

République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur
et de la recherche scientifique

Université du 08 mai 45, Guelma



Faculté des sciences économiques et commerciales
et sciences de gestion
Département des sciences de gestion

Mémoire présenté pour l'obtention
du diplôme de Master en sciences de gestion
Option: Techniques d'information et de communication
dans l'entreprise

Thème
Conception d'un système de suivi de
compte bancaire unifié
BADR Banque

Présenté par :
Bousnoubra Khadija
Bouguerme Wahiba

Sous la direction de :
Kelaiia Abdessalem

Juin 2012

111004.024
2012/123



Remerciement

Nous Tenons en premier lieu à remercier Dieux pour l'aide qu'il nous a

Accordé pour arriver au terme de notre projet et de notre cursus.

Nous devons une reconnaissance particulière à notre promoteur MrKlaiaia Abdesslem pour la confiance qu'elle nous a accordée en nous

Proposant ce projet, mais aussi pour sa disponibilité, sa gentillesse et Surtout sa générosité.

Nous remerciant aussi tous les enseignants du département de gestion

Sans oublié les personnels de la BADR banque Et surtout Mr Mouassa Sadoune

Les personnels de bureau d'informatique

Pour l'accueil qu'il nous dirigeons tout au long de notre stage, Mais surtout leurs coopérations et leur gentillesse.

Nos remerciements vont aussi à chacun des membres du jury pour avoir

Fait l'insigne honneur d'examiner notre travail, sans oublier nos Respects à tous nos professeurs

Pour n'oublier aucun, nous remercier tous ceux qui ont contribué de

Prés ou de loin à l'achèvement de ce travail.

KHADIJA, WAHIBA

Sommaire

1. Introduction.....	01
2. Problématique et objectifs visé.....	01
3. Présentation de la banque BADR.....	02
3.1. Historique.....	02
a. Période de 1982-1990.....	02
b. Période de 1991-1999.....	03
c. Période de 2000- Avril 2002.....	03
3.2. La structure de la BADR.....	04
a. La direction générale.....	04
b. Les directions centrales.....	04
Chapitre I: Capture des besoins et étude préliminaire	
1. Processus unifié, 2tup.....	06
1.1. Chemin fonctionnel ou branche gauche.....	07
1.2. Chemin technique ou branche droite.....	07
1.3. Chemin de conception ou branche du milieu.....	07
2. Présentation du langage UML.....	08
2.1. Les types de diagrammes officiels d'UML.....	09
3. Etude préliminaire.....	11
3.1. Identification des acteurs.....	11
3.2. Identification des messages.....	12
3.3. Capture des besoins fonctionnels.....	12
3.3.1. Manipulations financières ordinaires.....	12
3.3.2. Crédit.....	13
3.3.3. Placement (épargne).....	13
3.4. Analyse.....	13
3.5. Captures des besoins techniques.....	14
3.6. Conception préliminaire.....	15
3.6.1. Prototype.....	15
Chapitre II : Conception détaillée	
1. Déterminer les cas d'utilisations.....	17
2. Description préliminaire des cas d'utilisations.....	17

6.2. Identification de dictionnaire des données.....	42
Chapitre III : Codage et tests	
1. Codage.....	44
1.1. Présentation du langage utilisé (Borland Delphi7).....	44
1.2. Présentation du système de gestion des bases de données Microsoft Access.....	45
1.3. Vue sur le schéma relationnel de la base de données «CBBADR».....	46
1.4. Vue sur l'implémentation de l'application «SCBBADR».....	46
2. Test.....	47
Conclusion.....	48
Bibliographie	
Annexes	

1. Introduction

Actuellement, le monde continue dans son avancé technologique dans tous les secteurs et cela grâce à l'informatique qui est un ensemble de techniques de traitement automatique de l'information avec toutes ses applications. Cette dernière joue un rôle très important dans le développement du système informatique de l'entreprise et par conséquent toute l'entreprise.

La complexité croissante du système informatique de l'entreprise a conduit les concepteurs à s'intéresser aux méthodes de développement. Pour pouvoir traiter une information, il faut savoir comment la modéliser. Le formalisme de modélisation le plus répondu actuellement et qui est devenu incontournable est le langage de représentation UML (Unified Modeling Language). Ce langage offre une flexibilité marquante qui s'exprime par l'utilisation des diagrammes [Bersini, 2007]. Mais ce dernier à lui seul ne peut nous conduire à monter un système informatique, il nous faut une méthode ou un processus de développement (processus unifié ou Unified Process).

2. Problématique et objectifs visés

Comme dans toute autre banque en Algérie, un client au niveau de la Banque Algérienne de Développement Rural (BADR), est contraint d'avoir plusieurs comptes pour subvenir à ses besoins transactionnels financiers. Par exemple un compte salarié pour son traitement de salaire, un compte crédit s'il veut obtenir un crédit, un compte de placement s'il veut faire de l'épargne, etc....

En effet, attribuer plusieurs comptes à un seul client provoque, et cela par le témoignage du personnel responsable de la BADR Guelma, une lourdeur de gestion et cause un retard terrible sur les opérations financières du client.

L'objectif principal de la présente étude et de remodeler l'aspect structurel de la gestion d'un compte bancaire au niveau de la Banque Algérienne de Développement Rural, agence de Guelma. En d'autre terme, l'unification de toutes les transactions qu'un client peut effectuer (retrait, versement, virement, placement, etc....) sur un seul compte.

Cet objectif nécessite une modélisation. Pour cela, nous avons opté pour l'utilisation de la méthode 2TUP (Two Track Unified Process), l'un des processus unifiés les plus connus de nos jours [Roques, 2006]. Cette méthode ou processus offre la possibilité de modéliser les entités du domaine étudié, tout en permettant la structuration de l'information pour une meilleure organisation de la base de données. Ceci a pour conséquence, de tendre vers une meilleure exploitation en matière d'efficacité et rapidité de l'exploitation de l'information.

b. Période de 1991-1999

La loi 90/10 ayant mis en terme à la spécialisation des banques, la BADR a élargie son champ d'intervention vers les autres secteurs d'activités, et notamment, vers les PME/ PMI, tout en restant un partenaire privilégié du secteur agricole.

Sur le plan technique, cette étape a été celle de l'introduction des technologies informatiques :

- 1991 : mise en place du système « Swift » pour l'exécution des opérations de commerce international ;
- 1992 : mise en place du logiciel « SYBU », avec ces différents modules de traitement des opérations bancaires (gestion des prêts, gestion des opérations de caisse, gestion des placements, consultation à distance des comptes clientèles).
- 1992 : informatisation de l'ensemble des opérations du commerce extérieur. Les ouvertures des crédits documentaires sont, aujourd'hui, traitées en 24 heures maximum ;
- 1992 : introduction du nouveau plan des comptes au niveau des agences ;
- 1993 : achèvement de l'information de l'ensemble des opérations bancaires au niveau du réseau ;
- 1994 : mise en service de la carte de paiement et de retrait BADR ;
- 1996 : introduction du télétraitement (traitement et réalisation d'opérations bancaires à distance et en temps réel) ;
- 1998 : mise en service de la carte de retrait interbancaire.

c. Période de 2000- Avril 2002

L'étape actuelle se caractérise par l'implication intéressante des banques publiques dans la relance des investissements productifs et la mise en adéquation de leurs activités et du niveau de leurs prestations avec les principes de l'économie du marché.

En matière d'intervention dans le financement de l'économie, la BADR a considérablement augmenté le volume des crédits consentis aux PME/PMI du secteur privé (toutes branches confondues) tous en accroissent son aide au monde agricole et para agricole.

Afin de se mettre au diapason des profondes mutations économiques et sociales et répondre aux attentes de la clientèle, la BADR a mis en place un programme d'action quinquennal, axé notamment sur la modernisation de la banque, l'amélioration des prestations, ainsi que l'assainissement comptable et financier.

En choisissant cette méthode ou processus nous espérons baser notre projet sur un processus de développement bien défini qui va de la détermination des besoins fonctionnels attendus du système jusqu'à la conception et le codage final.

Le présent travail a été commencé par la présentation de la banque BADR (Banque Algérienne de Développement Rural), agence de Guelma. Ensuite nous avons appliqué le processus unifié 2TUP sur trois grandes parties. La première comporte la capture des besoins fonctionnels et techniques suivie par une étude préalable. La deuxième partie est consacrée exclusivement à une étude détaillée. La troisième partie est dédiée l'étape codage.

Enfin ce mémoire se termine par une conclusion.

3. Présentation de la banque BADR

3.1. Historique [Web1, 2009]

La Banque de l'Agriculture et du Développement Rural (BADR) est une banque publique créée en 1982 pour mission le développement du secteur agricole et la promotion du monde rural.

Constituée initialement de 140 agences cédées par la Banque Nationale d'Algérie (BNA), son réseau compte aujourd'hui 310 agences et 42 GRE (ex-succursales) (Direction régionale). Près de 7000 cadres et employés activent au niveau des structures centrales, régionales et locales. Son chiffre d'affaires s'élève à plus de 26.800.000.000 DA.

De part de la densité de son réseau et l'importance de ces effectifs, la BADR est considérée par le « BANKERS ALMANACH » (édition 2001) comme la première banque en Algérie. Elle occupe le premier rang au plan national et le 668^{ème} au niveau mondial, sur environ 4100 banques classées.

Etablissement à vocation agricole à sa création, la BADR est devenue au fil du temps, et notamment depuis la promulgation de la loi 10/90 du 14 Avril 1990 relative à la monnaie et au crédit, une banque universelle intervient dans le financement de tous les secteurs d'activités.

Trois étapes caractérisent l'évolution de la BADR :

a. Période de 1982-1990

Au cours de ces huit années, la BADR a eu pour objectif, d'asseoir sa présence dans le monde rural en ouvrant de nombreuses agences dans les zones à vocation agricole. Elle a acquis une notoriété et une expérience certaine dans le financement de l'agriculture, de l'agro-alimentaire et de l'industrie mécanique agricole. Cette spécialisation s'inscrivait, alors, dans un contexte d'économie planifiée où chaque banque publique avait son champ d'intervention.

3.2. La structure de la BADR

La BADR est structurée en 21 directions centrales, le réseau qui se compose de 42 GRE (ex-succursale), 310 agences réparties sur l'ensemble du territoire national.

a. La direction générale

Sous l'autorité du PDG elle est chargée du suivi de toutes les activités de la BADR, elle est assistée de six départements généraux :

- Inspection générale ;
- Direction de l'audit interne « DAI » ;
- Département de la communication « DC » ;
- Secrétariat de la direction ;
- Conseil commissions et comités ;
- Conseillers.

b. Les directions centrales

Elles sont au total 21 directions à savoir :

- Direction générale adjointe « ressources et emplois » ;
- Direction générale adjointe « informatique, comptabilité et trésorerie » ;
- Direction générale adjointe « administration et moyens » ;
- Division internationale ;
- Direction du financement des grandes entreprises (D.F.G.E) ;
- Direction « Informatique central » (D.I.C) ;
- Direction du personnel (D.P) ;
- Direction des opérations techniques avec l'étranger (D.O.T.E) ;
- Direction du financement des petites et moyennes entreprises (D.F.P.M.E) ;
- Direction « informatique réseau d'exploitation » (D.I.R.E) ;
- Direction de revalorisation des ressources humaines (D.R.R.H) ;
- Direction des relations internationales (D.R.I) ;
- Direction du financement des activités agricoles (D.F.A.A) ;
- Direction de la télétransmission et de la maintenance informatique (D.T.M.I) ;
- Direction des moyens généraux (D.M.G) ;
- Direction du contrôle et des statistiques (D.C.S) ;
- Direction des études de marches et produits (D.E.M.P) ;
- Direction de la comptabilité générale (D.C.G) ;
- Direction de la réglementation, études juridiques et du contentieux (D.R.E.J.C) ;
- Direction du suivi et du recouvrement (D.S.R) ;

- Direction de la trésorerie (D.T) ;
- Direction de la prévision et du contrôle de gestion (D.P.C.G).

Chapitre I: Capture des besoins et étude préliminaire

Nous allons commencer par présenter en détail le processus de développement que nous avons choisi pour réaliser les objectifs visés ainsi que le langage de représentation graphique sur lequel il s'articule (UML).

1. Processus unifié, 2TUP

Un processus unifié est un processus de développement qui définit une séquence d'étapes, en partie ordonnée, qui concoure à l'obtention d'un système informatique ou à l'évolution d'un système existant, pour produire des logiciels de qualité, qui répondent aux besoins des utilisateurs dans des temps et des coûts prévisibles. Ce processus comme son nom l'indique, a été le fruit de travail de plusieurs personnes voulant unifier les différentes méthodes objets existantes au moment de l'apparition de ce processus (vers la fin des années 90) comme Booch, OMT (Object Modeling Technique) et OOSE (Object Oriented Software Engineering). [Kazi Aouel et Rostane, 2007a]

Un processus unifié est caractérisé par :

- Il est itératif;
- Il est centré sur l'architecture ;
- Il est conduit par les cas d'utilisation et piloté par les risques. [Frédéric, 2001]

Comme il a été mentionné auparavant un des processus unifiés les plus connus est le processus 2TUP (Two Track Unified Process prononcez "toutiyoupi"). Ce dernier est fortement axé sur l'itératif et le modèle UML.

Le processus 2TUP propose un cycle de développement en Y (Figure 01), qui dissocie les aspects techniques des aspects fonctionnels imposés (en contraintes) au système d'information de l'entreprise. Il commence par une étude préliminaire ou de faisabilité qui consiste essentiellement à identifier les acteurs qui vont interagir avec le système à construire, les messages qu'échangent les acteurs et le système, à produire un cahier des charges (le système est une boîte noire, les acteurs l'entourent et sont reliés à lui, sur l'axe qui lie un acteur au système on met les messages que les deux s'échangent avec le sens).

Le processus s'articule ensuite autour de 3 phases essentielles:

1.1. Chemin fonctionnel ou branche gauche

- Capture des besoins fonctionnels: focalisation sur le métier des utilisateurs.
- L'analyse : obtenir une idée de ce que va réaliser le système en terme de métier.

1.2. Chemin technique ou branche droite

- La capture des besoins techniques: recenser toutes les contraintes sur les choix de dimensionnant et la conception du système.
- La conception générique : définir les composants nécessaires à la construction de l'architecture technique. Construire le squelette du système "un prototype".

1.3. Chemin de conception ou branche du milieu

- La conception préliminaire : intégrer le modèle d'analyse fonctionnelle dans l'architecture technique de manière à tracer la cartographie des composants du système à développer.
- La conception détaillée, qui étudie ensuite comment réaliser chaque composant.
- L'étape de codage : produire les composants et tester au fur et à mesure les unités de code réalisées.
- L'étape de recette : valider les fonctionnalités du système développé. [Web2, 2005]

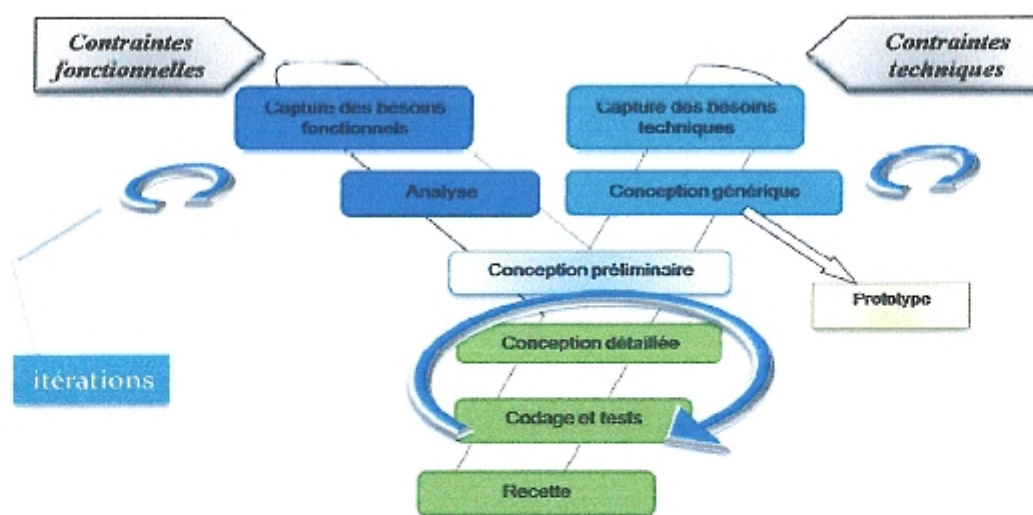


Figure 01 : Le processus de développement en Y. [Kazi Aouel et Rostane, 2007b]

2. Présentation du langage UML

Un des deux axes sur lesquels s'articule le processus 2TUP et le langage UML (Unified Modeling Language), qui se présente comme un langage de modélisation graphique et textuel destiné à comprendre et à définir des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, représenter des solutions et communiquer des points de vue. Il véhicule en particulier les concepts des approches par objets (classe, instance, classification, etc.), et intègre d'autres aspects tels que les associations, les fonctionnalités, les événements, les états et les séquences, etc

UML utilise de nombreux types de modèles, construits sur un certain nombre de concepts communs. Les modèles (ensemble de concepts) qu'UML propose sont très nombreux et couvrent l'ensemble des problèmes que l'on se pose lorsque l'on construit un logiciel, depuis la définition des besoins, jusqu'aux tests et recette.

