

MAKALAH UJIAN TENGAH SEMESTER
GOOGLE MAPS, GLOBAL POSITIONING SYSTEM (GPS), DAN LOCATION-BASED
SERVICES

Disusun untuk memenuhi tugas kuliah daring Pemrograman Mobile 2
oleh Bapak Nandang Hermanto, M.Kom.



DISUSUN OLEH

Dwi Candra Permana

18.11.0004 / IF 2018 A

PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS AMIKOM PURWOKERTO
PURWOKERTO
2020

A. GOOGLE MAPS

Google Maps adalah suatu peta dunia yang dapat kita gunakan untuk melihat suatu daerah. Dengan kata lain, Google Maps merupakan suatu peta yang dapat dilihat dengan menggunakan suatu browser. Kita dapat menambahkan fitur Google Maps dalam web yang telah kita buat atau pada blog kita yang berbayar maupun gratis sekalipun dengan Google Maps API. Google Maps API adalah suatu library yang berbentuk JavaScript.

Pada *Google Maps API* terdapat 4 jenis pilihan model peta yang disediakan oleh Google, di antaranya adalah:

- » **ROADMAP**, untuk menampilkan peta biasa 2 dimensi
- » **SATELLITE**, untuk menampilkan foto satelit
- » **TERRAIN**, untuk menunjukkan relief fisik permukaan bumi dan menunjukkan seberapa tingginya suatu lokasi, contohnya akan menunjukkan gunung dan sungai
- » **HYBRID**, akan menunjukkan foto satelit yang di atasnya tergambar pula apa yang tampil pada ROADMAP (jalan dan nama kota).

B. GPS (*GLOBAL POSITIONING SYSTEM*)

GPS adalah sistem navigasi yang menggunakan satelit yang didesain agar dapat menyediakan posisi secara instan, kecepatan dan informasi waktu di hampir semua tempat di muka bumi, setiap saat dan dalam kondisi cuaca apa pun.

Sedangkan alat untuk menerima sinyal satelit yang dapat digunakan oleh pengguna secara umum dinamakan GPS Tracker atau GPS Tracking, dengan menggunakan alat ini maka dimungkinkan user dapat melacak posisi kendaraan, armada ataupun mobil dalam keadaan Real-Time.

Cara Kerja GPS

Bagian yang paling penting dalam sistem navigasi GPS adalah beberapa satelit yang berada di orbit bumi atau yang sering kita sebut di ruang angkasa. Satelit GPS saat ini berjumlah 24 unit yang semuanya dapat memancarkan sinyal ke bumi yang lalu dapat ditangkap oleh alat penerima sinyal tersebut atau GPS Tracker.

3 bagian penting dalam sistem GPS :

- » **Control segment GPS**, terdiri dari lima stasiun yang berada di pangkalan Falcon Air Force, Colorado Springs, Ascension Island, Hawaii, Diego Garcia dan Kwajalein. Kelima stasiun ini adalah mata dan telinga bagi GPS. Sinyal-sinyal dari satelit diterima oleh bagian kontrol, kemudian dikoreksi, dan dikirimkan kembali ke satelit. Data koreksi lokasi yang tepat dari satelit ini disebut data ephemeris, yang kemudian nantinya dikirimkan ke alat navigasi yang kita miliki.
- » **GPS Space Segment** adalah terdiri dari sebuah jaringan satelit yang terdiri dari beberapa satelit yang berada pada orbit lingkaran yang terdekat dengan tinggi nominal sekitar 20.183

KM di atas permukaan bumi. Sinyal yang dipancarkan oleh seluruh satelit tersebut dapat menembus awan, plastik dan kaca, namun tidak bisa menembus benda padat seperti tembok dan rapatnya pepohonan.

- » **GPS User Segment**, terdiri dari antena dan prosesor receiver yang menyediakan positioning, kecepatan dan ketepatan waktu ke pengguna. Bagian ini menerima data dari satelit-satelit melalui sinyal radio yang dikirimkan setelah mengalami koreksi oleh stasiun pengendali (GPS Control Segment).

Fungsi dan kegunaan GPS

- » **GPS untuk Militer**

GPS dapat dimanfaatkan untuk mendukung sistem pertahanan militer. Lebih jauh dari itu bisa memantau pergerakan musuh saat terjadi peperangan, juga bisa menjadi penuntun arah jatuhnya bom sehingga bisa lebih tertarget.

- » **GPS untuk Navigasi**

Dalam kebutuhan berkendara sistem GPS pun sangat membantu, dengan adanya GPS Tracker terpasang pada kendaraan maka akan membuat perjalanan semakin nyaman karena arah dan tujuan jalan bisa diketahui setelah GPS mengirim posisi kendaraan kita yang diterjemahkan ke dalam bentuk peta digital.

- » **GPS untuk Sistem Informasi Geografis**

GPS sering juga digunakan untuk keperluan sistem informasi geografis, seperti untuk pembuatan peta, mengukur jarak perbatasan, atau bisa dijadikan sebagai referensi pengukuran suatu wilayah.

