

Analyse et conception

Application d'analyse et de collection de Tweets

Introduction

Cette application a été conçue dans l'objectif de permettre de retrouver et de collecter des tweets et leur méta données en fonction de différents critères puis de les visualiser sous différentes formes.

A propos de ce document (Document purpose)

Ce document a pour but de spécifier les différentes fonctionnalités de l'application et adapter au mieux le produit pour satisfaire les besoins des différents utilisateurs . Il met en évidence les besoins et les choix que nous avons eu à prendre durant la conception. rencontrées.

Portée du produit (Product Scope)

Ce produit permet à un utilisateur de collecter assez rapidement une certaine quantité de données de la plateforme Twitter en fonction de plusieurs critères de recherche, de stocker ces données et pouvoir y accéder ultérieurement . Mais aussi de suivre en direct l'actualité sur Twitter, ou d'un compte twitter particulier.

Public concerné et vue d'ensemble de ce document

L'application est destinée à toute personne voulant accéder à des données de Twitter que ça soit juste pour s'informer, pu pour les collecter des données à des fins de traitements.

2- Fonctionnalités du produit

Nous avons décidé que l'application possède plusieurs fonctionnalités que l'on peut classer en deux catégories :

➤ Besoins fonctionnels:

- Récupérer les tweets selon un critère défini par l'utilisateur (hashtag, langue, auteur, mot-clef, période de temps, dernier tweet, localisation, lieu)
- Stocker ces données dans une base de données locale
- Recevoir les tweets en direct
- Utiliser l'application via des lignes de commandes et via une interface graphique
- Stocker les données associées aux tweets
- gérer les collections de tweets (les lister, les exporter au format CSV, les supprimer).

Besoins non fonctionnels:

- l'application nécessite une connexion Internet
- l'application marche sur PC
- l'application doit être assez réactive: le temps entre la requête de recherche et l'affichage des résultats doit être acceptable
- l'application doit être assez ergonomique: la prise en main ne doit pas être difficile.

Besoins par utilisateur : l'application étant tout public, il n'y a pas de besoin particulier pour un stakeholder particulier

3- Environnement d'exécution (Operating Environment)

Le programme est un Release.jar qui s'exécute sur un OS (Windows, Linux, Mac.....).

4- Contraintes de conception et d'implémentation

Contraintes liées à la conception et à l'implantation :

- Utilisation de l'api twitter
- Base de données locale ou non
- interface graphique
- interface de programmation(ligne de commandes)
- visualisation dynamique

5- choix des solutions par rapport aux contraintes de conception

-API twitter: chaque membre du groupe a crée un api twitter et dispose de ses clés d'accès.

-Base de données: on a opté pour la base de données locale MySql

-visualisation dynamique: on effectue une visualisation dynamique des tweets au fur à mesure de leur arrivée , cependant ces tweets en direct ne sont pas stockés. Par contre on peut toujours les recuperer après grâce à une méthode qui recupère les tweets depuis un certain temps.

5- Manuel utilisateur (User Documentation)

Un manuel d'utilisation est disponible afin de faciliter la prise en main de l'application par l'utilisateur.

