

**Département Mathématiques et Informatique**

**Rapport du Contrôle**

**Filière :**

**« Ingénierie Informatique : Big Data et Cloud Computing »**

**II-BDCC**

**Examen JEE : Gestion des Crédits**

DBM Maroc – ACCENT

**Réalisé par :**

Yassine HACHGUER

**Année Universitaire : 2024-2025**

ENSET, Avenue Hassan II - B.P. 159 - Mohammedia - Maroc

05 23 32 22 20 / 05 23 32 35 30 – Fax : 05 23 32 25 46 Site Web: www.enset-media.ac.ma  
E-Mail : enset-media@enset-media.ac.ma

**Table de matière**

[Livrables 3](#_Toc198546388)

[1. Créer un repository Github qui porte le nom suivant : VotreNom-VotrePrenom-Exam-JEE 3](#_Toc198546389)

[Conception 4](#_Toc198546390)

[1. Établir une architecture technique du projet 4](#_Toc198546391)

[2. Établir un diagramme de classes qui montre les entités. On ne représentera que les attributs 5](#_Toc198546392)

[Implémentation 5](#_Toc198546393)

[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 5](#_Toc198546394)

[2. Couche DAO 6](#_Toc198546395)

[a. Créer les entités JPA 6](#_Toc198546396)

[b. Créer les interfaces JPA Repository basées sur Spring Data 6](#_Toc198546397)

[c. Tester la couche DAO avec une application qui alimente la base de données avec quelques enregistrements de test. 7](#_Toc198546398)

[3. Créer une couche service en créant les DTOs et les Mappers, en proposant les fonctionnalités que vous voyez importantes 8](#_Toc198546399)

[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) 10](#_Toc198546400)

[5. Proposer une application frontend en utilisant Angular Framework 13](#_Toc198546401)

