

**LAPORAN PRAKTIKUM PEMROGRAMAN  
PERANGKAT BERGERAK  
MODUL 12**



Oleh: Lintang Suminar Tyas Wening  
2211104009  
SE-06-01

**PRODI S1 REKAYASA PERANGKAT LUNAK  
FAKULTAS INFORMATIKA  
TELKOM UNIVERSITY PURWOKERTO  
2024**

## GUIDED

### A. GOOGLE MAPS API

Google Maps API adalah layanan dari Google yang dirancang untuk membantu developer membuat aplikasi dengan fitur peta atau maps. Melalui Google Maps API, kita bisa menambahkan marker, menggunakan fitur rute, mencari lokasi, dan berbagai fitur lainnya. Untuk mengintegrasikan Google Maps API ke dalam Flutter, kita bisa memanfaatkan paket Google Maps.

Berikut adalah langkah-langkah yang dapat diikuti untuk menambahkan Google Maps API ke dalam aplikasi Flutter:

1. Dapatkan API key melalui link berikut <https://cloud.google.com/maps-platform/>
2. Selanjutnya, enable Google Map SDK di tiap platform yang akan menggunakan Google Maps.
  - a. Aaaa Pergi ke <https://console.cloud.google.com/> (Google Delevopers Console)
  - b. Pilih project yang ingin menggunakan Google Maps
  - c. Pilih pada navigation menu, lalu pilih "Google Maps"
  - d. Pilih "APIs" di bawah menu Google Maps
  - e. Untuk mengaktifkan Google Maps di Android, pilih "Maps SDK for Android" pada section "Additional APIs", lalu pilih "ENABLE"
  - f. Untuk mengaktifkan Google Maps di iOS, pilih "Maps SDK for iOS" pada section "Additional APIs", lalu pilih "ENABLE"
  - g. Pastikan bahwa APIs telah aktif pada section "Enabled APIs"
  - h. Untuk lebih detail bisa cek di <https://developers.google.com/maps/gmp-get-started>
3. Andorid
  - a. Set minSdkVersion di android/app/build.gradle:

```
defaultConfig {  
    minSdkVersion 20  
    targetSdk = flutter.targetSdkVersion  
    versionCode = flutter.versionCode  
    versionName = flutter.versionName  
}
```

Ini dimaksudkan agar aplikasinya support atau bersedia pada Android SDK 20 atau lebih tinggi.

- b. Tambahkan API key pada manifest aplikasi

**android/app/src/main/AndroidManifest.xml**

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android">  
    <application  
        android:label="unguided12"  
        android:name="{applicationName}"  
        android:icon="@mipmap/ic_launcher">  
        <meta-data  
            android:name="com.google.android.geo.API_KEY"  
            android:value="AIzaSyDKYGFz6w9Z07-_XrhkRUNIfeqUmvqRz0s" />  
        <activity
```

#### 4. Hybrid Composition

Untuk menggunakan Hybrid Composition yang digunakan untuk merender GoogleMap pada widget Android, terapkan `AndroidGoogleMapsFlutter.userAndroidViewSurface` ke `true`.

### B. Menambah Packages Google Maps

Setelah mengikuti langkah diatas, sekarang adalah langkah-langkah menambahkan Google Maps ke layar aplikasi Flutter:

1. Pergi ke <https://www.pub.dev> , lalu cari packages Google Maps. Nama packagesnya adalah `google_maps_flutter`.
2. Cari versi yang paling terbaru lalu tambahkan pada file `pubspec.yaml`

```
dependencies:
  flutter:
    sdk: flutter

# The following adds the Cupertino Icons font to your application.
# Use with the CupertinoIcons class for iOS style icons.
cupertino_icons: ^1.0.8
google_maps_flutter: ^2.10.0
```

3. Selanjutnya, import packages ke dalam file Dart

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

4. Lalu, tambahkan widget GoogleMap ke file Dart

