



Documentation interface Java

Projet d'intégration 2015

Anthony Borghi

v1.0

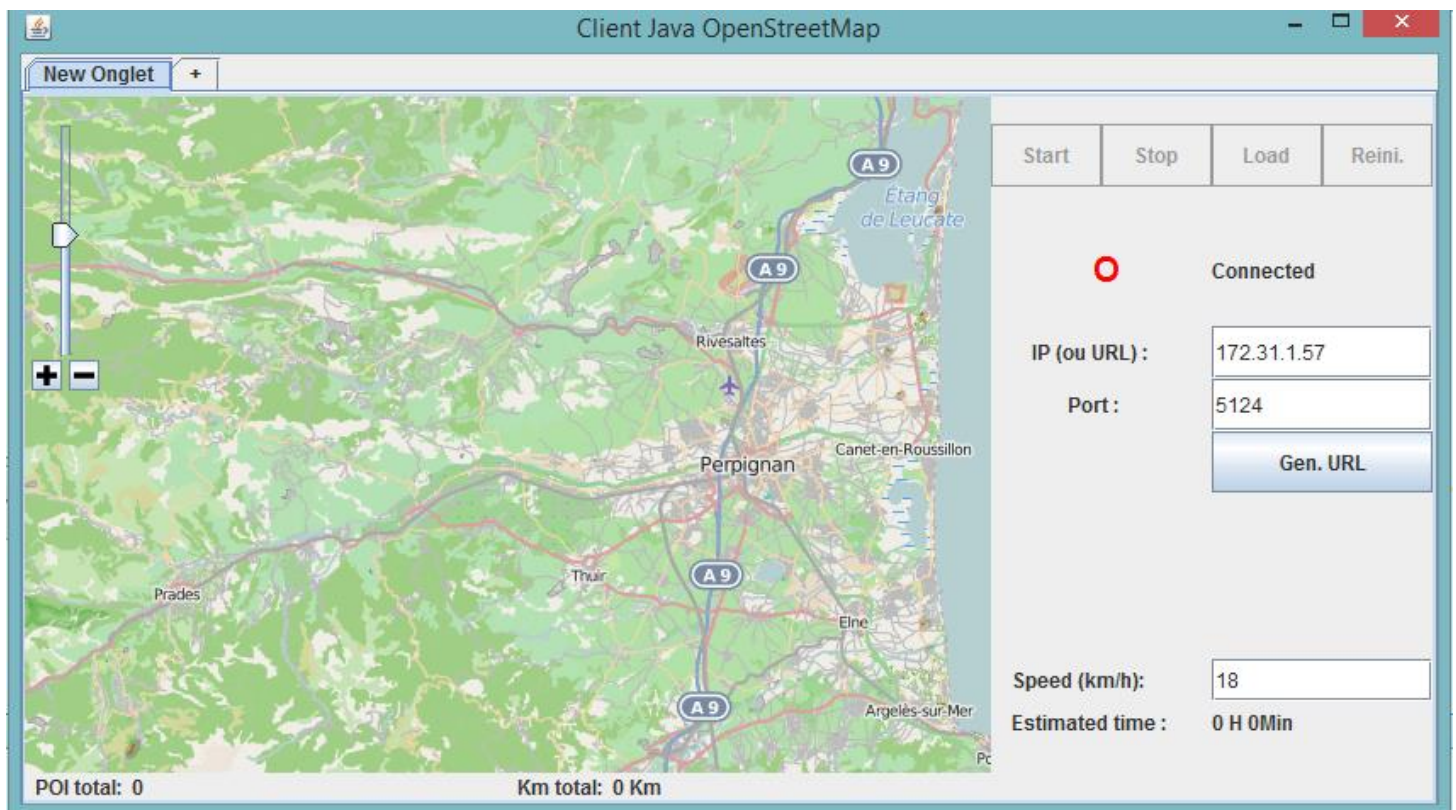
Sommaire :

I-Présentation.....	3
II-Interface	3
1-Les Champs.....	3
2-Les Boutons.....	4
3-Les Labels textes.....	4
4-Interactions avec la Carte.....	4
III-Problèmes rencontrés.....	5
1-Fermeture onglet	5
2-Traitement KML.....	5
3-Mise en cache de la carte.....	5
4-Déplacement du drone	5
5-Requête http Put.....	5
IV-Annexe technique	6
1-UML	6

I-Présentation

L'application Java doit recevoir un ID via une requête http sur la route « /transports/autofetch » puis grâce à cette ID récupérer via une autre requête http un fichier KML sur la route « /transports/id/route » le traiter simuler la trajectoire d'un drone qui suivrait la route définie dans ce fichier et renvoyer périodiquement la position de ce dernier au serveur via une requête http utilisant la méthode « PUT ».