Diagramme UML de ReleaseFinal

6- Quelques diagrammes

Diagramme de la base de données

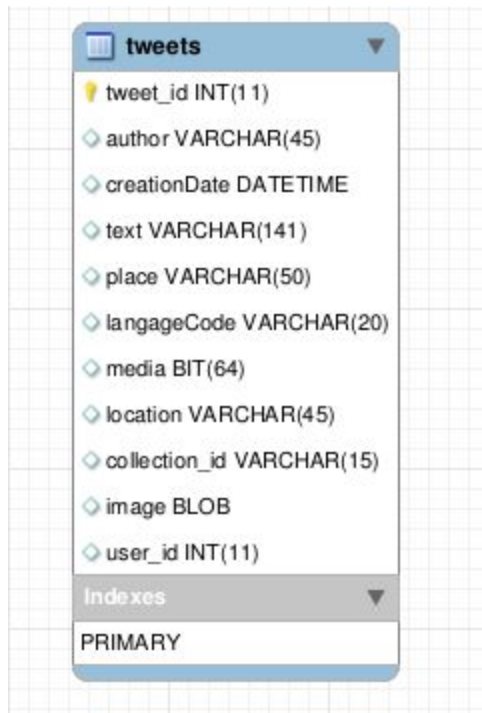


Diagramme UML de ReleaseFinal

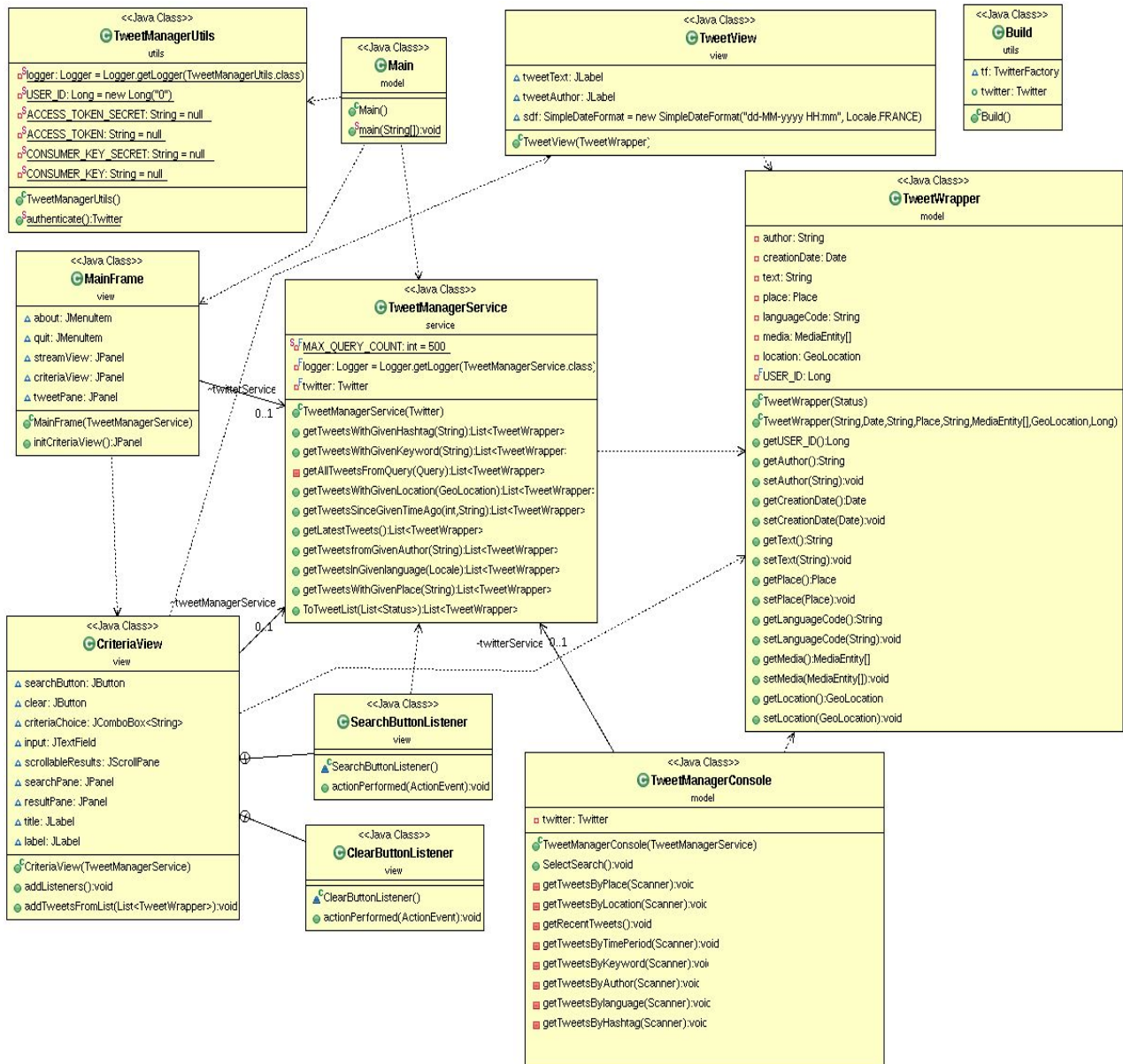


Diagramme UML de ReleaseFinal

