

Rapport du projet d'AAW : Réseau social en Java J2EE.

William Le Coroller & Romain Boutin

Mercredi 25 Février 2015

Table des matières

1	Introduction	1
2	Contexte	2
3	Choix techniques	3
4	Conception	4
4.1	Modèle base de données	4
4.2	Diagramme de classe	5
4.3	Couche DAO	6
4.4	Couche Service	7
4.5	Couche Présentation	7
5	Conclusion	8

1 Introduction

Le but de ce projet est la réalisation d'un réseau social en utilisant les technologies Java J2EE.

Le rapport est organisé en trois parties principales :

- La partie **contexte** qui présente les points que le projet doit respecter.
- La partie **technique** qui présente les choix techniques qui ont été effectués.
- La partie **conception** qui présente la conception des différentes couches de l'application.

2 Contexte

Le but de ce projet est de concevoir et implanter un réseau social simplifié.

Les contraintes suivantes nous ont été imposées :

- Chaque utilisateur est enregistré en base de données.
- Chaque utilisateur dispose d'une liste d'amis qui lui est associé.
- Chaque utilisateur pourra partager des fichiers avec ses amis.
- Chaque liste d'amis des utilisateurs est enregistré en base de données.
- Chaque utilisateur a la possibilité d'échanger des messages avec ses amis.
- Chaque utilisateur a la possibilité d'identifier ses amis dans ses publications.
- Dès lors il sera notifié d'une telle identification.

3 Choix techniques

Nous avons choisi d'utiliser comme technologies la combinaison des Enterprise JavaBeans (EJB) et JavaServer Faces (JSF). En effet, ces technologies nous ont paru plus intuitive et solide que leurs homologues JPA et Spring. L'utilisation redondante de JavaBeans étant indéniablement un atout en terme d'évolutivité et de maintenabilité.

4 Conception

4.1 Modèle base de données

Table T_USER

Cette table décrit l'ensemble des informations des utilisateurs inscrit sur le réseau social.

Table T_RELATIONSHIP

Cette table décrit l'ensemble des informations permettant d'établir une relation d'amitié entre deux utilisateurs inscrit sur le réseau social.

Table T_NOTIFICATION

Cette table recense l'ensemble des notifications des différents utilisateurs du réseau social.

Table T_MESSAGE

Cette table regroupe l'ensemble des messages publiques et privés échangés par les utilisateurs du réseau social.