Les diagrammes proposés sont au nombre de 9 (élargis en 13 diagrammes). Ils permettent de présenter les différentes facettes du logiciel lors des étapes du cycle de vie (analyse, conception, implémentation, architecture).

UML utilise :

- Diagramme de classes;
- Diagramme d'objets;
- Diagramme des cas d'utilisation;
- Diagramme de séquence;
- Diagramme de collaboration;
- Diagramme d'états;
- Diagramme d'activité;
- Diagramme d'implémentation;
- Diagramme de déploiement. [Rabeh, 2005a]

Les quatre autres ajoutés sont :

- Diagramme de paquetage;
- Diagramme de structures composites;
- Diagramme de temps;
- Diagramme global d'interaction.

2.1. Les types de diagrammes officiels d'UML

Diagramme	Description	Version
Diagramme de classes	Montre une collection d'éléments du modèle statique tels que des classes et des types, leur contenu, et les relations qui les lient.	Dans UML 1
Diagramme d'objets	Montre un ensemble d'objets et leurs liens à un instant donné. Généralement il représente un cas particulier d'un diagramme de classes ou d'un diagramme de communication.	Non officiel dans UML 1
Diagramme de Cas d'utilisation	Montre des cas d'utilisation, des acteurs ainsi que leurs relations.	Dans UML 1
Diagramme de séquence	Il montre des classificateurs en classant les messages échangés entre eux dans le temps.	Dans UML 1
Diagramme de Communication (diagramme de Collaboration)	Montre des instances de classes, leurs relations ainsi que l'échange de messages entre elles. Connus sous le nom des diagrammes de collaboration, ces diagrammes se concentrent sur l'organisation structurelle des objets expéditeurs et destinataires de messages.	Dans UML 1
Diagramme de machines à états (diagramme d'états)	Décrit les états d'un objet, aussi bien que les transitions entre les états. Appelé diagramme d'états ou encore diagramme d'états transitions.	Dans UML 1
Diagramme d'activités	Sert principalement à modéliser des processus métiers ou à décrire des opérations complexes, en incluant des flots de données.	Dans UML 1
Diagramme de déploiement	Montre l'architecture d'exécution des systèmes. Ceci inclut des nœuds et des environnements d'exécution de matériel ou de logiciel.	Dans UML 1
Diagramme de composants	Montre les composants d'une application, les relations entre composants ainsi que leurs interfaces.	Dans UML 1

Diagramme de paquetage	Montre comment sont organisés les éléments du modèle en paquetages ainsi que les dépendances entre ces derniers.	Non officiel dans UML 1
Diagramme de structures composites	Décrit la structure interne d'un classificateur (une classe, un composant, ou un cas d'utilisation), y compris les points d'interaction du classificateur à d'autres parties du système.	Nouveau à UML 2
Diagramme de temps	Montre le changement de l'état ou de la condition d'une instance d'un classificateur ou d'un rôle dans le temps.	Nouveau à UML 2
Diagramme global d'interaction	Il s'agit d'une variante du diagramme d'activités montrant le flot de contrôle dans un système ou dans un processus métier. Chaque état activité dans le diagramme peut représenter un autre diagramme d'interaction.	Nouveau à UML 2

Tableau 01 : Les types de diagrammes officiels d'UML. [Bennama, 2008a]

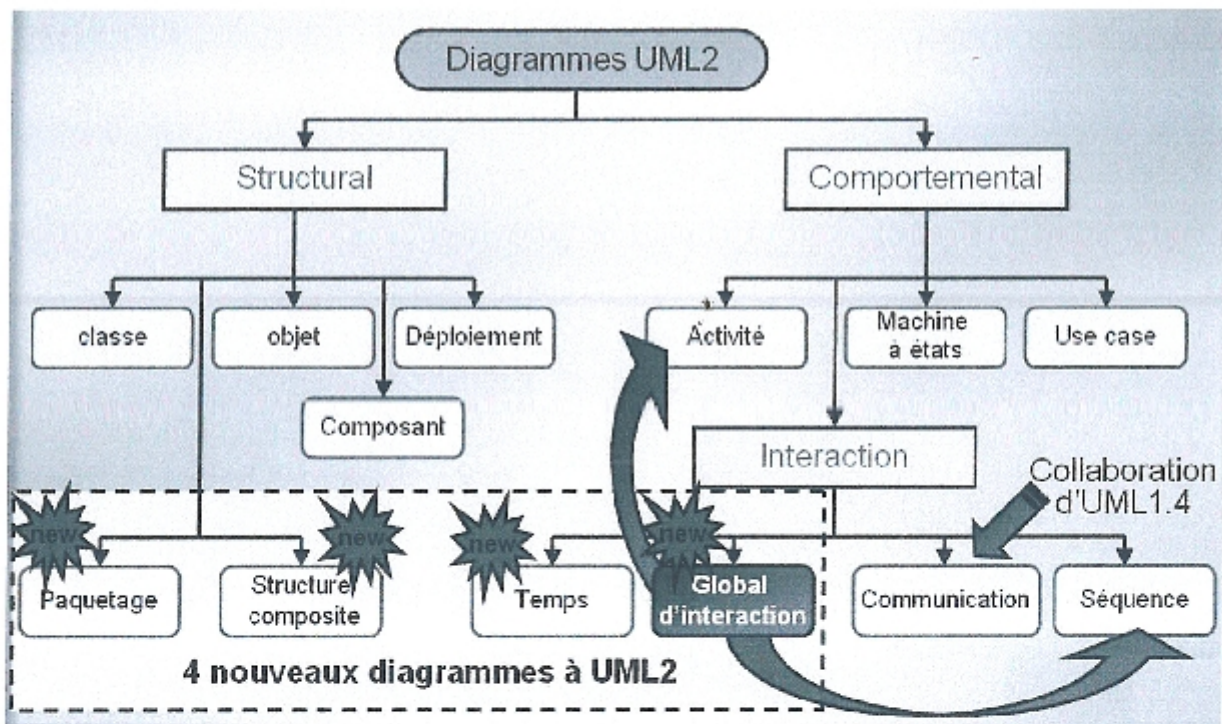


Figure 02 : Diagrammes UML. [Bennama, 2008b]

Ces diagrammes permettent de représenter:

- La structure statique du système, sur lequel, on a appliqué un concept de classification, pour le diagramme de classes.
- La structure statique du système en termes d'objets (au sens où on l'entend intuitivement) pour le diagramme d'objets.

- L'interaction d'un acteur extérieur au système avec le système pour le diagramme des cas d'utilisation.

- Une séquence d'enchaînement de messages permettant de décrire une interaction entre objets pour le diagramme de séquence.

- La collaboration entre les objets d'un système pour réaliser une certaine "action" pour le diagramme de collaboration.

- Les différents états d'un objet d'une classe pour le diagramme d'états.

- Les différentes activités (fonctions) du système à travers le diagramme d'activité.

- Les liens de dépendance entre les composants logiciels (exécution, compilation) pour le diagramme d'implémentation.

- La représentation de l'architecture matérielle et logicielle du système pour le diagramme de déploiement. [Rabeh, 2005b]

3. Etude préliminaire

L'étude préliminaire est la toute première étape du processus 2TUP. Elle consiste à effectuer un premier repérage des besoins fonctionnels et opérationnels, en utilisant principalement le texte, ou diagrammes très simples. Elle prépare les activités plus formelles de capture des besoins fonctionnels et de capture des besoins techniques.

3.1. Identification des acteurs

Un acteur est l'abstraction d'un rôle joué par des entités externes qui interagissent directement avec le système en cours d'étude tels que les utilisateurs du système, les dispositifs matériels ou tout autre système. [Kazi Aouel et Rostane, 2007c]

Au début de notre étude nous avons identifié les acteurs suivants susceptibles d'interagir avec le système futur :

- **Le Front-office :** Le Front-office, comme son nom l'indique, est littéralement l'interface de la banque avec le client. Le Front-office centralise et traite tous les besoins de la banque et de ses clients en termes de couverture et financement. [Web3, 2009]

- **Le Back-office :** Le Back-office n'est plus une simple fonction administrative appliquée aux opérations traditionnelles de la banque. Cette fonction couvre la gestion et la comptabilisation des opérations ainsi que le contrôle de gestion. [Web4, 2009]

- **Virement** : Le virement bancaire est une opération d'envoi (transfert) ou de réception (rapatriement) d'argent entre deux comptes bancaires. Ces comptes peuvent être dans la même banque (virement interne) ou dans deux banques différentes (virement externe).

3.3.2. Crédit

Faire un crédit c'est donner librement la disposition effective et immédiate d'un bien réel ou d'un pouvoir d'achat contre la promesse que le même bien, ou un bien équivalent, sera restitué dans un certain délai, le plus souvent avec rémunération du service rendu et du danger couru danger de perte partielle ou totale que comporte la nature même de ce service.

3.3.3. Placement (épargne)

C'est une forme d'épargne qui ne génère pas de frais mais elle crée des bénéfices (intérêts).

3.4. Analyse

A partir des informations obtenues lors des deux précédentes étapes, nous allons modéliser précisément la spécification fonctionnelle de manière à obtenir une idée de ce que va réaliser le système en terme de métier. Ceci va nous permettre dans un premier temps, de définir le rôle de chaque acteur dans le système :

Utilisateurs finaux	Description des besoins fonctionnels
Guichetier	L'application doit permettre au guichetier de : <ul style="list-style-type: none">• L'authentification• Valider un compte• Débitier un compte (retrait)• Créditer un compte (versement et placement)• Transfert entre deux comptes (virement)
Back-office	L'application doit permettre au Back-office de : <ul style="list-style-type: none">• L'authentification• Vérification des informations saisies du nouveau compte

Front-office	L'application doit permettre au Front-office de : <ul style="list-style-type: none">• L'authentification• Créer un nouveau compte• Supprimer un compte (fermeture d'un compte)
---------------------	--

Tableau02 : Description des besoins fonctionnels.

3.5. Captures des besoins techniques

La capture des besoins techniques recense toutes les contraintes sur les choix de dimensionnement et la conception du système. Les outils et le matériel sélectionné ainsi que la prise en compte des contraintes d'intégration avec l'existant.

Le principal défaut, à notre avis, du système « SYBU », d'ailleurs qui a nous conduit à mener ce projet, et comme il a été mentionné auparavant, c'est qu'un client est contraint d'avoir plusieurs comptes pour subvenir à ses besoins transactionnels financiers. Dire qu'il faut remodeler l'aspect structurel de la gestion d'un compte bancaire nous amène à remettre en cause tout le système « SYBU » ce qui donne du froid dans le dos des décideurs de la BADR banque. Cela nous a incités à essayer de monter un prototype basé sur notre idée et le livrer à la BADR banque pour le mettre sous test et décider par la suite.

Nous avons alors adopté le choix technique basé sur :

- Programmation sous le langage de programmation Delphi.
- Conception de la base de données sous Microsoft Access.

3.6. Conception préliminaire

3.6.1. Prototype

Le prototype préconisé pour le système informatique futur est comme suit:

a. Esquisse du menu général de l'application

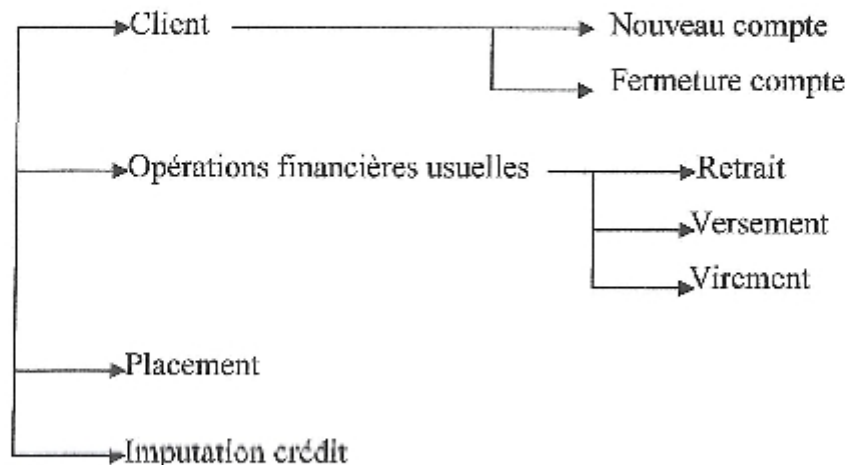


Figure 03 : Esquisse du menu général de l'application.

b. Architecture globale du système futur

L'architecture du système est similaire à celle en cours d'exploitation au niveau de la BADR banque:

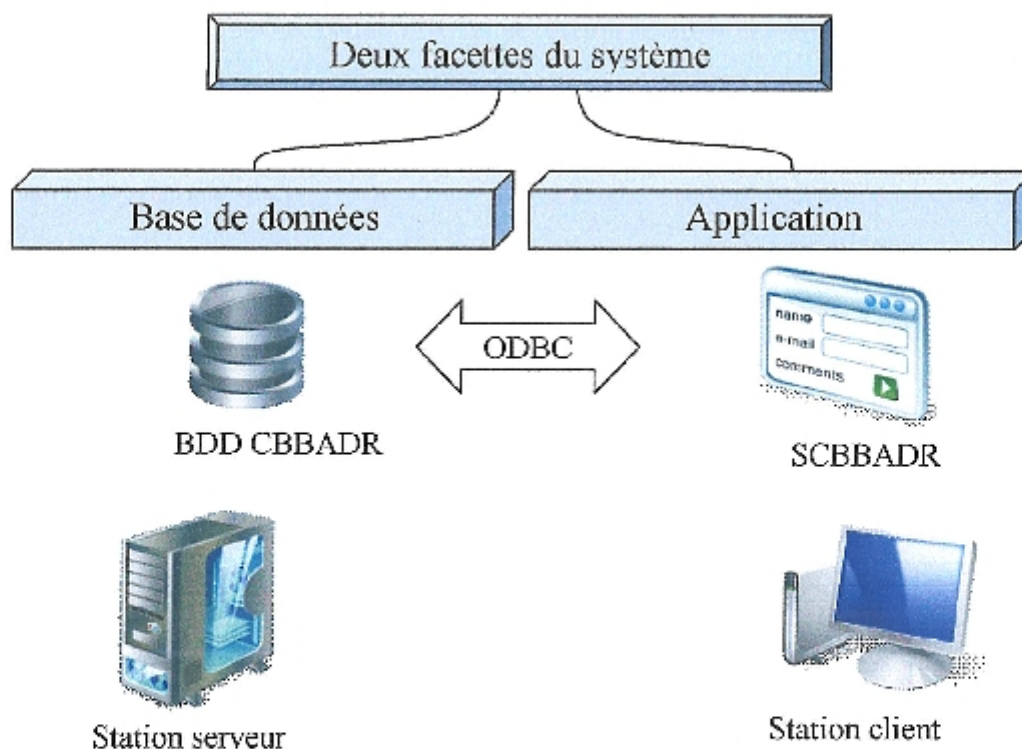


Figure 04 : Architecture du système futur.

b.1. Station serveur

La station serveur est destinée à contenir la base de données "CBBADR". Nous recommandons la configuration logicielle suivante :

Système d'exploitation : Microsoft Windows 2003 server ou plus.

Système de gestion de base de données : Microsoft SQL server 2003 ou plus.

La configuration matérielle recommandée pour le serveur de la base de données CBBADR est un serveur HP (processeur Proliant).

b.2. Station client

Les stations clientes sont destinées à supporter l'application "SCBBADR". Nous recommandons la configuration logicielle suivante :

Système d'exploitation : Microsoft Windows XP ou plus.

La configuration matérielle recommandée est un microordinateur compatible (PIV minimum avec 256 MO de RAM).

Néanmoins, il faut noter que le système matériel en place (au niveau de la BADR Guelma) peut largement gérer la future base de données.

b.3. Connexion à la base de données "CBBADR"

Pour la connexion à la base de données "CBBADR", nous avons utilisé la source de données ODBC (Open Data Base Connectivity) qui permet une grande performance à l'accès aux données.

b.4. Sécurité

La politique de sécurité adoptée dans la solution que nous proposons est basée sur la technologie active directory de Microsoft. L'authentification pour l'accès à la base de données se fait en se basant sur l'authentification Windows contrôlée par le contrôleur de domaine. Les utilisateurs sont authentifiés lors de l'ouverture d'une session Windows (Nom utilisateur, mot de passe).

