



## UNGUIDED 12



Disusun Oleh :  
**Nama : Ganes Gemi Putra**  
**Kelas : SE-07-02**  
**NIM : (2311104075)**

**Dosen : YUDHA ISLAMI SULISTYA S.Kom M.Cs**

**PROGRAM STUDI S1 SOFTWARE ENGINEERING**  
**FAKULTAS INFORMATIKA**  
**TELKOM UNIVERSITY**  
**PURWOKERTO**  
**2025**

## 1. Tujuan

Mengimplementasikan Google Maps dan fitur Place Picker pada aplikasi Flutter untuk menampilkan informasi lokasi yang ditunjuk pada peta.

## 2. Deskripsi Program

Aplikasi Flutter ini menampilkan peta Google Maps dan menyediakan fitur Place Picker. Pengguna dapat memilih lokasi pada peta, kemudian aplikasi menampilkan informasi lokasi berupa nama/label lokasi, alamat, dan koordinat latitude serta longitude. Lokasi yang dipilih juga ditandai dengan marker pada peta.

## 3. Input dan Output Program

### 3.1 Input

- Pengguna membuka aplikasi.
- Pengguna menekan tombol **Pick Place**.
- Pengguna menunjuk lokasi pada peta atau mencari lokasi melalui kolom pencarian pada Place Picker.

### 3.2 Output

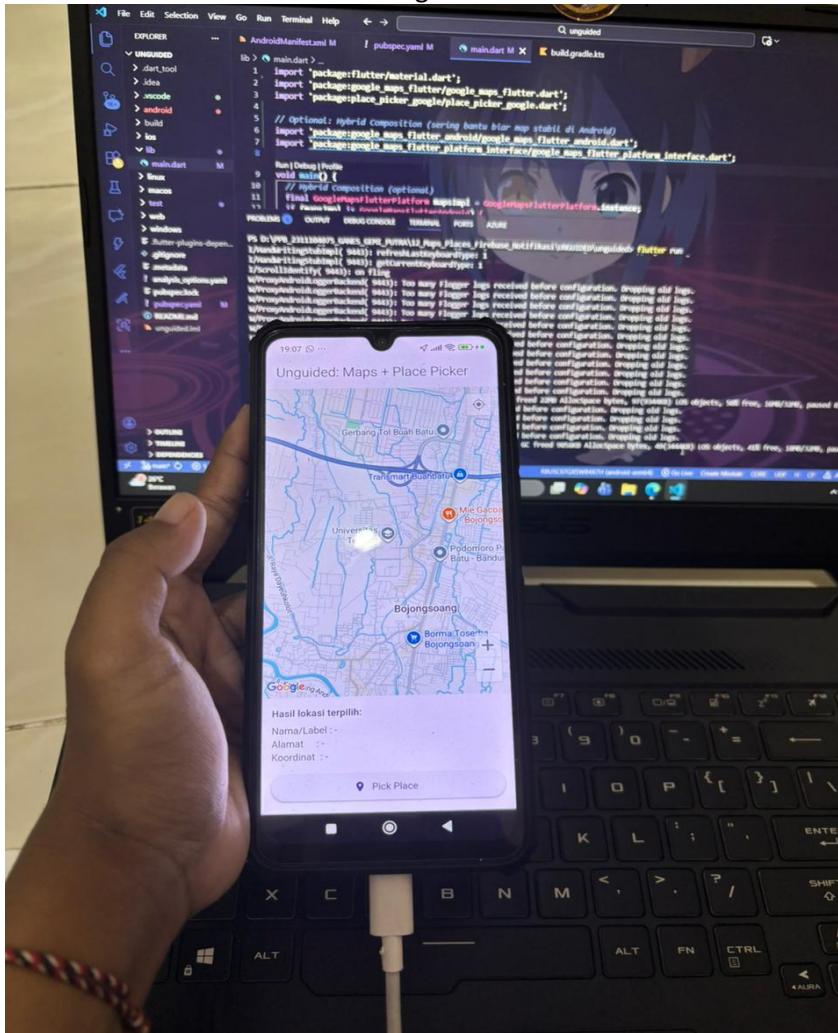
- Aplikasi menampilkan peta Google Maps.
- Aplikasi menampilkan halaman Place Picker.
- Aplikasi menampilkan lokasi yang ditunjuk pada peta.
- Aplikasi menampilkan informasi lokasi berupa:
  - Nama/Label lokasi
  - Alamat lokasi
  - Koordinat latitude dan longitude

