

**LAPORAN PRAKTIKUM
PEMROGRAMAN PERANGKAT BERGERAK**

**MODUL XII
Maps & Places**



**Disusun Oleh :
Arzario Irsyad Al Fatih/2211104032
SE 06 2**

**Asisten Praktikum :
Muhammad Faza Zulian Gesit Al Barru
Aisyah Hasna Aulia**

**Dosen Pengampu :
Yudha Islami Sulistya**

**PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024**

1. GUIDED

Source Code

- main.dart

```
import 'package:flutter/material.dart';
import 'package:praktikum_12/homepage.dart';

void main() async {
  runApp(const MyApp());
}

String? token;

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Map',
      theme: ThemeData(primarySwatch: Colors.orange),
      home: const MyMaps(),
    );
  }
}
```

- homepage.dart

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

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

  @override
  State<MyMaps> createState() => _MyMapsState();
}

class _MyMapsState extends State<MyMaps> {
  static const LatLng _kMapCenter =
    LatLng(19.018255973653343, 72.84793849278007);
  static const CameraPosition _kInitialPosition =
    CameraPosition(
```

```

        target: _kMapCenter,
        zoom: 11.0,
    );

    // Set marker
    Set<Marker> _createMarker() {
        return {
            const Marker(
                markerId: MarkerId("marker_1"),
                position: _kMapCenter,
                infoWindow: InfoWindow(title: 'Marker 1'),
                rotation: 90,
            ),
            const Marker(
                markerId: MarkerId("marker_2"),
                position: LatLng(-6.9733165, 107.6281415),
                infoWindow: InfoWindow(title: 'Marker 2'),
            ),
        };
    }

    // Show Place Picker
    void showPlacePicker() async {
        LocationResult result = await
Navigator.of(context).push(MaterialPageRoute(
    builder: (context) => PlacePicker(
        "AIzaSyB3q0WTwH3yT1Jhg3neUB4pKE-J9c4hcw4I",
        displayLocation: _kMapCenter,
    ),
));

        print(result);
    }

    @override
    Widget build(BuildContext context) {
        return Scaffold(
            appBar: AppBar(
                title: const Text('Google Maps Demo'),
            ),
            body: Column(
                children: [
                    InkWell(

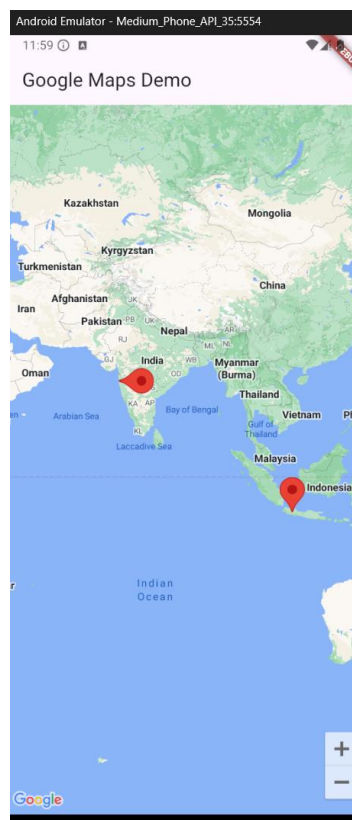
```

```

onTap: showPlacePicker,
child: Container(
  padding: const EdgeInsets.all(16.0),
  color: Colors.blueAccent,
  child: const Text(
    'Pick a Place',
    style: TextStyle(color: Colors.white, fontSize:
18),
  ),
),
),
),
Expanded(
  child: GoogleMap(
    initialCameraPosition: _kInitialPosition,
    myLocationEnabled: true,
    markers: _createMarker(),
  ),
),
],
),
);
}
}

```

Output



Deskripsi

Kode di atas adalah implementasi Flutter yang menggunakan Google Maps API dan package Place Picker untuk menampilkan peta interaktif dengan fitur penandaan lokasi (marker) dan pemilihan lokasi. Kelas **MyMaps** adalah sebuah widget stateful yang menampilkan peta menggunakan widget **GoogleMap**. Peta dimulai dengan posisi kamera awal (*kInitialPosition*) pada koordinat tertentu. Dua marker ditambahkan pada peta: satu di posisi awal (*kMapCenter*) dan satu lagi di lokasi berbeda. Terdapat tombol "Pick a Place" yang memanfaatkan package **place_picker** untuk menampilkan antarmuka pemilih lokasi dengan menggunakan API Key Google Maps. Lokasi yang dipilih oleh pengguna ditampilkan di konsol menggunakan `print(result)`. Widget ini juga mengaktifkan fitur lokasi pengguna (*myLocationEnabled*) untuk menampilkan posisi perangkat saat ini pada peta.

2. UNGUIDED

Source Code

- main.dart

```
import 'package:flutter/material.dart';
import 'package:praktikum_12/picker.dart';

void main() async {
  runApp(const MyApp());
}

String? token;

