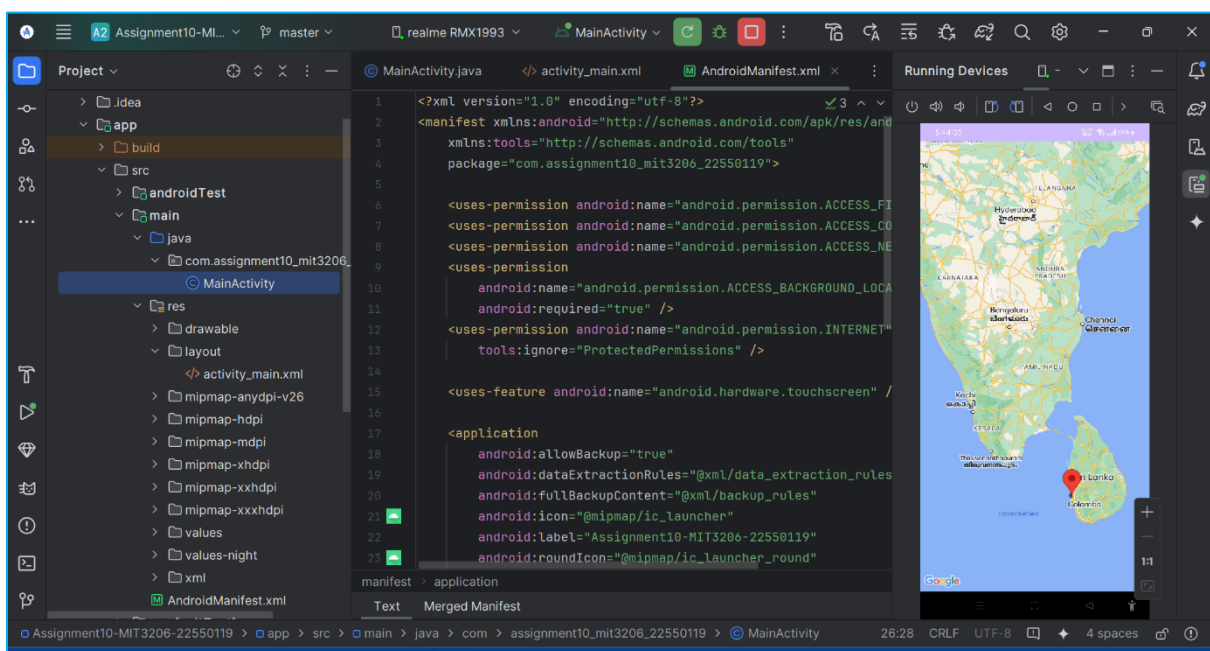


- ❖ Course Module : MIT3206 – Mobile Computing
- ❖ Course Lecturer : Senior Lecturer Gihan P. Seneviratne Sir

- ❖ Assignment 10 : Using Google Maps for Android
- ❖ Used Android Studio : Android Studio Koala | 2024.1.1
- ❖ GitHub Private Repository Link :

<https://github.com/NavinduMadusanka/Assignment10-MIT3206-22550119.git>

- ❖ Student Name : Kumarage Navindu Madusanka Dias (K.N.M. Dias)
- ❖ Student Index No : 22550119
- ❖ Student Registration No : 2022/MIT/011
- ❖ Email Address : [navindu09@gmail.com](mailto:navindu09@gmail.com)
- ❖ Contact No : +94702678624



Assignment10-MIT3206-22550119

## Assignment 10 : Using Google Maps for Android

Below is a summary of what I have learned and focused on in this assignment.

### 1. Android Components

- **API level**

Android Version	API Level	Version Name
Android 7.0	24	Nougat

- **Methods**

No	Method	Description
1	getMyLocation()	This method returns the currently displayed user location
2	clear()	This method removes everything from the map
3	onMapReady():	This function is called when the map is ready to be used
4.	buildGoogleApiClient()	This method is used to initialize the Google Play Services

- **Permissions**

- uses-permission - Access INTERNET
- uses-permission - ACCESS\_FINE\_LOCATION
- uses-permission - ACCESS\_COARSE\_LOCATION
- uses-permission - ACCESS\_NETWORK\_STATE
- uses-permission - ACCESS\_BACKGROUND\_LOCATION

- **Special Requirement**

meta - data - com.google.android.geo.API KEY

API Key & Value - "AlzaSyAl39FoH1tCOomeJt568kz2vpgzZdTnty0"

An API key is needed to access the Google Maps servers.

This key is free and we can use it with any of applications.

