Министерство науки и высшего образования Российской Федерации

Федерального государственного бюджетное образовательное учреждение высшего образования

«Российской экономический университет имени Г. В. Плеханова»

**МОСКОВСКИЙ ПРИБОРОСТРОИТЕЛЬНЫЙ ТЕХНИКУМ.**

ПРАКТИЧЕСКИЕ РАБОТЫ № 11

ДИСЦИПЛИНА: «РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ»

Тема: Работа с датчиками и создание гироскопа.

Специальность: 09.02.07 Информационные системы и программирование

Квалификация: Программист

Листов:

|  |  |  |
| --- | --- | --- |
| Выполнил студент |  | Проверил преподаватель |
| Группа П50-3-18 |  | \_\_\_\_\_\_\_\_\_\_\_А.О.Лясников |
| Пахомов Даниил Александрович |  | «\_\_\_» \_\_\_\_\_\_\_\_2020 года |

Москва 2021

ПРАКТИЧЕСКАЯ РАБОТА №11

Работа с GoogleMaps

Цель: создать программу, которая будет взаимодействовать с картой GoogleMaps и ставить маркеры в зависимости от заданных координат.

1. Создаем проект Google Maps Activity

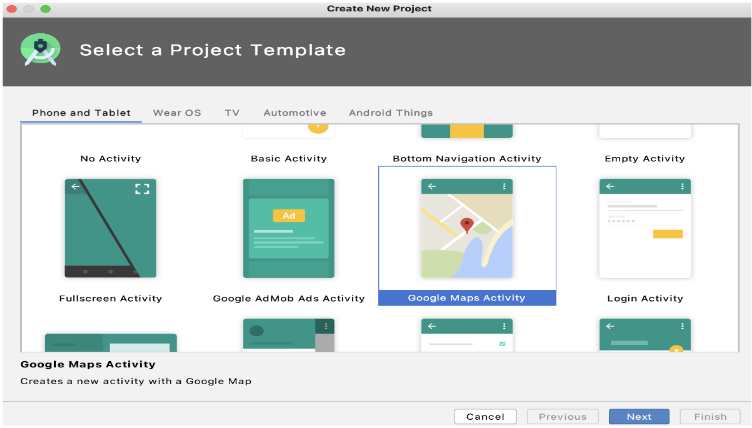
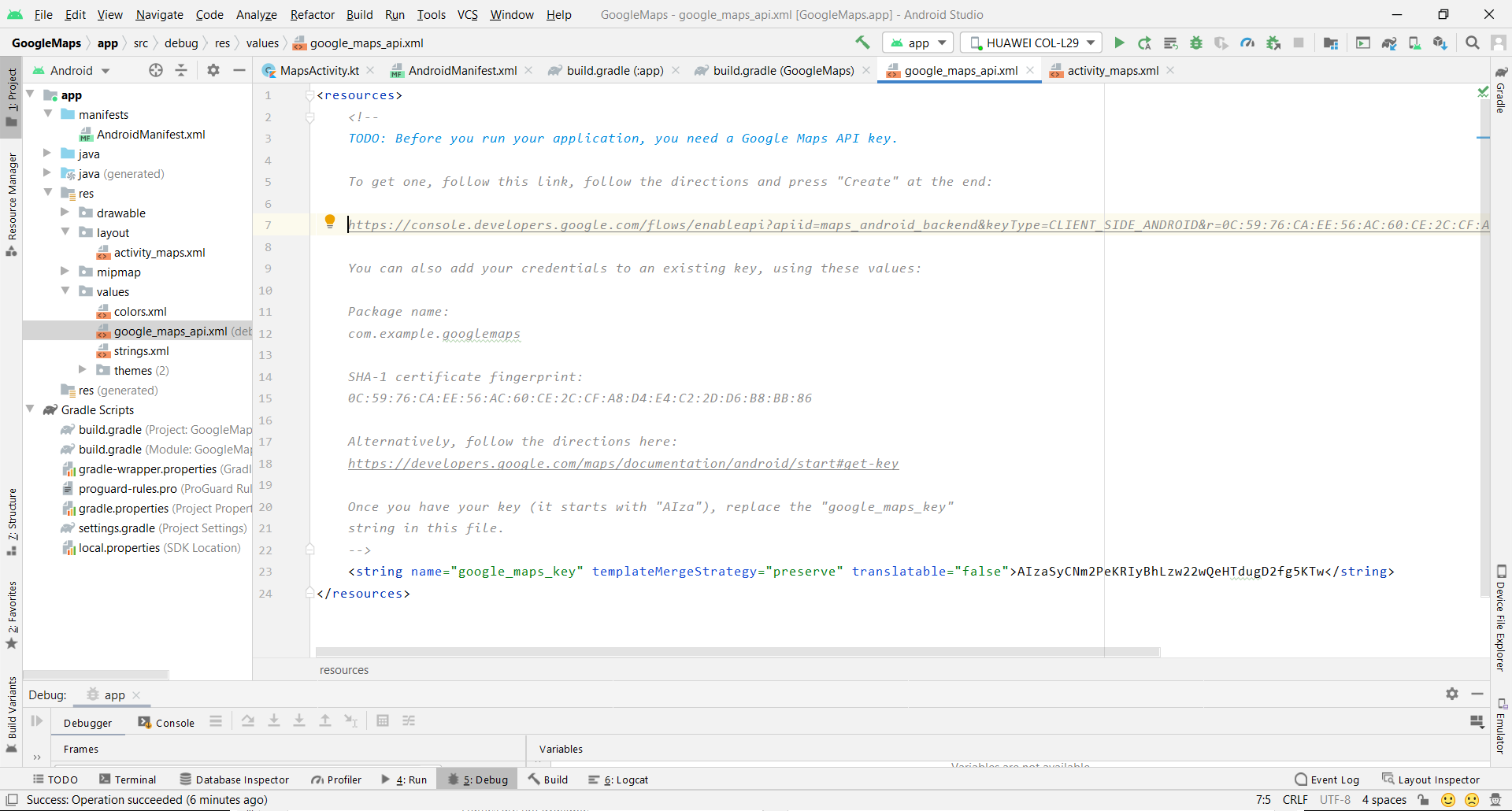


