МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Лабораторная работа № 20

По дисциплине: «Разработка программного обеспечения для мобильных платформ»

Тема: «Разработка приложения, демонстрирующего геолокационные возможности.»

Выполнила: Велесевич В. А.

Группа: ПО-10

Проверил: Козинский А. А.

Брест 2025

**Цель**: Разработка приложения, демонстрирующего геолокационные возможности.

**Постановка задачи:**

Разработать приложение, получающее координаты устройства и отслеживающее их изменения.

**MainActivity.java**

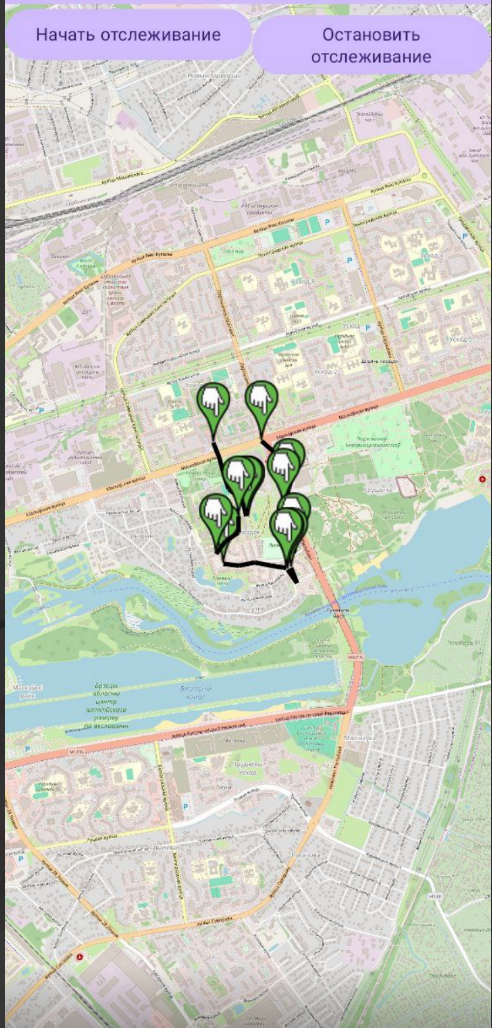
package com.example.lab8;  
import android.Manifest;  
import android.content.BroadcastReceiver;  
import android.content.Context;  
import android.content.Intent;  
import android.content.IntentFilter;  
import android.content.pm.PackageManager;  
import android.os.Build;  
import android.os.Bundle;  
import android.util.Log;  
import android.widget.Button;  
import android.widget.Toast;  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.app.ActivityCompat;  
import androidx.core.content.ContextCompat;  
import com.google.android.gms.location.\*;  
import org.osmdroid.config.Configuration;  
import org.osmdroid.tileprovider.tilesource.TileSourceFactory;  
import org.osmdroid.util.GeoPoint;  
import org.osmdroid.views.MapView;  
import org.osmdroid.views.overlay.Marker;  
import org.osmdroid.views.overlay.Polyline;  
import java.util.ArrayList;  
import java.util.List;  
public class MainActivity extends AppCompatActivity {  
 private static final int REQUEST\_LOCATION = 1;  
 private MapView mapView;  
 private Polyline polyline;  
 private List<GeoPoint> trackPoints = new ArrayList<>();  
 private boolean isFirstLocationUpdate = true;  
 private boolean firstPointSet = false;  
 private LocationDatabase db;  
 private LocationReceiver locationReceiver;  
 private Intent serviceIntent;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 Configuration.getInstance().load(getApplicationContext(),  
 getSharedPreferences("osmdroid", MODE\_PRIVATE));  
 setContentView(R.layout.activity\_main);  
 locationReceiver = new LocationReceiver();  
 IntentFilter filter = new IntentFilter();  
 filter.addAction("com.example.lab8.LOCATION\_UPDATED");  
 if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.O) {  
 registerReceiver(locationReceiver, filter, Context.RECEIVER\_EXPORTED);  
 }  
  
 db = new LocationDatabase(this);  
 mapView = findViewById(R.id.map);  
 mapView.setTileSource(TileSourceFactory.MAPNIK);  
 mapView.getController().setZoom(15);  
 if (!hasLocationPermission()) {  
 requestLocationPermission();  
 } else {  
 }  
  
