**PROJECT REPORT**

**On**

**Android Location Based Service**

**(CSE IV Semester Mini project PCS-404)**

**2020-2021**

****

**Submitted to: Submitted by:**

Mr. Dilip Kumar Gangwar Mr. Chiranjib Ghosh

(CC-IV sem-H) Roll. No.: 1918308

**Guided by:** CSE-H-IV-Sem

Mr. Avnish Panwar Session:2020-2021

(Resource Person)

**DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY**

**GRAPHIC ERA HILL UNVERSITY, DEHRADUN**

**CODE PART**

1. AndroidManifest.file

<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="com.example.location">  
  
 <uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION" />  
 <uses-permission android:name="android.permission.INTERNET"/>  
 <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.Location">  
 <activity android:name=".MainActivity">  
 <intent-filter>  
 <action android:name="android.intent.action.MAIN" />  
  
 <category android:name="android.intent.category.LAUNCHER" />  
 </intent-filter>  
 </activity>  
 </application>  
  
</manifest>

1. Activity.xml

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout 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="match\_parent"  
 android:orientation="vertical"  
 android:background="@drawable/location"  
 android:padding="20sp"  
 tools:context=".MainActivity">  
  
 <Button  
 android:id="@+id/b1"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="400dp"  
 android:background="@color/pink"  
 android:text="Get Location"/>  
  
  
  
  
</LinearLayout>

1. Colors.xml

<?xml version="1.0" encoding="utf-8"?>  
<resources xmlns:tools="http://schemas.android.com/tools">  
 <color name="purple\_200">#FFBB86FC</color>  
 <color name="purple\_500">#FF6200EE</color>  
 <color name="purple\_700">#FF3700B3</color>  
 <color name="teal\_200">#FF03DAC5</color>  
 <color name="teal\_700">#FF018786</color>  
 <color name="black">#FF000000</color>  
 <color name="white">#FFFFFFFF</color>  
 <color name="pink">#F67280</color>  
  
</resources>

1. String.xml

<resources>  
 <string name="app\_name">Location</string>  
</resources>

1. MainActivity.java Class

package com.example.location;  
  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.app.ActivityCompat;  
  
import android.Manifest;  
import android.annotation.SuppressLint;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.Toast;  
  
public class MainActivity extends AppCompatActivity {  
 Button btnLocation;  
 private static final int *REQUEST\_CODE\_PERMISSION*=2;  
 String mPermission = Manifest.permission.*ACCESS\_FINE\_LOCATION*;  
  
 GPSTracker gps;  
 @SuppressLint("WrongViewCast")  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 try{  
 if(ActivityCompat.*checkSelfPermission*(this,mPermission)  
 != LatPackageManager.*PERMISSION\_GRANTED*){  
  
 ActivityCompat.*requestPermissions*(this,new String[]  
 {mPermission},*REQUEST\_CODE\_PERMISSION*);  
 }  
 }catch (Exception e){  
 e.printStackTrace();  
 }  
  
 btnLocation = (Button) findViewById(R.id.*b1*);  
  
 btnLocation.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 gps = new GPSTracker(MainActivity.this);  
  
 if(gps.canGetLocation()){  
 double latitude = gps.getLatitude();  
 double longitude = gps.getLongitude();  
  
 Toast.*makeText*(getApplicationContext(),"Your Updated Location is - \nLat - "  
 +latitude+"\nLong: "+longitude,Toast.*LENGTH\_LONG*).show();  
 }else{  
 gps.showSettingAlert();  
 }  
 }  
 });  
 }  
}

1. GPSTracker.java Class

package com.example.location;  
  
import android.app.AlertDialog;  
import android.app.Service;  
import android.content.Context;  
import android.content.DialogInterface;  
import android.content.Intent;  
import android.location.Location;  
import android.location.LocationListener;  
import android.location.LocationManager;  
import android.os.Bundle;  
import android.os.IBinder;  
import android.provider.Settings;  
import android.util.Log;  
  
public class GPSTracker extends Service implements LocationListener {  
  
 private final Context mContext;  
  
 boolean isGPSEnabled = false;  
  
 boolean isNetworkEnabled = false;  
  
 boolean canGetLocation = false;  
  
