

# Twitter Map & Trend

## Projet de Java – Première partie Licence 3 Informatique 2009-2010

Rémi Forax [forax@univ-mlv.fr](mailto:forax@univ-mlv.fr) – John Chaussard [j.chaussard@esiee.fr](mailto:j.chaussard@esiee.fr)

### Contexte

Dans le but de développer une application appelée "Twitter Map & Trend", une entreprise a décidé de confier le développement à deux personnes, vous et une personne sachant faire des interfaces graphiques (vous l'année prochaine).

Le but de ce projet est donc de réaliser la partie dite métier de l'application "Twitter Map & Trend" sachant que la l'interface graphique vous sera fourni dans la seconde partie du projet

### L'application "Twitter Map & Trend"

Cette application a pour but d'afficher les conversations de différentes personnes (twits).

Le scénario nominal de l'application est le suivant:

Après avoir démarré l'application, utilisateur choisi une personne (un twit) ayant un identifiant sur twitter, si plusieurs personnes correspondent à l'identifiant, l'utilisateur choisi parmi une liste de choix possible. Une fois la personne choisie, l'application recherche toutes les amis (au sens de twitter) suivis par cette personne et download l'ensemble de leur discussion (on ne vous demande pas de sauvegarder les discussions sur disque). Puis index les discussions en fonction des mots contenus dans celle-ci. Si les discussions sont dans différentes langues, elles seront optionnellement traduites dans le langage de la personne.

Les mots-clés les plus fréquents contenus dans les discussions sont alors affichés à l'utilisateur. Si l'utilisateur sélectionne un mot-clé, l'ensemble des discussions contenant ce mot-clé sont alors affichés à l'utilisateur qui peut alors les lire. L'ordre dans lequel les discussions s'affichent est important (le ranking), les discussions les plus importantes doivent apparaître avant les discussions moins importantes.

De plus, si un utilisateur choisit une discussion, l'endroit d'où la discussion a été émise ou à défaut l'endroit de résidence de l'utilisateur twitter (twit) ayant écrit la discussion doit s'afficher sur une petite carte présente sur l'interface graphique.

### APIs à utiliser

L'application va utiliser un ensemble de services Web pour obtenir les informations nécessaires à son bon fonctionnement.

- Les personnes (avec leur langage, leur emplacement, etc), leurs conversations seront récupérées en utilisant l'API de twitter
- Les coordonnées GPS correspondant à l'emplacement d'une personne seront récupérées en utilisant au choix de l'utilisateur:
  - l'API de géolocalisation de Google
  - l'API de géolocalisation de Yahoo
- La traduction des discussions sera faite en utilisant une API Java qui utilise le service Web de traduction de Google

Pour la gestion de l'index des discussions, même s'il existe beaucoup de bibliothèques très bien qui font cela en Java, on vous demande d'écrire votre propre indexeur.

Vous pouvez utiliser pour cela l'ensemble des classes du package `java.util`.

## Tâches

Tâches à effectuer pour la première partie du projet

- Ecrire un ensemble de tests vérifiant que l'on peut récupérer les informations à partir de twitter, google et yahoo  
Chaque test devra tester uniquement un type requête particulier.
- Proposer une architecture du projet c-a-d un ensemble de classes permettant de faire fonctionner la partie métier du logiciel (donc pas de partie graphique dans vos classes)  
Vous utiliserez un diagramme de classes (avec les relations d'association et d'héritage) puis pour chaque vous décrierez sa responsabilité (ce pourquoi elle existe, à quoi elle sert) ainsi que le rôle de l'ensemble de ses méthodes publiques.
- Proposer un algorithme de ranking des discussions
- Ecrire un document présentant votre vue du projet reformulé avec vos mots, la description de votre algorithme de ranking, la description de l'architecture, qu'une description de chaque tests (sans le code) avec une introduction, des phrases écrites en français et une conclusion (le document devra faire une vingtaine de pages).

## Rendu

Ce projet est à faire par binome (donc 2 personnes, pas 1 ni 3).

Une archive ZIP (pas rar, tar, 7z ou xz) contenant le document demandé format PDF et les fichiers source (commentés) des tests devra être envoyé au plus tard le 13 décembre 2009 au soir à 23h59 aux deux adresses figurant au début de ce document.

Le sujet étant assez ouvert, toute tentative triche, de plagiat (par exemple du Web) ou de travail à plus que deux sera sévèrement puni.

## Références

<http://en.wikipedia.org/wiki/Twitter>

[http://en.wikipedia.org/wiki/Class\\_diagram](http://en.wikipedia.org/wiki/Class_diagram)

<http://apiwiki.twitter.com/>

<http://code.google.com/intl/fr/apis/maps/documentation/geocoding/>

<http://developer.yahoo.com/maps/rest/V1/geocode.html>

<http://code.google.com/p/google-api-translate-java/>

Parser XML en Java (API SAX):

<http://java.sun.com/webservices/reference/tutorials/jaxp/html/intro.html>