- » **GPS untuk Sistem Pelacakan Kendaraan**

Fungsi ini hampir sama dengan navigasi, jika dalam navigasi menggunakan perangkat penerima sinyal GPS berikut penampil titik koordinatnya dalam satu perangkat, sedangkan untuk kebutuhan sistem pelacakan adalah alat penampil dan penerima sinyal berbeda lokasi.

- » **GPS untuk Pemantau Gempa**

Saat ini teknologi GPS yang terus ditingkatkan menghasilkan tingkat ketelitian dan keakuratan yang sangat tinggi sehingga GPS dapat dimanfaatkan untuk memantau pergerakan tanah di bumi. Dengan hal itu maka para pakar Geologi dapat memperkirakan kemungkinan terjadinya gempa di suatu wilayah.

C. Location Based Service (LBS)

Location Based Service (LBS) atau layanan berbasis lokasi adalah sebuah layanan informasi yang dapat diakses dengan perangkat bergerak melalui jaringan dan mampu menampilkan posisi secara geografis keberadaan perangkat bergerak tersebut. *Location Based Service* dapat berfungsi sebagai layanan untuk mengidentifikasi lokasi dari seseorang atau suatu objek tertentu, seperti menemukan lokasi mesin ATM terdekat atau mengetahui keberadaan teman.

Unsur Utama LBS

» **Location Manager (API Maps)**

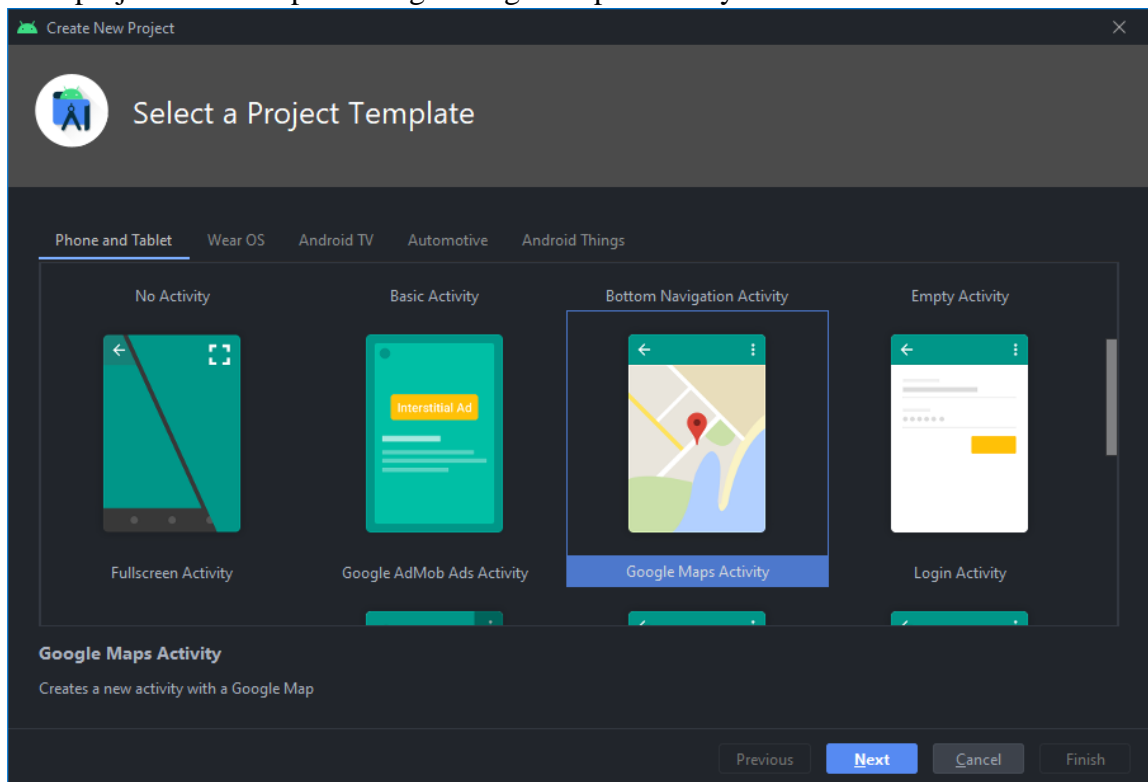
Menyediakan perangkat bagi sumber atau *source* untuk LBS, *Application Programming Interface (API) Maps* menyediakan fasilitas untuk menampilkan atau memanipulasi peta. Paket ini berada pada “**com.google.android.apps.maps**”.

» **Location Providers (API Location)**

Menyediakan teknologi pencarian lokasi yang digunakan oleh perangkat. API Location berhubungan dengan data GPS (*Global Positioning System*) dan data lokasi *real-time*. *API Location* berada pada paket Android yaitu dalam paket “**android.location**”. Lokasi, perpindahan, serta kedekatan dengan lokasi tertentu dapat ditentukan melalui *Location Manager*.

