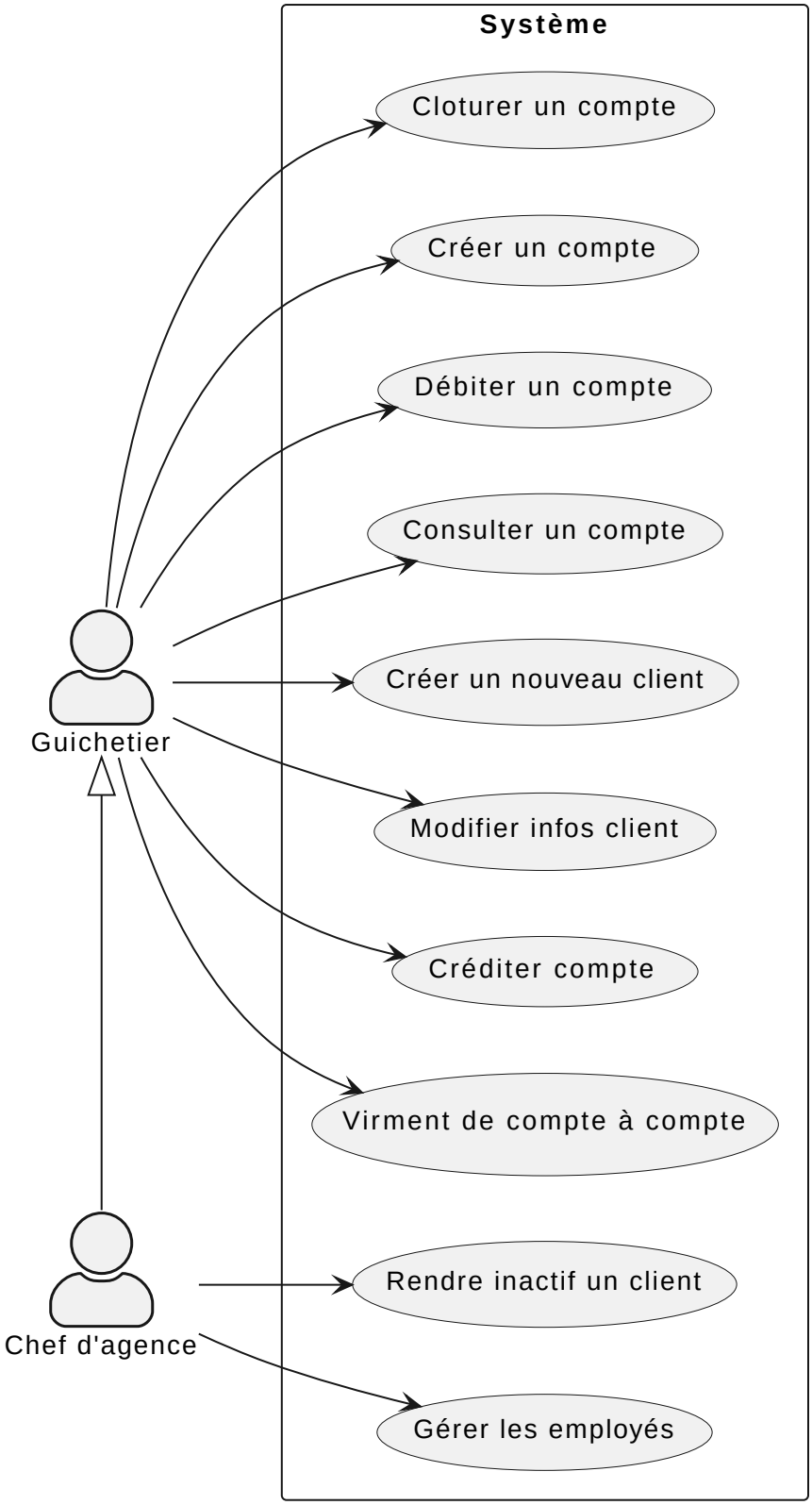
Documentation Technique V1

Sommaire

1. Présentation de l’application

1.1. Use Case global de l’application



Dans le diagramme des Use Case ci-dessus, nous pouvons observer qu'il y a deux types d'utilisateurs présents dans l'application, le guichetier et le chef d'agence.

