



C E N T R E
D'ENSEIGNEMENT
ET DE RECHERCHE
EN INFORMATIQUE



Master Informatique

ILSEN

UE Modélisation du logiciel

Rapport d'activité

12 décembre 2018

Groupe M1-ILSEN ALT

Delfeil CASANOVA
Malvina GONTARD

CERI - LIA
339 chemin des Meinajariès
BP 1228
84911 AVIGNON Cedex 9
France

Tél. +33 (0)4 90 84 35 00
Fax +33 (0)4 90 84 35 01
<http://ceri.univ-avignon.fr>

Sommaire

Titre	1
Sommaire	2
1 Designs patterns trouvés	3
2 Les designs patterns	3
2.1 State	3
2.2 Factory Method	4
2.2.1 Factory Client	4
2.2.2 Factory Reseau	5
2.2.3 Factory Carte Bancaire	6
2.3 Template Method	6
2.4 Builder	7
2.5 Memento	8
2.6 MVC	9
3 Modélisation Globale, Vue d'ensemble	10
4 Scénarios	11
4.1 Création d'un compte client pour personne physique majeur	11
4.2 Création d'un compte client pour personne moral	11
4.3 Création d'un compte client pour personne physique mineur	12

1 Designs patterns trouvés

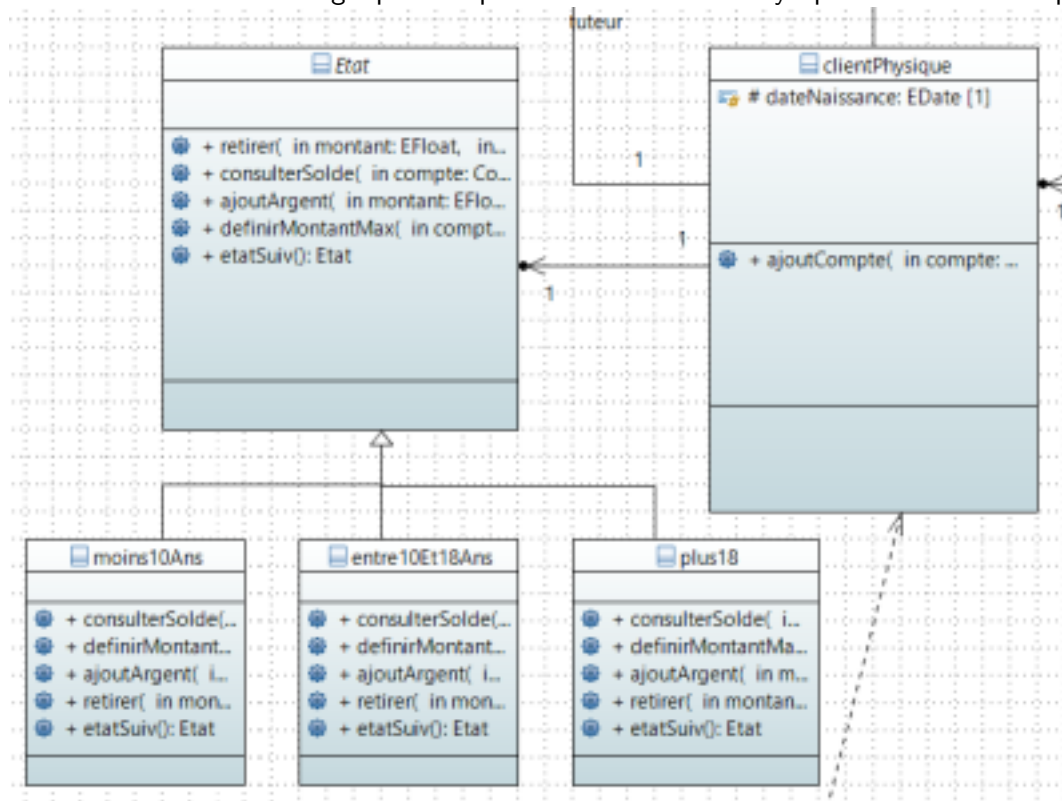
Voici les design patterns que nous utilisons dans cet exercice, nous vous expliquerons par la suite où nous les avons utilisés et comment :

- State
- 3 Factory methods
- Template method
- Builder
- Memento
- MVC

2 Les designs patterns

2.1 State

Nous utilisons ce design pattern pour la classe ClientPhysique située dans la partie Client.