Les utilisateurs futurs du système possèdent des profils qui varient selon leurs tâches dans le suivi des dossiers clients: Guichetier, Front-office, Back-office.

Chapitre II : Conception détaillée

Au cours de ce chapitre nous allons concrétiser les besoins fonctionnels décelés au cours du premier chapitre.

1. Déterminer les cas d'utilisations

Un cas d'utilisation est une unité cohérente représentant une fonctionnalité visible de l'extérieur. Il réalise un service de bout en bout, avec un déclenchement, un déroulement et une fin, pour l'acteur qui l'initie. Un cas d'utilisation modélise donc un service rendu par le système, sans imposer le mode de réalisation de ce service. [Abdat et mahdaoui, 2007a]

2. Description préliminaire des cas d'utilisations

Voici une description préliminaire des cas d'utilisations :

2.1. Nouveau compte

Intention : Ouvrir un nouveau compte.

Action : Créer un nouveau compte, créer un nouveau client, suivre les dossiers (vérifier, accepter ou refuser), modifier les informations des clients.

2.2. Retrait

Intention : Retirer de l'argent

Action : Valider un compte, vérifier la somme, débiter le compte.

2.3. Versement

Intention : Dépôt d'une somme d'argent dans le but d'imputer un crédit.

Action : Valider le compte, créditer le compte, imputation le crédit.

2.4. Virement

Intention : Transférer un montant d'argent d'un compte à un autre.

Action : Valider le compte, débiter le compte, créditer le compte récepteur si l'opération est interne, transmission des informations inhérentes au compte étranger.

2.5. Placement

Intention : Placement d'argent dans un compte.

Action : Valider le compte, créditer le compte.

2.6. Imputation crédit

Intention : Rembourser d'un crédit.

Action : Valider le compte, débiter le compte.

2.7. Fermeture d'un compte

Intention : Fermer un compte.

Action : Solder le compte, supprimer un compte.

3. Description détaillée des cas d'utilisations

3.1. Présentation globale

L'identification des cas d'utilisation une première fois, nous donne un aperçu des fonctionnalités futures que doit implémenter le système.

Pour constituer les cas d'utilisation, nous allons considérer l'intention fonctionnelle de l'acteur par rapport au système dans le cadre de l'émission ou de la réception de chaque message. [Abdat et mahdaoui, 2007b]

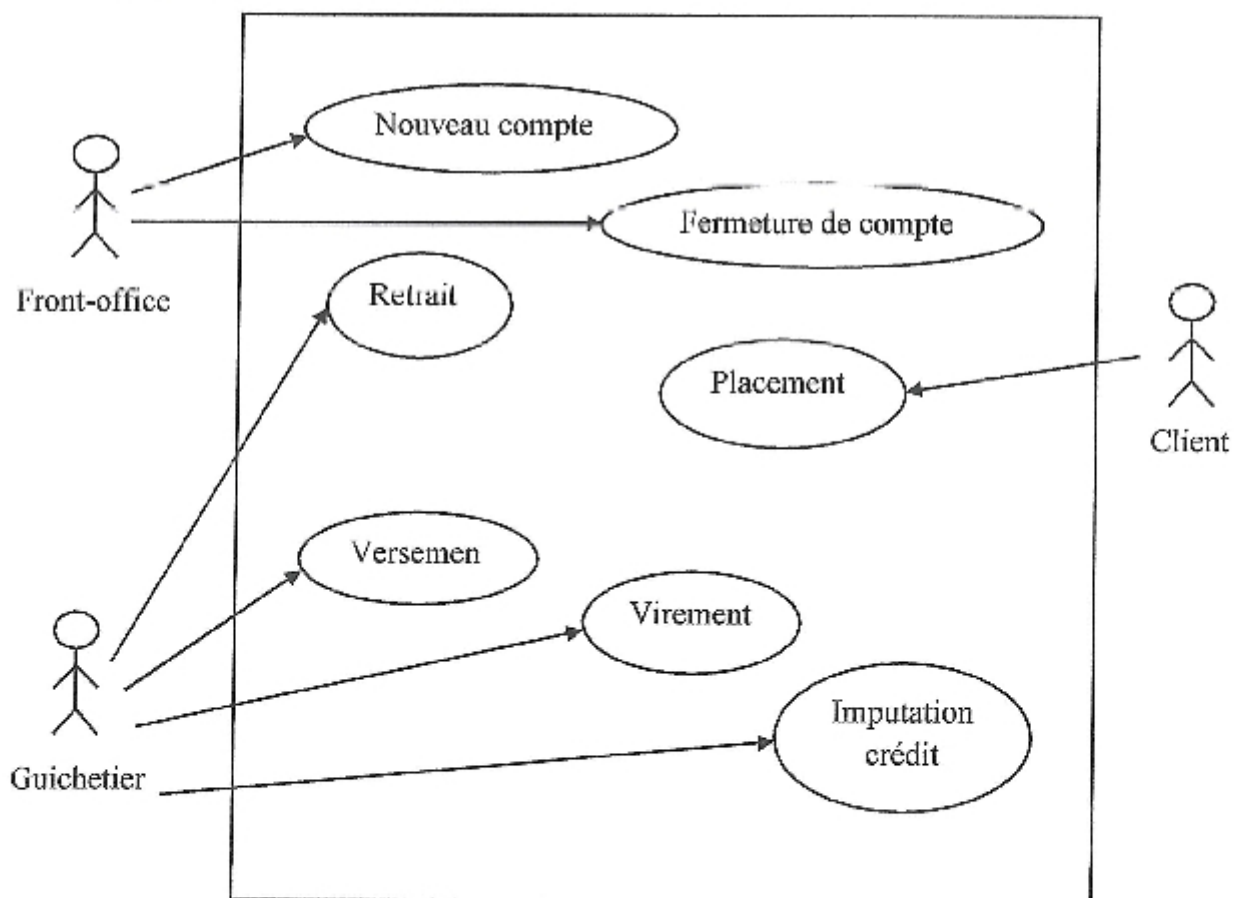


Figure 05 : cas d'utilisation générale.

3.2. Description détaillée des cas d'utilisations

3.2.1. Nouveau compte

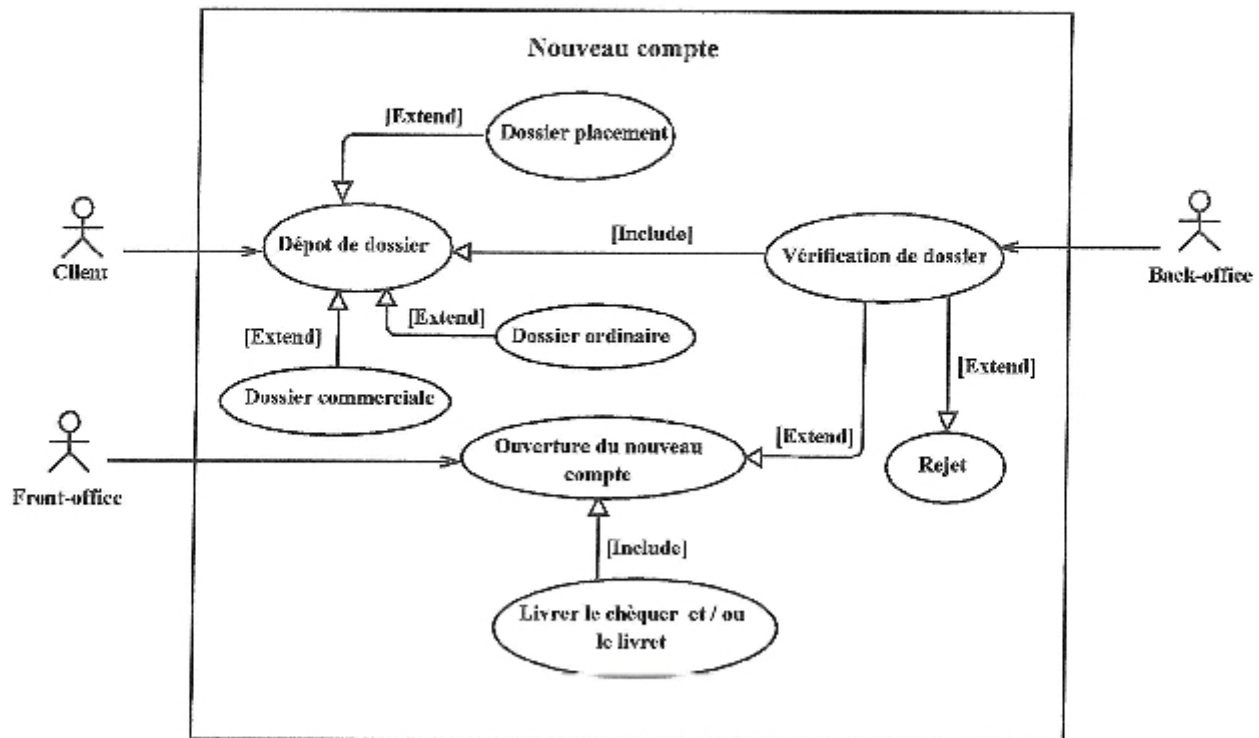


Figure 06 : Cas d'utilisation Nouveau compte.

Scénario : Ce cas d'utilisation commence lorsque le client dépose le dossier au niveau du front-office.

Enchaînement (a) : Vérification de dossier.

Le front-office transmet le dossier au back-office.

Le back-office vérifie le dossier.

Enchaînement (b) : Créer un nouveau compte.

Si le dossier est complet alors le front-office crée nouveau compte;

Sinon déclencher : [Exception01 : Dossier incomplet].

Exceptions :

[Exception01 : Dossier incomplet] : Informer le client que le dossier est incomplet.

Ce cas d'utilisation se termine lorsque le front-office livre le carnet de chèque et/ou le livret d'épargne au client.

3.2.2. Retrait

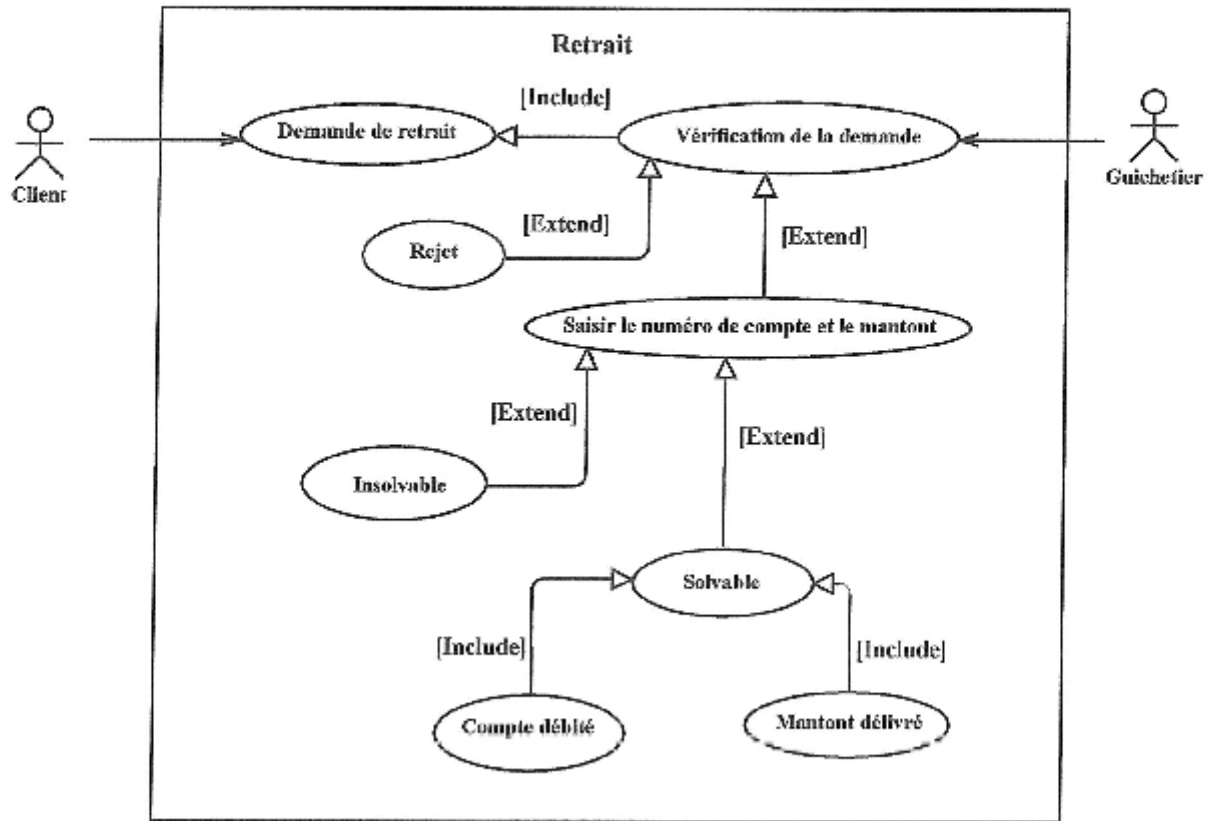


Figure 07 : Cas d'utilisation Retrait.

Scénario : Ce cas d'utilisation commence lorsque le client présente la demande de retrait (la demande fait par le chèque ou le livret) au niveau de guichetier.

Enchaînement (a) : Valider le compte.

Après la vérification le guichetier valide le compte.

Enchaînement (b) : Vérification du solde.

Si le compte est validé alors le guichetier saisie le montant ;

L'application vérifier le solde ;

Sinon déclencher : [Exception01 : Compte non valide].

Enchaînement (c) : Débiter le compte.

Si le solde > le montant alors l'application débite le compte ;

Sinon déclencher : [Exception02 : Le solde < le montant].

Exceptions :

[Exception01 : Compte non valide]: un message d'erreur est affiché sur l'écran avisant l'utilisateur que le compte est non valide.

[Exception02 : Le solde < le montant]: un message d'erreur est affiché sur l'écran avisant l'utilisateur que le solde est au montant demandé.

Ce cas d'utilisation se termine lorsque le guichetier paye le client ou l'avisant que son solde est insuffisant.

3.2.3. Versement

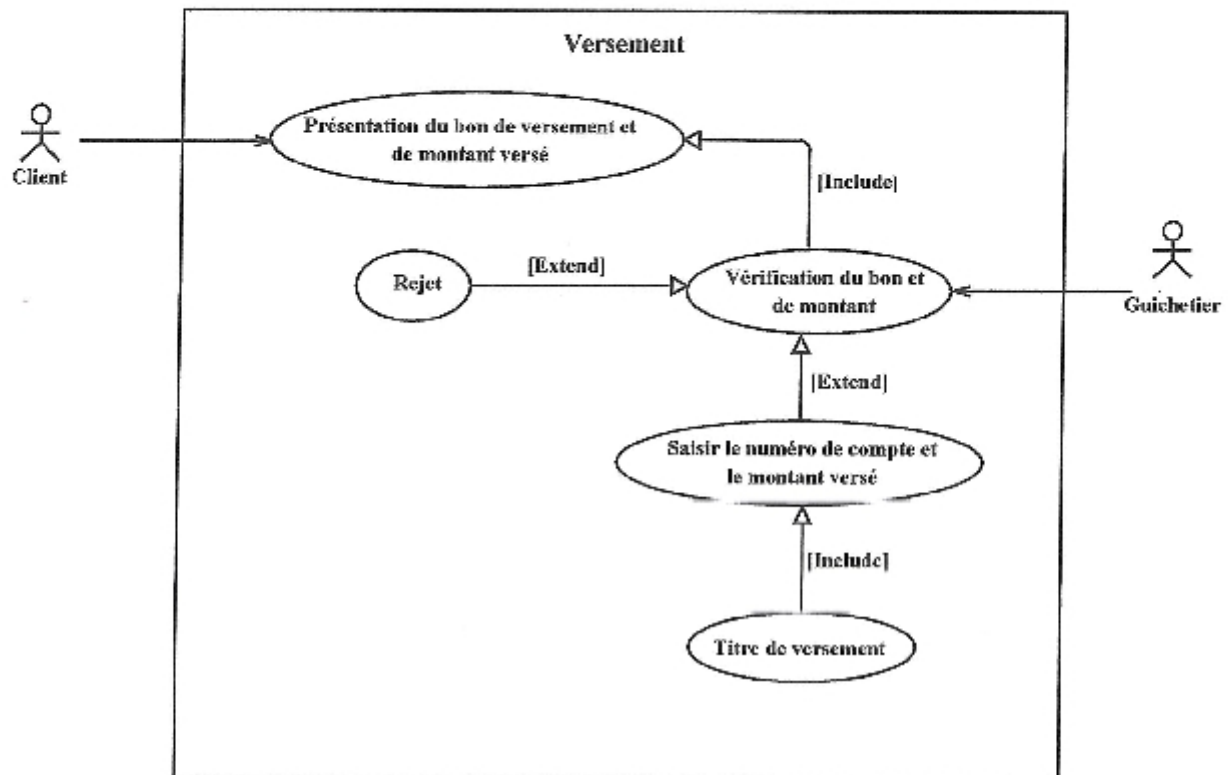


Figure 08 : Cas d'utilisation Versement.

