text("Place Picker"),
      ),
      body: Center(
        child: ElevatedButton(
          onPressed: () async {
            // Navigasi ke Place Picker
            final selectedPlace = await Navigator.push(
              context,
              MaterialPageRoute(
                builder: (context) => PlacePicker(
                  apiKey: googleMapsApiKey,

```

```

        initialPosition: LatLng(-6.2088,
106.8456), // Lokasi awal (Jakarta)
        useCurrentLocation: true,
        onPlacePicked: (result) {
            Navigator.pop(context, result);
        },
    ),
),
);

        if (selectedPlace != null) {
            print("Latitude:
${selectedPlace.geometry!.location.lat}");
            print("Longitude:
${selectedPlace.geometry!.location.lng}");
        }
    },
    child: Text("Pick a Place"),
),
),
);
}
}

```