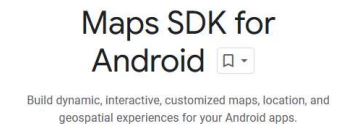
Android
-
mapy

Peter Borovanský
KAI, I-18
MS-Teams: 2sf3ph4, List, github
borovan 'at' ii.fmph.uniba.sk

Google Maps

- **API key, podpisovanie appky**
- **Google Console**
- **MarcelP – OSM, Mapbox**



Google Maps

Na prácu s balíkom `com.google.android.maps.*` potrebujeme API kľúč pre Google Maps vygenerovaný pomocou **SHA1** kľúča – čo je váš *finger-print*.
(presnejšie finger-print vašej inštalácie na vašom počítači)

Google Maps verzia **V2** (pár rokov už jediná alternatíva Gmaps Verzia 2):
https://developers.google.com/maps/documentation/android/

Google Maps API-key sa generuje pomocou **SHA1** kľúča, ktorý je vygenerovaný v súbore .keystore pri inštalácii Android Studio (po reinštale počítača idete s novým finger-printom!)

Dôsledok1 ☹
Ak si rebuildujete niektorý zo zverejených kódov (mojich, či cudzích), nepôjde vám skôr:

- 1) ako si aplikáciu nezaregistrujete v **Google Console** (vaším SHA1-kľúčom+package)
- 2) vygenerovaným **API-key** z **Google Console** nepodpíšete váš build.

Dôsledok2 ☺
ak nerebuildujete project s vašim SHA1, tak .apk je ok, a pôjde vám nainštalovať aj spustiť

Získanie SHA1

existuje viacero spôsobov

```
C:\Users\borovan>dir c:\users\borovan\.android\*.keystore
 Volume in drive C is System
 Volume Serial Number is 0006-3F8C

 Directory of c:\users\borovan\.android

12/31/2018  01:47 PM             1,259 debug.keystore
               1 File(s)          1,259 bytes
               0 Dir(s)  73,137,201,152 bytes free
```

Potrebujeme získať náš SHA1 kľúč, v cmdline spustíme „správny" keytool.exe:

c:\Program Files\Android\Android Studio\jbr\bin\keytool.exe

```
keytool -v -list -alias androiddebugkey
    -keystore "c:\Users\<user>\.android\debug.keystore"
    -storepass android
    -keypass android
```

```
[4 files and 0 directories found]
c:\Program Files\Android\Android Studio\jbr\bin\keytool.exe
c:\Program Files\JetBrains\GoLand 2024.1.1\jbr\bin\keytool.exe
c:\Program Files\JetBrains\IntelliJ IDEA 2024.1.1\jbr\bin\keytool.exe
c:\Program Files\JetBrains\PyCharm 2024.2.2\jbr\bin\keytool.exe
```

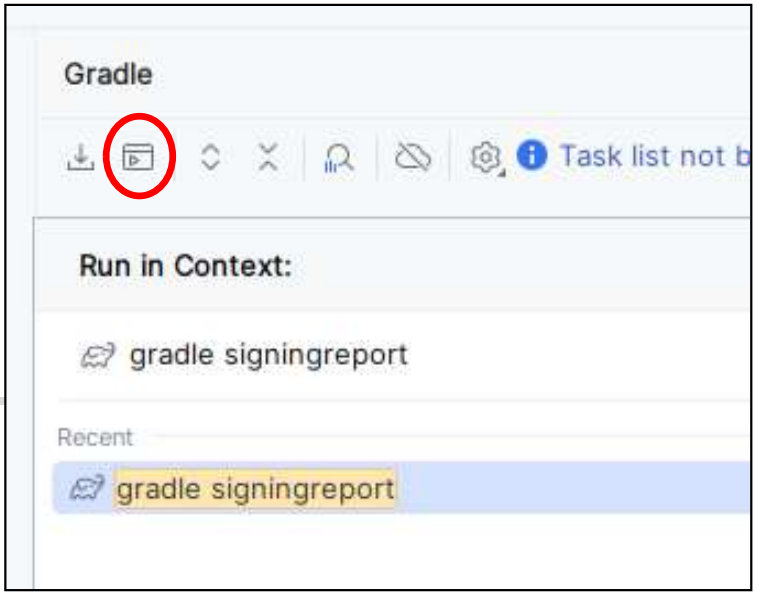
Ten sa získa (pre účely ladenia) z debug.keystore file

- Windows: `c:\Users\<user>\.android\debug.keystore`
- Linux: /home/*user*/.android/

```
C:\Windows\System32\cmd.exe

d:\borovan\VMA\VMA\Prednasky\09>"c:\Program Files\Android\Android Studio\jbr\bin\keytool.exe" -v -list -alias androiddebugkey -keystore "c:\Users\boro
van\.android\debug.keystore" -storepass android -keypass android
Alias name: androiddebugkey
Creation date: 2. 7. 2024
Entry type: PrivateKeyEntry
Certificate chain length: 1
Certificate[1]:
Owner: C=US, O=Android, CN=Android Debug
Issuer: C=US, O=Android, CN=Android Debug
Serial number: 1
Valid from: Tue Jul 02 15:15:27 CEST 2024 until: Thu Jun 25 15:15:27 CEST 2054
Certificate fingerprints:
         SHA1: 3A:CF:42:C[illegible]:02:E1:82
         SHA256: 1B:F0:1C:E7:58:CC:1D:CE:16:7B:8B:C9:2B:0F:2D:41:38:1A:9B:12:E2:CA:C0:0D:A2:0D:96:32:35:72:CD:FF
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 2048-bit RSA key
Version: 1

d:\borovan\VMA\VMA\Prednasky\09>
```

Získanie SHA1

(v Android Studio)



- Gradle
  - Execute Gradle Task: gradle signingreport
- Run Project[SigningReport]
- už mám SHA-1 a čo s ním ? ...bude ho treba v Google Cloud Console

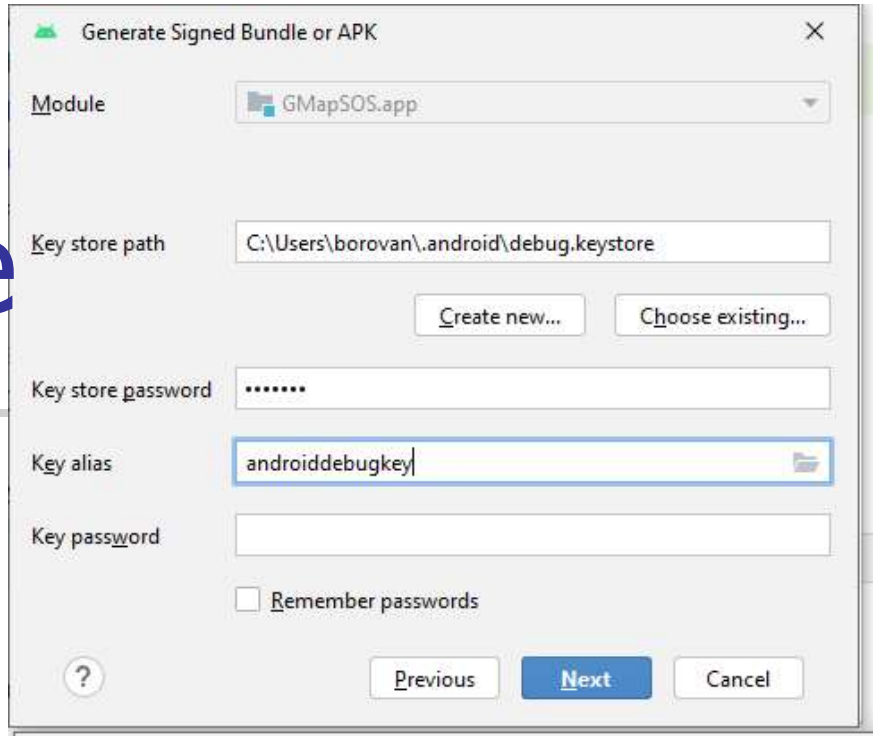
```
Store: C:\Users\borovan\.android\debug.keystore
Alias: AndroidDebugKey
MD5: 43:3E:04:B4:9E:F5:39:01:54:31:60:EF:56:1E:C0:EC
SHA1: 3A:CF:42 [redacted] :02:E1:82
SHA-256: 1B:F0:1C:E7:58:CC:1D:CE:16:7B:8B:C9:2B:0F:2D:41:38:1A:9B:12:E2:CA:C0:0D:A2:0D:96:32:35:72:CD:FF
Valid until: Thursday, June 25, 2054
```

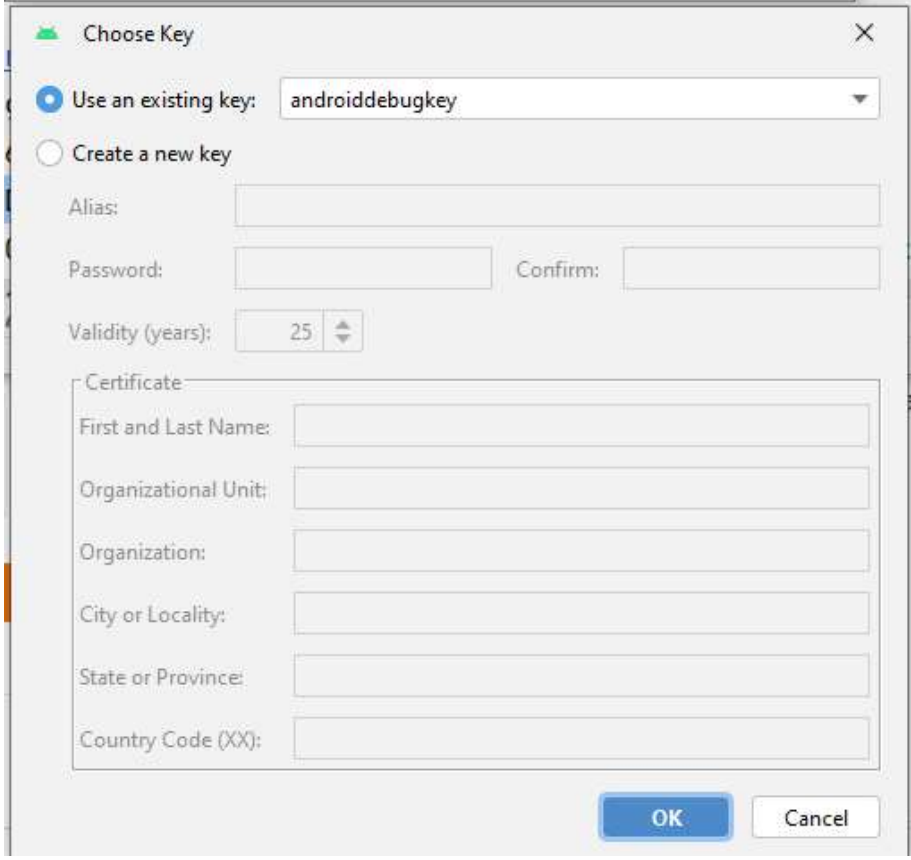
Debug keystore

(manažment certifikátov)

```
$ keytool -genkey -v
  -keystore debug.keystore
  -storepass android
  -alias androiddebugkey
  -keypass android
  -keyalg RSA -keysize 2048
  -validity 10000
```

