










Création d'une application Web JAVA EE basé sur Spring MVC Coté Serveur avec Thymleaf et Spring Security

Besoin Client :

On souhaite créer une application qui permet de gérer des comptes bancaires.

- Chaque compte est défini par un code, un solde et une date de création
- Un compte courant est un compte qui possède en plus un découvert
- Un compte épargne est un compte qui possède en plus un taux d'intérêt
- Chaque compte appartient à un client
- Chaque client est défini par son code et son nom
- Il existe deux types d'opérations : Versement et Retrait
- Une opération est définie par un numéro, une date et un montant

Exigences fonctionnelles :

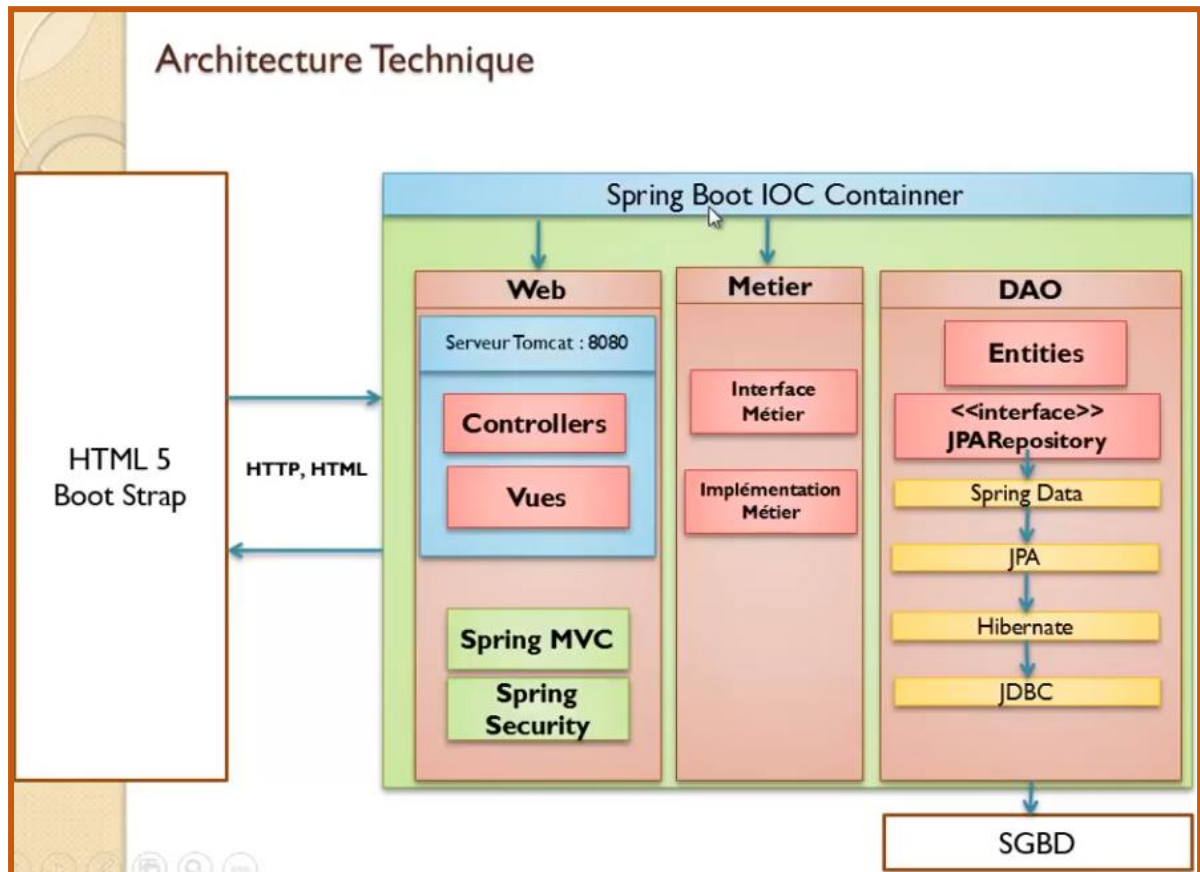
- L'application doit permettre de :
 - Gérer des clients :
 -  Ajouter un client
 -  Consulter tous les clients
 -  Consulter les clients dont le nom contient un mot clé
 - Gérer les comptes :
 -  Ajouter un compte
 -  Consulter un compte
 - Gérer les opérations
 -  Effectuer un versement d'un montant dans un compte
 -  Effectuer un retrait d'un montant dans un compte
 -  Effectuer un virement d'un montant d'un compte vers un autre
 -  Consulter les opérations d'un compte page par page
- Les opérations nécessitent une opération d'authentification.

Exigences Techniques :

- Les données sont stockées dans une base de données MySQL
- L'application se compose de trois couches :

- La couche DAO qui est basée sur Spring Data, JPA, Hibernate et JDBC
- La couche Métier
- La couche Web basée sur MVC côté serveur en utilisant Thymeleaf
- La sécurité est basée sur Spring Security

Architecture Technique

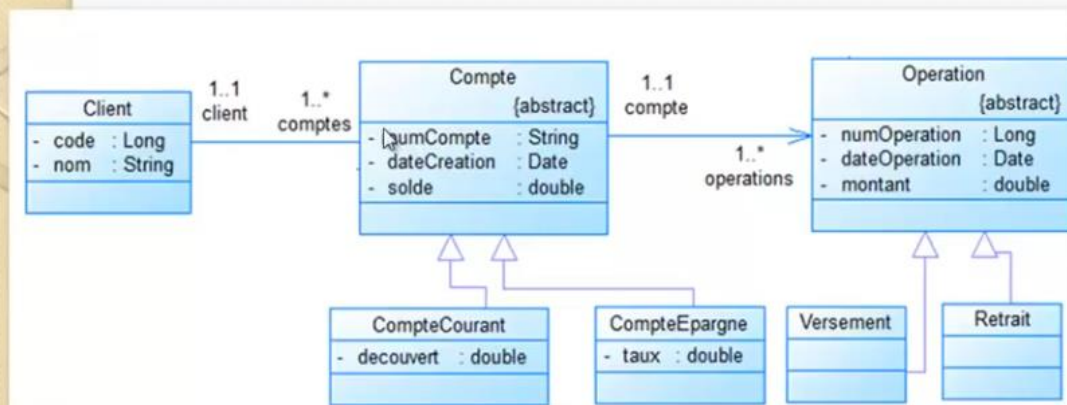


Travail à effectuer :

- Etablir une architecture technique du projet
- Etablir un diagramme de classes qui montre les entités, la couche DAO et la couche métier.
- Créer un projet Spring Boot qui contient les éléments suivants :
 - Les entités
 - La couche DAO (Interfaces Spring Data)
 - La couche métier (Interfaces et implémentations)
 - La couche Web :

- Les contrôleurs Spring MVC
- Les vues basées sur Thymeleaf
- Sécuriser l'application en utilisant un système d'authentification basé sur Spring Security
- Diagramme de Classe des entités

Diagramme de classes des entités et MLDR



- **MLRD** : En utilisant la stratégie Single Table pour l'héritage
 - **T_CLIENTS** (**CODE_CLI**, **NOM_CLI**)
 - **T_COMPTES** (**NUM_CPTE**, **TYPE_PTE**, **DATE_CR**, **SOLDE**, **DEC**, **TAUX**, **#CODE_CLI**)
 - **T_OPERATIONS** (**NUM_OP**, **TYPE_OP**, **DATE_OP**, **MONTANT**, **#NUM_CPTE**)