Remerciements

Je tiens à exprimer toute ma reconnaissance à mon encadreur Madame Emna Souissi. Je la remercie de m'avoir encadré, orienté, aidé et conseillé… « G*rand Merci* ».

J'adresse mes sincères remerciements à tous les professeurs, intervenants et toute l'équipe de l'UVT qui ont guidé mes réflexions et ont accepté de m'aider et répondre à mes questions durant mes études en mastère professionnel… *« Merci »*

Je remercie également mon encadreur Monsieur Mounir Chokri, ingénieur principal à la CPG, pour son aimable encadrement et toutes ses consignes et recommandations. J'adresse également des chaleureux remerciements pour toute l'équipe de la division centrale financière de la CPG… « *Merci* ».

Mes sincères remerciements s'adressent à mes parents, mon mari, mes enfants et mes frères pour leur encouragement, leur soutien et leur grand amour… « *Grand Merci* ».

Sommaire

Remerciements 1
Sommaire 2
Liste des figures 5
Liste des tableaux 6
Introduction Générale 7
Chapitre1 : Présentation Générale 9
Introduction 10
I. Présentation de l'organisme d'accueil 10
I.1 Historique 10
I.2 Activités 11
I.2.1 L'extraction 11
I.2.2 La production 11
I.2.3 La commercialisation 11
I.3 L'organigramme de la Direction Centrale Financière 11
II. Présentation du sujet 12
II.1 Le module « Gestion des cautions bancaires » 12
II.2 Le module « Gestion des assurances de paiement des exportations» 13
II.3 Le module « Gestion de facturation des services » 13
II.4 La Gestion des Utilisateurs 13
Conclusion 14
Chapitre2 : Etude Préalable 15
Introduction 16
I. Etude de l'existant 16
I.1 Description de l'existant 16
I.2 Critique de l'existant 16
I.3 Solutions proposées 17
I.4 Solution retenue 17
II. Etude de la solution finale 17
II.1 Spécification des besoins 17
II.1.1 Les besoins fonctionnels 17

II.1.2 Les besoins non fonctionnels 20
II.2 Le langage de modélisation 20
II.3 Les diagrammes de cas d'utilisation 21
II.3.1 Présentation des acteurs 21
II.3.2 Diagramme de cas d'utilisation général 21
II.3.3 Principaux diagrammes de cas d'utilisation 22
II.4. Diagrammes de séquence 26
Conclusion 28
Chapitre3 : Conception Détaillée 29
Introduction 30
I. Modèle de cycle de vie 30
II. Conception architecturale 31
II.1 Conception de l'Architecture Logique 31
III.2 Conception de l'Architecture Physique 32
III.3 Conception de l'Aspect Statique 33
III.3.1 Diagramme de Classes 33
III.3.2 Conception de la Base de Données 35
III.4 Conception Aspect Dynamique 35
III.4.1 Diagramme d'activité 35
III.4.2 Diagramme de Séquence raffiné 38
III. Conception du niveau présentation 40
III.1 Modèle de navigation 40
III.2 Charte graphique 42
Conclusion 43
Chapitre 4 : Réalisation 44
Introduction 45
I. Environnement du travail 45
I.1 Environnement matériel 45
I.2 Environnement logiciel 45
I.2.1 Création de la base de données 45
I.2.2 Outils de développement 46
II. Les composantes applicatives réalisées 47
II.1 Interface de connexion 47
II.2 Menu Gestion des cautions 49
II.2 Menu Gestion des assurances 53

II.3 Menu Gestion de facturation 57
II.4 Menu Gestion des utilisateurs..... 60
Conclusion 61
Conclusion Générale..... 62
Webographie..... 63

Liste des figures

Figure 1: Organigramme de la direction centrale financière 12
Figure 2: Diagramme de cas d'utilisation général 22
Figure 3: Gestion des utilisateurs 23
Figure 4: Gestion des cautions 24
Figure 5: Gestion des fournisseurs 25
Figure 6: Gestion des banques 26
Figure 7: Diagramme de séquence "Gestion des utilisateurs" 27
Figure 8: Diagramme de séquence "Proroger caution" 28
Figure 9: le cycle de vie en V 31
Figure 10: Architecture MVC 32
Figure 11: Architecture 3-tiers 33
Figure 12: Diagramme de classes 34
Figure 13: Diagramme d'activité "gestion des utilisateurs" 36
Figure 14: Diagramme d'activité "gestion des cautions" 37
Figure 15: Diagramme d'activité "Exécution de cautions bancaires" 38
Figure 16: Diagramme de séquence "Ajout caution" 39
Figure 17: Diagramme de séquence "Assurance facture commerciale" 40
Figure 18: Modèle de navigation 41
Figure 19: Installation Oracle Data base 11g Express Edition 45
Figure 20: Création de la base de données 46
Figure 21: connexion accepté 47
Figure 22: Connexion refusée 1er cas 48
Figure 23: Connexion refusée 2éme cas 48
Figure 24: Connexion refusée 3éme cas 49
Figure 25: Menu Gestion des cautions 49
Figure 26: Saisie d'une nouvelle caution 50
Figure 27: Traitement des cautions 51
Figure 28: Lettre de main levée 52
Figure 29: Traitement des cautions 53
Figure 30: Liste des fournisseurs 53
Figure 31: Menu Gestion des assurances 54
Figure 32: Gestion des clients 54
Figure 33: Gestion des taux 55
Figure 34: Gestion des factures 55
Figure 35: Lettre d'assurance 56
Figure 36: Liste des factures d'assurance 56
Figure 37: Gestion de facturation 57
Figure 38: Ajout d'un bénéficiaire 57
Figure 39: Liste des factures 58
Figure 40: Paiement de facture 59
Figure 41: Liste des bénéficiaires 60
Figure 42: Menu administrateur 60
Figure 43: Gestion des utilisateurs 61

Liste des tableaux

Tableau 1: solutions proposées 17
Tableau 2 : Description textuelle du cas "Ajouter Utilisateur" 23
Tableau 3: Description textuelle du cas "Modifier commande" 24
Tableau 4: Description textuelle du cas "Supprimer" 25
Tableau 5: Description textuelle du cas "Consulter banque" 26
Tableau 6: Comparaison entre les types des procédés 30
Tableau 7: Les tables de la base de données 35

Introduction Générale

De nos jours, l''informatique est considérée comme un outil indispensable à toute entreprise qui ne veut pas rester en marge de la mondialisation. En effet, elle est apparue depuis longtemps comme un soutien aux différentes opérations logistiques des entreprises.

Avec la mondialisation technologique et le besoin croissant des sociétés en matière d'information et de capacité logistique, ce soutien a évolué dans le temps et dans l'espace, et même s'est accru grâce à l'amélioration des systèmes informatiques, ainsi qu'à la perpétuelle performance du matériel informatique.

Cela s'est matérialisé notamment par la réduction progressive des coûts et la rapidité du traitement des données. Cette évolution continue de sorte qu'il est impossible, voire impensable aujourd'hui, de gérer une entreprise sérieuse sans l'informatiser.

Avec l'introduction de l'informatique comme moyen de gestion des entreprises, le monde de l'entreprise vient de trouver une solution à ses multiples problèmes liés, notamment, au coût de l'information, à la communication, à l'enregistrement des données et aux différents calculs comptables.

Désormais, les évolutions récentes de l'informatique appellent à reconsidérer la dynamique structurelle et fonctionnelle des entreprises. A cet égard, et dans le contexte actuel de concurrence accrue, le système d'information des entreprises demande à être actualisé en permanence suivant la nature de l'environnement concurrentiel et c'est dans ce contexte que notre travail se situe.

En effet, dans le cadre du stage du projet de fin d'études pour l'obtention du diplôme du mastère professionnel en Nouvelles Techniques des Télécommunications et des Réseaux, la Compagnie des phosphates de Gafsa m'a confié l'étude, la conception et le développement d'une application qui s'intitule « Gestion des cautions bancaires, des assurances des exportations et des facturations de services ». Ce travail était réalisé au sein de la direction centrale financière et plus précisément au sein du service informatique.

Le présent rapport explique nettement les différentes étapes suivies pour réaliser le travail demandé. Nous commençons le rapport par une présentation générale, dans laquelle nous présentons l'organisme d'accueil ainsi que le sujet de l'application.

Dans le second chapitre, nous nous intéressons à faire une étude théorique basée essentiellement sur deux modules à savoir l'étude de l'existant et l'étude de la solution proposée.

Le chapitre suivant est réservé à la conception détaillée. Nous présentons dans ce chapitre les différents aspects conceptuels de l'application.

Nous terminons par le chapitre réalisation au cours duquel nous expliquons le fonctionnement du nouveau système à travers des captures d'écran de quelques interfaces de l'application et cela après avoir présenté l'environnement matériel et logiciel du travail.

En conclusion, nous récapitulons les différentes étapes suivies pour l'élaboration de ce projet et nous signalons ses différents atouts.

Chapitre1 : Présentation Générale

Introduction

Dans ce chapitre, nous décrivons le contexte général du projet. Nous présentons d'abord l'entreprise au sein de laquelle se déroule le stage et nous enchainons par la suite par l'exposé du sujet en expliquant ses différents modules.

I. Présentation de l'organisme d'accueil

La CPG est une entreprise tunisienne d'exploitation de phosphates basée à Gafsa. Elle est rattachée en 1994 au Groupe Chimique Tunisien. Elle figure parmi les plus importants producteurs de phosphates, occupant la cinquième place mondiale avec une production de presque huit millions de tonnes en 2009.

Compagnie des Phosphates de Gafsa

I.1 Historique

C'était en avril 1885, lors d'une prospection dans la région de Metlaoui, partie occidentale du sud du pays, que Philippe THOMAS, géologue amateur français, a découvert des couches puissantes de phosphates de calcium sur le versant Nord de JEBEL THELJA.

D'autres prospections géologiques et des explorations de grande envergure ont suivi cette découverte décisive. Celles-ci ont révélé l'existence d'importants gisements de phosphates au sud et au Nord de l'île de Kasserine.

A partir de 1896, date de création de "la Compagnie des Phosphates et de Chemin de Fer de Gafsa," une nouvelle activité industrielle des phosphates a vu le jour dans le pays. Les premières excavations ont commencé dans la région de Metlaoui et vers 1900, la production des phosphates marchands a atteint un niveau de 200,000 tonnes. En 1996 c'est la fusion des deux structures commerciales de la Compagnie des Phosphates de Gafsa et du Groupe Chimique Tunisien. Avec une expérience centenaire dans l'exploitation et la commercialisation des phosphates tunisiens, la CPG figure parmi les plus gros producteurs de phosphates dans le monde.