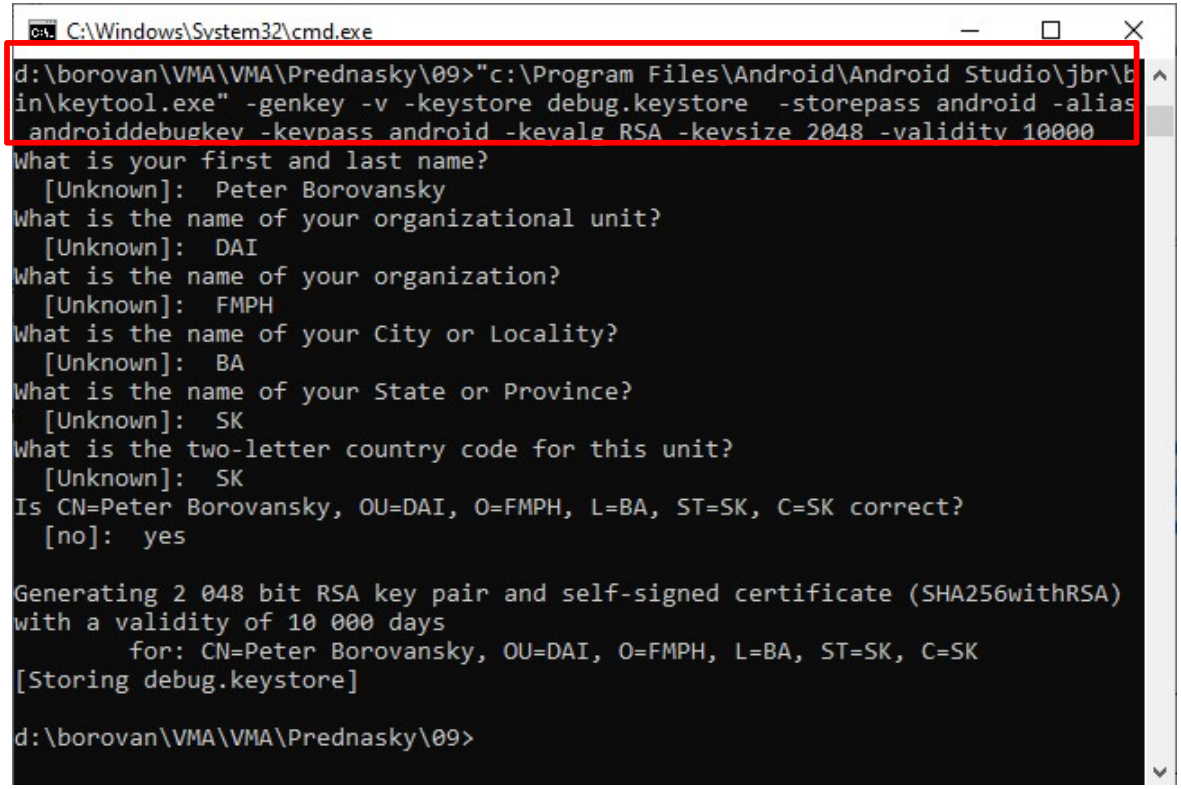
Keystore name: "debug.keystore"
Keystore password: "android"
Key alias: "androiddebugkey"
Key password: "android"

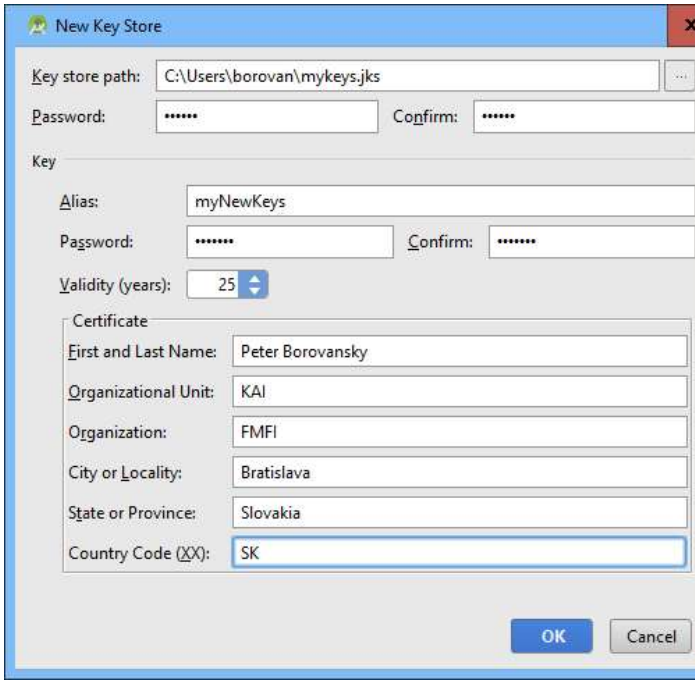




Debug keystore
(manažment certifikátov)

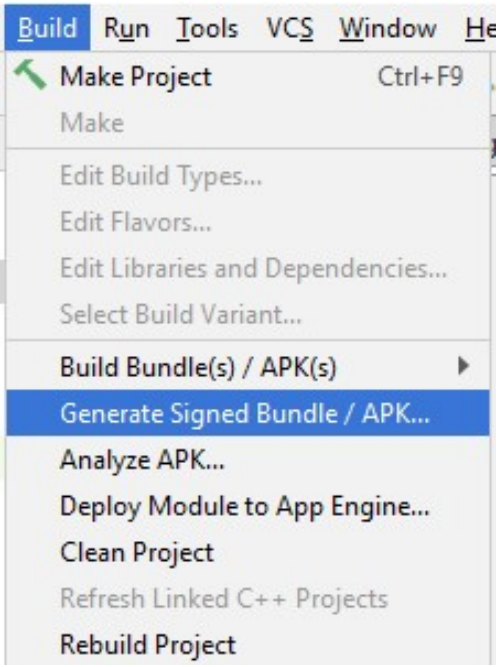
- **debug.keystore** obsahuje jeden alebo viac privátnych kľúčov (certifikátov).
- **debug.keystore** nám automaticky vyrobí Android-Studio pri inštalácii AS
- Android Studio nám automaticky podpíše každú apku kľúčom debug.keystore
- vieme vygenerovať vlastný **keystore**/certifikát, pomocou `keytool`,resp.v AS
- v AS Build/Generate Signed APK https://developer.android.com/studio/publish/app-signing.html

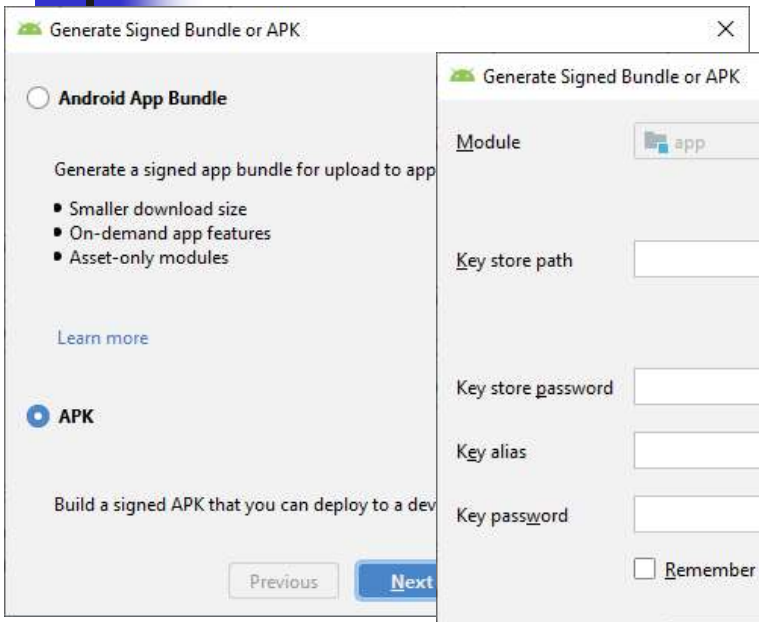


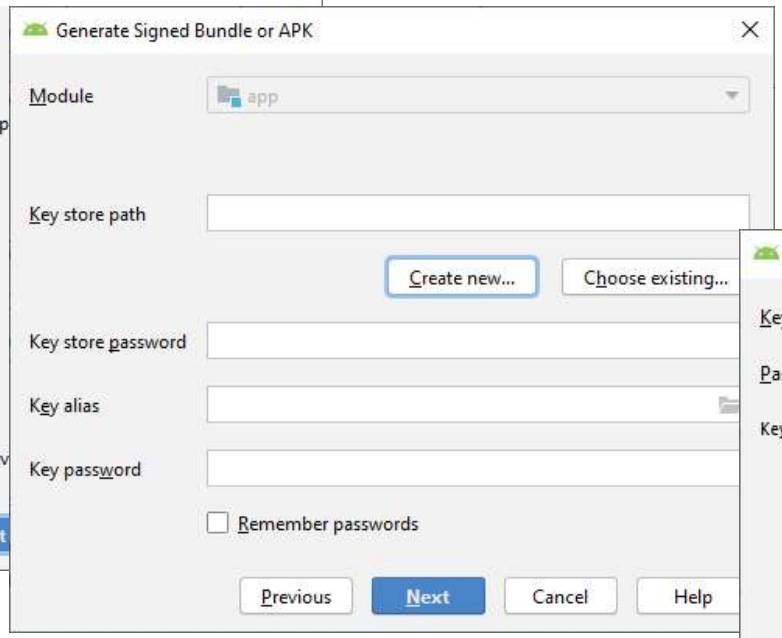


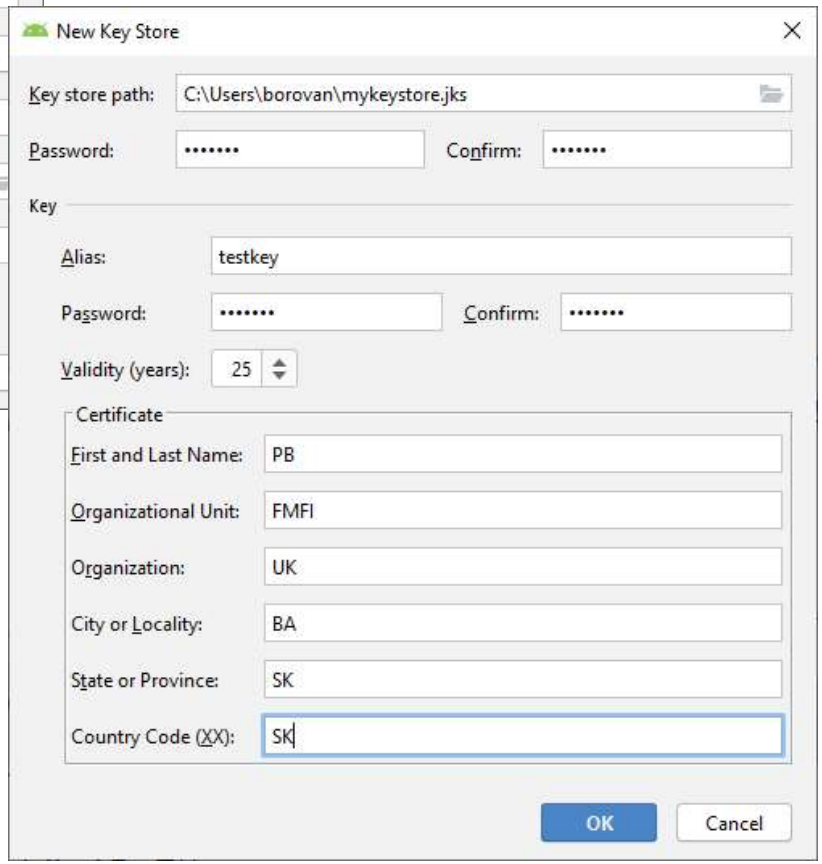
Generovanie kľúča

začnite v Build/Generate Signed APK

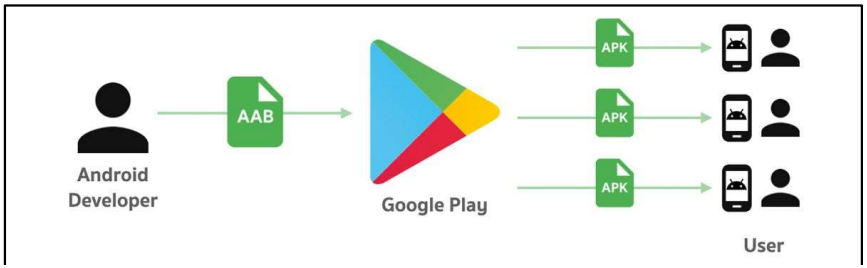








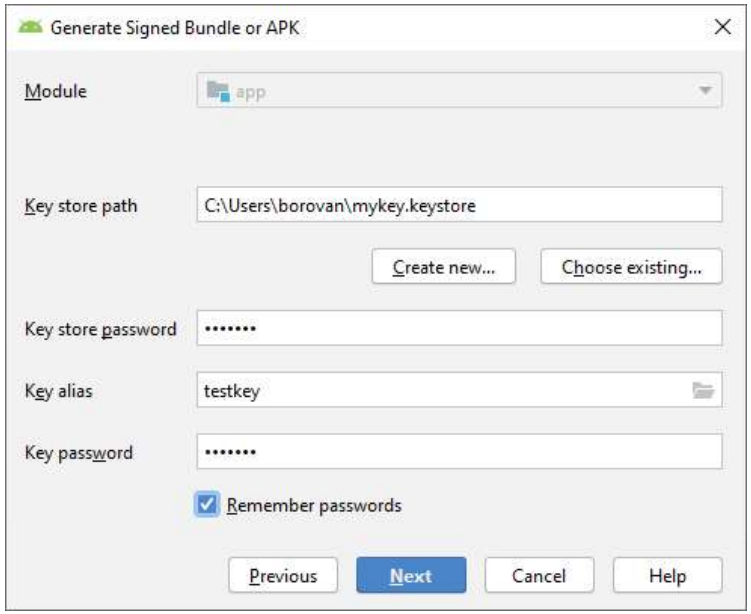
App bundles are publishing format, whereas **APK (Android application Package)** is the packaging format which eventually will be installed on device. Google uses app bundle to generate and serve optimized APKs for each user's device configuration, so they download only the code and resources they need to run your app. Therefore, users can get smaller and more optimized downloads.