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Map',
      theme: ThemeData(primarySwatch: Colors.orange),
      home: const MyApp(),
    );
  }
}
```

- picker.dart

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

void main() {
  runApp(MyApp());
}

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

class MapsScreen extends StatefulWidget {
  @override
  _MapsScreenState createState() => _MapsScreenState();
}

class _MapsScreenState extends State<MapsScreen> {
  static const String apiKey =
    'AIzaSyCcGQMxcxRNqGTzL1shPmHq8CGKYsg80Qc';
  static final LatLng _kInitialLocation =
    LatLng(19.018255973653343, 72.84793849278007);

  LatLng? _pickedLocation;
  GoogleMapController? _mapController;
  List<dynamic> _nearbyPlaces = [];
  bool _showNearbyPlaces = false;

  // Fetch nearby places
  Future<void> _fetchNearbyPlaces(LatLng location) async {
    final url = Uri.parse(
      'https://maps.googleapis.com/maps/api/place/nearbysearch/
json'
      '?location=${location.latitude},${location.longitude}'
    );
  }
}
```

```

        '&radius=1500'
        '&key=$apiKey',
    );

    final response = await http.get(url);
    if (response.statusCode == 200) {
        final data = json.decode(response.body);
        print("Nearby places response: $data");
        setState(() {
            if (data['results'] != null &&
data['results'].isNotEmpty) {
                _nearbyPlaces = data['results'];
                _showNearbyPlaces = true;
            } else {
                _showNearbyPlaces = false;
            }
        });
    } else {
        throw Exception('Failed to fetch nearby places');
    }
}

// Navigate to Place Picker
Future<void> _pickPlace() async {
    Navigator.push(
        context,
        MaterialPageRoute(
            builder: (context) => PlacePicker(
                apiKey: apiKey,
                initialLocation:
                    _kInitialLocation,
                onPlacePicked: (result) {
                    Navigator.of(context).pop();
                    setState(() {
                        _pickedLocation = LatLng(
                            result.geometry!.location.lat,
                            result.geometry!.location.lng,
                        );
                    });
                },
            ),
        ),
    );
}

```

```

}

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text('Place Picker Demo'),
      actions: [
        IconButton(
          icon: Icon(Icons.search),
          onPressed: _pickPlace,
        ),
      ],
    ),
    body: Stack(
      children: [
        GoogleMap(
          initialCameraPosition: CameraPosition(
            target: _kInitialLocation,
            zoom: 11.0,
          ),
          onMapCreated: (controller) {
            _mapController = controller;
          },
          onTap: (LatLng location) {
            setState(() {
              _pickedLocation = location;
            });
          },
          markers: _pickedLocation == null
            ? {}
            : {
                Marker(
                  markerId: MarkerId('picked-location'),
                  position: _pickedLocation!,
                ),
              },
        ),
        if (_pickedLocation != null)
          Positioned(
            bottom: 100,
            left: 10,
            right: 10,

```



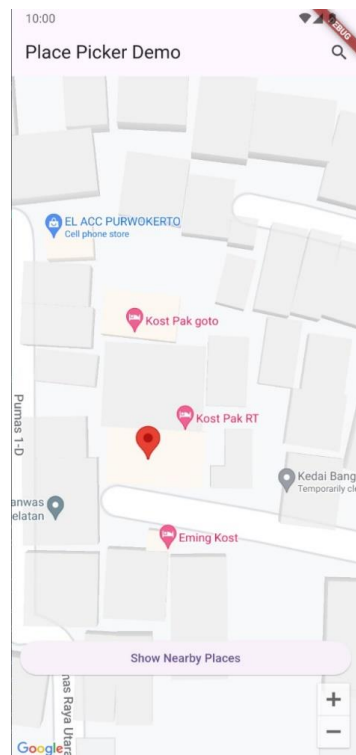
```

        child: ElevatedButton(
          onPressed: () async {
            if (_pickedLocation != null) {
              await _fetchNearbyPlaces(_pickedLocation!);
            }
          },
          child: Text('Show Nearby Places'),
        ),
      ),
    if (_showNearbyPlaces)
      Positioned(
        bottom: 10,
        left: 10,
        right: 10,
        child: Container(
          height: 250,
          child: ListView.builder(
            itemCount: _nearbyPlaces.length,
            itemBuilder: (context, index) {
              final place = _nearbyPlaces[index];
              return ListTile(
                title: Text(place['name']),
                subtitle: Text(place['vicinity'] ?? ''),
              );
            },
          ),
        ),
      ),
    ],
  ),
);
}

extension on LocationResult {
  get geometry => null;
}

```

Output



Deskripsi

Kode di atas adalah aplikasi Flutter yang menggunakan Google Maps API dan Google Places API untuk menampilkan peta interaktif dengan fitur pemilihan lokasi, menambahkan marker, dan menampilkan tempat-tempat terdekat (nearby places). Kelas `MapsScreen` adalah widget stateful yang memanfaatkan widget `GoogleMap` untuk menampilkan peta dengan posisi awal (*initialLocation*) yang telah ditentukan. Pengguna dapat memilih lokasi di peta dengan mengetuk titik tertentu, atau menggunakan `PlacePicker` untuk memilih lokasi secara interaktif. Lokasi yang dipilih ditampilkan sebagai marker pada peta.

Fitur tambahan:

1. **Fetch Nearby Places:** Saat lokasi dipilih, pengguna dapat menekan tombol "Show Nearby Places", yang akan memanggil Google Places API melalui endpoint `nearbysearch` untuk mengambil daftar tempat di sekitar lokasi yang dipilih. Hasilnya ditampilkan dalam bentuk daftar di bagian bawah layar.
2. **Place Picker Integration:** Tombol pencarian di AppBar membuka antarmuka `Place Picker` untuk memudahkan pengguna memilih lokasi menggunakan alat pencarian dan peta.

Aplikasi ini mengintegrasikan HTTP request untuk mengambil data dari API Google, serta menggunakan package `google_maps_flutter` dan `place_picker_google` untuk menampilkan peta dan fitur terkait. `apiKey` diperlukan untuk mengakses layanan API Google Maps dan Places.