L'énoncé dit qu'un client a un âge et que selon son âge, il a l'autorisation ou pas d'accéder à ses comptes. A un moment donné un client à un âge fixe soit :

- inférieur à 10 ans
- entre 10 et 18 ans
- supérieur à 18 ans

Un client est forcément associé à une et une seule catégorie obligatoirement, donc pour cela nous avons choisi d'utiliser le Design Pattern State, ce qui permet qu'un client est un état. En fonction de cet état le client a des droits différents et donc les fonctions `RetirerArgent()` sont sur définies en fonction de ses droits.

- inférieur à 10 ans : ne peut pas retirer de l'argent
- entre 10 et 18 ans : peut retirer de l'argent, sous montant maximum défini
- supérieur à 18 ans : pas de restriction

2.2 Factory Method

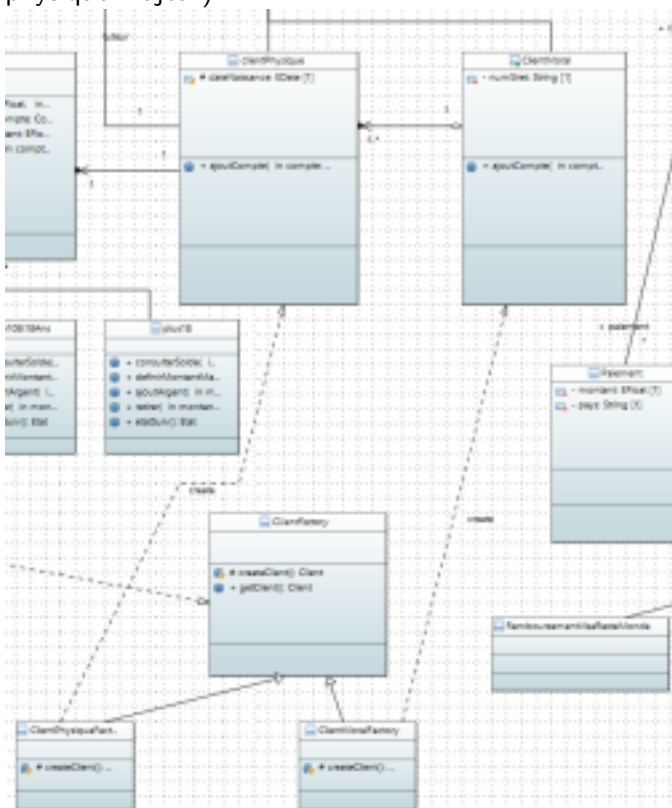
Le factory Method permet d'instancier des objets dont le type est dérivé d'un type abstrait

2.2.1 Factory Client

Notre factory Client nous permet de créer un client de base avec :

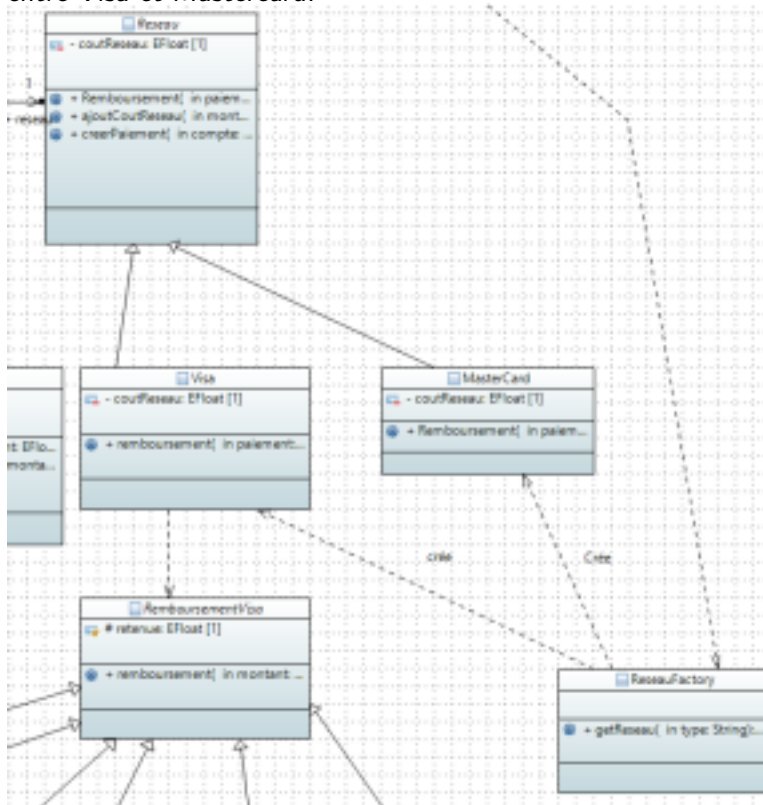
- un nom
- une adresse postal
- une adresse mail
- un numéro de téléphone

