

Projet Java

L3 STRI Messagerie

Ce document décrit les étapes proposées pour développer le logiciel demandé.

A l'issue du développement, vous remettrez un document (format PDF) constitué d'un rapport, d'un guide utilisateur et de la **javadoc**, par courriel aux enseignants responsables du projet (la date exacte vous sera communiquée lors des séances de BE).

Le rapport devra notamment comprendre :

- Un paragraphe qui reformule les besoins
- L'organisation de travail, c'est-à-dire la répartition des tâches au sein des binômes (et du trinôme pour les groupes impairs).
- La modélisation UML des besoins (classes, cas d'utilisation etc.). Sélectionnez les diagrammes UML que vous jugerez pertinents pour la compréhension de votre logiciel.

Lors de la validation du projet, il vous faudra présenter votre application qui compile et fonctionne (sur les ordinateurs de l'université ou les vôtres) et répondre aux questions sur sa mise en œuvre et son code source. Vous remettrez les sources immédiatement après la validation.

I. Le sujet

La société STRI souhaite mettre en place un service de messagerie simplifiée (chat) afin que les collaborateurs puissent communiquer entre eux. Le service de messagerie possède plusieurs canaux, chaque canal a un titre et contient une liste des messages des utilisateurs.

Un collaborateur est caractérisé par son nom, son prénom, son grade. Suivant son grade il peut avoir accès ou non à la création d'un canal de discussion, à l'édition du titre du canal, accès au canal.

Un collaborateur peut être en ligne, hors ligne, occupé ou absent. Il a la possibilité de mettre une justification à son absence.

On souhaite notamment :

- Ajouter un utilisateur;
- Promouvoir un utilisateur;
- Ajouter des canaux, éditer le titre;
- Poster des messages;
- .

II. Travail

Les entités

Dans un premier temps, vous développerez les classes d'entités nécessaires pour le projet.

Chaque classe doit être commentée à l'aide du format **javadoc** (les attributs, les constructeurs, les getters et setters s'il y en a, les autres méthodes...).

La phase de modélisation devrait vous avoir permis de déterminer les attributs de chaque classe. Vous avez à votre disposition des structures de données dont vous connaissez le fonctionnement, comme les `ArrayList` et les `Hashtable`. Vous redéfinirez la méthode `toString` afin de faire des affichages durant le développement.

Vous utiliserez une classe de test pour vérifier la validité de vos méthodes au fur et à mesure.

Les méthodes

Voici quelques exemples de méthodes qu'il peut vous être demandé de développer :

- afficher les messages d'un canal;
- afficher l'ensemble des canaux;
- afficher les personnes du canal ; avec leur grade;
- mettre à jour un canal ;
- mettre à jour une personne.

Cette liste est **minimale**. Elle ne vous dispense pas de réfléchir à d'autres méthodes pour compléter et/ou améliorer votre logiciel.

Vous développerez les *controls*, *boundaries* et interfaces graphiques nécessaires pour rendre le logiciel convivial.

III. Plagiat et travail collectif

Toute fraude, même partielle, et tout travail collectif entre différents groupes de projet seront sanctionnés dans la notation du projet (et éventuellement par un recours au conseil de discipline). Notez également qu'il est illégal d'utiliser des sources sans en citer les auteurs.

IV. Conseils

Vous pouvez suivre les conseils donnés sur le Moodle concernant Eclipse, la rédaction d'un rapport, le format javadoc... En cas de doutes, n'hésitez pas à nous envoyer un mail : <mailto:barboni@irit.fr>.

Les séances proposées pour développer ce projet ne vous dispensent pas de travailler chez vous.