D. DASAR PEMBUATAN MAPS DAN LOCATION-BASED

1. Buat project baru dan pilih design Google Maps Activity.



2. Tambahkan beberapa dependency berikut ke dalam file build.gradle (Module: app)

```
dependencies {  
    implementation fileTree(dir: 'libs', include: ['*.jar'])  
    implementation 'com.android.support:appcompat-v7:27.0.0'  
    implementation 'com.google.android.gms:play-services-maps:15.0.1'  
    implementation 'com.android.support.constraint:constraint-  
layout:1.0.2'  
    implementation 'org.jetbrains:annotations-java5:15.0'  
    testImplementation 'junit:junit:4.12'  
    androidTestImplementation 'com.android.support.test:runner:1.0.2'
```

```

        androidTestImplementation
        'com.android.support.test.espresso:espresso-core:3.0.2'
        implementation 'com.google.android.gms:play-services-places:16.0.0'
        implementation 'com.google.android.gms:play-services-
location:16.0.0'
    }

```

3. Tambahkan beberapa permission berikut ke dalam manifest AndroidManifest.xml.

```

<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"
/>
<uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATION" />

```

4. Tambahkan beberapa metadata berikut ke dalam application AndroidManifest.xml

```

<meta-data
    android:name="com.google.android.gms.version"
    android:value="@integer/google_play_services_version" />
<meta-data
    android:name="com.google.android.geo.API_KEY"
    android:value="@string/google_maps_key" />

```

5. Tentukan API Key yang akan diaplikasikan ke dalam project di res > values > google_maps_api.xml

```

<resources>
    <!--
        TODO: Before you run your application, you need a Google Maps API
        key.

        To get one, follow this link, follow the directions and press
        "Create" at the end:

        https://console.developers.google.com/flows/enableapi?apiid=maps_android_
        _backend&keyType=CLIENT_SIDE_ANDROID&r=EB:A3:89:32:3E:F7:F6:FC:F0:7A:EE:
        FF:23:A9:87:4F:70:AD:27:B7%3Bcom.example.vendi.mymaps

        You can also add your credentials to an existing key, using these
        values:

        Package name:
        EB:A3:89:32:3E:F7:F6:FC:F0:7A:EE:FF:23:A9:87:4F:70:AD:27:B7

        SHA-1 certificate fingerprint:
        EB:A3:89:32:3E:F7:F6:FC:F0:7A:EE:FF:23:A9:87:4F:70:AD:27:B7

        Alternatively, follow the directions here:
        https://developers.google.com/maps/documentation/android/start#get-
        key

        Once you have your key (it starts with "AIza"), replace the
        "google_maps_key"
    
```

```

string in this file.
-->
<string name="google_maps_key" templateMergeStrategy="preserve"
translatable="false">AIzaSyDjkDzp8Qf*****qn6gyFY3z****UsIFHw</string>
</resources>

```

Untuk mendapatkan API Key, klik link yang diberikan pada file tersebut. Ikuti petunjuk yang ada di halaman website yang tampil di layar.

6. Buat Android Resource baru di folder values dengan nama ids.xml. Isi file tersebut dengan baris kode berikut.

```

<?xml version="1.0" encoding="utf-8"?>
<resources>
    <item name="btnCreateRoutes" type="id" />
    <item name="etOrigin" type="id" />
</resources>

```

7. Tambahkan baris kode layout di bawah ini pada file activity_maps.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <EditText
        android:id="@id/etOrigin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Lokasi anda"
        android:importantForAutofill="no" />

    <EditText
        android:id="@id/etDestination"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Lokasi Tujuan"
        android:importantForAutofill="no" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="end"
        android:orientation="horizontal">

        <Button
            android:id="@id/btnNavToTarget"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:padding="16dp"
            android:text="Navigasi ke Tujuan"

```

```

        android:visibility="gone"
        tools:ignore="ButtonStyle" />

<Button
    android:id="@+id/btnFindAsMaps"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:padding="16dp"
    android:text="Cari Lokasi" />

<Button
    android:id="@+id/btnCreateRoutes"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:padding="16dp"
    android:text="Buat Rute" />

</LinearLayout>

<fragment
    android:id="@+id/map"
    android:name="com.google.android.gms.maps.SupportMapFragment"
    android:layout_width="match_parent"
    android:layout_height="370dp" />

<RelativeLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:padding="8dp"
    android:orientation="vertical">

    <ImageView
        android:layout_width="20dp"
        android:layout_height="20dp"
        android:src="@drawable/ic_place_black_24dp" />

    <TextView
        android:id="@+id/tvDistance"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:text="Jarak tempuh: 0 km"
        android:textSize="14sp"
        android:layout_marginStart="30dp" />

    <ImageView
        android:layout_width="20dp"
        android:layout_height="20dp"
        android:layout_marginTop="20dp"
        android:src="@drawable/ic_timer_black_24dp" />

    <TextView

```

```

        android:id="@+id/tvDuration"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:layout_marginTop="20dp"
        android:text="Waktu perkiraan sampai: 0 min"
        android:textSize="14sp" />

    </RelativeLayout>
</LinearLayout>

```

8. Tambahkan implementasi OnMapReadyCallback pada class MapsActivity.java.

```

. . .

public class MapsActivity extends AppCompatActivity implements
    OnMapReadyCallback {

. . .

```

9. Setelah itu, tambahkan implementasi method (override) di class tersebut.