Le guichetier est l'employé "de base" du système. Il peut réaliser certaines opérations:

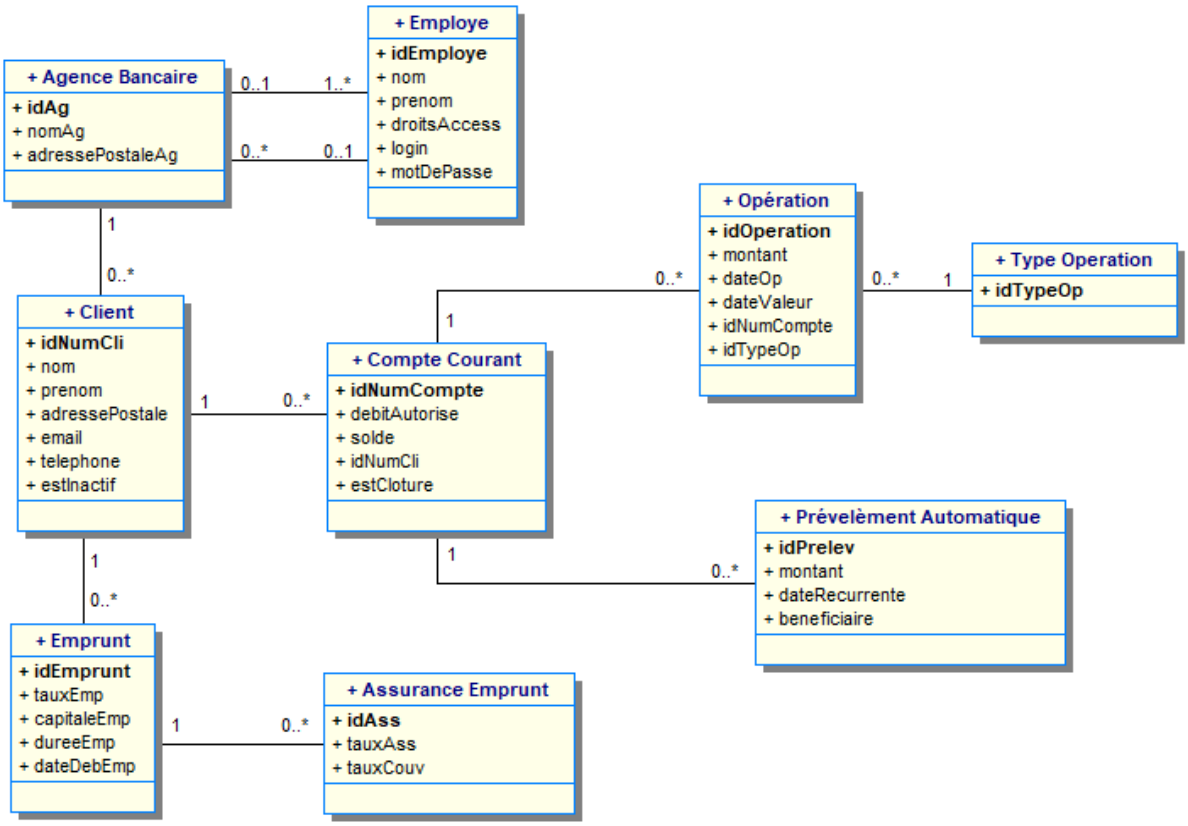
- Modifier les informations client (adresse, téléphone, ...)
- Créer un nouveau client
- Consulter un compte

- Débiter un compte (BD)
- Créditer un compte
- Cloturer un compte
- Créer un compte
- Effectuer des virements compte à compte

Le chef d'agence est un type d'utilisateur héritant de guichetier, il peut effectuer toutes les opérations de celui-ci en plus de:

- rendre inactif un client.
- Gérer les employés, ou faire le CRUD (Create Read Update Delete)
  - Créer un employé
  - Consulter un employé
  - Modifier un employé
  - Supprimer un employé