I.2 Activités

Le phosphate est la principale richesse du gouvernorat de Gafsa. Cette substance est exploitée par la Compagnie des Phosphates de Gafsa qui joue un rôle important dans le développement de son environnement. Les activités de la CPG sont : l'extraction la production et la commercialisation du phosphate.

I.2.1 L'extraction

Les méthodes d'extraction étaient au début classiques, par la suite la société a fait recours de plus en plus aux méthodes mécanisées et beaucoup plus évoluées. Il existe en fait neufs carrières d'extraction. L'extraction est assurée selon la recette suivante :

- Criblé : la séparation de gros et fines de tout venant.
- Trié : la séparation manuelle du refus du criblé.
- Broyé : la réduction granulométrique des blocs du minerai.

Etant précisé que le phosphate (BTS : brut, trié, séché) de la compagnie est extrait à partir de neufs carrières d'extraction.

I.2.2 La production

La compagnie produit deux qualités principales de phosphates naturels marchands :

- Qualité 65-68% filtrée et séchée : utilisée pour la production des engrais chimiques.
- Qualité 60-62% pour l'application directe.

A la fin de ce processus de production, le phosphate est soit stocké dans les cuves des usines, soit chargé dans les wagons pour être expédié vers les locaux et à l'étranger.

I.2.3 La commercialisation

Cette opération se réalise à deux niveaux :

- Local : le phosphate est commercialisé aux usines d'engrais chimiques de gabes et de Sfax soit 80% de la totalité de la production.
- Etranger : la C.P.G exporte sa production en Europe, en Asie et en Amérique latine.

I.3 L'organigramme de la Direction Centrale Financière

Vu la complexité de l'organigramme de la CPG, nous allons nous contenter de présenter ci-dessous l'organigramme de la direction centrale financière de la CPG au sein de laquelle se déroule ce stage de fin d'étude :

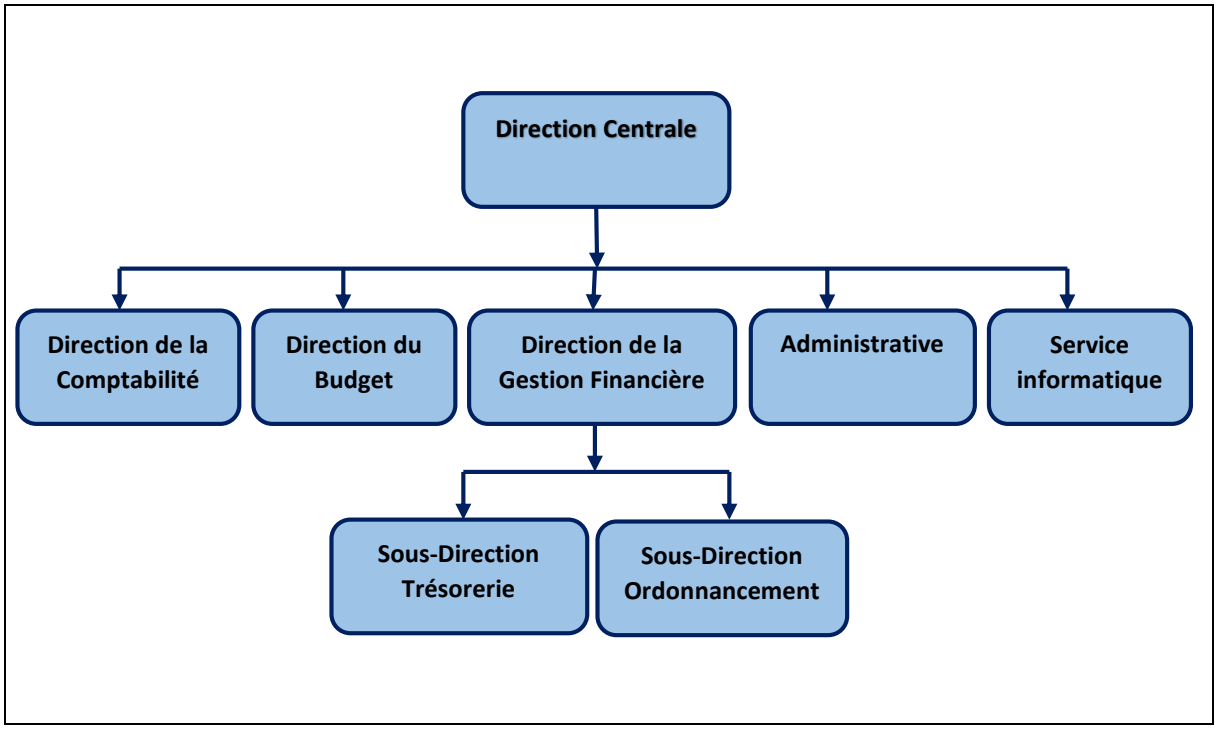


Figure 1: Organigramme de la direction centrale financière

II. Présentation du sujet

Comme nous l'avons déjà mentionné, le sujet de notre application est la conception et le développement d'une application pour la gestion des cautions bancaires, des assurances des exportations et des facturations de services au sein de la CPG.

Les deux principaux objectifs de ce travail sont l'amélioration de l'application existante pour la gestion des cautions bancaires ainsi que le développement de deux nouveaux modules pour la gestion des assurances des exportations et des facturations de services.

II.1 Le module « Gestion des cautions bancaires »

Pour chaque commande passée avec un fournisseur, la CPG exige une caution de garantie selon un taux qui sera fixé dans la commande elle-même. A l'échéance de la caution, et selon le sort du marché avec le fournisseur, le responsable des cautions procède soit à la libération (main levée), soit à la prorogation vers une autre date soit à l'exécution (versement du montant au profil de la CPG) de la dite caution, il y'aura par conséquence plusieurs échanges avec les banques et les fournisseurs. Ce module doit assurer les fonctions suivantes :

- La gestion des banques
- La gestion des fournisseurs
- La gestion des cautions bancaires
- La gestion des commandes
- La gestion des devises.

II.2 Le module « Gestion des assurances de paiement des exportations»

Chaque année et pour chaque client à l'étranger, la CPG ouvre une assurance chez la Compagnie Tunisienne Pour l'Assurance du Commerce Extérieur, «Cotunace». Cette assurance va couvrir le risque du non-paiement de la facture commerciale de la part du client étranger, en contrepartie, la Cotunace reçoit une prime d'assurance selon un taux bien déterminé et sera fixé annuellement. Ce module doit assurer les fonctions suivantes :

- La gestion des clients
- La gestion des factures de vente
- La gestion des taux d'assurance.

II.3 Le module « Gestion de facturation des services »

La CPG offre des services aux différents bénéficiaires tels que : La location d'engin (camion, bus, etc.), La location des bâtiments (maisons, dépôts, etc.), La vente des ferrailles métalliques, des roues, des bois, etc. Le système doit assurer la gestion des factures et leurs paiements. Les principales fonctionnalités assurées par ce module sont :

- La gestion des bénéficiaires
- La gestion des services
- La gestion des facturations
- La gestion des paiements

II.4 La Gestion des Utilisateurs

En fonction du public cible, les utilisateurs d'une application informatique peuvent varier. Les utilisateurs de notre application sont :

- L'administrateur,
- Le responsable des cautions
- Le responsable des assurances des exportations

- Le responsable des facturations de services.

Conclusion

Dans ce chapitre, nous avons essayé d'expliquer le contexte du projet et dévoiler l'objectif du travail demandé. Avant de développer toute application informatique, il est absolument nécessaire d'étudier le système existant, de dégager ses fonctionnalités, et surtout d'en connaître les failles afin de trouver les solutions adéquates pour le travail demandé. Cette étude sera le sujet du chapitre suivant.

Chapitre2 : Etude Préalable

Introduction

L'étude préalable constitue une étape préliminaire pour la réalisation d'une application. En effet, elle permet d'analyser, d'évaluer et de critiquer le fonctionnement habituel, tout en élaborant la liste des solutions possibles afin d'en retenir une comme solution finale. Nous commençons ce chapitre par une étude de l'existant. Ensuite, nous réservons le reste du chapitre à l'étude de la solution retenue.

I. Etude de l'existant

I.1 Description de l'existant

La CPG utilise actuellement une ancienne application locale qui traite seulement le module de gestion des cautions bancaires, et qui prend en charge les fonctionnalités suivantes :

- Dès réception, l'utilisateur saisie les données relatives à la caution bancaire telles que : numéro, date, banque, fournisseur, etc.
- A l'échéance, l'utilisateur envoie des lettres de prorogation, de main levée ou d'exécution à la banque concernée.
- En cas de prorogation de caution, l'utilisateur saisie la nouvelle date et met à jour les données des cautions.

I.2 Critique de l'existant

L'application actuelle de gestion des cautions bancaires souffre d'un ensemble d'inconvénients dont nous citons :

- Les lettres de prorogation, de main levée ou d'exécution à envoyer aux banques sont réalisées manuellement avec Microsoft Word.
- La sauvegarde des données se fait sous forme de fichiers.
- Manque d'organisation des champs dans les interfaces.
- Multiplicité de bugs.
- Absence de gestion des utilisateurs et droits d'accès.
- Système non sécurisé.

En ce qui concerne les deux modules : gestion des assurances de paiement des exportations et gestion des facturations de services, nous avons constaté que tous les traitements sont réalisés manuellement. Donc, il n'existe pas une application dédiée à ces deux modules.

I.3 Solutions proposées

Afin de résoudre les problèmes sus indiqués, plusieurs solutions peuvent être mise en place. Parmi ces solutions nous citons :

Solution proposée	Critiques
○ Le téléchargement et l'utilisation d'une application libre à partir du web.	➔ Nous ne pouvons ni la maintenir ni la modifier à moins qu'elle soit open source.
○ L'achat d'une application prête pour la gestion de ces trois modules.	➔ Le coût de l'achat sera très élevé.
○ Le développement d'une nouvelle application qui respecte la procédure déjà établie.	➔ Cette solution nécessite un développement sur mesure.

Tableau 1: solutions proposées

I.4 Solution retenue

Parmi les solutions sus indiquées, nous avons opté pour le développement d'une nouvelle application pour la gestion des cautions bancaires, assurances de paiement des exportations et facturation des services qui s'intitule « **GCAF** ». Cette application doit répondre à tous les besoins qui seront détaillés dans la partie suivante de ce chapitre.