 Location location;  
 double latitude;  
 double longitude;  
  
 private static final long *MIN\_DISTANCE\_CHANGE\_FOR\_UPDATES* = 10;  
  
 private static final long *MIN\_TIME\_BW\_UPDATES* = 1000 \* 60 \* 1;  
  
 protected LocationManager locationManager;  
  
 public GPSTracker(Context context) {  
 this.mContext = context;  
 getLocation();  
 }  
  
 public Location getLocation() {  
 try{locationManager = (LocationManager) mContext.getSystemService(*LOCATION\_SERVICE*);  
  
 isGPSEnabled = locationManager.isProviderEnabled(LocationManager.*GPS\_PROVIDER*);  
 isNetworkEnabled = locationManager.isProviderEnabled(LocationManager.*NETWORK\_PROVIDER*);  
  
 if (!isGPSEnabled && !isNetworkEnabled) {  
  
 } else {  
 this.canGetLocation = true;  
 if (isNetworkEnabled) {  
 locationManager.requestLocationUpdates(LocationManager.*NETWORK\_PROVIDER*,  
 *MIN\_TIME\_BW\_UPDATES*, *MIN\_DISTANCE\_CHANGE\_FOR\_UPDATES*, this);  
  
 Log.*d*("Network", "Network");  
 if (locationManager != null) {  
 location = locationManager.getLastKnownLocation(LocationManager.*NETWORK\_PROVIDER*);  
  
 if (location != null) {  
 latitude = location.getLatitude();  
 longitude = location.getLongitude();  
 }  
 }  
 }  
  
 if (isGPSEnabled) {  
 if (location == null) {  
 locationManager.requestLocationUpdates(locationManager.*GPS\_PROVIDER*,  
 *MIN\_TIME\_BW\_UPDATES*, *MIN\_DISTANCE\_CHANGE\_FOR\_UPDATES*, this);  
  
 Log.*d*("GPS Enabled", "GPS Enabled");  
 if (locationManager != null) {  
 location = locationManager  
 .getLastKnownLocation(LocationManager.*GPS\_PROVIDER*);  
  
 if (location != null) {  
 latitude = location.getLatitude();  
 longitude = location.getLongitude();  
 }  
 }  
  
 }  
 }  
 }  
 }catch(Exception e)  
  
 {  
 e.printStackTrace();  
 }  
 return location;  
}  
  
public void stopGPS() {  
 if (locationManager != null) {  
 locationManager.removeUpdates(GPSTracker.this);  
 }  
}  
  
public double getLatitude() {  
 if (location != null) {  
 latitude = location.getLatitude();  
 }  
 return latitude;  
}  
  
public double getLongitude() {  
 if (location != null) {  
 longitude = location.getLongitude();  
  
 }  
 return longitude;  
}  
  
  
 public boolean canGetLocation() {  
 return this.canGetLocation;  
 }  
 public void showSettingAlert() {  
 AlertDialog.Builder alertmez = new AlertDialog.Builder(mContext);  
  
 alertmez.setTitle("GPS is Setting");  
  
 alertmez.setMessage("GPS is not Working Properly");  
  
 alertmez.setPositiveButton("Settings", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int which) {  
 Intent intent = new Intent(Settings.*ACTION\_LOCATION\_SOURCE\_SETTINGS*);  
 mContext.startActivity(intent);  
 }  
 });  
  
 alertmez.setNegativeButton("Dismissed", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int which) {  
 dialog.cancel();  
 }  
 });  
 alertmez.show();  
 }  
 @Override  
 public void onLocationChanged(Location location){  
  
 }  
 @Override  
 public void onProviderDisabled(String provider){  
  
 }  
 @Override  
 public void onProviderEnabled(String provider){  
  
 }  
 @Override  
 public void onStatusChanged(String provider, int status, Bundle extras){  
  
 }  
 @Override  
 public IBinder onBind(Intent arg0){  
 return null;  
 }  
  
 }