1.2. Diagramme de classe de la base de données



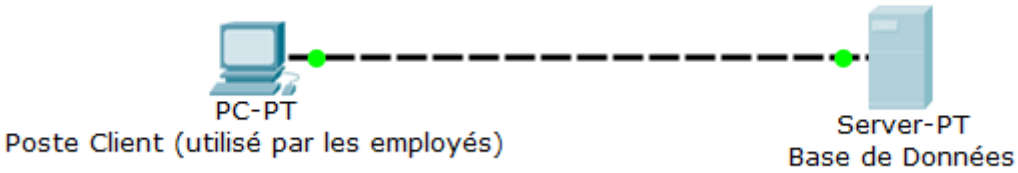
- Une agence bancaire est composée de plusieurs employés.
- Un employé peut être chef d'agence d'une ou plusieurs agences.
- Un employé peut être un chef d'agence ou un guichetier selon les droits d'accès qu'il possède
- Une agence bancaire est composée de plusieurs clients.

- Un client possède des informations qu'il est possible de modifier.
- Un client peut devenir inactif.
- Un client peut effectuer des opérations sur un compte par l'intermédiaire des employés (débit, crédit, emprunt).
- Un compte contient des informations auxquelles il est possible d'accéder.
- Un client peut avoir plusieurs comptes.
- Un emprunt peut être assuré.

Pour l'instant l'application ne possède que certaines des fonctionnalités du diagramme ci-dessus.

2. Architecture

2.1. Architecture générale (poste client, serveurs, ...) et rôle de chaque élément



Les postes client exécutent l'application faite en javaFX et se connectent à la base de données oracle, permettant de manipuler les données des clients et des employés.

2.2. Ressources externes (.jar, ...) utilisées et rôles

L'application utilise la librairie JavaFX pour permettre de disposer d'une interface graphique.

Elle utilise également la librairie externe ojdbc6.jar pour interagir avec la base de données du système des agences bancaires et exécuter des instructions SQL avec Java.

2.3. Structuration en packages de l'application documentée. Principes retenus pour cette structuration

L'application est structurée en plusieurs packages. Le modèle retenu est le modèle est le modèle MVC (modèle, view, controller). De plus on regroupe les packages en deux catégories:

Application :

- Package tools : Contient contenant les outils de l'application.

- Package view : Contient les classes de l'interface graphique
- Package control : Contient les fonctionnalités de l'application

Cette catégorie contient les données des différentes classes pour pouvoir utiliser les différentes fonctionnalités de l'application.

Model :

- Package data : Contient les classes de données.
- Package orm : Contient les classes de gestion des données.
- Package orm.exception : Contient les classes d'exceptions.

Il contient des classes de base de données correpondant à certaines tables de la base de données. Il permet également de manipuler les données de la base de données (classe orm).

3. Fonctionnalités

3.1. Modifier les informations d'un client

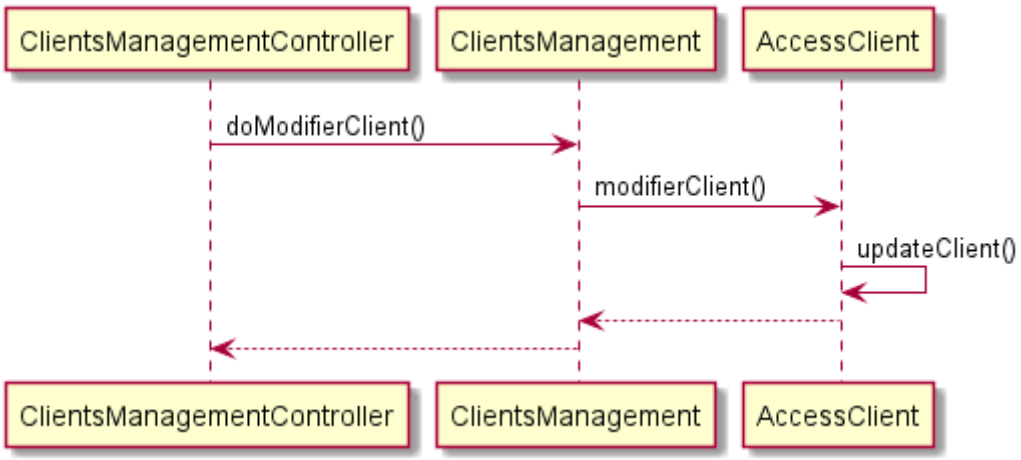
Partie "modifier infos client" du diagramme des use case