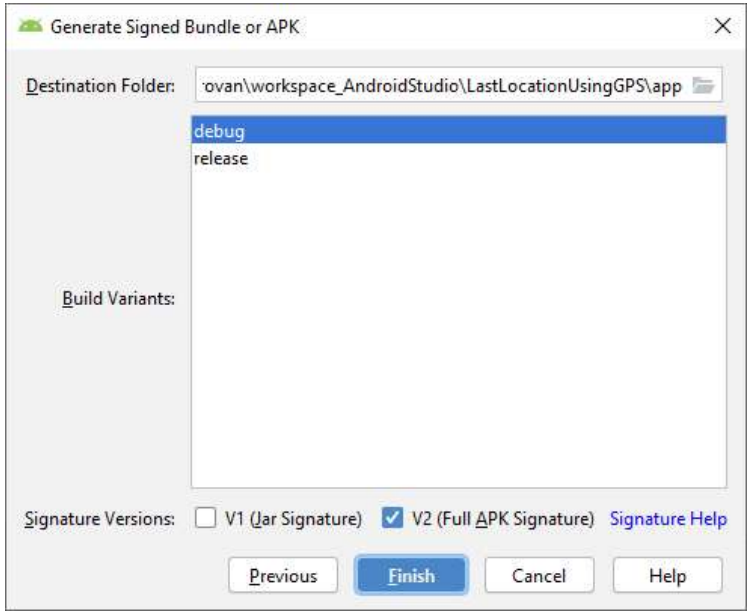


Podpisovanie aplikácie (debug)

(použitie certifikátov)

Vygenerovaný kľúč potom slúži na podpísanie .apk





AS automaticky podpisuje .apk pri builde kľúčom `androiddebugkey`

https://developer.android.com/studio/publish/app-signing.html#release-mode

Podpisovanie aplikácie (release)

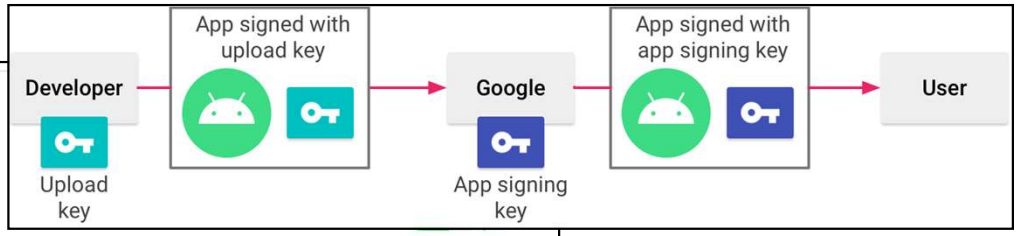
(použitie certifikátov)

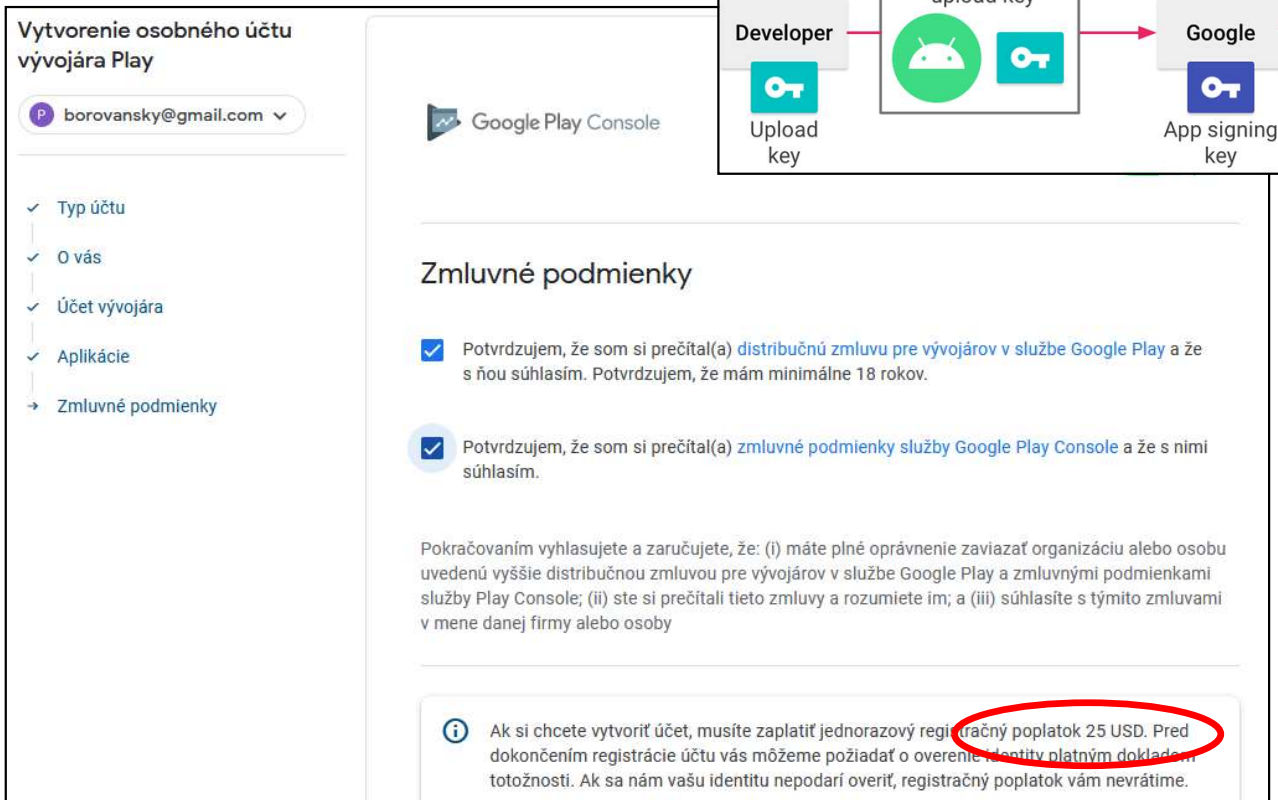
- Potrebujete Upload key a App signing key:

https://developer.android.com/studio/publish/app-signing

- Potrebujete Google Play Account, Google Play Console:

https://play.google.com/apps/publish/signup/

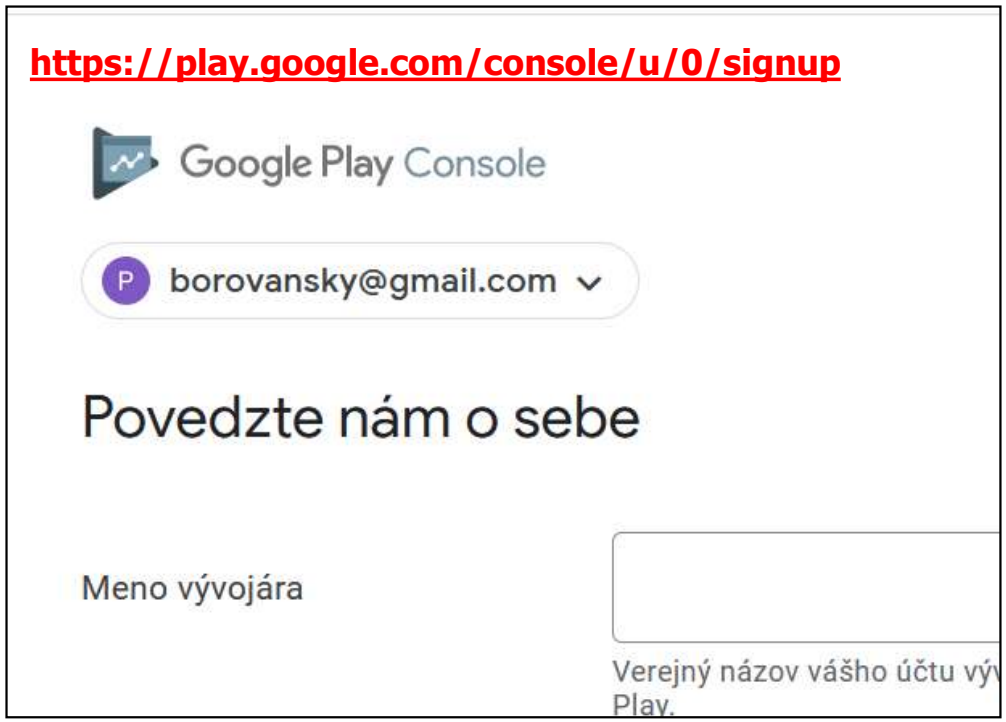




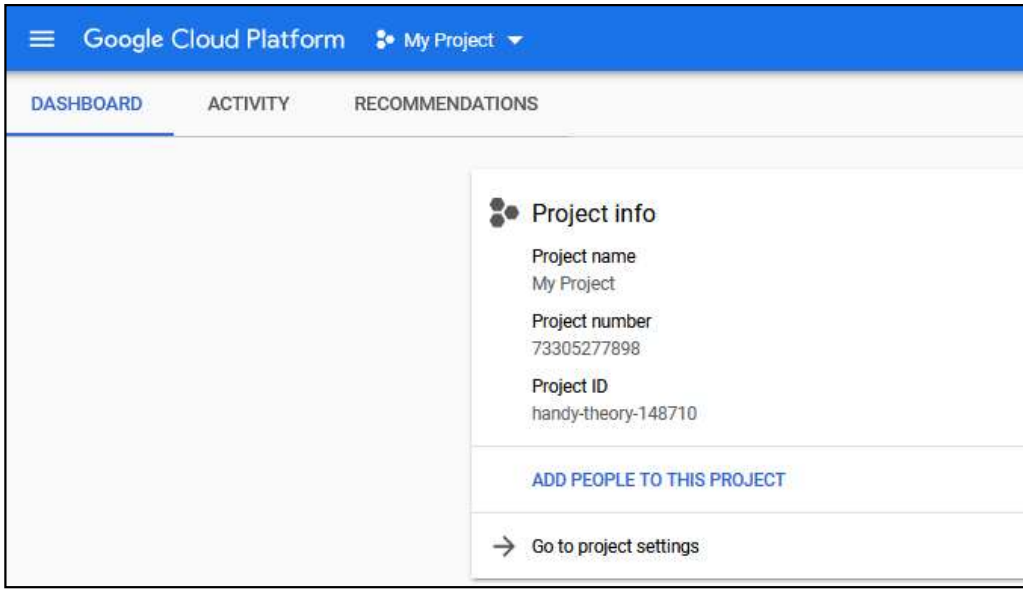
Google (Play/Cloud) Console

- Nie je konzola ako konzola !!!

https://play.google.com/console/u/0/signup



https://console.cloud.google.com



Cloud Console – Step 1

Step 1 Set up your project | Step 2 Enable APIs or SDKs | Step 3 Get an API Key

Console | Cloud SDK

1. In the Google Cloud Console, on the project selector page, click **Create Project** to begin creating a new Cloud project.

   Go to the project selector page |...

2. Make sure that billing is enabled for your Cloud project. Confirm that billing is enabled for your project.

Google Cloud offers a $0.00 charge trial. The trial expires at either end of 90 days or after the account has accrued $300 worth of charges, whichever comes first. Cancel anytime. Google Maps Platform features a recurring $200 monthly credit. For more information, see Billing account credits and Billing.

