5I-IN9 Développement web – TP3 – Services web

Introduction

Dans le cadre de ce TP, nous allons continuer à travailler sur l'application réalisée lors du TP2. Nous la compléterons par des ajouts fonctionnels et nous intégrerons également de nouvelles technologies ainsi que de nouveaux concepts.

Cette approche de TP évolutif nous permettra de simuler l'évolution d'un projet dans le cadre d'un développement en entreprise.

Prérequis

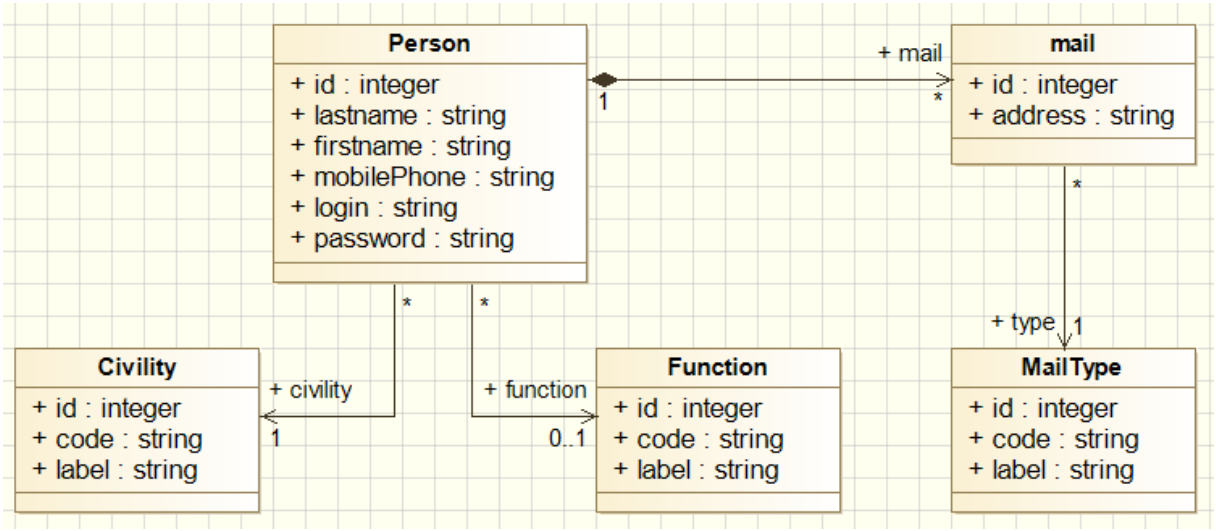
Afin de pouvoir commencer à travailler sur ce TP, vous devez avoir finalisé et envoyé par mail le TP2.

A la pratique

Partie 1 – Modification de la couche de données

Après une utilisation du modèle mis en place et retour du client final, nous nous apercevons que les utilisateurs peuvent disposer de plusieurs adresses mails et que celles-ci disposent d'un type :
« Privé », « Professionnel »

Nous proposons donc de faire évoluer le schéma comme suit :



a) Faite évoluer le modèle de données de manière à gérer les modifications tout en respectant la nomenclature de code existante.
b) Migrer les données existantes dans le nouveau modèle (adresses mails)

Partie 2 – Mise en place d'une API REST

L'application mise en place semble répondre aux différents besoins du client final. Cependant, il souhaite disposer des données de l'application à travers d'autres outils existants ou à venir. Ces applications peuvent ou non être développées dans le langage Java.

Afin de répondre à ce nouveau besoin, nous proposons la mise en place d'une API REST nous permettant de garantir l'interopérabilité des systèmes et la mise à disposition de manière sécurisée des données du référentiel.

Consigne particulière : Implémenter l'API REST dans l'application en utilisant uniquement les technologies vues en cours.

Partie 3 – Remplacer l'authentification session par un jeton

Dans le cas d'une application backend offrant des API REST, il est fréquent de disposer d'une authentification pouvant également se faire par une API et utilisant un système de jeton pour fonctionner.

Inspirez-vous des concepts vus en cours afin de mettre en place une authentification par jeton en remplacement de l'authentification par session vue lors du TP2.

Partie 4 – Migration du code existant (*Facultatif*)

Nous avons répondu aux besoins du client dans les deux parties précédentes, mais l'incorporation de ces ajouts nous apporte une double gestion de nos données. Effectivement, nous disposons de servlets et tags JSP pour manipuler nos objets, mais aussi d'une API REST.

Cette double approche pour une même fonctionnalité est source dysfonctionnent tant fonctionnellement qu'au niveau applicatif. Pour toutes ces raisons, nous proposons de refactoriser le code existant afin que la manipulation des objets et l'authentification passe uniquement par les API REST.

Migrer l'application afin que les écrans de l'application ne manipule plus directement les données mais passe obligatoirement par les API REST.

Allez plus loin

Afin d'aller plus loin dans ce TP, vous pouvez compléter les fonctionnalités afin de présenter les différents concepts étudiés et de vous approprier le projet.

Voici quelques idées :

- Ajout de nouveaux écrans dans l'application
- Amélioration de la sécurité applicative (ex : expiration du token, complexification du token, chiffrement du mot de passe en base, etc.)
- Ajout de contraintes permettant de garantir l'intégrité du modèle (ex : Si je supprime une personne, je dois également supprimer les mails lui étant associés, bloquer la suppression d'une fonction si celle-ci est utilisée par une autre entité, etc.).
- Etc.