```

@Override
public void onMapReady(GoogleMap googleMap) {

}

```

10. Tambahkan beberapa baris kode di bawah ini ke dalam event onCreate. Kode tersebut sebagai dasar implementasi Google Maps pada aplikasi yang dibuat.

```

private GoogleMap mMap;
private EditText etOrigin;
private EditText etDestination;
private List<Marker> originMarkers = new ArrayList<>();
private List<Marker> destinationMarkers = new ArrayList<>();
private List<Polyline> polylinePaths = new ArrayList<>();
private Button btnNavToTarget;
private LatLng destination;
private ProgressDialog progressDialog;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_maps);

    if (ContextCompat.checkSelfPermission(this,
        Manifest.permission.ACCESS_FINE_LOCATION)
        != PackageManager.PERMISSION_GRANTED) {

        // Izin lokasi ditolak
        if (ActivityCompat.shouldShowRequestPermissionRationale(this,

```



```

        Manifest.permission.ACCESS_FINE_LOCATION)) {
            Toast.makeText(this, "Membutuhkan Izin Lokasi",
                Toast.LENGTH_SHORT).show();
        } else {
            // permintaan akses lokasi tanpa izin
            ActivityCompat.requestPermissions(this,
                new
                String[]{Manifest.permission.ACCESS_FINE_LOCATION,
                    Manifest.permission.ACCESS_COARSE_LOCATION},
                    1);
        }

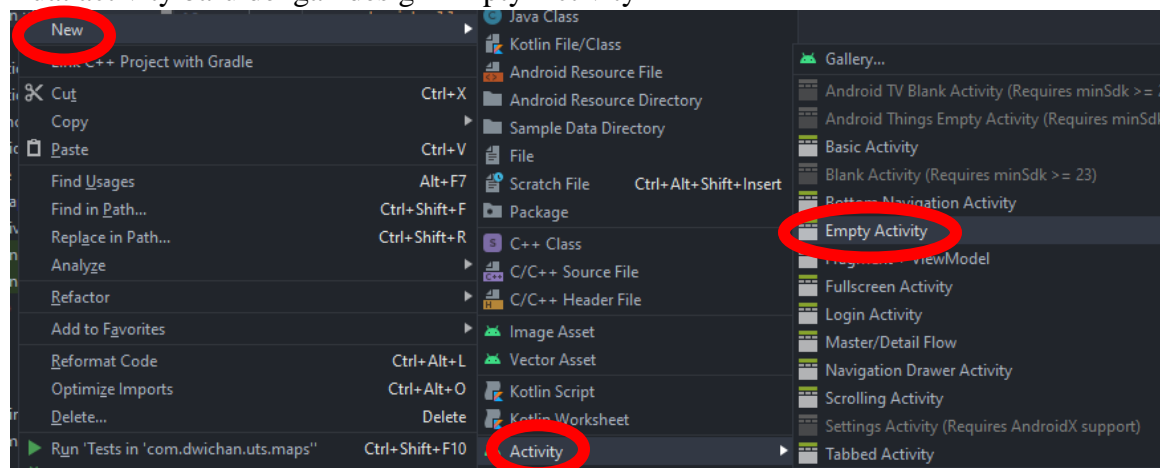
    } else {
        // Izin lokasi diberikan
        Toast.makeText(this, "Izin Lokasi diberikan",
            Toast.LENGTH_SHORT).show();
    }

    // Dasar pembuatan Maps
    SupportMapFragment mapFragment = (SupportMapFragment)
        getSupportFragmentManager()
            .findFragmentById(R.id.map);
    mapFragment.getMapAsync(this);
    // akhir dari dasar pembuatan Maps
}

```

E. MENEMUKAN LOKASI YANG TERSEDIA DAN MENEMUKAN LOKASI DENGAN KRITERIA TERTENTU

1. Buat activity baru dengan design Empty Activity



2. Beri nama FindAsMapsActivity ke activity baru, lalu tambahkan kode berikut ke dalam activity_find_as_maps.xml

```

<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"

```

```

xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:padding="12dp"
tools:context=".FindAsMapsActivity">

<EditText
    android:id="@+id/tvCriteria"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Cari nama tempat atau kriteria..."
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

<Button
    android:id="@+id/btnSearch"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:text="Cari"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/tvCriteria" />

<Button
    android:id="@+id/btnCancel"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:text="Batal"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/tvCriteria" />

</android.support.constraint.ConstraintLayout>

```

3. Tambahkan style baru ke dalam res > values > styles.xml

```

<style name="FindAsMaps" parent="Theme.AppCompat.Dialog">

</style>

```

4. Buka AndroidManifest.xml. Ubah property pada tag <activity> yang terkait dengan FindAsMapsActivity menjadi seperti berikut.