Scénario : Ce cas d'utilisation commence lorsque le client présente le bon de versement et le montant versé.

Enchaînement (a) : Validation du compte.

Après la vérification, le guichetier valide le compte.

Enchaînement (b) : Créditer le compte.

Si le compte est validé le guichetier saisie le montant versé ;

L'application crédite le montant au compte ;

Si non déclencher : [Exception 01 : Compte non valide].

Exceptions :

[Exception 01 : Compte non valide] : Un message d'erreur est affiché à l'écran avisant l'utilisateur que le compte est non valide.

Ce cas d'utilisation se termine lorsque le guichetier livre le titre de versement au client.

3.2.4. Virement

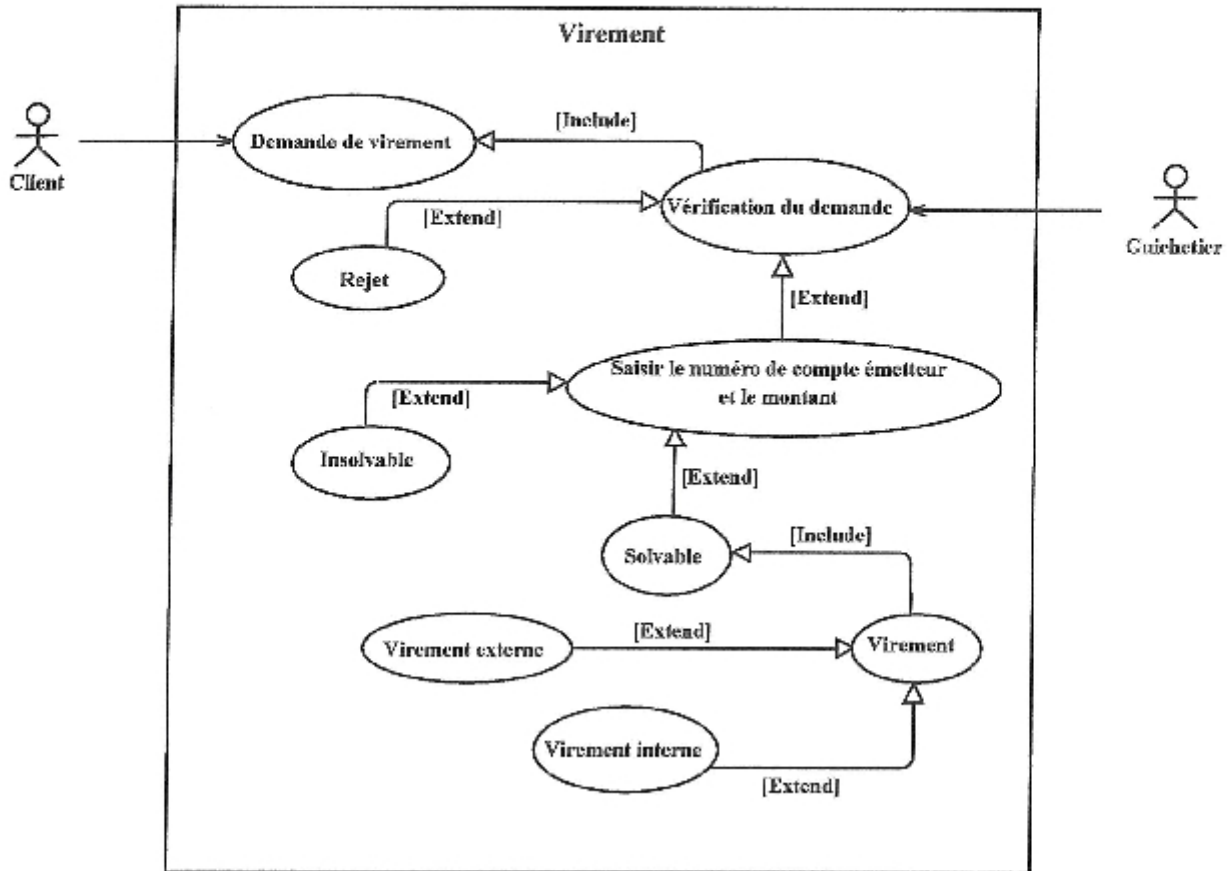


Figure 09 : Cas d'utilisation Virement.

Scénario : Ce cas d'utilisation commence lorsque le client présente la demande de virement.

Enchaînement (a) : Valider le compte émetteur.

Après la vérification le guichetier valide le compte.

Enchaînement (b) : Vérification du solde.

Si le compte est validé alors le guichetier saisit le montant versé et le numéro de compte récepteur si le virement est interne ;

L'application vérifie le solde ;

Sinon déclencher : **[Exception01 : Compte non valide]**.

Enchaînement (c) : Transfert du montant versé.

Si le solde est supérieur au montant alors l'application transfère le montant au compte récepteur en débitant le compte émetteur et en créditant le compte récepteur si le virement est interne, commencer l'opération de virement externe sinon ;

Sinon déclencher : **[Exception02 : Le solde < le montant]**.

Exceptions :

[Exception01 : Compte non valide]: un message d'erreur est affiché sur l'écran avisant l'utilisateur que le compte est non valide.

[Exception02 : Le solde < le montant]: un message d'erreur est affiché sur l'écran avisant l'utilisateur que le solde est inférieur au montant à verser.

Ce cas d'utilisation se termine lorsque le guichetier livre le titre de virement ou le refus de l'opération de virement.

3.2.5. Placement

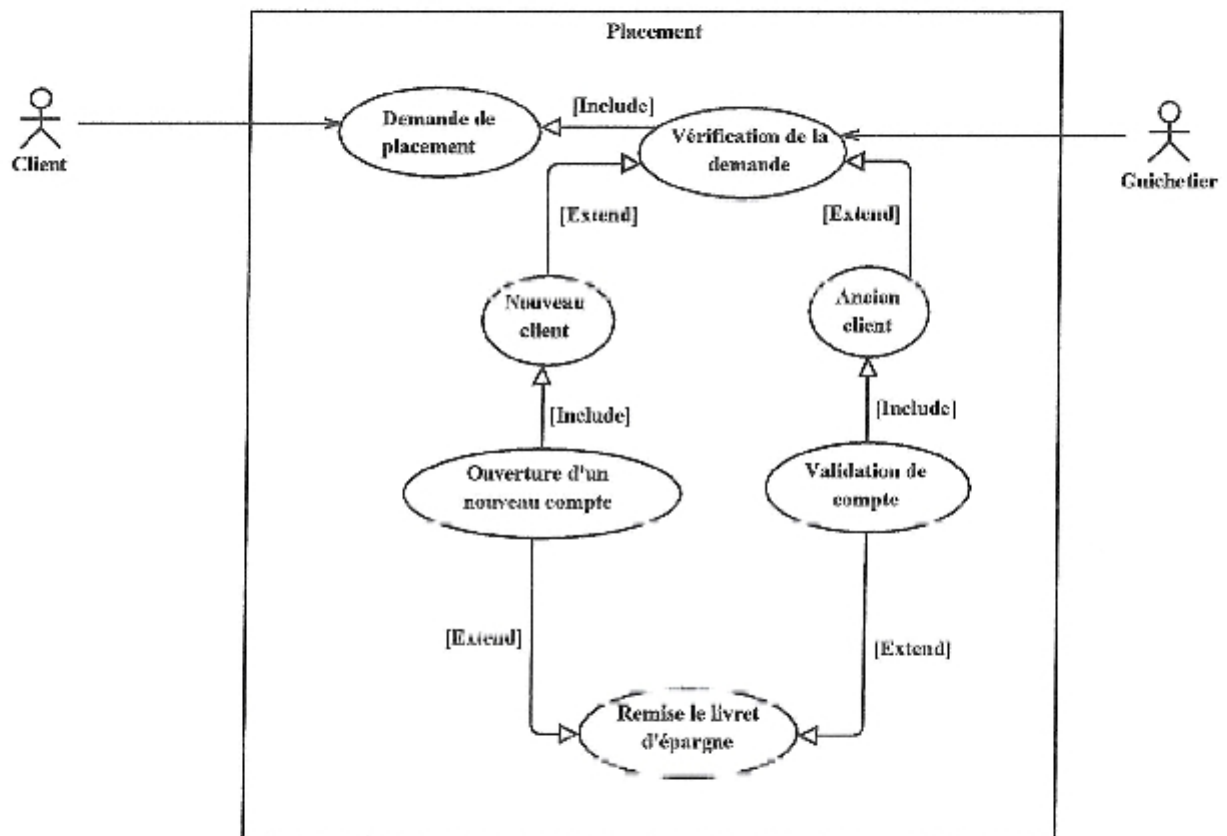


Figure 10 : Cas d'utilisation Placement.

Scénario : Ce cas d'utilisation commence lorsque le client demande un placement au niveau du guichetier.

Enchaînement (a) : Validation du compte.

Si le client possède déjà un compte alors le guichetier directement validé le compte ;

Sinon déclencher : **[Exception 01 : Nouveau client]**.

Enchaînement (b) : Créditer le montant au compte.

Si le compte valide alors le guichetier crédite montant au compte ;

4. Diagrammes d'activité

Le diagramme d'activité est un diagramme comportemental d'UML, permettant de représenter le déclenchement d'événements en fonction des états du système et de modéliser des comportements parallèles (multiprocessus). [Abdat et mahdaoui, 2007c]

4.1. Nouveau compte

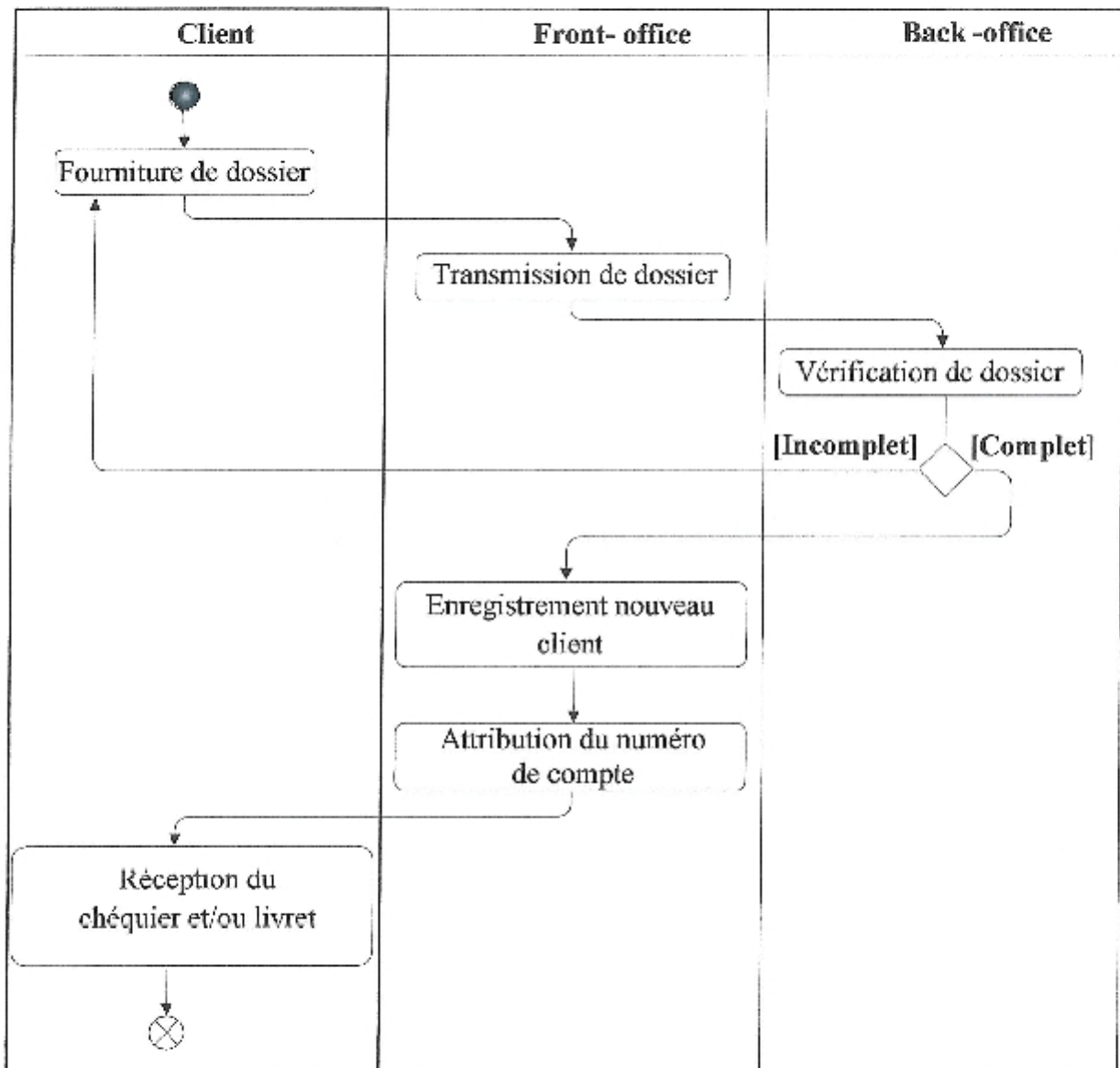


Figure13 : Diagramme d'activité Nouveau compte.

4.2. Retrait

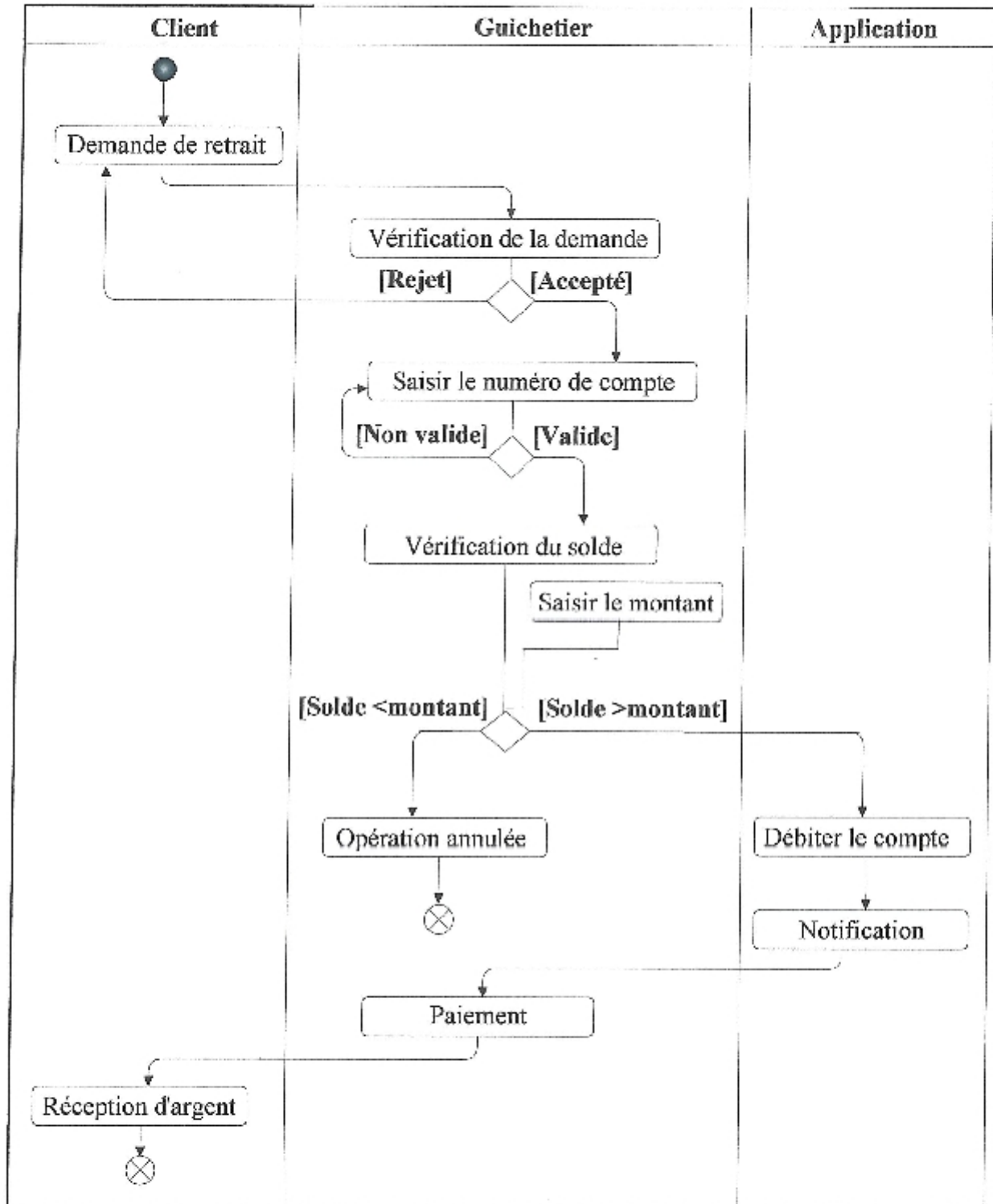


Figure14 : Diagramme d'activité Retrait.