# Livrables

## Créer un repository Github qui porte le nom suivant : VotreNom-VotrePrenom-Exam-JEE

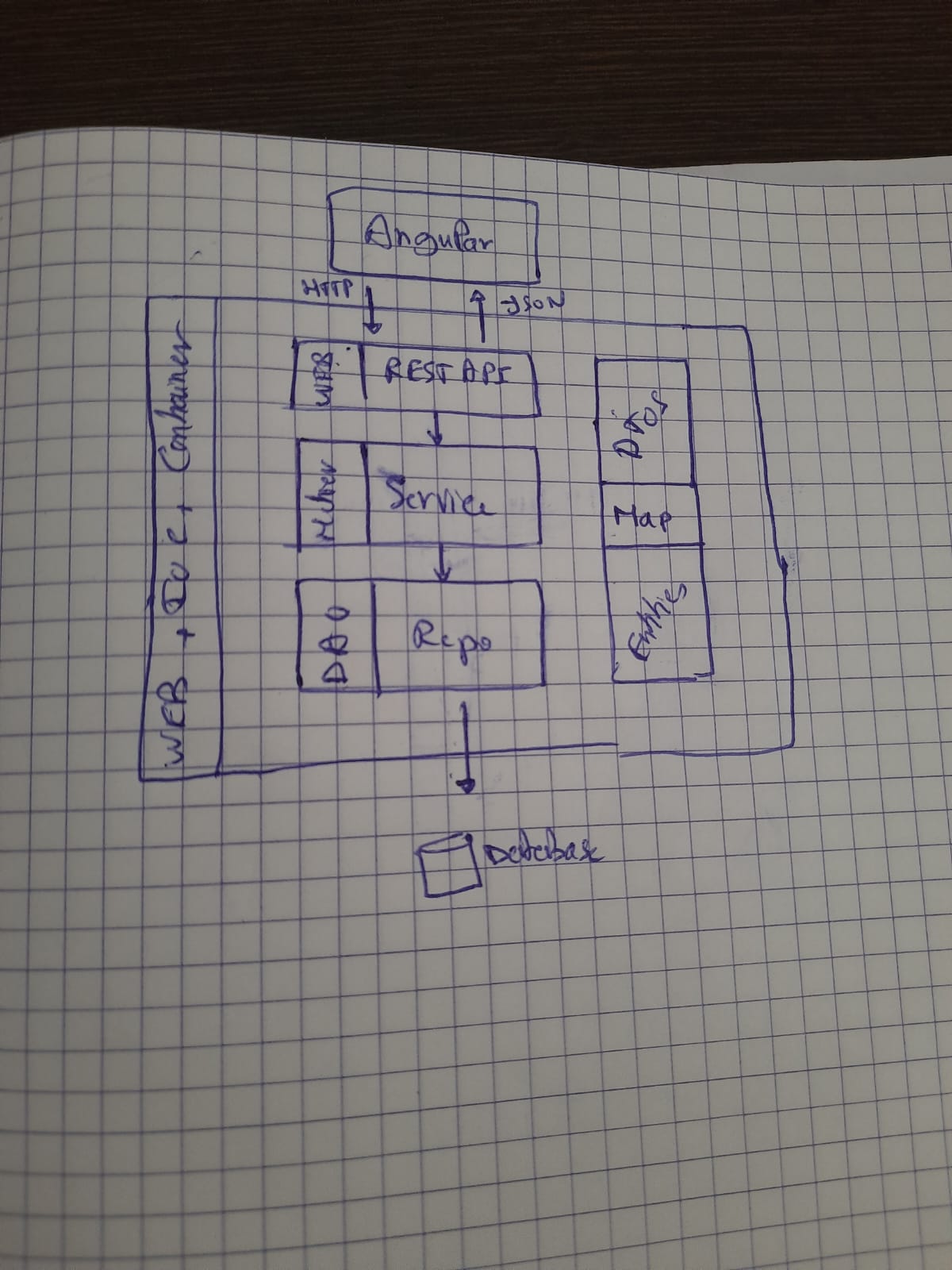


**J’ai 2 branch :**

**La branch Main Contient le Backend du projet   
La branch Master Contient le Frontend du projet**

# Conception

## Établir une architecture technique du projet

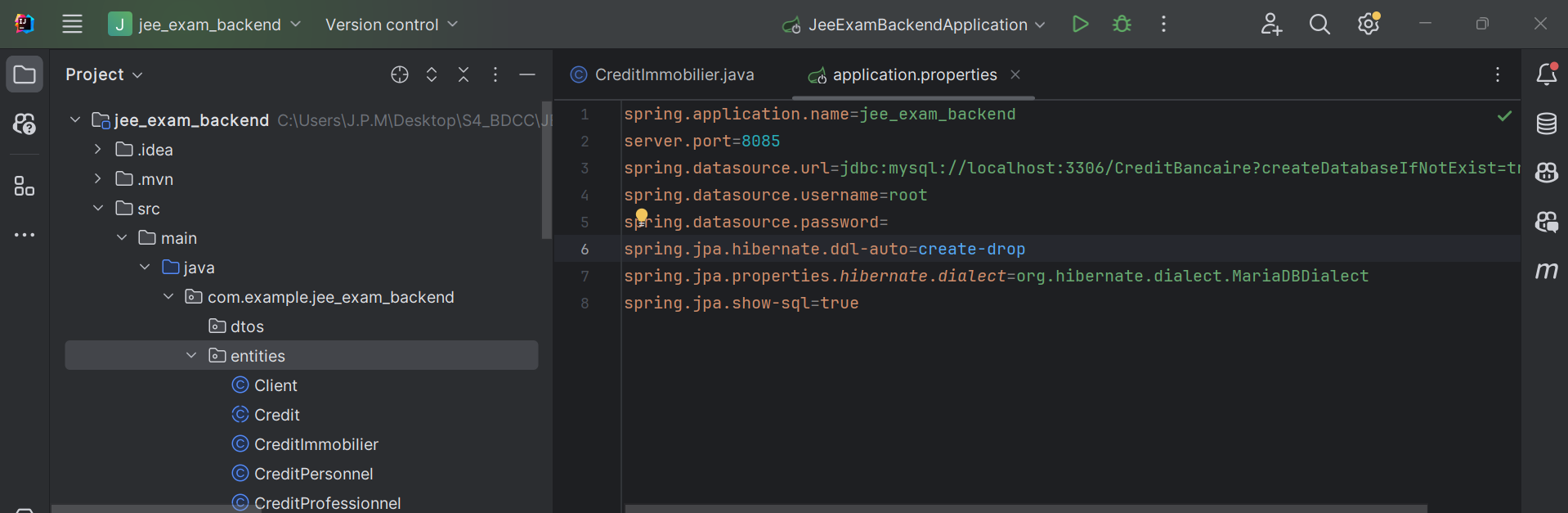


## Établir un diagramme de classes qui montre les entités. On ne représentera que les attributs



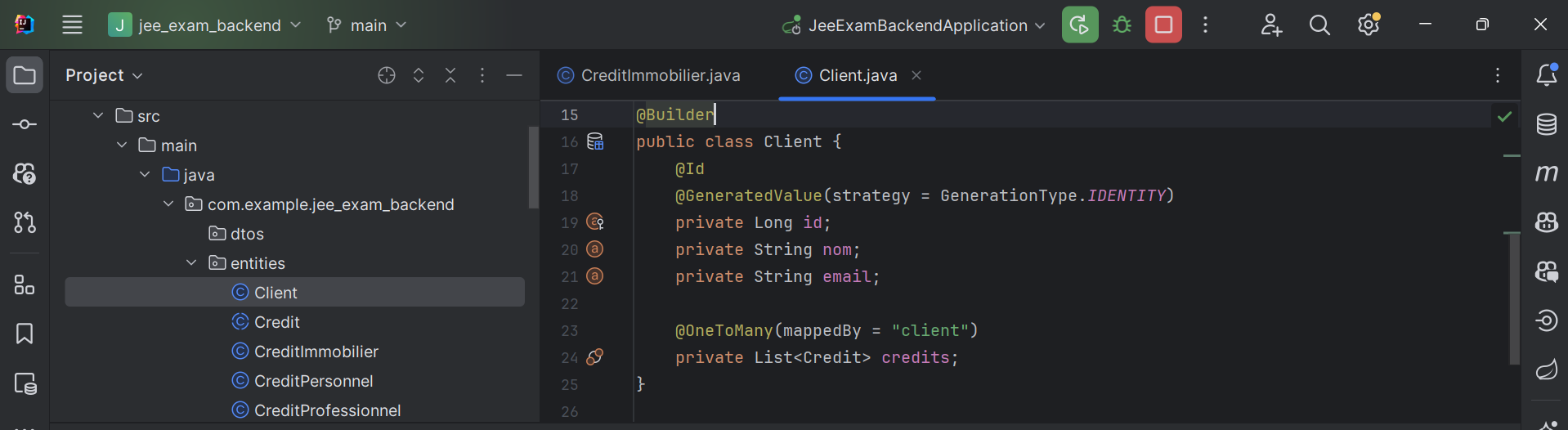