II-Interface



1-Les Champs

Les Champs sont tous pré-rempli avec des valeurs par défaut.

IP : doit contenir l'IP du serveur avec lequel on veut communiquer (utilisé les commande ifconfig sous linux ou ipconfig sous windows)

Port : Port utilisé par le serveur pour communiquer avec le client java

Speed : Vitesse en Km/h, est verrouillé après l'utilisation du bouton « Load » (par défaut 18Km/h vitesse moyenne d'un quadricoptère)

2-Les Boutons

Gen. URL : Utilise les informations des champs IP et Port (<http://IP:Port/>) ce bouton déverrouille le bouton « Load » et passe le statu « Connected » à vrai (La led passe de rouge à vert).

Load : lance la requête d'autofetch, puis la requête de récupération du KML, qui est enregistré sur la machine, verrouille le champ Speed, affiche le temps estimé, et dessine la route sur la carte, déverrouille le bouton « Start » et « Reini. ».

Start : lance la simulation et déverrouille « Stop » .

Stop : Termine la simulation au point où elle est au moment de l'utilisation.

Reini. : Réinitialise toute l'interface et la remet au même stade qu'au démarrage de l'application, n'impacte qu'un seul onglet.

+ : Ajoute un nouvel onglet indépendant des autres et qui peut gérer une autre route (ne pas en ouvrir trop pour ne pas ralentir l'application).

3-Les Labels textes

POI Total : le nombre de POI sur la route.

Km total : la distance total de la route en Km.

Estimated Time : Le temps estimé au format « XX H XX Min ».

4-Interactions avec la Carte

La souris permet d'interagir avec la carte, la roulette permet de zoomer et le clic droit permet de recentrer la carte, de plus le zoom prend en compte la position du curseur pour zoomer.

Il existe aussi des boutons sur la carte qui permettent de zoomer « + » « - » ou même cliquer directement sur la jauge pour zoomer plus vite.

III-Problèmes rencontrés

1-Fermeture onglet

J'ai cherché comment insérer un bouton dans un onglet pour le fermer au niveau du bouton d'onglet. Par manque de temps et gestion de priorité j'ai dû passer outre ce problème purement esthétique. Ceci dit une solution possible est de créer un JPanel composé d'un JLabel et d'un JButton qui sera utilisé avec un constructeur particulier du JTabbedPane.

2-Traitement KML

Pour la gestion KML, J'ai eu beaucoup de mal à trouver de la documentation et des exemples, Après quelques recherches sur internet j'ai trouvé les bibliothèques JDom et XPath qui permettent de traiter les fichiers XML, et j'ai donc pu traiter les KML récupérer les coordonnées et dessiner le trajet sur la carte.

3-Mise en cache de la carte

Suite à des problèmes rencontrés avec le serveur DNS, qui empêche la carte de s'afficher, j'ai voulu chercher un moyen de la mettre sur la machine pour la charger depuis un fichier et non pas depuis internet, j'ai trouvé un constructeur qui prend un paramètre TileCache qui semble permettre de faire ça mais par manque de temps je ne me suis pas penché sur cette partie, mais cela pourrait être intéressant de regarder cette fonction pour gagner en fluidité du client et en stabilité.

4-Déplacement du drone

Il fut complexe de trouver un moyen de simuler le déplacement du drone et de trouver un moyen de déterminer la position du drone à un instant t .

J'ai pu réaliser un algorithme qui va en réalité avec un pourcentage du temps écoulé par rapport au temps estimé déterminé un pourcentage de la distance effectuée par rapport à la distance totale.

5-Requête http Put

J'ai rencontré un problème avec la partie PUT des requêtes http, le problème était en réalité qu'il suffisait dans l'header de rajouter « Content-Type : application/json ».

IV-Annexe technique

1-UML

Schéma représentant le fonctionnement global