4.2 Diagramme de classe

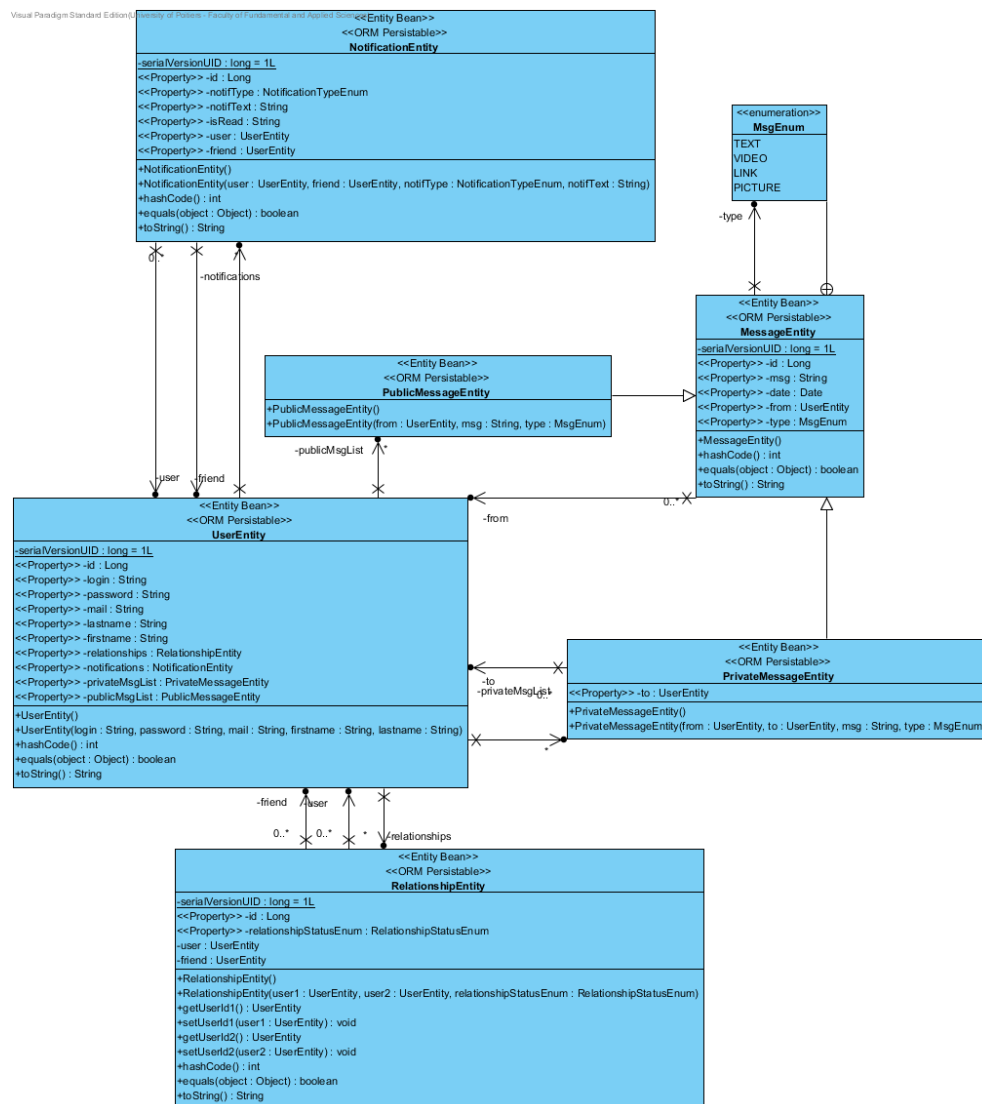


FIGURE 1 – Le diagramme de classe associé au package « entity »

4.3 Couche DAO

La couche DAO effectue l'interfaçage entre la base de données et la couche service/métier. Elle fournit au travers des entités et des Entreprises JavaBeans une modélisation des données stockées dans la base de données.

Package Entity

Les classes présentes dans ce package sont les classes « entités ». Ces classes donnent un cadre uniforme pour l'accès aux données en réalisant un mapping des différentes tables présentes en base de données.

- **MessageEntity**, **PublicMessageEntity** et **PrivateMessageEntity** : Ces classes représentent la table « T_MESSAGE ».
- **NotificationEntity** : Cette classe représente la table « T_NOTIFICATION ».
- **RelationshipEntity** : Cette classe représente la table « T_RELATIONSHIP ».
- **UserEntity** : Cette classe représente la table « T_USER ».

Package SessionBeanLocal et SessionBean

Les classes présentes dans ce package sont les interfaces locales des Beans de session propre à EJB.

L'ensemble des classes de ce package présente une interface assez similaire avec la déclaration des méthodes « save », « update » et « delete » permettant respectivement de sauvegarder ou créer, de mettre à jour ou de détruire une entité et de répercuter ce changement en base de données à l'aide d'un objet « EntityManger ».

4.4 Couche Service

Aussi appelé « couche métier », elle fait le lien entre la couche présentation et la couche DAO. Avec EJB, cette couche se résume à l'implémentation de JavaBeans dit de service fournissant donc des méthodes de plus haut niveau afin de traité les requêtes utilisateurs plus simplement.

Package BeanLocal et Bean

L'implémentation a été effectué de telle sorte que pour chaque fonctionnalité est associé le service correspondant. On voit bien ici la force des EJB.

4.5 Couche Présentation

Il s'agit de la couche la plus proche de l'utilisateur. C'est elle qui présente à l'utilisateur l'ensemble des actions, via des pages web, qu'il peut effectuer.

Package ManagedBeans

Pour un soucis de gestion de session il a été décidé de créer un managedBean plus important. Les autres managedBeans représentent des fonctionnalités indépendantes de l'utilisateurs.

Gestion des messages

On a choisi par simplicité d'imposer à l'utilisateur de poster des vidéos ou des photos via des liens de fichiers déjà uploadé. On applique juste un traitement sur les différents liens pour les afficher correctement.

5 Conclusion

La quasi-totalité des fonctionnalités attendues ont été implémentées à l'exception de l'identification d'un amis dans un message.

Le réseau social est fonctionnel bien qu'austère. Il respecte l'ensemble des directives d'implémentation des technologies J2EE traitées en cours.

On regrette néanmoins de ne pas avoir pu rajouté des fonctionnalités de confort ainsi que de vérification afin de rendre l'expérience utilisateur plus ergonomique et intuitive.