# Implémentation

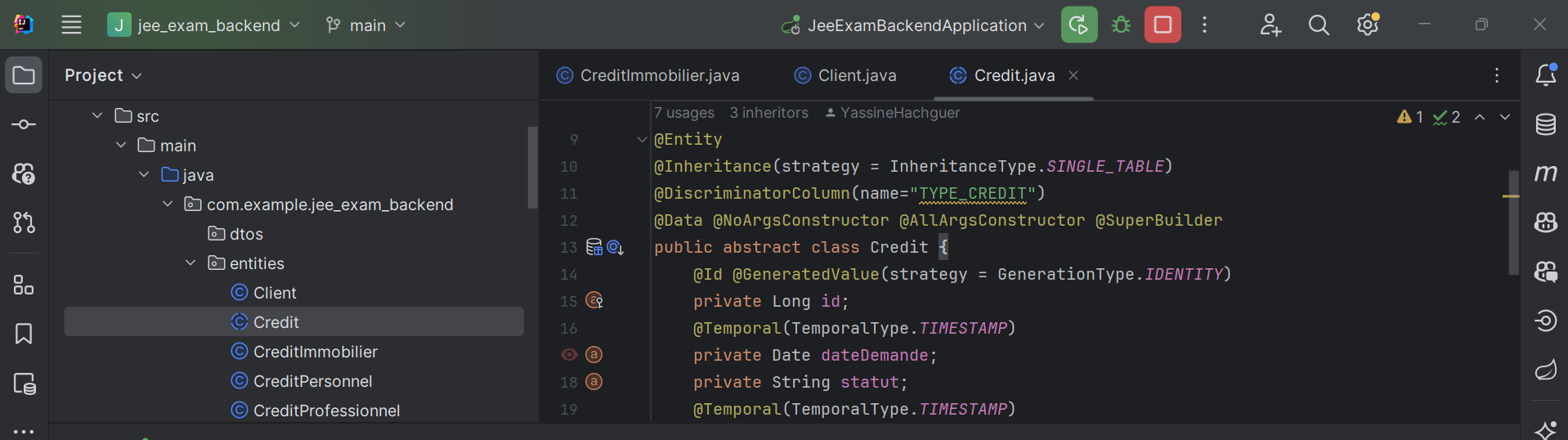
## Créer un projet Spring boot avec les dépendances requises. Les identifiants du projet GroupId, ArtifactId et le package de base doivent contenir votre nom et prénom

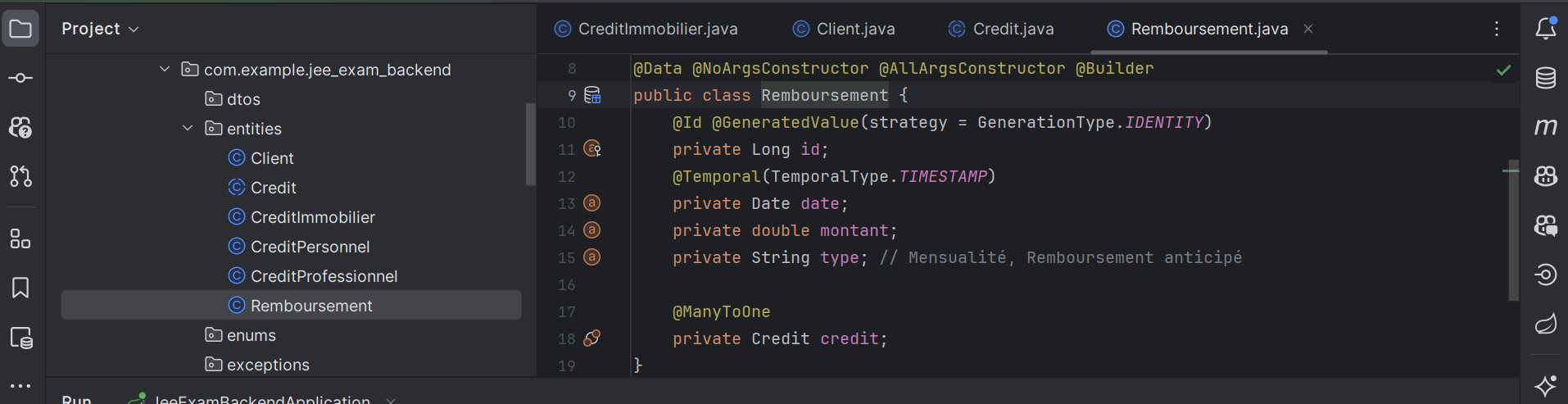


## Couche DAO

### Créer les entités JPA

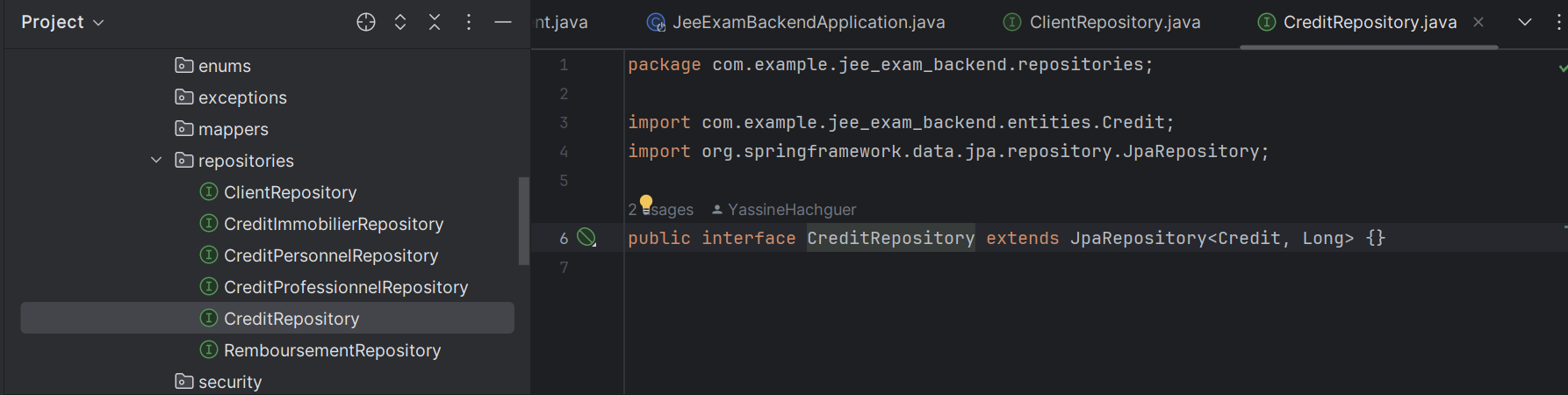


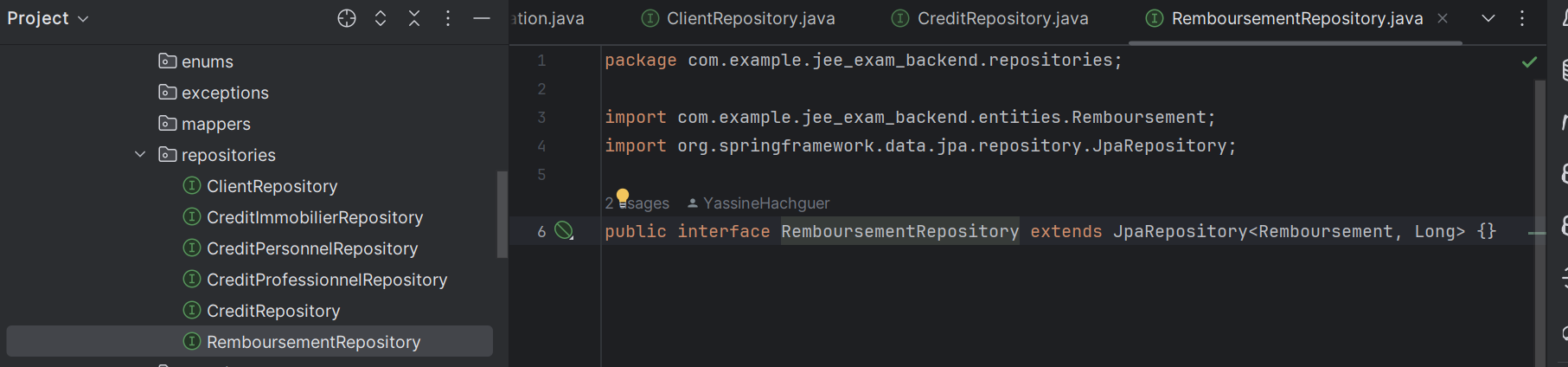




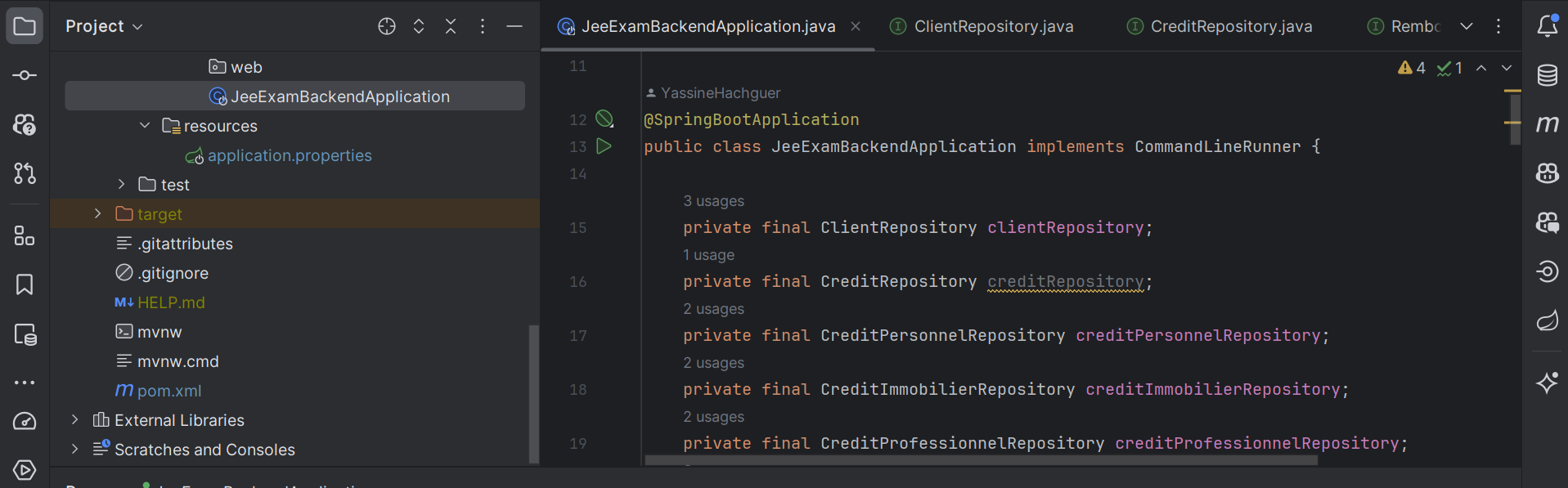
### Créer les interfaces JPA Repository basées sur Spring Data