Рисунок - Создание проекта

1. Создаем API\_KEY

Для создания API\_KEY нужно перейти по ссылке которая находиться в файле google\_maps\_api.xml.



1. AndroidManifest

|  |
| --- |
| *<?*xml version="1.0" encoding="utf-8"*?>* <manifest xmlns:android="http://schemas.android.com/apk/res/android"  package="com.example.googlemaps">   *<!--  The ACCESS\_COARSE/FINE\_LOCATION permissions are not required to use  Google Maps Android API v2, but you must specify either coarse or fine  location permissions for the "MyLocation" functionality.  -->* <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.GoogleMaps">   *<!--  The API key for Google Maps-based APIs is defined as a string resource.  (See the file "res/values/google\_maps\_api.xml").  Note that the API key is linked to the encryption key used to sign the APK.  You need a different API key for each encryption key, including the release key that is used to  sign the APK for publishing.  You can define the keys for the debug and release targets in src/debug/ and src/release/.  -->* <meta-data  android:name="com.google.android.geo.API\_KEY"  android:value="@string/google\_maps\_key" />   <activity  android:name=".MapsActivity"  android:label="@string/title\_activity\_maps">  <intent-filter>  <action android:name="android.intent.action.MAIN" />   <category android:name="android.intent.category.LAUNCHER" />  </intent-filter>  </activity>  </application>  </manifest> |