```

<activity android:name="com.dwichan.uts.maps.FindAsMapsActivity"
    android:theme="@style/FindAsMaps" />

```

5. Tambahkan kode berikut pada event onClick di class FindAsMapsActivity.java

```

@Override
protected void onCreate(Bundle savedInstanceState) {

```

```

super.onCreate(savedInstanceState);
setContentView(R.layout.activity_find_as_maps);

setTitle("Cari Lokasi");

final TextView tvCriteria = findViewById(R.id.tvCriteria);
Button btnSearch = findViewById(R.id.btnSearch);
Button btnCancel = findViewById(R.id.btnCancel);

btnSearch.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        // Menemukan lokasi yang tersedia dan dengan kriteria
        tertentu
        if (tvCriteria.getText().toString().equals("")) {
            Toast.makeText(FindAsMapsActivity.this, "Tentukan
            kriteria pencariannya!", Toast.LENGTH_SHORT).show();
        } else {
            String criteria = tvCriteria.getText().toString();
            Uri uriGoogleMaps = Uri.parse("geo:0,0?q=" + criteria);
            Intent mapIntent = new Intent(Intent.ACTION_VIEW,
            uriGoogleMaps);
            mapIntent.setPackage("com.google.android.apps.maps");
            startActivity(mapIntent);
        }
        // akhir dari Menemukan lokasi yang tersedia dan dengan
        kriteria tertentu
    }
});

btnCancel.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        setResult(RESULT_CANCELED);
        finish();
    }
});
}

```

6. Tambahkan kode berikut pada event onCreate di class MapsActivity.java

```

Button btnFindAsMaps = findViewById(R.id.btnFindAsMaps);
btnFindAsMaps.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent i = new Intent(MapsActivity.this,
        FindAsMapsActivity.class);
        startActivityForResult(i, REQUEST_FIND_AS_MAPS);
    }
});

```

Perhatikan kode berikut:

```
if (tvCriteria.getText().toString().equals("")) {
    Toast.makeText(FindAsMapsActivity.this, "Tentukan kriteria
pencariannya!", Toast.LENGTH_SHORT).show();
} else {
    String criteria = tvCriteria.getText().toString();
    Uri uriGoogleMaps = Uri.parse("geo:0,0?q=" + criteria);
    Intent mapIntent = new Intent(Intent.ACTION_VIEW, uriGoogleMaps);
    mapIntent.setPackage("com.google.android.apps.maps");
    startActivity(mapIntent);
}
```

Kode tersebut adalah kode yang akan mengirimkan intent ke dalam aplikasi Google Maps (**com.google.android.apps.maps**) yang berisi Uri yang memiliki parameter yang pengguna ketikkan untuk melakukan pencarian lokasi yang tersedia dan menemukan lokasi dengan kriteria tertentu.

F. MENEMUKAN LOKASI YANG DITENTUKAN

Buka class MapsActivity.java. Tambahkan kode berikut pada bagian event onMapReady.

```
@Override
public void onMapReady(GoogleMap googleMap) {
    // Menemukan lokasi yang ditentukan
    LatLng amik = new LatLng(-7.400798, 109.231160);
    mMap = googleMap;
    mMap.addMarker(new MarkerOptions().position(amik).title("Universitas
AMIKOM Purwokerto"));
    mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(amik, 16));
    // akhir dari Menemukan lokasi yang ditentukan
}
```

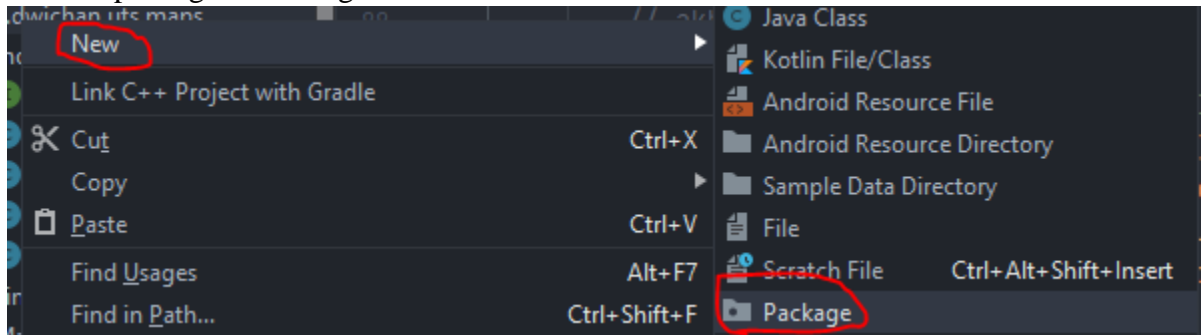
Perhatikan kode berikut.

```
LatLng amik = new LatLng(-7.400798, 109.231160);
mMap = googleMap;
mMap.addMarker(new MarkerOptions().position(amik).title("Universitas AMIKOM
Purwokerto"));
mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(amik, 16));
```

Kode tersebut berfungsi untuk menentukan lokasi di mana letak yang akan diberi marker. Dalam kasus ini, Universitas AMIKOM Purwokerto adalah lokasi yang diberikan marker pada Maps.

G. MEMBUAT RUTE DAN MENCARI RUTE TERCEPAT

1. Buatlah package baru dengan nama model.



2. Buatlah beberapa class berikut ke dalam project.
 - a. Route.java

