

Android alkalmazásfejlesztés

Google Maps **V1**

Google Maps **V2**

OpenStreetMap, **Mapsforg**e

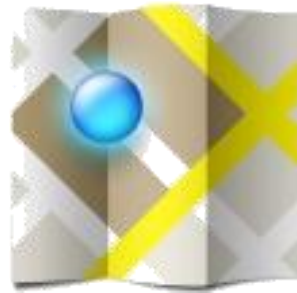
OE-NIK

2013. március 18.

Sicz-Mesziár János

sicz-mesziar.janos@
nik.uni-obuda.hu





Google Maps **V1** *deprecated*

Google Maps V1

- ◎ Jól használható külső library. 😊
- ◎ Használatához regisztráció és az ÁSZF elfogadása kell!
- ◎ Google Maps API kulcs beszerzése:
 - Google API telepítés Android SDK Manager-ben
 - MD5 lenyomat készítés:

```
C:\Program Files\Java\<JDK_version>\bin>keytool.exe -list -alias  
androiddebugkey -keystore "c:\Users\<Username>\.android\debug.keystore"  
-storepass android -keypass android -v
```
 - API kulcsszerzés, MD5 lenyomat alapján:
<http://code.google.com/intl/hu-HU/android/maps-api-signup.html>
- ◎ AndroidManifest-ben library használat hozzáadása:
Application fül > Application Nodes > Add... > Uses Library
Name: ***com.google.android.maps***
- ◎ AndroidManifest-ben internet használat kérés:
Permissions fül > Add... > Uses Permission
Name: ***android.permission.INTERNET***

Google Maps **V1** a gyakorlatban

◎ MapView elhelyezése a layout-on:

```
<com.google.android.maps.MapView  
    android:layout_width="fill_parent"  
    android:layout_height="fill_parent"  
    android:enabled="true"  
    android:clickable="true"  
    android:id="@+id/myMap"  
    android:apiKey="<-- API KEY -->" />
```

◎ Google API target beállítása:

Projects menü > Properties > Android > Project Build
Target > Google APIs kiválasztása

Emulátort is át kell állítani!!!

◎ Osztályunk egy MapActivity leszámazott:

```
class MyMapActivity extends MapActivity{  
    protected boolean isRouteDisplayed() {}  
    MapView myMap = (MapView) findViewById(R.id.myMap);  
}
```

Zoom Controls, Map Controller

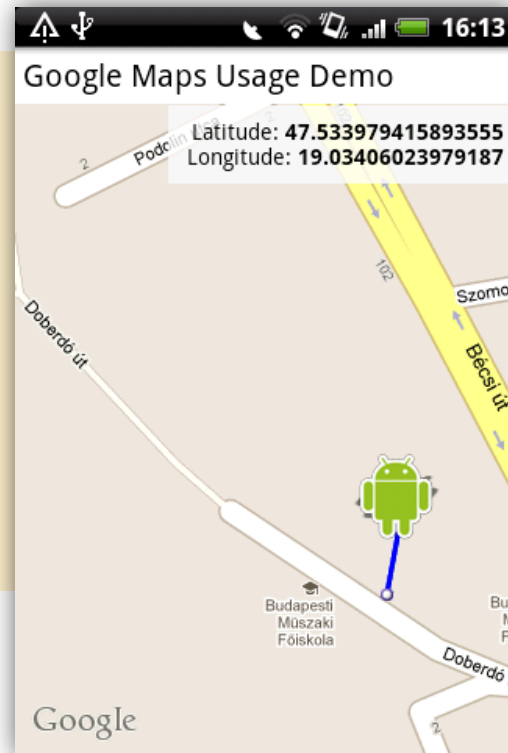
- ◉ Beépített +/- zoom engedélyezése:

```
myMap.setBuiltInZoomControls(true);  
myMap.getZoomButtonsController().setAutoDismissed(false); // Automatikus elrejtés tiltása
```

- ◉ Egyéni zoom gombokhoz saját ZoomControl kell.

