



ĐẠI HỌC QUỐC GIA HÀ NỘI  
**VNU**  
Since 1906

Vietnam National University (VNU)

Institut de la Francophonie pour l'Innovation (IF)



---

# RAPPORT TP1

**Option :** *Systèmes Intelligents et Multimédia (SIM)*

***Simple application de gestion bancaire***

***Génie Logiciel***

**Promotion : XXII**

**Auteurs :** Kengnih Hippolyte et OUSSEINI.H Abdoul-Djalil

**Sous la supervision de :** Dr HO Tuong Vinh

*Année académique: 2017/2018*



## Sommaire

### II.La spécification de l'application

### III.La conception

- 1.Diagramme de classe
- 2.Diagramme de Séquence
- 3.Diagramme de Cas d'utilisation

### IV.L'implémentation et test

#### 1.Implémentation

#### 2.Test

### V.Conclusion

## ***SOMMAIRE***

<b>SOMMAIRE</b> .....	3
<b>INTRODUCTION</b> .....	4
<b>I. LA SPECIFICATION DE L'APPLICATION</b> .....	5
<b>II. LA CONCEPTION</b> .....	5

## ***INTRODUCTION***

Dans un monde de plus en plus moderne, disposer d'un compte bancaire est nécessaire pour sécuriser son argent, percevoir un salaire, une allocation ou faire des achats, payer son loyer ou ses factures... En ouvrant un compte bancaire, vous pouvez disposer d'une multitude de solutions pour gérer votre argent au quotidien en toute sécurité.

Ainsi donc, Les établissements bancaires et financiers doivent faire face à l'augmentation de la clientèle qui est de plus en plus croissante, pour y répondre il est important de mettre en œuvre un programme quasi infallible permettant d'administrer les comptes des clients.

Notre travail consistera à élaborer une application de gestion de comptes bancaires avec Java. Le but du projet est d'implémenter l'application avec une démarche orientée objet, il ne suffit pas d'implémenter mais il faut aussi décrire de manière visuelle et graphique les besoins et les solutions fonctionnelles et techniques du projet logiciel, ceci nous amènera donc à modéliser avec UML. Dans l'étude qui suivra, nous présenterons la spécification de notre application, sa conception à travers les différentes classes, son implémentation et le test.

## I. LA SPECIFICATION DE L'APPLICATION

---

Le succès de tout projet dépend étroitement de la capacité de ses acteurs à respecter point par point les clauses définies dans le cahier de charges. Il s'agit là de faire une brève présentation des opérations que peut faire la banque afin de donner une vision globale de notre application, elle permettra éventuellement de :

- Gestion des clients : ici, il s'agira d'enregistrer un client, consulter un client, modifier un client, supprimer un client, faire une recherche sur un client donné ;
- Gestions des comptes : ici nous allons créer les comptes des différents clients, le gestionnaire aura la possibilité de créer plusieurs comptes différents pour un client donné, nous pourra faire une recherche à l'aide de l'identification du client afin de savoir si ce dernier possède un ou plusieurs comptes (compte courant ou épargne). Le numéro de compte et le numéro d'identification du titulaire du compte seront générés automatiquement par le système. Le gestionnaire pourra aussi faire une mise à jour du solde des comptes (le système calculera automatique les intérêts à ajouter à chaque compte et modifiera le solde de tous les comptes par le solde + l'intérêt correspondant).
- Gestions des opérations bancaires : ici, le gestionnaire aura la possibilité d'effectuer un dépôt ou un retrait (lors du retrait, le système vérifiera automatiquement si le montant à retirer n'est pas supérieur au solde en compte ; si dès est le cas, l'opération sera annulé) dans un compte précis. Le gestionnaire pourra aussi visualiser les différentes transactions effectuées, leur date, et les comptes ayant un solde de moins de 100€

## II. LA CONCEPTION

---

Une bonne méthodologie de réalisation de logiciel suppose une bonne maîtrise de l'analyse et de la conception, cette dernière nous offre tous les modèles destinés à assurer le fonctionnement du logiciel. Ces modèles permettent d'explicitier les fonctionnalités. De manière globale, elle offre une vue panoramique sur l'ensemble des éléments et les interactions pris en compte dans la conception. Pour ce faire nous utiliseront UML pour modéliser notre futur système.