4.3. Versement

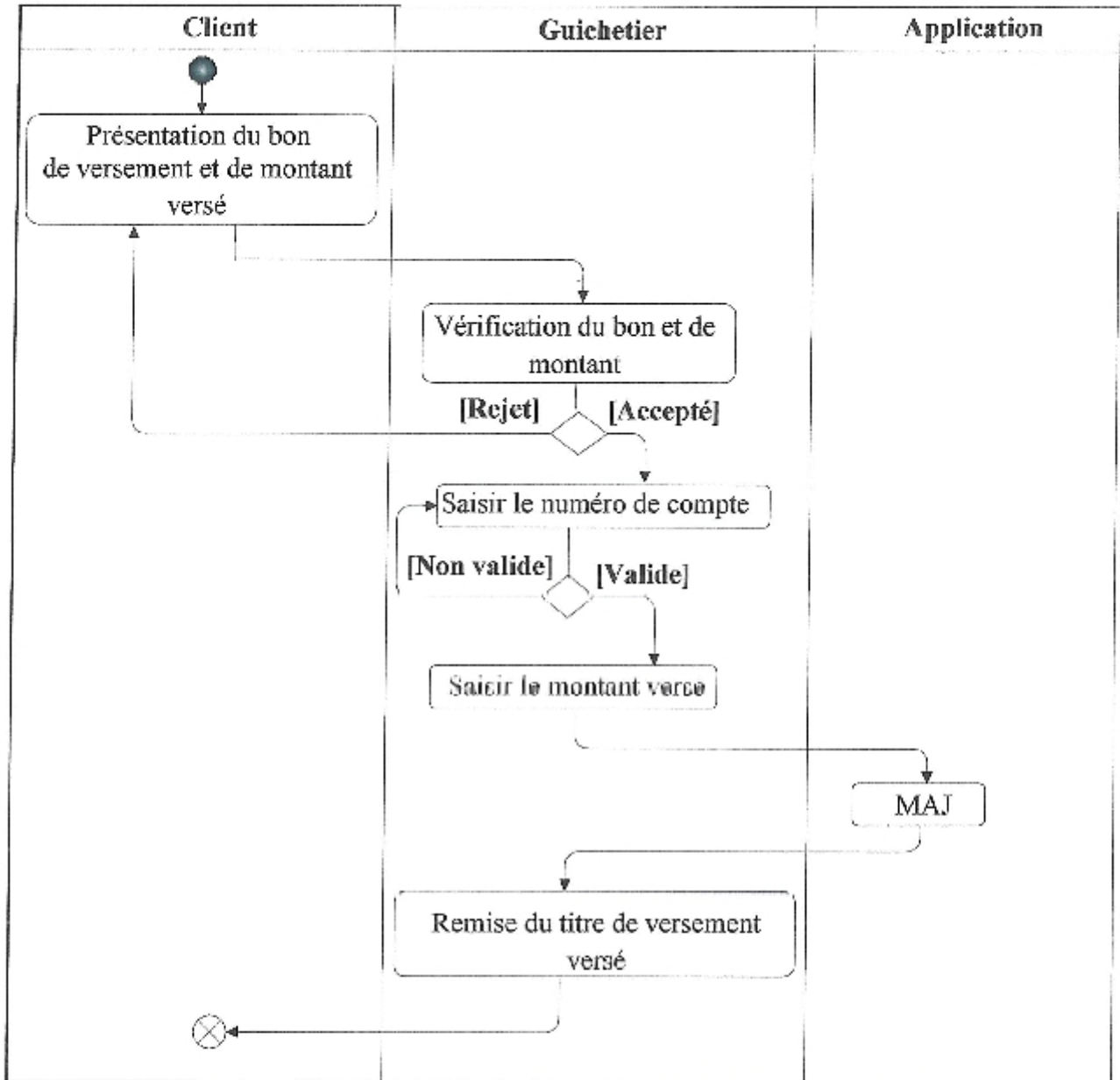


Figure15 : Diagramme d'activité Versement.

4.4. Virement

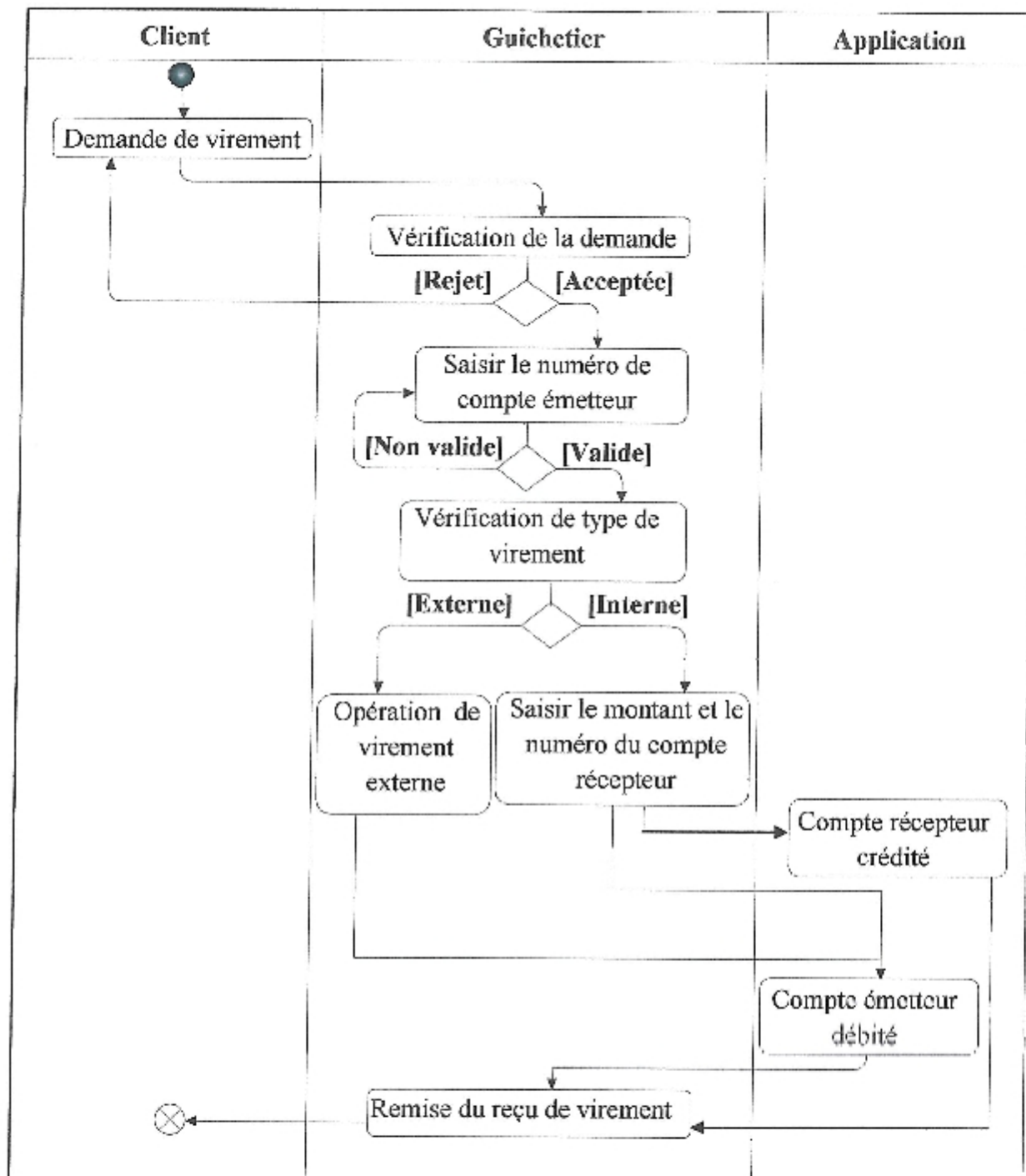


Figure16: Diagramme d'activité Virement.

4.5. Placement

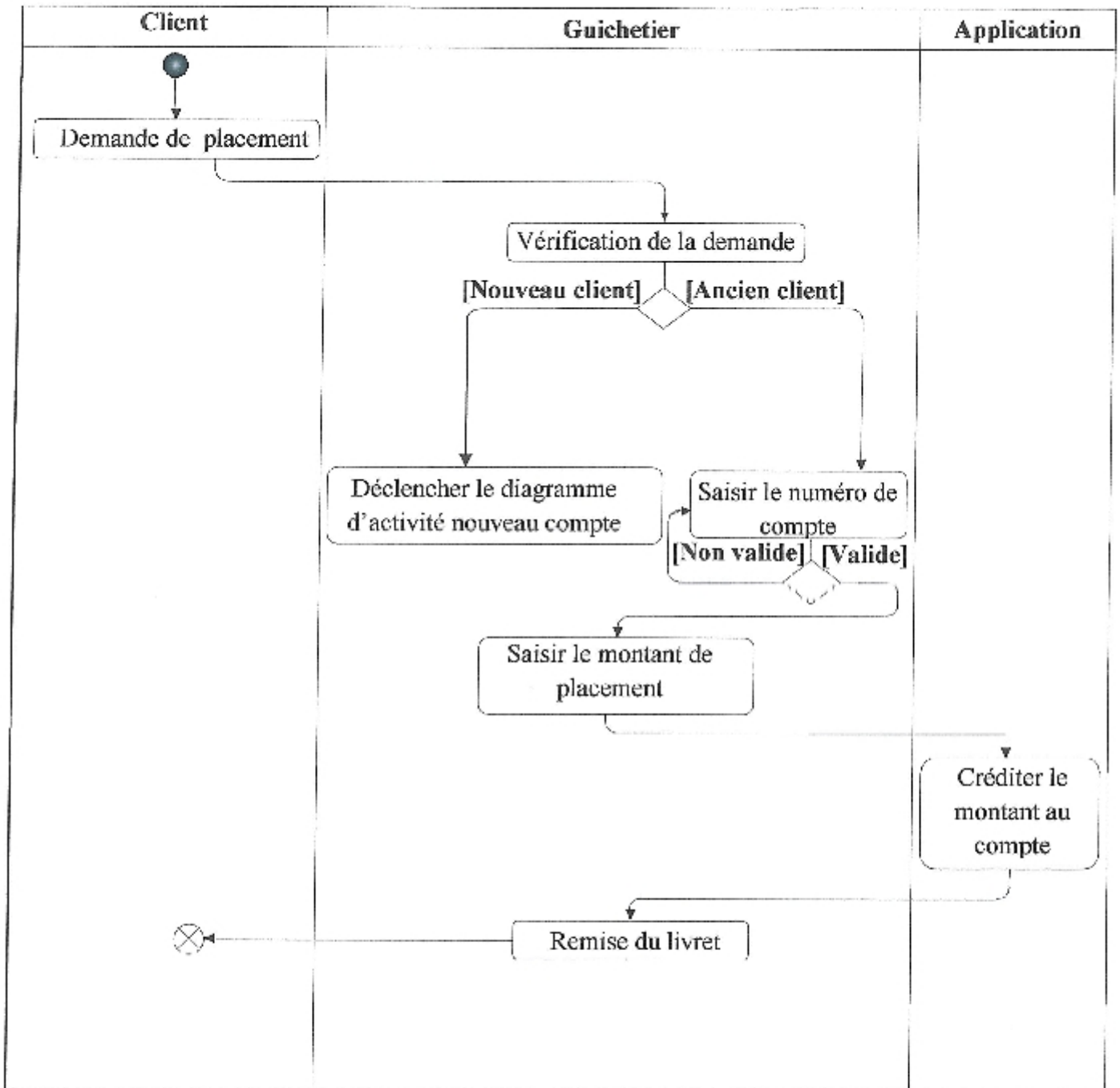


Figure17 : Diagramme d'activité Placement.

4.6. Imputation crédit

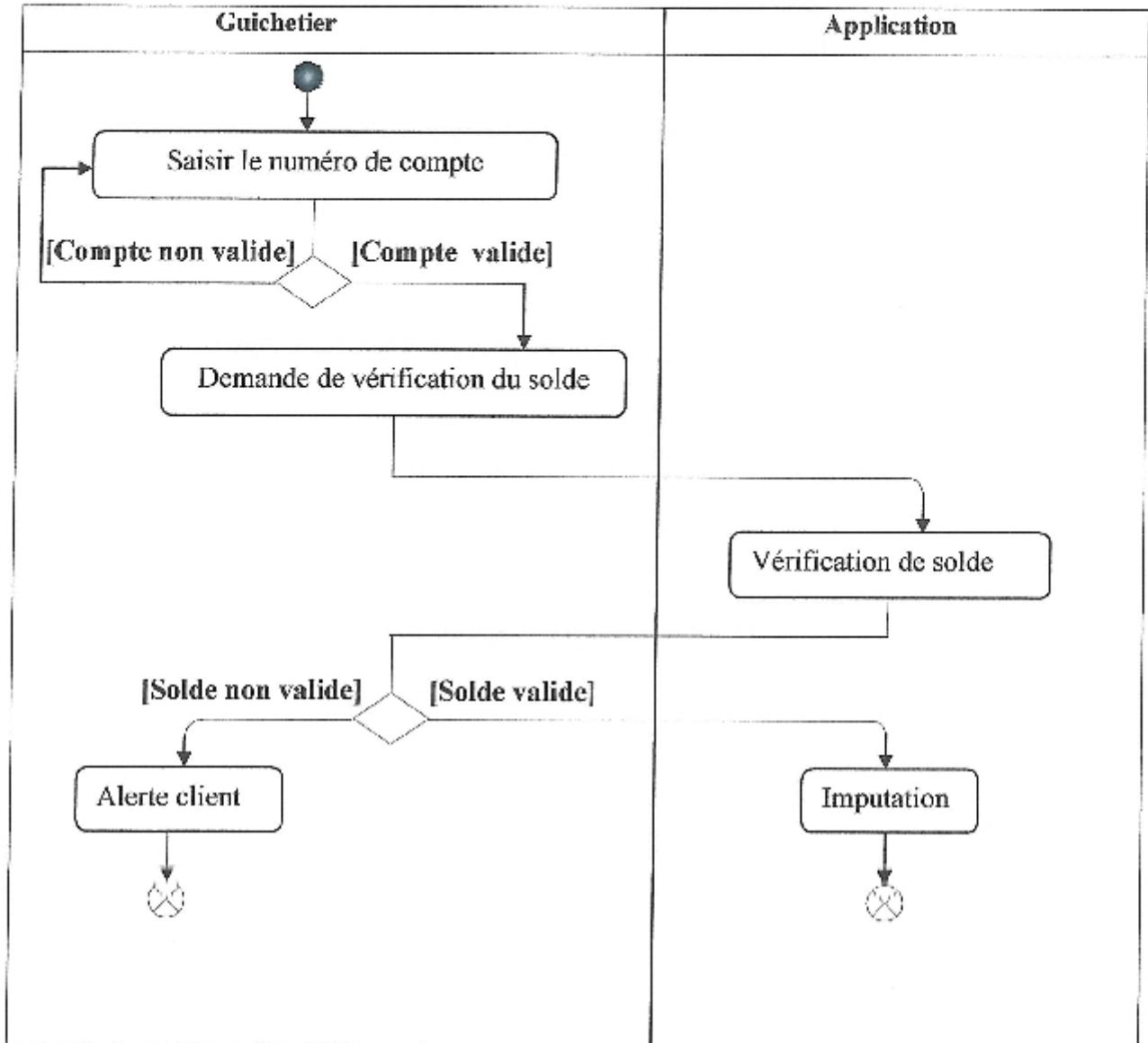


Figure 18 : Diagramme d'activité Imputation crédit.