G-Fa

Google™

Invoice

Invoice number: 4603377165

Google Cloud EMEA Limited
Velasco
Clanwilliam Place
Dublin 2
Ireland
VAT number: IE3668997OH

Bill to
Peter Borovansky
B
B
81105 Bratislava
Slovakia

Details

Invoice number	4603377165
Invoice date	Oct 31, 2022
Billing ID	0195-6
Account ID	01C0F(

Google Cloud

Total in EUR	**€0.00**

Summary for Oct 1, 2022 - Oct 31, 2022

Subtotal in EUR	€0.00
VAT (20%)	€0.00
Total in EUR	€0.00

Poučenie z krízového vývoja

Nezadávajte číslo vašej karty skôr ako si neprečítate podmienky, resp. majte jednu „internetovú" platobnú kartu s obnosom, ktorého stratu viete poľahky prežiť

...stalo sa ...

Google Maps API

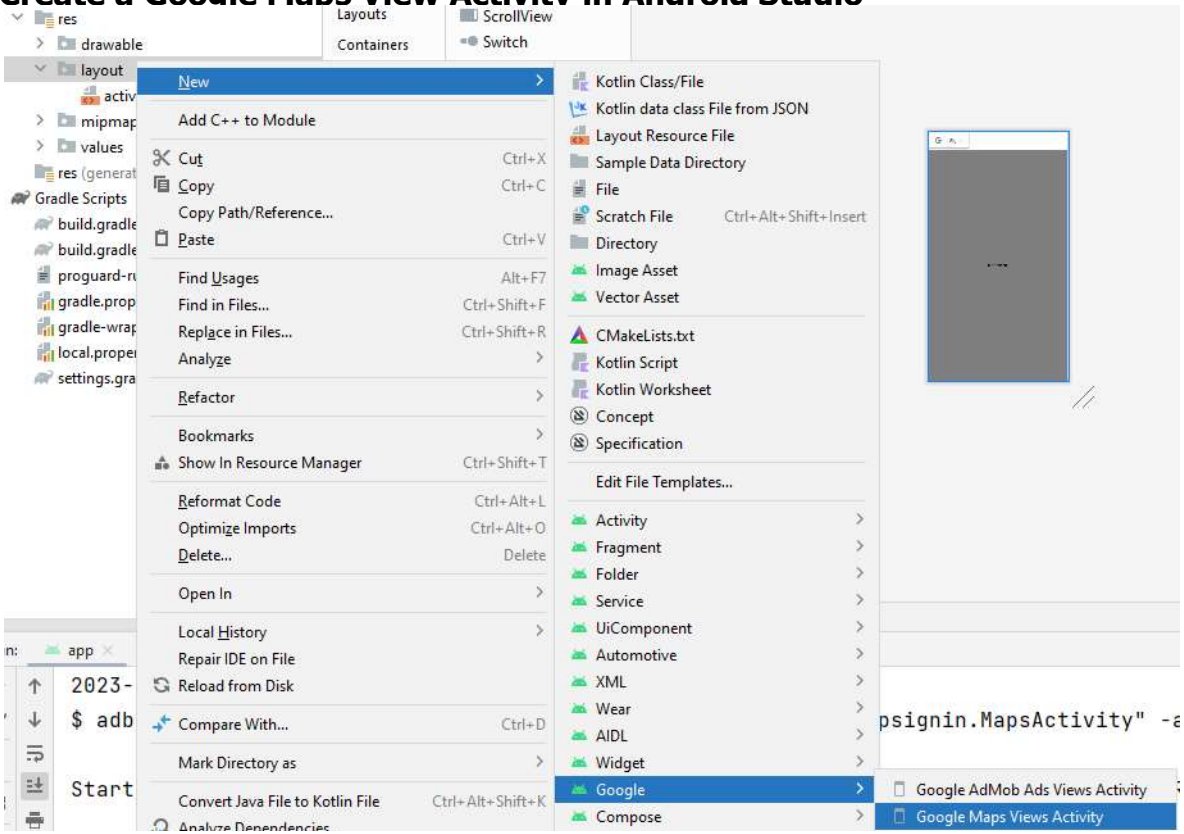
Treba **dôsledne** (!!!) prejsť oficiálnym návodom:

- https://developers.google.com/maps/documentation/android-api/start

Set up the development environment

1. Android Studio Arctic Fox or later is required. If you haven't already done so, download and install it.
2. Ensure that you are using the Android Gradle plugin version 7.0 or later in Android Studio.

Create a Google Maps View Activity in Android Studio



API_KEY z Cloud Console

The AndroidManifest.xml file contains instructions on getting a Google Maps API key and then adding it to your local.properties file.
Do not add your API key to the AndroidManifest.xml file.
Doing so stores your API key less securely.
Instead, follow the instructions to create a **Cloud project** and configure an API key.

local.properties

```
sdk.dir=C\:\\Users\\borovan\\AppData\\Local\\Android\\Sdk
MAPS_API_KEY=AIzaSyB31D……XSqLKtK4SATo
```

AndroidManifest.xml

```
<!--
    TODO: Before you run your application, you need a Google Maps API key.

    To get one, follow the directions here:

      https://developers.google.com/maps/documentation/android-sdk/get-api-key

    Once you have your API key (it starts with "AIza"), define a new property in your
    project's local.properties file (e.g. MAPS_API_KEY=Aiza...), and replace the
    "YOUR_API_KEY" string in this file with "${MAPS_API_KEY}".
-->
<meta-data
   android:name="com.google.android.geo.API_KEY"
   android:value="${MAPS_API_KEY}" />
```

Cloud Console – Step 2

project-level build.gradle

```
plugins {
    alias(libs.plugins.google.android.libraries.mapsplatform.secrets.gradle.plugin) apply false
}
// Top-level build file where you can add configuration options common to all sub-
projects/modules.
plugins {
    alias(libs.plugins.android.application) apply false
    alias(libs.plugins.kotlin.android) apply false
    alias(libs.plugins.google.android.libraries.mapsplatform.secrets.gradle.plugin) apply
false
}
```

module-level build.gradle

```
alias(libs.plugins.google.android.libraries.mapsplatform.secrets.gradle.plugin)
```

local.properties

```
MAPS_API_KEY=AIzaSyB31D.....XSqLKtK4SATo
```

AndroidManifest.xml

```
<meta-data
  android:name="com.google.android.geo.API_KEY"
  android:value="${MAPS_API_KEY}" />
```

Project:GMAPEmpty.zip

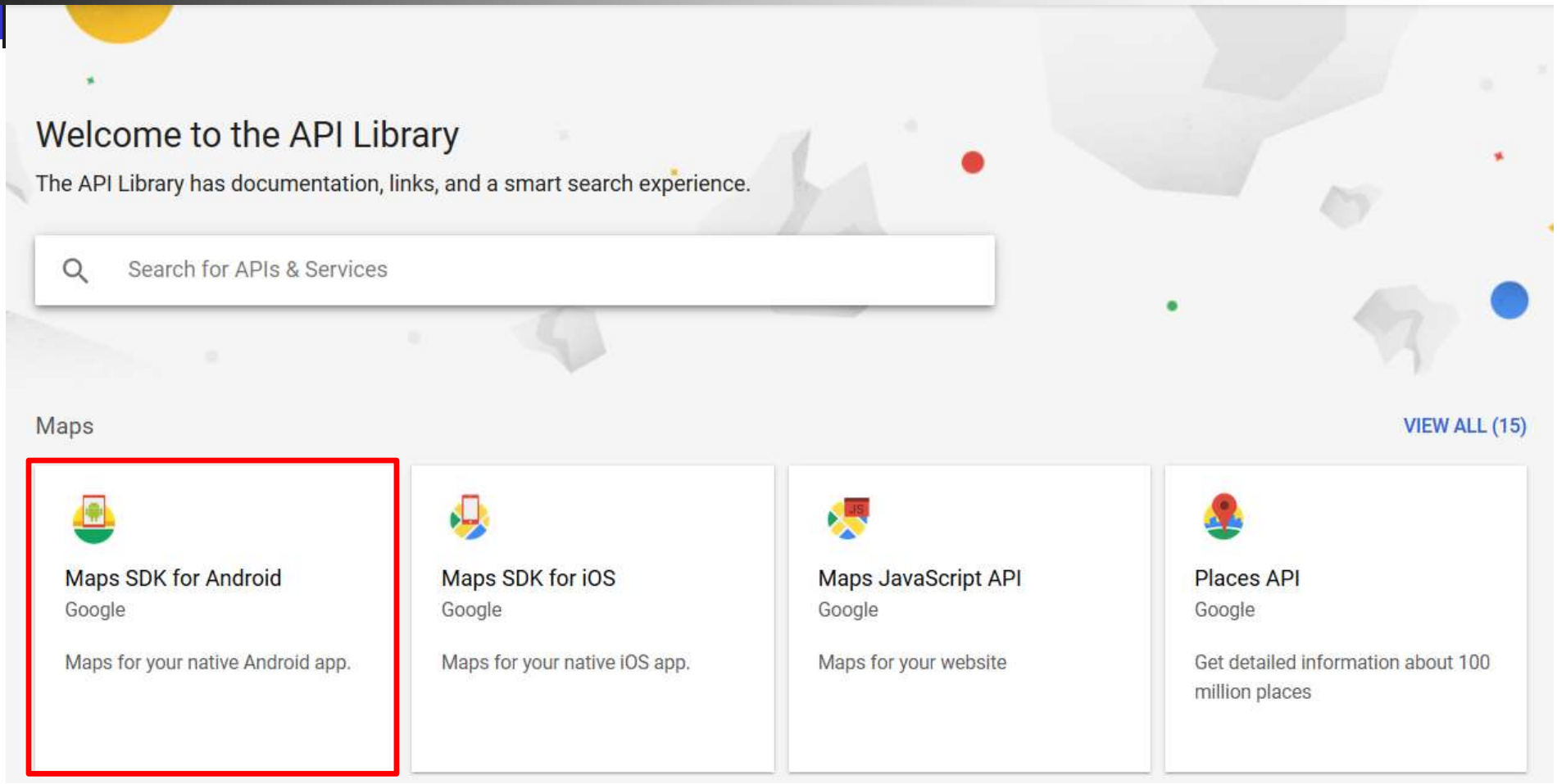
Nepodpísana appka

Authorization failure. Please see https://developers.google.com/maps/documentation/android-sdk/start for how to correctly set up the map.
2023-11-20 17:40:42.499 3496-3615 Google Android Maps SDK com.example.xxxxx123233
E In the Google Developer Console (https://console.developers.google.com)
Ensure that the "Maps SDK for Android" is enabled.
Ensure that the following Android Key exists:
API Key: YOUR_API_KEY
Android Application (<cert_fingerprint>;<package_name>):
7A:94:75:11:DD:3D:57:2A:……;com.example.xxxxx123233

Authorization failure. Please see https://developers.google.com/maps/documentation/android-sdk/start for how to correctly set up the map.
2023-11-20 17:45:36.578 3944-4052 Google Android Maps SDK com.example.xxxxx123233
E In the Google Developer Console (https://console.developers.google.com)
Ensure that the "Maps SDK for Android" is enabled.
Ensure that the following Android Key exists:
API Key: AIzaSyAJvZGfTlRxIs4oyDDb1xwJHN_m5VL2G88
Android Application (<cert_fingerprint>;<package_name>):
7A:94:75:11:DD:3D:57:2A:36:ED:2A:……;com.example.xxxxx123233

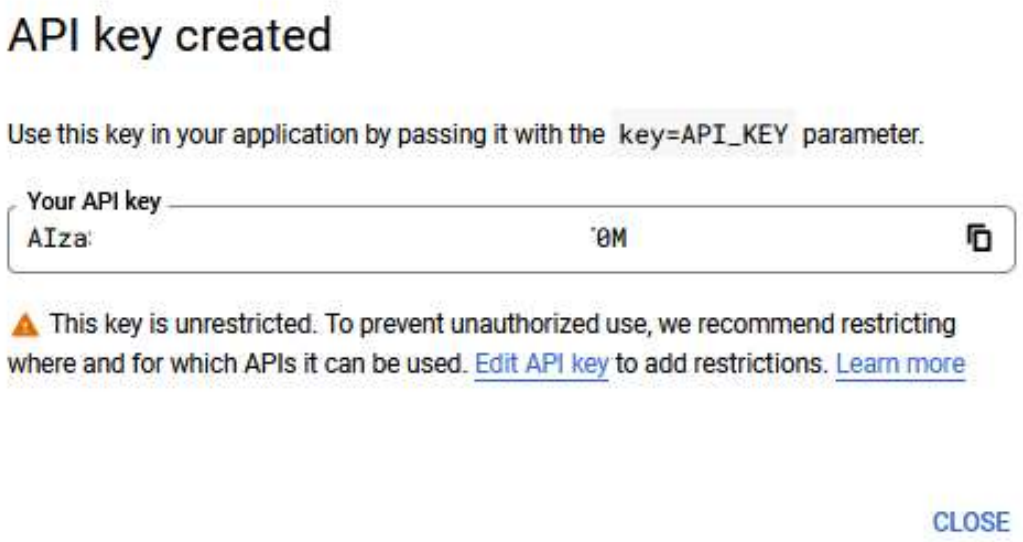
Google Developer Console

(https://console.developers.google.com/)