UML est un langage pour :

- ❖ Visualiser chaque symbole graphique à une sémantique ;
- ❖ Spécifier de manière précise et complète, sans ambiguïté ;
- ❖ Construire les classes, les relations SQL peuvent être générés automatiquement.

Le langage UML comprend 9 diagrammes principaux à savoir :

- ❖ Le diagramme des cas d'utilisation ;
- ❖ Le diagramme de classe ;
- ❖ Le diagramme d'activité ;
- ❖ Le diagramme d'Etat transition ;
- ❖ Le diagramme de déploiement ;
- ❖ Le diagramme de séquence ;
- ❖ Le diagramme d'objet ;
- ❖ Le diagramme de composant ;
- ❖ Le diagramme de collaboration.

Dans notre cas d'étude, nous nous limiterons à la modélisation de trois (03) diagrammes à savoir :

- Le diagramme des cas d'utilisation : représente la structure des fonctionnalités nécessaires aux utilisateurs du système. Il est normalement utilisé lors des étapes de capture des besoins fonctionnels et techniques ;
- Le diagramme des classes : sûrement l'un des diagrammes les plus importants dans un développement orienté objet. Sur la branche fonctionnelle, ce diagramme est prévu pour développer la structure des entités manipulées par les utilisateurs. En conception, le diagramme de classes représente la structure d'un code orienté objet ;
- Le diagramme de séquence : représente les échanges de messages entre objets, dans le cadre d'un fonctionnement particulier du système.

## 1. Diagramme de Cas d'utilisation

Pour illustrer les fonctionnalités de notre application. Il permet de ressortir les différents acteurs du système, les fonctionnalités et ainsi que les relations entre eux.

