Les étapes de la création du projet

Titre du projet : mini banque interactive

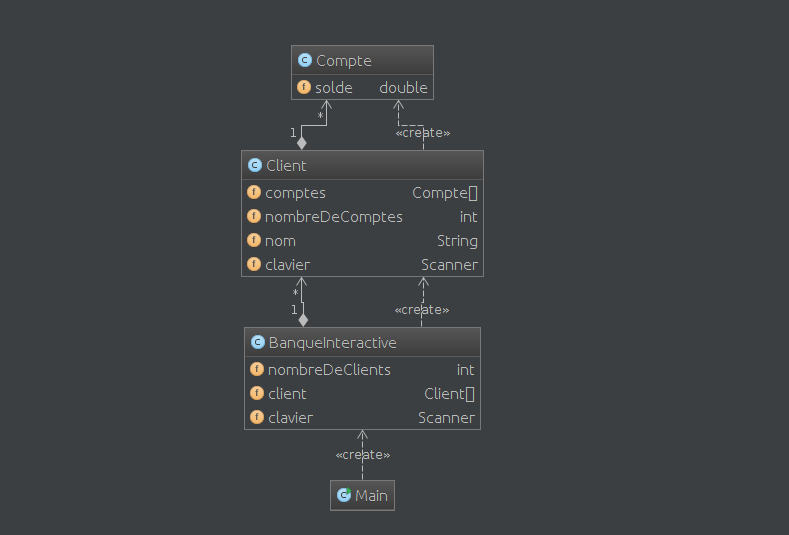
1. 1-Architecture Logiciel :

Identification des principaux acteurs.

Identification des principaux objets.

Représentation des objets un utilisant le diagramme de classe UML.

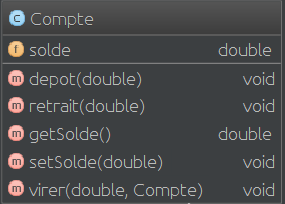
Détermination des relations entre les objets.



1. 2-Création des classes:

**1-La classe Compte :**

La première étape est d’écrire une classe Compte. Celle-ci contient 1 attributs privés ainsi que 5 méthodes publiques :



* 1. Attributs :

La classe Comptes contient 1 attribut :

-Solde

* 1. Setters et Getters :

La classe Comptes contient 1 getters et 1 setters :

getSolde () //pour obtenir la valeur du solde

setSolde() //pour modifier la valeur du solde

* 1. Méthodes :

La classe Comptes contient 3 méthodes :

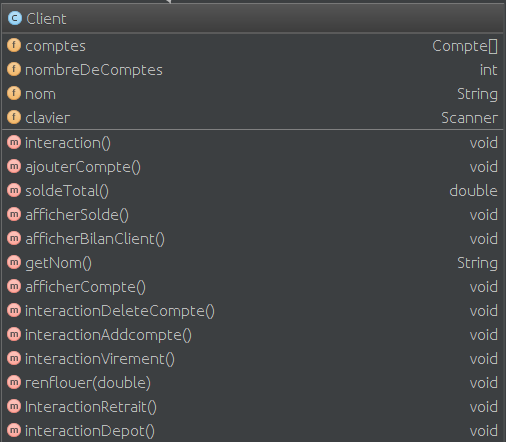
-depot() //pour faire un dépôt sur le compte.

-retrait() //pour faire un retrait sur le compte.

-virer () //pour faire un virement entre 2 comptes

**2-La classe Client :**

La classe Client possède 4 attributs privés ,1 constructeur et possède également 13 méthodes :



* 1. Attributs :

La classe Client possède 4 attributs :

-comptes

-nombreDeComptes

-nom

-clavier //pour la saisi du clavier

* 1. Setters et Getters :

La classe Client possède 1 getters et 0 setters :

-getNom () //pour obtenir le nom du client

* 1. Constructeur :

La classe Client possède 1 constructeur :

-Client() //pour créer un nouveau client

* 1. Méthodes :

La classe Client possède 12 méthodes :

-ajouterCompte() //pour créer ou ajouter un nouveau compte pour un client

-interactionAddcompte() //complète la méthode ajouter compte pour l’affichage

-soldeTotal() //calcule le solde total du client dans ces comptes

-afficherSolde() //complète la méthode soldeTotal pour l'affichage

-afficherBilanClient() //affiche tous les comptes du client avec leur solde

-interactionDeleteCompte() // pour supprimer un compte du client

-renflouer() // pour effectuer des renflouement de compte (complète la méthode interactionRetrait() )

-InteractionRetrait() //pour faire un retrait dans son compte

-interactionDepot() //pour ajouter des fond dans son compte

-interactionVirement() //pour effectuer des virement entre 2 compte

-afficherCompte() //affiche les comptes du client

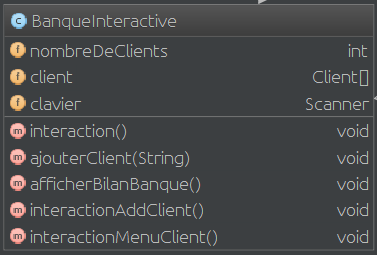
-interaction() //affiche le menu du client

-

**3-La classe BanqueInteractive:**

De même qu’un client peut avoir plusieurs comptes, une banque peut avoir plusieurs clients.

la classe BanqueInteractive, possédant entre autre 3 attribue privés et 5 méthodes :



* 1. Attributs :

La classe Client possède 4 attributs :

-nombreDeClients

-client

-clavier //pour la saisi du clavier

* 1. Méthodes :

La classe Client possède 5 méthodes :

-ajouterClient //pour ajouter un nouveau client

-afficherBilanBanque() //pour afficher l'ensemble des client avec leurs compte

-interactionAddClient() //complète la méthode ajouterClient pour l'affichage

-interactionMenuClient() //affiche le menu client

-interaction() //affiche le menu de la banque ( le menu principale )

**3-La classe Main:**

c'est une classe qui permet de faire fonctionner notre banque interactive ,

elle utilise seulement la méthode main .