

Rapport projet Spring

Description du projet :

Dans ce projet, il est question de développer une application web à l'aide du framework Spring. Cette application a pour but la vente de produits alimentaires en ligne.

L'application est censée contenir plusieurs fonctionnalités telles que :

- Création, édition, suppression et listing de produits de consommables.

La vue de création d'un produit permet de renseigner au minimum le nom, le code du produit, la quantité et la date de péremption. Le listing permet d'afficher le nom, le code du produit, la quantité, le nombre de jours restant avant péremption, et une information visuelle si le produit est périmé.

- La création d'un compte client, le login d'un utilisateur et l'affichage du compte utilisateur courant (partie V)

L'application sera développée sous le framework Spring couplé à Hibernate, via l'IDE IntelliJ sur Windows 10.

Membres du projet :

- Inapogui Isshia
- Apprin Rubens
- Luquet de Saint Germain Guillaume
- Elkatrani Ismail

Liste des entités persistées

Les utilisateurs et les produits sont des entités persistées en base de donnée car ce sont les informations principales que notre application doit gérer :

- Tout les produits : car l'application doit permettre la création, édition, suppression et le listing de produits.

- Les utilisateurs : car l'application doit permettre la création de comptes clients.

Cycle de vie des sessions Hibernate

Les sessions sont créées à chaque transaction entre l'application et la base de données, que ce soit pour une création, une suppression, une édition, un listing de produits ou encore la création d'un utilisateur.

Nous avons décidés d'établir une session par transaction, cela nous permet de séparer les différentes opérations (création, suppression, édition, listing) et, dans le cas où une transaction échoue, il n'y aura pas d'incidence sur les autres transactions.

Principales classes

Les classes principales :

- La classe Produit qui contient donc les attributs persistés de produits (lié à la table Produit en Base de données).
- La classe Utilisateur qui contient les attributs persistés d'un utilisateur (lié à la table Utilisateur en Base de données).
- TransactionManager : permet de créer une SessionFactory complète, avec tous les paramètres important à sa mise en place.
- DataSource qui contient les informations de la base de données associée.
- HibernateProperties qui contient les informations du comportement d'hibernate.
- LocalSessionFactoryBean qui lui récupère la dataSource, et les hibernateProperties et créer une SessionFactory.
- Les classes ProduitDao et UtilisateurDao contiennent des méthodes qui implémentent respectivement ProduitDaoImpl et UtilisateurDaoImpl.
- La classe ProduitDaoImpl permet de créer un produit, éditer un produit, lister les produits et supprimer un produit.
- La classe UtilisateurDaoImpl permet de créer un utilisateur, récupérer le nom d'un utilisateur et la connexions de ses utilisateurs.

Leur rôle est essentiel dans le fonctionnement primaire de l'application, car ces deux classes permettent un mapping objet-relationnel, ou chaque attribut persisté correspond à une colonne de la Table associée.

Grâce à l'annotation `@Configuration`, Spring va créer une instance unique de cette classe utilisée comme configuration (remplace le `hibernate.cfg.xml`).

Lien direct vers le projet : [ici](#).