1. MapsActivity

|  |
| --- |
| package com.example.googlemaps  import android.content.Context import android.content.SharedPreferences import android.location.Geocoder import android.os.Bundle import android.os.Parcelable import android.util.Log import android.view.View import android.widget.Toast import androidx.appcompat.app.AppCompatActivity import com.google.android.gms.maps.CameraUpdateFactory import com.google.android.gms.maps.GoogleMap import com.google.android.gms.maps.OnMapReadyCallback import com.google.android.gms.maps.SupportMapFragment import com.google.android.gms.maps.model.LatLng import com.google.android.gms.maps.model.MarkerOptions import kotlinx.android.parcel.Parcelize import kotlinx.android.synthetic.main.activity\_maps.\* import java.util.\* import kotlin.collections.HashSet  class MapsActivity : AppCompatActivity(), OnMapReadyCallback {   private lateinit var mMap: GoogleMap  private lateinit var test: Geocoder  private var latitude: Double = -1.0  private var longitude: Double = -1.0  private lateinit var sp: SharedPreferences  private var latList: MutableList<String> = *mutableListOf*()  private var longList: MutableList<String> = *mutableListOf*()   override fun onCreate(savedInstanceState: Bundle?) {  super.onCreate(savedInstanceState)  setContentView(R.layout.*activity\_maps*)   val mapFragment = *supportFragmentManager* .findFragmentById(R.id.*map*) as SupportMapFragment  mapFragment.getMapAsync(this)   sp = getSharedPreferences("SHARED\_PREF", Context.*MODE\_PRIVATE*)  val sizeLat = sp.getInt("latlist\_size", 0)  val sizeLong = sp.getInt("longlist\_size", 0)  for (i in 0 *until* sizeLat) {  latList.add(sp.getString("latlist\_" + i, null)!!)  longList.add(sp.getString("longlist\_" + i, null)!!)  }  }    override fun onMapReady(googleMap: GoogleMap) {  mMap = googleMap  *// Add a marker in Sydney and move the camera* for (i in 0 *until* latList.size) {  val point = LatLng(latList[i].*toDouble*(), longList[i].*toDouble*())  mMap.addMarker(  MarkerOptions().position(point)  )  }  }   fun Add(view: View) {  test = Geocoder(this)  val thread: Thread = object : Thread() {  override fun run() {  var t = test.getFromLocationName(City.*text*.toString(), 1)  if (!t.isEmpty()) {  latitude = t[0].*latitude* longitude = t[0].*longitude* latList.add(latitude.toString())  longList.add(longitude.toString())  Log.i("tag", t[0].*latitude*.toString() + " " + t[0].*longitude*.toString())  } else {  runOnUiThread **{** Toast.makeText(  this@MapsActivity,  "Данные были введены не корректно",  Toast.*LENGTH\_SHORT* ).show()  **}** }   }  }  thread.start()  Thread.sleep(1000)   val editor = sp.edit()  editor.putInt("latlist\_size", latList.size)  editor.putInt("longlist\_size", longList.size)  for (i in 0 *until* latList.size) {  editor.putString(  "latlist\_" + i,  latList[i]  )  editor.putString("longlist\_" + i, longList[i])  }   editor.apply()  val point = LatLng(  latitude,  longitude  )  mMap.addMarker(MarkerOptions().position(point).title(City.*text*.toString()))  } } |

1. Activity\_maps

|  |
| --- |
| *<?*xml version="1.0" encoding="utf-8"*?>* <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:map="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:orientation="vertical"  android:padding="20dp">   <EditText  android:id="@+id/City"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:hint="Введите город" />    <Button  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Добавить точку"  android:gravity="center"  android:onClick="Add"  android:layout\_gravity="center"/>  <fragment  android:id="@+id/map"  android:name="com.google.android.gms.maps.SupportMapFragment"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".MapsActivity" />     </LinearLayout> |

