

Un annuaire des anciens étudiants du Master

1 Introduction

Le but de cette page est de présenter le projet qui va servir de base à votre évaluation dans cette unité d'enseignement.

2 Cahier des charges

L'objectif du mini projet est simple : gérer à l'aide de la technologie J2EE un annuaire des étudiants du Master deuxième année. Plus précisément,

- Chaque étudiant est doté d'un mot de passe qui l'authentifie,
- Les étudiants sont regroupés dans des promotions. Une promotion comporte une année, un titre, un ID et deux ou trois paragraphes pour la décrire.
- Prenez exemple sur l'annuaire actuel du Master¹ pour savoir quelles informations sont à gérer (nom, prénom, adresse postale, adresse électronique, site WEB, nom de la société, site WEB de la société, date de naissance, etc.).
- Les étudiants peuvent modifier les informations qui les concernent en donnant leur mot de passe.
- en cas d'oubli, les étudiants peuvent recevoir un mail pour leur rappeler leur mot de passe.
- Un administrateur (qui a son propre mot de passe) peut détruire, modifier ou ajouter des entrées dans l'annuaire (c'est à dire des étudiants et des promotions). L'ajout de plusieurs étudiants dans une promotion doit être réalisé en une seule opération.

3 Étape 1 : Couche de stockage (JPA)

Objectifs :

- Conception et création de la base de données.
- Conception et création des *JavaBeans* permettant de représenter les données de l'annuaire (étudiants, promotions, etc...).
- Utilisation de la technologie JPA pour gérer la persistance des *JavaBeans*.

Architecture : Vous devez mettre en place une couche de service d'accès aux données (souvent appelée DAO pour *Data Access Object*), Cette couche est constituée d'une interface (**indépendante du choix de JPA**) et d'une implantation (liée à JPA). Cette interface **pourrait** ressembler à ceci :

¹<http://masterinfo.univ-mrs.fr/annuaire/>

```
public interface AnnuaireDAO {

    // récupérer les ID des promotions
    Collection<String> listPromotions() throws ...;

    // lire une promotion (avec/sans étudiants)
    Promotion findPromotion(String id, boolean withStudents) throws ...;

    // ajout d'une nouvelle promotion
    void addPromotion(Promotion p) throws ...;

    ...
}
```

Vous devez **absolument** prévoir une classe de test unitaire *Junit* pour valider les méthodes offertes par votre implantation.

4 Étape 2 : Couche métier (RMI)

Objectifs :

- Offrir sous la forme d'un serveur RMI une interface métier vers l'annuaire des anciens du Master.
- Prévoir les méthodes permettant de renvoyer les promotions et les entrées de chaque promotion.
- Prévoir les méthodes permettant de modifier les promotions et les entrées.

Architecture : Ce serveur RMI disposera d'au moins deux classes d'objets distants : l'annuaire à proprement parler (une seule instance) et un utilisateur authentifié (plusieurs instances).

Validation : Pour tester cette couche métier RMI, vous devrez développer une (ou plusieurs) classe(s) de test unitaire (avec JUNIT) de manière à valider vos choix d'implantation.

Contraintes :

- Le serveur RMI est indépendant des choix de stockage (JPA).
- Chaque méthode du serveur RMI est une transaction.
- Le serveur RMI se charge de l'authentification.

5 Échéances des étapes 1 et 2

Ce projet est à rendre, **par binôme**, avant le **9 janvier 2012**. Documents attendus : une archive ZIP (de la forme « nom1-nom2.zip ») structurée contenant

- un répertoire javadoc pour la javadoc des classes,
- un répertoire src contenant les sources Java,
- une petite explication (moins de cinq pages) au format PDF.

6 Étape 3 : La couche de présentation (JSP)

Cette étape consiste à mettre en place une application WEB basée sur la technologie JSP/Servlet qui respecte les contraintes suivantes :

- L'application doit être organisée suivant le principe MVC. Le contrôleur sera implanté sous la forme d'une Servlet (avec Spring éventuellement)

- Dans la mesure du possible utilisez les balises de contrôle de la JSTL (boucles, tests et affichage).
- Faites en sorte que votre application produise des pages XHTML en version stricte accompagnées d'une feuille de style CSS (utilisez les pages JSPX et un plugin firefox de vérification de la validité²).

Dans la réalisation de votre logiciel, respectez les étapes suivantes (une étape doit être **terminée et testée** avant de passer à la suivante) :

1. Parcours de l'annuaire.
2. Modification d'une entrée après authentification (prenez soin de vérifier les données proposées par l'utilisateur).
3. Modification de toutes les entrées après authentification de l'administrateur.
4. Ajout d'une promotion et des étudiants par l'administrateur.
5. Suppression d'un étudiant ou d'une promotion par l'administrateur.
6. Mise à jour de la description d'une promotion par l'administrateur.

7 Échéance de l'étape 3

Ce projet est à rendre, avant le **Lundi 30 janvier 2012**. Documents attendus : une archive ZIP (de la forme « nom1-nom2.zip ») qui regroupe

- un rapport (fichier `rapport.pdf` de moins de 10 pages),
- les sources de l'application WEB (répertoire WEB),
- les sources du serveur RMI (répertoire RMI) avec bien entendu la couche DAO.
- la javadoc des sources Java (RMI et WEB) dans un répertoire javadocs.

La forme du rendu doit être **ABSOLUMENT** respectée. Des soutenances seront organisées.

²<http://users.skynet.be/mgueury/mozilla/>