

Android alkalmazásfejlesztés

Google Maps használata
OSM, mint alternatív térkép

OE-NIK

2012. október 13.

Sicz-Mesziár János

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



Google Maps

- ◎ 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 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

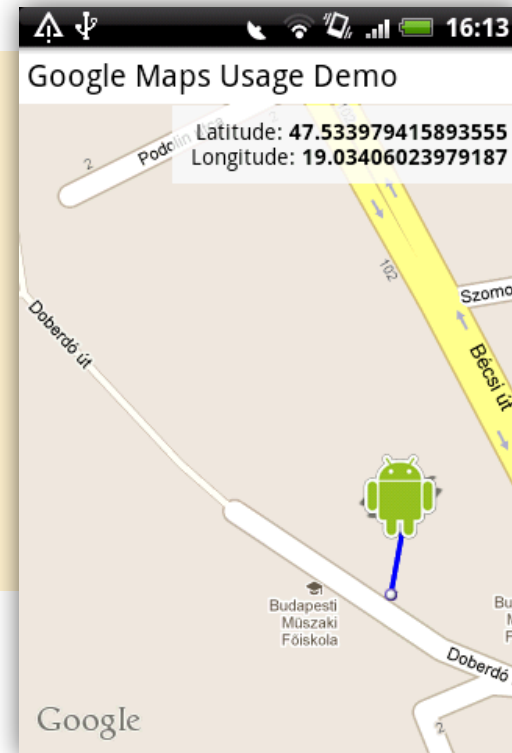
◉ Beépített +/- zoom engedélyezése:

```
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);
```

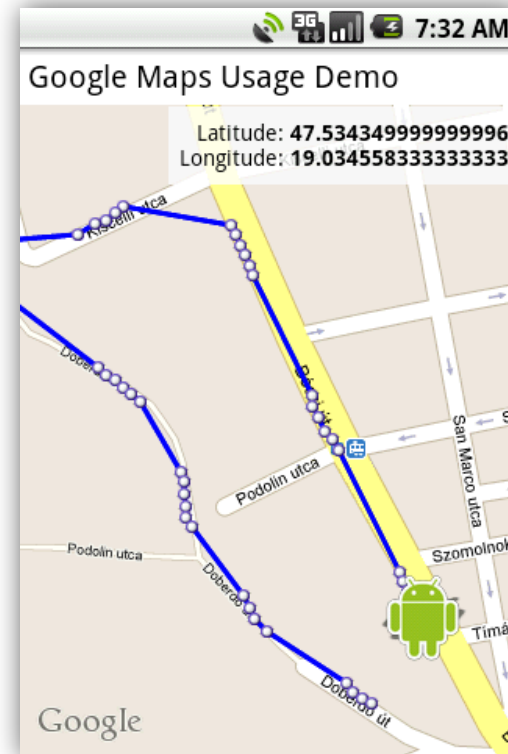
⊙ **Overlay feltöltése adatokkal**

```
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)  
);
```



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

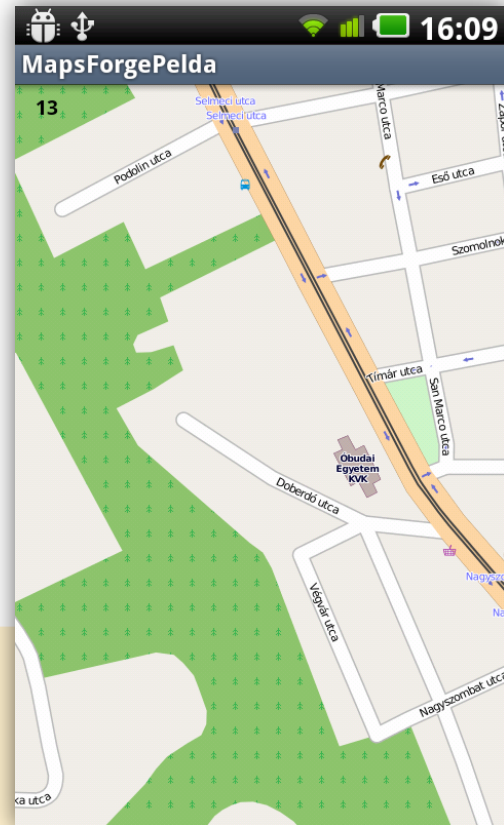
Java kód

```
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/MapFileWriter)

☉ Térkép fájl átmásolása SD-kártyára

☉ Java kód módosítása

☺ Segítségül egy előre elkészített script:
[MapsForgeGenerate-v4](#)

```
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

