
opengeophone Documentation

Version 1.0.0

openMairie

28 June 2013

Table des matières

1	Manuel de l'utilisateur	3
1.1	installation	3
1.2	Utilisation	4
2	Description technique du client	13
2.1	Installation d'eclipse avec android	13
2.2	Principes de fonctionnement de la librairie cordova	13
2.3	Installation des sources du projet openGeoPhone sous eclipse	13
2.4	Ajouter les librairies CORDOVA	17
2.5	Construire le chemin vers libs/cordova-x.x.x.jar	17
2.6	les droits et le paramétrage de l'application	21
2.7	L'appel à l'application html/js et menu	22
3	Description technique du serveur	25
3.1	Installation sur le serveur	25
3.2	Cartographie	25
4	Bibliographie	29
5	Contributeurs	31

Note : Cette création est mise à disposition selon le Contrat Paternité-Partage des Conditions Initiales à l'Identique 2.0 France disponible en ligne <http://creativecommons.org/licenses/by-sa/2.0/fr/> ou par courrier postal à Creative Commons, 171 Second Street, Suite 300, San Francisco, California 94105, USA.

openGeoPhone a pour objet de proposer l'implémentation d'une base d'appel sous postgresql (avec postgis) ou sous mysql depuis un smartphone.

L'objectif est de proposer aux agents en mobilité d'une collectivité territoriale (police municipale, services techniques) un outil de collecte de données géolocalisées permettant de gérer des événements sur la voie publique : affichage publicitaire, occupation du domaine public, problèmes de circulation, incivilités ...

Il s'agit de récupérer la photo de l'événement concerné, une géolocalisation ainsi que les identifiants du smartphone et de les traiter dans une application (à implémenter avec openMairie)

Pour se faire, il est créé coté client une application ANDROID dont l'objectif est de :

- catégoriser et identifier par un texte l'événement
- transmettre une photo préalablement allégée pour un réseau mobile
- transmettre la géolocalisation de l'événement
- synchroniser une transmission avec une base locale (sqlite) en cas de perte de réseau ou en cas de non réponse du serveur

Du côté serveur, il est créé :

- les scripts permettant de télécharger la photographie et d'insérer les données transmises par le client android
- un script de flux geo rss permettant de diffuser les informations
- un script de cartographie permettant de géolocaliser les appels (en coordonnées géographiques)
- dans une version ultérieure, il sera proposé une application "openMairie" de traitement des appels

Les librairies utilisées sont les suivantes :

- côté client : cordova, jquery mobile
 - coté serveur : openlayers
-

Note : La librairie cordova permet d'étendre les scripts clients à d'autres systèmes d'exploitation : iPhone, BlackBerry, windows phone ...

la librairie CORDOVA permet un accès client aux fonctionnalités du smartphone avec du javascript.

Ce document a pour but de guider les utilisateurs et les développeurs dans la prise en main du projet.

Bonne lecture et n'hésitez pas à nous faire part de vos remarques à l'adresse suivante : contact@openmairie.org !

Manuel de l'utilisateur

Il est proposé de décrire l'utilisation de l'application ANDROID dans ce chapitre.

1.1 Installation

1.1.1 Téléchargement sur le site de l'adullact

L'application android est téléchargeable à l'adresse suivante (opengeophone.apk) :

http://adullact.net/frs/?group_id=810

1.1.2 Installation de l'application

L'application n'étant pas téléchargée sur le Play Store, vous devez autoriser l'installation d'application de sources différentes que le Play Store.

Pour ce faire, il suffit de cocher la case dans Paramètres - Sécurité - Sources inconnues.

Puis aller dans le répertoire de téléchargement, double cliquer sur Opegeophone.apk et lancer l'installation en choisissant le mode installeur.

1.1.3 Configuration du serveur d'appel

Lors de la première utilisation aucun serveur d'appel n'est configuré.

Si vous tentez de faire une saisie, vous aurez un message d'erreur.

Pour configurer ce lien il vous faudra cliquer sur le bouton "configuration" situé en haut à droite de l'application.

Vous serez ensuite dirigé sur une page contenant un formulaire, qui vous permettra de saisir le lien du site d'appel, qui vous sera communiqué par l'administrateur de l'application.

Le lien doit obligatoirement commencer par http et finir par /.

Si le lien est incorrect, les appels seront sauvegardés sur le mobile !



The screenshot shows the OpenGeoPhone application interface. At the top, there is a dark header with the text "OpenGeoPhone" on the left and a "Retour" button on the right. Below the header, the text "Lien de votre serveur distant :" is displayed above a large, empty rounded rectangular input field. Underneath the input field, there is instructional text: "Commençant par http:// et finissant par /" and "Exemple: http://extranet.arles.fr/". Below this text is a rounded rectangular button labeled "Mettre à jour". At the bottom of the form, the text "Lien du serveur actuel :" is followed by the URL "http://extranet.arles.fr/opengeophone/".

L'application est opérationnelle !

1.2 Utilisation

L'application possède 2 modes de connexions et un mode de synchronisation :

- Le mode connecté est utilisé en cas de procédure normale.
- Le mode déconnecté est utilisé lorsque l'application n'arrive pas à joindre le serveur distant.
 - Dans ce cas les datas sont sauvegardées sur votre mobile et seront transférées au serveur lorsque vous ferez une synchronisation.
- La synchronisation permet d'envoyer au serveur les données stockées sur le mobile lors du mode déconnecté.

1.2.1 Mode connecté

Voici les étapes du mode connecté :

N'oubliez pas d'activer votre GPS !

Etape 1 :

- Saisie du formulaire
- Cliquez sur le bouton Valider



OpenGeoPhone Configuration

Catégorie :

Tags

Description de la situation :

Tag

Valider

Synchronisation

Etape 2 :

- Géolocalisation (Si le mobile ne trouve pas votre position au bout de 15s, la procédure est arrêtée. Retour à l'étape 1).



Si la précision de géolocalisation est supérieure à 100m, un message sera affiché mais n'empêchera pas le fonctionnement de l'application.

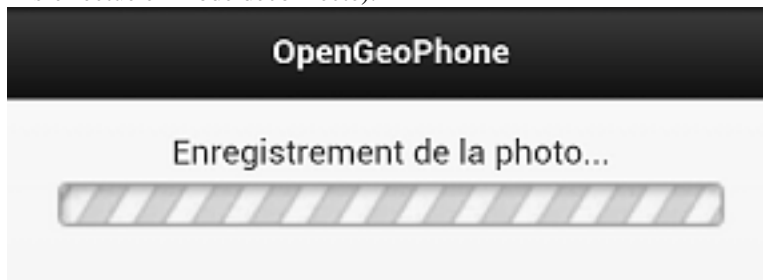
Etape 3 :

- Capture de la photo via l'appareil photo par défaut du mobile



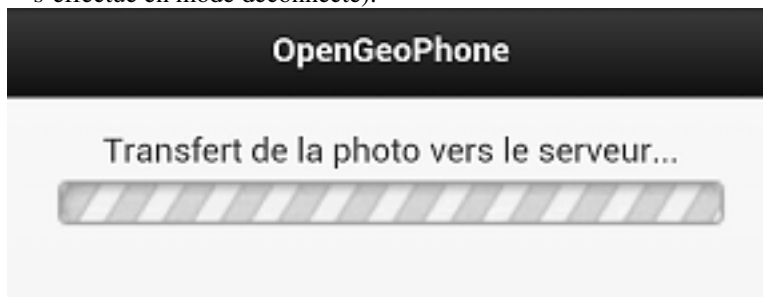
Etape 4 :

- Enregistrement sur le mobile et renommage de la photo (Si le serveur ne répond pas au bout de 10s, la sauvegarde s'effectue en mode déconnecté).



