

Kévin OUAHMAD
Paul LEMETTRE

Description

Description de l'architecture du projet :

model :

Ensemble de classes Java représentant les modèles métiers
(User, Service, Colocation, Achieved Service, Address)

dao :

Ensemble d'interface des différentes Dao pour chaque modèle.

controller :

Ensemble de classes Java qui sont en fait des servlets (HttpServlet) c'est à elles que s'adresse les requêtes des utilisateurs, elle utilise ensuite la Dao correspondante afin de récupérer un objet Java pour le "transmettre à l'utilisateur"

dbms :

Ensemble de classes Java implémentant les différentes Dao de chaque modèle. Elle sont en lien direct avec la base de données et renvoie des objets Java directement aux controllers

exception :

Ensemble des exceptions pouvant survenir lors des tests

security :

Classes Java qui permettant d'encoder (SHA-256) les mots de passes des utilisateurs

config :

InitialisationDaoFactory permet la connection à la base de donnée, elle utilise le fichier WEB-INF/dao.properties qui contient l'ensemble des informations pour se connecter à la base de donnée

Exemple d'utilisation d'une requête : (Quand le ? précède une variable c'est que la variable peut-être nulle)

get : /users/?mail=email :

permet d'afficher l'utilisateur avec le mail correspondant, ou tout les utilisateurs

s'adresse à la servlet Users (dans /controller), ensuite la UserDaoMem est utilisé (dans dbms) qui implémente UserDao, UserDaoMem se charge de récupérer les informations de la base de données et renvoie un objet java (ici User ou une Collection de User si mail n'est pas définie, User est définie dans model) ce résultat est renvoyé à la servlet Users qui parse la réponse en JSON et l'affiche.