

---

# Flutter Google Maps Integration: Displaying Maps & Locations

**Subtitle:** Learn how to integrate Google Maps in Flutter apps and work with markers and user location.

**Website Name:** haas.dev

**Website Link:** <https://dev-roast-app.vercel.app>

---

## Introduction

Maps are essential for apps that need **location services or navigation features**. Flutter supports Google Maps integration to **display maps, add markers, and track user location**.

---

## Step 1: Add Google Maps Dependency

```
dependencies:  
  google_maps_flutter: ^2.3.0  
  location: ^6.1.0
```

Run:

```
flutter pub get
```

Import in Dart:

```
import 'package:google_maps_flutter/google_maps_flutter.dart';  
import 'package:location/location.dart';
```

---

## Step 2: Add API Key

- Get a Google Maps API key from Google Cloud
- Add it to `android/app/src/main/AndroidManifest.xml`:

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

**Exercise:** Configure your API key for Android and iOS.

---

### Step 3: Display a Basic Map

```
GoogleMap(  
  initialCameraPosition: CameraPosition(  
    target: LatLng(37.42796133580664, -122.085749655962),  
    zoom: 14.0,  
  ),  
)
```

**Exercise:** Display a map centered at your city coordinates.

---

### Step 4: Add Markers

```
Set<Marker> markers = {  
  Marker(  
    markerId: MarkerId('1'),  
    position: LatLng(37.42796133580664, -122.085749655962),  
    infoWindow: InfoWindow(title: 'My Marker'),  
  ),  
};  
  
GoogleMap(  
  initialCameraPosition: CameraPosition(  
    target: LatLng(37.42796133580664, -122.085749655962),  
    zoom: 14.0,  
  ),  
  markers: markers,  
)
```

**Exercise:** Add multiple markers to represent favorite places.

---

### Step 5: Get User Location

```
Location location = Location();  
LocationData userLocation = await location.getLocation();  
  
GoogleMap(  
  initialCameraPosition: CameraPosition(  
    target: LatLng(userLocation.latitude!, userLocation.longitude!),  
    zoom: 14,  
  ),  
)
```

**Exercise:** Center the map on the user's current location.

---

## Step 6: Mini Project

Build a **Flutter Location App**:

- Show Google Map with user location
  - Add markers for points of interest
  - Update map when location changes
  - Bonus: Add info windows and marker click events
- 

## Key Takeaways

- Google Maps adds **interactive maps and navigation features**
  - Markers and info windows improve **user engagement**
  - Location services enable apps to be **context-aware and real-world ready**
  - Mastering maps is essential for **travel, delivery, or location-based apps**
- 

Skipping Google Maps integration makes your app **less dynamic and interactive**, limiting location-based functionality.

Visit **haas.dev** for more Flutter guides, tutorials, and complete project resources.

Website Name: haas.dev

Website Link: <https://dev-roast-app.vercel.app>

---