Puis nous pouvons choisir si celui est un client Moral ou un client physique et en fonction de ce choix nous sommes redirigé vers une nouvelle factory qui pour le client Moral permet de paramétrer le numéro de SIRET ainsi que le client physique responsable et pour un client physique nous pouvons paramétrer son âge ainsi que si besoin son tuteur (Client physique mineur tutoré par un client physique majeur).



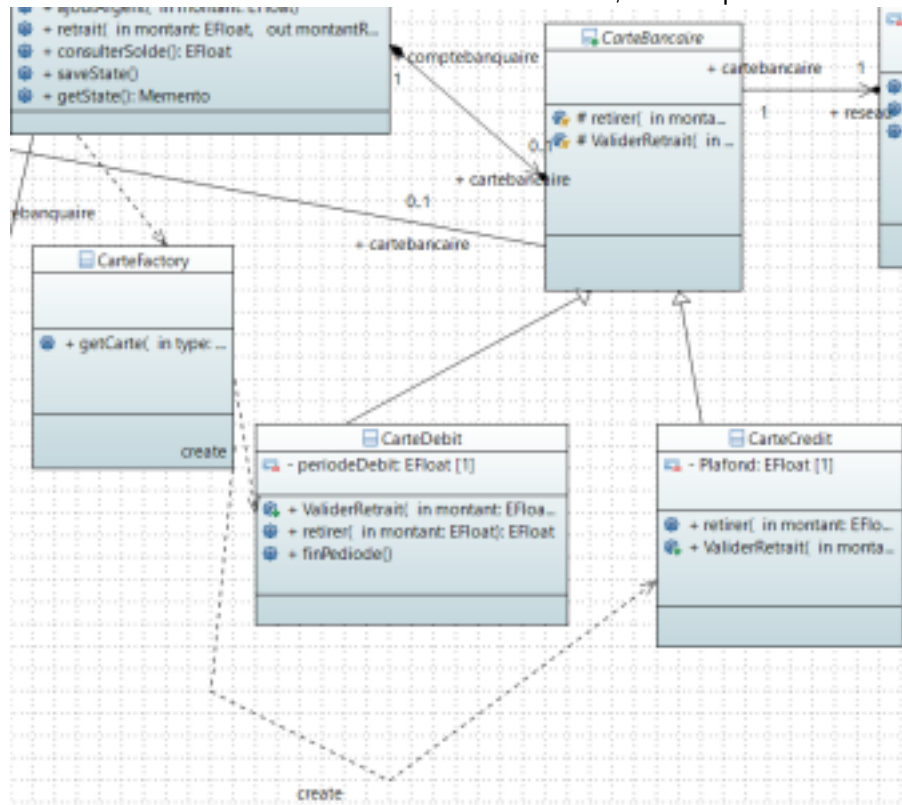
2.2.2 Factory Réseau

La factory réseau nous permet de choisir un réseau pour la création d'une carte on peut choisir entre Visa et Mastercard.