```
package com.dwichan.uts.maps.model;

import com.google.android.gms.maps.model.LatLng;
import java.util.List;

public class Route {
    public Distance distance;
    public Duration duration;
    public String endAddress;
    public LatLng endLocation;
    public LatLng startLocation;
    public String startAddress;
    public List<LatLng> points;
}
```

- b. Duration.java

```
package com.dwichan.uts.maps.model;

public class Duration {
    public String text;
    public int value;

    public Duration(String text, int value) {
        this.text = text;
        this.value = value;
    }
}
```

- c. Distance.java

```
package com.dwichan.uts.maps.model;

public class Distance {
```

```

    public String text;
    public int value;

    public Distance(String text, int value) {
        this.text = text;
        this.value = value;
    }
}

```

d. DirectionFindList.java

```

package com.dwichan.uts.maps.model;

import java.util.List;

public interface DirectionFindList {
    void onDirectionFindStart();

    void onDirectionFindSuccess(List<Route> routes);
}

```

e. DirectionFindPath.java

```

package com.dwichan.uts.maps.model;

import android.annotation.SuppressLint;
import android.os.AsyncTask;

import com.google.android.gms.maps.model.LatLng;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.UnsupportedEncodingException;
import java.net.MalformedURLException;
import java.net.URL;
import java.net.URLEncoder;
import java.util.ArrayList;
import java.util.List;

public class DirectionFindPath {
    private static final String DIRECTION_URL_API =

"https://maps.googleapis.com/maps/api/directions/json?";
    private static final String GOOGLE_API_KEY =
        "AIzaSyDjkDzp8Qf*****qn6gyFY3z*****UsIFHw";
}

```

```

        private final DirectionFindList listener;
        private final String origin;
        private final String destination;

        public DirectionFindPath(DirectionFindList listener, String
origin, String destination) {
            this.listener = listener;
            this.origin = origin;
            this.destination = destination;
        }

        public void execute() throws UnsupportedOperationException {
            listener.onDirectionFindStart();
            new DownloadRawData().execute(createUrl());
        }

        private String createUrl() throws UnsupportedOperationException
{
            String urlOrigin = URLEncoder.encode(origin, "utf-8");
            String urlDestination = URLEncoder.encode(destination,
"utf-8");

            return DIRECTION_URL_API + "origin=" + urlOrigin +
"&destination="
                + urlDestination + "&key=" + GOOGLE_API_KEY;
        }

        @SuppressWarnings("StaticFieldLeak")
        private class DownloadRawData extends AsyncTask<String, Void,
String> {
            @Override
            protected String doInBackground(String... params) {
                String link = params[0];
                try {
                    URL url = new URL(link);
                    InputStream is =
url.openConnection().getInputStream();
                    StringBuilder buffer = new StringBuilder();
                    BufferedReader reader = new BufferedReader(new
InputStreamReader(is));

                    String line;
                    while ((line = reader.readLine()) != null) {
                        buffer.append(line).append("\n");
                    }
                    return buffer.toString();
                } catch (MalformedURLException e) {
                    e.printStackTrace();
                } catch (IOException e) {
                    e.printStackTrace();
                }
                return null;
            }
        }
    }

```

```

    }

    @Override
    protected void onPostExecute(String res) {
        try {
            parseJson(res);
        } catch (JSONException e) {
            e.printStackTrace();
        }
    }

    private void parseJson(String data) throws JSONException {
        if (data == null)
            return;
        List<Route> routes = new ArrayList<>();
        JSONObject jsonData = new JSONObject(data);
        JSONArray jsonRoutes =
jsonData.getJSONArray("routes");
        for (int i = 0; i < jsonRoutes.length(); i++) {
            JSONObject jsonRoute =
jsonRoutes.getJSONObject(i);
            Route route = new Route();

            JSONObject overview_polylineJson =
jsonRoute.getJSONObject("overview_polyline");
            JSONArray jsonLegs =
jsonRoute.getJSONArray("legs");
            JSONObject jsonLeg = jsonLegs.getJSONObject(0);
            JSONObject jsonDistance =
jsonLeg.getJSONObject("distance");
            JSONObject jsonDuration =
jsonLeg.getJSONObject("duration");
            JSONObject jsonEndLocation =
jsonLeg.getJSONObject("end_location");
            JSONObject jsonStartLocation =
jsonLeg.getJSONObject("start_location");

            route.distance = new
Distance(jsonDistance.getString("text"),
jsonDistance.getInt("value"));
            route.duration = new
Duration(jsonDuration.getString("text"),
jsonDuration.getInt("value"));
            route.endAddress =
jsonLeg.getString("end_address");
            route.startAddress =
jsonLeg.getString("start_address");
            route.startLocation = new
LatLng(jsonStartLocation.getDouble("lat"),
jsonStartLocation.getDouble("lng"));
            route.endLocation = new

```



```

LatLng(jsonEndLocation.getDouble("lat"),
jsonEndLocation.getDouble("lng"));
        route.points =
decodePolyLine(overview_polylineJson.getString("points"));

        routes.add(route);
    }
    listener.onDirectionFindSuccess(routes);