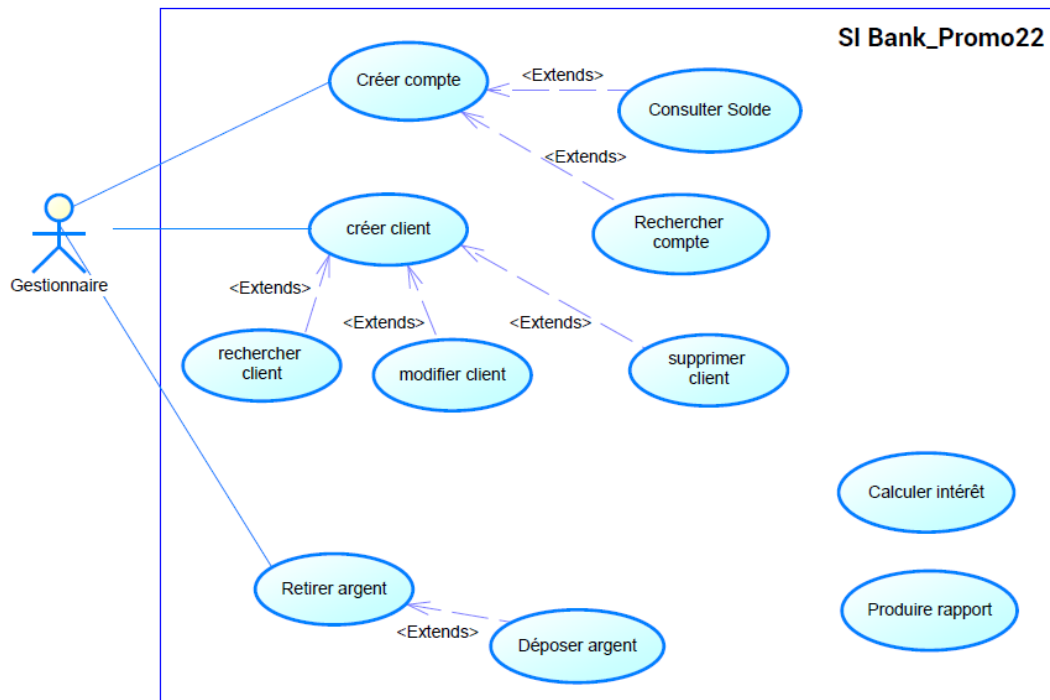


Figure 1: Diagramme de Cas d'utilisation

## 2. Diagramme des classes

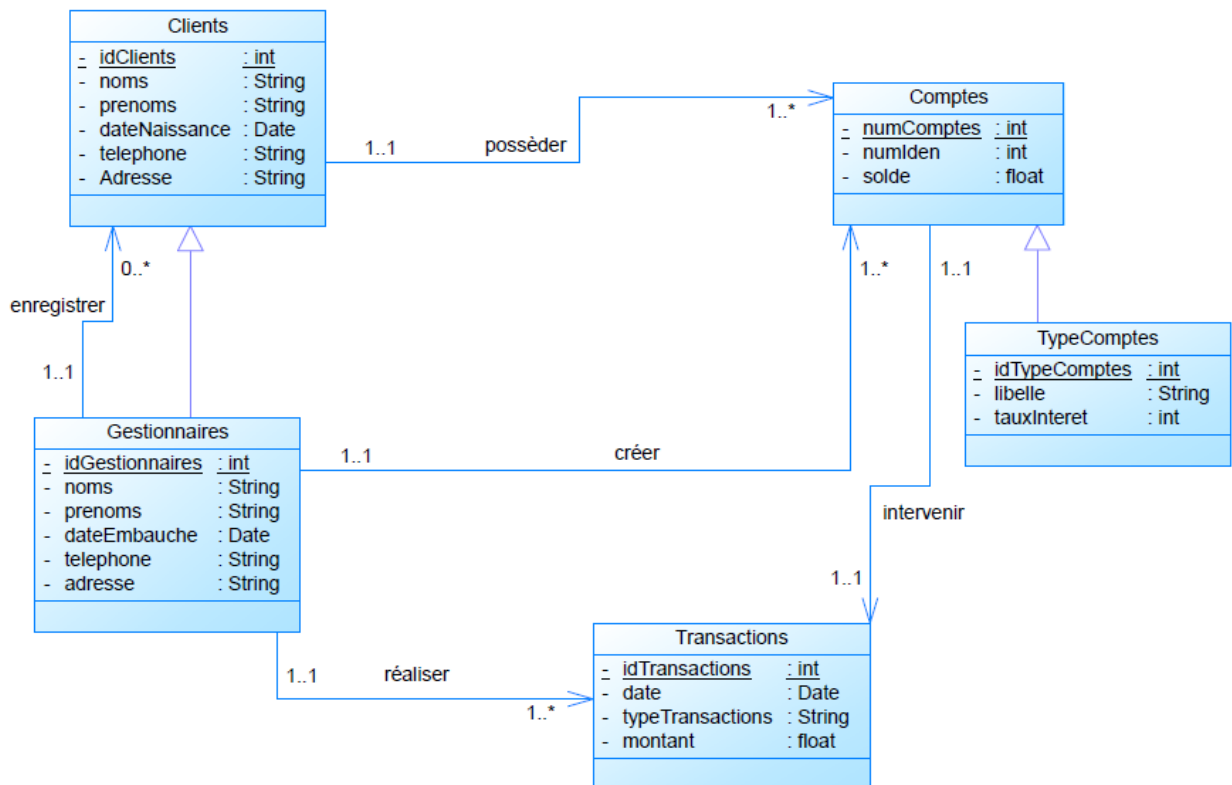


Figure 2: Diagramme des classes