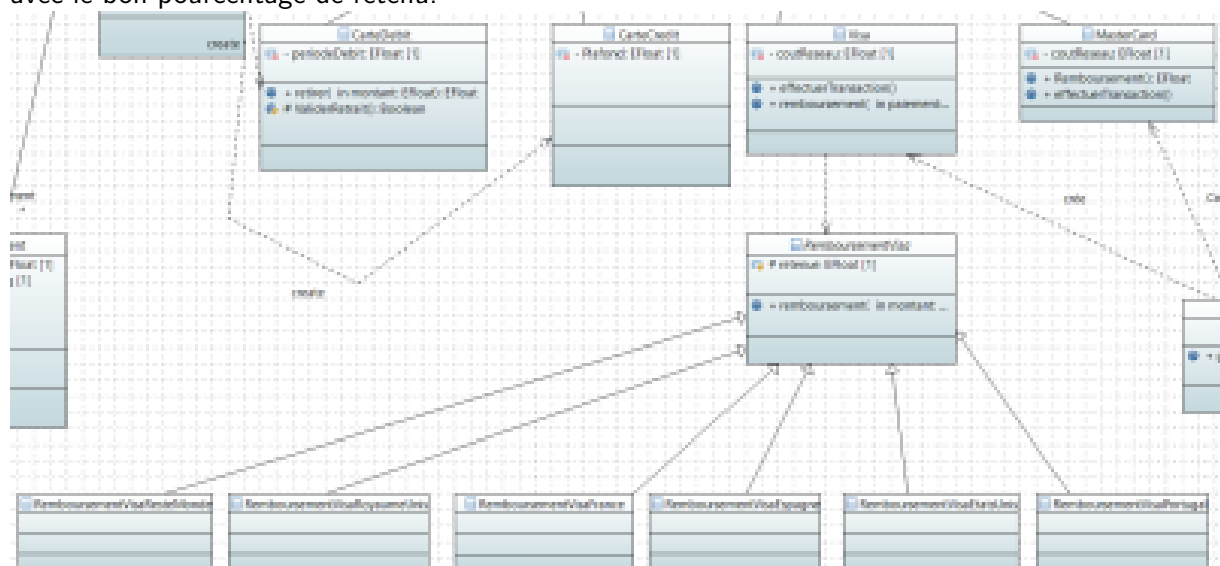
2.2.3 Factory Carte Bancaire

Cette dernière factory nous permet de créer une carte de Paiement qui est soit à crédit soit à débit. Pour créer cette carte il nous faut un client, un compte et réseau déjà créé.



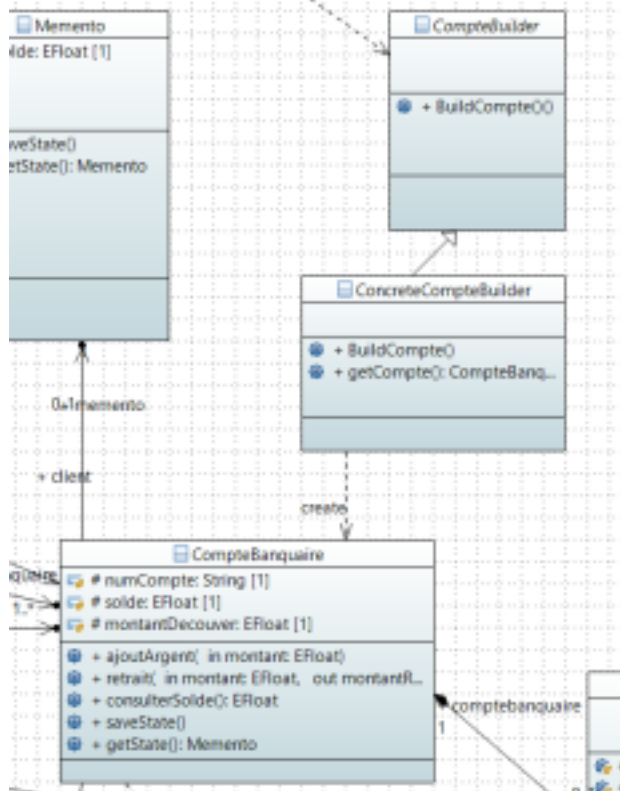
2.3 Template Method

Le Template Method "permet de reporter dans des sous-classes certaines étapes de l'une des opérations d'un objet, ces étapes étant alors décrites dans les sous-classes" (citation du cours). Pour ce problème nous l'utilisons lorsqu'un client veut se faire rembourser un paiement effectué précédemment. En fonction du pays où a été fait le paiement la retenue (en %) ne sera pas la même. C'est à dire que pour chaque pays données nous re-définissons la fonction **RemboursementVisa()** avec le bon pourcentage de retenue.



