AMRITA VISHWA VIDYAPEETHAM

20CYS404

ANDROID PROGRAMMING LAB

EXPERIMENT 6

GPS TRACKING APP

DONE BY

DINAKAR S

CH.EN.U4CYS21013

GPS Tracking Android Application

Objective: The objective of this experiment is to develop a GPS tracking Android application that records the user's start and stop locations and calculates the distance traveled. The app also displays the current latitude and longitude in real time. Additionally, an image logo named "amrit" was added to improve the user interface.

Tools and Technologies Used:

- **Programming Language:** Java
- **Development Environment**: Android Studio
- **Libraries**: Android LocationManager, Android Manifest
- **Testing Devices:** Android device or emulator

Implementation Steps:

Project Setup:

- A new Android project was created in Android Studio with an empty activity named MainActivity.
- The required permissions (`ACCESS_FINE_LOCATION` and `ACCESS_COARSE_LOCATION`) were added to the AndroidManifest.xml to access GPS data.

UI Design:

- The layout of the app was created in `activity_main.xml` using a `RelativeLayout`.
- The UI consists of four `TextView` elements for displaying current GPS location, name, register number, and calculated distance.
- Two buttons, "Start GPS Service" and "Stop GPS Service," were added to manage the tracking process.
- An `ImageView` was included to display the "amrit" logo below the current location information to enhance the visual appeal of the app.

Code Functionality:

- The `MainActivity` contains the logic to manage GPS tracking.
- A `LocationManager` and `LocationListener` are used to request location updates and display the latitude and longitude.
- When the "Start GPS Service" button is pressed, the application captures the start location and listens for further location updates.
- On pressing the "Stop GPS Service" button, the app captures the stop location and calculates the distance between the start and stop points using the `distanceTo()` method.
- A logo named "amrit" is displayed using an `ImageView` to provide branding to the app.

Key Code Elements:

- **Permissions Handling:**
    - The app requests the `ACCESS_FINE_LOCATION` permission at runtime to ensure compliance with Android security policies.

- **Location Tracking:**
  - The `startGPS()` method initializes the GPS tracking and captures the start location.
  - The `stopGPS()` method stops the GPS tracking and calculates the distance traveled.
- **Distance Calculation:**
  - The distance between the start and stop locations is calculated using `startLocation.distanceTo(stopLocation)` and displayed in meters.
- **ImageView for Logo:**
  - The logo was added to the UI by placing an `ImageView` below the current location `TextView` in the layout XML file.
  - The image was added to the `res/drawable` directory, and `android:src="@drawable/amrit"` was used to reference it in the XML.

Challenges Faced:

- **Permissions Handling:** Handling runtime permissions required careful implementation to ensure the app doesn't crash if the user denies access.
- **Location Accuracy:** Testing on different devices showed that the accuracy of the GPS signal varies depending on the hardware, leading to potential inconsistencies in distance calculation.

Codes:

MainActivity.java

```
package com.example.gps;



import android.Manifest;

import android.content.Intent;

import android.content.pm.PackageManager;

import android.location.Location;

import android.location.LocationListener;

import android.location.LocationManager;

import android.os.Bundle;

import android.provider.Settings;

import android.view.View;

import android.widget.Button;

import android.widget.TextView;
```

```
import android.widget.Toast;


import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;


public class MainActivity extends AppCompatActivity {
    TextView t1, distanceView, nameView, registerNumberView;
    Button startButton, stopButton;
    LocationManager locationManager;
    LocationListener locationListener;


    Location startLocation, stopLocation;


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


        t1 = findViewById(R.id.t1);
        distanceView = findViewById(R.id.distanceView);
        nameView = findViewById(R.id.nameView);
        registerNumberView = findViewById(R.id.registerNumberView);
        startButton = findViewById(R.id.startButton);
        stopButton = findViewById(R.id.stopButton);


        // Set name and register number
```

```
        nameView.setText("Name: Dinakar S");
        registerNumberView.setText("Register Number: 21013");

        locationManager = (LocationManager) getSystemService(LOCATION_SERVICE);

        locationListener = new LocationListener() {
            @Override
            public void onLocationChanged(@NonNull Location location) {
                t1.setText("Latitude: " + location.getLatitude() + ", Longitude: " + location.getLongitude());
            }

            @Override
            public void onProviderDisabled(@NonNull String provider) {
                Toast.makeText(MainActivity.this, "Please enable GPS", Toast.LENGTH_SHORT).show();
                Intent intent = new Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS);
                startActivity(intent);
            }
        };

        startButton.setOnClickListener(v -> startGPS());
        stopButton.setOnClickListener(v -> stopGPS());
    }

    private void startGPS() {
        if (ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
            ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.ACCESS_FINE_LOCATION}, 1);
            return;
```

```
    }
    locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER, 2000, 5, locationListener);

    startLocation = locationManager.getLastKnownLocation(LocationManager.GPS_PROVIDER);
    if (startLocation != null) {
        Toast.makeText(this, "Start Location Captured", Toast.LENGTH_SHORT).show();
    }
}

private void stopGPS() {
    stopLocation = locationManager.getLastKnownLocation(LocationManager.GPS_PROVIDER);
    if (stopLocation != null) {
        Toast.makeText(this, "Stop Location Captured", Toast.LENGTH_SHORT).show();
        double distance = calculateDistance(startLocation, stopLocation);
        distanceView.setText("Distance: " + distance + " meters");
    }
    locationManager.removeUpdates(locationListener);
}

private double calculateDistance(Location start, Location stop) {
    if (start != null && stop != null) {
        return start.distanceTo(stop);
    }
    return 0.0;
}

@Override
```

```
    public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {

        if (requestCode == 1 && grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION_GRANTED) {

            startGPS();

        }

    }

}
```