- **Newly learned key points in this assignment**

- How to get Google API Key
- Working with Map activity

- **Dependencies**

Dependencies in the build.gradle (:app),

- implementation libs.play.services.maps  
(implementation 'com.google.android.gms:play-services-maps:19.0.0')
- implementation libs.play.services.location  
(implementation 'com.google.android.gms:play-services-location:21.3.0')

- **tools:targetApi="34" in AndroidManifest.xml**

- **uses-feature android:name="android.hardware.touchscreen"**

- **android.enableJetifier=true in gradle.properties**

## **2. Functionality of the mobile application**

- A Location Client to provide location services
- A MapFragment to laying out Android content and display maps
- Develop an Android App which accepts a location or the Place
- Display the relevant map on the screen

### 3. Running the Application on my android mobile device

I was running the android app for testing in my android mobile device.

My android mobile device is Realme X2 RMX1993.

Below is a photo of my android mobile device (Realme X2 RMX1993) while the app was running.

My android mobile device display setting is set as dark mode option.

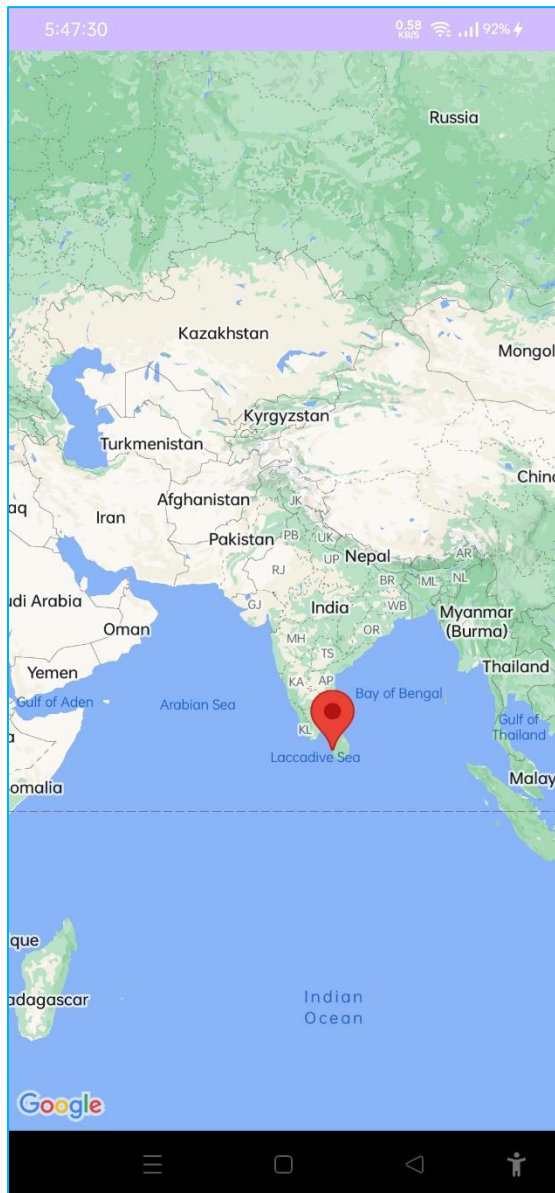


Photo 1 : Assignment10-MIT3206-22550119 in  
Realme X2 RMX1993

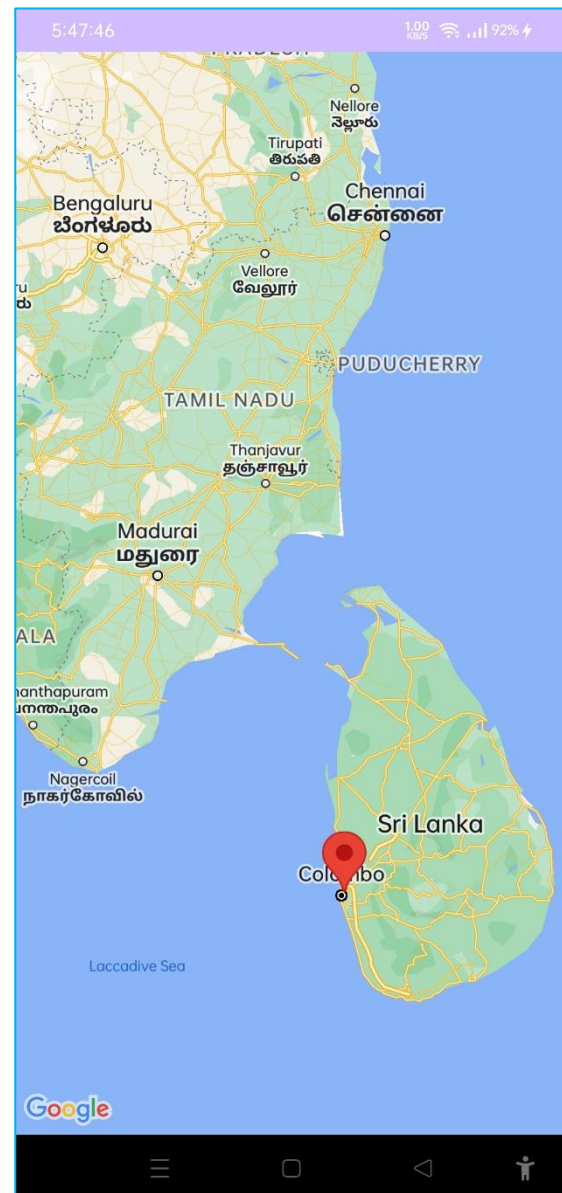
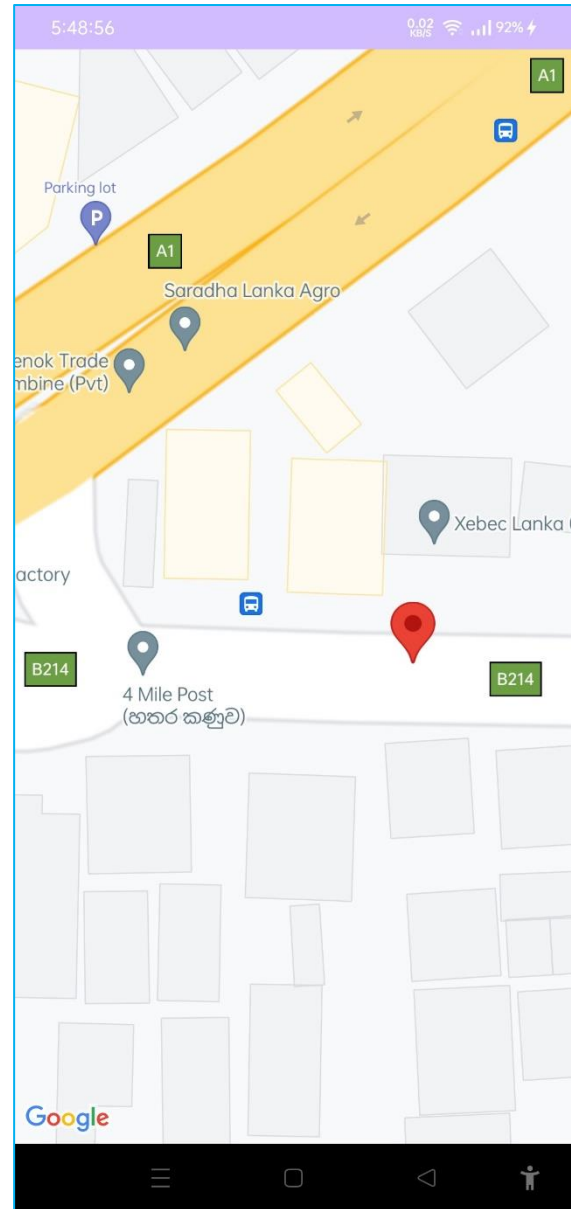


Photo 2 : Assignment10-MIT3206-22550119 in  
Realme X2 RMX1993



**Photo 3 : Assignment10-MIT3206-22550119 in  
Realme X2 RMX1993**



**Photo 4 : Assignment10-MIT3206-22550119 in  
Realme X2 RMX1993**

## 4. Main Coding files

- MainActivity.java

```
package com.assignment10_mit3206_22550119;

import android.Manifest;
import android.content.pm.PackageManager;
import android.location.Location;
import android.os.Bundle;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.core.app.ActivityCompat;
import androidx.fragment.app.FragmentActivity;

import com.google.android.gms.location.FusedLocationProviderClient;
import com.google.android.gms.location.LocationServices;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;

public class MainActivity extends FragmentActivity implements
OnMapReadyCallback {
    Location currentLocation;