Térkép irányítása MapController-rel:

```
MapController mc = myMap.getController();  
mc.setCenter(  
    new GeoPoint(47533571, 19033851));  
mc.setZoom(17);  
mc.scrollBy(120, 120); // Pixelben értendő  
mc.animateTo(new GeoPoint(...));
```



Pontok és pozíciók elhelyezése

⦿ Kell egy ItemizedOverlay<OverlayItem> leszármazott:

- Hozzá egy konstruktor, amiben az alap Marker-t megadjuk
- Egy listába felgyűjtjük a réteghez tartozó pontokat
- Előírt/ajánlott metódusok megvalósítása:
 - OverlayItem createltem(int i) // i.-dik elem
 - int size() // listánk elemszáma
 - + : void addOverlay(OverlayItem overlay) // hozzáadás

⦿ Overlay hozzáadása a térképhez

```
MyItemizedOverlay pointsOverlay =  
    new MyItemizedOverlay(pointDefaultMarker); //Drawable  
List<Overlay> mapOverlays = myMap.getOverlays();  
mapOverlays.add(pointsOverlay);
```

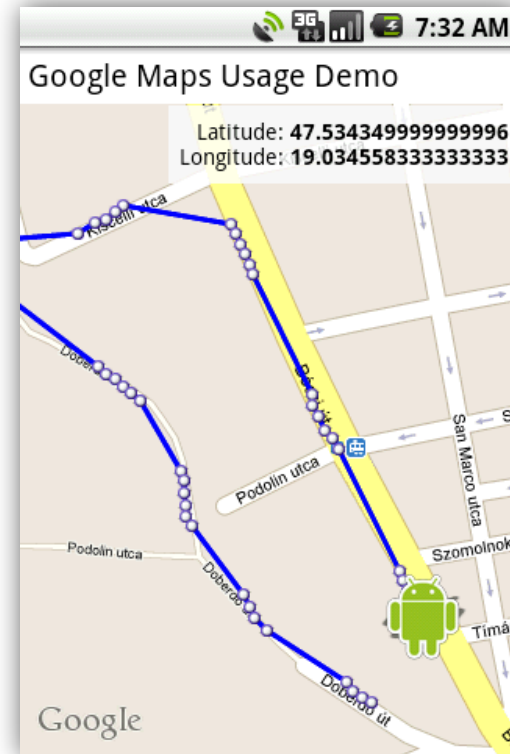
⦿ Overlay feltöltése adatokkal

```
OverlayItem item = new OverlayItem(new GeoPoint(  
    (int)(location.getLatitude() * 1E6),  
    (int)(location.getLongitude()* 1E6)  
), "Hello point", "Deatils...");  
pointsOverlay.addOverlay(item);  
myMap.invalidate();
```

Rajzolás térképre

© Adott Overlay draw() metódusának felüldefiniálása:

```
public void draw(Canvas canvas, MapView mapView, boolean shadow) {
    Projection projection = mapView.getProjection();
    if(overlays.size() > 0 && projection != null){
        Point startPoint = new Point();
        projection.toPixels(overlays.get(0).getPoint(), startPoint);
        line.reset(); // Az első pont hozzáadása a Path-hoz = Kezdőpont
        line.moveTo(startPoint.x, startPoint.y);
        // Végig megyünk a pontokon és hozzáadjuk a Path-hoz
        for(OverlayItem i : overlays){
            Point point = new Point();
            projection.toPixels(
                i.getPoint(),
                point
            );
            line.lineTo(point.x, point.y);
        }
        // És akkor a lényeg, a rajzolás 😊
        canvas.drawPath(line, p);
    }
    super.draw(canvas, mapView, shadow);
}
```



Oda-vissza geo kódoló

⊙ Lényege, hogy koordinátából címet, illetve címből koordinátát fejtsünk vissza.

⊙ Geo kódoló:

```
Geocoder geoCoder =  
    new Geocoder(getBaseContext(), Locale.getDefault());  
List<Address> addresses = geoCoder.getFromLocation(  
    p.getLatitudeE6()/1E6, p.getLongitudeE6()/1E6, 1);  
// addresses.get(0).getAddressLine(i);
```

⊙ Visszakódoló:

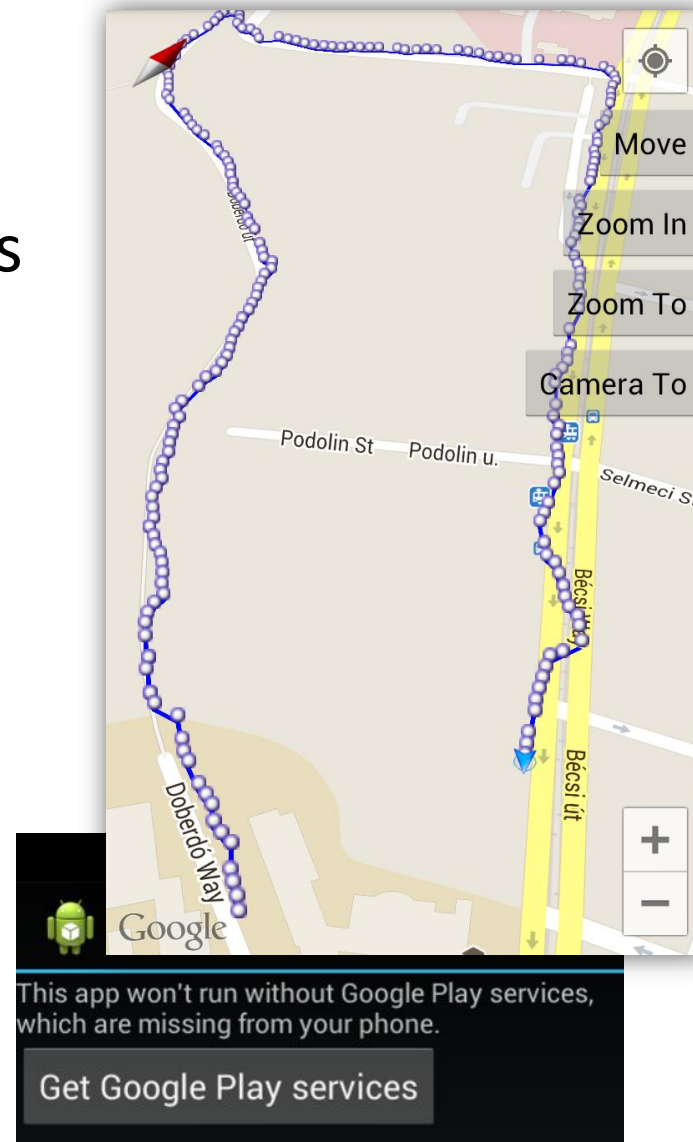
```
List<Address> addresses =  
    geoCoder.getFromLocationName(„Budapest, Bécsi út“, 5);  
p = new GeoPoint(  
    (int) (addresses.get(0).getLatitude() * 1E6),  
    (int) (addresses.get(0).getLongitude() * 1E6)  
);
```




Google Maps V2

Újdonságok

- ◉ Minimum Android 2.2 kell
- ◉ Google Play SDK része lett
- ◉ Vektorgrafikus megjelenítés
- ◉ Fragment támogatás
- ◉ 3D térkép, OpenGL 2.0
- ◉ Kisebbs adatforgalom
- ◉ Jobb cache használat
- ◉ Több interakció: forgatható kamera-nézet
- ◉ In-door támogatás
- ◉ TileOverlay támogatás
[x,y,z → bitmap]



Google Maps V2 API kulcs beszerzése

1. Certificate SHA-1 lenyomat

```
keytool -list -v -keystore  
"C:\Users\[user_name]\.android\debug.keystore"  
-alias androiddebugkey  
-storepass android  
-keypass android
```

2. API projekt regisztráció - Google API Console

- <https://code.google.com/apis/console/>
- Create project... (ha szükséges)
- Services > Google Maps Android V2-t engedélyezzük

3. API kulcs beszerzése

- API Access
- Create new Android key...
- SHA-1 lenyomat + package név bemásolása

Google Maps V2 a gyakorlatban

◎ AndroidManifest.xml

- Az <Appliaction> végére:

```
<meta-data
    android:name="com.google.android.maps.v2.API_KEY"
    android:value="your_api_key"/>
```

- OpenGL 2.0 megkötés

```
<uses-feature
    android:glEsVersion="0x00020000"
    android:required="true"/>
```

- Speciális Jogosultság hozzáadása

```
<permission
    android:name="[package].permission.MAPS_RECEIVE"
    android:protectionLevel="signature"/>
<uses-permission
    android:name="[package].permission.MAPS_RECEIVE,,
/>
```

Google Maps V2 a gyakorlatban (2)

© AndroidManifest.xml

```
<uses-permission
    android:name="android.permission.INTERNET" />
<uses-permission
    android:name="android.permission.ACCESS_NETWORK_
STATE" />
<uses-permission
    android:name="android.permission.WRITE_EXTERNAL_
STORAGE" />
<uses-permission
    android:name="com.google.android.providers.gsf.pe
    rmission.READ_GSERVICES" />
```

© Google Play Service library hozzáadása

- File > Import > Existing Android Code into Workspace
- <android_sdk>/extras/google/google_play_services/libproject/google-play-services_lib
- Project > Properties > Android > Library, Add...