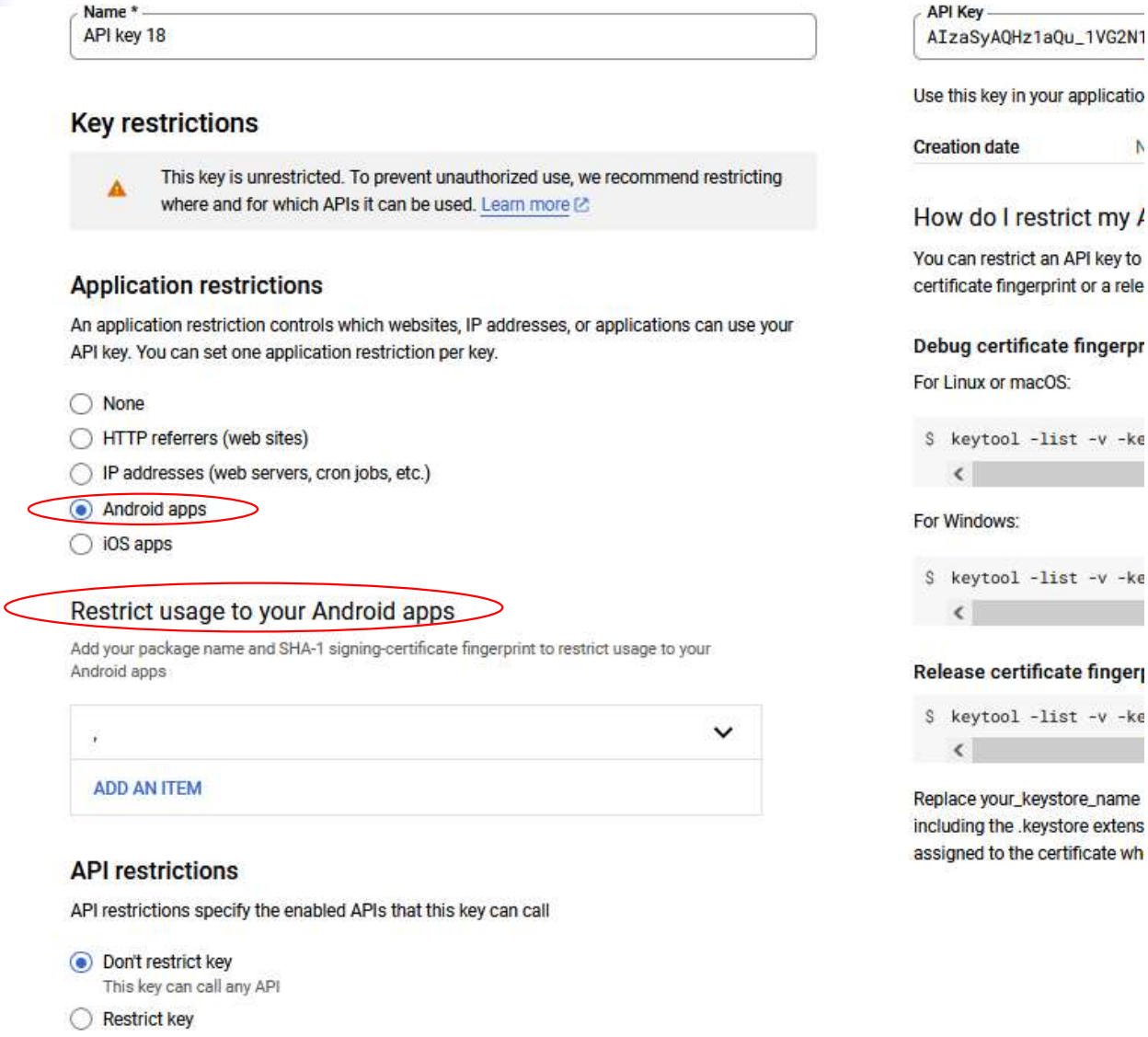
Cloud Console – Step 3

1. Go to the Google Maps Platform/Credentials page, click Create credentials/API key.
   The API key created dialog displays your newly created API key.
2. Click Close.
3. The new API key is listed on the Credentials page under API keys.

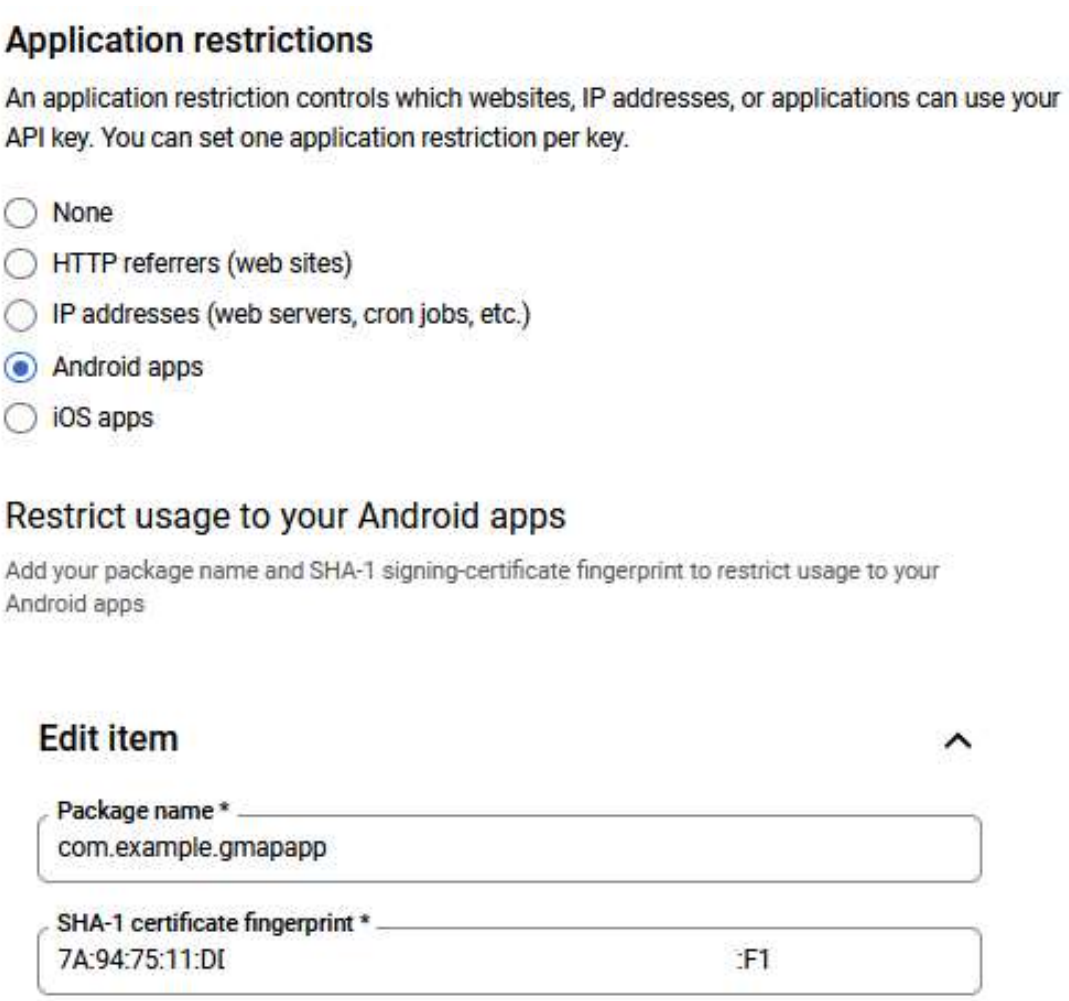


Project:GMAPEmpty.zip

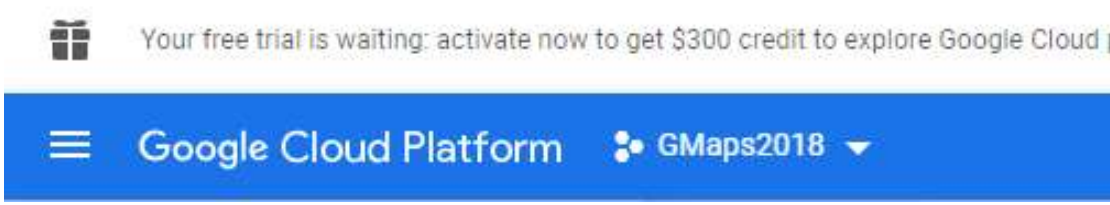
Restrict API Key



Restrict API Key



API Key

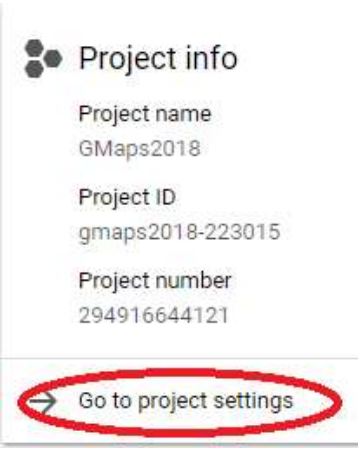


https://console.developers.google.com/apis/credentials/

API key

This API key can be used in this project and with any API that supports it. To use this key in your application, pass it with the `key=API_KEY` parameter.

Creation date	Nov 6, 2016, 12:26:20 PM
Created by	borovansky@gmail.com (you)



API key

AIzaSyB6e5[redacted]30EeC4QD1r8

Name

API key VMA 2016

Key restriction

Key restriction lets you specify which web sites, IP addresses, or apps can use this key. Learn more

- None
- HTTP referrers (web sites)
- IP addresses (web servers, cron jobs, etc.)
- Android apps
- iOS apps

API key pre Google Maps sa pridel'uje pre dvojicu, alebo niekoľko dvojíc, (package name, SHA1-v prostredí, kde kompilujete)

Restrict usage to your Android apps (Optional)

Add your package name and SHA-1 signing-certificate fingerprint to restrict usage to your Android apps

Get the package name from your AndroidManifest.xml file. Then use the following command to get the fingerprint:

```
$ keytool -list -v -keystore mystore.keystore
```

Package name	SHA-1 certificate fingerprint
pokus.example.com.myapplicationx	05:0F:34:5E:FD:E[redacted]6:8E:57:A2:80:1C:D1:91
pokus.example.com.gmapv2	05:0F:34:5E:FD:E[redacted]6:8E:57:A2:80:1C:D1:91
com.example.mapdemo	05:0F:34:5E:FD:E[redacted]6:8E:57:A2:80:1C:D1:91

Najčastejšie chyby s GMapsAPI

(jemne serioznejší pohľad)

http://ddewaele.github.io/GoogleMapsV2WithActionBarSherlock/part6

Beware of Map API key caching

ak package name, zmenu aplikujte aj na Google Console

ak zmeníte API key/package name, radšej odinštalujte starú verziu m-appky, nainštalujte opäť novú, API kľúč môže byť nacachovaný...