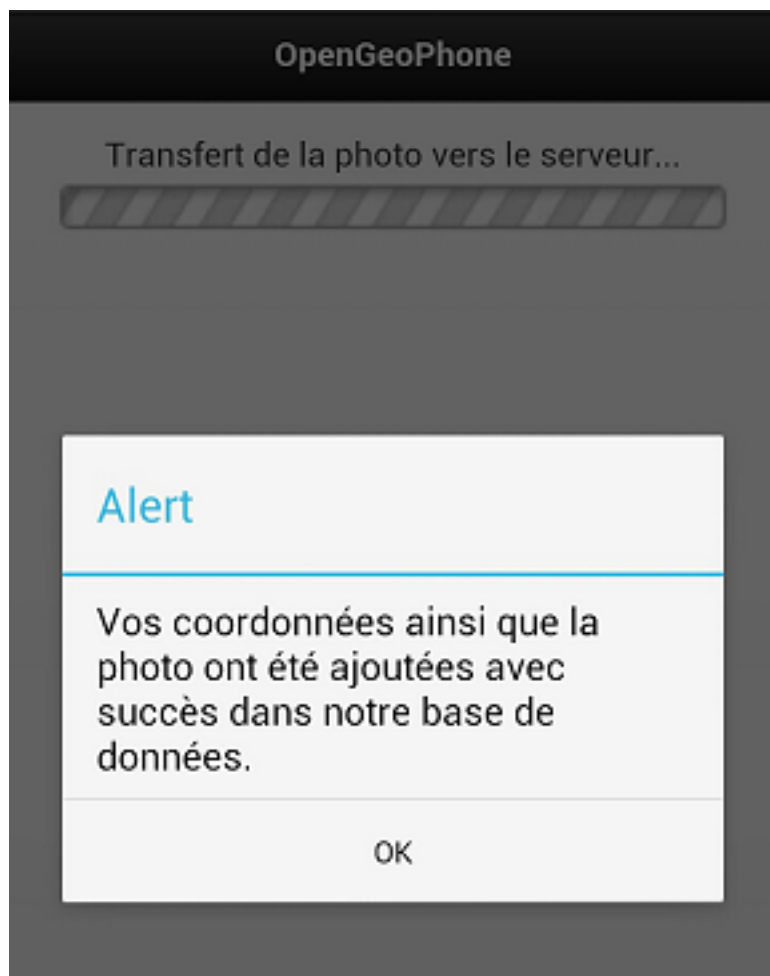
Etape 5 :

- Transfert de la photo et des données sur le serveur (Si le serveur ne répond pas au bout de 10s, la sauvegarde s'effectue en mode déconnecté).



Etape 6 :

- Procédure normale terminée (Retour à l'étape 1)



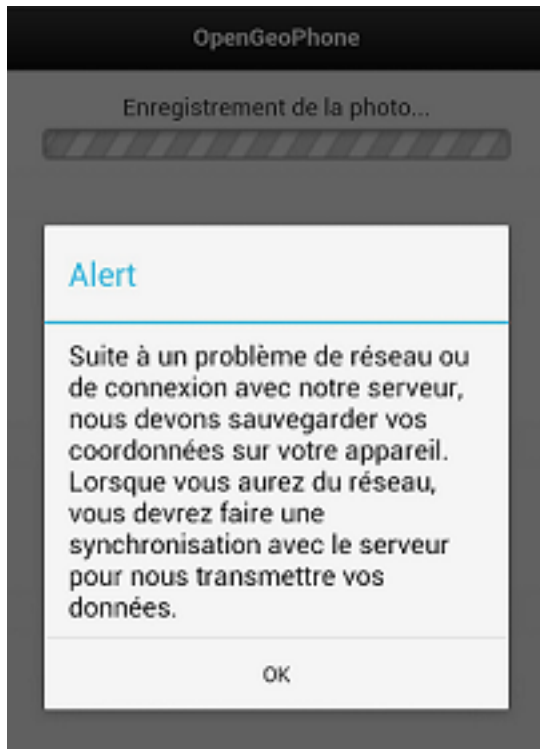
1.2.2 Mode déconnecté

Voici les étapes du mode déconnecté :

Dans le cas ou le serveur n'a pas répondu au bout de 10s pendant les étapes 4 ou 5.

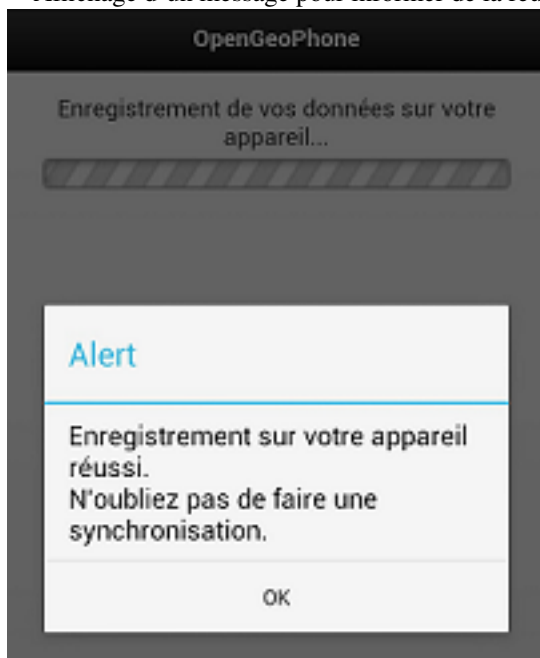
Etape 1 :

- Affichage d'un message pour informer l'utilisation du mode déconnecté



Etape 2 :

- Affichage d'un message pour informer de la réussite de l'enregistrement (Retour à l'étape 1 du mode connecté).



1.2.3 Synchronisation

Voici les étapes de la synchronisation :

Après avoir effectué des enregistrements en mode déconnecté, vous devez synchroniser l'application avec le serveur.

Etape 1 :

- Cliquez sur le bouton Synchronisation



OpenGeoPhone Configuration

Catégorie :
Autre

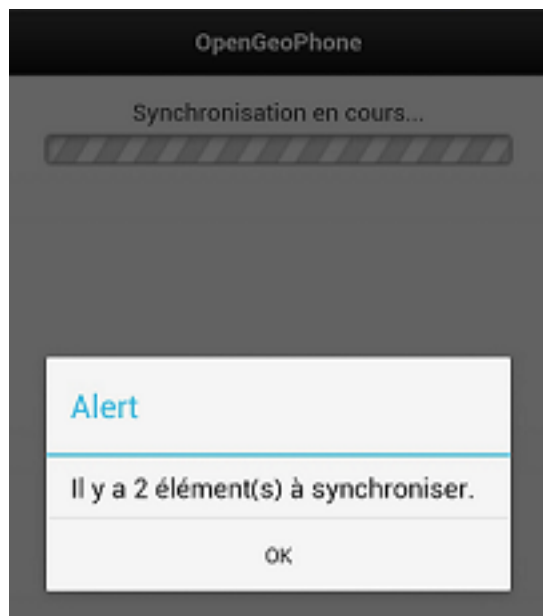
Description de la situation :

Valider

Synchronisation

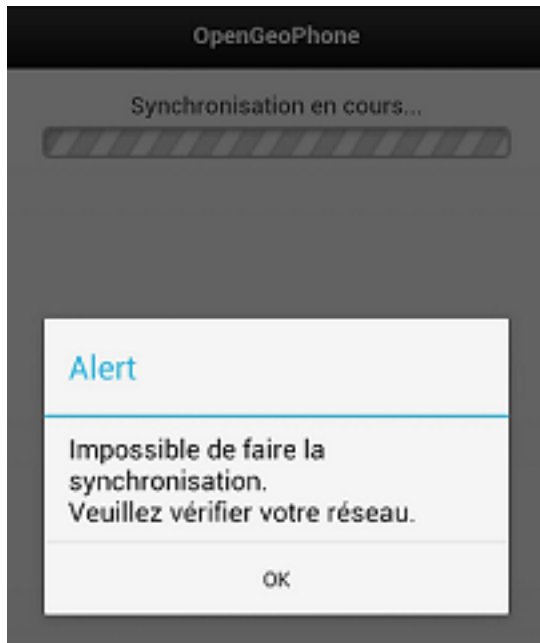
Etape 2 :

- Affichage d'un message pour vous informer du nombre de synchronisation (Si il n'y a rien à synchroniser un message sera afficher).