4.7. Fermeture de compte

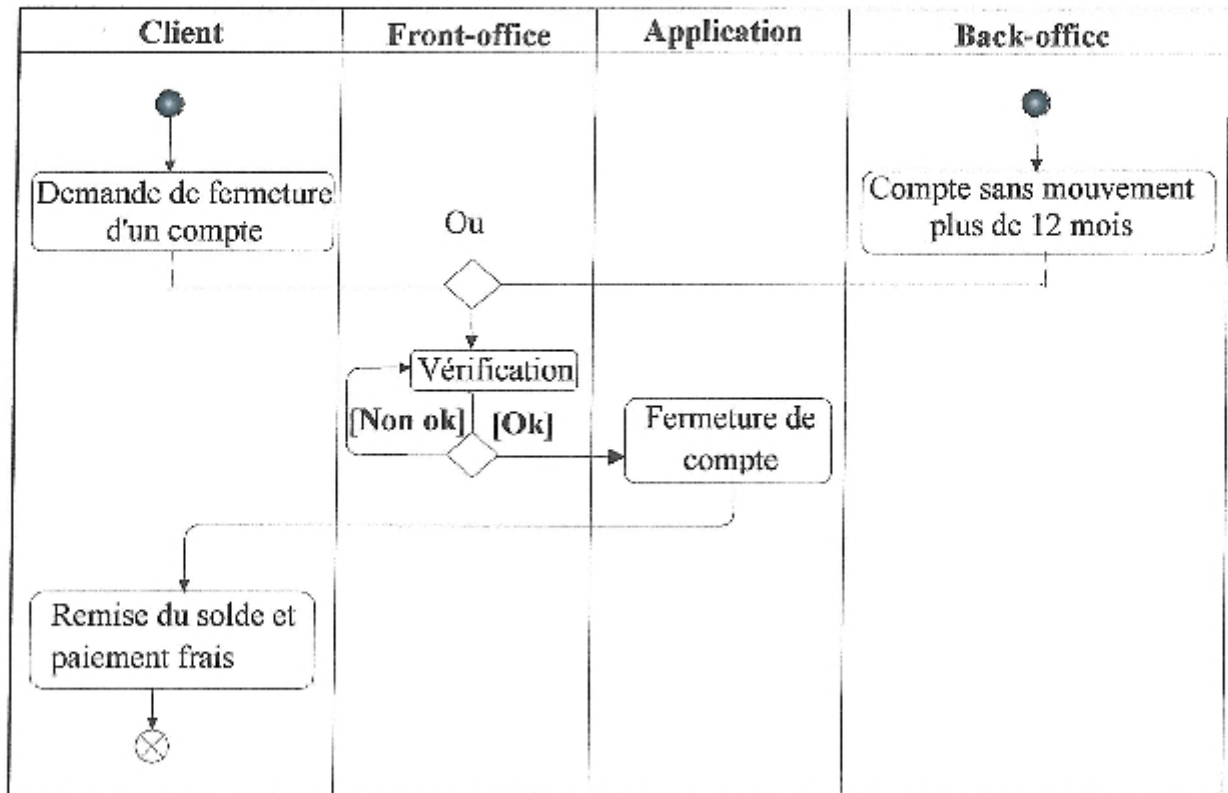


Figure19 : Diagramme d'activité fermeture de compte.

5. Diagrammes de séquence

Un diagramme de séquence est la représentation séquentielle du déroulement des traitements et des interactions entre les éléments du système et/ou de ses acteurs. [Abdat et mahdaoui, 2007d]

5.1. Nouveau compte

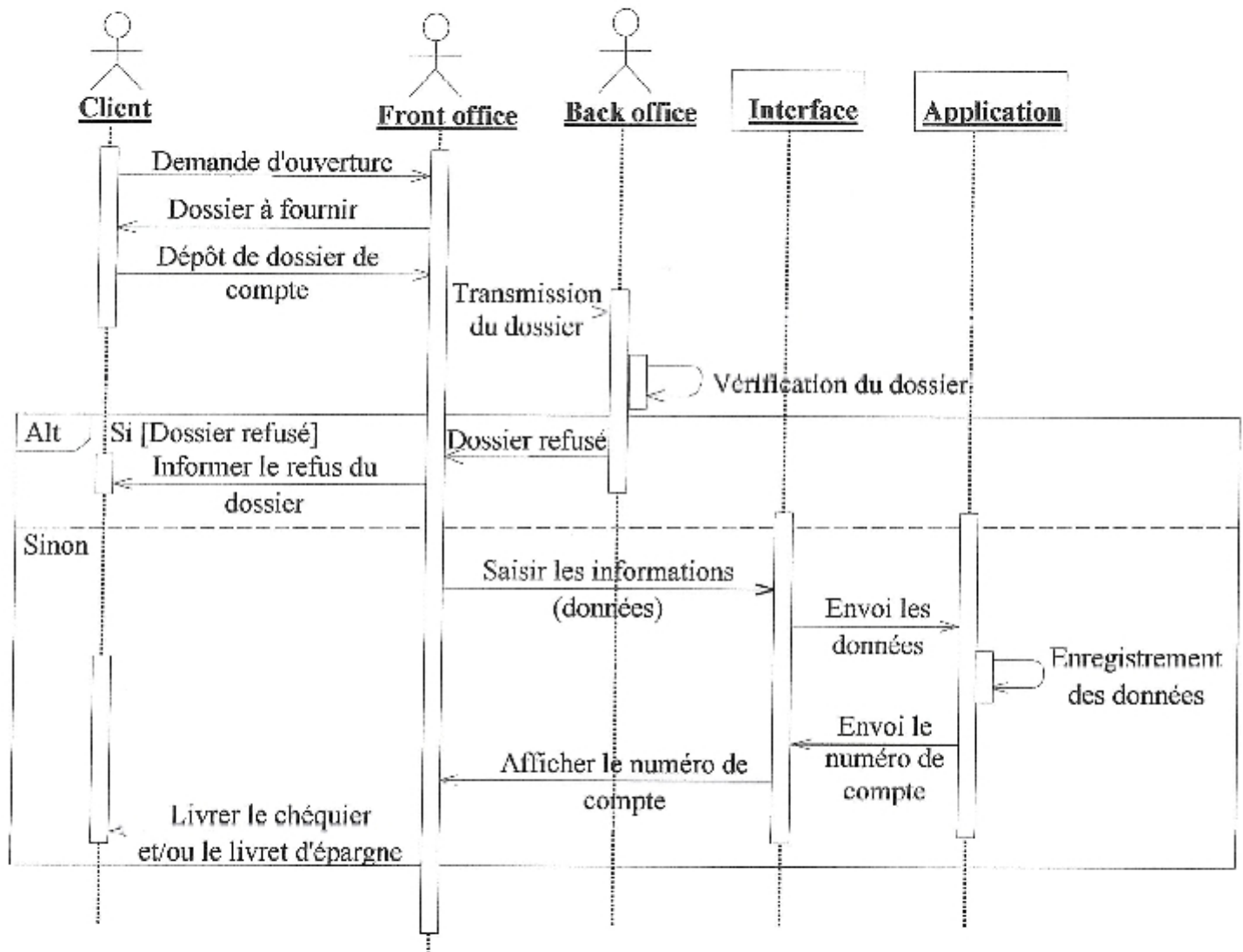


Figure20 : Diagramme de séquence Nouveau compte.

5.2. Retrait

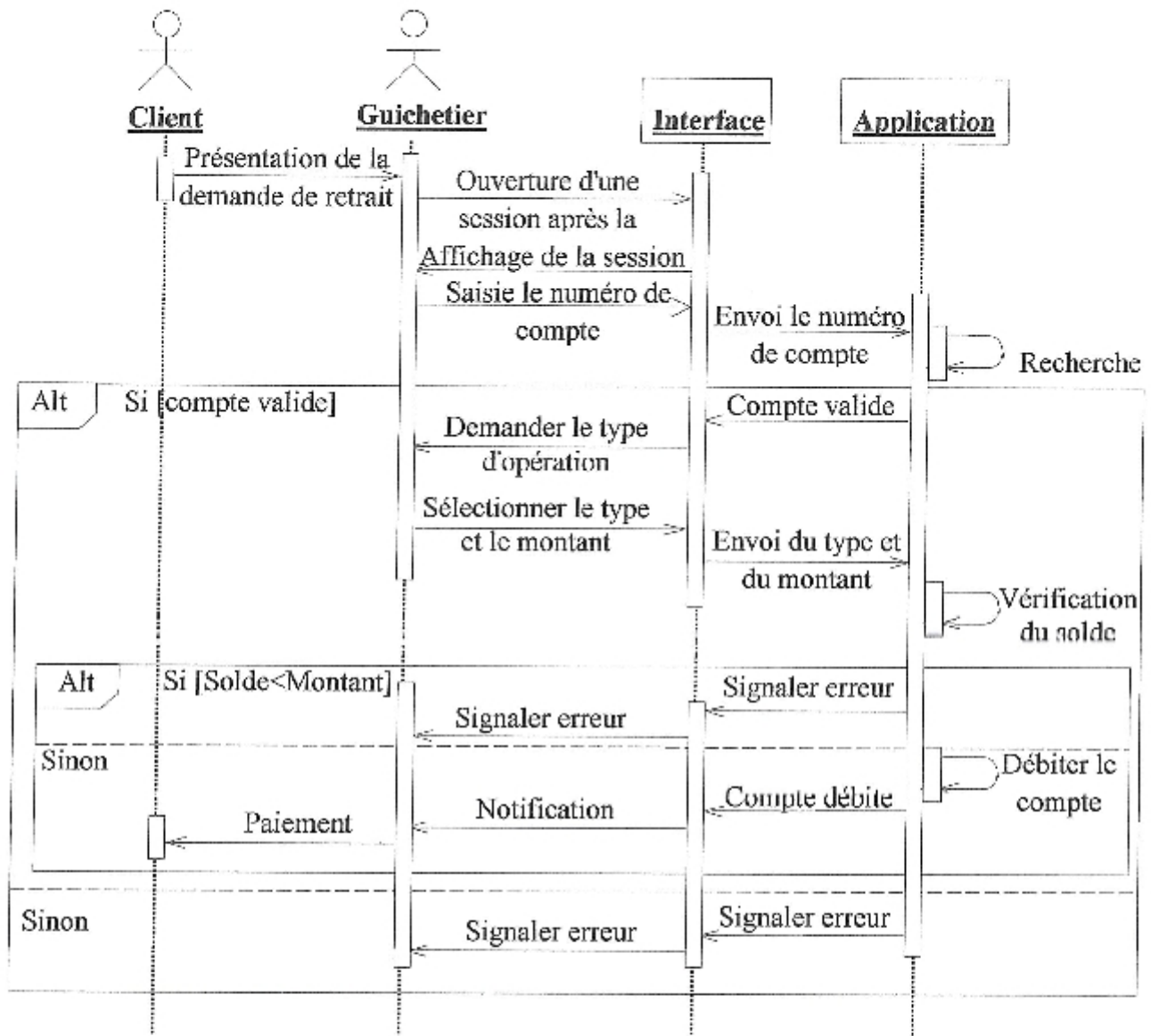


Figure21 : Diagramme de séquence Retrait.

5.3. Versement

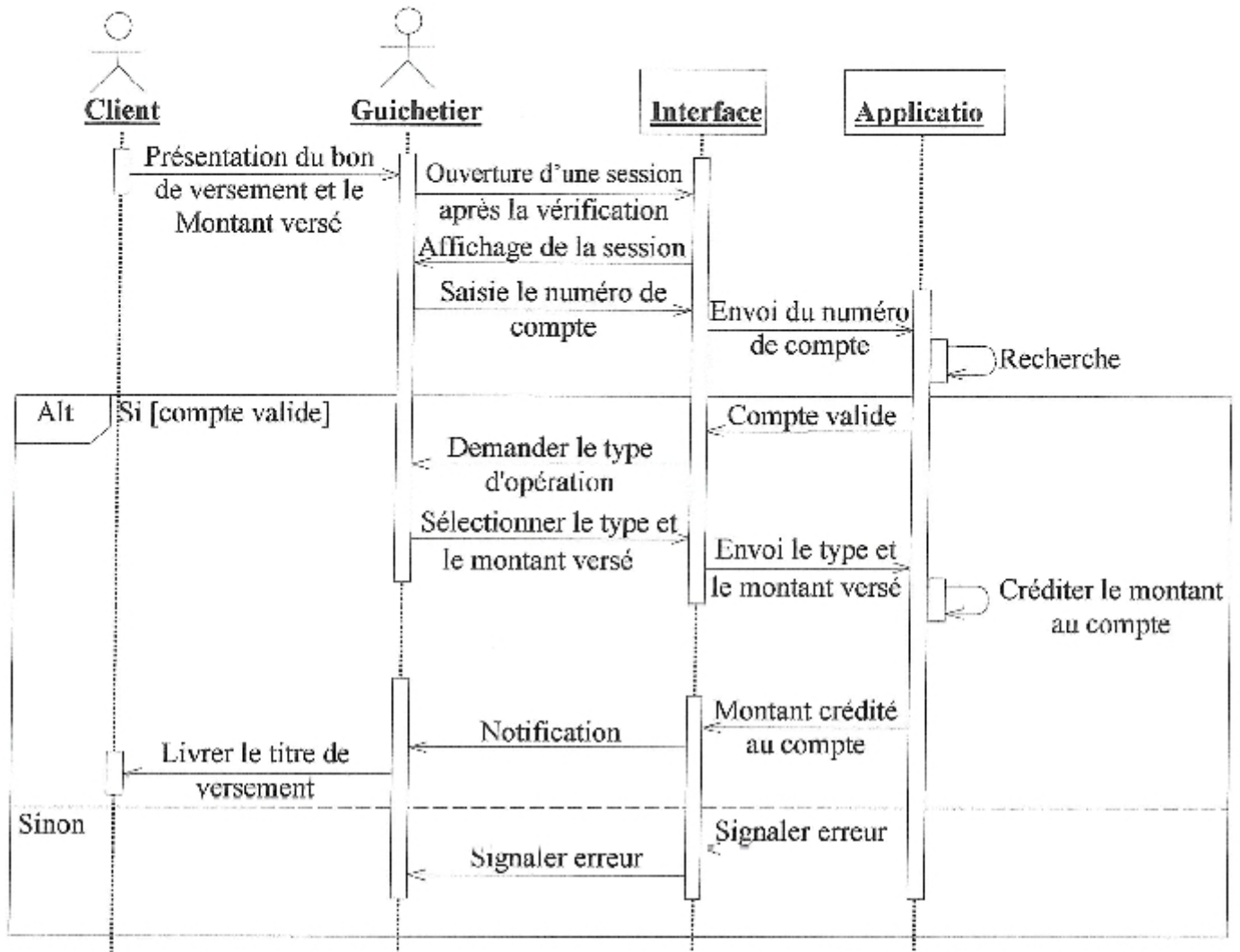


Figure22: Diagramme de séquence Versement.