Google Maps V2 a gyakorlatban (3)

◎ activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<fragment
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/map"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:name=
        "com.google.android.gms.maps.SupportMapFragment"/>
```

◎ MainActivity.java

```
public class MainActivity extends FragmentActivity {
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        GooglePlayServicesUtil.
            isGooglePlayServicesAvailable(getContext());
    }
}
```

Vezérlés térkép konfiguráció

◎ XML leíróban

```
<fragment
```

```
.....
```

```
xmlns:map="http://schemas.android.com/apk/res-auto"
```

```
map:cameraTargetLat="-33.796923"
```

```
map:cameraTargetLng="150.922433,,
```

```
map:cameraBearing="112.5"
```

```
map:cameraTilt="30"
```

```
map:cameraZoom="13"
```

```
map:mapType="normal"
```

```
map:uiCompass="false"
```

```
map:uiRotateGestures="true"
```

```
map:uiScrollGestures="false"
```

```
map:uiTiltGestures="true"
```

```
map:uiZoomControls="false"
```

```
map:uiZoomGestures="true" />
```

tilt = dőlés mértéke
bearing = elfordulás

Map types:

NORMAL, SATELLITE, HYBRID, TERRAIN, NONE

◎ Programkódból

```
MapFragment.newInstance(  
    new GoogleMapOptions()  
        .mapType(MAP_TYPE_NORMAL)  
        .compassEnabled(true)  
        .rotateGesturesEnabled(true)  
        .tiltGesturesEnabled(true)  
        .scrollGesturesEnabled(true)  
        .zoomControlsEnabled(true)  
        .zoomGesturesEnabled(true)  
);
```

Vezérlés **kamera mozgatás**

◎ Pozíció megváltoztatása: center, zoom

```
map.moveCamera(CameraUpdateFactory.newLatLngZoom(  
    new LatLng(47.533599, 19.033899),  
    14 ));  
  
map.animateCamera(CameraUpdateFactory.zoomIn());  
map.animateCamera(  
    CameraUpdateFactory.zoomTo(10), 2000, null);
```

◎ Kamera tetszőleges pozícióba

```
CameraPosition pos = new CameraPosition.Builder()  
    .target(new LatLng(47.533599, 19.033899))  
    .zoom(17)  
    .bearing(90)  
    .tilt(30)  
    .build();  
  
map.animateCamera(  
    CameraUpdateFactory.newCameraPosition(pos));
```

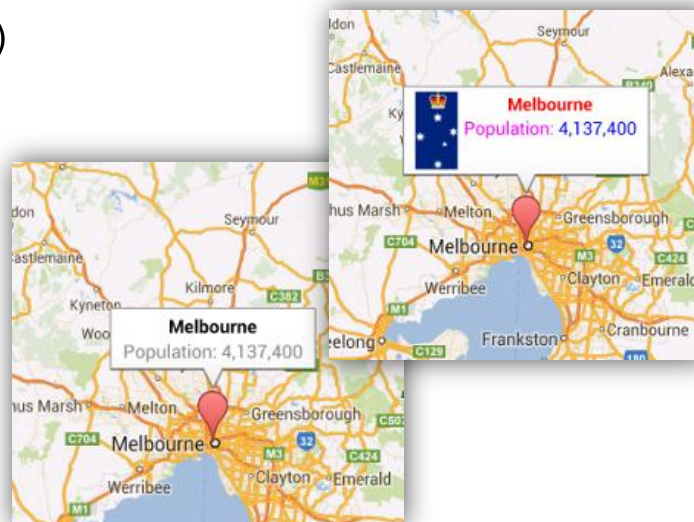

Markers, Pins

◎ Marker hozzáadása

```
Marker marker = map.addMarker(new MarkerOptions()  
    .position(new LatLng(  
        location.getLatitude(),  
        location.getLongitude()))  
    .title("Cím felirat")  
    .snippet("Részletező szöveg")  
    .icon(BitmapDescriptorFactory  
        .fromResource(icon))  
    .draggable(false)  
);
```

◎ Marker eltávolítása

```
marker.remove();
```



Rajzolás térképre

◎ Alakzatok: Polyline, Polygon, Circle

◎ Polyline

```
Polyline line = map.addPolyline(new PolylineOptions()  
    .add(new LatLng(37.35, -122.0))  
    .add(new LatLng(37.45, -122.0))  
    .add(new LatLng(37.45, -122.2))  
    ...  
    .width(25)  
    .color(Color.BLUE)  
);
```

◎ Polyline bővítése utólag