Partie du diagramme de classes données nécessaires :

- En lecture : client
- En mise à jour : client

cf. doc. utilisateur "Comment modifier les informations personnelles d'un client ?"

Diagramme de séquence:



3.2. Créer un nouveau client

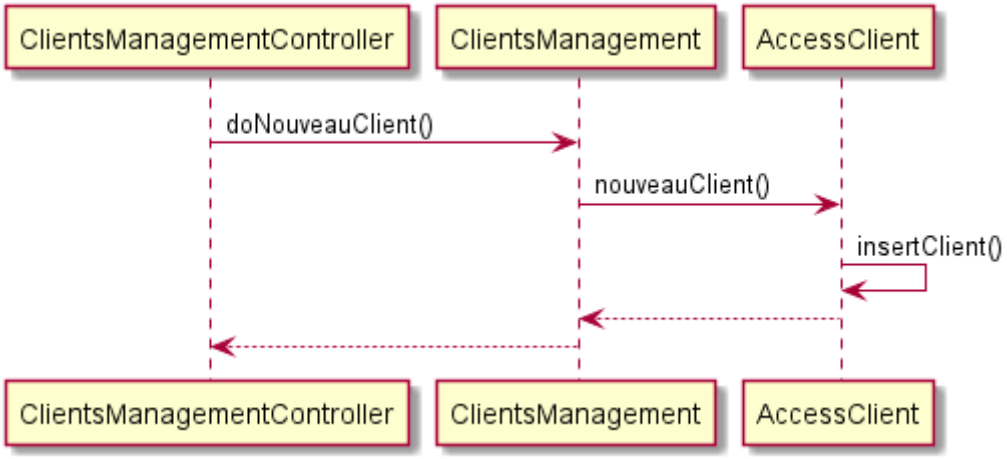
Partie "créer un nouveau client" du diagramme des use case

Partie du diagramme de classes données nécessaires :

- En lecture : Client
- En mise à jour : Client

cf. doc. utilisateur "Comment ajouter un client ?"

Diagramme de séquence :



3.3. Consulter un compte

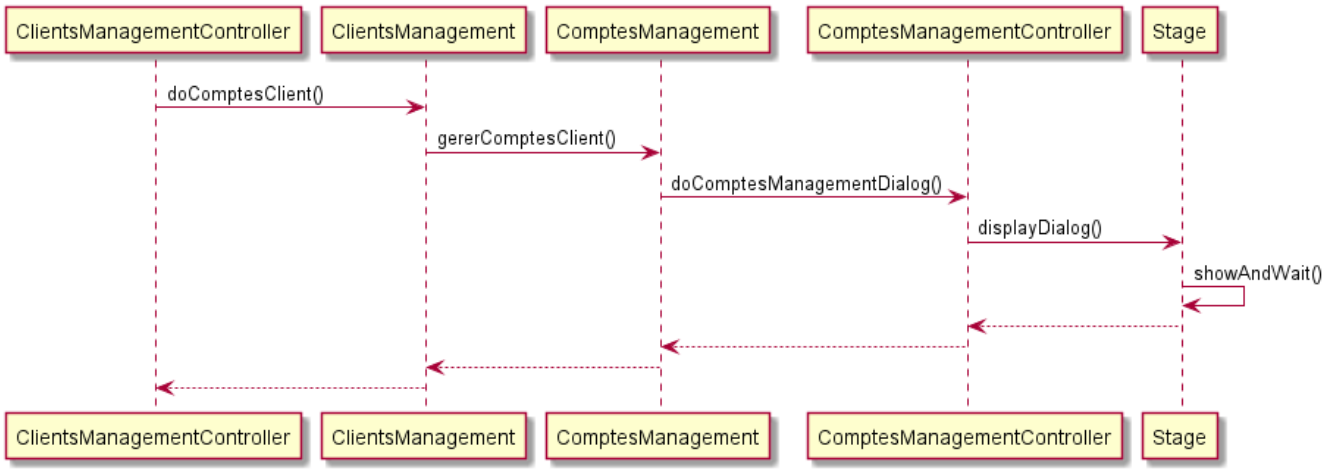
Partie "consulter un compte" du diagramme des UC