Etape 3 :

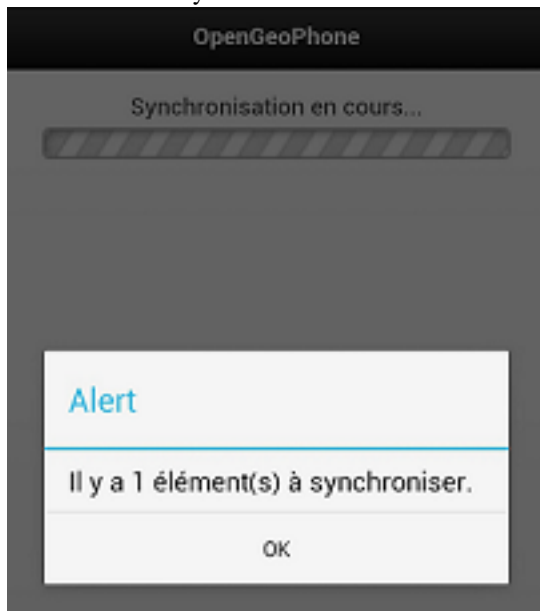
- Execution des étapes 4, 5 et 6 du mode connecté
- Si le serveur ne répond toujours pas aux étapes 4, 5, ou 6



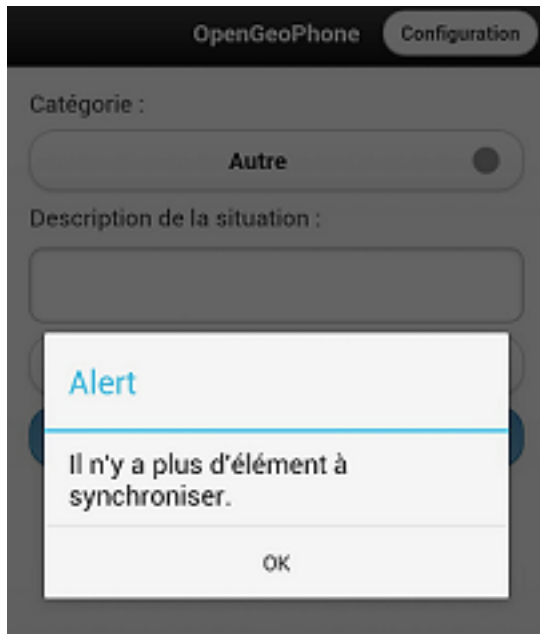
Etape 4 :

Synchronisation suivante

– Si il reste des synchronisations



– Si il n'y a plus rien à synchroniser



Description technique du client

Il est proposé une analyse technique de l'application client (pour développeur) :

- l'installation du projet sous eclipse pour éventuellement le modifier
- la description technique de l'application

Il est présenté ci-dessous l'installation des sources du projet sur eclipse

2.1 Installation d'eclipse avec android

Note : pre requis, il faut installer :

- eclipse
 - sdk android
 - cordova (ex phonegap)
-

Sous Windows télécharger l'android developer tools pour eclipse et sdk android :

<http://developer.android.com/sdk/index.html>

La librairie cordova est téléchargeable au lien :

<http://phonegap.com>

2.2 Principes de fonctionnement de la librairie cordova

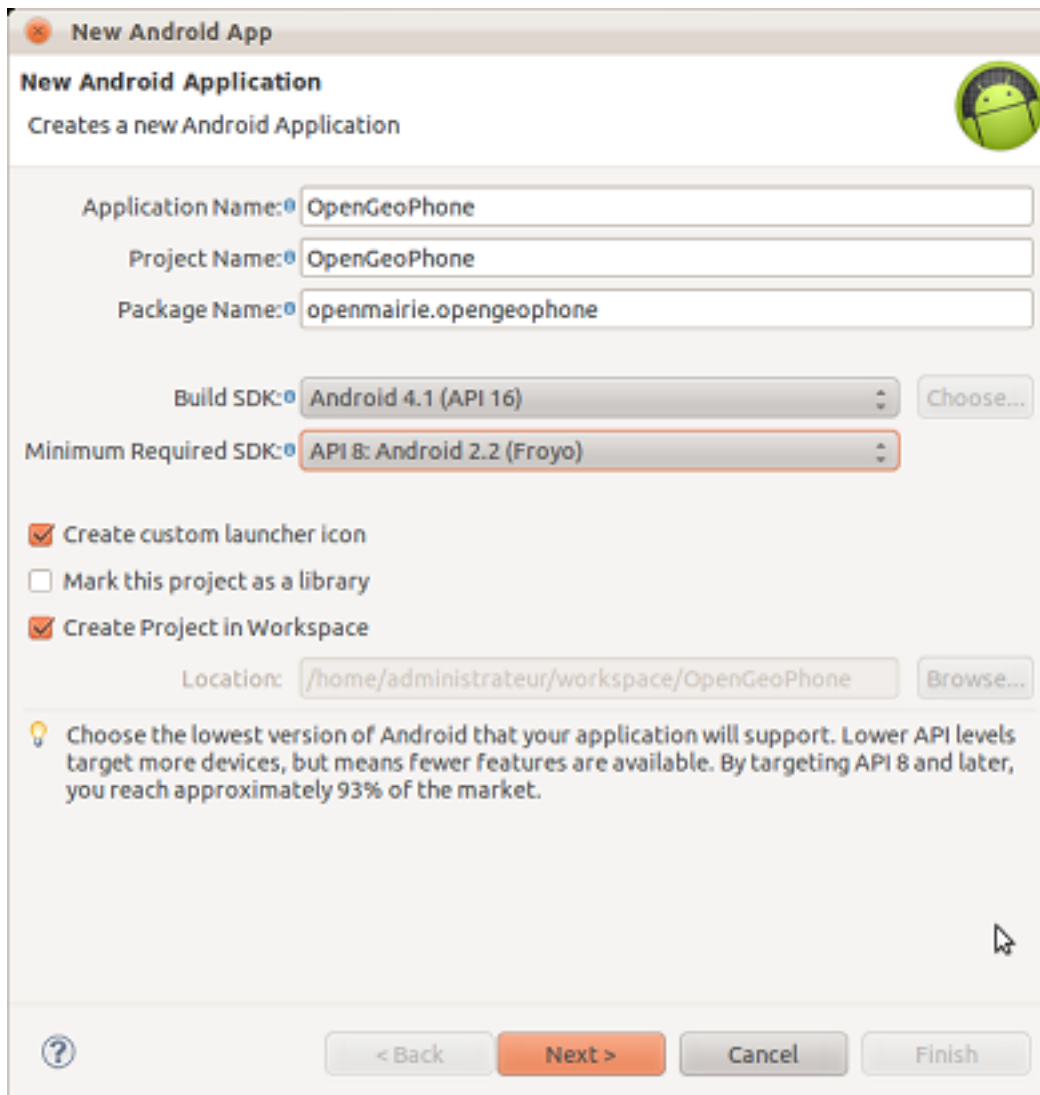
La librairie cordova permet de développer une application html/js encapsulée dans une application mobile.