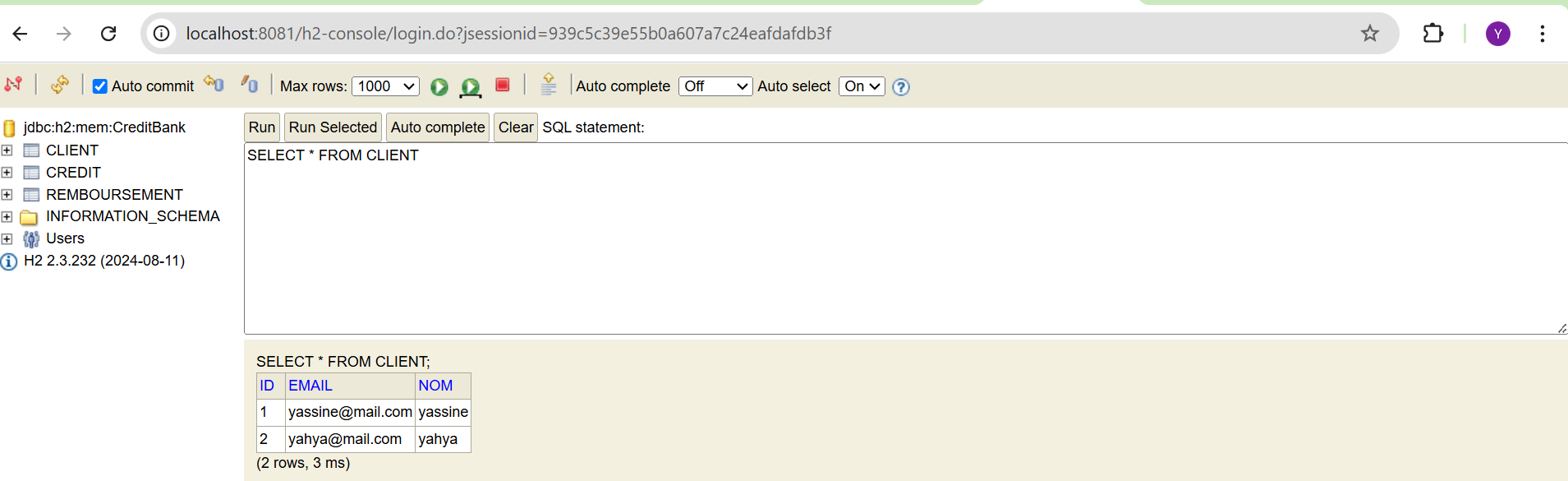


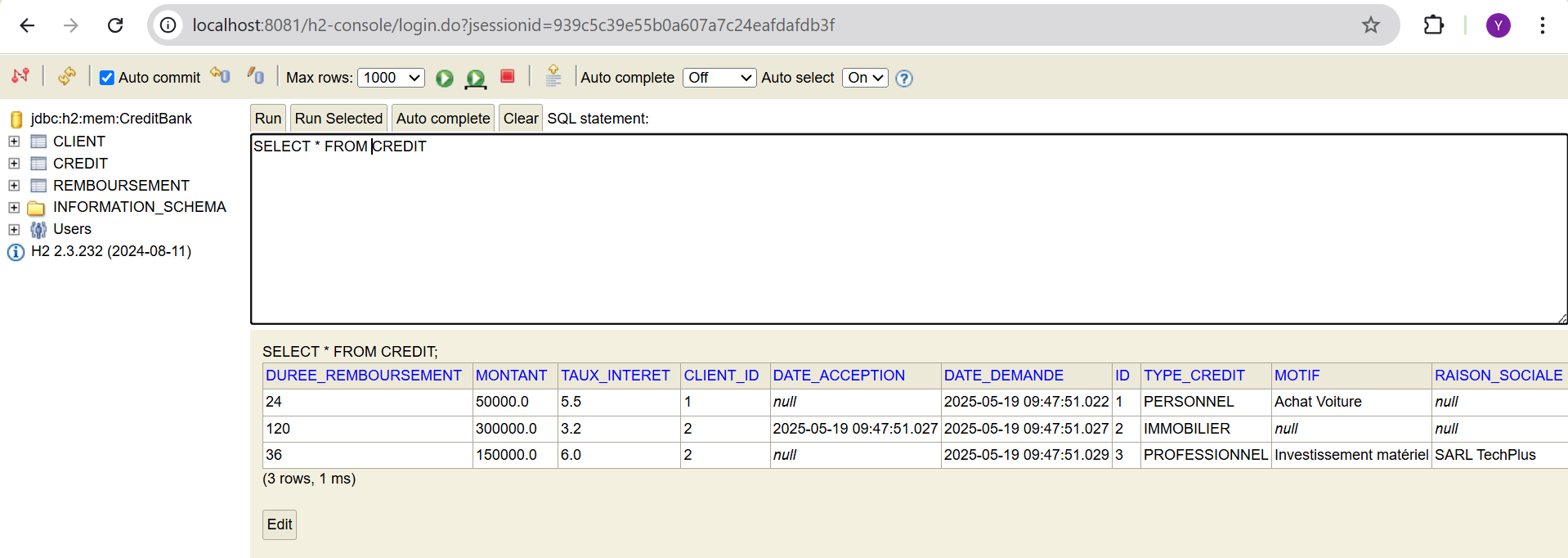


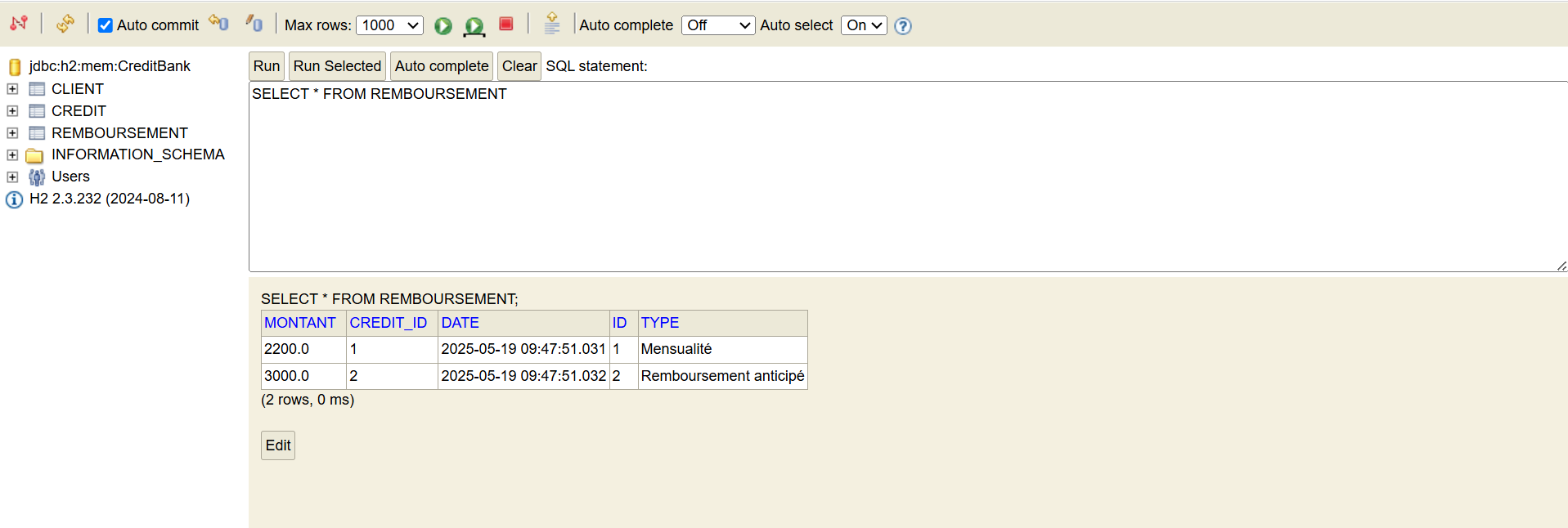


### Tester la couche DAO avec une application qui alimente la base de données avec quelques enregistrements de test.

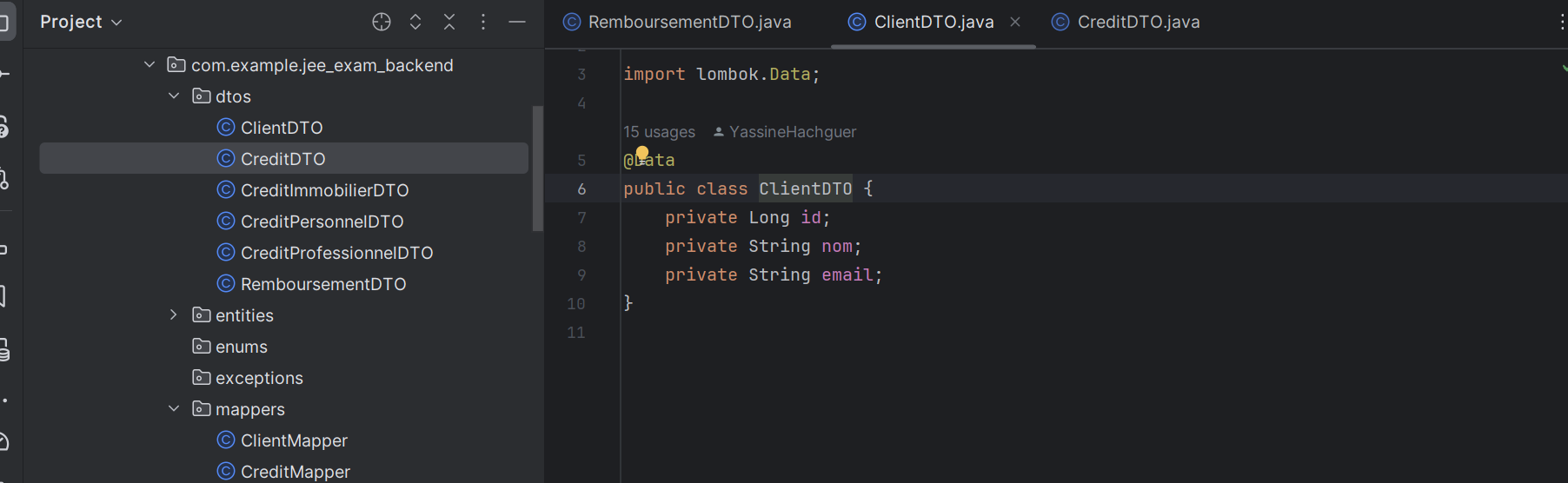