II. Etude de la solution finale

II.1 Spécification des besoins

L'étape de spécification des besoins consiste à définir l'ensemble des fonctionnalités que la GCAF doit fournir. Ces besoins seront répartis en deux groupes fonctionnels et non fonctionnels.

II.1.1 Les besoins fonctionnels

Les besoins fonctionnels expriment une action que doit effectuer le système en réponse à une demande (sorties qui sont produites pour un ensemble donné d'entrées). Notre application est répartie sur trois modules. Chaque module possède ses propres besoins fonctionnels. Pour cela, nous allons détailler les fonctionnalités de chaque module à part.

- **Module gestion des cautions bancaires**
  - Gestion des commandes :
    - Ajout d'une nouvelle commande.
    - Modification des données relatives à une commande.
    - Suppression d'une commande.
    - Consultation des commandes.
  - Gestion des fournisseurs :
    - Ajout d'un fournisseur.
    - Modifier un fournisseur.
    - Supprimer un fournisseur.
    - Consultation des fournisseurs.
  - Gestion des banques :
    - Ajout d'une nouvelle banque.
    - Modification des données relatives à une banque.
    - Suppression d'une banque.
    - Consultation des banques.
  - Gestion des cautions bancaires :
    - Ajout d'une caution.
    - Modification d'une caution.
    - Suppression d'une caution.
    - Consultation des cautions.
  - Gestion des devises :
    - Ajout d'une devise.
    - Modification des devises.
    - Suppression d'une devise.
    - Consultation des devises.

- **Module assurances de paiement des exportations**
  - Gestion des factures :
    - Ajout d'une nouvelle facture commerciale.
    - Modification des données relatives à une facture.
    - Suppression d'une facture.
    - Consultation des factures.

- Gestion des clients :
  - Ajout d'un nouveau client.
  - Modification de données relatives à un client.
  - Suppression d'un client.
  - Consultation des clients.
- Gestion des taux d'assurance :
  - Ajout d'un nouveau taux d'assurance à appliquer par année.
  - Modification de données relatives à un taux.
  - Suppression d'un taux d'assurance.
  - Consultation des taux.

❖ **Module facturation des services**

- Gestion des services :
  - Ajout d'un nouveau service.
  - Modification des données relatives à un service.
  - Suppression d'un service.
  - Consultation des services.
- Gestion des bénéficiaires :
  - Ajout d'un bénéficiaire.
  - Modification des données relatives à un bénéficiaire.
  - Suppression d'un bénéficiaire.
  - Consultation des bénéficiaires.
- Gestion des factures de services :
  - Ajout d'une nouvelle facture de service.
  - Modification d'une facture.
  - Suppression d'une facture.
  - Consultation des factures.

❖ **Gestion des utilisateurs**

- Ajout d'un utilisateur (utilisateur caution, utilisateur assurance, etc.).
- Modification des données de l'utilisateur.
- Suppression d'un utilisateur.
- Consultation des utilisateurs.

II.1.2 Les besoins non fonctionnels

Un besoin non-fonctionnel est une condition qui spécifie les critères qui peuvent être utilisés pour juger du fonctionnement d'un système, plutôt que des comportements de celui-ci. Cela devrait être mis en contraste avec les exigences fonctionnelles qui définissent le comportement spécifique ou les fonctions désirés.

Même si ces besoins n'étant pas décisifs au fonctionnement du système, ils représentent un bon signe de la nature du système :

- Autonomie du système : le système s'exécute et fonctionne entièrement sans avoir recours à d'autres applications
- Convivialité : l'interface utilisateur doit être conviviale pour assurer un accès aisé aux données.
- Ergonomie : l'application doit être ergonomique et facile à utiliser.
- Navigabilité : elle doit être aussi navigable en donnant la possibilité à l'utilisateur d'accéder aux différentes rubriques à partir de n'importe qu'elle page grâce à la présence d'un menu.
- Gestion des erreurs : Les erreurs doivent être signalées par des messages d'erreur explicites.

II.2 Le langage de modélisation

Un langage de modélisation est un langage artificiel qui peut être utilisé pour exprimer de l'information ou de la connaissance ou des systèmes dans une structure qui est définie par un ensemble cohérent de règles. Un langage de modélisation peut être graphique ou textuel.

- Les langages de modélisation graphiques utilisent des techniques de diagrammes avec des symboles associés à des noms qui représentent les concepts et des lignes qui connectent les symboles et qui représentent les relations et les diverses autres annotations graphiques pour représenter les contraintes.
- Les langages de modélisation textuels utilisent typiquement des mots-clés standardisés accompagnés de paramètres pour rendre les expressions interprétables par les ordinateurs.

Il existe plusieurs langages de modélisation dont nous citons : Sysml, Bisiness Process Modeling Notation, Energy Systems Language, IDEF, Unified Modeling Language et ses antécédents (OMT, Booch, OOSE), etc. Parmi tous ces langages nous avons choisi de modéliser notre application avec l'UML. [1]

II.3 Les diagrammes de cas d'utilisation

Un cas d'utilisation définit le comportement d'un système ou la sémantique de toute autre entité en spécifiant une séquence d'actions, avec des variantes, que l'activité réalise en interagissant avec les acteurs de l'entité.

Un diagramme de cas d'utilisation est un graphe d'acteurs, un ensemble de cas d'utilisation englobés par la limite du système, des relations de communication entre les acteurs et les cas d'utilisation, et des généralisations de ces cas d'utilisation. [1]

II.3.1 Présentation des acteurs

Un acteur est une abstraction extérieure au système à modéliser (l'organisation lorsque l'on modélise le métier, le système informatique dans un second temps) qui interagit avec lui. Il s'agit de rôles joués par des personnes, de logiciels, de matériels, etc. On détermine un acteur principal en se demandant « À qui est destiné le système à modéliser ? ». [1]

Les acteurs en interaction avec notre système sont :

- L'administrateur : son rôle consiste à gérer les utilisateurs de l'application.
- Le responsable des cautions : son rôle consiste à gérer le module des cautions bancaires.
- Le responsable des assurances : son rôle consiste à gérer le module assurance des exportations.
- Le responsable des facturations : son rôle consiste à gérer le module de facturation des services.

II.3.2 Diagramme de cas d'utilisation général

Nous présentons, dans la figure N°2, le diagramme de cas d'utilisation global de notre application afin de mieux comprendre son fonctionnement :

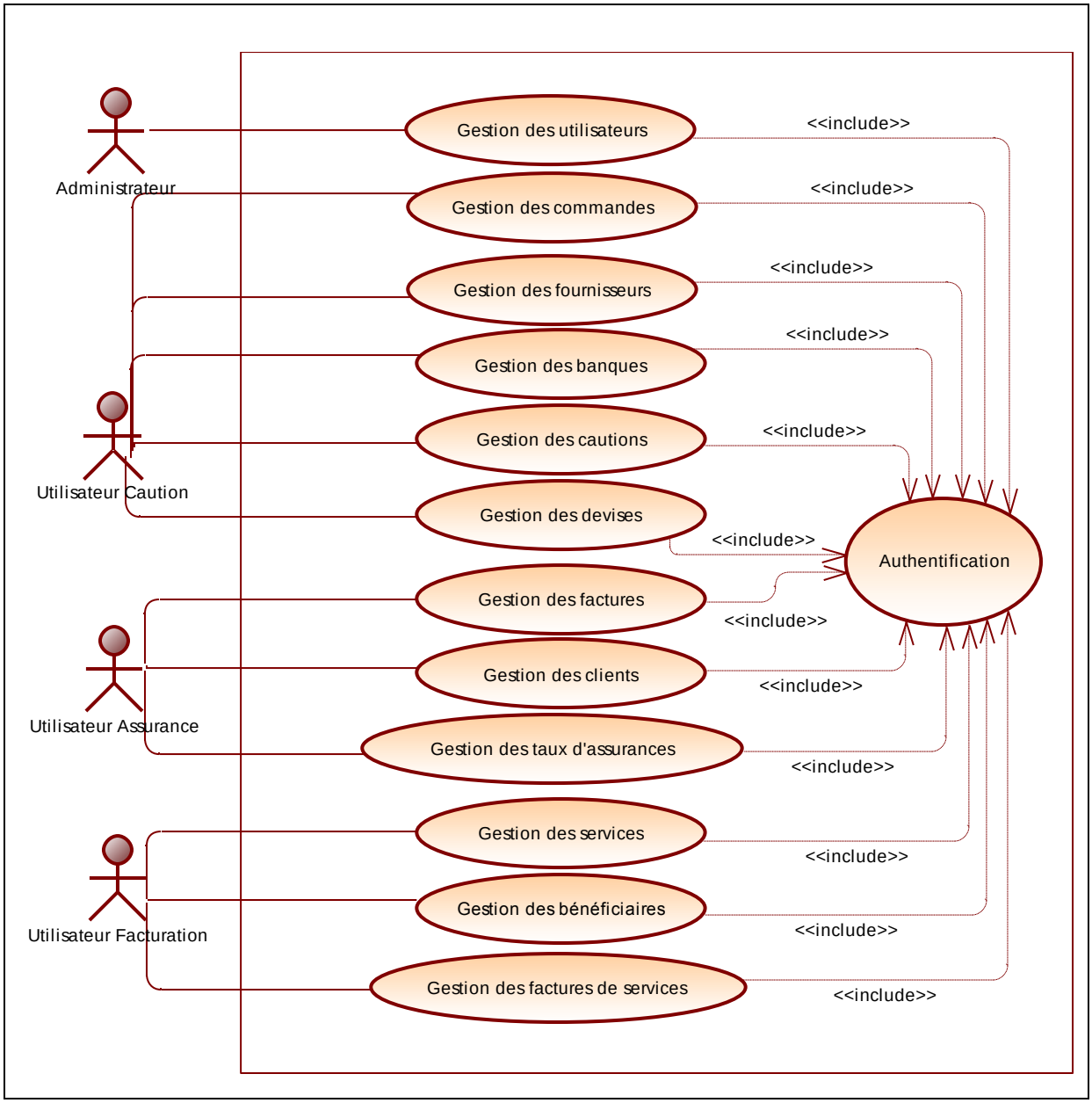


Figure 2: Diagramme de cas d'utilisation général

Dans le paragraphe suivant, nous allons présenter quelques diagrammes de cas d'utilisation ainsi que leurs descriptions textuelles. Les autres diagrammes sont pareils à ceux que nous allons présenter dans ce qui suit.