En principe les scripts sont applicables aux autres systèmes d'exploitation supportés par cordova.

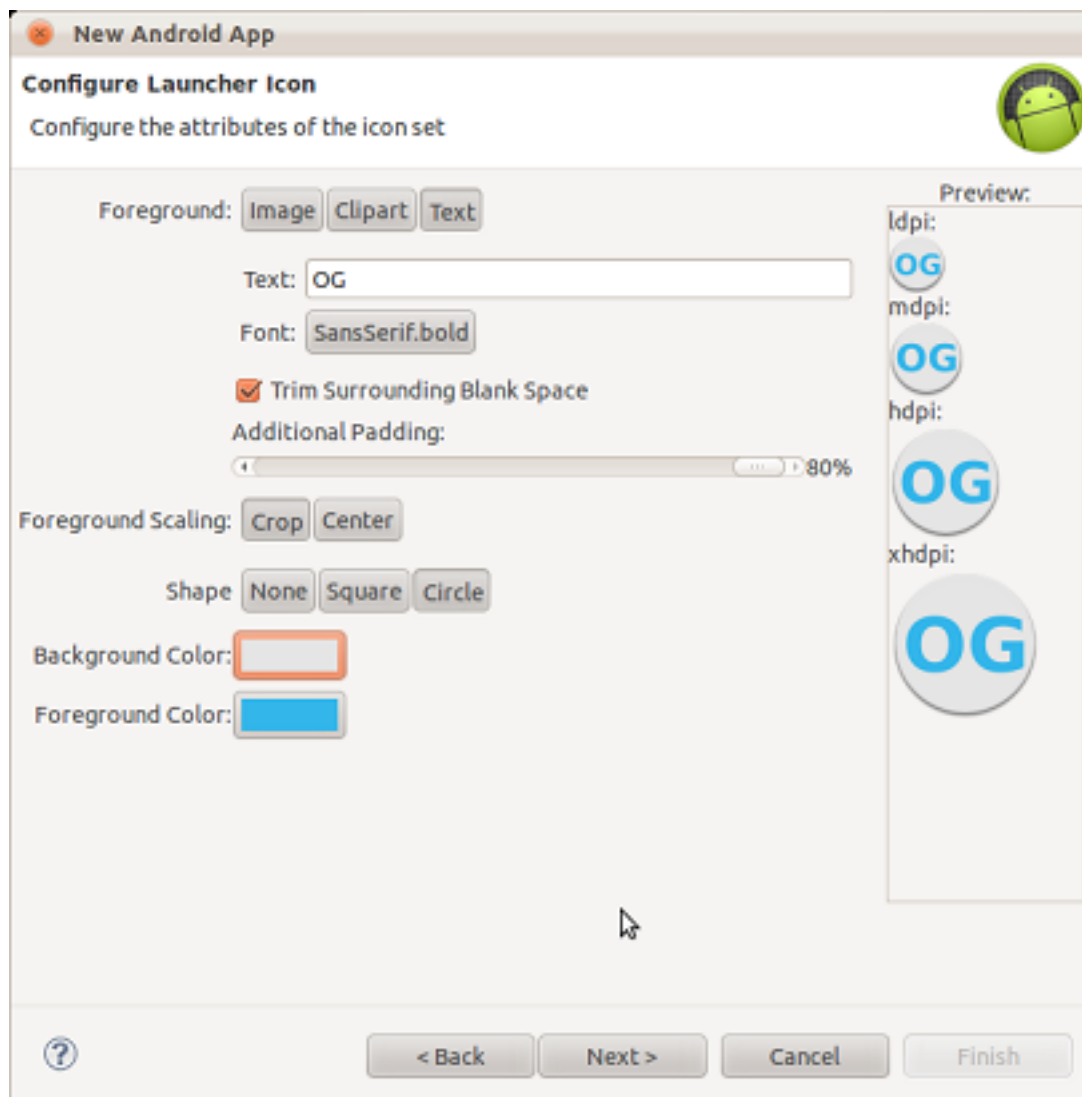
<http://docs.phonegap.com/>

2.3 Installation des sources du projet openGeoPhone sous eclipse

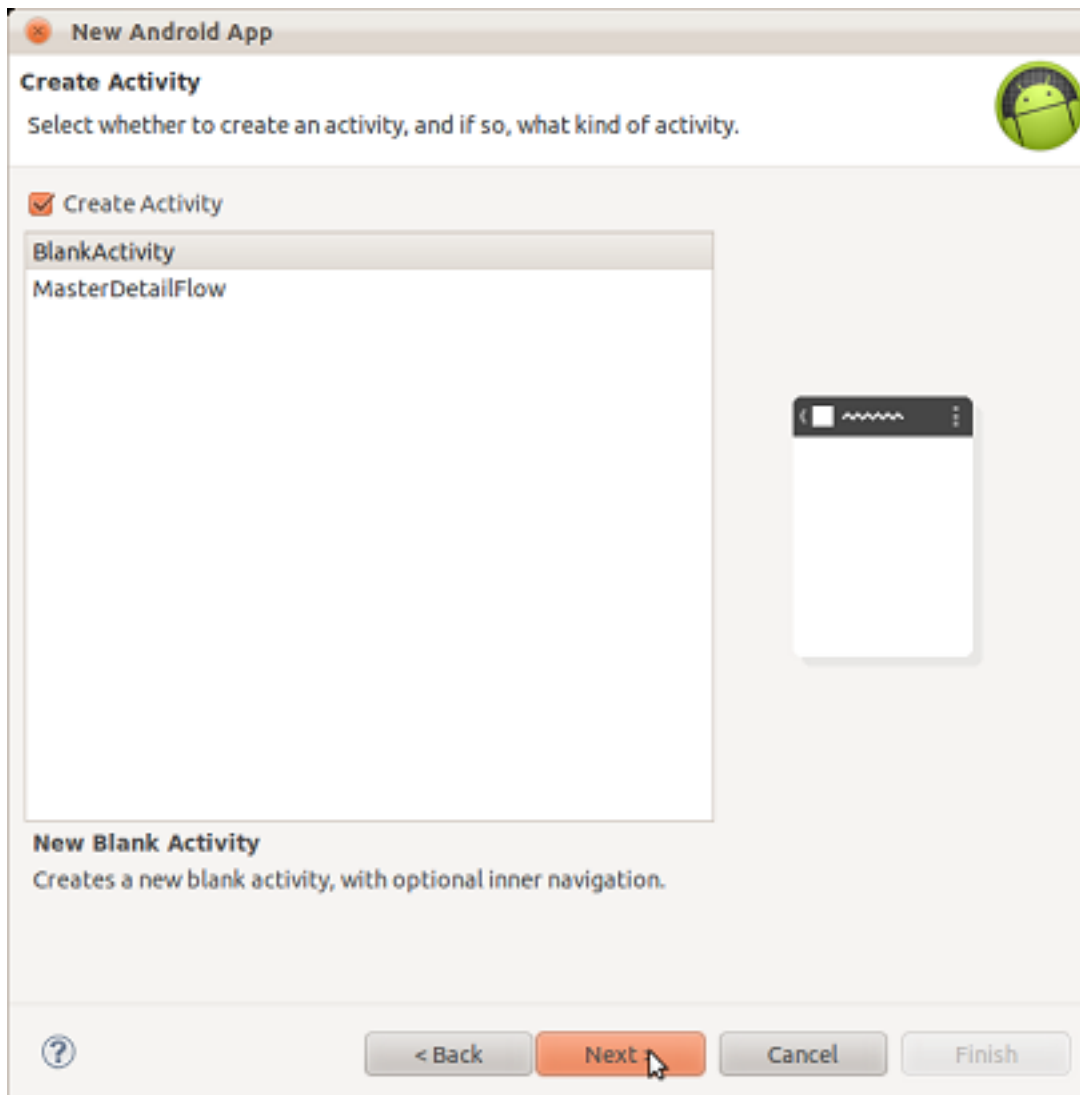
Créer le projet Android



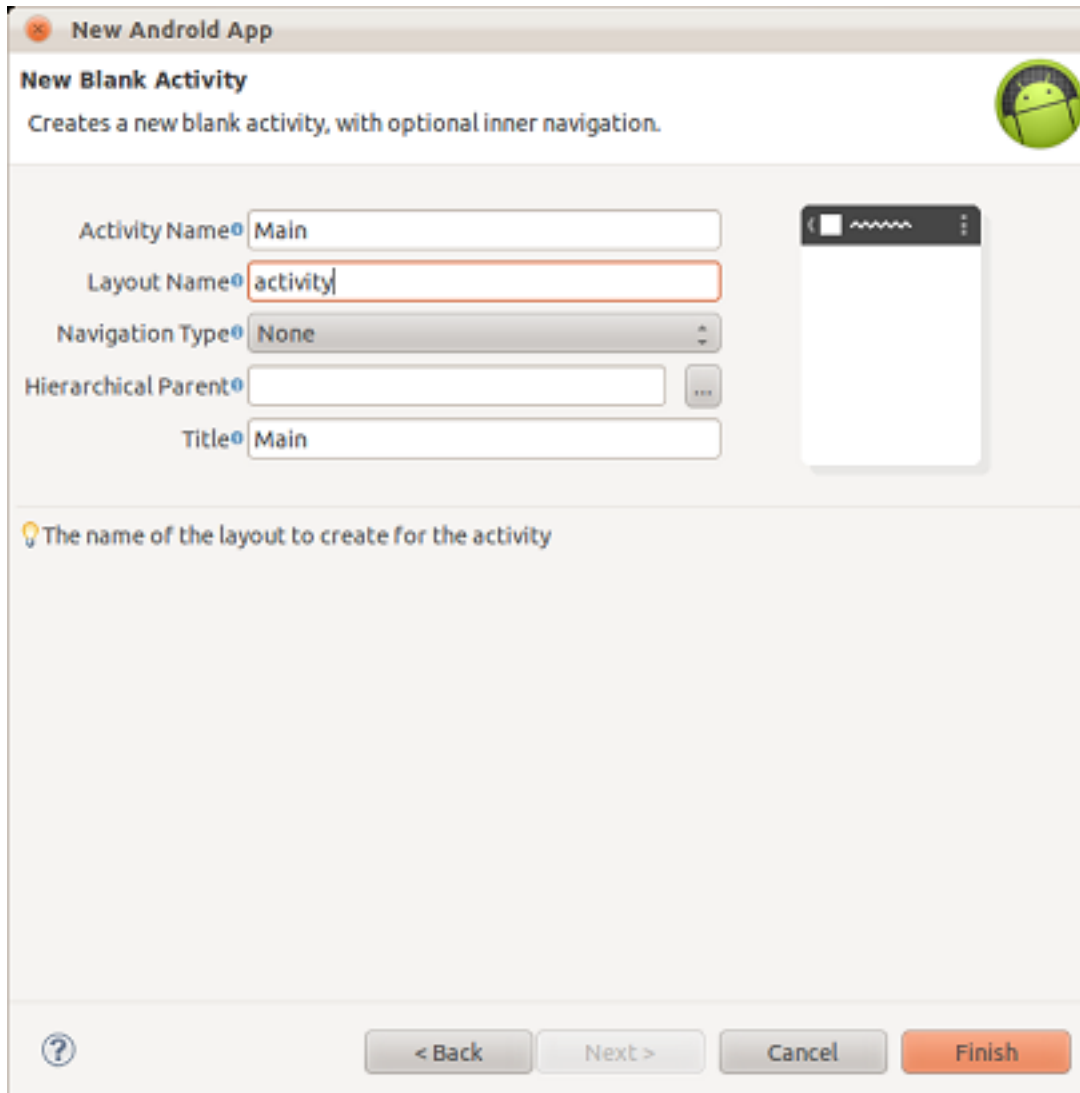