## 5. Hasil Pengujian (Screenshot Output)

### Screenshot 1 – Tampilan Awal Aplikasi

Isi screenshot:

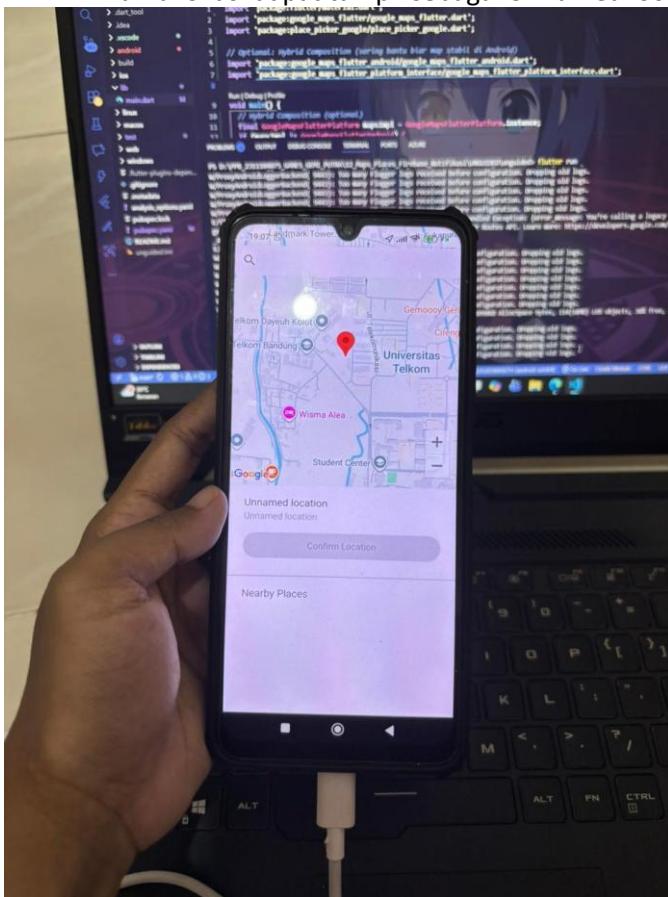
- Tampilan Google Maps
- Judul aplikasi “Unguided: Maps + Place Picker”
- Tombol **Pick Place** pada halaman utama
- Nama/Label lokasi
- Alamat lokasi
- Koordinat latitude dan longitude



## Screenshot 2 – Halaman Place Picker

Isi screenshot:

- Tampilan Place Picker
- Pin lokasi pada peta
- Nama lokasi dapat tampil sebagai *Unnamed location*



## 6. Pembahasan

Pada implementasi Place Picker, nama lokasi dapat tampil sebagai “*Unnamed location*” apabila pengguna memilih titik koordinat yang tidak terdaftar sebagai tempat resmi (Point of Interest) pada Google Maps. Hal ini merupakan perilaku normal dari Google Maps API. Informasi koordinat tetap dapat diperoleh dan digunakan oleh aplikasi.

## 7. Kesimpulan

Aplikasi Flutter telah berhasil mengintegrasikan Google Maps dan Place Picker. Pengguna dapat memilih lokasi pada peta dan aplikasi dapat menampilkan informasi lokasi yang dipilih sesuai dengan tujuan Tugas Mandiri (Unguided).

## 8. Source Code

File source code yang digunakan pada aplikasi ini:

➤ **main.dart**

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

// Optional: Hybrid Composition (sering bantu biar map stabil di Android)
import 'package:google_maps_flutter_android/google_maps_flutter_android.dart';
import
'package:google_maps_flutter_platform_interface/google_maps_flutter_platform_interface.dart';

void main() {
// Hybrid Composition (optional)
final GoogleMapsFlutterPlatform mapsImpl = GoogleMapsFlutterPlatform.instance;
if (mapsImpl is GoogleMapsFlutterAndroid) {
    mapsImpl.useAndroidViewSurface = true;
}

runApp(const MyApp());
}

class MyApp extends StatelessWidget {
const MyApp({super.key});

@override
Widget build(BuildContext context) {
    return const MaterialApp(
        debugShowCheckedModeBanner: false,
        home: MapsHomePage(),
    );
}
}

class MapsHomePage extends StatefulWidget {
const MapsHomePage({super.key});