```
body: GoogleMap(
  initialCameraPosition: _kInitialPosition,
  //onMapCreated: onMapCreated,
  myLocationEnabled: true,
```

5. Buat fungsi `_kMapCenter` dan `_kInitialPosition` dengan kode sebagai berikut:

```
class _MyHomePageState extends State<MyHomePage> {
  static final LatLng _kMapCenter =
    const LatLng(19.018255973653343, 72.84793849278007);
  static final CameraPosition _kInitialPosition =
    CameraPosition(target: _kMapCenter, zoom: 11.0, tilt: 0, bearing: 0);
```

6. Berikut adalah tampilan kode yang lengkap

```
1 import 'package:flutter/material.dart';
2 import 'package:google_maps_flutter/google_maps_flutter.dart';
3
4 class MyHomePage extends StatefulWidget {
5   const MyHomePage({super.key});
6
7   @override
8   State<MyHomePage> createState() => _MyHomePageState();
9 }
10
11 class _MyHomePageState extends State<MyHomePage> {
12   static final LatLng _kMapCenter =
13     const LatLng(19.018255973653343, 72.84793849278007);
14   static final CameraPosition _kInitialPosition =
15     CameraPosition(target: _kMapCenter, zoom: 11.0, tilt: 0, bearing: 0);
```

```

16
17 @override
18 Widget build(BuildContext context) {
19   return Scaffold(
20     appBar: AppBar(
21       title: const Text('Google Maps Demo'),
22       centerTitle: true,
23       backgroundColor: const Color.fromARGB(255, 113, 255, 246),
24     ),
25     body: GoogleMap(
26       initialCameraPosition: _kInitialPosition,
27       //onMapCreated: onMapCreated,
28       myLocationEnabled: true,
29     ),
30   );
31 }
32 }
33

```

7. Berikut tampilannya



### C. Menambahkan Akses di Manifest

Secara default, map akan menampilkan lokasi yang sudah kita definisikan pada `initialCameraPosition` yang ada pada parameter widget. Jika pengguna ingin menampilkan lokasi mereka, ubah pengaturan `myLocationEnabled` menjadi `true`. Berikut barisan kode untuk menampilkan lokasi kita saat ini:

```

body: GoogleMap(
  initialCameraPosition: _kInitialPosition,
  //onMapCreated: onMapCreated,
  myLocationEnabled: true,

```

#### D. Menambahkan Marker pada Google Maps

Marker merupakan cara untuk menunjukkan lokasi tertentu. Untuk membuat marker pada map, berikut adalah barisan kodenya:

```
// Membuat marker
Set<Marker> _createMarker() {
  return {
    Marker(
      markerId: MarkerId("Mumbai"),
      position: _kMapCenter,
      infoWindow: InfoWindow(title: 'Mumbai'),
      rotation: 0,
    ), // Marker
    Marker(
      markerId: MarkerId("Taman Andang"),
      position: LatLng(-7.4397127924524895,
        109.24341678454714), // Koordinat contoh lainnya // LatLng
      infoWindow: InfoWindow(title: 'Taman Andang')), // Marker
  };
}
```

#### E. Place Picker

Place picker merupakan plugin untuk memberi informasi terkait lokasi yang sedang ditunjuk oleh map. Untuk menggunakan place picker, terlebih dahulu kita tambahkan API Google Maps dan ubah beberapa pengaturan dengan mengikuti langkah-langkah sebelum sub bab ini. Jika sudah mengubah pengaturan yang diminta, terlebih dahulu kita tambahkan package ke dalam pubspec.yaml seperti ini:

```
1 import 'package:flutter/material.dart';
2 import 'package:google_maps_flutter/google_maps_flutter.dart';
3 import 'package:place_picker_google/place_picker_google.dart';
```

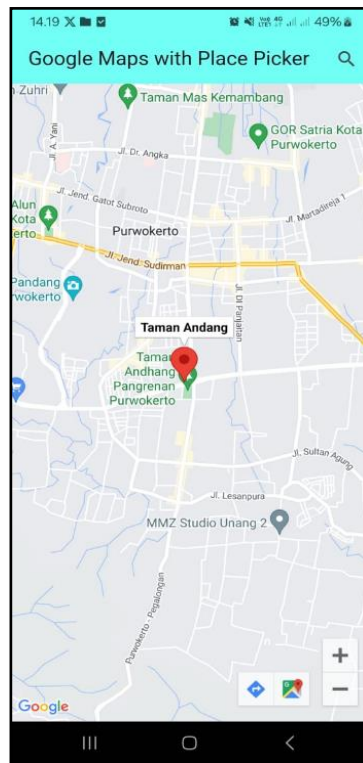
Setelah itu, kita buat method seperti yang akan di bahas pada sub bab ini, lalu gunakan onTap di Button atau InkWell untuk memanggil method tersebut. Berikut barisan kodenya:

```
// Fungsi untuk menampilkan Place Picker
void _showPlacePicker() async {
  // Menampilkan Place Picker dan menangkap hasilnya
  LocationResult? result = await Navigator.of(context).push(MaterialPageRoute(
    builder: (context) => PlacePicker(
      apiKey: apiKey, // API key untuk Google Maps
      onPlacePicked: (LocationResult result) {
        // Menampilkan alamat tempat yang dipilih
        debugPrint("Place picked: ${result.formattedAddress}");
      },
      initialLocation: _kMapCenter, // Menampilkan lokasi awal
      searchInputConfig: const SearchInputConfig(
        padding: EdgeInsets.symmetric(horizontal: 16.0, vertical: 8.0),
        autofocus: false,
        textDirection: TextDirection.ltr,
      ), // SearchInputConfig
      searchInputDecorationConfig: const SearchInputDecorationConfig(
        hintText: "Search for a building, street or ...",
      ), // SearchInputDecorationConfig
    ), // PlacePicker
  )); // MaterialPageRoute
}
```

Berikut tampilan marker map Gambar 1 (Mumbai) dan Gambar 2 (Taman Andang)



*Gambar 1*



*Gambar 2*

## UNGUIDED

### Tugas Mandiri

1. 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, screenshot output, dan deskripsi program. Kreativitas menjadi nilai tambah.*

**Jawab :**

**Source Code :**

- Main dart

```
1  import 'package:flutter/material.dart';
2  import 'package:unguided12/mainpage.dart';
3
4  void main() {
5      runApp(const MyApp());
6  }
7
8  class MyApp extends StatelessWidget {
9      const MyApp({super.key});
10
11     @override
12     Widget build(BuildContext context) {
13         return MaterialApp(
14             debugShowCheckedModeBanner: false,
15             title: 'Modul 12 - Maps & Places',
16             theme: ThemeData(
17                 useMaterial3: true,
18             ),
19             home: const MyHomePage(),
20         );
21     }
22 }
```

- Mainpage