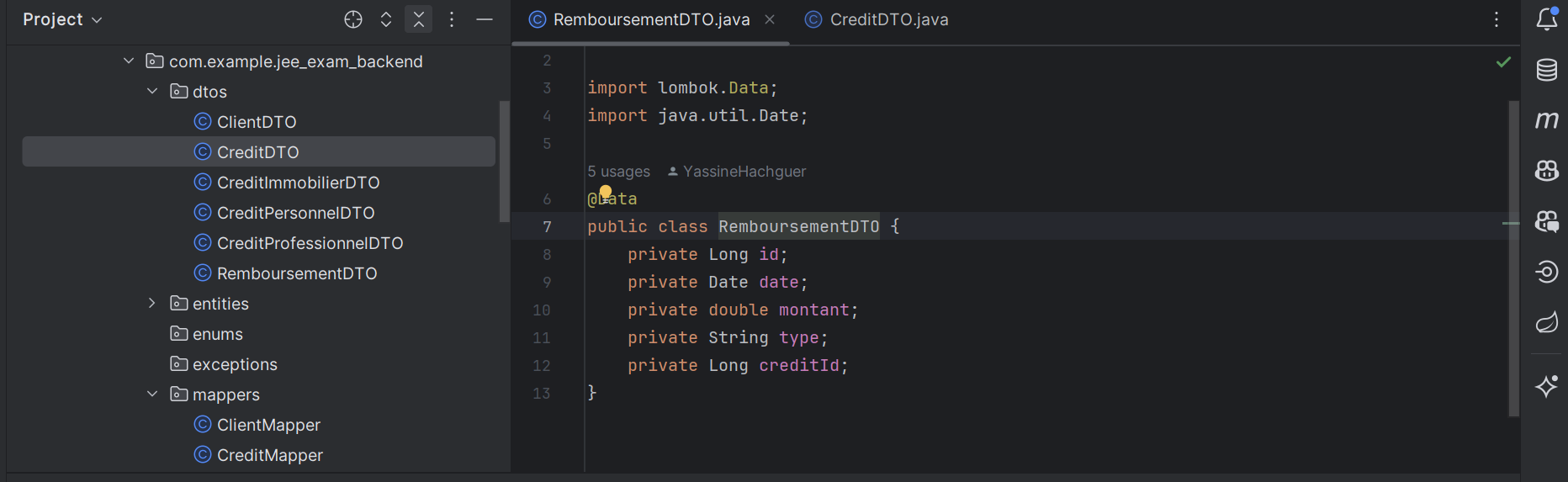




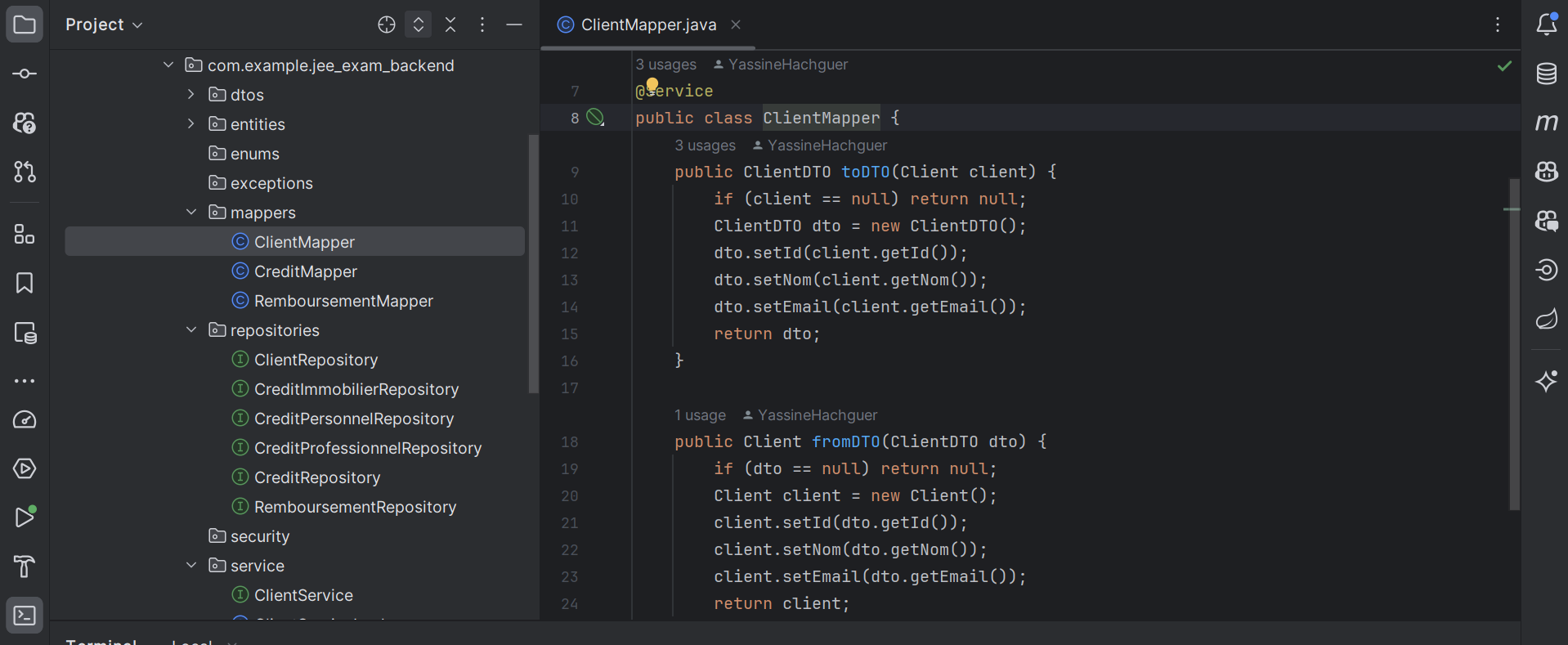


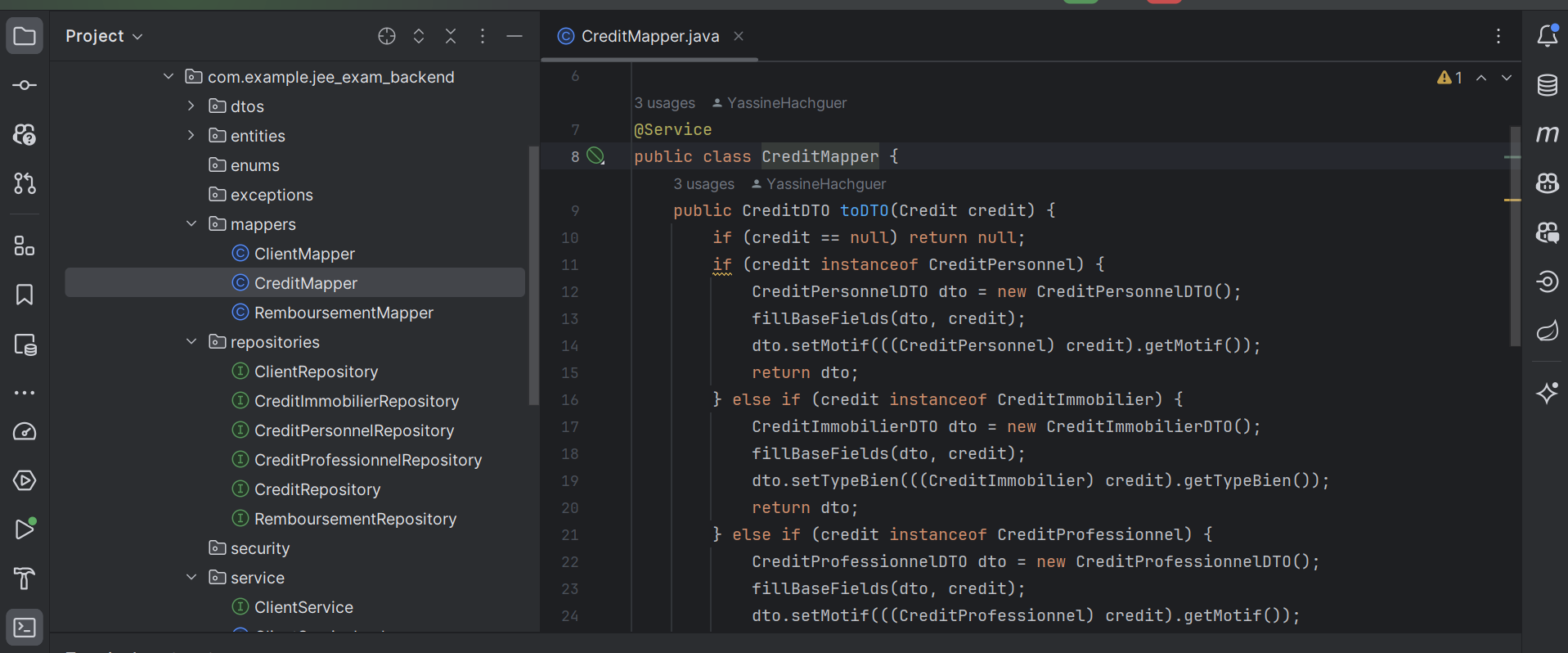
## Créer une couche service en créant les DTOs et les Mappers, en proposant les fonctionnalités que vous voyez importantes

