

Examen Architecture Distribuée et Middleware

Durée: 3h00

On souhaite concevoir et développer une application Web JEE basée sur le Framework Spring et Angular. L'application permet de gérer des crédits bancaires. Cette épreuve couvre une partie de l'application dont les règles de gestion sont les suivantes :

- Un client peut effectuer plusieurs crédits.
- Il existe trois types de crédits : Crédit Personnel, Crédit Immobilier et Crédit Professionnel
- Un crédit peut avoir plusieurs remboursements
- Un client est défini par son id, son nom et son email
- Un crédit est défini par son id, la date de demande, son statut (En cours, Accepté, rejeté), date d'acception, son montant, sa durée de remboursement et son taux d'intérêt.
- Un crédit personnel est un crédit qui possède en plus son motif (exemple achat de voiture, études, travaux)
- Un crédit immobilier est un crédit qui possède en plus le type du bien fiancé (Appartement, Maison, ou Local Commercial)
- Un crédit Professionnel est un crédit qui possède en plus le motif et la raison sociale de l'entreprise.
- Un remboursement est défini par son id, sa date, son montant et son type (Mensualité, Remboursement anticipé)

Les données sont stockées dans une base de données de type H2 ou MySQL.

L'application se compose des couches suivantes :

- La couche DAO est basée sur Spring Data, JPA, Hibernate et JDBC
- La couche service (métier) définie par une interface et une implémentation
- La couche Web est basée sur Spring MVC en utilisant des REST Controllers

La sécurité de l'application est basée sur Spring Security avec Json Web Token

Travail demandé :

Rendre le projet (code source) et un rapport au format PDF contenant les réponses aux questions suivantes (à déposer sur Classroom à 12H30) :

A. Livrables :

1. Créer un repository Github qui porte le nom suivant : VotreNom-VotrePrenom-Exam-JEE
2. Déposer le lien de ce repository dans classroom
3. Pour la suite de l'épreuve, il faudrait effectuer un commit et un push pour chaque période de 20 minutes.
4. Rendre un rapport au format PDF à déposer dans classroom

B. Conception :

5. Établir une architecture technique du projet
6. Établir un diagramme de classes qui montre les entités. On ne représentera que les attributs.

C. Implémentation :

1. Créer un projet Spring boot avec les dépendances requises. Les identifiants du projet GroupId, ArtifactId et le package de base doivent contenir votre nom et prénom.
2. Couche DAO
 - a. Créer les entités JPA
 - b. Créer les interfaces JPA Repository basées sur Spring Data
 - c. Tester la couche DAO avec une application qui alimente la base de données avec quelques enregistrements de test.
3. Créer une couche service en créant les DTOs et les Mappers, en proposant les fonctionnalités que vous voyez importantes
4. Créer les Web services (Rest Controllers) en proposant les fonctionnalités que vous estimez importantes. Tester les REST API en générant la documentations SWAGGER (Open API Doc)
5. Proposer une application frontend en utilisant Angular Framework
6. Sécuriser d'accès aux applications backend et frontend application en se basant sur Spring Security et Json Web Token avec un système d'authentification des utilisateurs avec 3 types de rôles « ROLE_CLIEN », « ROLE_EMPLOYE » et « ROLE_ADMIN » en choisissant des autorisations appropriées à ses rôles
7. Apporter des améliorations additionnelles à votre projet