II.3.3 Principaux diagrammes de cas d'utilisation

- **Diagramme de cas d'utilisation : gestion des utilisateurs**

Comme l'indique la figure N°3, dès que l'administrateur accède à l'application, il va consulter la liste des utilisateurs, ensuite il peut soit ajouter, soit modifier ou bien supprimer un utilisateur :

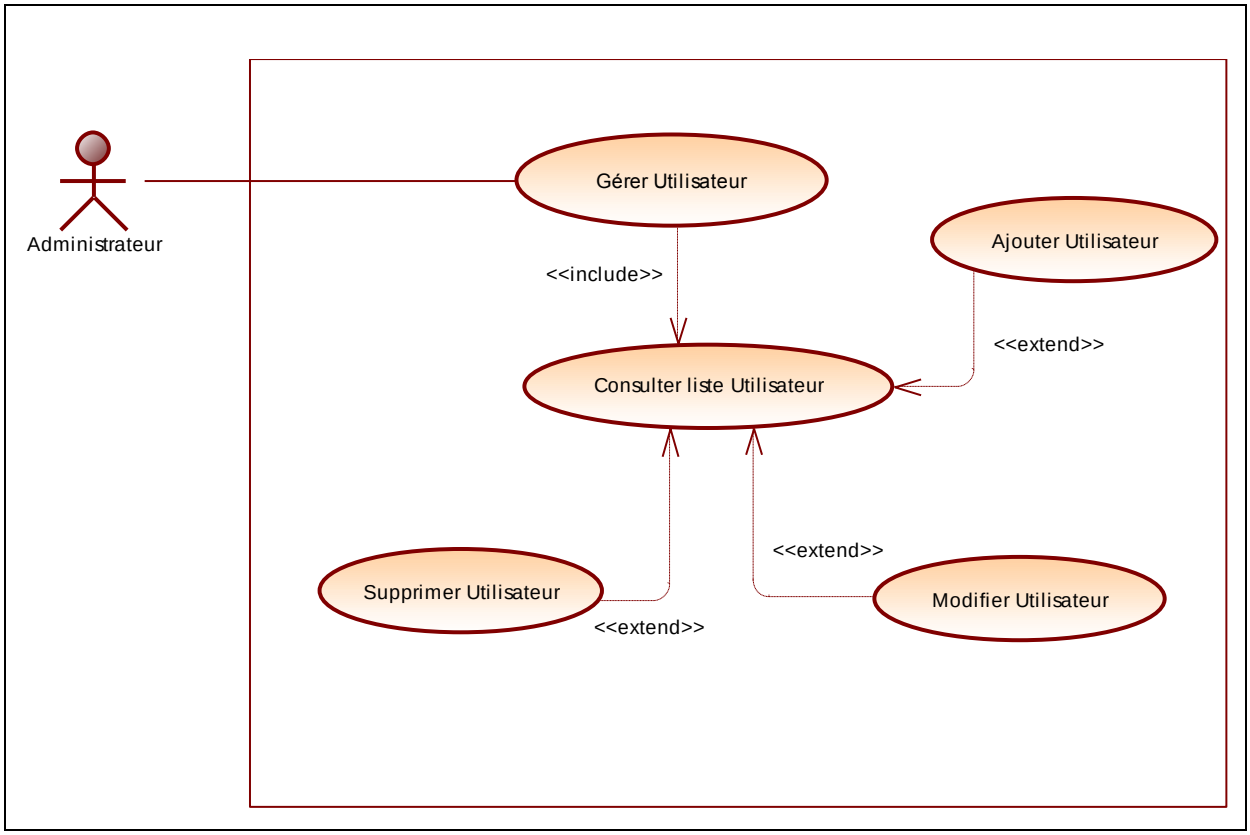


Figure 3: Gestion des utilisateurs

Description textuelle du cas d'utilisation « Ajouter Utilisateur »

Cas D'utilisation	Ajouter Utilisateur
But	Ajouter un nouvel utilisateur
Acteur	Administrateur du système
Pré conditions	L'administrateur du système s'est authentifié avec succès
Scénario nominal	1. L'administrateur demande d'ajouter un nouvel utilisateur. 2. Le système affiche un formulaire d'informations. 3. L'administrateur remplie le formulaire et valide les données saisies. 4. Le système ajoute le nouvel utilisateur à la base de données et affiche un message de confirmation.
Extension	Le système ne permet pas l'ajout du nouvel utilisateur (utilisateur déjà existant).

Tableau 2 : Description textuelle du cas "Ajouter Utilisateur"

Diagramme de cas d'utilisation : gestion des commandes

La figure N°4 représente le diagramme de cas d'utilisation «gestion des commandes ». Cette gestion est uniquement faite par l'utilisateur caution :

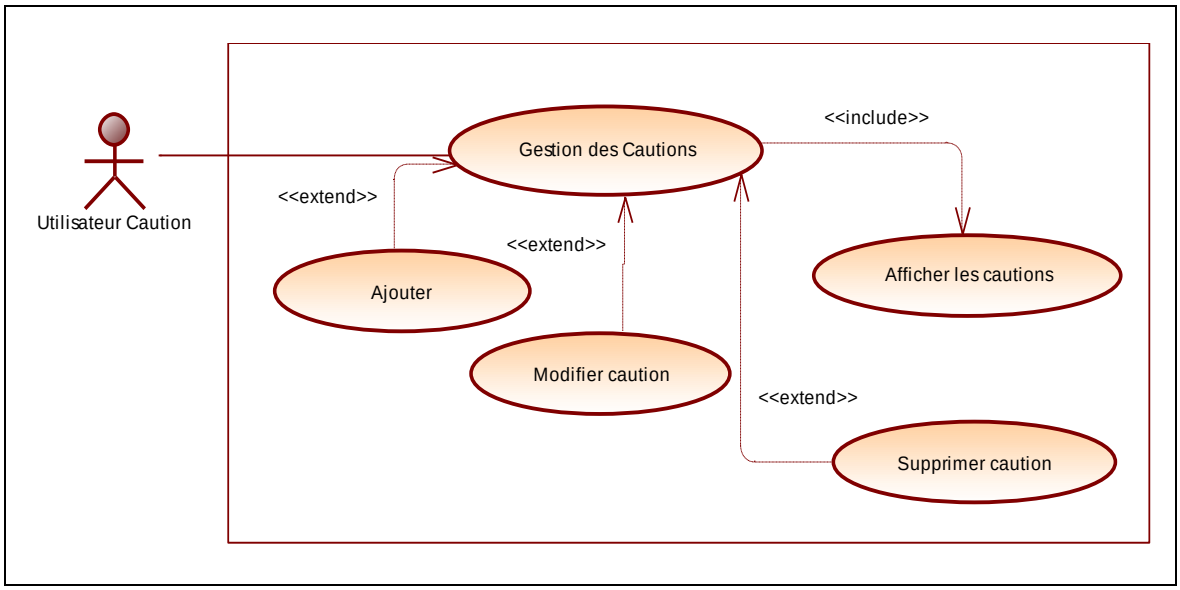


Figure 4: Gestion des cautions

Description textuelle du cas d'utilisation « Modifier commande»

Cas D'utilisation	Modifier commande
But	Modifier les données relatives à une commande déjà existante
Acteur	Responsable des cautions
Pré conditions	- L'utilisateur s'est authentifié avec succès - La commande existe dans la base de données
Scénario nominal	1. Le responsable demande de modifier les données d'une commande. 2. Le système affiche un formulaire comportant les données déjà enregistrées. 3. L'utilisateur modifie le formulaire et valide les nouvelles données. 4. Le système enregistre les données et affiche un message indiquant que la modification s'est fait avec succès.
Extension	- L'utilisateur annule la modification. - Le système ne permet pas la modification (champs obligatoire vide).

Tableau 3: Description textuelle du cas "Modifier commande"

Diagramme de cas d'utilisation : gestion des fournisseurs

La gestion des fournisseurs ainsi que toutes ses tâches sont illustrées par le diagramme de cas d'utilisation suivant :

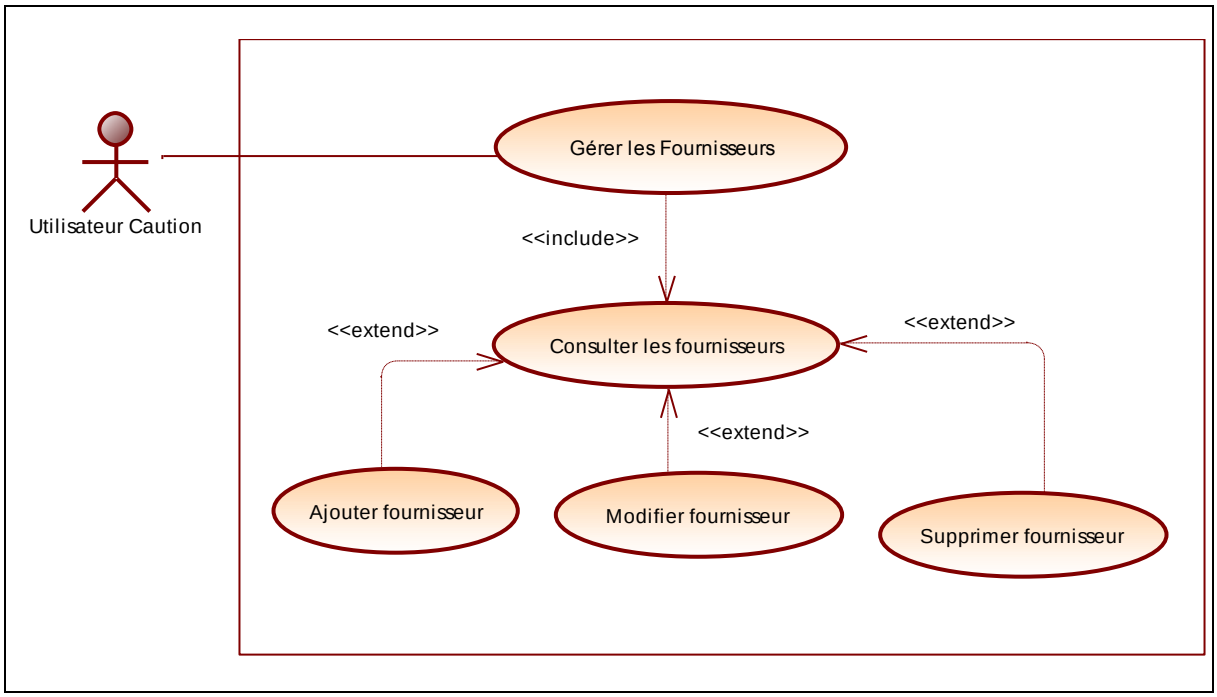


Figure 5: Gestion des fournisseurs

Description textuelle du cas d'utilisation « Supprimer fournisseur»

Cas D'utilisation	Supprimer fournisseur
But	Supprimer un fournisseur de la base de données.
Acteur	Responsable des cautions.
Pré conditions	- L'utilisateur s'est authentifié avec succès. - Le fournisseur existe dans la base de données.
Scénario nominal	1. Le responsable demande de supprimer un fournisseur de la base de données. 2. Le système affiche un message pour demander la confirmation de la demande de suppression. 3. L'utilisateur confirme la demande de suppression du fournisseur. 4. Le système supprime le fournisseur et renvoie un message indiquant que la suppression s'est fait avec succès.
Extension	L'utilisateur décide d'annuler la suppression.