Commencer le nom du projet par une majuscule



Création de l'icône en "text"



Appuyer sur “next”



Modification du nom : activity et layout. Appuyer sur “Finish”

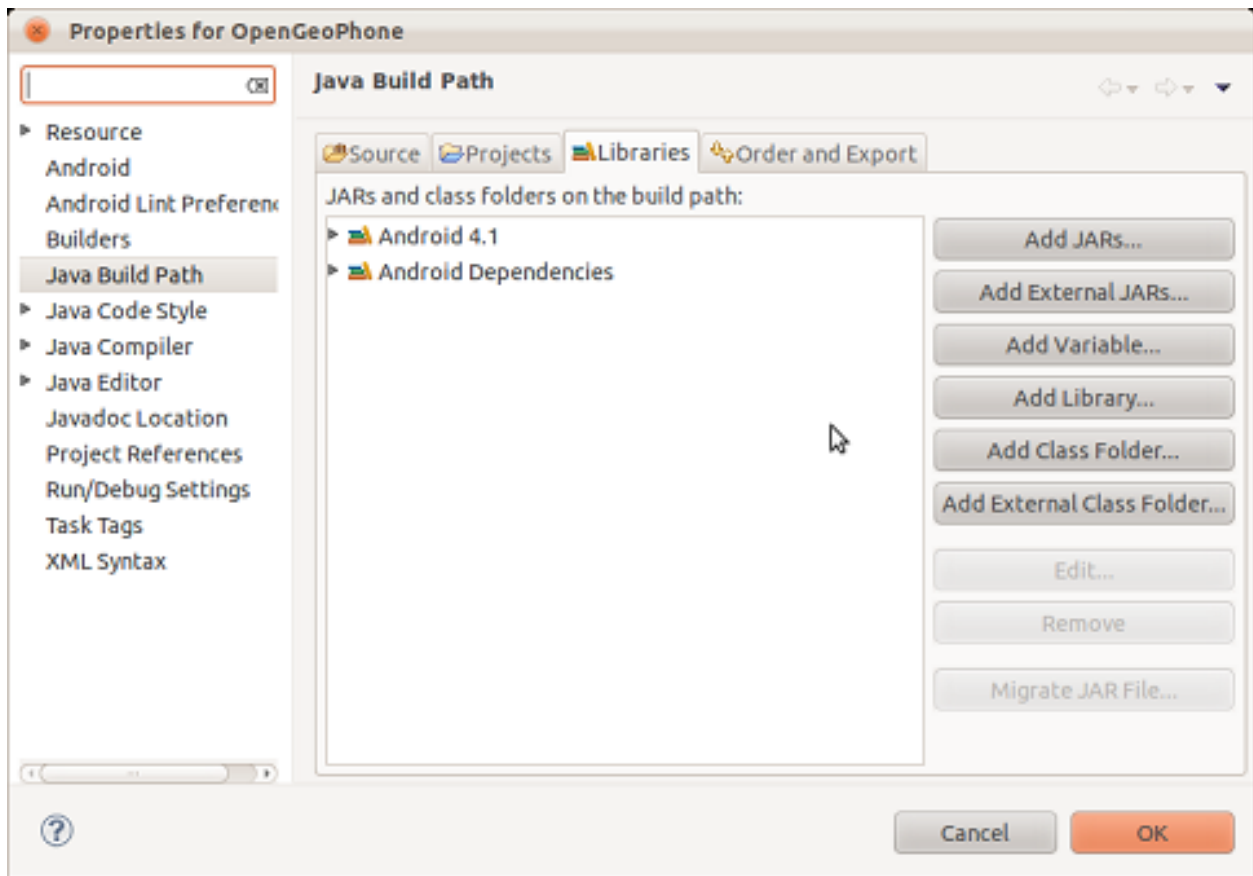
2.4 Ajouter les librairies CORDOVA

Il faut ajouter la librairie cordova-x.x.x.jar dans le projet ainsi que le repertoire res/xml :