### 3. Diagramme de séquence

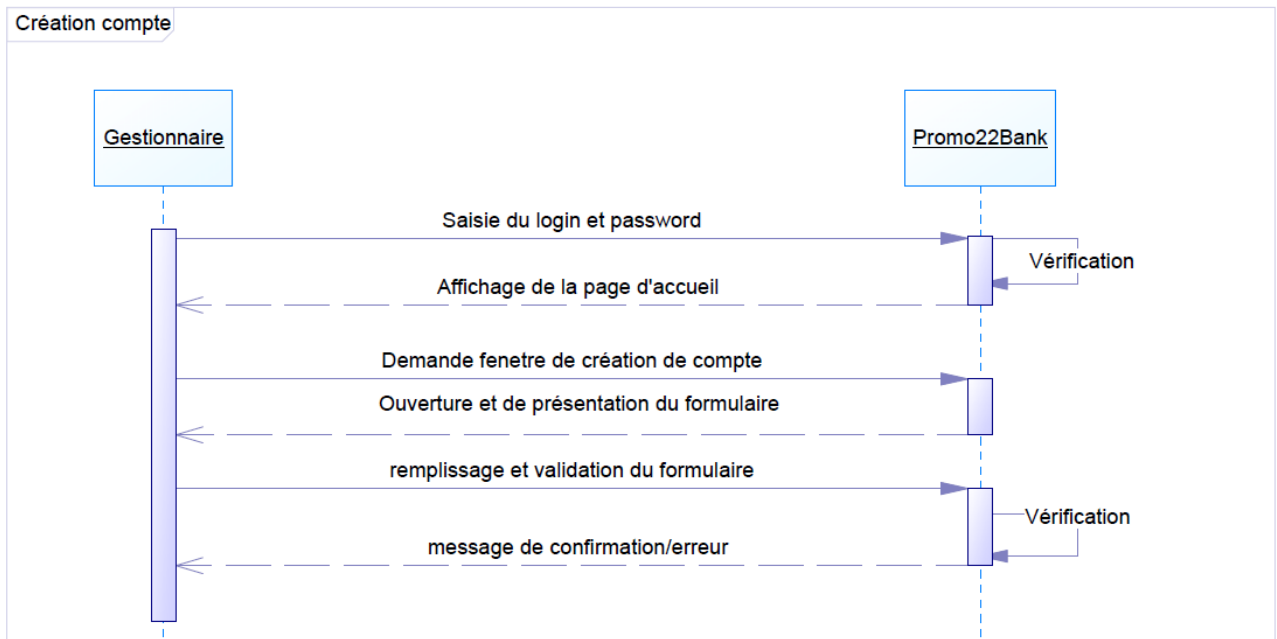


Figure 3: Diagramme de séquence de création d'un compte

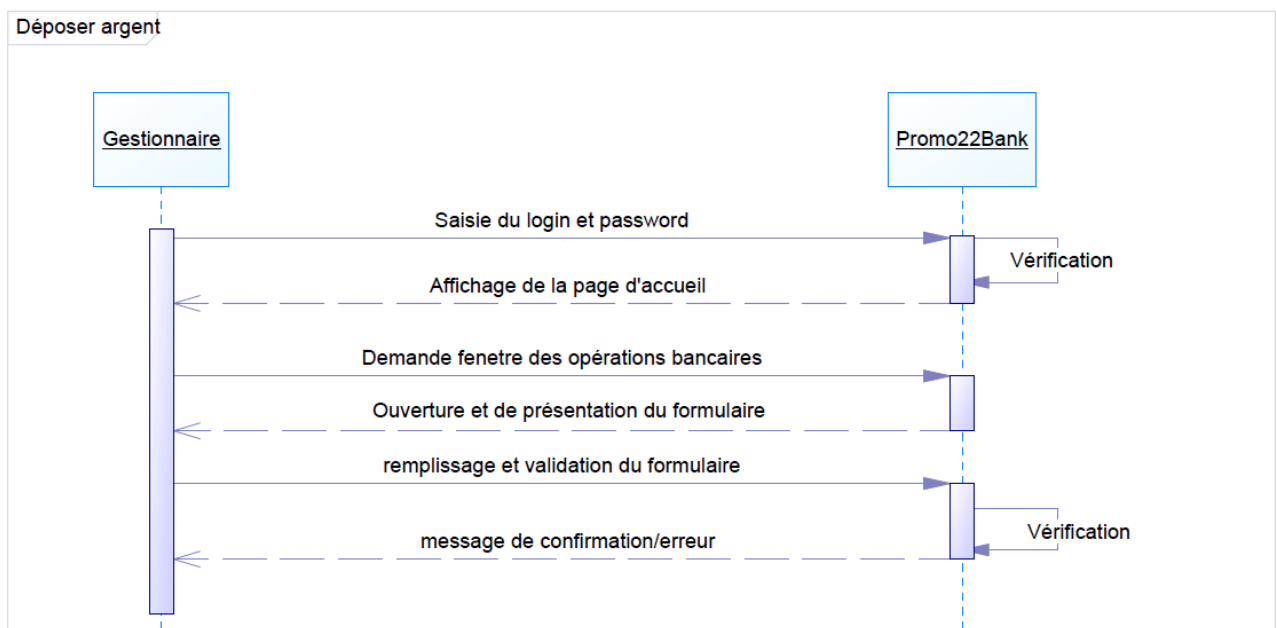


Figure 4: Diagramme de séquence Déposer argent