5.4. Virement

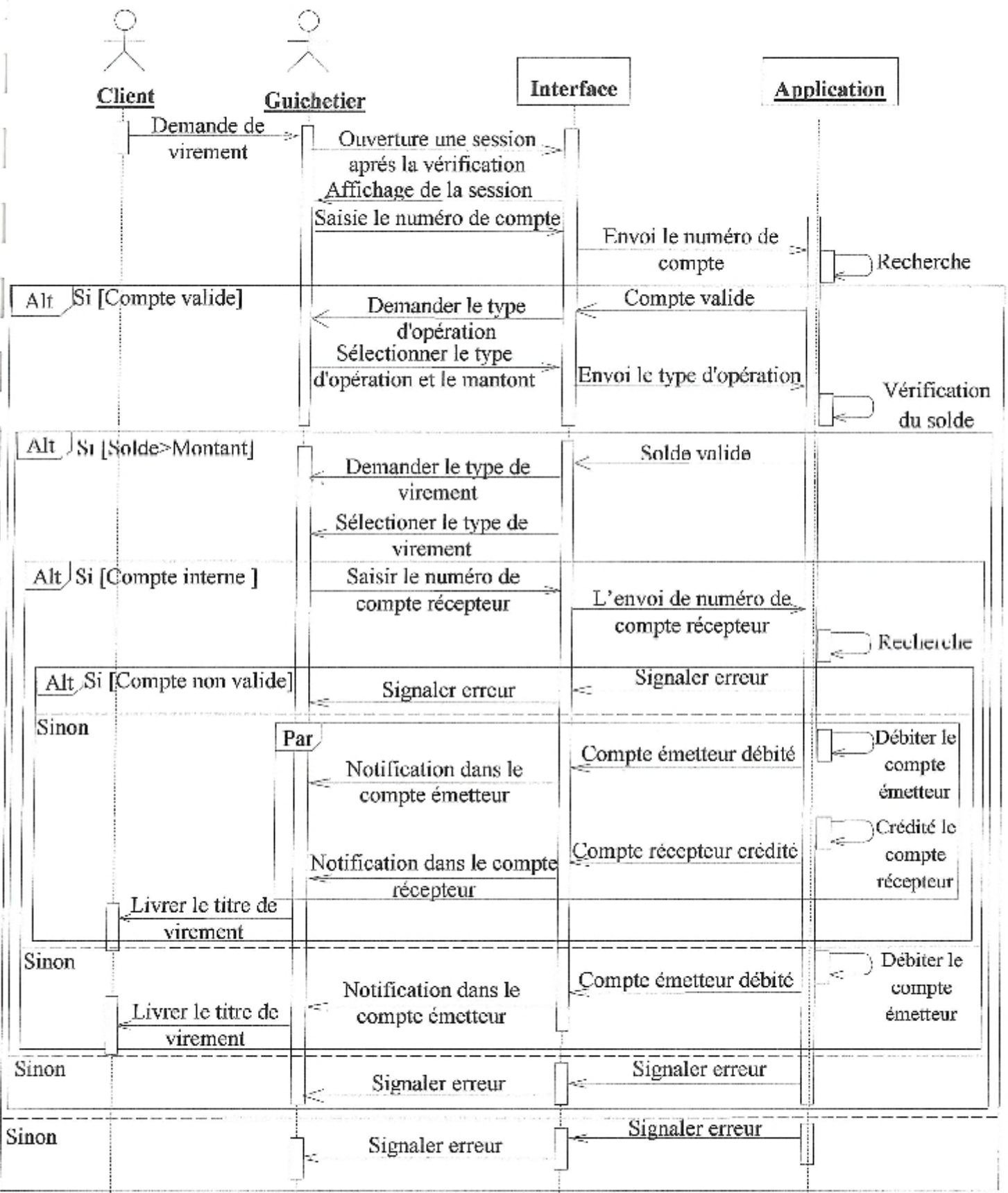


Figure23 : Diagramme de séquence Virement.

5.5. Placement

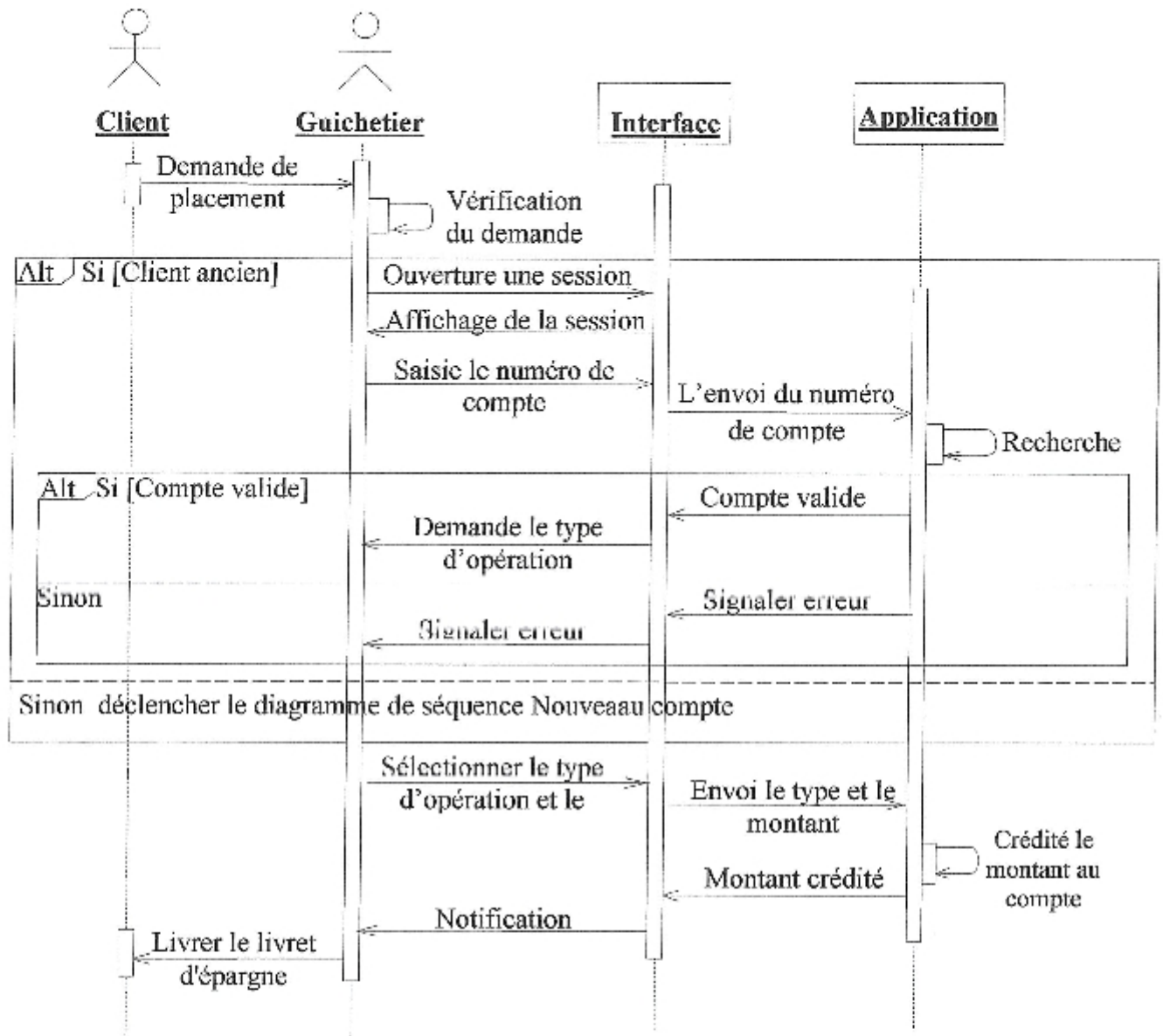


Figure24 : Diagramme de séquence Placement.

5.6. Imputation crédit

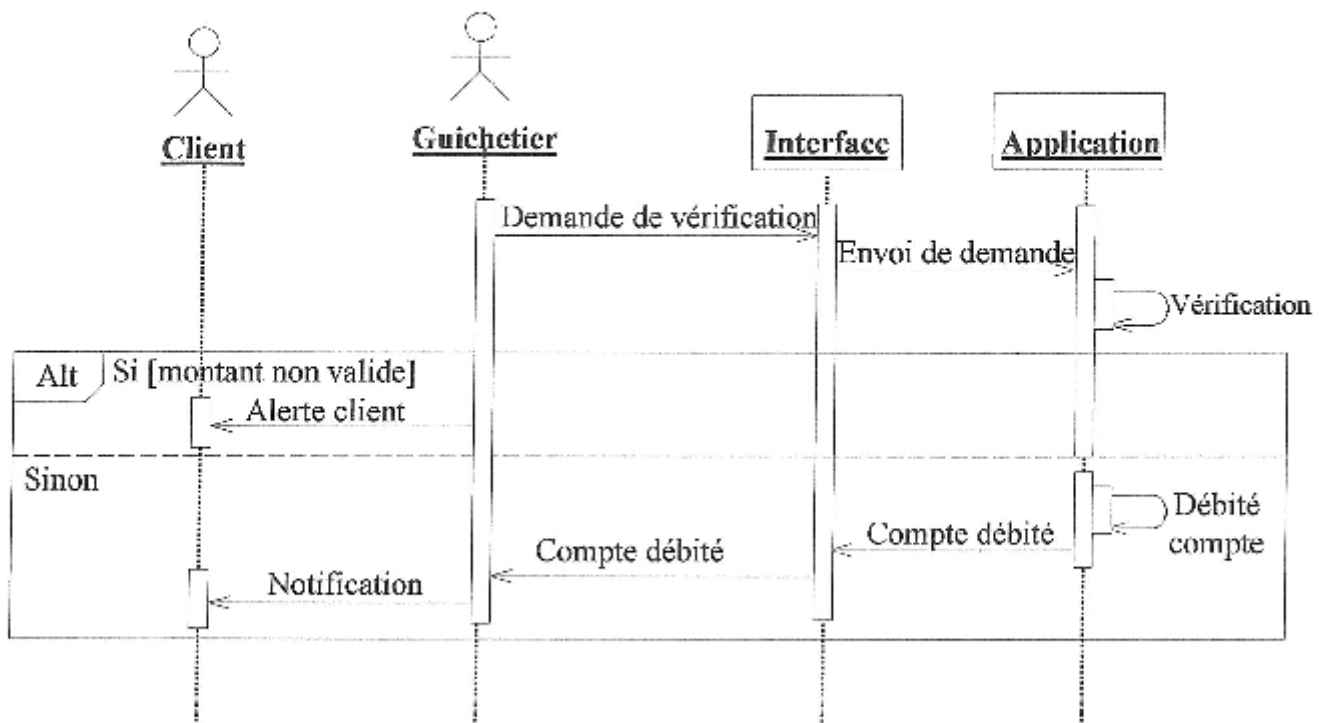


Figure25 : Diagramme de séquence Imputation crédit.

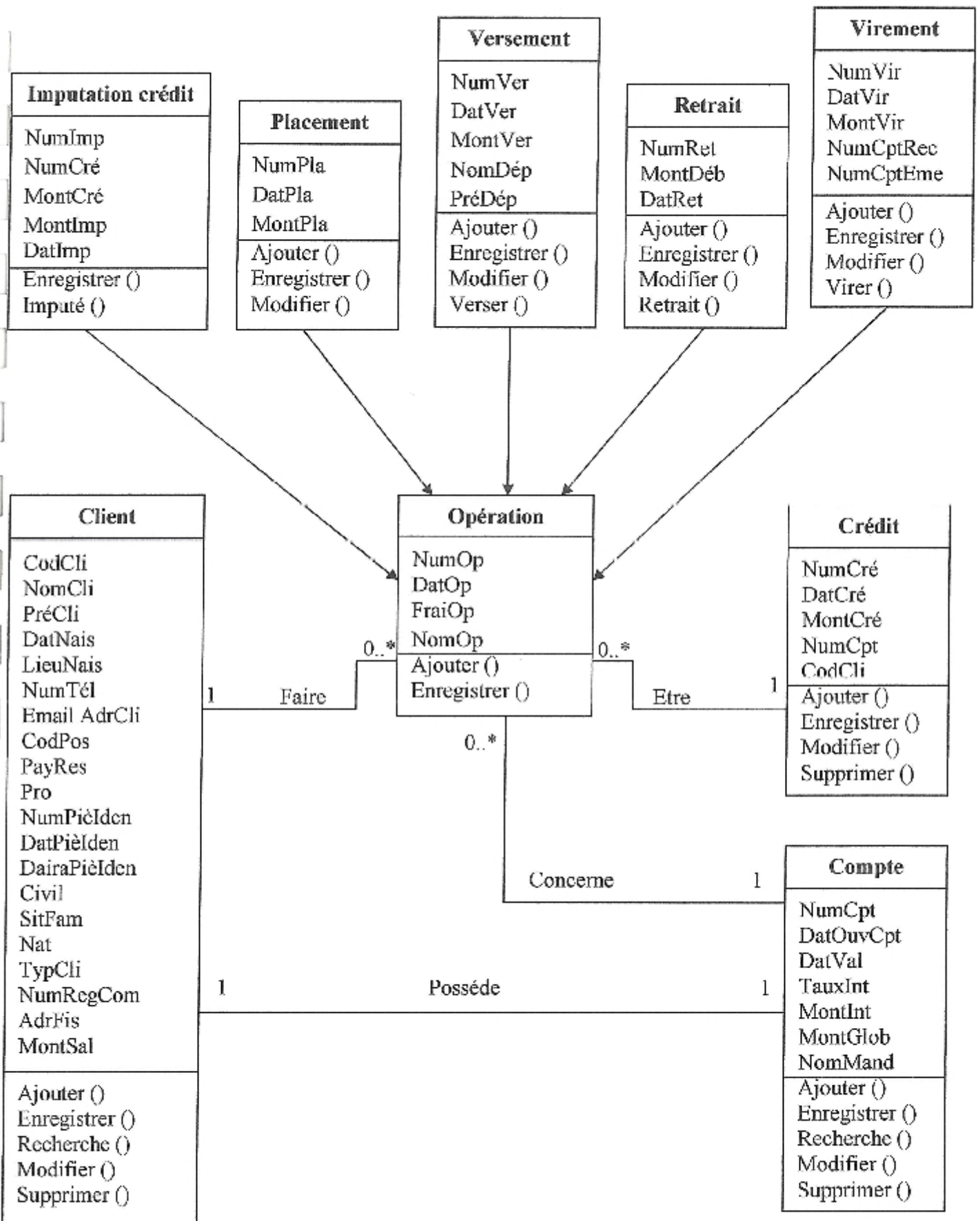


Figure27 : Diagramme de classes

6.2. Identification du dictionnaire de données

Nom	Désignation	Type	Taille	Remarque
NumCpt	Numéro de compte	N	20	
DatOuvCpt	Date d'ouverture de compte	D	08	JJ/MM/AAAA
DatVal	Date de valeur	D	08	JJ/MM/AAAA
TauxInt	Taux d'intérêt	N	05	
MontInt	Montant d'intérêt	AN	10	
MontGlob	Montant globale	AN	15	
NomMand	Nom mandataire	A	20	
NumOp	Numéro d'opération	N	05	
NomOp	Nom d'opération	A	20	
DatOp	Date d'opération	D	08	JJ/MM/AAAA
FraiOp	Frais d'opération	N	05	
NumVer	Numéro de versement	N	05	
DatVer	Date de versement	D	08	JJ/MM/AAAA
MonVer	Montant versé	AN	10	
NomDép	Nom de déposant	A	20	
PréDép	Prénom de déposant	A	20	
NumRet	Numéro de retrait	N	05	
DatRet	Date de retrait	D	08	JJ/MM/AAAA
MonDéb	Montant débité	AN	10	
NumVir	Numéro de virement	N	05	
DatVir	Date de virement	D	08	JJ/MM/AAAA
MontVir	Montant de virement	AN	10	
NumCptRec	Numéro de compte récepteur	N	20	
NumCptEmc	Numéro de compte émetteur	N	20	
CodCli	Code client	N	10	
NomCli	Nom de client	A	20	

PréCli	Prénom de client	A	20	
DatNais	Date de naissance	D	08	JJ/MM/AAAA
LieuNais	Lieu de naissance	AN	20	
NumTél	Numéro de téléphone	N	15	
AdrCli	Adresse de client	AN	40	
CodPos	Code de postale	N	08	
PayRes	Pays résidence	A	20	
Pro	Profession	A	30	
NumPièIden	N°pièce d'identité	AN	12	
DatPièIden	Date pièce d'identité	D	08	JJ/MM/AAAA
DairaPièIden	Daira pièce d'identité	A	30	
Civil	Civilité	A	10	
SitFam	Situation familiale	A	10	
Nat	Nationalité	A	30	
TypCli	Type de client	A	15	
NumRegCom	Numéro registre commerce	AN	15	
AdrFis	Adresse fiscale	AN	20	
MontSal	Montant revenu mensuel (salaire)	AN	10	
NumCré	Numéro de crédit	N	10	
MontCré	Montant crédité	AN	15	
DatImp	Date d'imputation	D	08	JJ/MM/AAAA
DatCré	Date de crédit	D	08	JJ/MM/AAAA
MontImp	Montant d'imputation	AN	15	

Tableau 03 : dictionnaire des données

Chapitre III : Codage et tests

Dans ce chapitre nous allons étaler la phase de codage et tests. Cette dernière représente le fruit de toute l'étude effectuée auparavant.

1. Codage

Le résultat du codage est la base de données « CBBADR » et l'application « SCBBADR ».