Tableau 4: Description textuelle du cas "Supprimer"

Diagramme de cas d'utilisation : gestion des banques

D'après la figure N°6, L'utilisateur caution peut gérer les banques après avoir consulter leur liste. En effet, il peut ajouter, modifier ou supprimer une banque :

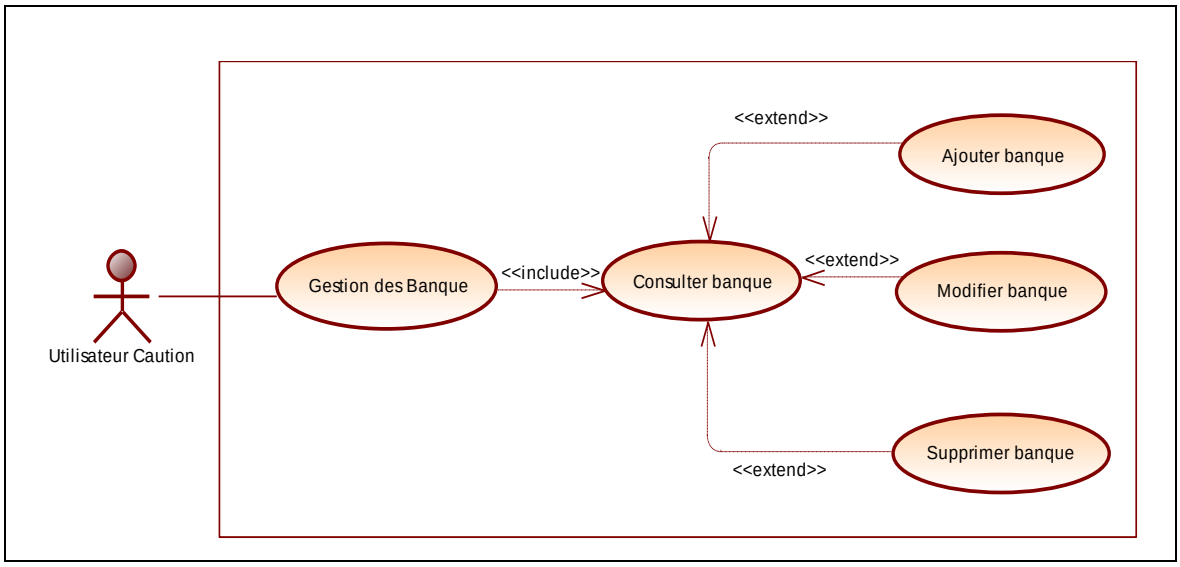


Figure 6: Gestion des banques

Description textuelle du cas d'utilisation « Consulter banque»

Cas D'utilisation	Consulter banque.
But	Consulter les données relatives à une banque.
Acteur	Responsable des cautions
Pré conditions	- L'utilisateur s'est authentifié avec succès - La banque existe dans la base de données
Scénario nominal	1. Le responsable demande de consulter une banque. 2. Le système affiche un formulaire comportant les données relatives à la banque.

Tableau 5: Description textuelle du cas "Consulter banque"

II.4. Diagrammes de séquence

Le diagramme de séquence met en valeur les échanges de messages entre acteurs et système de manière chronologique, Il modélise donc à la fois des activités (fonctionnalités) et des communications (interactions) pour les entités concernées.

Dans ce qui suit, nous allons présenter quelques scénarios de notre application, nous choisissons les scénarios relatifs à :

- gestion des utilisateurs.
- prorogation des cautions.

Diagramme de séquence : Gestion des utilisateurs

L'administrateur est le responsable de la création, modification et suppression des utilisateurs qui seront autorisés à utiliser l'application selon des droits déjà définis.

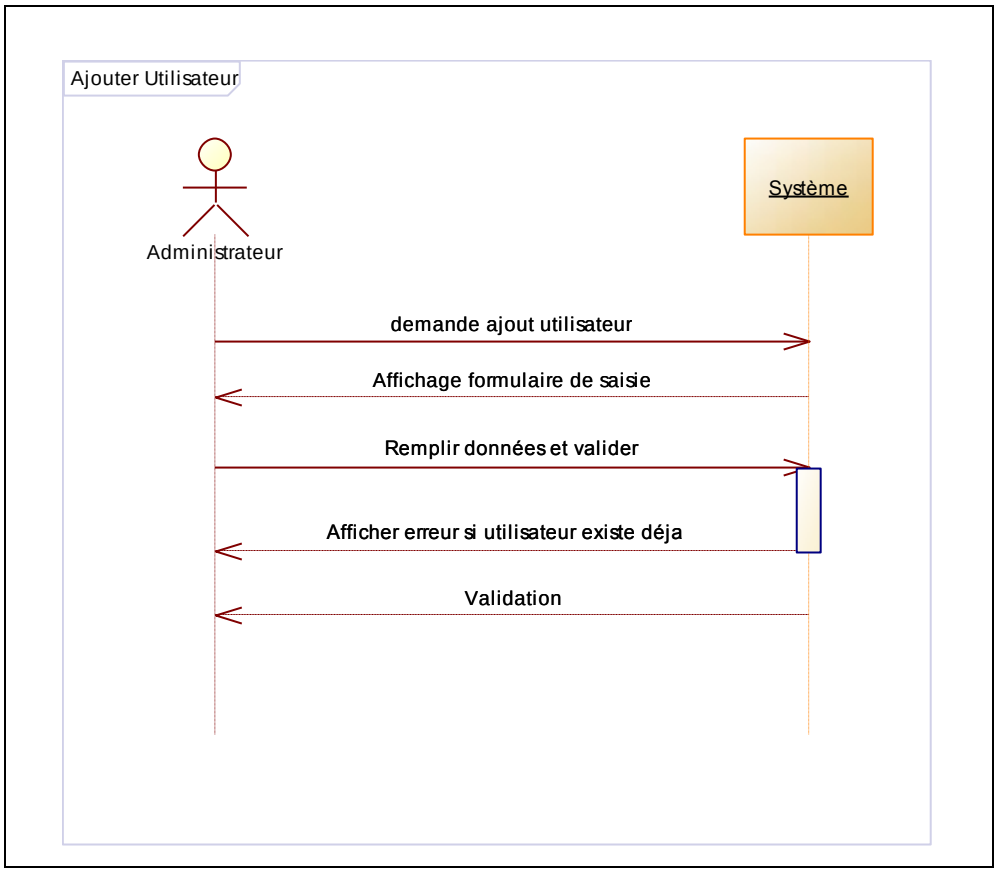


Figure 7: Diagramme de séquence "Gestion des utilisateurs"

Diagramme de séquence : Prorogation des cautions

Avant l'échéance des cautions et dans le cas où la commande n'est pas encore clôturée, le responsable caution lance une demande de prorogation auprès de la banque concernée pour prolonger la durée de validité de la dite caution. Après réception de l'accord de prorogation, l'utilisateur intervient et saisie la nouvelle date d'échéance dans l'application.

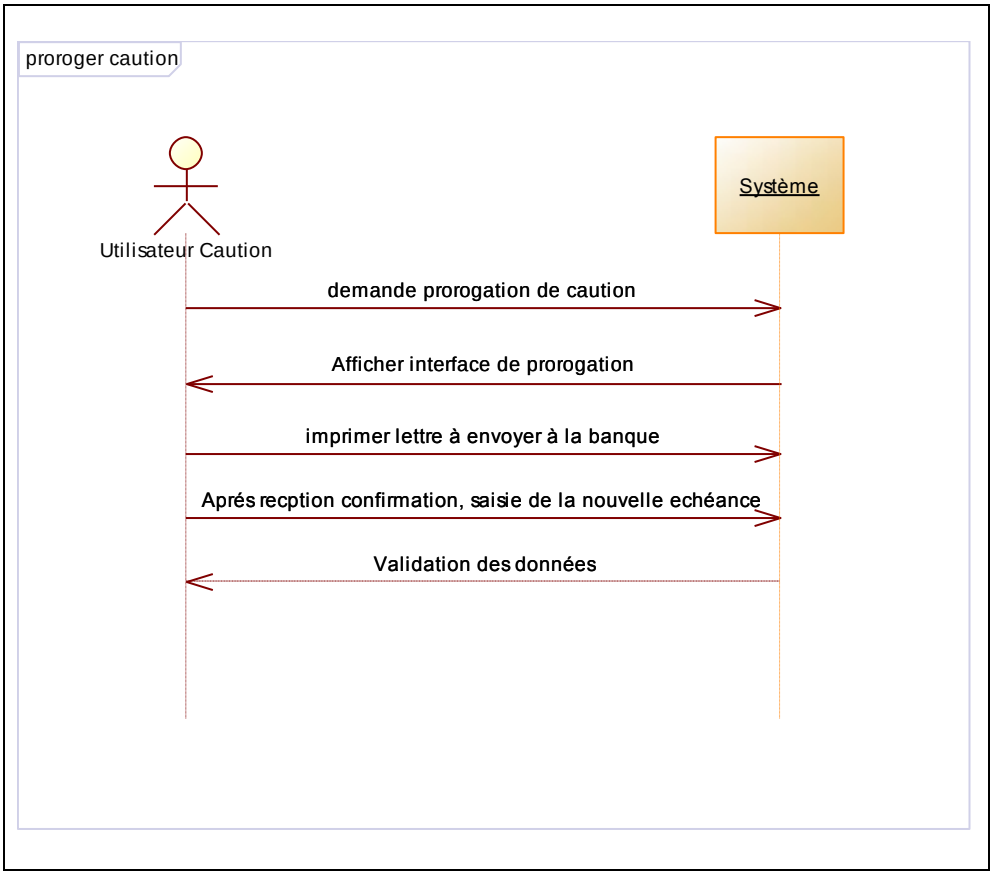


Figure 8: Diagramme de séquence "Proroger caution"

Conclusion

La phase de spécification des besoins nous a permis de déterminer les principales fonctionnalités de notre système à travers la présentation des besoins fonctionnels et non fonctionnels. Nous avons réussi à élaborer les diagrammes de cas d'utilisation relatifs aux besoins spécifiés et nous avons présenté quelques diagrammes de séquence afin de mieux comprendre le fonctionnement du futur système. Tout ce travail nous a permis de passer à la phase la plus importante dans notre travail : la conception.

