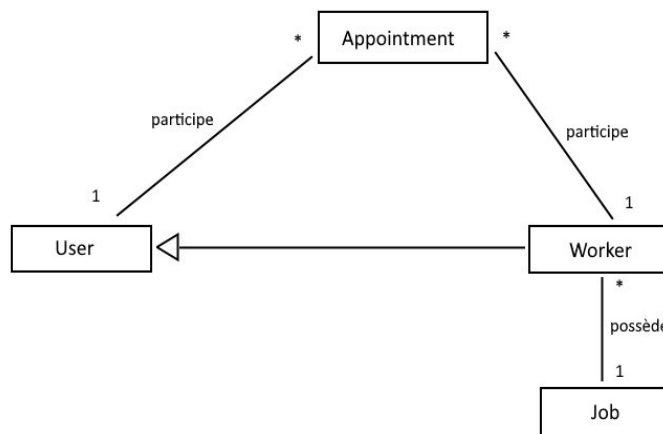


TP1: 'un Doctolib pour voir les profs'

Importez le projet et exécutez la base de données (run-hsqldb-server). Ouvrez l'afficheur de tables (show-hsqldb) et cochez « HSQL Database Engine Server ». A chaque changement de la base de données, appuyez sur : View>Refresh Tree.

On est parti du projet fourni par Olivier Barais et on a construit nos propres classes métiers. Nos classes métiers respectent le diagramme UML ci-dessous.



Question 1)

On a transformé la classe User en entité. On a mis la liste de 'appointment' en Transient en attendant de s'occuper de la classe Appointment.

Question 2)

On s'est ensuite occupé de la classe Worker ainsi que la classe Job. On a géré l'association Worker=>Job. Un worker a un job, il s'agit d'une association unidirectionnelle, donc côté worker on a ajouté l'annotation '@ManyToOne' et on a pas mis de « mappedBy » (étant donné que unidirectionnel).

Question 3)

On a créé une interface DAO générique 'IgenericDao' composé des méthodes DAO basiques (save, update, delete, deleteById) ainsi que la méthode findAll qui renvoie tous les objets d'une classe. Ensuite on implémenté cette interface avec la classe abstraite 'AbstractJpaDao'. Finalement, pour chaque classe métier, on a créé une classe DAO qui étend cette classe abstraite. Pour la classe UserDAO, on a créé une méthode particulière qui retourne le worker le mieux noté.

Question 4)

Question 5)

Notre modèle contient une relation d'héritage (entre User et Worker). Les classes DAO contiennent des requêtes. On a ajouté des requêtes nommées. Elles sont définies dans des annotations des classes métiers.