Partie du diagramme de classes données nécessaires :

- En lecture : Client, Compte Courant, Opération, Type Opération

cf. doc. utilisateur "Comment accéder aux comptes d'un client ?"

Diagramme de séquence :



3.4. Débiter un compte

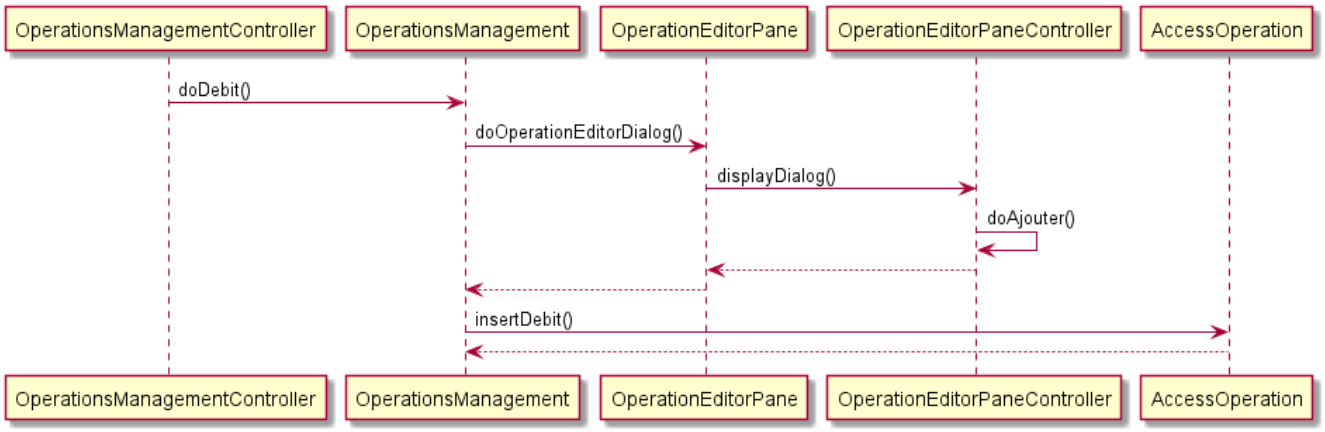
Partie "Débiter un compte" du diagramme des UC

Partie du diagramme de classes données nécessaires :

- En lecture : Client, Compte Courant, Opération, Type Opération
- En mise à jour : Opération

cf. doc. utilisateur "Comment enregistrer un débit manuellement ?"

Diagramme de séquence :



3.5. Rendre inactif un client

Partie "rendre inactif un client" du diagramme des UC

Partie du diagramme de classes données nécessaires :

- En lecture : Client
- En mise à jour : Client

cf. doc. utilisateur "Comment modifier les informations personelles d’un client ?"

Diagramme de séquence :