Chapitre3 : Conception Détaillée

Introduction

Après l'étape de l'étude préalable et la spécification des besoins, nous réservons cette partie à la conception détaillée de notre système. En effet, au cours de ce chapitre nous allons décrire les différentes étapes de conception de la partie statique (des données), ainsi que celles de la partie dynamique (des traitements).

I. Modèle de cycle de vie

Le cycle de vie d'un logiciel est un ensemble séquentiel de phases, dont le nom et le nombre sont déterminés en fonction des besoins de projet, permettant généralement le développement d'un produit ou d'un service. [2]

Le modèle de cycle de vie est la description d'un processus couvrant les phases de : Création d'un produit, sa distribution sur un marché, sa disparition [3]. Le procédé logiciel est une application d'un modèle pour un projet spécifique, qui peut inclure une certaine adaptation [4]. Il existe deux types de modèles de procédé :

	Caractéristiques
Méthodes Classiques	-Modèles stricts. -Etapes très clairement définies. -Documentation très fournie. -Fonctionne bien avec les gros projets et les projets gouvernementaux.
Méthodologies Agiles	-Modèles incrémentaux et itératifs. -Petites et fréquentes livraisons. -Accent sur le code et moins sur la documentation. -Convient aux projets de petites et moyennes tailles.

Tableau 6: Comparaison entre les types des procédés

Dans notre application, vu que tous les documents nécessaires nous ont été fournis et que nous avons dès le début élaboré un plan bien déterminé pour notre projet, nous avons choisi d'utiliser une méthode classique à savoir : le cycle en V.

Ce modèle peut être expliqué par la figure N°9. En effet, chaque étape d'une branche a son correspondant dans l'autre branche, c'est à dire qu'une étape de conception correspond à une étape de test qui lui est spécifique. A tel point, d'ailleurs, qu'une étape de test peut être élaborée dès que la phase de conception correspondante est terminée, indépendamment du reste du projet.

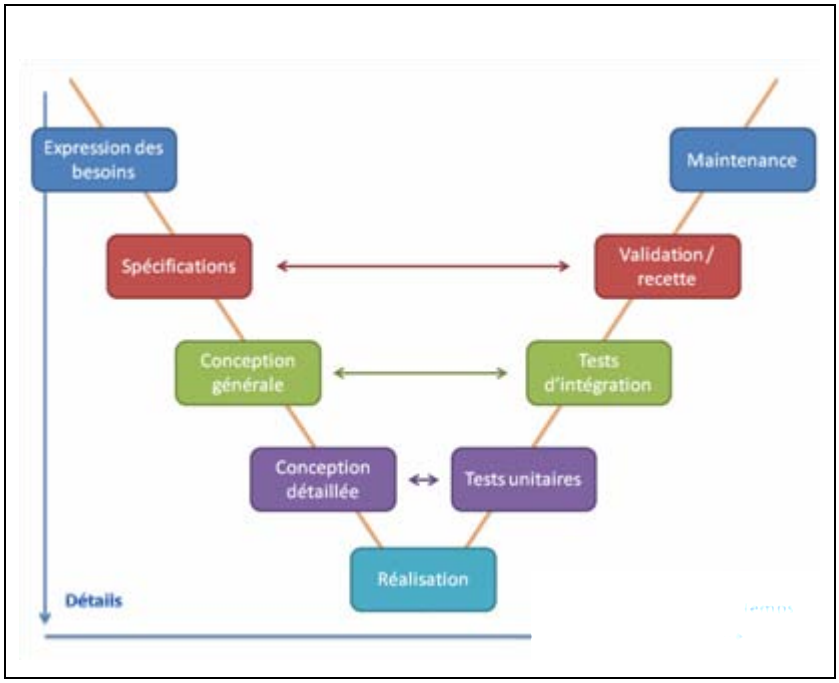


Figure 9: le cycle de vie en V [5]

II. Conception architecturale

Une architecture est l'organisation fondamentale d'un système, constituée de ses composants, des relations entre ces composants et avec l'environnement et des principes guidant sa conception et son évolution [6]. Dans ce qui suit nous allons présenter l'architecture logique ainsi que celle physique que nous avons mise en place pour notre application.

II.1 Conception de l'Architecture Logique

Dans notre application nous avons choisi le modèle MVC (Modèle-Vue-Contrôleur) comme architecture logique. Ce modèle sert à séparer la couche interface utilisateur des autres parties du système. Il est constitué de trois types de composants :

- ✓ Modèle : rassemble des données du domaine, des connaissances du système. Contient les classes dont les instances doivent être vues et manipulées.
- ✓ Vue : utilisé pour présenter/afficher les données du modèle dans l'interface utilisateur.
- ✓ Contrôleur : contient les fonctionnalités nécessaires pour gérer et contrôler les interactions de l'utilisateur avec la vue et le modèle. [7]

L'architecture MVC nous permet les avantages suivants :

- ✓ Elle est appropriée pour les systèmes interactifs, particulièrement ceux impliquant plusieurs vues du même modèle de données.

- ✓ Elle peut être utilisée pour faciliter la maintenance de la cohérence entre les données distribuées.
- ✓ Elle offre un gain de temps de maintenance et d'évolution de l'application.

La figure N°10 nous démontre le principe de cette architecture :

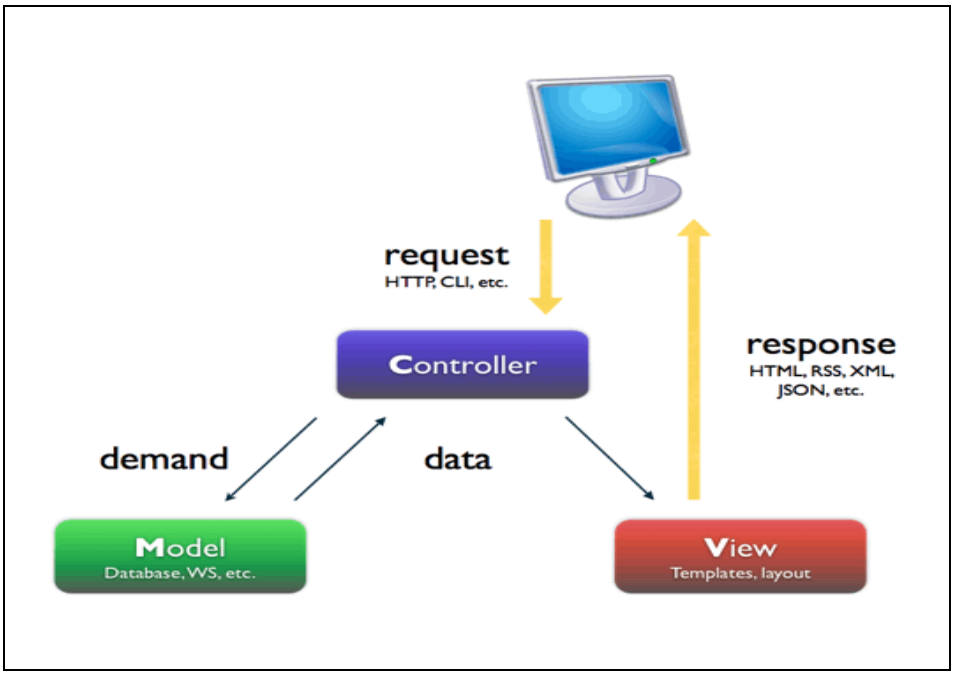


Figure 10: Architecture MVC

III.2 Conception de l'Architecture Physique

L'architecture 3-tiers est le modèle que nous avons choisi pour élaborer notre système. Dans cette architecture, il faut distinguer trois couches/éléments :

- ✓ La couche présentation (affichage) associée au client qui de fait est dit "léger" dans la mesure où il n'assume aucune fonction de traitement à la différence du modèle 2-tiers.
- ✓ La couche fonctionnelle liée au serveur, qui dans de nombreux cas est un serveur Web muni d'extensions applicatives.
- ✓ La couche de données liée au serveur de base de données (SGBD).

Nous avons choisi cette architecture pour plusieurs raisons dont nous citons principalement :

- ✓ une plus grande flexibilité/souplesse
- ✓ une plus grande sécurité (la sécurité peut être définie pour chaque service)
- ✓ de meilleures performances (les tâches sont partagées) [8]

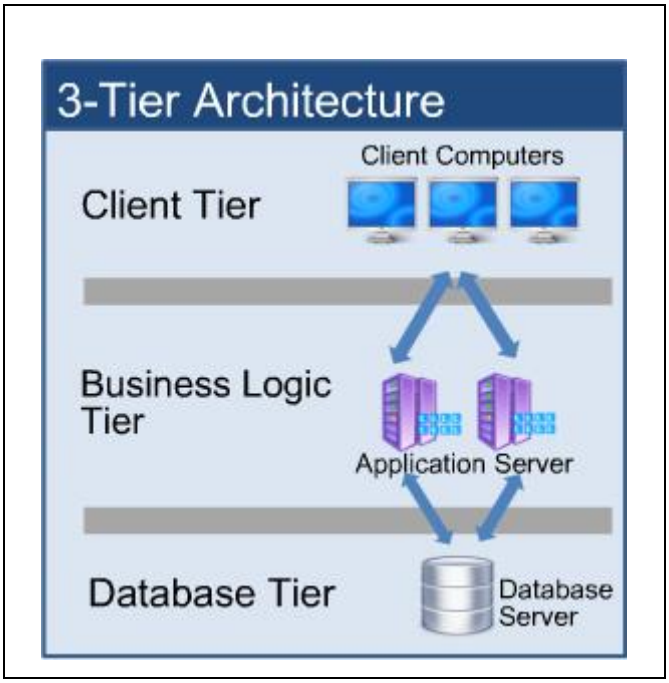


Figure 11: Architecture 3-tiers [9]