```
1 import 'package:flutter/material.dart';
2 import 'package:google_maps_flutter/google_maps_flutter.dart';
3 import 'package:place_picker_google/place_picker_google.dart';
4
5 class MyHomePage extends StatefulWidget {
6   const MyHomePage({super.key});
7
8   @override
9   State<MyHomePage> createState() => _MyHomePageState();
10 }
11
12 class _MyHomePageState extends State<MyHomePage> {
13   static final LatLng _kMapCenter = LatLng(19.018255973653343,
14     72.84793849278007); // Koordinat yang digunakan (contohnya lokasi di Mumbai)
15   static final CameraPosition _kInitialPosition =
16     CameraPosition(target: _kMapCenter, zoom: 11.0, tilt: 0, bearing: 0);
17
18   // API Key untuk Google Maps
19   final String apiKey = 'YOUR_GOOGLE_MAPS_API_KEY';
20
21   // Membuat marker
22   Set<Marker> _createMarker() {
23     return {
24       Marker(
25         markerId: MarkerId("Mumbai"),
26         position: _kMapCenter,
27         infoWindow: InfoWindow(title: 'Mumbai'),
28         rotation: 0,
29       ),
30       Marker(
31         markerId: MarkerId("Taman Andang"),
32         position: LatLng(-7.4397127924524895,
33           109.24341678454714), // Koordinat contoh lainnya
34         infoWindow: InfoWindow(title: 'Taman Andang')),
35     };
36   }
37
38   // Fungsi untuk menampilkan Place Picker
39   void _showPlacePicker() async {
40     // Menampilkan Place Picker dan menangkap hasilnya
41     LocationResult? result = await Navigator.of(context).push(MaterialPageRoute(
42       builder: (context) => PlacePicker(
43         apiKey: apiKey, // API key untuk Google Maps
44         onPlacePicked: (LocationResult result) {
45           // Menampilkan alamat tempat yang dipilih
46           debugPrint("Place picked: ${result.formattedAddress}");
47         },
48         initialLocation: _kMapCenter, // Menampilkan lokasi awal
49         searchInputConfig: const SearchInputConfig(
50           padding: EdgeInsets.symmetric(horizontal: 16.0, vertical: 8.0),
51           autofocus: false,
52           textDirection: TextDirection.ltr,
53         ),
54         searchInputDecorationConfig: const SearchInputDecorationConfig(
55           hintText: "Search for a building, street or ...",
56         ),
57       ),
58     ));
59
60     // Pengecekan jika result tidak ada
61     if (result != null) {
62       debugPrint("Place picked: ${result.formattedAddress}");
63     } else {
64       debugPrint("No place selected.");
65     }
66   }
67
68   @override
69   Widget build(BuildContext context) {
70     return Scaffold(
71       appBar: AppBar(
72         title: const Text('Google Maps with Place Picker'),
73         centerTitle: true,
74         backgroundColor: const Color.fromARGB(255, 113, 255, 246),
75         actions: [
76           IconButton(
77             icon: const Icon(Icons.search),
78             onPressed:
79               _showPlacePicker, // Menampilkan Place Picker saat ikon pencarian ditekan
80           ),
81         ],
82       ),
83       body: GoogleMap(
84         initialCameraPosition: _kInitialPosition,
85         myLocationEnabled: true,
86         markers: _createMarker(), // Menambahkan marker pada peta
87       ),
88     );
89   }
90 }
91
```

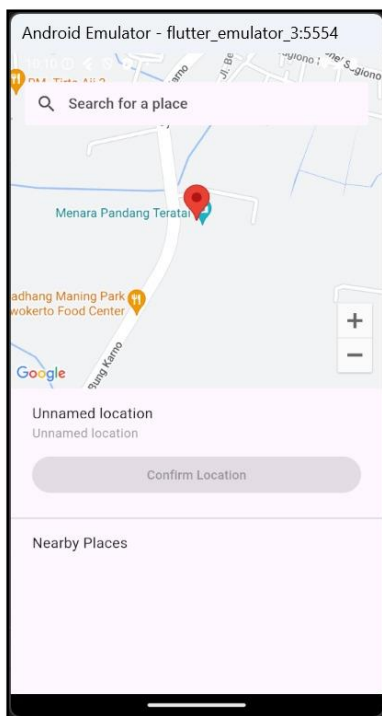


## Screenshot Output :

- Tampilan lokasi yang telah di marker



- Tampilan info lokasi



**Deskripsi Program :**

Program ini merupakan aplikasi Flutter yang memanfaatkan Google Maps API dan Place Picker untuk melihat peta interaktif dengan fitur tambahan seperti marker dan pemilihan lokasi. Peta yang ditampilkan memiliki koordinat awal di Mumbai dengan pengaturan kamera awal yang mencakup tingkat zoom, kemiringan, dan orientasi. Pada peta ini, terdapat dua marker, yaitu di Mumbai dan Taman Andang, yang masing-masing dilengkapi dengan *InfoWindow* yang memberikan informasi lokasi yang ditampilkan saat marker ditekan.