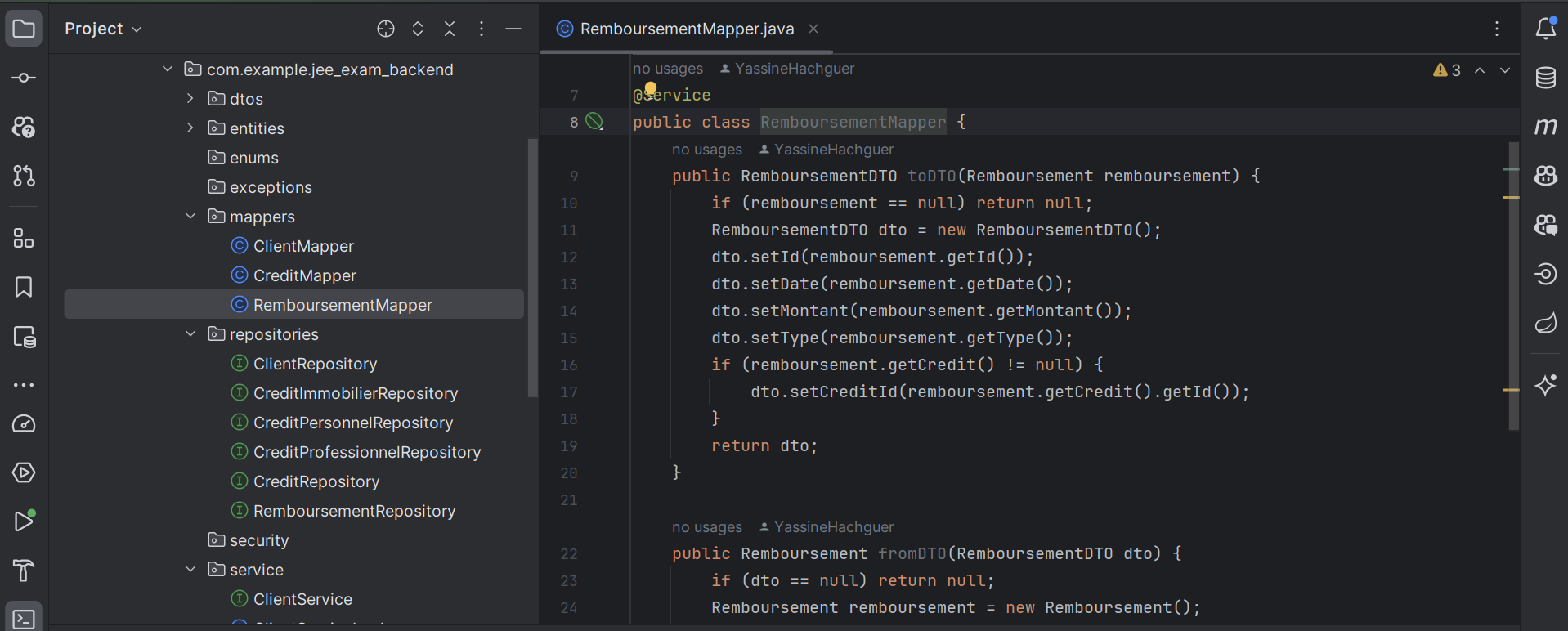




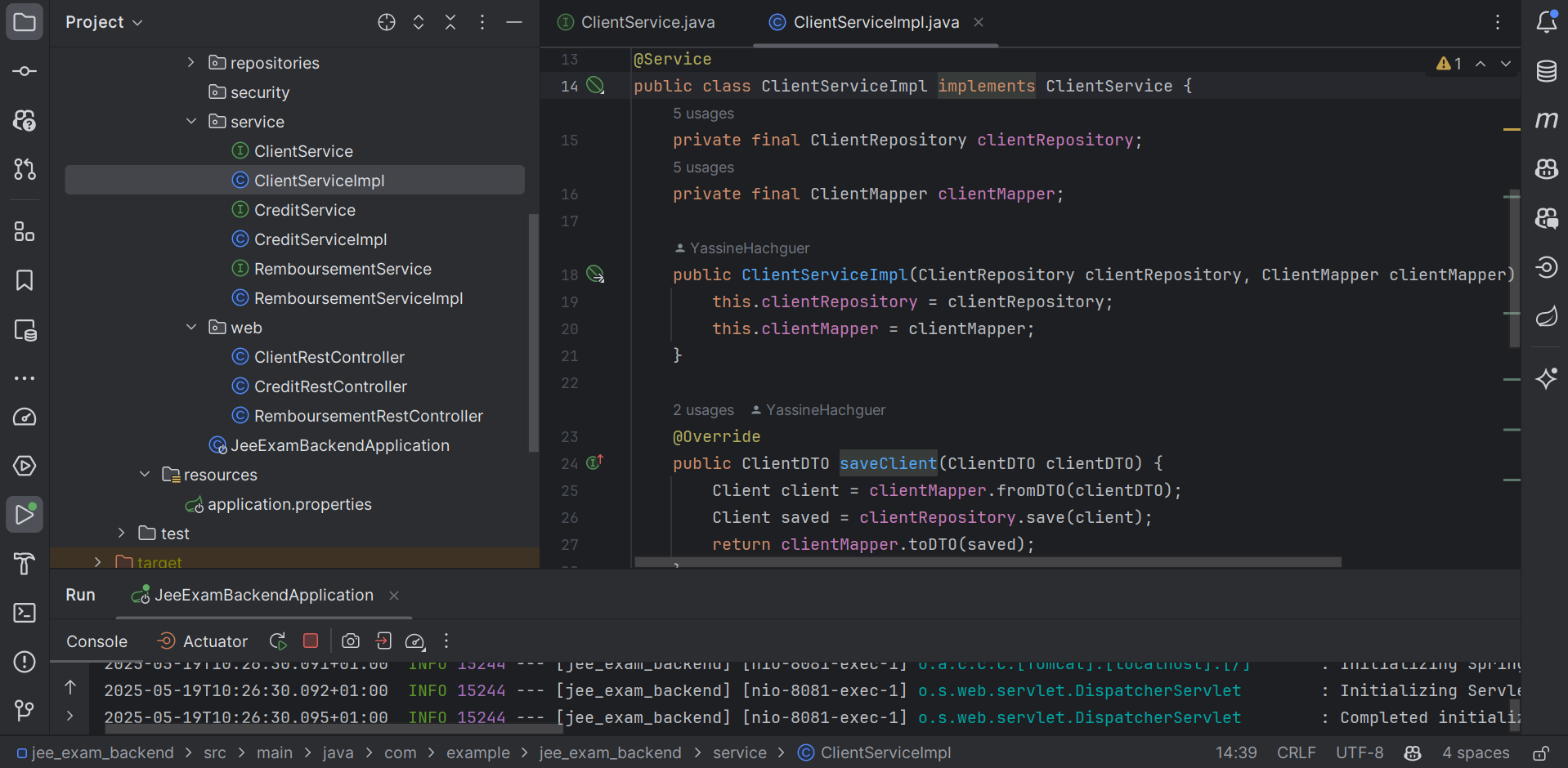


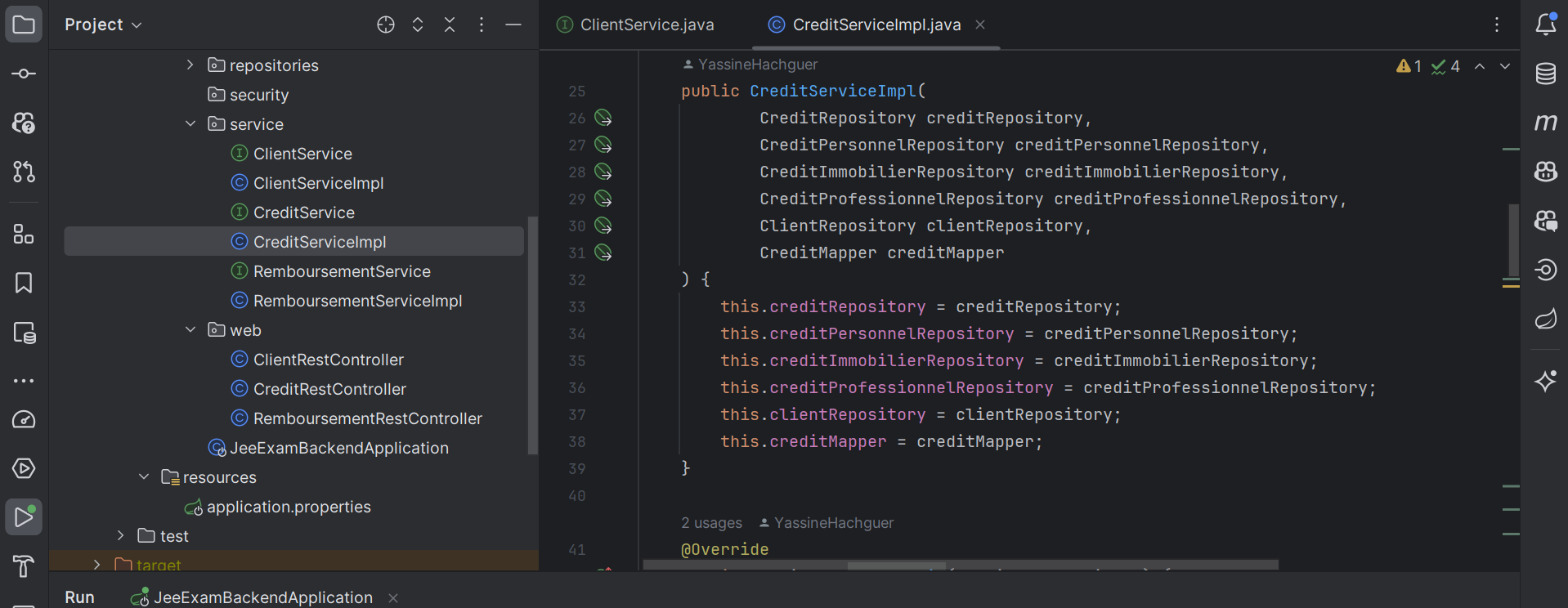


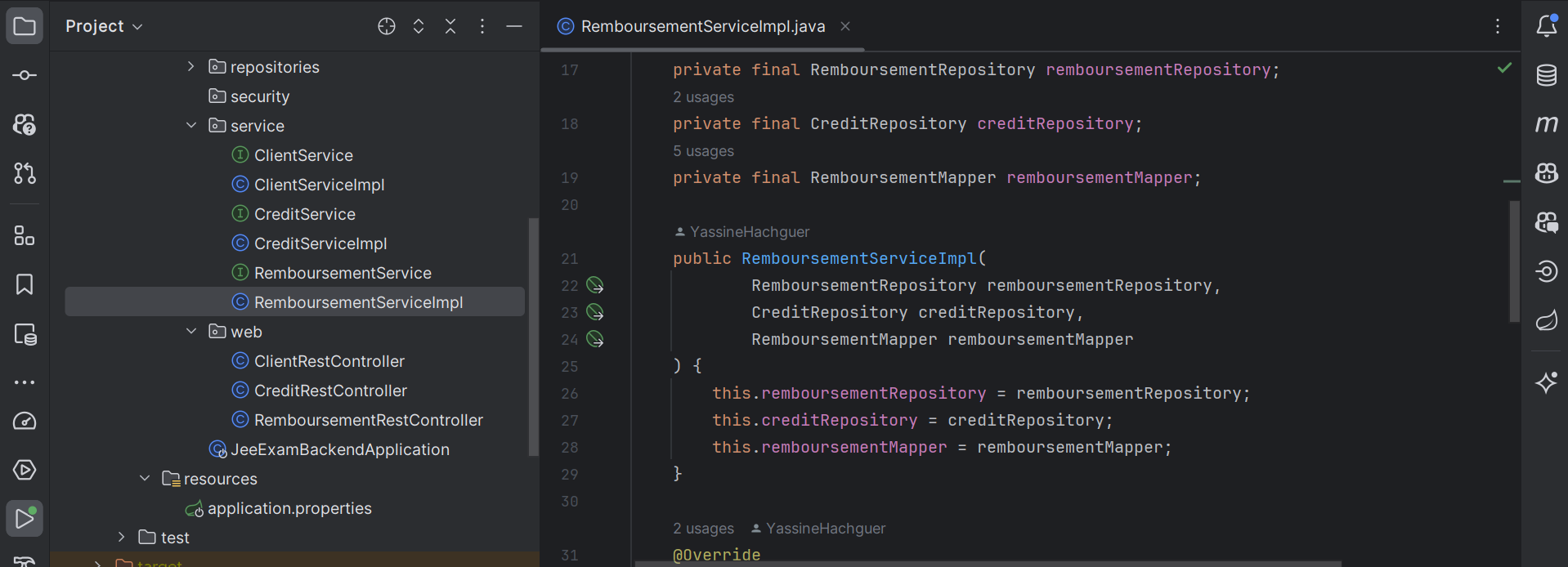


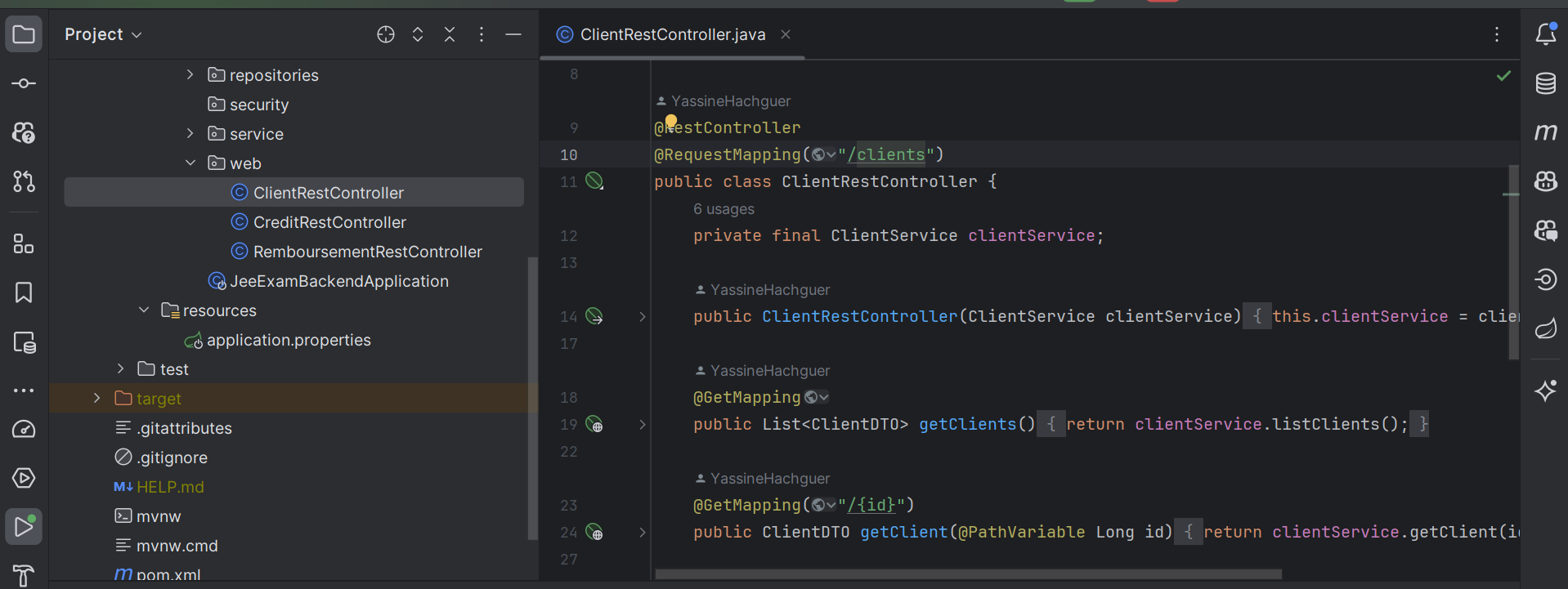


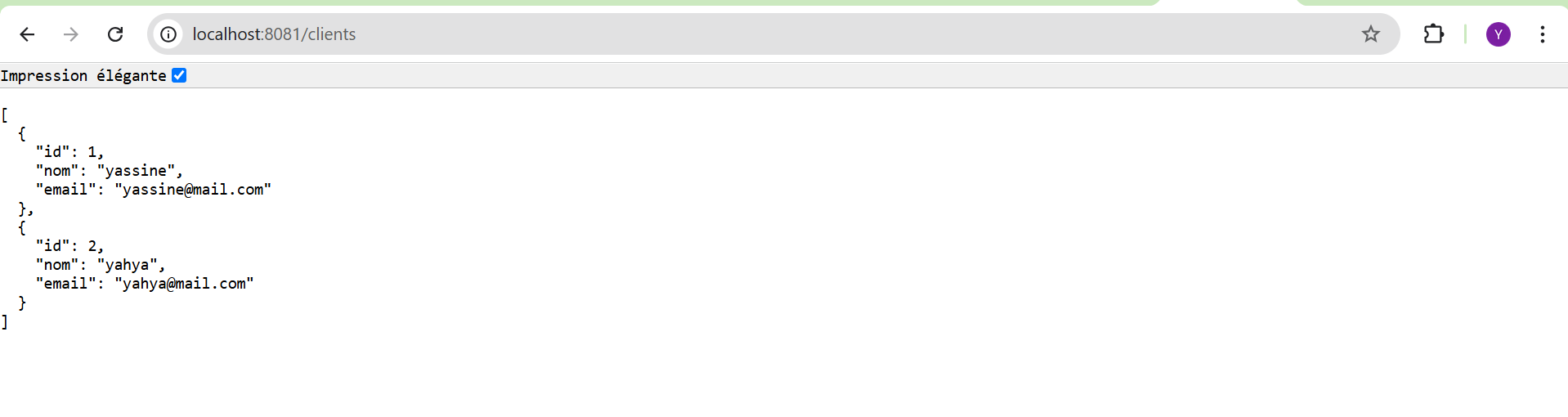
## Créer les Web services (Rest Controllers) en proposant les fonctionnalités que vous estimez importantes. Tester les REST API en générant la documentations SWAGGER (Open API Doc)