[DS Rendre Inactif] | *images/DS-Rendre-Inactif.png*

3.6. Créditer un compte

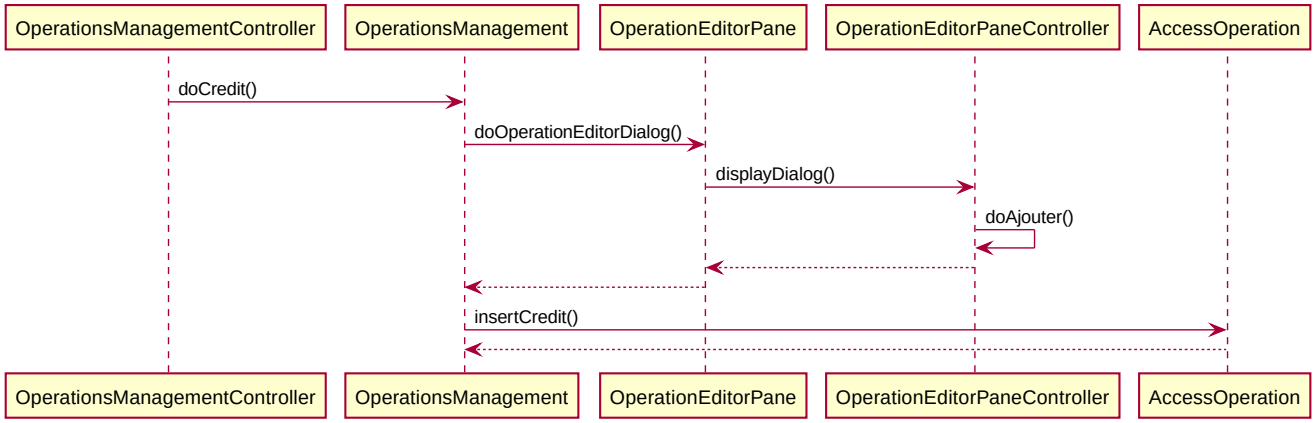
En cours de développement

Partie "Créditer un compte" du diagramme des UC

Partie du diagramme de classes données nécessaires :

- En lecture : Client, Compte Courant, Opération, Type Opération
- En mise à jour : Opération

Diagramme de séquence:



cf. doc. utilisateur "Comment enregistrer un crédit manuellement ?"

3.7. Cloturer un compte

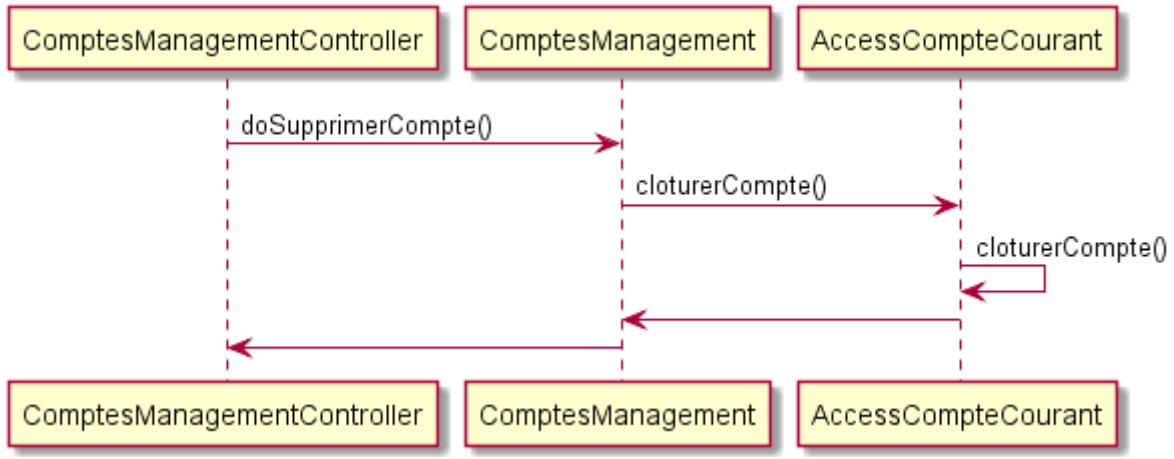
Partie "cloturer un compte" du diagramme des UC

Partie du diagramme de classes données nécessaires :

- En lecture : Client, Compte Courant
- En mise à jour : Client, Compte Courant

cf. doc. utilisateur "Comment cloturer un compte client déjà existant ?"

