

# TD/TP 3 : Android

## Géolocalisation

Romain Raveaux

### I. Objectif

A partir du TP précédent faire les points suivants :

**A°) Faire le suivi de sa propre position GPS.**

**B°) Positionner sur une carte les stations Yélo (Vé lo) de La Rochelle.**

### II°) GPS Tracking

A°) Il vous faut instancier un « LocationManager »

```
//---use the LocationManager class to obtain GPS locations---  
lm = (LocationManager) getSystemService(Context.LOCATION_SERVICE) ;
```

B°) Ensuite, il vous faut choisir le fournisseur(provider) de la géolocalisation

Par exemple, vous pouvez utiliser le GPS pour vous fournir votre positions :

```
String provider =lm.GPS_PROVIDER
```

Note : Il existe des méthodes beaucoup plus propres pour faire cela (getBestProviders)

C°) Un listener : Un listener écoute les changements d'état d'un composant. Dans notre cas, il nous faut un listener pour écouter les changement d'état de notre LocationManager.

Instanciation du listener de position :

```
locationListener = new MyLocationListener();
```

Association du listener de position avec avec le LocationManager

```
lm.requestLocationUpdates(  
    provider,  
    20000,  
    100,  
    locationListener);
```

Expliquer les paramètres de la méthode : requestLocationUpdates

## D°) Définition de la classe *MyLocationListener*

```
private class MyLocationListener implements LocationListener
{
    @Override
    public void onLocationChanged(Location loc) {
        if (loc != null) {

            Toast.makeText(getBaseContext(),
                "Location changed : Lat: " + loc.getLatitude() +
                " Lng: " + loc.getLongitude(),
                Toast.LENGTH_SHORT).show();

            p = new GeoPoint(
                (int) (loc.getLatitude() * 1E6),
                (int) (loc.getLongitude() * 1E6));

            mc.animateTo(p);

            mapView.invalidate();

        }
    }
}
```

To be continued...

Tester est faire fonctionner le GPS Tracking.

Utiliser **DDMS** pour tester votre GPS avec le fichier « Canal\_de\_Marans.gpx ». Ce fichier est sur Moodle (Cours PM4).

## III°) Signaler les stations Yélos et sa position GPS

Maintenant on vous demande d'afficher sur la carte votre position GPS courante ainsi que les stations Yélo.

Votre position sera représentée par un marqueur bleu et les stations Yélo par des marqueurs verts. Ces images de marqueurs sont placées sur moodle (Cours PM4).

Pour mener à bien ce TP, il vous faudra les coordonnées GPS des stations Yélo. Ces coordonnées sont sur Moodle (Cours PM4).

Les stations Yélo seront chargées une fois seulement lors de la création de l'application. Votre position GPS sera réactualisée à chaque « onLocationChanged »

A°) Il faut charger les ressources.

Dans notre cas, nous utiliserons deux ressources, les deux images de marqueurs.

Déterminer dans quel répertoire placer vos ressources ?

Charger les ressources :

```
bluepin = BitmapFactory.decodeResource(  
    getResources(), R.drawable.bluepin);  
greenpin = BitmapFactory.decodeResource(  
    getResources(), R.drawable.greenpin);
```

B°) Créer un tableau de points GPS

```
stations = new ArrayList<GeoPoint>();
```

C°) Ajouter tous les points « Yélo » dans ce tableau

D°) Les marqueurs Yélo doivent être sur-imprimés sur la carte (OverLay en anglais)

Créer une classe StationOverlay qui héritera de com.google.android.maps.Overlay

E°) Dans la méthode « Draw » de cette classe

```
@Override  
    public boolean draw(Canvas canvas, MapView mapView,  
        boolean shadow, long when)  
    {  
        super.draw(canvas, mapView, shadow);  
  
        return true;  
    }  
  
        //---translate the GeoPoint to screen pixels---  
Point screenPts = new Point();  
  
        //---add the marker---  
  
for(GeoPoint pi: stations){  
    mapView.getProjection().toPixels(pi, screenPts);  
    canvas.drawBitmap(greenpin, screenPts.x, screenPts.y-32,  
null);  
}
```

Que fait cette méthode ?

F°) Associer la classe StationOverlay à la MapView

```
listOfOverlays = mapView.getOverlays();
```

```
listOfOverlays.add(mapOverlayim);
```

```
StationOverlay mapOverlayim = new StationOverlay();  
listOfOverlays.add(mapOverlayim);
```

Attention : Les stations Yélo seront chargées une fois seulement lors de la création de l'application.

Tester sur l'émulateur que tout fonctionne.

G°) Créer la classe MeOverlay qui affiche votre position courante avec le marqueur bleu. MeOverlay sera instancié et ajouté à `listOfOverlays` lors de l'appel de la fonction `onLocationChanged`.

Attention : supprimer la position précédente du `listOfOverlays` pour ne pas surcharger la mémoire.

H°) Tester sur l'émulateur que tout fonctionne.