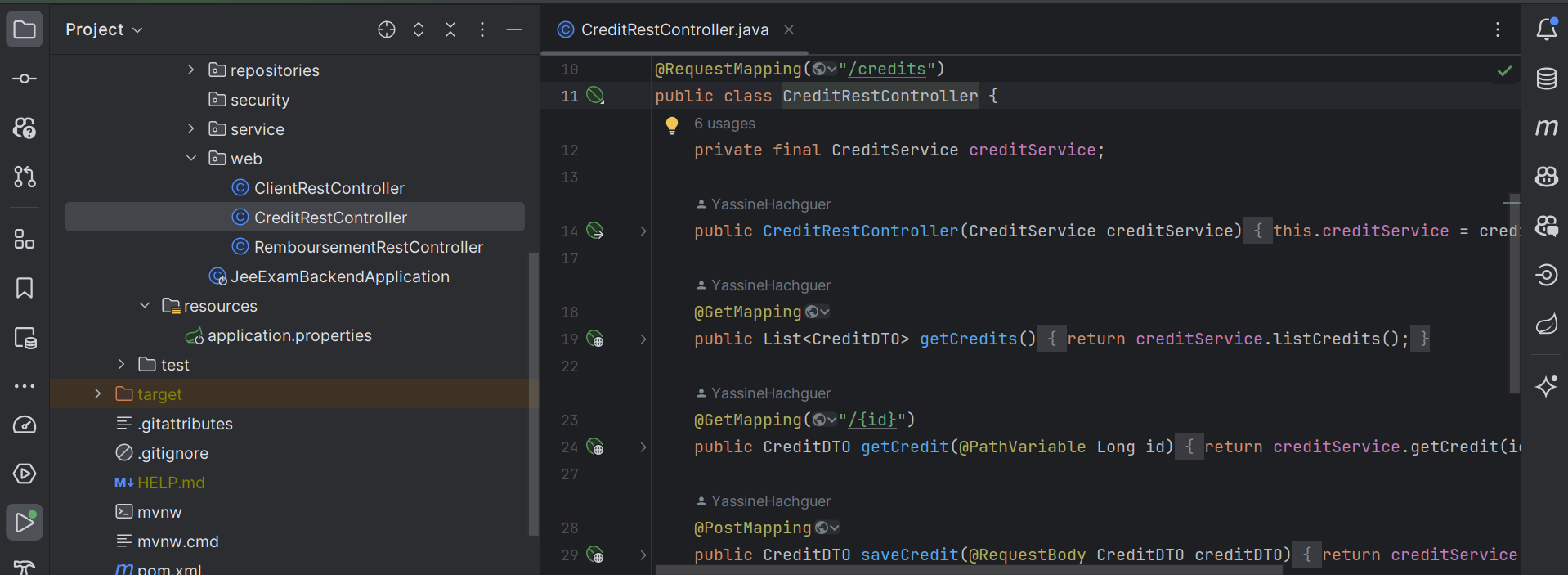


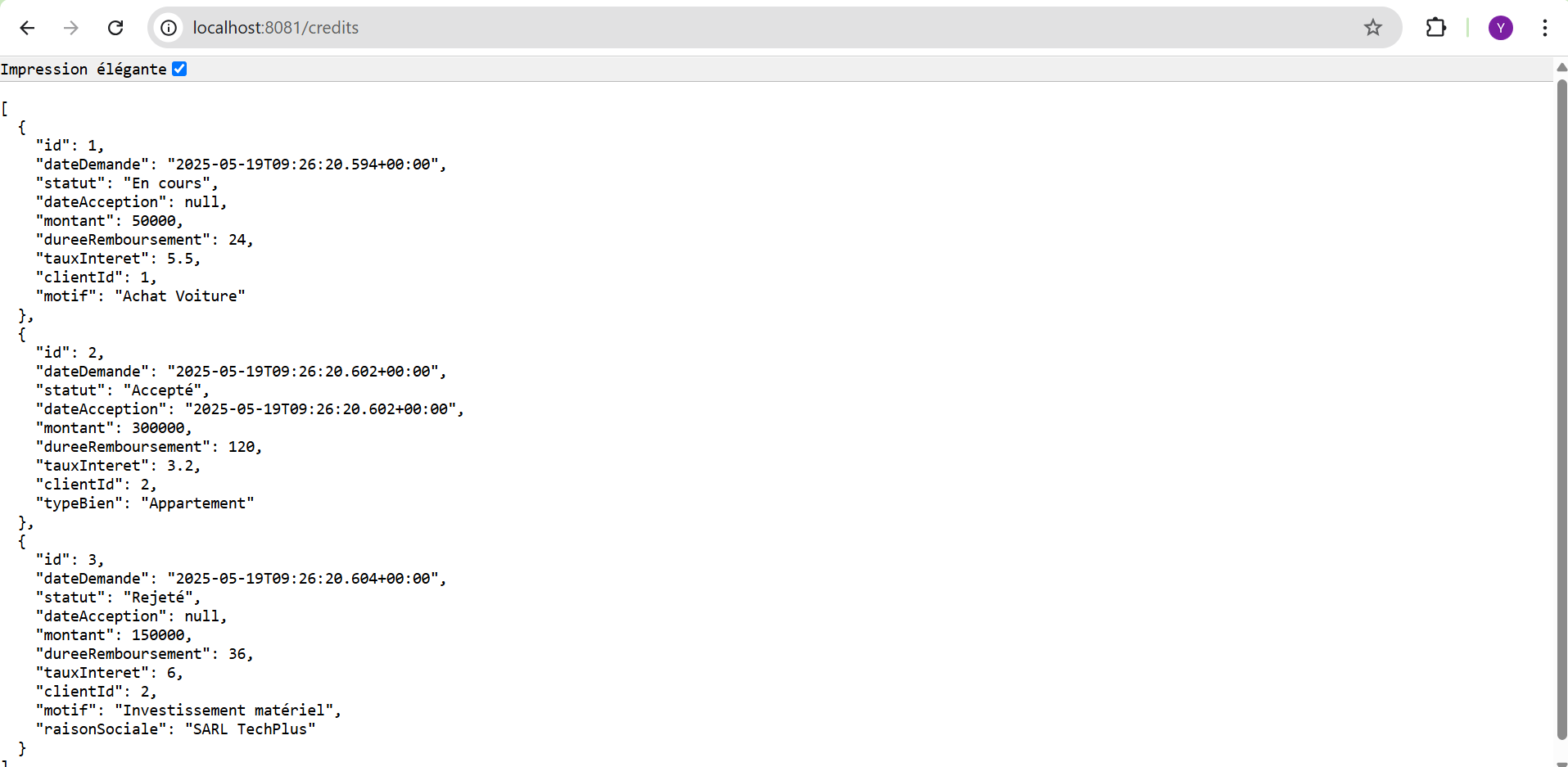


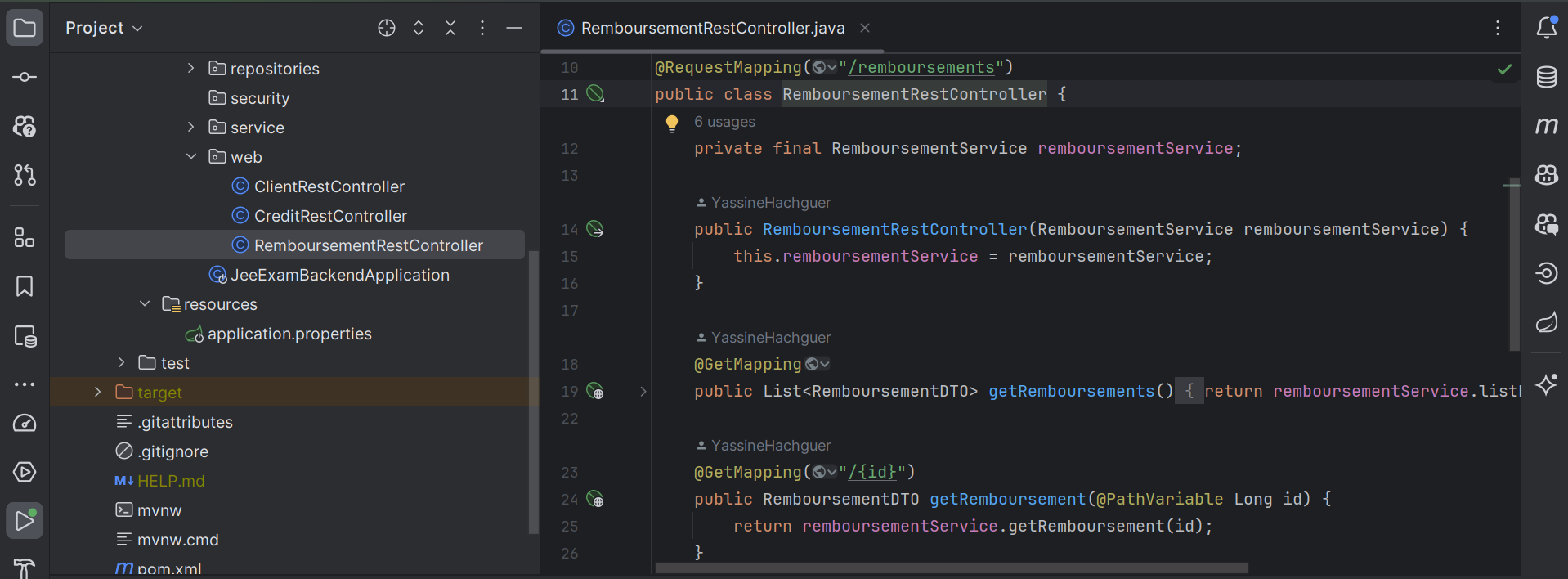


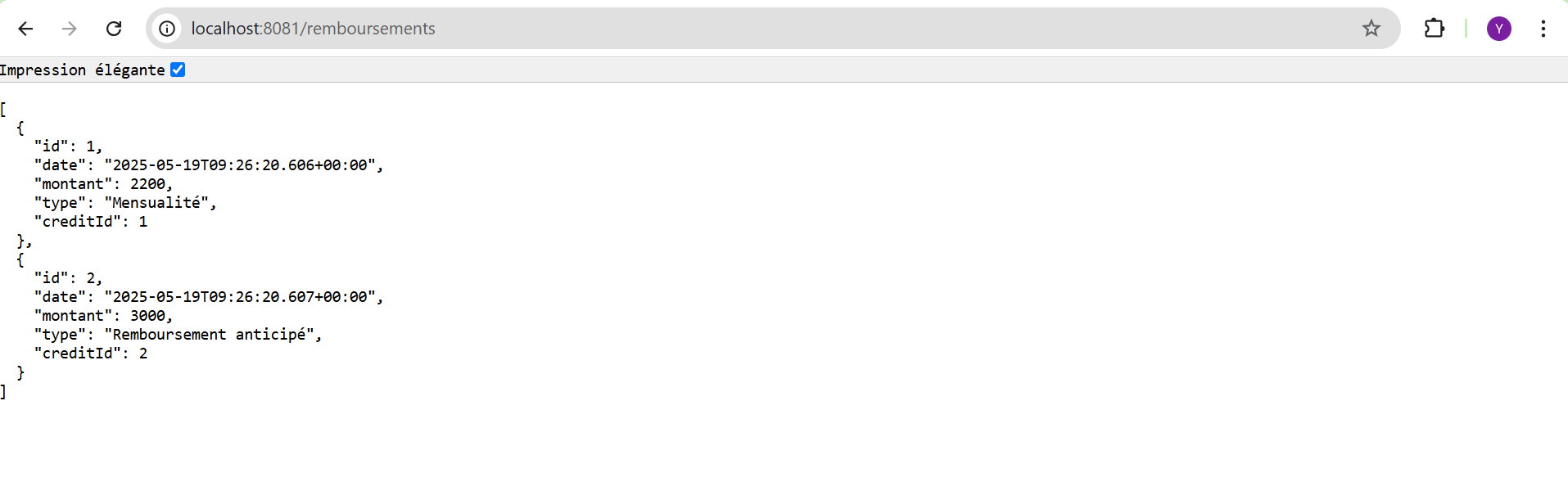


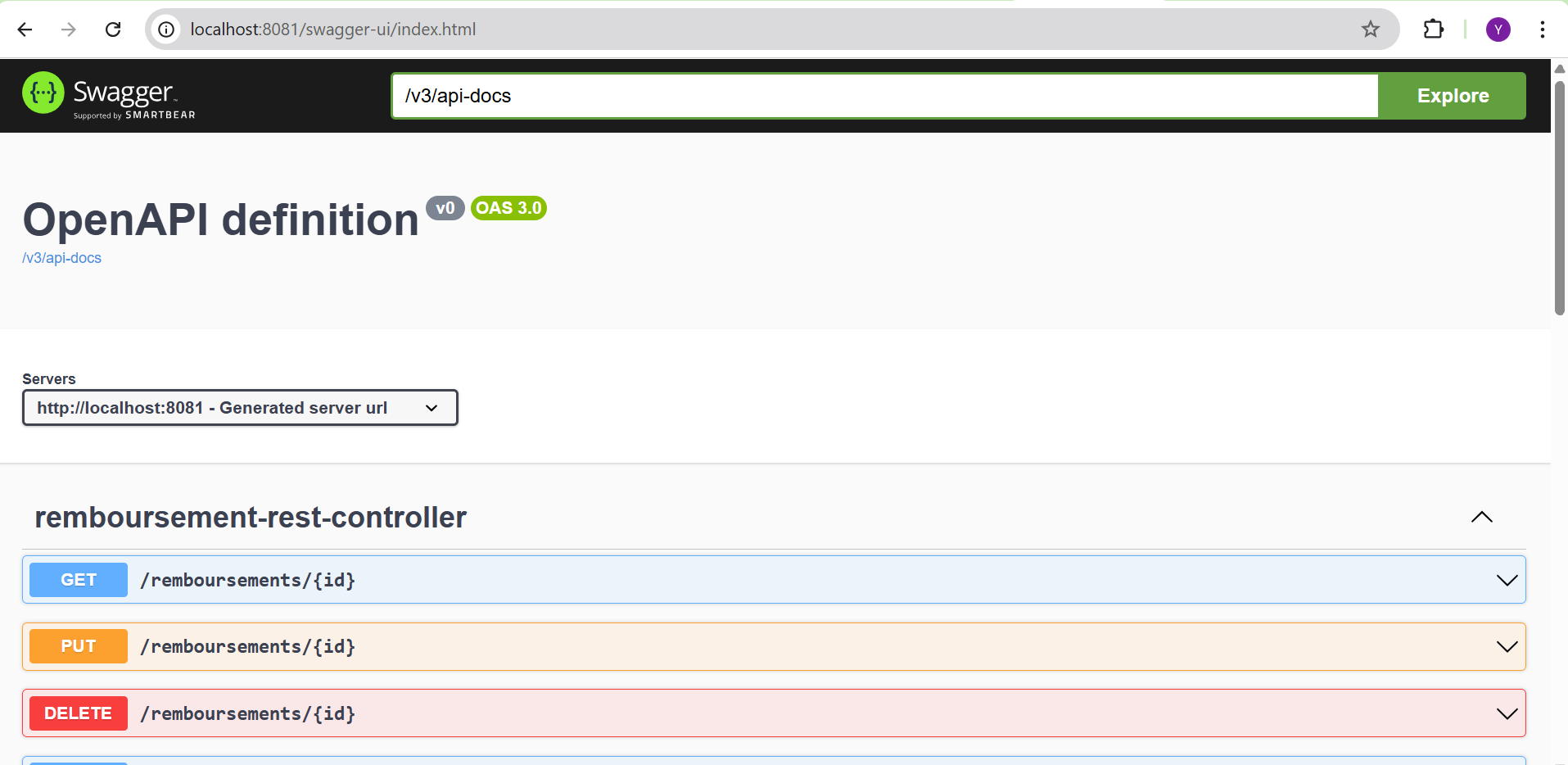


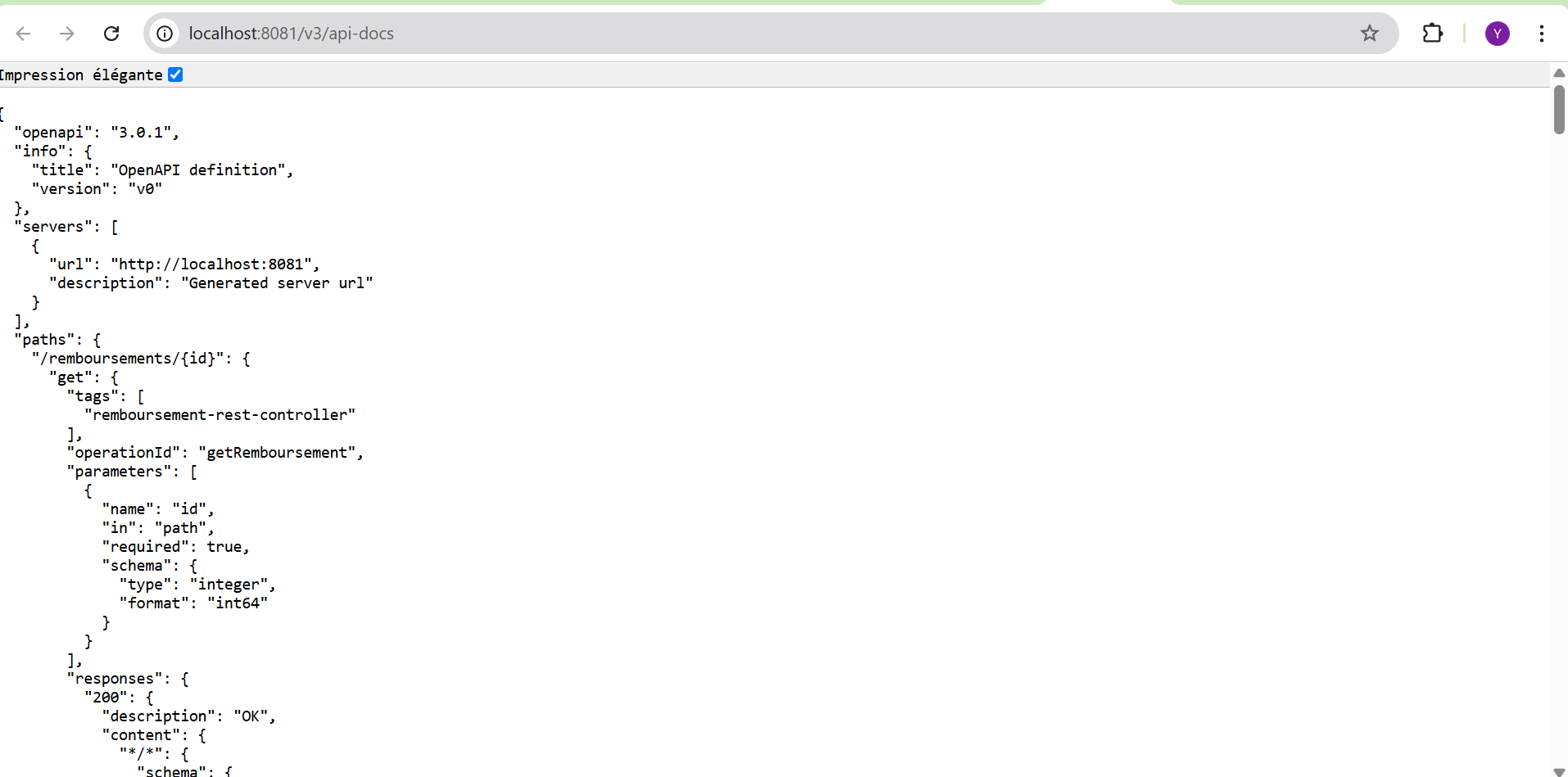










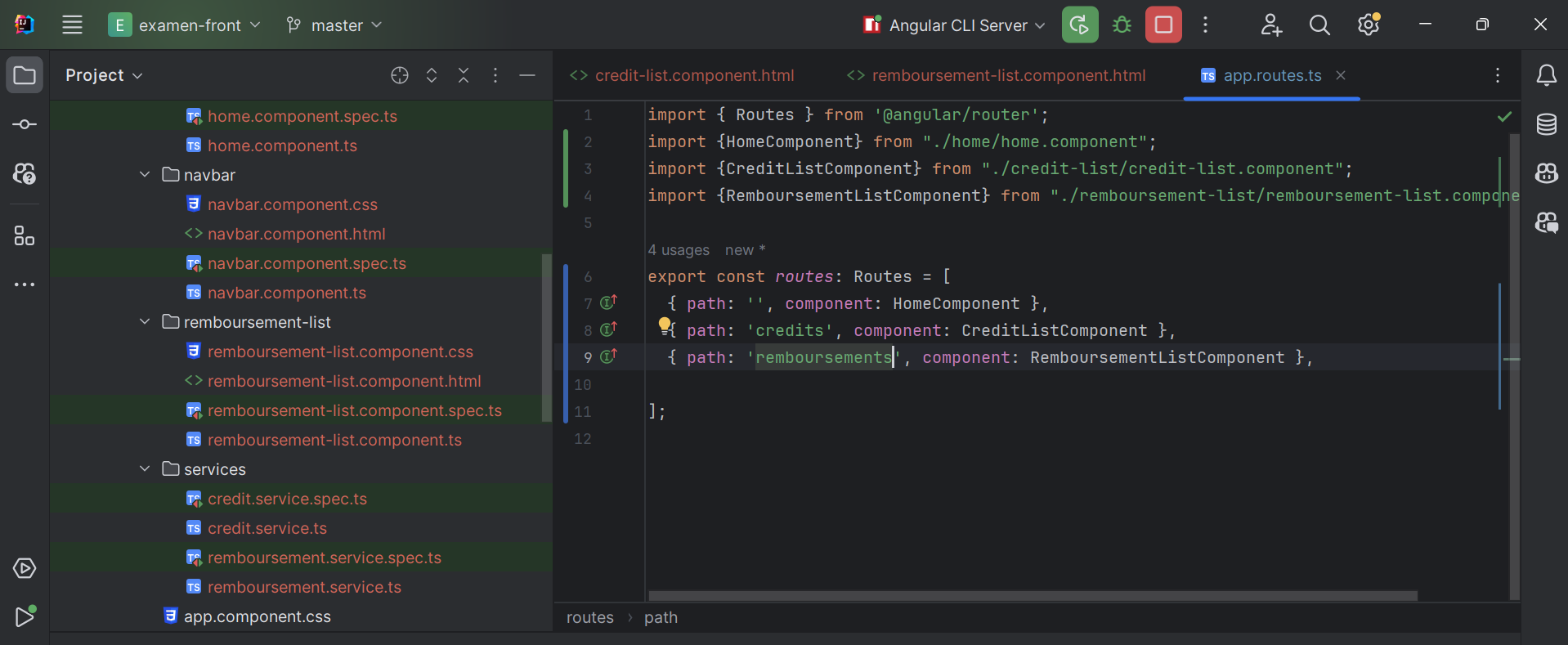


## Proposer une application frontend en utilisant Angular Framework

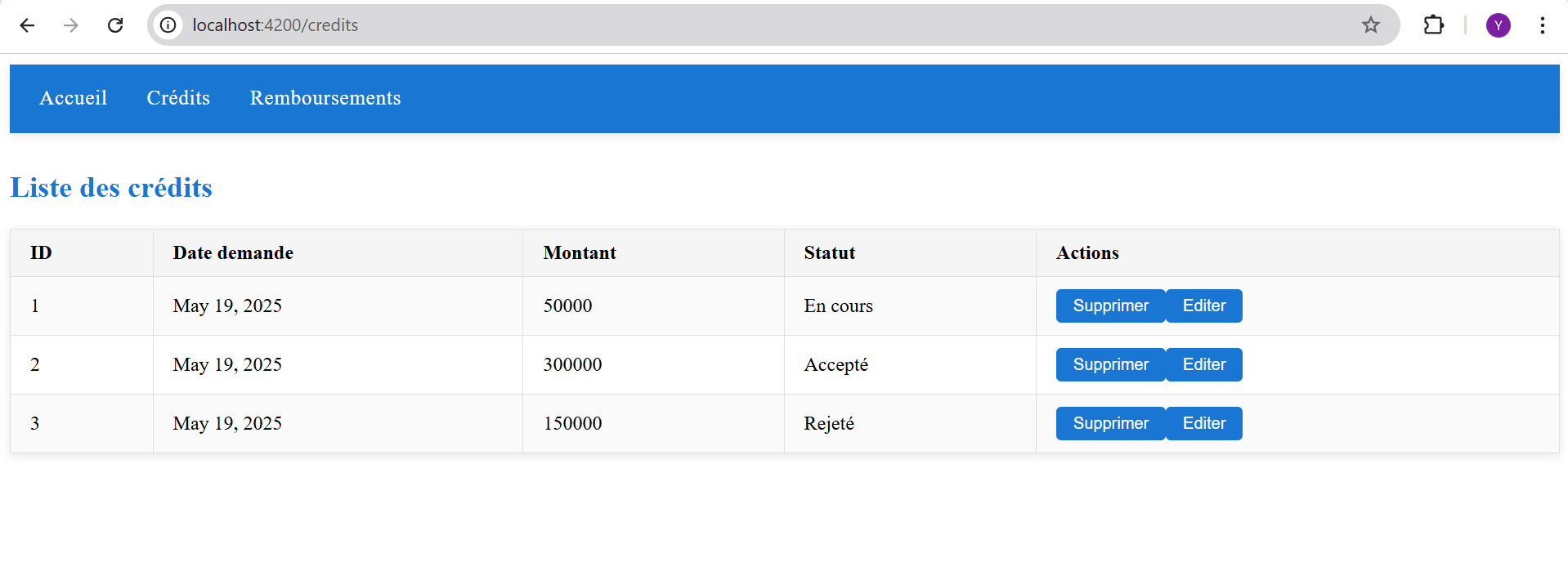