Diagramme de séquence :



3.8. Créer un compte

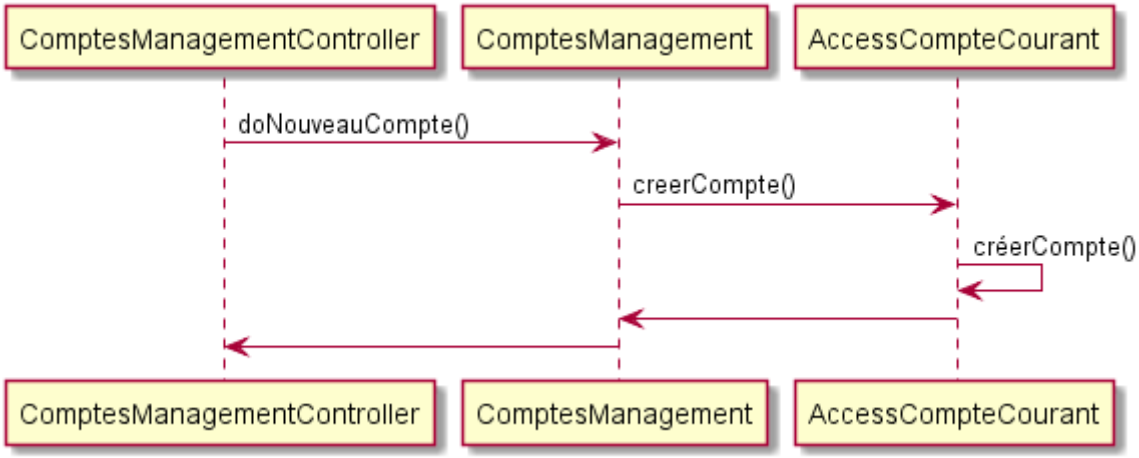
Partie "créer un compte" du diagramme des UC

Partie du diagramme de classes données nécessaires :

- En lecture : Client, Compte Courant
- En mise à jour : Client, Compte Courant

cf. doc. utilisateur "Comment créer un nouveau compte client ?"

Diagramme de séquence :



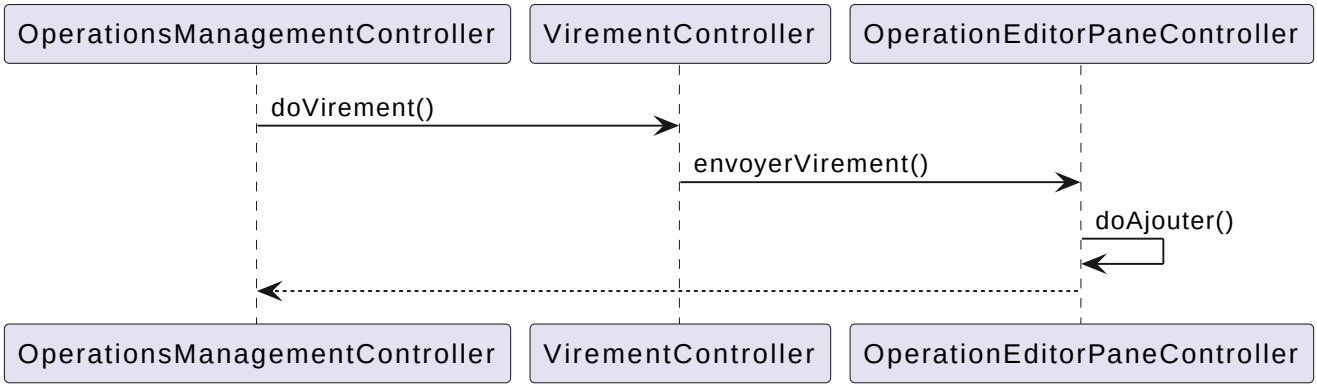
3.9. Effectuer des virements compte à compte

Partie "Virement de compte à compte" du diagramme des UC

Partie du diagramme de classes données nécessaires :

- En lecture : Client, Compte Courant, Opération, Type Opération
- En mise à jour : Opération

Diagramme de séquence:



cf. doc. utilisateur "Comment effectuer un virement ?"

3.10. Gérer les employés, ou faire le CRUD (Create Read Update Delete) :

En cours de développement

3.10.1. Créer un employé

3.10.2. Consulter un employé

3.10.3. Modifier un employé

3.10.4. Supprimer un employé