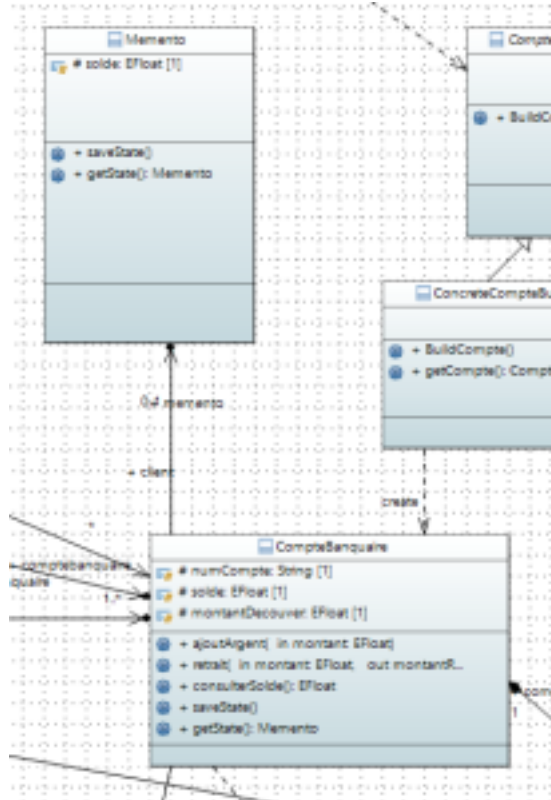
2.4 Builder

Nous utilisons le design pattern Builder dans la partie Compte bancaire. Celui-ci nous permet de pouvoir créer un compte bancaire pour un client donnée. Et donc de pouvoir renseigner chaque paramètre demandé, afin que le compte soit configuré correctement.



2.5 Memento

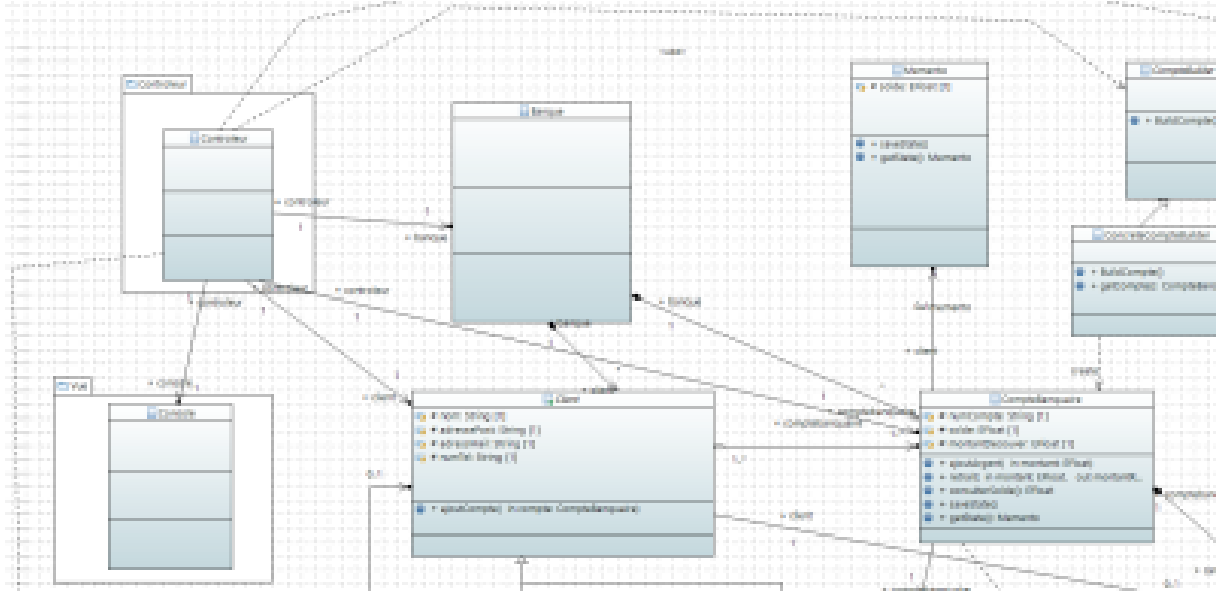
Ce design pattern nous permet de sauvegarder les mouvements sur un compte bancaire. Nous pouvons enregistrer l'ancien solde et de pouvoir le comparer avec le solde courant pour savoir si le dernier mouvement a été un ajout, un retrait... et le montant de ce dernier mouvement sur le compte du client.