@override
State<MapsHomePage> createState() => _MapsHomePageState();
}

class _MapsHomePageState extends State<MapsHomePage> {
// ✅ GANTI INI
static const String apiKey = "AlzaSyAlo-EOPFCls5IN6ixqlnNBGe1M4BjIzVk";

// Default posisi awal (boleh ganti)
static const LatLng kMapCenter = LatLng(-6.9733165, 107.6281415);
static const CameraPosition kInitialPosition =
    CameraPosition(target: kMapCenter, zoom: 14);

GoogleMapController? _mapController;
final Set<Marker> _markers = {};

String pickedName = "-";
String pickedAddress = "-";
LatLng? pickedLatLng;

@override
Widget build(BuildContext context) {
    return Scaffold(
        appBar: AppBar(
```

```

title: const Text("Unguided: Maps + Place Picker"),
),
body: Column(
children: [
Expanded(
child: GoogleMap(
initialCameraPosition: kInitialPosition,
onMapCreated: (c) => _mapController = c,
myLocationEnabled: true,
myLocationButtonEnabled: true,
markers: _markers,
),
),
Container(
width: double.infinity,
padding: const EdgeInsets.all(14),
child: Column(
crossAxisAlignment: CrossAxisAlignment.start,
children: [
const Text(
"Hasil lokasi terpilih:",
style: TextStyle(fontWeight: FontWeight.bold),
),
const SizedBox(height: 6),
Text("Nama/Label : $pickedName"),
Text("Alamat : $pickedAddress"),
Text(
"Koordinat : ${pickedLatLng == null ? '-' : '${pickedLatLng!.latitude}, ${pickedLatLng!.longitude}'}",
),
const SizedBox(height: 12),
SizedBox(
width: double.infinity,
child: ElevatedButton.icon(
 onPressed: () => openPlacePicker(context),
 icon: const Icon(Icons.place),
 label: const Text("Pick Place"),
),
),
],
),
],
),
),
);
}

Future<void> openPlacePicker(BuildContext context) async {
final LocationResult? result =
await Navigator.of(context).push<LocationResult>(
MaterialPageRoute(
builder: (_) => PlacePicker(
apiKey: apiKey,
initialLocation: pickedLatLng ?? kMapCenter,
),
),
);
if (result == null) return;

final latLng = result.latLng;
if (latLng == null) return;
}

```

```

setState(() {
  pickedLatLng = LatLng(latLng.latitude, latLng.longitude);
  pickedName = result.name ?? "Unnamed location";
  pickedAddress = result.formattedAddress ?? "Alamat tidak tersedia";

  _markers
  ..clear()
  ..add(
    Marker(
      markerId: const MarkerId("picked_place"),
      position: pickedLatLng!,
      infoWindow: InfoWindow(
        title: pickedName,
        snippet: pickedAddress,
      ),
    ),
  );
});

await _mapController?.animateCamera(
  CameraUpdate.newLatLngZoom(pickedLatLng!, 16),
);
}
}
}

```

## ➤ AndroidManifest.xml

```

<manifest xmlns:android="http://schemas.android.com/apk/res/android">
  <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
  <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>

  <application
    android:label="unguided"
    android:name="${applicationName}"
    android:icon="@mipmap/ic_launcher">
    <!-- ✅ GANTI VALUE API KEY -->
    <meta-data
      android:name="com.google.android.geo.API_KEY"
      android:value="AlzaSyAlo-EOPFCls5lN6ixqlnNBGe1M4BjlzVk"/>

    <activity
      android:name=".MainActivity"
      android:exported="true"
      android:launchMode="singleTop"
      android:taskAffinity=""
      android:theme="@style/LaunchTheme"
      android:configChanges="orientation|keyboardHidden|keyboard|screenSize|smallestScreenSize|locale|layoutDirection|fontScale|screenLayout|density|uiMode"
      android:hardwareAccelerated="true"
      android:windowSoftInputMode="adjustResize">
      <!-- Specifies an Android theme to apply to this Activity as soon as
          the Android process has started. This theme is visible to the user
          while the Flutter UI initializes. After that, this theme continues
          to determine the Window background behind the Flutter UI. -->
      <meta-data
        android:name="io.flutter.embedding.android.NormalTheme"
        android:resource="@style/NormalTheme"
      />
      <intent-filter>
        <action android:name="android.intent.action.MAIN"/>
        <category android:name="android.intent.category.LAUNCHER"/>
      
```

```

</intent-filter>
</activity>
<!-- Don't delete the meta-data below.
     This is used by the Flutter tool to generate GeneratedPluginRegistrant.java -->
<meta-data
    android:name="flutterEmbedding"
    android:value="2" />
</application>
<!-- Required to query activities that can process text, see:
     https://developer.android.com/training/package-visibility and
     https://developer.android.com/reference/android/content/Intent#ACTION_PROCESS_TEXT.