Ak u seba prekompilujete (napr. moju) m-appku, tak nepôjde...

musíte jej vygenerovať API key zodpovedajúci vášmu SHA1

Failing to provide the correct map permissions, API Key

najčastejšie sa vám nezobrazí mapa, okopírujte potrebné permissions napr. zo súboru, resp. iného, čo funguje

https://github.com/ddewaele/GoogleMapsV2WithActionBarSherlock/blob/master/GoogleMapsV2WithActionBarSherlock/AndroidManifest.xml

```
E/Google Maps Android API: Authorization failure.  Please see
https://developers.google.com/maps/documentation/android-api/start for how to correctly set up the map.
E/Google Maps Android API: In the Google Developer Console (https://console.developers.google.com)
    Ensure that the "Google Maps Android API v2" is enabled.
    Ensure that the following Android Key exists:
```

Najčastejšie chyby s GMapsAPI

(jemne serioznejší pohľad)

http://ddewaele.github.io/GoogleMapsV2WithActionBarSherlock/part6

API key problem

váš package name-SHA1-GMAPS_API_Key musia súvisieť cez Google Console:

- `Binary XML file line #2: Error inflating class fragment`
- `Caused by: java.lang.RuntimeException: API key not found. Check that`
- `Google Maps Android API(4040): Failed to contact Google servers.`

Zlé SDK

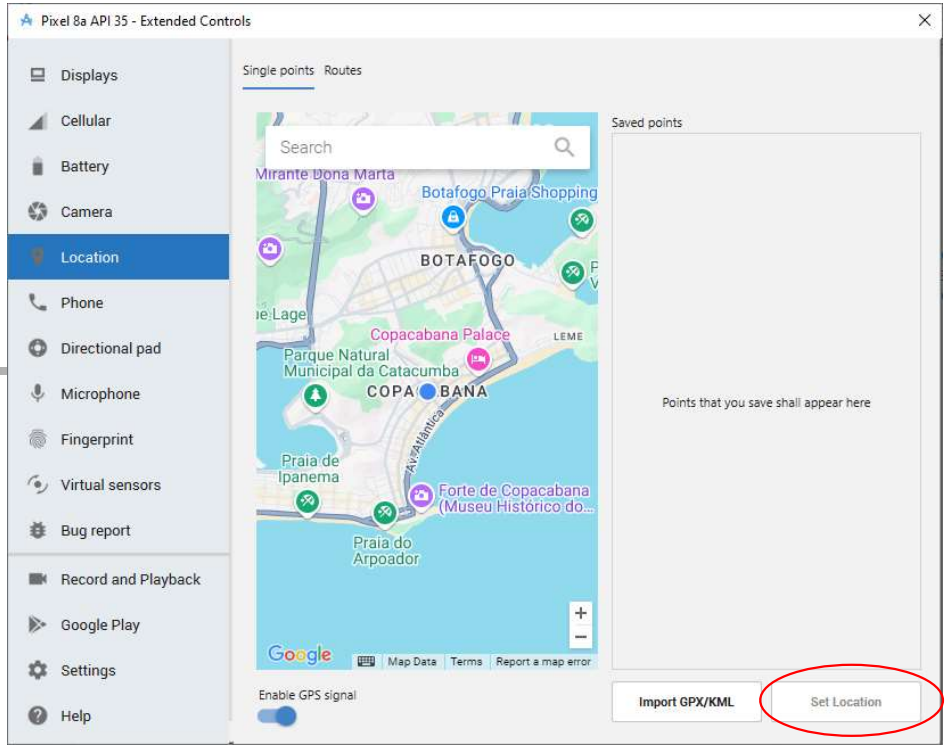
```
java.lang.RuntimeException: Unable to start activity
ComponentInfo{com.ecs.google.maps.v2.actionbarsherlock/com.ecs.google.maps.v2.simple.SimpleMapAct
ivity}:
android.view.InflateException: Binary XML file line #2: Error inflating class fragment at
android.app.ActivityThread.performLaunchActivity(ActivityThread.java:1651)
```

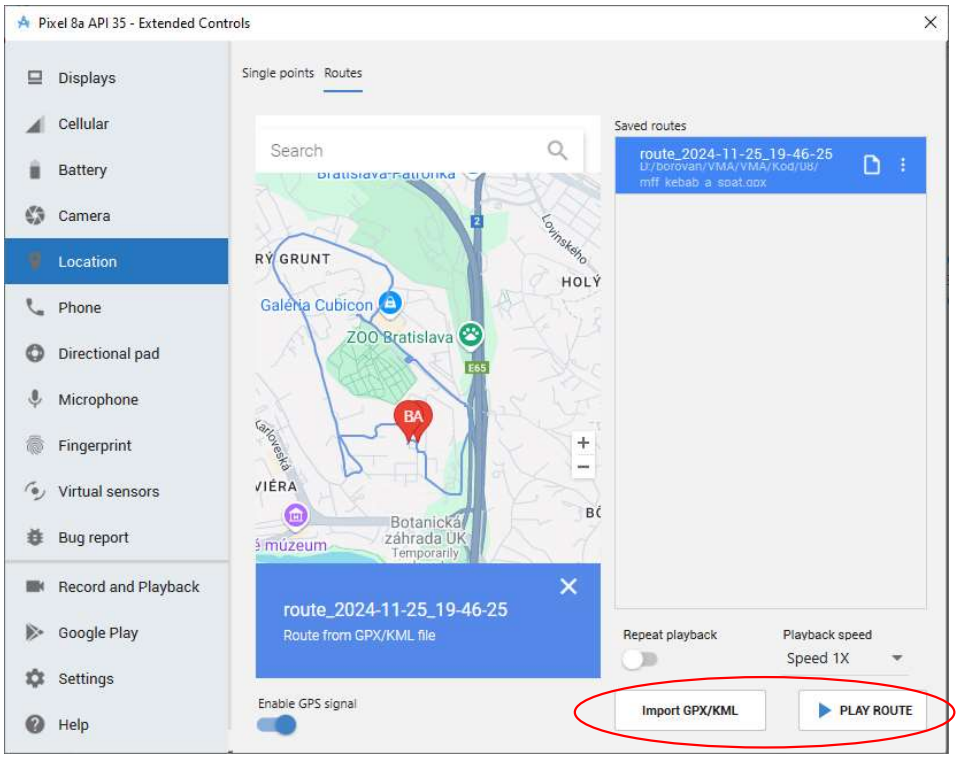
A mnohé iné problémy

http://ddewaele.github.io/GoogleMapsV2WithActionBarSherlock/part6

Gmap

emulator 2024





V úspešnom prípade
(v Android Studiu)

Ak sa všetko podarí, dostanete Layout s **SupportMapFragmentom**, a zobrazí sa mapa

```
class MapsActivity : AppCompatActivity(), OnMapReadyCallback {
    private lateinit var mMap: GoogleMap
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_maps) -- obsahuje Map fragment
        val mapFragment = supportFragmentManager
                .findFragmentById(R.id.map) as SupportMapFragment
        mapFragment.getMapAsync(this) -- otvorenie Gmaps chvíľku trvá
    } -- keď sa naložia, zavolá sa callback onMapReady v tejto triede

override fun onMapReady(googleMap: GoogleMap) { -- tu začína život m-appky
    mMap = googleMap                    -- referenciu na Gmapu si odložíme
    // Add a marker in Sydney and move the camera
    val sydney = LatLng(-34.0, 151.0)                   -- Sydney
    mMap.addMarker(MarkerOptions()                      -- balónik
          .position(sydney)
          .title("Marker in Sydney"))
    mMap.moveCamera(CameraUpdateFactory.newLatLng(sydney))
}
```

Project:GMAPEmpty.zip

Layout s MapFragmentom

```
<fragment
    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:id="@+id/map"
    android:name="com.google.android.gms.maps.SupportMapFragment
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="pokus.example.com....MapsActivity" />
```

typy mapových podkladov:

```
mMap.setMapType(
    GoogleMap.MAP_TYPE_SATELLITE)
    GoogleMap.MAP_TYPE_HYBRID)
    GoogleMap.MAP_TYPE_TERRAIN)
    GoogleMap.MAP_TYPE_NORMAL)
    -- pre offline mapy (zložitejšie)
    GooglemMap.MAP_TYPE_NONE)
```

Project:GMAPEmpty.zip

Permissions

- AndroidManifest.xml

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

- Žiadanie povolenia v runtime

```
val permission = ActivityCompat.checkSelfPermission(this,
                                    Manifest.permission.ACCESS_FINE_LOCATION)
if (permission == PackageManager.PERMISSION_GRANTED) {
    mMap.isMyLocationEnabled = true
} else {
    requestPermissions(
        arrayOf(Manifest.permission.ACCESS_FINE_LOCATION),
        LOCATION_REQUEST_CODE)
}
```

- Callback

```
override fun onRequestPermissionsResult(requestCode: Int,
          permissions: Array<String>, grantResults: IntArray) {
    when (requestCode) {
        LOCATION_REQUEST_CODE -> { ....
```

Project:GMAPEmpty.zip

Elementy Gmap API

- **`MapView : View`**, ktorá vie zobraziť Gmaps
- **`SupportMapFragment:Fragment`** - môže byť umiestnený v rámci fragmentu
- **`GoogleMap`** – hlavná trieda sa vytvorí s vytvorením MapView, SupportMF
- **`Marker`** – ikona na mape definovaná polohou latitude+longitude
- **`Shapes`** – útvary, napr. Polyline (lomená čiara), Polygon (n-uholník)
- **`UiSettings`** – nastavenie užívateľského rozhrania, napr. zoom-level, ...
- **`Overlays -`** vrstvy
- My Location – ak je povolená, zobrazuje sa button MyLocation, ktorý vycentruje mapu podľa aktuálnej polohy

https://developers.google.com/maps/documentation/android-sdk/groundoverlay

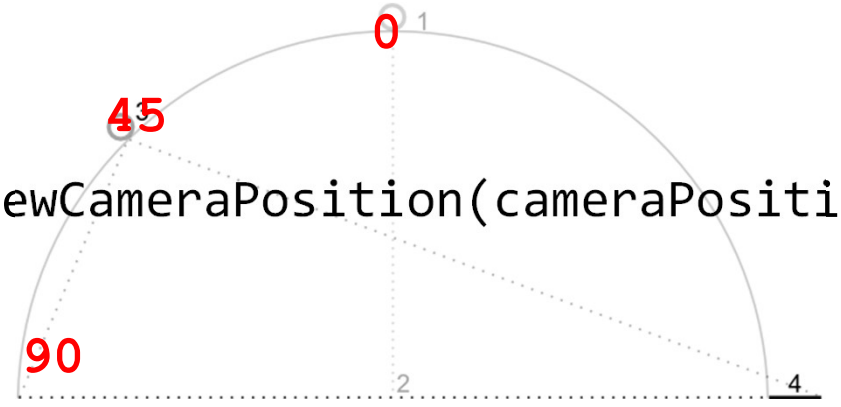
Marker/MarkerOptions

```
                                        // zobrazí moju polohu
mMap.isMyLocationEnabled = true         // button na mape
val mff = LatLng(48.151901, 17.068422) // cache z prémie
val MFF = mMap.addMarker(MarkerOptions()
    .position(mff)       // žiadne 1E6,ale slušná trieda LatLng
    .icon(BitmapDescriptorFactory  // ikona markera
                .fromResource(R.drawable.andro_cube))
    .draggable(true)     // vieme marker posúvať
    .alpha(0.5f)         // 0=transparent, 1=nontransparent
    .flat(true)          // marker sa nezoomuje s mapou
    .title("MFF")        // popis markera
    .anchor(0.0f, 1.0f) // pozícia ikony relatívne k position
    .snippet("Kockáči") // popis
    .rotation(90.0f)     // natočenie
)
```

Project:GMap.zip

Camera

```
mMap.isBuildingsEnabled = true
mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(mff, 6))
mMap.animateCamera(CameraUpdateFactory.zoomIn()) //zoomOut
mMap.animateCamera(CameraUpdateFactory.zoomTo(13),2000,null)

val cameraPosition = CameraPosition.Builder()
    .target(mff)      // kamera nasmerovaná na cieľ
    .zoom(17)         // finálny zoom level
    .bearing(90)      // azimut kamery, 90=východ
    .tilt(30)         // horizontalne natočenie 30-90
    .build()
mMap.animateCamera(
   CameraUpdateFactory.newCameraPosition(cameraPosition))
```



Porovnanie: ESB vs. FMFI

Project:GMap.zip

Keška MFF

LukášM.

FilipB.