1. Результат работы программы

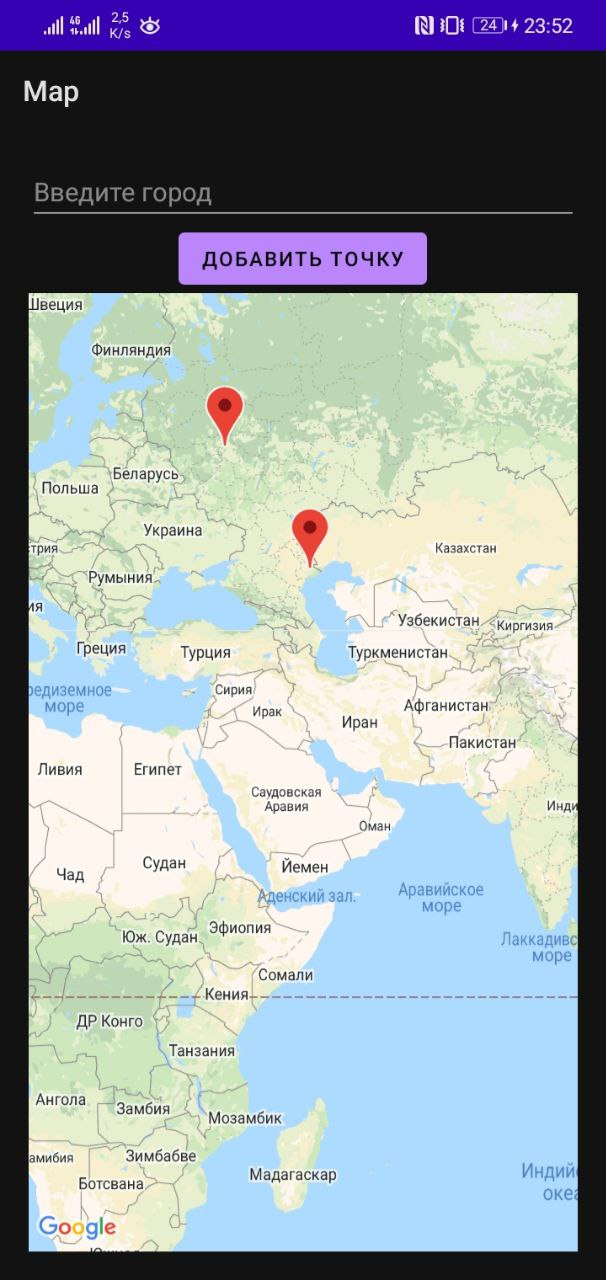


Рисунок - вывод сохраненных маркеров

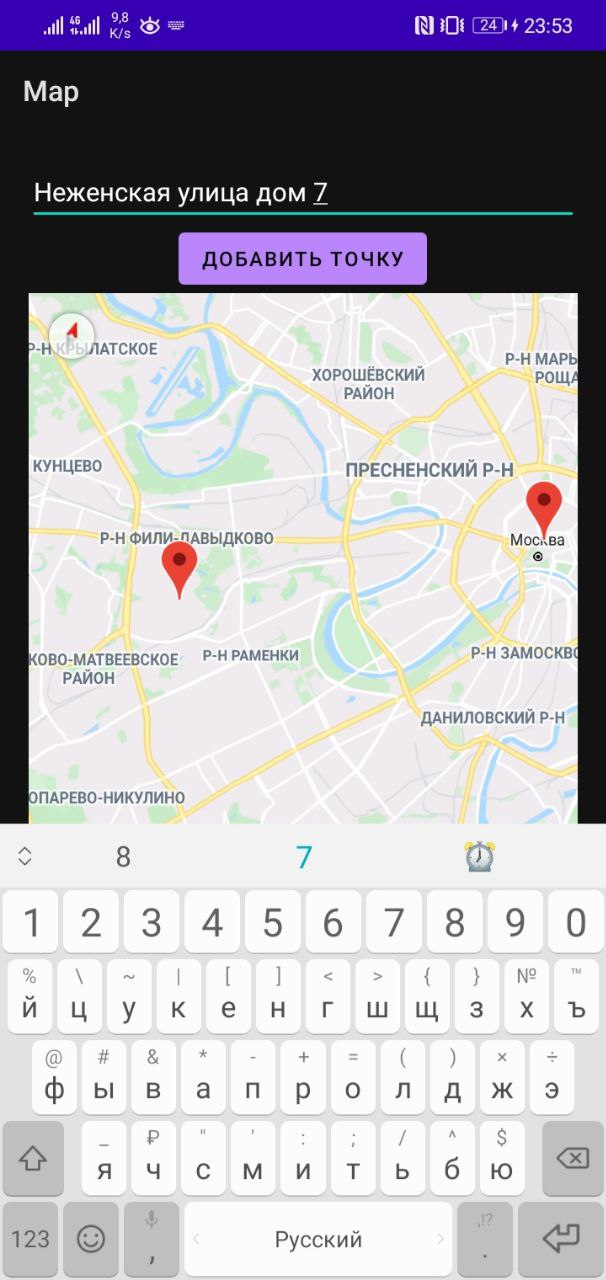


Рисунок - добавление новых маркеров

Вывод: была создана программа, которая будет взаимодействовать с картой GoogleMaps и ставить маркеры в зависимости от заданных координат.