Activity_main.xml

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

<RelativeLayout 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"

    tools:context=".MainActivity">



    <TextView

        android:id="@+id/t1"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:text="Current Location"

        android:layout_marginTop="50dp"

        android:layout_centerHorizontal="true" />



    <!-- Name TextView -->

    <TextView

        android:id="@+id/nameView"

        android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
    android:text="Name: "
    android:layout_below="@id/t1"
    android:layout_marginTop="20dp"
    android:layout_centerHorizontal="true" />

<!-- Register Number TextView -->
<TextView
    android:id="@+id/registerNumberView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Register Number: "
    android:layout_below="@id/nameView"
    android:layout_marginTop="10dp"
    android:layout_centerHorizontal="true" />

<ImageView
    android:id="@+id/logoImageView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@drawable/amrit"
    android:layout_marginTop="20dp"
    android:layout_below="@id/registerNumberView"
    android:layout_centerHorizontal="true" />

<Button
    android:id="@+id/startButton"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Start GPS Service"
        android:layout_below="@id/logoImageView"
        android:layout_marginTop="20dp"
        android:layout_centerHorizontal="true" />

    <Button
        android:id="@+id/stopButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Stop GPS Service"
        android:layout_below="@id/startButton"
        android:layout_marginTop="20dp"
        android:layout_centerHorizontal="true" />

    <TextView
        android:id="@+id/distanceView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Distance: "
        android:layout_below="@id/stopButton"
        android:layout_marginTop="20dp"
        android:layout_centerHorizontal="true" />
</RelativeLayout>
```

Android Manifest.xml

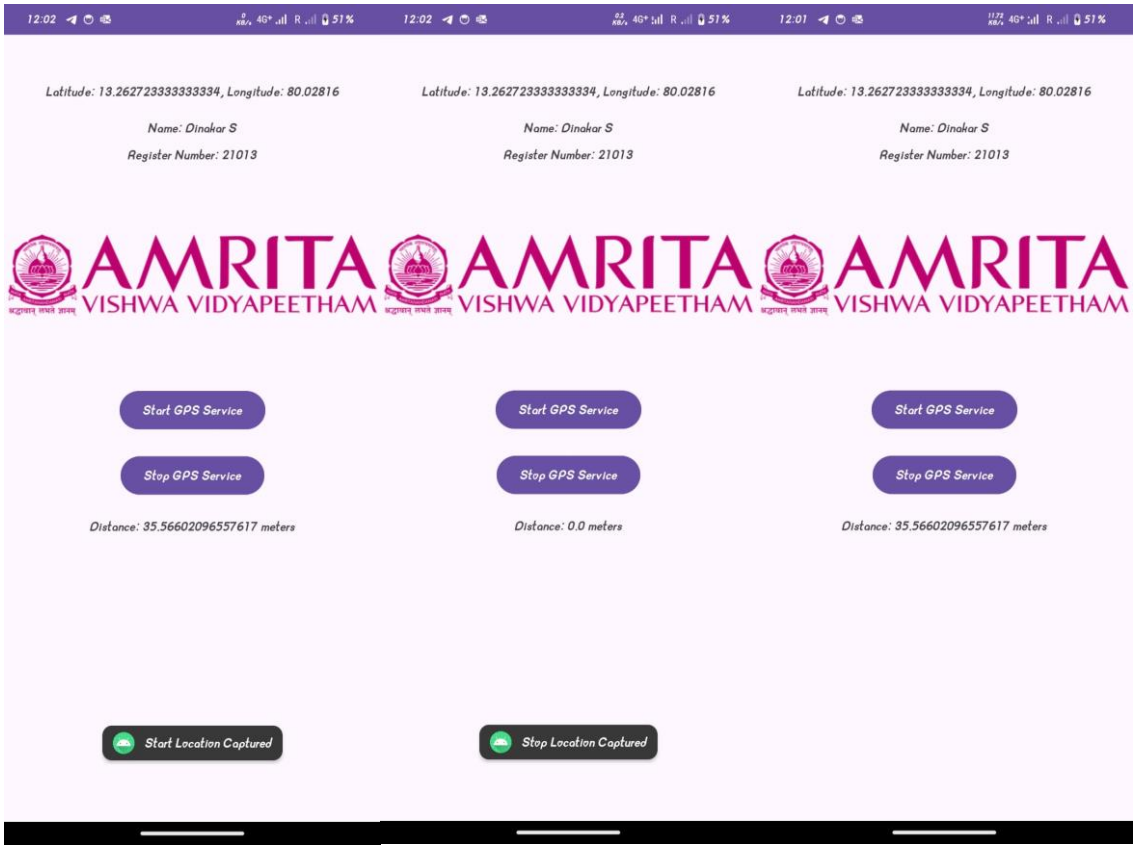
```
<?xml version="1.0" encoding="utf-8"?>
```

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">

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

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Gps"
        tools:targetApi="31">
        <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>
```

Screenshots



Conclusion: This experiment successfully implemented a GPS tracking application capable of capturing start and stop locations, displaying real-time GPS coordinates, and calculating the distance traveled. The "amrit" logo was added to enhance the visual appeal of the app. The experiment helped to understand the implementation of Android's `LocationManager`, handling permissions, and calculating distances using the `Location` class.