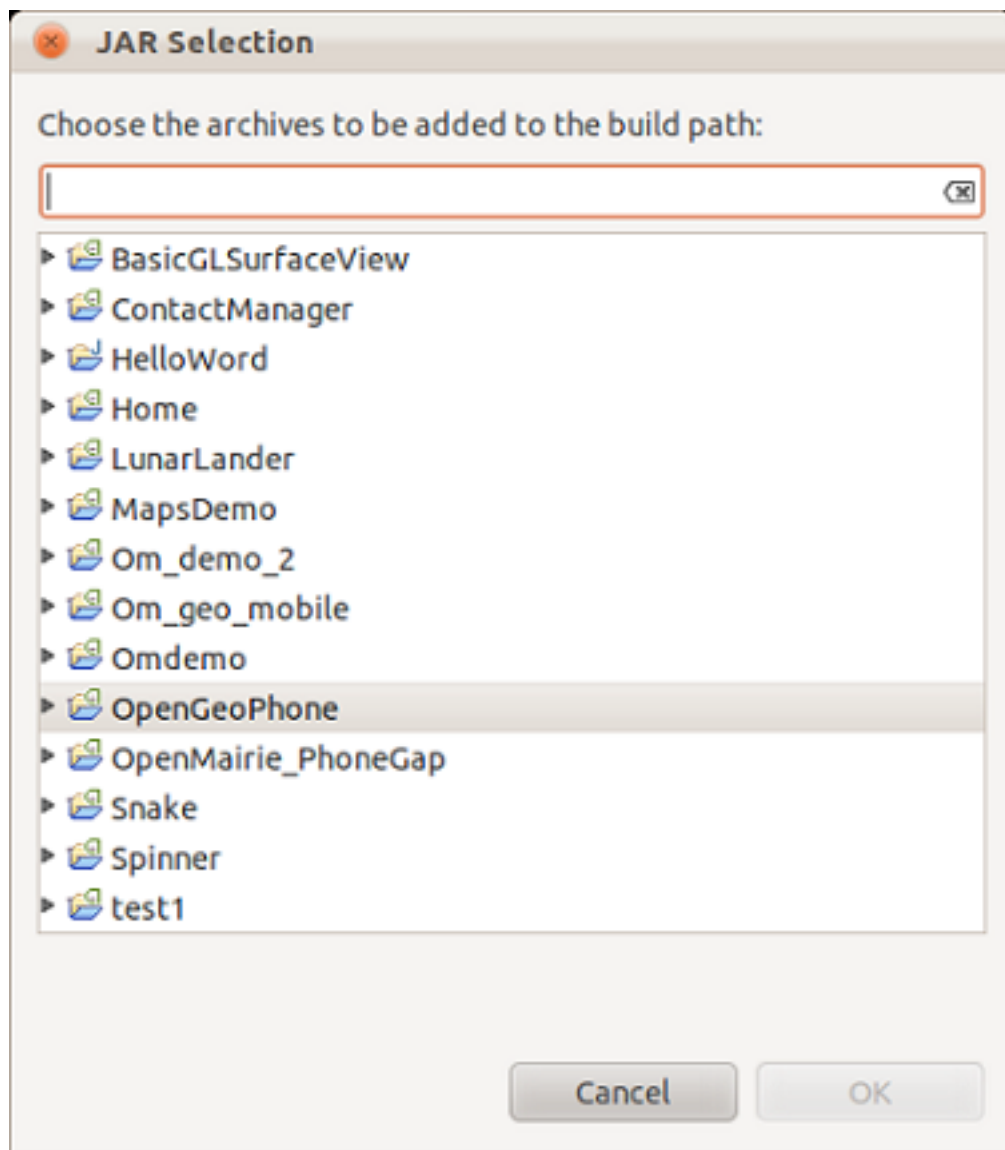
- res/xml : adresse des plugins cordova
- libs/cordova-x.x.x.jar

2.5 Construire le chemin vers libs/cordova-x.x.x.jar

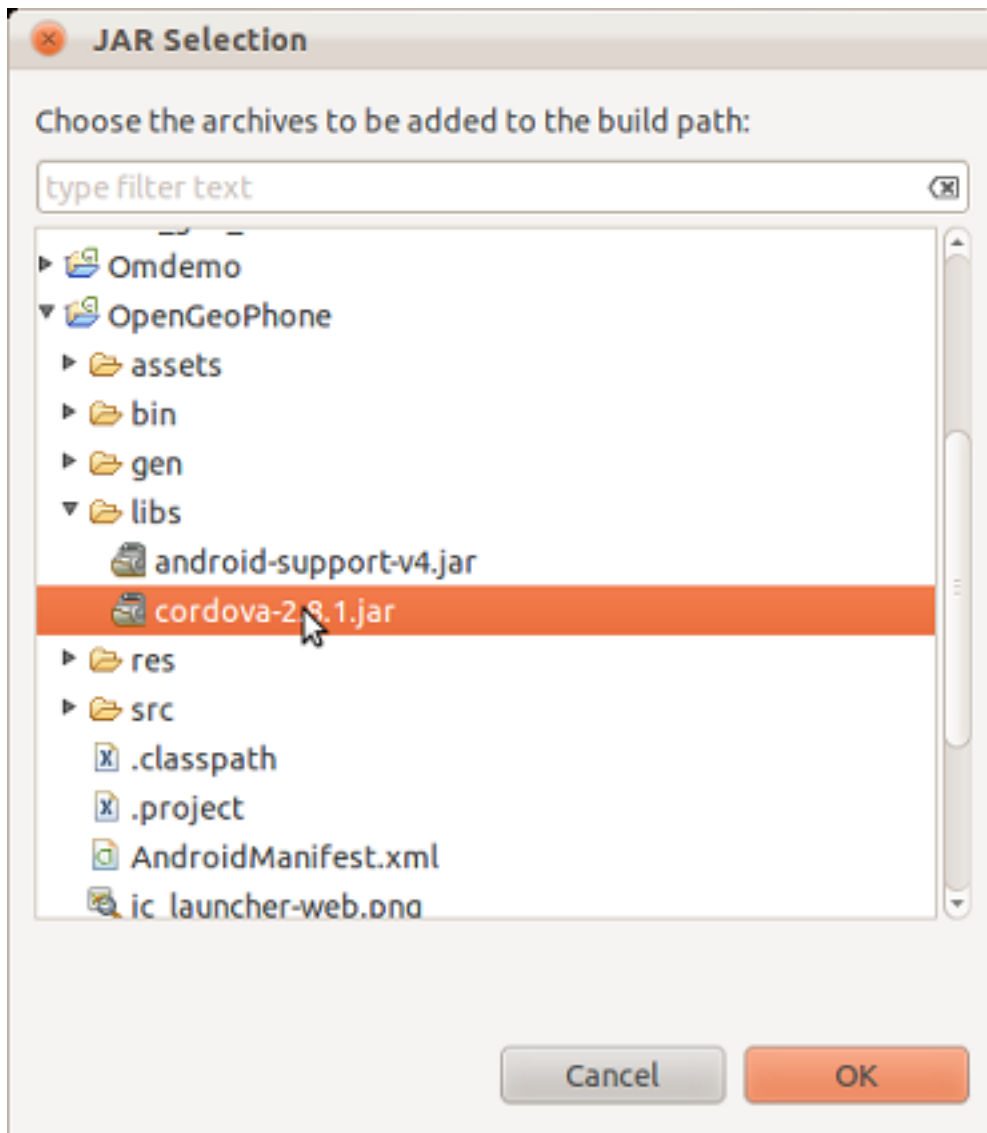
Sur le repertoire libs (sous eclmipse), faire un clic droit et choisir “build path”, puis “configure build path”