    FusedLocationProviderClient fusedLocationProviderClient;
    private static final int REQUEST_CODE = 101;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        fusedLocationProviderClient =
LocationServices.getFusedLocationProviderClient(this);
        fetchLocation();
    }
    private void fetchLocation() {
        if (ActivityCompat.checkSelfPermission(
            this, android.Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission(
            this, android.Manifest.permission.ACCESS_COARSE_LOCATION)
!= PackageManager.PERMISSION_GRANTED) {
            ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.ACCESS_FINE_LOCATION}, REQUEST_CODE);
            return;
        }
        Task<Location> task =
fusedLocationProviderClient.getLastLocation();
        task.addOnSuccessListener(new OnSuccessListener<Location>() {
            @Override
            public void onSuccess(Location location) {
                if (location != null) {
                    currentLocation = location;
                    Toast.makeText(getApplicationContext(),
currentLocation.getLatitude() + "" + currentLocation.getLongitude(),
Toast.LENGTH_SHORT).show();
                }
            }
        });
    }
}
```



```

        SupportMapFragment supportMapFragment =
        (SupportMapFragment)
        getSupportFragmentManager().findFragmentById(R.id.myMap);
        assert supportMapFragment != null;
        supportMapFragment.getMapAsync(MainActivity.this);
    }
}

});
}

@Override
public void onMapReady(GoogleMap googleMap) {
    LatLng latLng = new LatLng(currentLocation.getLatitude(),
currentLocation.getLongitude());
    MarkerOptions markerOptions = new
MarkerOptions().position(latLng).title("I am here!");
    googleMap.animateCamera(CameraUpdateFactory.newLatLng(latLng));
    googleMap.animateCamera(CameraUpdateFactory.newLatLngZoom(latLng,
5));
    googleMap.addMarker(markerOptions);
}

@Override
public void onRequestPermissionsResult(int requestCode, @NonNull
String[] permissions, @NonNull int[] grantResults) {
    switch (requestCode) {
        case REQUEST_CODE:
            if (grantResults.length > 0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
                fetchLocation();
            }
            break;
    }
}
}
}