```
List<LatLng> points = line.getPoints();  
points.add(position);  
line.setPoints(points);
```



OpenStreetMap

mint alternatíva

OpenStreetMap (OSM)

Mi az OSM? – Közösségi térkép 😊

„Az OpenStreetMap egy szabadon szerkeszthető térkép az egész világról. Olyan emberek készítik, mint Te.

Az OpenStreetMap lehetővé teszi neked, hogy szabadon megtekintsd, szerkeszd és használd a földrajzi adatokat, bárhol is vagy a Földön. „ - www.openstreetmap.org

Miért lehet jó Android tekintetében?

- ⦿ Egy ingyenes alternatíva a Google Maps kiváltására.
- ⦿ Online és offline térképadatbázis, ∃ vektorgrafikus is.
- ⦿ Android támogatás - hasonló MapView osztály.
- ⦿ Nem kell API kulcs.
- ⦿ Nem lesz mindig teljesen ingyenes a Google Maps???

MapsForge

◎ Projektek: [MapsForge](#), [Osmand](#), [Osmdroid](#), ...

◎ MapsForge

- Projekt 2008-ban indult a Freie Universität Berlin számítógép-tudomány intézetében.
- Jelenleg ~10 diák és tudományos személyzet vesz részt benne.

◎ MapsForge által nyújtott eszközök:

- Térkép renderelés (offline)
- Útvonal és navigáció
- POI index és keresés
- Map rétegek, ...

◎ Térkép tekintetében:

- Online, csempék (tile) letöltése
- **Offline, renderelés**



[GTUG November 2010](#)

MapsForge - Online

Library letöltése

- [mapsforge-map-0.2.x.jar](#)

Hozzáadása

- Project properties / Java Build Path / Libraries / Add JARs

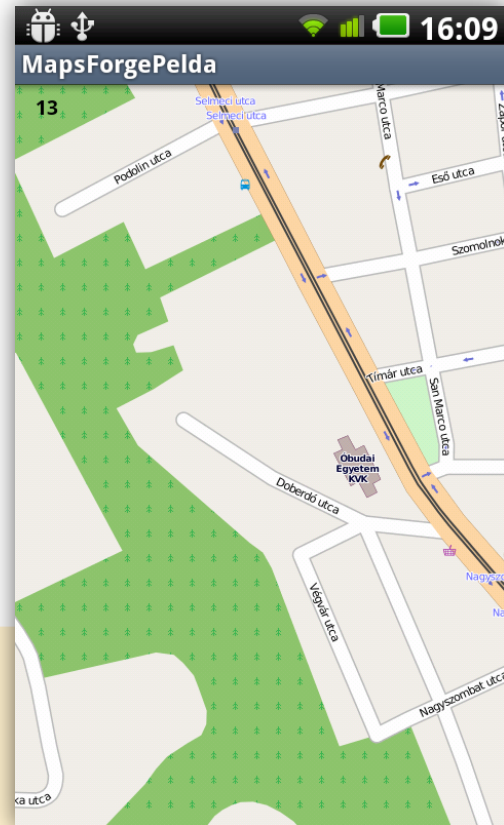
Java kód

```
MapView mapView = new MapView(this);  
mapView.setClickable(true);  
mapView.setBuiltInZoomControls(true);  
// mapView.setFpsCounter(true);  
mapView.setMapViewMode(MapViewMode.MAPNIK_TILE_DOWNLOAD);
```

```
MapController mc = mapView.getController();  
mc.setCenter(new GeoPoint(47.533571, 19.033851));  
mc.setZoom(17);
```

```
setContentview(mapView);
```

INTERNET
WRITE_EXTERNAL_STORAGE



MapsForge - Offline

☉ Térkép beszerzése (*.osm.pbf):

- <http://download.geofabrik.de/osm/>

☉ Térkép fájl létrehozása (*.map):

- <http://code.google.com/p/mapsforge/wiki/MapFileWriter>
[Osmosis](http://code.google.com/p/mapsforge/wiki/Osmosis)

☉ Térkép fájl átmásolása SD-kártyára

☉ Java kód módosítása

☺ Segítségül egy előre elkészített script:
[MapsForgeGenerate-v5](#)

```
MapView mapView = new MapView(this);
mapView.setClickable(true);
mapView.setBuiltInZoomControls(true);
mapView.setMapViewMode(MapViewMode.CANVAS_RENDERER);
mapView.setMapFile("/sdcard/budapest.map");
```

```
MapController mc = mapView.getController();
mc.setCenter(new GeoPoint(47.533571, 19.033851));
mc.setZoom(17);
```

INTERNET

WRITE_EXTERNAL_STORAGE