}

// perhitungan untuk menentukan rute
@org.jetbrains.annotations.NotNull
private List<LatLng> decodePolyLine(final String poly) {
    int len = poly.length();
    int index = 0;
    List<LatLng> decoded = new ArrayList<>();
    int lat = 0;
    int lng = 0;

    while (index < len) {
        int b;
        int shift = 0;
        int result = 0;
        do {
            b = poly.charAt(index++) - 63;
            result |= (b & 0x1f) << shift;
            shift += 5;
        } while (b >= 0x20);
        int dlat = ((result & 1) != 0 ? ~(result >> 1) :
(result >> 1));
        lat += dlat;

        shift = 0;
        result = 0;
        do {
            b = poly.charAt(index++) - 63;
            result |= (b & 0x1f) << shift;
            shift += 5;
        } while (b >= 0x20);
        int dlng = ((result & 1) != 0 ? ~(result >> 1) :
(result >> 1));
        lng += dlng;

        decoded.add(new LatLng(
            lat / 100000d, lng / 100000d
        ));
    }

    return decoded;
}

```

```
}  
}
```

3. Buka MapsActivity.java. Lakukan implementasi class ke DirectionFindList saat pendefinisian class.

```
. . .  
  
public class MapsActivity extends AppCompatActivity implements  
OnMapReadyCallback, DirectionFindList {  
  
. . .
```

4. Override method yang dibutuhkan oleh DirectionFindList

```
@Override  
public void onDirectionFindSuccess(List<Route> routes) {  
  
}  
  
@Override  
public void onDirectionFindStart() {  
  
}
```

5. Isilah method onDirectionFindStart dan onDirectionFindSuccess yang tadi dengan beberapa perintah berikut.

```
@Override  
public void onDirectionFindStart() {  
    btnNavToTarget.setVisibility(View.GONE);  
    progressDialog = ProgressDialog.show(this, "Sebentar...",  
        "Aku lagi nyari lokasinya, tunggu ya...", true);  
  
    if (originMarkers != null) {  
        for (Marker marker : originMarkers) {  
            marker.remove();  
        }  
    }  
  
    if (destinationMarkers != null) {  
        for (Marker marker : destinationMarkers) {  
            marker.remove();  
        }  
    }  
  
    if (polylinePaths != null) {  
        for (Polyline polyline : polylinePaths) {  
            polyline.remove();  
        }  
    }  
}
```

```

    }
}

@Override
public void onDirectionFindSuccess(List<Route> routes) {
    progressDialog.dismiss();
    polylinePaths = new ArrayList<>();
    originMarkers = new ArrayList<>();
    destinationMarkers = new ArrayList<>();

    // Membuat dan mencari rute tercepat
    for (Route route : routes) {

mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(route.startLocation,
18));

        String distances = "Jarak tempuh: " + route.distance.text;
        ((TextView) findViewById(R.id.tvDistance)).setText(distances);

        String durations = "Waktu perkiraan sampai: " +
route.duration.text;
        ((TextView) findViewById(R.id.tvDuration)).setText(durations);

        originMarkers.add(mMap.addMarker(new MarkerOptions()
            .title(route.startAddress)
            .position(route.startLocation)));
        destinationMarkers.add(mMap.addMarker(new MarkerOptions()
            .title(route.endAddress)
            .position(route.endLocation)));

        destination = route.endLocation;
        btnNavToTarget.setVisibility(View.VISIBLE);

        PolylineOptions polylineOptions = new PolylineOptions().
            geodesic(true).
            color(Color.BLUE).
            width(10);

        for (int i = 0; i < route.points.size(); i++)
            polylineOptions.add(route.points.get(i));

        polylinePaths.add(mMap.addPolyline(polylineOptions));
    }
    // akhir dari Membuat dan Mencari rute tercepat

    mMap.animateCamera(CameraUpdateFactory.zoomTo(8f));
}

```

6. Override method onActivityResult dan berikan beberapa perintah untuk menerima data lokasi dari class PlaceAutocomplete.

```
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    if (requestCode == 1) {
        if (resultCode == RESULT_OK) {
            // retrieve the data by using getPlace() method.
            Place place = PlaceAutocomplete.getPlace(this, data);
            Log.e("Tag", "Place: " + place.getAddress() +
place.getPhoneNumber());

            ((EditText) findViewById(R.id.etOrigin))
                .setText(place.getName());
        } else if (resultCode == PlaceAutocomplete.RESULT_ERROR) {
            Status status = PlaceAutocomplete.getStatus(this, data);
            // TODO: Handle the error.
            Log.e("Tag", status.getStatusMessage());
        } // The user canceled the operation.
    } else if (requestCode == 2) {
        if (resultCode == RESULT_OK) {
            // retrieve the data by using getPlace() method.
            Place place = PlaceAutocomplete.getPlace(this, data);
            Log.e("Tag", "Place: " + place.getAddress() +
place.getPhoneNumber());

            ((EditText) findViewById(R.id.etDestination))
                .setText(place.getName());
        } else if (resultCode == PlaceAutocomplete.RESULT_ERROR) {
            Status status = PlaceAutocomplete.getStatus(this, data);
            // TODO: Handle the error.
            Log.e("Tag", status.getStatusMessage());
        } // The user canceled the operation.
    }
}
```

7. Tambahkan kode berikut ke dalam event onCreate.