Google Maps

google.com/maps/@48.1512272,17.0688162,59a,35y,350.06h,66.13t/data=!3m1!1e3

Search Google Maps

See travel times, traffic and nearby places

FreeFood

Map

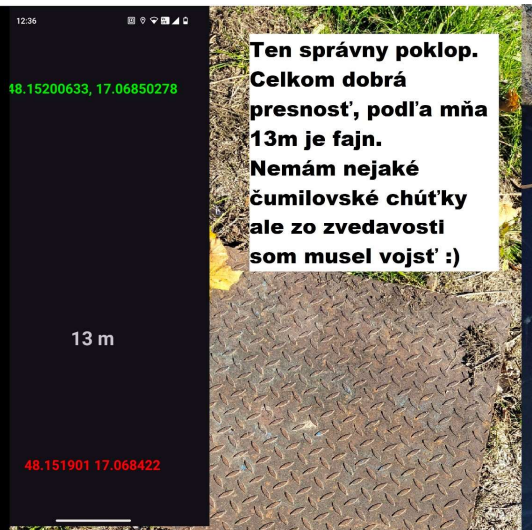
Google

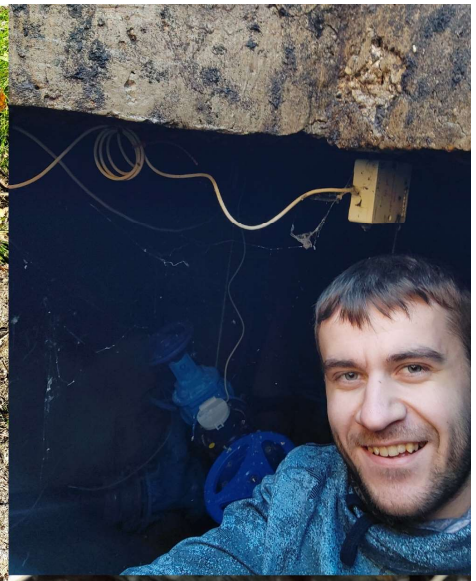
Imagery ©2020 Google, Data SIO, NOAA, U.S. Navy, NGA, GEBCO, Landsat / Copernicus, Imagery ©2020 Maxar Technologies, Map data ©2020 Slovakia Terms Send feedback 10 m

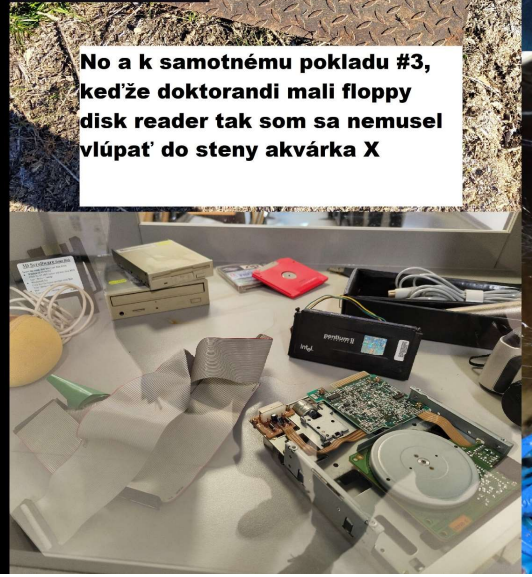
2D

MichalV

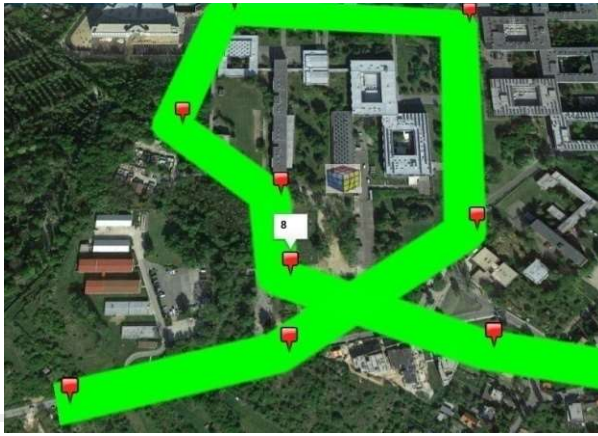
problém, že „Darovanej kobyle do zubov nehľadia”



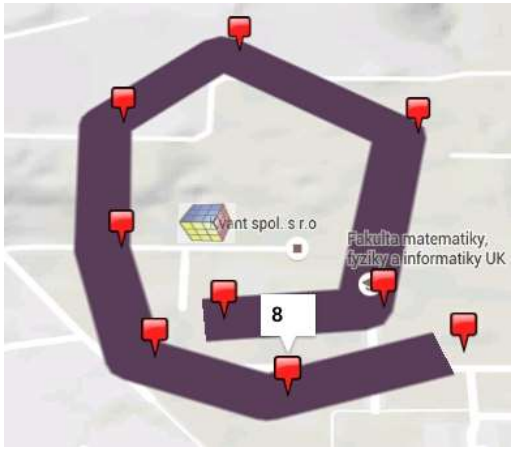




onMapClick, PolyLine



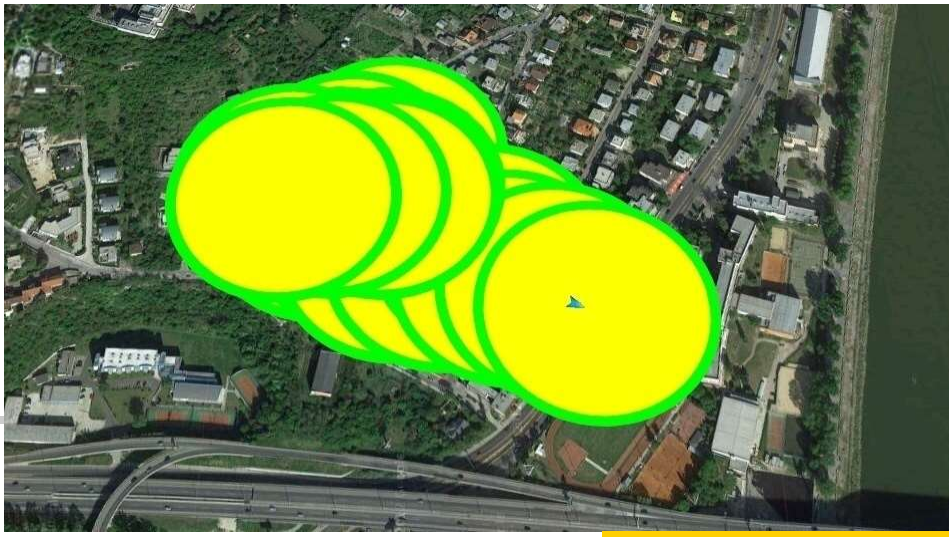
```
clickedLine = mMap.addPolyline(PolylineOptions())
                                  // ak chceme klikať do mapy
mMap.setOnMapClickListener {
    latlng ->   // onClickListener – nefunguje pri satelitných m.
        val clickedPoints = clickedLine.points
        clickedPoints.add(latlng)
        clickedLine.points = clickedPoints
        clickedLine.color = ...
        clickedLine.width = ...

        mMap.addMarker(MarkerOptions()
          .title("" + clickedPoints.size())
          .icon(BitmapDescriptorFactory
                .fromResource(R.drawable.marker))
          .position(latlng))
```



Project:GMap.zip

GPS Location

(prvá možnosť - GPS)



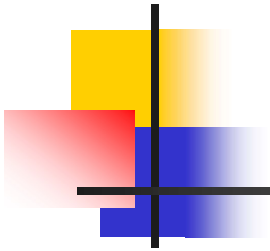
Project:GMap.zip

Použijeme `LocationManager`, ako v minulej prednáške, a `LocationListener`

```
override fun onLocationChanged(loc:Location) {
  val latlng = LatLng(loc.getLatitude(), loc.getLongitude())
  val circle = mMap.addCircle(CircleOptions(latlng)
          .center()
          .radius(15)          // polomer kruhu v metroch
          .strokeColor(Color.GREEN)
          .fillColor(Color.YELLOW))
  circle.setZIndex((float)System.currentTimeMillis()) }
```

```
  val bounds =                        // ak si mimo...
       mMap.getProjection().getVisibleRegion().latLngBounds
  if(!bounds.contains(latlng))   // preanimuj na novú polohu
    mMap.animateCamera(CameraUpdateFactory.newLatLng(latlng)
```

Project:GMap.zip



uiSettings

```
map.uiSettings
        .isMyLocationButtonEnabled
        .isCompassEnabled
        .isRotateGesturesEnabled
        .isScrollGesturesEnabled
        .isScrollGesturesEnabledDuringRotateOrZoom
        .isZoomGesturesEnabled
        .isTiltGesturesEnabled
        .isMapToolbarEnabled
```

Project:GMap.zip

LocationServices API

(Fused Location API)

https://developers.google.com/location-context/fused-location-provider

Simple, battery-efficient location API for Android

Apps can take advantage of the signals provided by multiple sensors in the device to determine device location. However, choosing the right combination of signals for a specific task in different conditions is not simple. Finding a solution that is also battery-efficient is even more complicated.

The fused location provider is a location API in Google Play services that intelligently combines different signals to provide the location information that your app needs.

The fused location provider manages the underlying location technologies, such as GPS and Wi-Fi, and provides a simple API that you can use to specify the required quality of service. For example, you can request the most accurate data available, or the best accuracy possible with no additional power consumption.

```
/**
 * Provides the entry point to the Fused Location Provider API.
 */
private FusedLocationProviderClient mFusedLocationClient;

/**
 * Represents a geographical location.
 */
protected Location mLastLocation;

private String mLatitudeLabel;
private String mLongitudeLabel;
private TextView mLatitudeText;
private TextView mLongitudeText;

/**
 * Provides a simple way of getting a device's location and is well
 * applications that do not require a fine-grained location and tha
 * updates. Gets the best and most recent location currently availa
 */
private void getLastLocation() {
    if(ContextCompat.checkSelfPermission( context: this, Manifest.pers
        mFusedLocationClient.getLastLocation()
                .addOnCompleteListener( activity: this, new OnCompleteL
                    @Override
                    public void onComplete(@NonNull Task<Location>
                        if (task.isSuccessful() && task.getResult()
                            mLastLocation = task.getResult();

                            mLatitudeText.setText(String.format(Loc
                                    mLatitudeLabel,
```

Support for common location scenarios

Last Known Location

Location Settings

Project:GMapSOS.zip

LocationServices API

(Fused Location API)

https://developers.google.com/location-context/fused-location-provider

Iná možnosť, ako získať poslednú polohu na vyššej úrovni ako od GPS senzora

- neriešite location providera (GPS/NETWORK/wifi),
- vysoká presnosť, nízka spotreba baterky,
- ale používajú Google Play Services, do build.gradle doplníte:

```
dependencies {
    implementation 'com.google.android.gms:play-services-location:18.0.0‘
    implementation 'com.google.android.gms:play-services-maps:18.0.0'
}
```

- `import com.google.android.gms.*...`

- `fusedLocationProviderClient = LocationServices.getFusedLocationProviderClient(this@MapsActivity)`

-
```
locationCallback = object : LocationCallback() {
    override fun onLocationResult(locationResult: LocationResult) {
        for (location in locationResult.locations) { // pri zmene polohy
                ...  location je aktuálna poloha
        }
 }  }
```

Project:GMapSOS.zip

Fused Location API

(onResume, onPause)

https://developer.android.com/training/location/retrieve-current.html

```
override fun onResume() {
    super.onResume()
    val interval = (10 * 1000).toLong()        // 10 seconds, in milliseconds
    val fastestInterval = (1 * 1000).toLong()  // 1 second, in milliseconds
    val minDisplacement = 0f

    // Create the LocationRequest object
    val mLocationRequest = LocationRequest.create()      // criteria
        .setPriority(LocationRequest.PRIORITY_BALANCED_POWER_ACCURACY)
        .setInterval(interval)        -- ako často chceme update polohy
        .setFastestInterval(fastestInterval) -- ako často vieme spracovať update
                  polohy-pre prípad, že iná apka intenzívnejšie používa GPS
        .setSmallestDisplacement(minDisplacement)

    fusedLocationProviderClient.requestLocationUpdates(
        mLocationRequest,
        locationCallback,
        Looper.getMainLooper())
}
override fun onPause() {
    super.onPause()
    fusedLocationProviderClient.removeLocationUpdates(locationCallback)
}
```

```
LocationRequest.PRIORITY_HIGH_ACCURACY ~GPS
LocationRequest.PRIORITY_BALANCED_POWER_ACCURACY; ~100m,block
LocationRequest.PRIORITY_LOW_POWER;   ~ 10km,mesto
LocationRequest.PRIORITY_NO_POWER; berie polohu od iných klientov
```