III.3 Conception de l'Aspect Statique

III.3.1 Diagramme de Classes

La classe est un concept abstrait qui permet de représenter toutes les entités d'un système. Elle est définie par son nom, ses attributs et ses opérations. Le diagramme de classes représente les classes intervenant dans le système. C'est une représentation statique des éléments qui composent un système et de leurs relations. [10]

Nous présentons, ci-dessous, le diagramme de classe relatif à notre application :

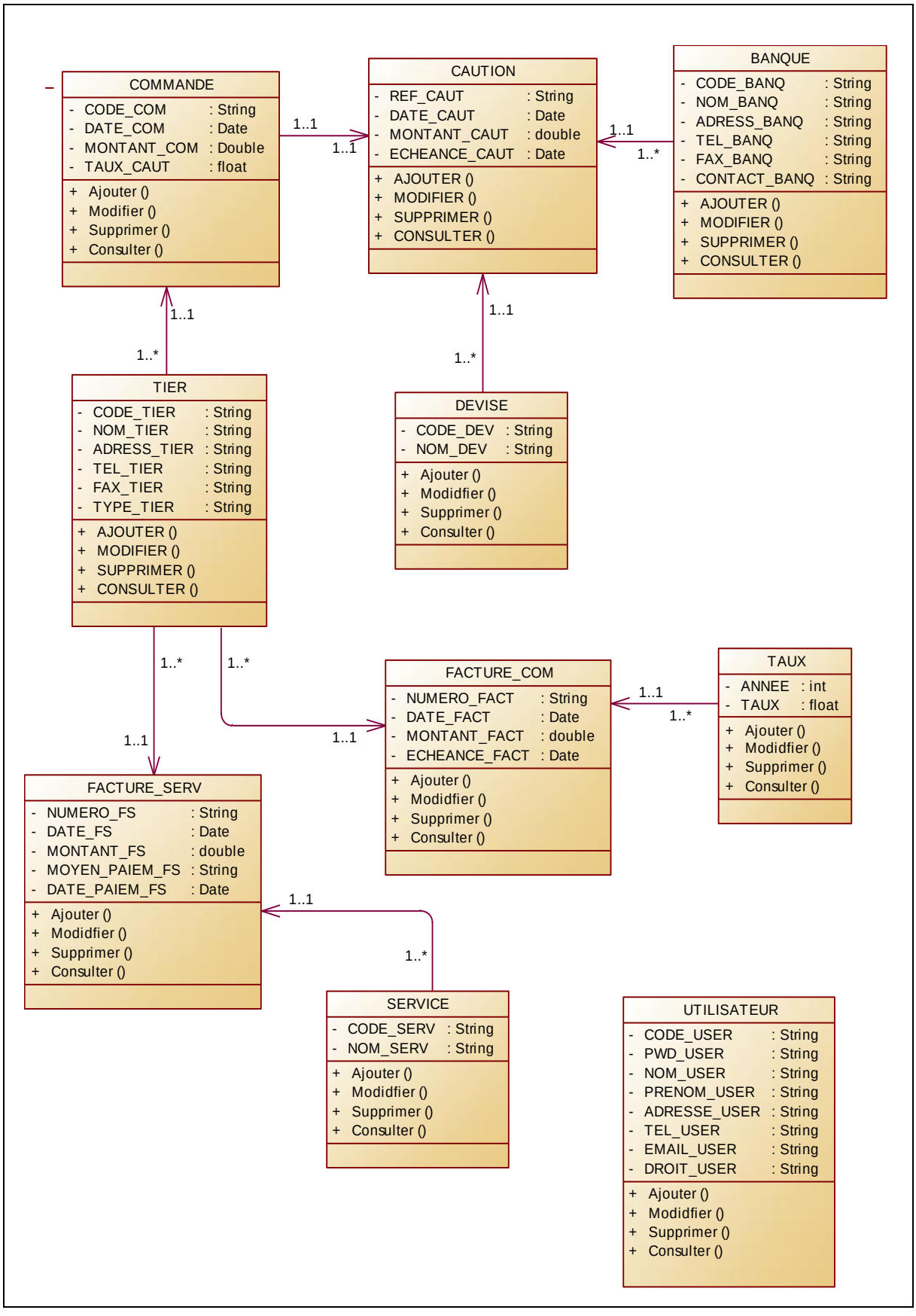


Figure 12: Diagramme de classes

III.3.2 Conception de la Base de Données

D'après le diagramme de classe, notre système comporte 10 tables :

Nom de la table	Description
COMMANDE	comprend les données relatives aux commandes.
BANQUE	représente la liste des banques
TIERS	représente selon le type de tiers la liste des fournisseurs, clients et bénéficiaires de l'application GCAF.
CAUTION	représente la liste des cautions bancaires.
DEVISE	représente la liste des devises utilisées.
FACTURE_COM	représente les factures commerciales de vente de phosphate.
TAUX	représente les taux d'assurances à appliquer par année.
SERVICE	représente la liste des services fournis par la CPG.
FACTURE_SERV	représente la liste des factures de services.
UTILISATEUR	représente la liste des utilisateurs autorisés à utiliser l'application.

Tableau 7: Les tables de la base de données

III.4 Conception Aspect Dynamique

III.4.1 Diagramme d'activité

Le diagramme d'activité représente la dynamique du système. Il montre l'enchaînement des activités d'un système ou même d'une opération. IL représente le flot de contrôle qui retrace le fil d'exécution et qui transite d'une activité à l'autre dans le système. Il est le plus approprié pour modéliser la dynamique d'une une tâche, d'un cas d'utilisation lorsque le diagramme de classe n'est pas encore stabilisé. Nous présentons dans ce qui suit quelques diagrammes d'activité. [11]

Diagramme d'activité de gestion des utilisateurs

Le diagramme ci-dessous explique la gestion des utilisateurs :

- ➔ L'administrateur doit tout d'abord s'authentifier : si le test des données saisies échoue (données non conformes), l'utilisateur ne pourra pas accéder au système.
- ➔ Dans le cas contraire (test valide), une liste des utilisateurs déjà enregistrés s'affiche et l'administrateur peut dans ce cas se contenter de consulter cette liste, puis il abandonne l'application, ou bien il peut à son choix ajouter, modifier ou supprimer un ou plusieurs utilisateurs.

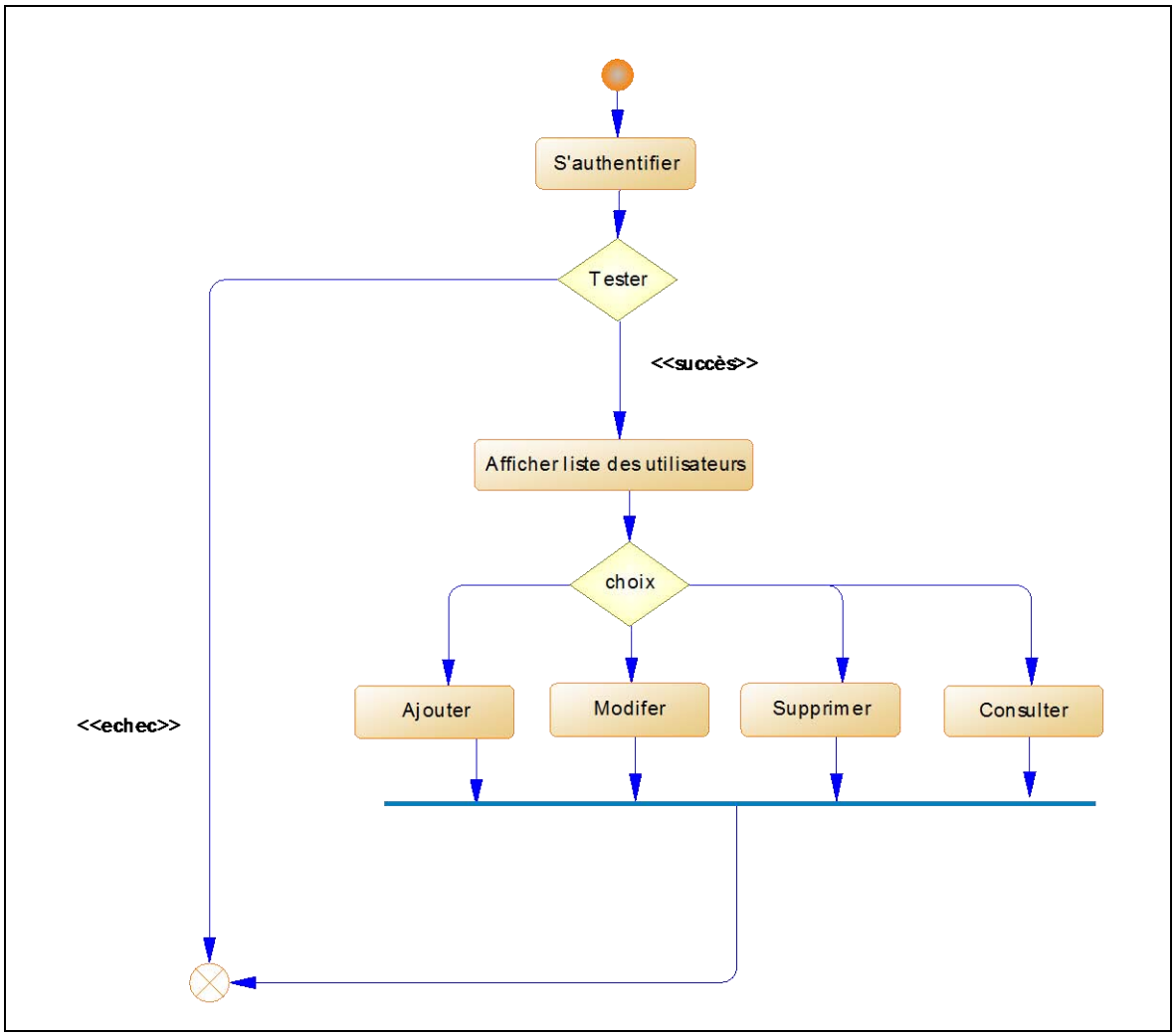


Figure 13: Diagramme d'activité "gestion des utilisateurs"

Diagramme d'activité de gestion des cautions

La figure suivante présente le diagramme d'activité de gestion des cautions :

➔ Après une authentification réussite, l'utilisateur accède à l'interface caution pour consulter la liste des cautions, modifier, supprimer ou ajouter une nouvelle caution.