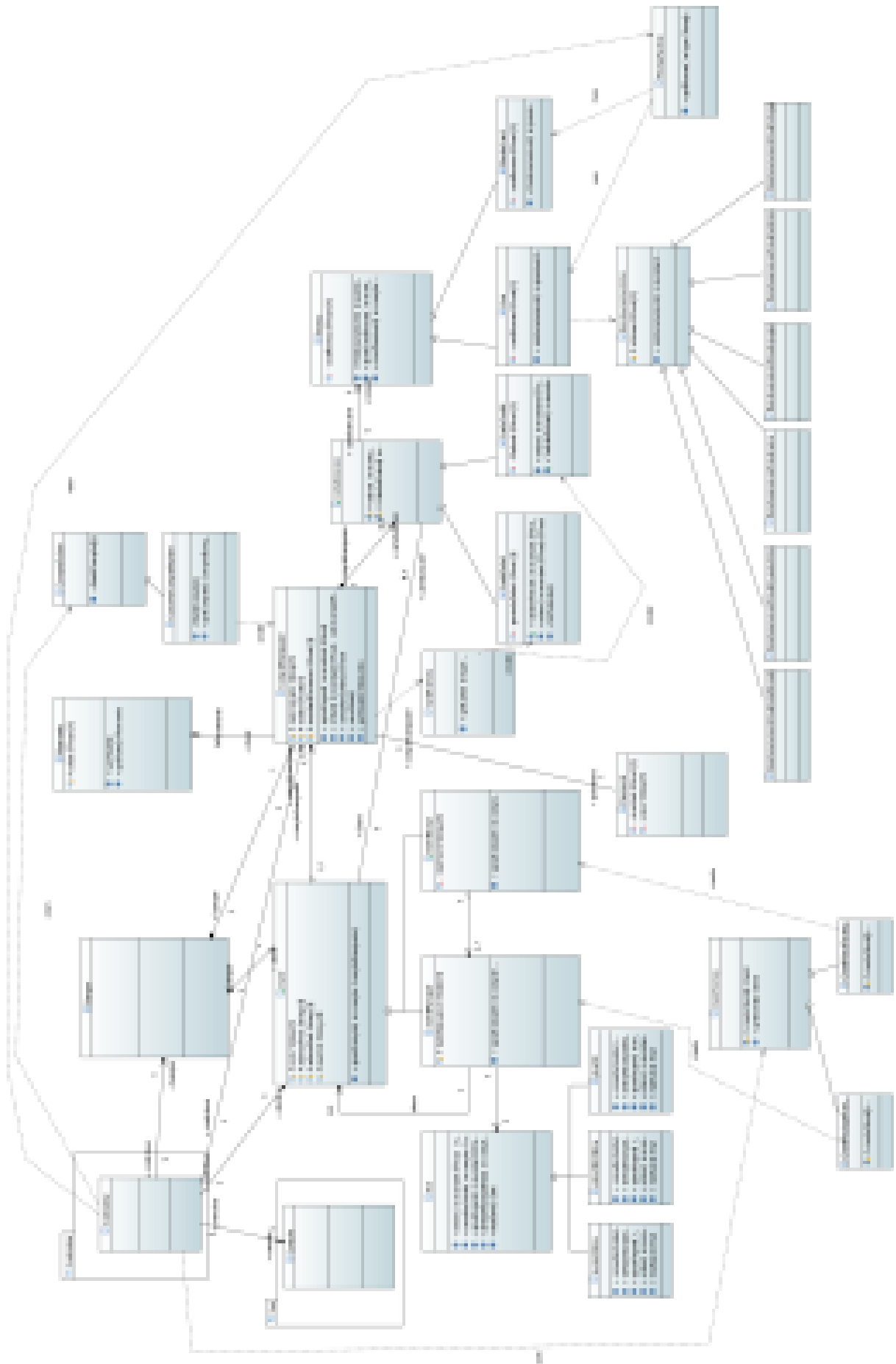
2.6 MVC

Le design Patern MVC signifie "Modèle Vue Contrôleur". Pour modéliser ce design pattern nous avons fait 3 packages :

- un premier, contrôleur, qui implémente le design pattern médiateur. Il a accès au FactoryClient et FactoryRéseau ainsi qu'au Builder de compte bancaire.
- un second, modèle, qui est composé de toute notre modélisation avec nos classes Compte Bancaire, Client, CarteBancaire,...
- un dernier package qui est la vue et qui est une classe intermédiaire avec le terminal



3 Modélisation Globale, Vue d'ensemble



4 Scénarios

Ce qui est écrit dans le terminal : couleur noir

Ce qui est écrit par l'utilisateur : couleur bleu

4.1 Création d'un compte client pour personne physique majeur

Que voulez-vous faire ? : 'créer Client', 'Se connecter avec un client', 'afficher les clients' ou 'Quitter'

créer Client

Quel type de client voulez-vous créer ? : 'moral' ou 'physique'

physique

Veuillez saisir le nom du client

bob

Veuillez saisir l'adresse du client

12 Chemin du Vas 84810 Aubignan

Veuillez saisir le mail du client

bob@bob.fr

Veuillez saisir le numéro de téléphone du client

0605040302

Veuillez saisir l'âge du client

21

Voulez vous ajouter un tuteur ?

non

Le client physique est créé et stocké dans notre base de données.

4.2 Création d'un compte client pour personne moral

Que voulez-vous faire ? : 'créer Client', 'Se connecter avec un client', 'afficher les clients' ou 'Quitter'

créer Client

Quel type de client voulez-vous créer ? : 'moral' ou 'physique'

moral