Project:GMapSOS.zip

Fused Location API

(LocationCallback)

https://developer.android.com/training/location/retrieve-current.html

```
locationCallback = object : LocationCallback() {
                                        // pri zmene polohy podľa LocationRequest
  override fun onLocationResult(locationResult: LocationResult) {
     for (location in locationResult.locations){
       Log.w(TAG, "onLocationResult= ${location?.latitude}, ${location?.longitude}")
       val savedLastLocation = LatLng(location.latitude, location.longitude)
       val options = MarkerOptions()
                           .position(savedLastLocation)
                           .title("I am here!")
       mMap.addMarker(options)
       mMap.animateCamera(CameraUpdateFactory.newLatLngZoom(savedLastLocation, 15f))
     }
  }
}
```

```
binding.sendSosBTN.setOnClickListener {
    val sendIntent = Intent(Intent.ACTION_VIEW)
    sendIntent.data = Uri.parse("sms:")
    sendIntent.putExtra("sms_body",
        "My location at \nLatitude: ${savedLastLocation.latitude} \n" +
                "Longitude: ${savedLastLocation.longitude}")
    startActivity(sendIntent)
}
```

Project:GMapSOS.zip

Poloha vs. poloha

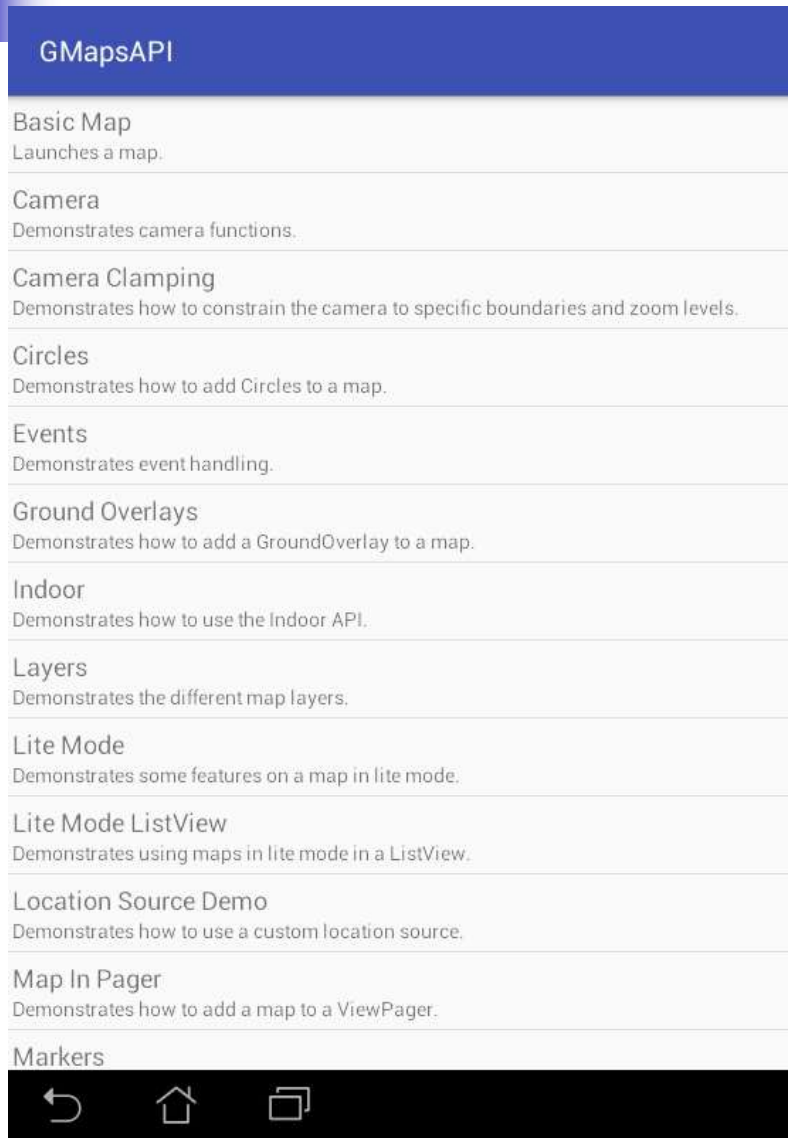
https://antoniohongkr.wordpress.com/2013/08/19/google-play-service-analysis-4-choice-between-google-play-location-service-and-android-location-service/

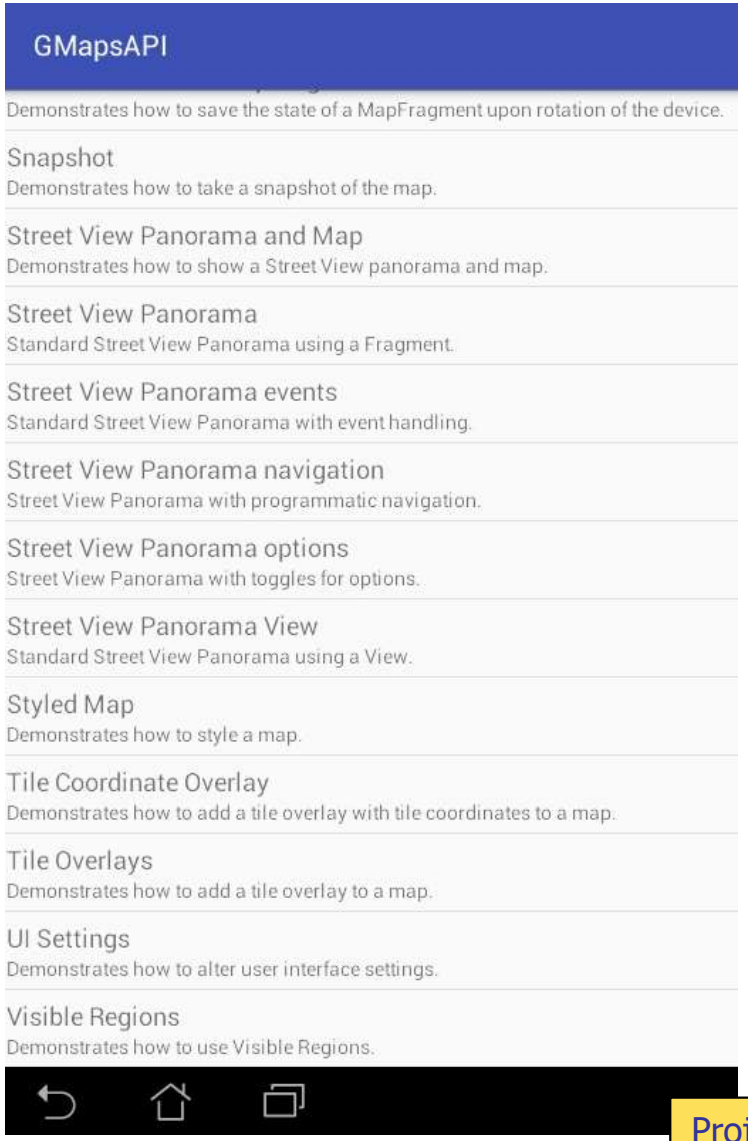
Priority	Typical location update interval	Battery drain per hour (%)	Accuracy
HIGH_ACCURACY	5 seconds	7.25% **~1/14**	~10 meters
BALANCED_POWER	20 seconds	0.6%	~40 meters
NO_POWER	N/A	small	~1 mile

Demo pre inšpirácie

(kód je v jave)

https://github.com/googlesamples/android-play-location

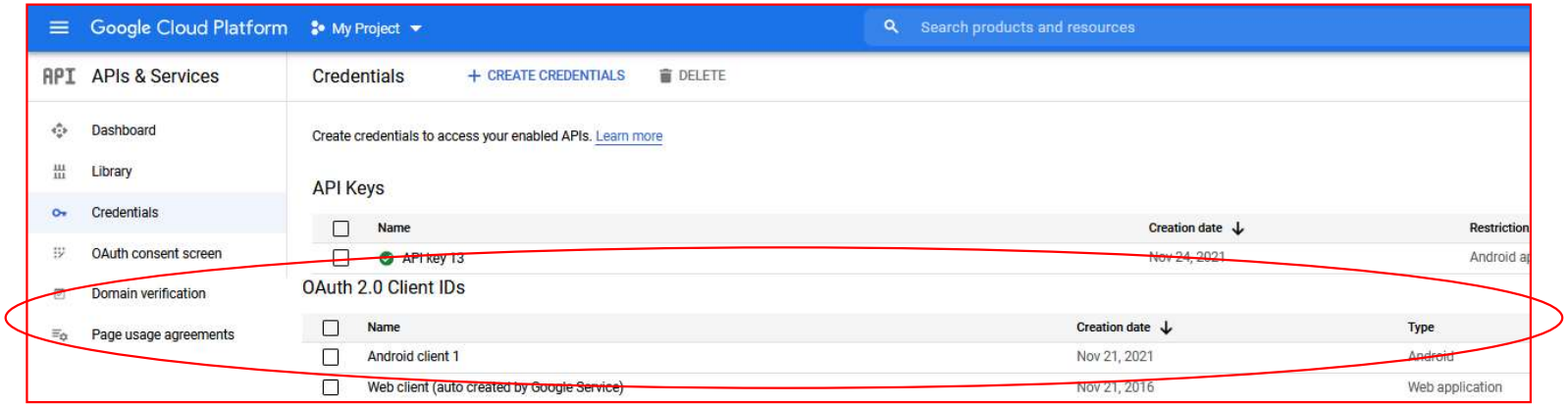




Project:ApiDemos.zip

GoogleSignIn

```
val gso = GoogleSignInOptions.Builder(GoogleSignInOptions.DEFAULT_SIGN_IN)
    .requestEmail()
    .build()

// Build a GoogleSignInClient with the options specified by gso.
val mGoogleSignInClient = GoogleSignIn.getClient(this, gso)
// Set the dimensions of the sign-in button.
sign_in_button.setSize(SignInButton.SIZE_STANDARD)
sign_in_button.setOnClickListener {
    val signInIntent = mGoogleSignInClient.signInIntent
    startActivityForResult(signInIntent, SIGN_IN_REQUESTCODE)
}
val mGoogleApiClient = mGoogleSignInClient.asGoogleApiClient()
```



GMapsSignIn.zip

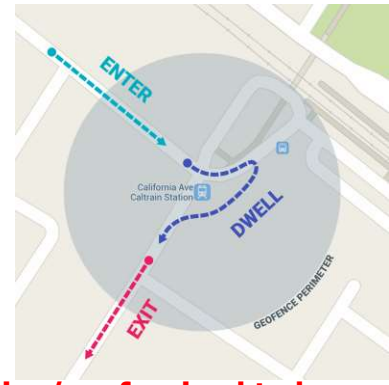
GoogleSignIn

```
override fun onActivityResult(requestCode : Int, resultCode : Int, data : Intent?) {
    super.onActivityResult(requestCode, resultCode, data)

    if (requestCode == SIGN_IN_REQUESTCODE) {
        // The Task returned from this call is always completed, no need to attach
        // a listener.
        val task : Task<GoogleSignInAccount> =
                    GoogleSignIn.getSignedInAccountFromIntent(data)
        handleSignInResult(task)
}   }
private fun handleSignInResult(completedTask: Task<GoogleSignInAccount>) {
    try {
        val account = completedTask.getResult(ApiException::class.java)
        Log.w(TAG, "signInResult:success account= ${account.displayName}")
    } catch (e: ApiException) {
        Log.w(TAG, "signInResult:failed code=" + e.statusCode)
    }
}
```

GMapsSignIn.zip

Geofencing

ploty a zábradlia



https://developer.android.com/training/location/geofencing.html

- detekuje vchod/východ/trvanie v oblasti, ktorá môže expirovať=zmiznúť
- AndroidManifest.xml:

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
<application
      <service android:name=".GeofenceTransitionsIntentService"/>
<application/>
```

Kruhová ohrada (zoznam ohraničený na 100 oblastí):

```
mGeofenceList.add(
   Geofence.Builder()
        .setRequestId(entry.getKey())
        .setCircularRegion(entry.getValue().latitude,
                           entry.getValue().longitude,
                Constants.GEOFENCE_RADIUS_IN_METERS)
        .setExpirationDuration(Constants.GEOFENCE_EXPIRATION_IN_MILLISECONDS)
        .setTransitionTypes(Geofence.GEOFENCE_TRANSITION_ENTER |
                           Geofence.GEOFENCE_TRANSITION_EXIT)
        .build())
```