     In particular, this is used by the Flutter engine in io.flutter.plugin.text.ProcessTextPlugin. -->
<queries>
    <intent>
        <action android:name="android.intent.action.PROCESS_TEXT"/>
        <data android:mimeType="text/plain"/>
    </intent>
</queries>
</manifest>

```

## ➤ pubspec.yaml

```

name: unguided
description: "A new Flutter project."
# The following line prevents the package from being accidentally published to
# pub.dev using 'flutter pub publish'. This is preferred for private packages.
publish_to: 'none' # Remove this line if you wish to publish to pub.dev

# The following defines the version and build number for your application.
# A version number is three numbers separated by dots, like 1.2.43
# followed by an optional build number separated by a +.
# Both the version and the builder number may be overridden in flutter
# build by specifying --build-name and --build-number, respectively.
# In Android, build-name is used as versionName while build-number used as versionCode.
# Read more about Android versioning at https://developer.android.com/studio/publish/versioning
# In iOS, build-name is used as CFBundleShortVersionString while build-number is used as CFBundleVersion.
# Read more about iOS versioning at
#
# https://developer.apple.com/library/archive/documentation/General/Reference/InfoPlistKeyReference/Articles/CoreFoundationKeys.html
# In Windows, build-name is used as the major, minor, and patch parts
# of the product and file versions while build-number is used as the build suffix.
version: 1.0.0+1

environment:
  sdk: ^3.9.2

# Dependencies specify other packages that your package needs in order to work.
# To automatically upgrade your package dependencies to the latest versions
# consider running 'flutter pub upgrade --major-versions'. Alternatively,
# dependencies can be manually updated by changing the version numbers below to
# the latest version available on pub.dev. To see which dependencies have newer
# versions available, run 'flutter pub outdated'.
dependencies:
  flutter:
    sdk: flutter

  google_maps_flutter: ^2.6.1
  place_picker_google: ^0.0.20

# The following adds the Cupertino Icons font to your application.

```

```
# Use with the CupertinoIcons class for iOS style icons.  
cupertino_icons: ^1.0.8  
  
dev_dependencies:  
  flutter_test:  
    sdk: flutter  
  
  # The "flutter_lints" package below contains a set of recommended lints to  
  # encourage good coding practices. The lint set provided by the package is  
  # activated in the `analysis_options.yaml` file located at the root of your  
  # package. See that file for information about deactivating specific lint  
  # rules and activating additional ones.  
  flutter_lints: ^5.0.0  
  
  # For information on the generic Dart part of this file, see the  
  # following page: https://dart.dev/tools/pub/pubspec  
  
  # The following section is specific to Flutter packages.  
flutter:  
  
  # The following line ensures that the Material Icons font is  
  # included with your application, so that you can use the icons in  
  # the material Icons class.  
  uses-material-design: true  
  
  # To add assets to your application, add an assets section, like this:  
  # assets:  
  #   - images/a_dot_burr.jpeg  
  #   - images/a_dot_ham.jpeg  
  
  # An image asset can refer to one or more resolution-specific "variants", see  
  # https://flutter.dev/to/resolution-aware-images  
  
  # For details regarding adding assets from package dependencies, see  
  # https://flutter.dev/to/asset-from-package  
  
  # To add custom fonts to your application, add a fonts section here,  
  # in this "flutter" section. Each entry in this list should have a  
  # "family" key with the font family name, and a "fonts" key with a  
  # list giving the asset and other descriptors for the font. For  
  # example:  
  # fonts:  
  #   - family: Schyler  
  #     fonts:  
  #       - asset: fonts/Schyler-Regular.ttf  
  #       - asset: fonts/Schyler-Italic.ttf  
  #         style: italic  
  #   - family: Trajan Pro  
  #     fonts:  
  #       - asset: fonts/TrajanPro.ttf  
  #       - asset: fonts/TrajanPro_Bold.ttf  
  #         weight: 700  
  #  
  # For details regarding fonts from package dependencies,  
  # see https://flutter.dev/to/font-from-package
```