Pour la structure du projet on a 3 composants :

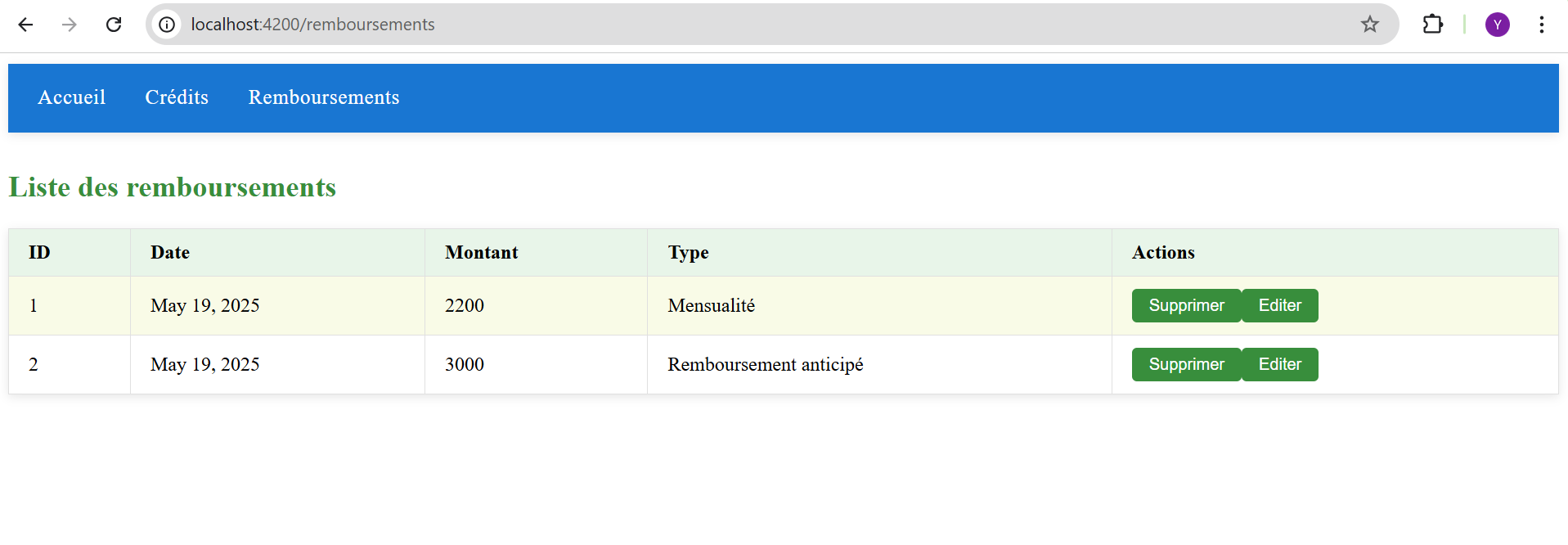
Home, Navbar, crédit-list et remboursement-list

Et pour gérer les requêtes on a 2 services









## 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\_CLIENT», « ROLE\_EMPLOYE » et « ROLE\_ADMIN » en choisissant des autorisations appropriées à ses rôles