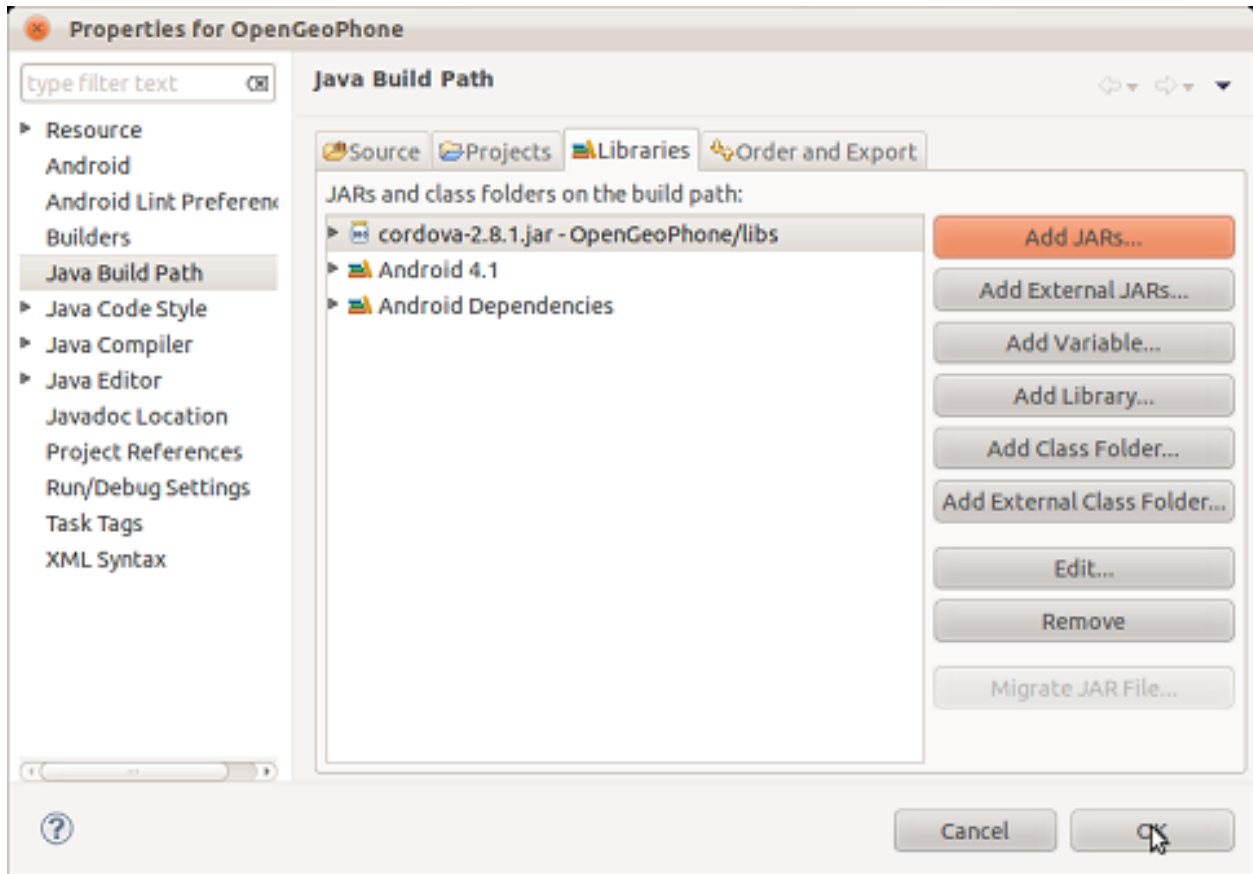
Aller sur l'onglet "libraries" et Appuyer sur "add jars"



Choisir OpenGeoPhone



Choisir libs/cordova-x.x.x.jar



Cliquer sur OK

2.6 les droits et le paramétrage de l'application

Il s'agit de créer les droits de fonctionnement sur le téléphone.

Copier dans le fichier AndroidManifest.xml :

Droits

```
<supports-screens
    android:largeScreens="true"
    android:normalScreens="true"
    android:smallScreens="true"
    android:resizeable="true"
    android:anyDensity="true" />
<uses-permission android:name="android.permission.VIBRATE" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_LOCATION_EXTRA_COMMANDS" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.BROADCAST_STICKY" />
```

Le lien suivant permet d'avoir le détail des permissions :

<http://developer.android.com/reference/android/Manifest.permission.html>

L'Orientation de l'écran doit être uniquement en paysage (problème de rechargement de l'application en cas de changement d'orientation)

```
android:screenOrientation="portrait "
```

éventuellement "landscape"

le nom de l'application doit s'afficher dans le tableau de bord

```
android:label="@string/app_name" >
```

2.7 L'appel à l'application html/js et menu

Le script java main.java dans src/openmairie/opengeophone doit être modifié pour :

- l'appel à la librairie cordova (import ...)
- la classe d'extension de MainActivity
- la ligne super.loadUrl(...)

Il faut aussi mettre les méthodes en "public"

Ci dessous, il est affiché le code de main.java dans src/openmairie/opengeophone

```
package openmairie.opengeophone;

import android.os.Bundle;
import android.view.Menu;
import org.apache.cordova.*; // package cordova

public class Main extends DroidGap { // méthode DroitGap

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        super.loadUrl("file:///android_asset/www/index.html"); // lien sur l'application html/js
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.activity, menu);
        return true;
    }
}
```

L'application est contenu dans le répertoire assets/www du projet

Elle contient :

- un fichier index.html qui propose une interface au client : formulaire de saisie data pour serveur, configuration, synchronisation
- un répertoire css et image qui contient les styles et thèmes jquery-mobile
- un répertoire js qui contient les scripts clients :
 - avancement.js : utilise les fonctions de géolocalisation, de capture d'image et de transfert de fichier et données sur le serveur
 - local.js : permet de stocker les datas en local sur sqlite
 - serveur.js : permet de configurer le lien du serveur distant

Le fichier cordova.x.x.x.js est dans le répertoire js et permet de faire le lien avec la librairie java cordova.x.x.x.jar
Ces deux librairies sont à adapter suivant le système du client utilisé

Description technique du serveur

Il est proposé de décrire dans ce chapitre l'installation sur le serveur (pour l'administrateur).