 Button startButton = findViewById(R.id.startTrackingButton);  
 Button stopButton = findViewById(R.id.stopTrackingButton);  
 startButton.setOnClickListener(v -> startTrackingService());  
 stopButton.setOnClickListener(v -> stopTrackingService());  
  
 loadSavedLocations();  
 }  
  
  
 @Override  
 protected void onDestroy() {  
 super.onDestroy();  
 if (locationReceiver != null) {  
 unregisterReceiver(locationReceiver);  
 }  
 stopTrackingService();  
 }  
 private void startTrackingService() {  
 serviceIntent = new Intent(this, LocationService.class);  
 if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.O) {  
 Toast.makeText(this, "Запуск сервиса", Toast.LENGTH\_SHORT).show();  
 startForegroundService(serviceIntent);  
 } else {  
 startService(serviceIntent);  
 }  
 }  
  
 private void stopTrackingService() {  
 if (serviceIntent != null) {  
  
  
 Marker marker = new Marker(mapView);  
 marker.setPosition(trackPoints.get(trackPoints.size() - 1));  
 marker.setTitle("Конец");  
 mapView.getOverlays().add(marker);  
  
  
 stopService(serviceIntent);  
 serviceIntent = null; // Убираем Intent после остановки сервиса  
 }  
 Log.d("STOPSERVICE", "Остановка сервиса");  
 }  
  
  
 private boolean hasLocationPermission() {  
 return ContextCompat.checkSelfPermission(this, Manifest.permission.ACCESS\_FINE\_LOCATION) == PackageManager.PERMISSION\_GRANTED;  
 }  
  
 private void requestLocationPermission() {  
 ActivityCompat.requestPermissions(this, new String[]{Manifest.permission.ACCESS\_FINE\_LOCATION}, REQUEST\_LOCATION);  
 }  
  
  
 void addLocationToMap(double lat, double lon) {  
 GeoPoint point = new GeoPoint(lat, lon);  
 trackPoints.add(point);  
  
 if (isFirstLocationUpdate) {  
 mapView.getController().setCenter(point);  
 isFirstLocationUpdate = false;  
  
 Marker marker = new Marker(mapView);  
 marker.setPosition(trackPoints.get(0));  
 marker.setTitle("Начало");  
 mapView.getOverlays().add(marker);  
 }  
   
 updatePolyline();   
 mapView.invalidate();  
 }  
  
 private void updatePolyline() {  
 if (trackPoints.size() >= 2) {  
 if (polyline != null) {  
 mapView.getOverlays().remove(polyline);  
 }  
 polyline = new Polyline();  
 polyline.setPoints(trackPoints);  
 mapView.getOverlays().add(polyline);  
 }  
 }  
 private void drawPolyline(List<GeoPoint> points) {  
 if (points.size() < 2) return; // Линия должна содержать хотя бы 2 точки  
  
 Log.d("LOADSAVEDLOCATIONS", "DrawPolyline");  
  
 polyline = new Polyline();  
 polyline.setPoints(points);  
   
 mapView.getOverlayManager().add(polyline); // Добавляем линию на карту  
 mapView.invalidate(); // Обновляем карту  
 }  
  
 private void loadSavedLocations() {  
 List<Track> tracks = db.getAllTracks();  
  
 Log.d("LOADSAVEDLOCATIONS", tracks.toString());  
  
 for (Track track : tracks) {  
 List<GeoPoint> savedPoints = db.getLocations(track.getId());  
  
 Log.d("LOADSAVEDLOCATIONS", String.valueOf(track.getId()));  
 Log.d("LOADSAVEDLOCATIONS", savedPoints.toString());  
  
  
 if (!savedPoints.isEmpty()) {  
 drawPolyline(savedPoints);  
  
 Marker marker = new Marker(mapView);  
 marker.setPosition(savedPoints.get(0));  
 marker.setTitle("Начало " + track.getId());  
 mapView.getOverlays().add(marker);  
  
 marker = new Marker(mapView);  
 marker.setPosition(savedPoints.get(savedPoints.size() - 1));  
 marker.setTitle("Конец " + track.getId());  
 mapView.getOverlays().add(marker);  
  
 if (!firstPointSet) {  
 mapView.getController().setCenter(savedPoints.get(0));  
 firstPointSet = true;  
 }  
 @Override  
 public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {  
 super.onRequestPermissionsResult(requestCode, permissions, grantResults);  
 if (requestCode == REQUEST\_LOCATION) {  
 if (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION\_GRANTED) {  
 //startTrackingService();  
 } else {  
 Toast.makeText(this, "Разрешение не предоставлено.", Toast.LENGTH\_LONG).show();  
 }  
 }  
 }  
}

**LocationDatabase.java**

package com.example.lab8;  
import android.content.BroadcastReceiver;  
import android.content.Context;  
import android.content.Intent;  
import android.util.Log;  
  
public class LocationReceiver extends BroadcastReceiver {  
 @Override  
 public void onReceive(Context context, Intent intent) {  
 double latitude = intent.getDoubleExtra("latitude", 0);  
 double longitude = intent.getDoubleExtra("longitude", 0);  
  
 Log.d("LocationReceiver", "Получены координаты: " + latitude + ", " + longitude);  
 if (context instanceof MainActivity) {  
 ((MainActivity) context).addLocationToMap(latitude, longitude);  
 }  
 }  
}

**Track.java**

package com.example.lab8;  
public class Track {  
 private long id;  
 private String name;  
 public Track(long id, String name) {  
 this.id = id;  
 this.name = name;  
 }  
 public long getId() {  
 return id;  
 }  
 public String getName() {  
 return name;  
 }  
}



**Вывод:** разработала приложение, демонстрирующее геолокационные возможности.