Veuillez saisir le nom du client

bob entreprise

Veuillez saisir l'adresse du client

13 Chemin du Vas 84810 Aubignan

Veuillez saisir le mail du client

bobentreprise@entreprise.fr

Veuillez saisir le numéro de téléphone du client

0404040404

Veuillez saisir le numéro SIRET

789 789 789 78978

Veuillez saisir le nom du client responsable

bob

Voulez-vous ajouter un autre responsable ? 'oui', 'non'

non

Notre client moral est créé et stocké dans notre base de données. Celui-ci a comme responsable notre client physique majeur créé précédemment.

4.3 Création d'un compte client pour personne physique mineur

Que voulez-vous faire ? : 'créer Client', 'Se connecter avec un client', 'afficher les clients' ou 'Quitter'

[créer Client](#)

Quel type de client voulez-vous créer ? : 'moral' ou 'physique'

[physique](#)

Veillez saisir le nom du client

[bub](#)

Veillez saisir l'adresse du client

[12 Chemin du Vas 84810 Aubignan](#)

Veillez saisir le mail du client

[bub@bub.fr](#)

Veillez saisir le numéro de téléphone du client

[0605040201](#)

Veillez saisir l'âge du client

[15](#)

Veillez saisir le nom du client tuteur

[bob](#)

Le client physique mineur est créé et stocké dans notre base de données. Celui-ci à un tuteur obligatoirement car il est mineur. Ce tuteur est notre client bob qui est majeur et qui a été créé précédemment.

Si nous entrons en nom de tuteur (ou de responsable) un nom non connu par la base de données alors le terminal affiche :

Voulez-vous choisir un autre tuteur, ou créer un client ? 'créer', 'autre tuteur'