Modern Education Society's College of Engineering, Pune-01.

SUBJECT: ANDROID DEVELOPMENT	
NAME:	
CLASS:	ROLL NO:
SEMESTER/YEAR:-	EXAM NO:
DATE OF PERFORMANCE:	DATE OF SUBMISSION:
EXAMINED:-	REMARKS:

EXPERIMENT NO -15

TITLE: **Design a mobile application using Google maps and GPS to trace the location.**

OBJECTIVE:

1. To study and design a mobile application using Google maps and GPS to trace the location .

PREREQUISITE:

Students should be aware of Android Studio platform

TOOLS USED:

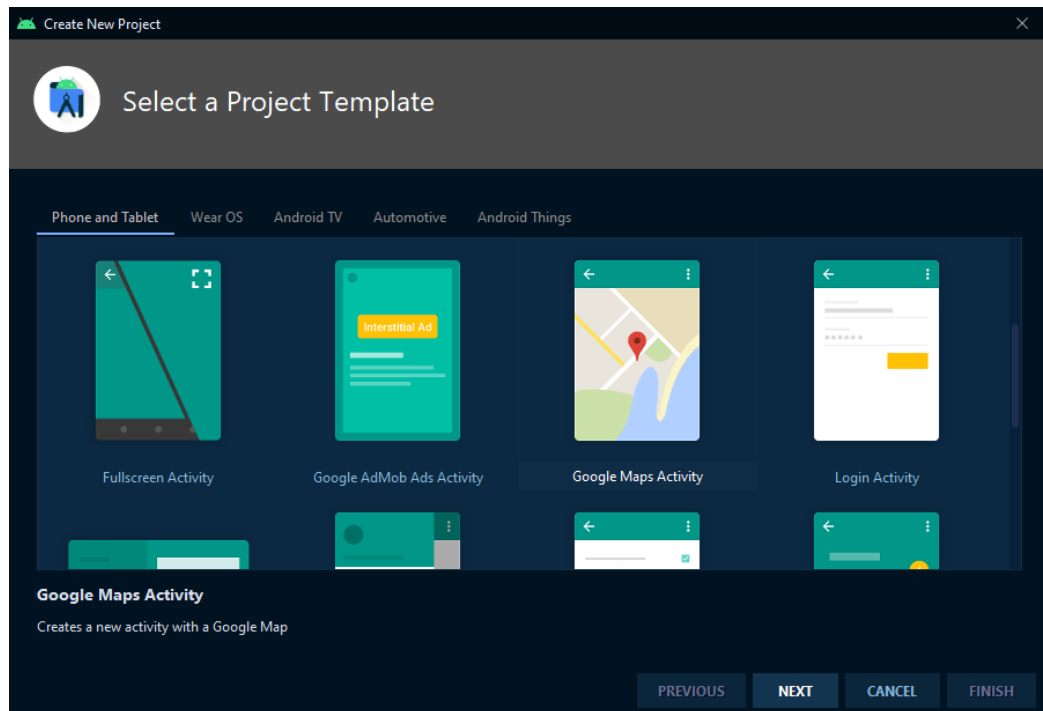1.   Android studio (Electric Eel)

THEORY:

 In this tutorial, you will learn how to create an application that uses Google Maps.

Location services have recently become common among social apps that we use from day to day. Furthermore, these services can be used to track a lost or stolen device. Therefore, knowing how to implement location services can allow you to create more productive applications.

Create a new project:

Create a new project on Android Studio. On the select Project template page, choose the Google Maps Activity template.

A few things to note

Once you open the AndroidManifest.xml, you will see these auto-filled details:

- ACCESS_FINE_LOCATION permission. This accesses the user's precise location. Mostly used when you require the most accurate location. Another type of location accuracy is the ACCESS_COARSE_LOCATION, which is less accurate. Therefore, we shall be working with the ACCESS_FINE_LOCATION for accurate readings.

Conclusion:

Code

1. MainActivity.java

```java
package com.myapp.maping;

import android.content.ActivityNotFoundException;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

// creating variables for
// edit texts and button.
EditText sourceEdt, destinationEdt;
Button trackBtn;

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

// initializing our edit text and buttons
sourceEdt = findViewById(R.id.EdtSource);
destinationEdt = findViewById(R.id.Destination);
trackBtn = findViewById(R.id.BtnTrack);

// adding on click listener to our button.
trackBtn.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
// calling a method to draw a track on google maps.
drawTrack(sourceEdt.getText().toString(), destinationEdt.getText().toString());
}
});
}

private void drawTrack(String source, String destination) {
try {
// create a uri
Uri uri = Uri.parse("https://www.google.co.in/maps/dir/" + source + "/" + destination);

// initializing a intent with action view.
Intent i = new Intent(Intent.ACTION_VIEW, uri);
```

```
// below line is to set maps package name
i.setPackage("com.google.android.apps.maps");

// below line is to set flags
i.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);

// start activity
startActivity(i);
} catch (ActivityNotFoundException e) {
// when the google maps is not installed on users device
// we will redirect our user to google play to download google maps.
Uri uri =
Uri.parse("https://play.google.com/store/apps/details?id=com.google.android.apps.maps");

// initializing intent with action view.
Intent i = new Intent(Intent.ACTION_VIEW, uri);

// set flags
i.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);

// to start activity
startActivity(i);
}
}
}
```

2. Activity_main.xml

```
Xml Code
<?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"
tools:context="com.myapp.maping.MainActivity">

<!--edit text for our source location-->
<EditText
android:id="@+id/EdtSource"
android:layout_width="367dp"
android:layout_height="wrap_content"
android:layout_margin="20dp"
android:hint="Source"
android:padding="8dp" />

<!--edit text for our destination location-->
<EditText
android:id="@+id/Destination"
android:layout_width="match_parent"
```

```
android:layout_height="wrap_content"
android:layout_margin="20dp"
android:hint="Destination"
android:padding="8dp" />

<!--button to draw route on maps-->
<Button
android:id="@+id/BtnTrack"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_margin="20dp"
android:background="@color/purple_500"
android:padding="8dp"
android:text="Show Track"
android:textAllCaps="false"
android:textColor="@color/white" />

</LinearLayout>
```

Output