```

- activity\_main.xml

```

<?xml version="1.0" encoding="utf-8"?>

<androidx.fragment.app.FragmentContainerView
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/myMap"
    android:name="com.google.android.gms.maps.SupportMapFragment"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity" />

```

- AndroidManifest.xml

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    package="com.assignment10_mit3206_22550119">

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

```

```

/>
<uses-permission
    android:name="android.permission.ACCESS_BACKGROUND_LOCATION"
    android:required="true" />
<uses-permission android:name="android.permission.INTERNET"
    tools:ignore="ProtectedPermissions" />

<uses-feature android:name="android.hardware.touchscreen" />

<application
    android:allowBackup="true"
    android:dataExtractionRules="@xml/data_extraction_rules"
    android:fullBackupContent="@xml/backup_rules"
    android:icon="@mipmap/ic_launcher"
    android:label="Assignment10-MIT3206-22550119"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/Theme.Assignment10MIT320622550119"
    tools:targetApi="34">

    <meta-data
        android:name="com.google.android.geo.API_KEY"
        android:value="AIzaSyAl39FoH1tCOomeJt568kz2vpgzZdTnty" />

    <activity
        android:name=".MainActivity"
        android:exported="true">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />

            <category android:name="android.intent.category.LAUNCHER"
/>
        </intent-filter>
    </activity>
</application>

</manifest>

```

- **build.gradle (:app)**

```

plugins {
    alias(libs.plugins.android.application)
}

android {
    namespace 'com.assignment10_mit3206_22550119'
    compileSdk 34

    defaultConfig {
        applicationId "com.assignment10_mit3206_22550119"
        minSdk 24
        targetSdk 34
        versionCode 1
        versionName "1.0"

        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
    }

    buildTypes {
        release {

```



```
        minifyEnabled false
        proguardFiles getDefaultProguardFile('proguard-android-
optimize.txt'), 'proguard-rules.pro'
    }
}
compileOptions {
    sourceCompatibility JavaVersion.VERSION_1_8
    targetCompatibility JavaVersion.VERSION_1_8
}
dependencies {
    implementation libs.appcompat
    implementation libs.material
    implementation libs.activity
    implementation libs.constraintlayout
    implementation libs.play.services.maps
    implementation libs.play.services.location
    testImplementation libs.junit
    androidTestImplementation libs.ext.junit
    androidTestImplementation libs.espresso.core
}
```

- **gradle.properties**

```
# Project-wide Gradle settings.
# IDE (e.g. Android Studio) users:
# Gradle settings configured through the IDE *will override*
# any settings specified in this file.
# For more details on how to configure your build environment visit
# http://www.gradle.org/docs/current/userguide/build_environment.html
# Specifies the JVM arguments used for the daemon process.
# The setting is particularly useful for tweaking memory settings.
org.gradle.jvmargs=-Xmx2048m -Dfile.encoding=UTF-8
# When configured, Gradle will run in incubating parallel mode.
# This option should only be used with decoupled projects. For more
details, visit
# https://developer.android.com/r/tools/gradle-multi-project-decoupled-
projects
# org.gradle.parallel=true
# AndroidX package structure to make it clearer which packages are bundled
with the
# Android operating system, and which are packaged with your app's APK
# https://developer.android.com/topic/libraries/support-library/androidx-rn
android.useAndroidX=true
# Enables namespacing of each library's R class so that its R class
includes only the
# resources declared in the library itself and none from the library's
dependencies,
# thereby reducing the size of the R class for that library
android.nonTransitiveRClass=true
android.enableJetifier=true
```