```
Button btnCreateRoutes = findViewById(R.id.btnCreateRoutes);
btnCreateRoutes.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        sendRequest();
    }
});
```

8. Perintah `sendRequest()` akan error. Tambahkan method baru bernama `sendRequest` dan isi method tersebut dengan beberapa baris perintah.

```
private void sendRequest() {
    String origin = etOrigin.getText().toString();
    String destination = etDestination.getText().toString();
    if (origin.isEmpty()) {
        Toast.makeText(this, " Tolong Masukkan Alamat Anda!",
            Toast.LENGTH_SHORT).show();
        return;
    }
    if (destination.isEmpty()) {
        Toast.makeText(this, "Tolong Masukkan Tujuan anda !",
            Toast.LENGTH_SHORT).show();
        return;
    }

    try {
        new DirectionFindPath(this, origin, destination).execute();
    } catch (UnsupportedEncodingException e) {
        e.printStackTrace();
    }
}
```

Perhatikan kode berikut

```
try {
    new DirectionFindPath(this, origin, destination).execute();
} catch (UnsupportedEncodingException e) {
    e.printStackTrace();
}
```

Perintah tersebut akan memanggil API Google Maps untuk mencari rute tercepat yang diteruskan ke class `DirectionFindPath`. Sebelum diproses, perintah yang terdapat `onDirectionFindStart` akan dijalankan pertama. Lalu apabila lokasi sudah ditemukan, method `onDirectionFindSuccess` akan dijalankan. Jika terjadi error, hasil Stack Trace akan dicetak dan disimpan ke dalam logcat.

H. MENGETAHUI LOKASI DAN MEMPERBAHARUI LOKASI TERBARU

1. Buka `MapsActivity.java`. Lakukan implementasi interface pada pendefinisian class ke `GoogleMap.OnMyLocationButtonClickListener` dan `GoogleMap.OnMyLocationClickListener`.

```
...

public class MapsActivity extends AppCompatActivity implements
    OnMapReadyCallback, DirectionFindList,
    GoogleMap.OnMyLocationButtonClickListener,
    GoogleMap.OnMyLocationClickListener {
```

- ...
2. Lakukan override untuk beberapa method yang dibutuhkan oleh interface `GoogleMap.OnMyLocationButtonClickListener` dan `GoogleMap.OnMyLocationClickListener`.

```
@Override
public boolean onMyLocationButtonClick() {
    return false;
}

@Override
public void onMyLocationClick(@NonNull Location location) {
}
```

3. Berikan beberapa perintah di bawah ini ke method-method yang tadi di override.

```
@Override
public boolean onMyLocationButtonClick() {
    Toast.makeText(this, "My Location clicked!",
        Toast.LENGTH_SHORT).show();
    return true;
}

// mengetahui lokasi saat ini
@Override
public void onMyLocationClick(@NonNull Location location) {
    if (originMarkers != null) {
        for (Marker marker : originMarkers) {
            marker.remove();
        }
    }

    if (destinationMarkers != null) {
        for (Marker marker : destinationMarkers) {
            marker.remove();
        }
    }

    if (polylinePaths != null) {
        for (Polyline polyline : polylinePaths) {
            polyline.remove();
        }
    }

    LatLng myLocation = new LatLng(location.getLatitude(),
        location.getLongitude());
    mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(myLocation, 18));
}
```

Method `onMyLocationButtonClick` akan dijalankan ketika user menekan tombol My Location, dan method `onMyLocationClick` akan dijalankan setelah method `onMyLocationButtonClick` selesai dijalankan.

4. Tambahkan perintah berikut ke dalam method `onMapReady`.

```
@Override
public void onMapReady(GoogleMap googleMap) {

    . . .

    if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED &&
ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
        return;
    }

    // mengetahui lokasi saat ini serta memperbaruinya
    mMap.setMyLocationEnabled(true);
}
```



Perintah `mMap.setMyLocationEnabled(true);` akan mengaktifkan tombol My Location seperti pada gambar di samping.

REFERENSI

- D, L., & Edit, R. (2010, April 18). *Launching Google Maps Directions via an intent on Android*. Diambil kembali dari Stack Overflow:
<https://stackoverflow.com/questions/2662531/launching-google-maps-directions-via-an-intent-on-android>
- Google Maps Intents for Android*. (2020, November 13). Diambil kembali dari Google Developers: <https://developers.google.com/maps/documentation/urls/android-intents>
- Intent : Latihan Codelab - Intent dengan ResultActivity*. (2016, September 21). Diambil kembali dari Dicoding Indonesia: <https://www.dicoding.com/academies/14/tutorials/135>
- Intent : Latihan Codelab - Intent Implicit*. (2016, September 21). Diambil kembali dari Dicoding Indonesia: <https://www.dicoding.com/academies/14/tutorials/133>

TAUTAN TERKAIT

Rekaman Demo Project:

<https://youtu.be/pRRdgREjejI>

Source Code Project:

<https://github.com/dwichan0905/MapsPertemuan5PMO2>