3.1 Installation sur le serveur

3.1.1 Le repertoire mobile

Les scripts d'integration dans la base de données sont dans le repertoire serveur/mobile

- rename_*sgbd*.php : renomme les images en fonction de la base de données
 - upload.php : rapatrie les images capturées sur le serveur
 - sauvegarde_*sgbd*.php : sauvegarde les datas dans la base de données
- les photos sont stockées dans le repertoire /mobile/images

3.1.2 Le paramétrage de la base de données

Dans mobile/config.php, configurer le sgbd mysql ou pgsq

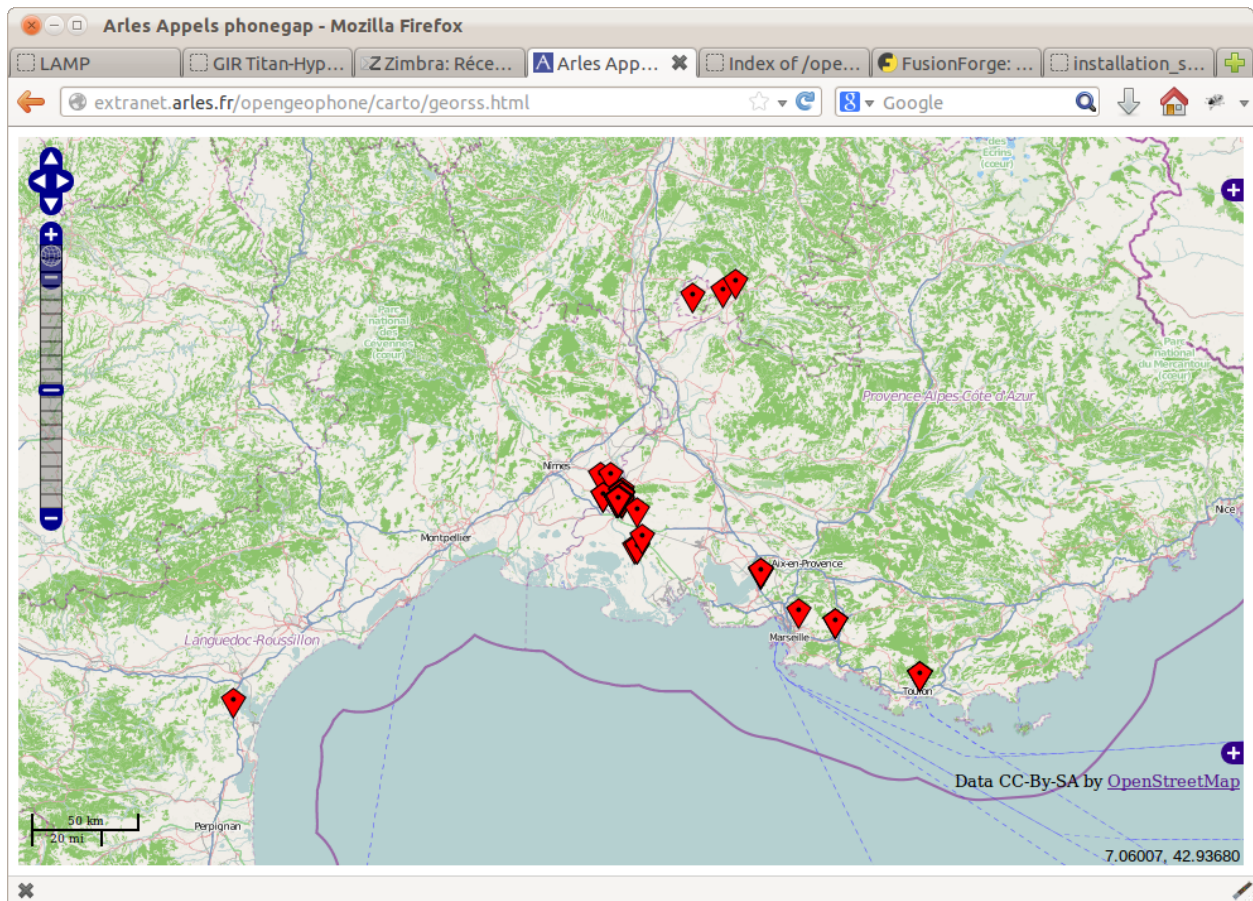
3.1.3 Créer la base de données :

Dans data/pgsql ou data/mysql : executer le script init_metier.sql

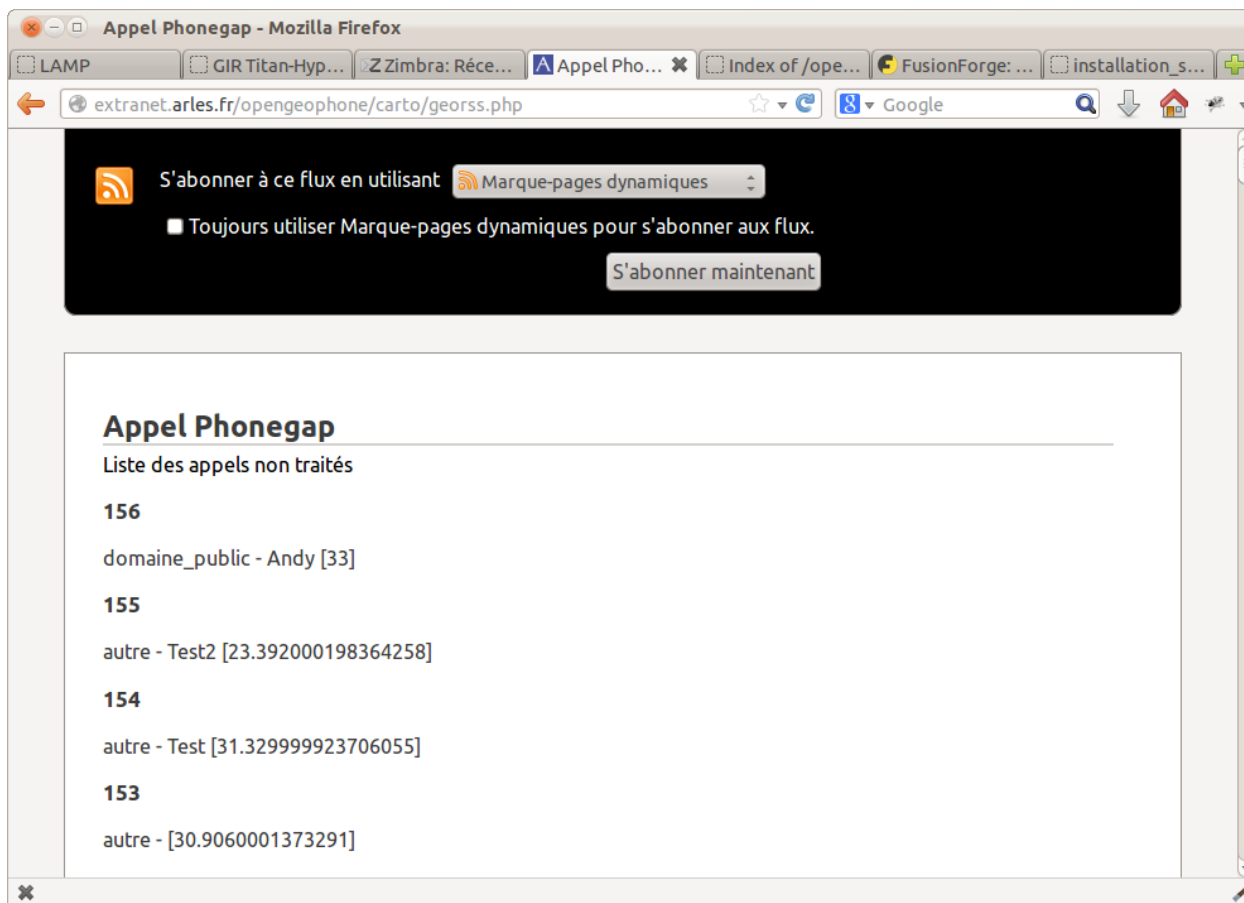
Pour postgresql, executer init_metier_sig.sql, votre base doit avoir l'extension postgis.

3.2 Cartographie

Dans le repertoire carto, il y a 2 scripts : georss.html et georss.php et la librairie OpenLayers.



- georss.html permet d'afficher les points d'appel sur fond open street map avec la librairie OpenLayers



– georss.php permet de fabriquer un flux georss disponible pour une autre application

Note : Vous devez avoir un serveur apache et une base de données installée (mysql ou postgresql)

Bibliographie

- <http://www.openmairie.org/telechargement/openMairie-Guidedudveloppeur.pdf/view>
- <http://developer.android.com/sdk/index.html>
- <http://phonegap.com>
- <http://docs.phonegap.com/>

Contributeurs

(par ordre alphabétique)

- Andy Rabadan
- François Raynaud