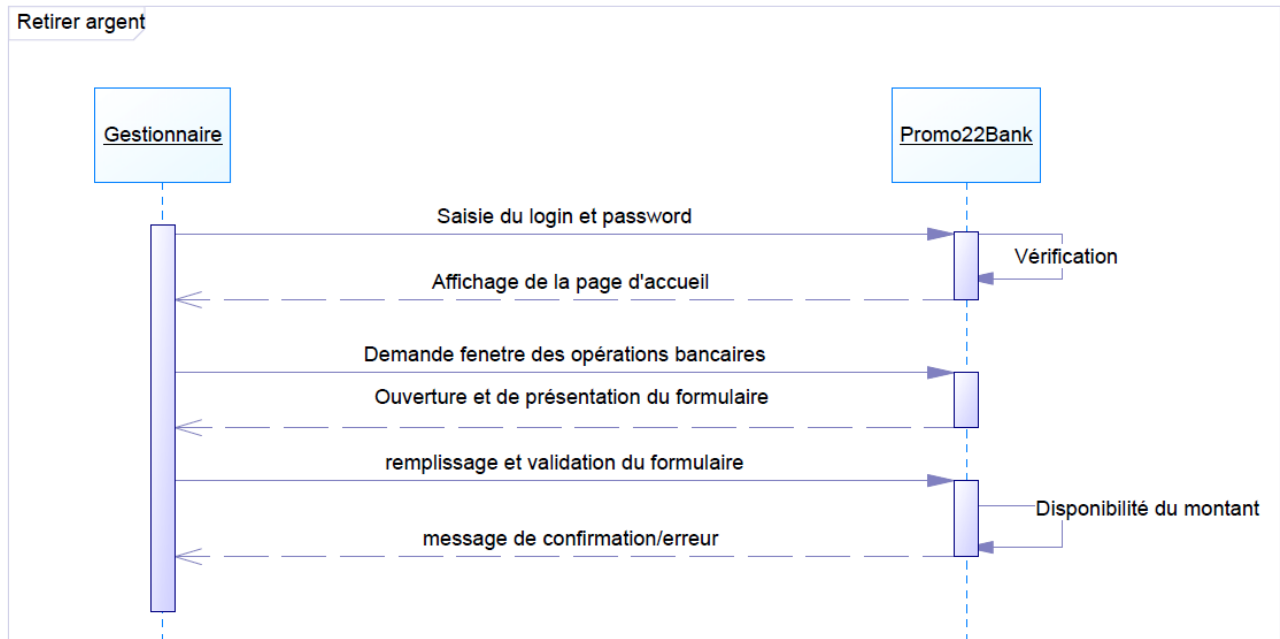


Figure 5: Diagramme de séquence Retirer argent

A partir de ces diagrammes nous avons pu générer le schéma de la base de données suivant :

