**Objective**

In this lab, we will extend our previous work on GPS location tracking. We will add a button to our app that, when clicked, will open Google Maps to display the current location and find nearby restuarants.

**Steps**

1. **Import the android content Intent and uri.**

import android.content.Intent;  
import android.net.Uri;

**1. Update the Layout**

Open the activity\_main.xml file and add a new Button widget. Give it an ID, such as btn\_searchNearbyRestaurant.

<Button  
 android:id="@+id/btn\_searchNearbyRestaurant"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="172dp"  
 android:text="Search Nearby Restaurant"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"/>

**2. Set up the Button Click Listener**

In your MainActivity.java file, set up a click listener for your button. When the button is clicked, it should create an Intent to open Google Maps.

btn\_searchNearbyRestaurant = findViewById(R.id.*btn\_searchNearbyRestaurant*);

**3. Create the Google Maps Intent**

Inside the setOnClickListener method, create an Intent that opens Google Maps at the current location. You’ll need to get the current latitude and longitude from your GPS location tracking code.

btn\_searchNearbyRestaurant.setOnClickListener(v -> {  
 String latitude = **XXXXXXXXXXXXXXXX (To be completed)**

String longitude = **XXXXXXXXXXXXXXXX (To be completed)**  
  
 Intent it = new Intent(Intent.*ACTION\_VIEW*);  
 *//set zoom level and query for restaurants* it.setData(Uri.*parse*("geo:" + latitude + "," + longitude + "?z=18&q=restaurants"));  
 startActivity(it);  
});

**4. Test Your App**

Run your app on an Android device. Click the button and verify that Google Maps opens at the correct location.

A screenshot of a phone

Description automatically generated