1.1. Présentation du langage utilisé (Borland Delphi7)

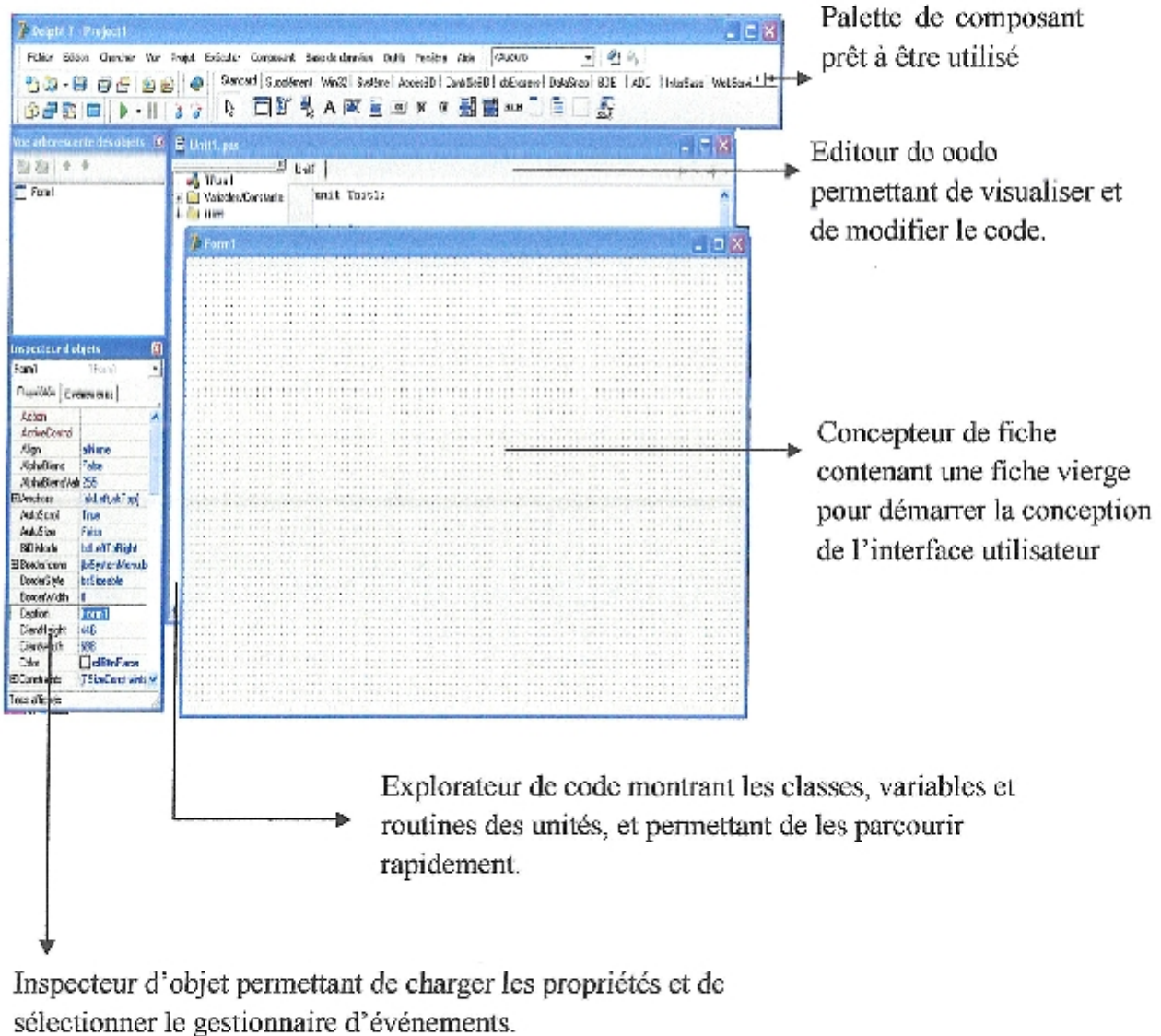


Figure28 : L'interface du langage DELPHI7.

Delphi est un langage de programmation permettant de développer des applications pour le système d'exploitation Microsoft Windows. Il incarne la suite logique de légendaire Turbo PASCAL. Delphi est un outil moderne, qui fait appel à une conception visuelle de l'application, à la programmation objet. De plus, il prend en charge le maintien automatique une partie du code source [IDEI, 2001]

1.2. Présentation du système de gestion des bases de données Microsoft Access

Pour des raisons de rapidité et de simplicité de déploiement, nous choisis Microsoft Access pour implémenter notre base de données «CBBADR». Ainsi les décideurs de la BADR banque pourrons vite voir l'apport de notre proposition rapidement. S'ils optent pour notre solutions la base de données «CBBADR» pourra être migrée vers Microsoft SQL serveur.

Microsoft Access est un système de gestion de base de données monoposte très performant. C'est produit de Microsoft incorporé dans office.

Pour notre application nous avons utilisé la version XP 2003.

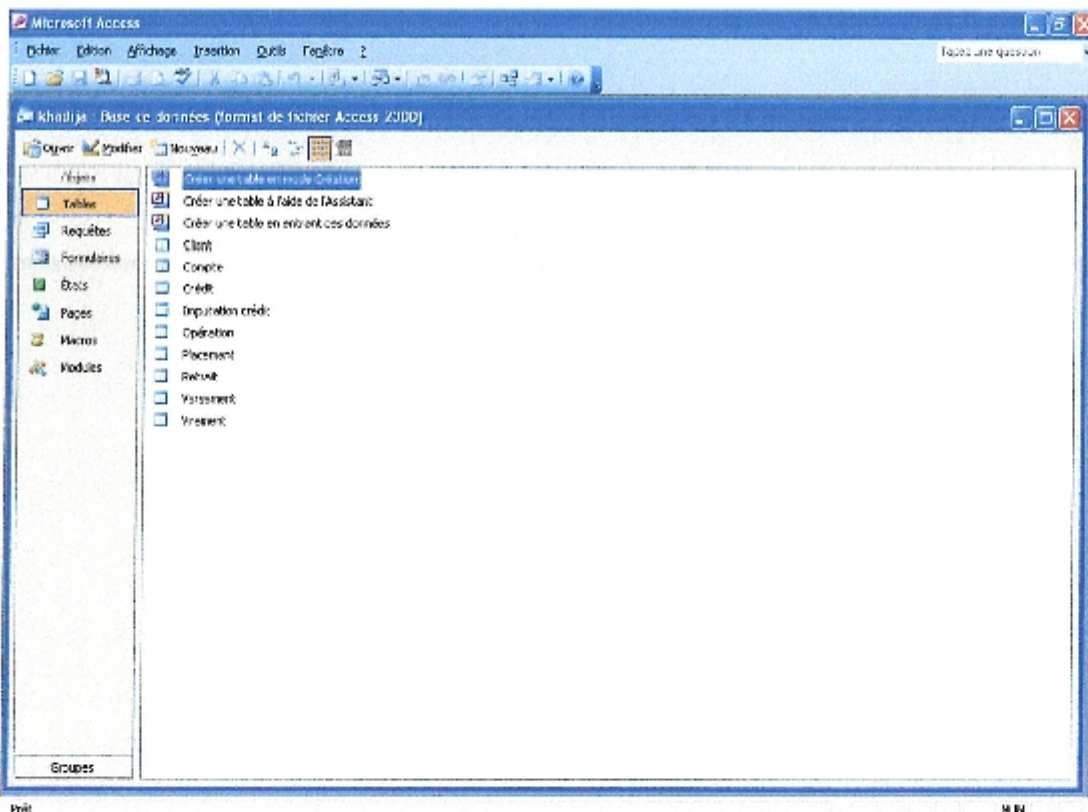


Figure29 : L'interface du SGBD Microsoft Access.

1.3. Vue sur le schéma relationnel de la base de données «CBBADR»

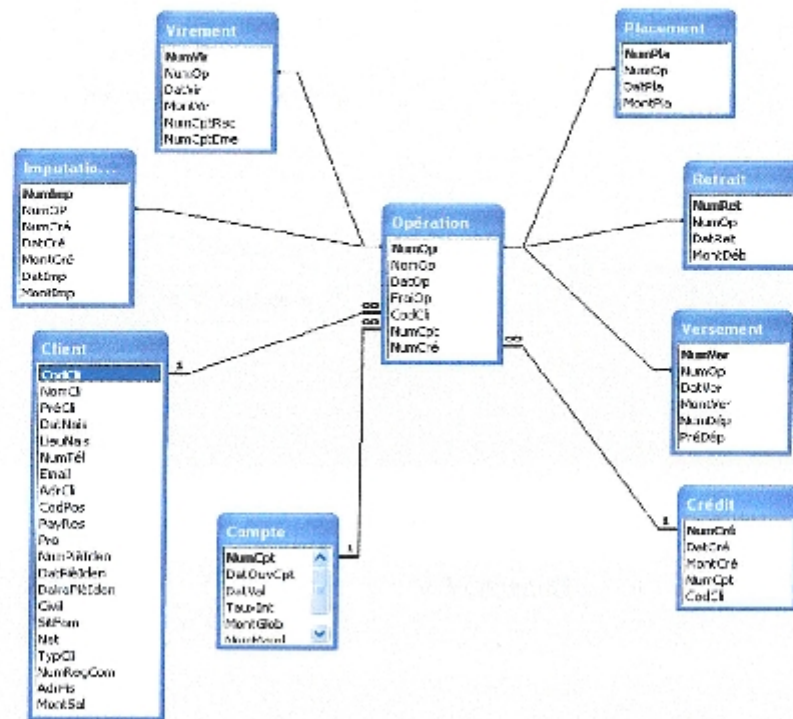


Figure30 : Vue sur le schéma relationnel de la base de données «CBBADR».

1.4. Vue sur l'implémentation de l'application «SCBBADR»

Le menu général de l'application «SCBBADR» et le suivant ;



Figure31 : Vue sur l'implémentation de « SCBBADR».

Conclusion

Le travail que nous avons présenté dans le présent mémoire, définit les différentes étapes d'analyse, de conception et de réalisation d'un système de suivi de compte bancaire unifié au niveau de la banque BADR de la wilaya de Guelma.

Pour cela nous avons effectué un stage au niveau de ladite banque. Ce stage a été pour nous une expérience exceptionnelle tant au plan professionnel que personnel. On a fait la connaissance de personnes accueillantes, disponibles et généreuses qui ont fait de ce stage une grande réussite.

En parallèle avec ce stage, nous avons appris à travailler et montré l'intérêt de la modélisation avec le processus unifié 2TUP (Two Track Unified Process).

Nous nous sommes fixé l'objectif de faire la transition entre le système en place, basé sur une multitude de comptes pour chaque client selon ses besoins (ordinaire, placement, crédit, etc....), et un système basé sur un compte bancaire unifié.

Il faut noter que toutes les étapes du processus 2TUP ont été parcourues à l'exception de la dernière et qui est la Recette que ne sera réalisée qu'après la mise en place du système conçu.

Enfin, nous estimons que nous avons parcouru un grand chemin vers l'accomplissement de cet objectif et nous espérons que cette solution sera prise en compte dans les horizons des décideurs de la BADR banque.

Bibliographie

[Kazi Aouel et Rostane, 2007a] : Kazi Aouel Bassim et Rostane Zakaria. Suivre des enseignements du LMD par application de la méthode 2TUP. Faculté des Sciences de l'Ingénieur Département d'Informatique- Université Abou Bekr Belkaid de Tlemcen, novembre 2007. Page12.

[Frédéric, 2001] : Digallo Frédéric. Méthodologie des systèmes d'information - UML Cours du Cycle Probatoire. CNAM ANGOULEME, 2000-2001. Page 9.

[Kazi Aouel et Rostane, 2007b] : Kazi Aouel Bassim et Rostane Zakaria. Suivre des enseignements du LMD par application de la méthode 2TUP. Faculté des Sciences de l'Ingénieur Département d'Informatique- Université Abou Bekr Belkaid de Tlemcen, novembre 2007. Page14.

[Rabeh, 2005a] : Rabeh. Cours UML-Java. La version UML 1.5. 2005. Page 06.

[Bennama, 2008a] : Miloud Bennama. Translation des diagrammes globaux d'interaction d'UML 2.0 vers les réseaux de Pétri hiérarchiques. Institut National de formation en Informatique(I.N.I). Oued Smar, Alger. Septembre 2008. Page09.

[Bennama, 2008b] : Miloud Bennama. Translation des diagrammes globaux d'interaction d'UML 2.0 vers les réseaux de Pétri hiérarchiques. Institut National de formation en Informatique(I.N.I). Oued Smar, Alger. Septembre 2008. Page14.

[Rabeh, 2005b] : Rabeh. Cours UML-Java. La version UML 1.5. 2005. Page08.

[Kazi Aouel et Rostane, 2007c] : Kazi Aouel Bassim et Rostane Zakaria. Suivre des enseignements du LMD par application de la méthode 2TUP. Faculté des Sciences de l'Ingénieur Département d'Informatique- Université Abou Bekr Belkaid de Tlemcen, novembre 2007. Page19. -

[Kazi Aouel et Rostane, 2007d] : Kazi Aouel Bassim et Rostane Zakaria. Suivre des enseignements du LMD par application de la méthode 2TUP. Faculté des Sciences de l'Ingénieur Département d'Informatique- Université Abou Bekr Belkaid de Tlemcen, novembre 2007. Page20.

[Abdat et Mahdaoui, 2007a] : Nadia Abdat, Latifa Mahdaoui, Pratique des systèmes d'information UML outil du génie logiciel. Copyright Eurl page bleues internationales édition, Septembre 2007. Page48.

[Abdat et Mahdaoui, 2007b] : Nadia Abdat, Latifa Mahdaoui, Pratique des systèmes d'information UML outil du génie logiciel. Copyright Eurl page bleues internationales édition, Septembre 2007. Page67.

[Abdat et Mahdaoui, 2007c] : Nadia Abdat, Latifa Mahdaoui, Pratique des systèmes d'information UML outil du génie logiciel. Copyright Eurl page bleues internationales édition, Septembre 2007. Page55.

[Abdat et Mahdaoui, 2007d] : Nadia Abdat, Latifa Mahdaoui, Pratique des systèmes d'information UML outil du génie logiciel. Copyright Eurl page bleues internationales édition, Septembre 2007. Page32

[Gaby, 2004] Joseph GABAY. Merise et UML pour la modélisation des systèmes d'information, volume 5. Dunod édition, Mars 2004. Page28.

[Latour, 2001] Jean-Pierre LATOUR. Les diagrammes UML; les conséquences du passage à l'OO - Annexe. Division Consultance - Section des Recherches, Février 2001. Page.

[Rosenthal et al, 2001] : Pascal pare camille Rosenthal-Sabroux Nasser Kettani, Dominique Mignet. De Merise à UML. Eryolles édition, France, October 2001. Page39.

[Del, 2001] : Del, Edition bleues internationales et bleues lignes édition et distribution. 2001. page31.

Web1 : <http://www.BADR-banque.com> ,2009.

[WEB2.2005] : <http://www.edition-eyrolles.com> , 2005.

[Web3, 2009] : <http://www.cambiste.com>, 2009.

[web4, 2009] : <http://www.eurocase.net>, 2009.

Annexe

DEMANDE DE CARTE CBRI

C.R.E. Guelma 024

A.L.E. Guelma 821

DATE / / 2011

CONTRAT N°:

PLAFOND DE RETRAIT :

Je soussigné(e) M, Mme, Melle..... sollicite la banque de l'agriculture et du développement rural pour l'octroi d'une carte interbancaire de retrait (CBRI).

Ci-après :

Adresse :
Profession :
Date et lieu de naissance :
Montant revenu mensuel :
N° de compte :

Date et signature du titulaire du compte	Signature du bénéficiaire	Signature du directeur d'agence

Fonctionnement du compte : nombre d'incidents

Néant Incident Plusieurs

Antériorité du compte :

Moins de 03 mois plus de 03 mois

Avis de l'agent chargé de l'opération

Résumé

Dans toutes les banques algériennes le client est confronté à ouvrir plusieurs comptes pour subvenir aux besoins de ses transactions financières (compte ordinaire pour le traitement de salaire, compte placement pour un épargne ou un investissement, compte crédit pour un crédit, etc....). L'objectif du présent mémoire est de faire une transition entre le système en place dans la banque algérienne, basé sur une multitude de comptes pour chaque client, et un système basé sur un compte bancaire unifié et cela en proposons un système automatisé basé sur ce dernier. Pour concrétiser ce travail nous avons effectué un stage au niveau de la banque algérienne de développement rural (BADR), agence de la wilaya de Guelma. Pour la conduite du processus de développement nous avons opté pour le processus unifié 2TUP.

ملخص

يجد الزبون نفسه في جميع البنوك الجزائرية مجبرا على التعامل مع العديد من الحسابات البنكية لتعاملاته المالية (حساب للأجر العادي، حساب للتوفير أو للاستثمار، حساب للحصول على قرض، الخ...). إن الهدف من هذه المذكرة هو الانتقال من النظام الحالي في البنك الجزائري المبني أساسا على التعامل بعدد من الحسابات البنكية لكل زبون، إلى نظام يقوم على حساب بنكي موحد وهذا بعرض نظام آلي مبني على أساس هذا الأخير. ولتحقيق هذا الهدف قمنا بتربص على مستوى البنك الجزائري، للتقنية الريفية (BADR)، وكالة ولاية قالة. وقد اخترنا لتسيير عملية التصميم الطريقة الموحدة 2TUP.