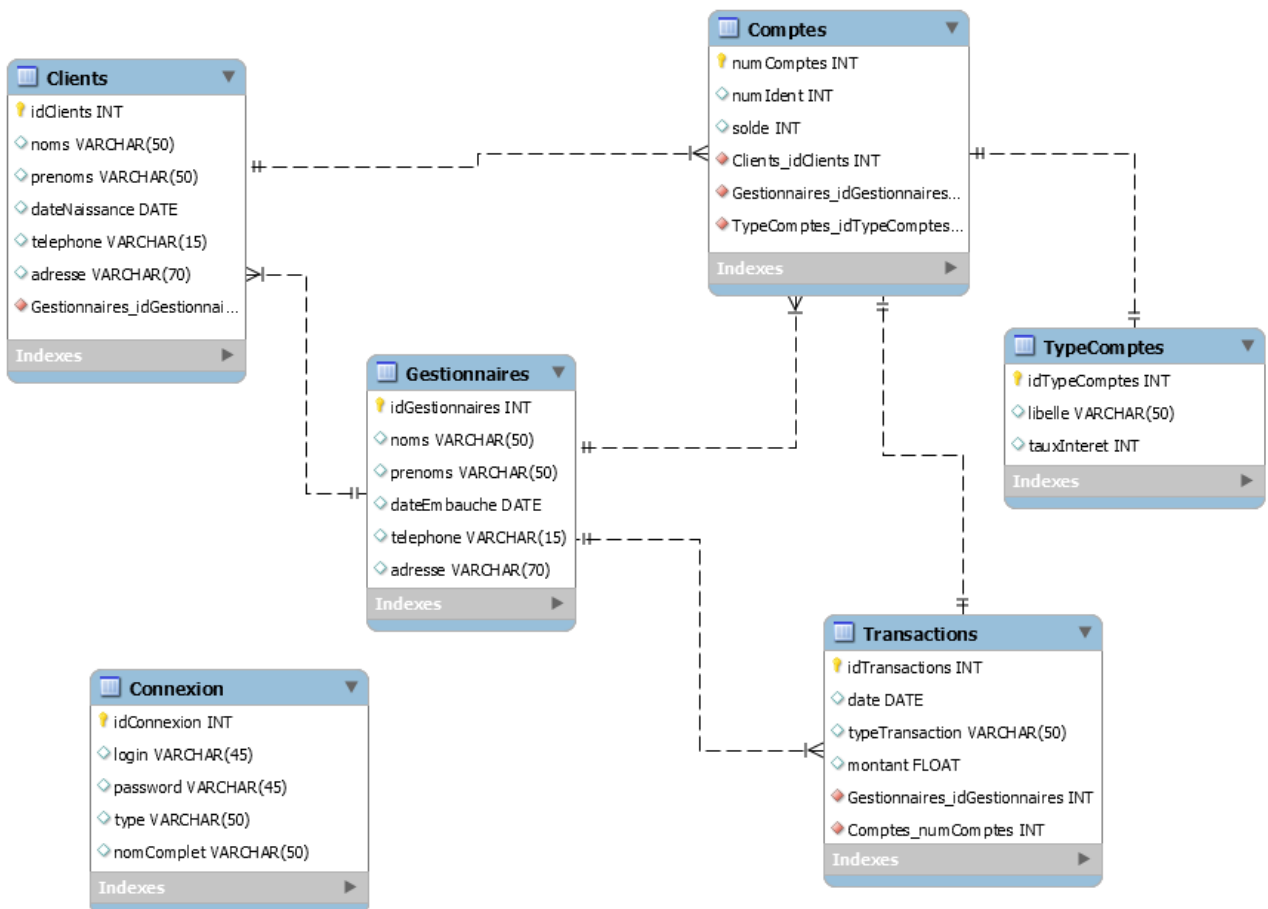


Figure 6: Architecture de la base de données de notre application

### **III. IMPLEMENTATION ET TESTS**

---

- 1. Implémentation**
- 2. Tests**



## ***CONCLUSION***

---



## *ANNEXE*

---