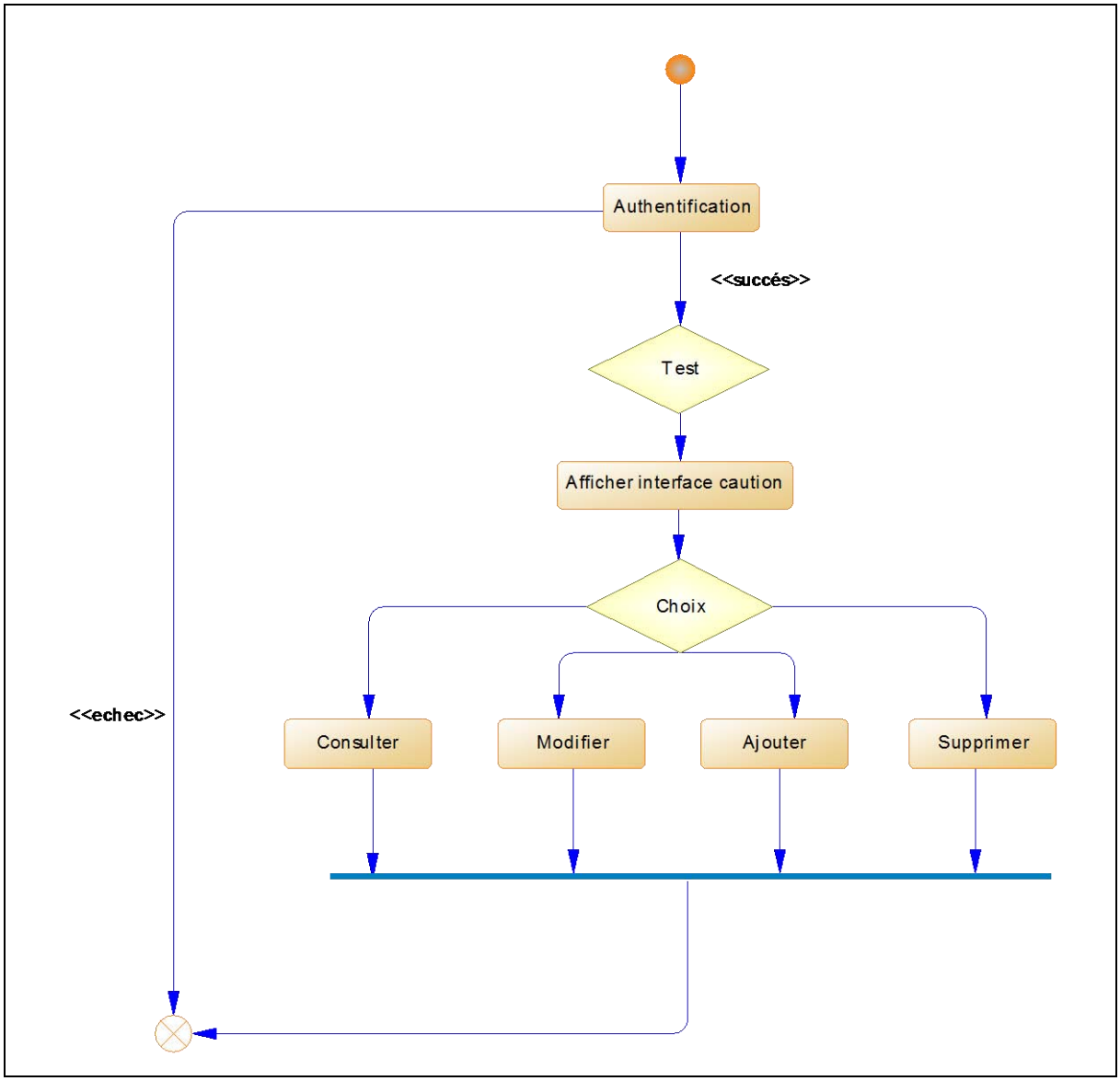


Figure 14: Diagramme d'activité "gestion des cautions"

Diagramme d'activité : Exécution de cautions bancaires

Le diagramme suivant traite le cas d'exécution d'une caution bancaire :

➔ L'utilisateur caution s'authentifie, une interface de traitement caution s'affiche, l'utilisateur imprime une demande d'exécution et l'envoie vers la banque.

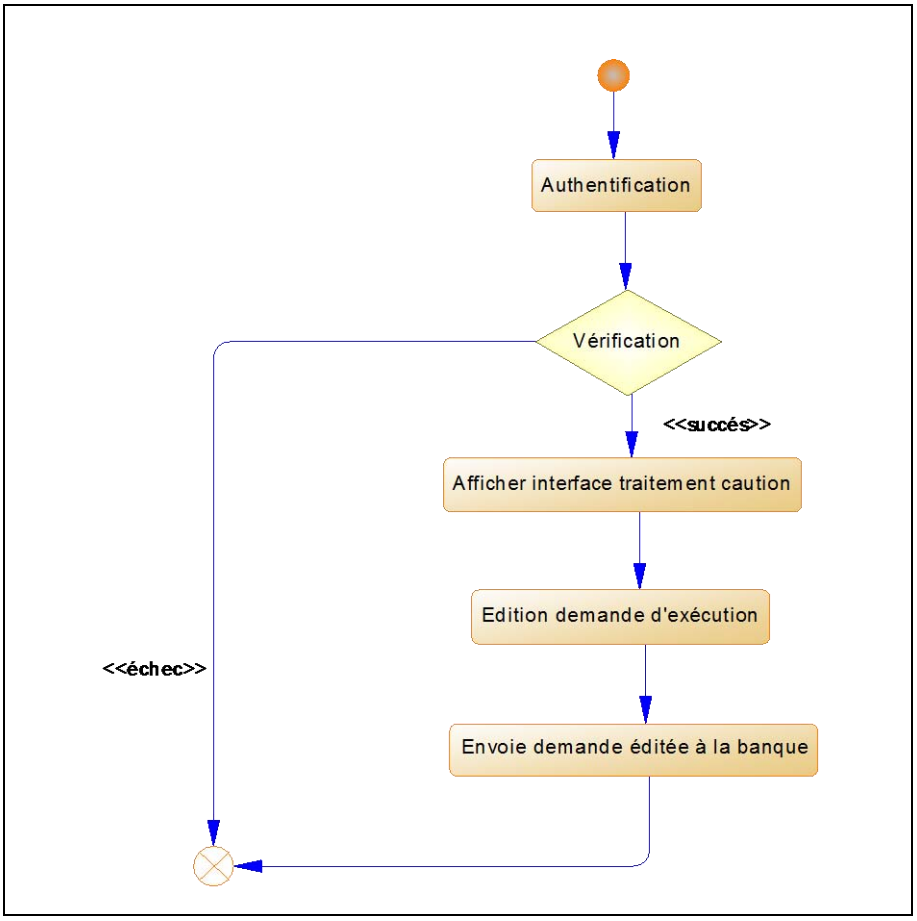


Figure 15: Diagramme d'activité "Exécution de cautions bancaires"

III.4.2 Diagramme de Séquence raffiné

Les diagrammes de séquences permettent de représenter des collaborations entre objets selon un point de vue temporel, on y met l'accent sur la chronologie des envois de messages. Dans cette partie, nous allons donner les diagrammes de séquence relatifs aux cas d'utilisation déjà présentés.

Diagramme de séquence Ajout caution

La figure ci-dessous présente le diagramme de séquence Ajout caution dans lequel nous décrivons la manière avec laquelle l'utilisateur peut ajouter une nouvelle caution.

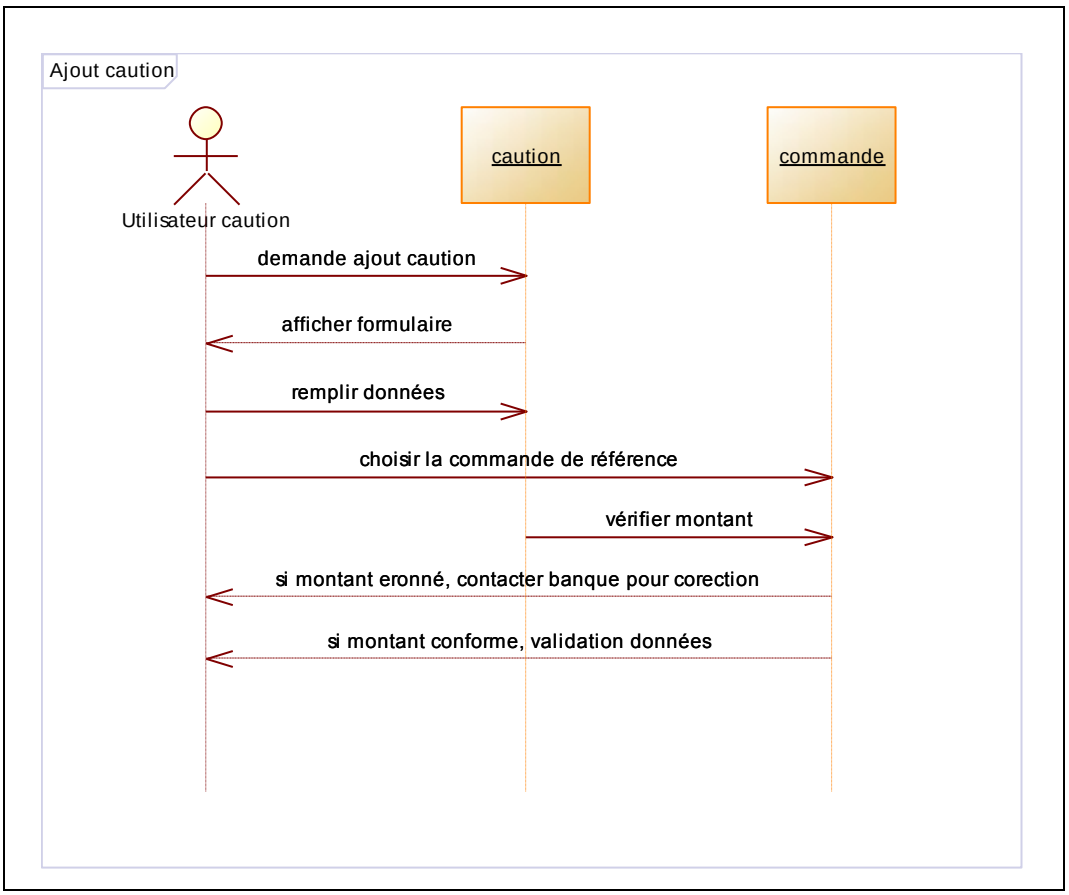


Figure 16: Diagramme de séquence "Ajout caution"

Diagramme de séquence Assurance facture commerciale

Le diagramme suivant nous explique les étapes nécessaires à l'assurance d'une facture commerciale.

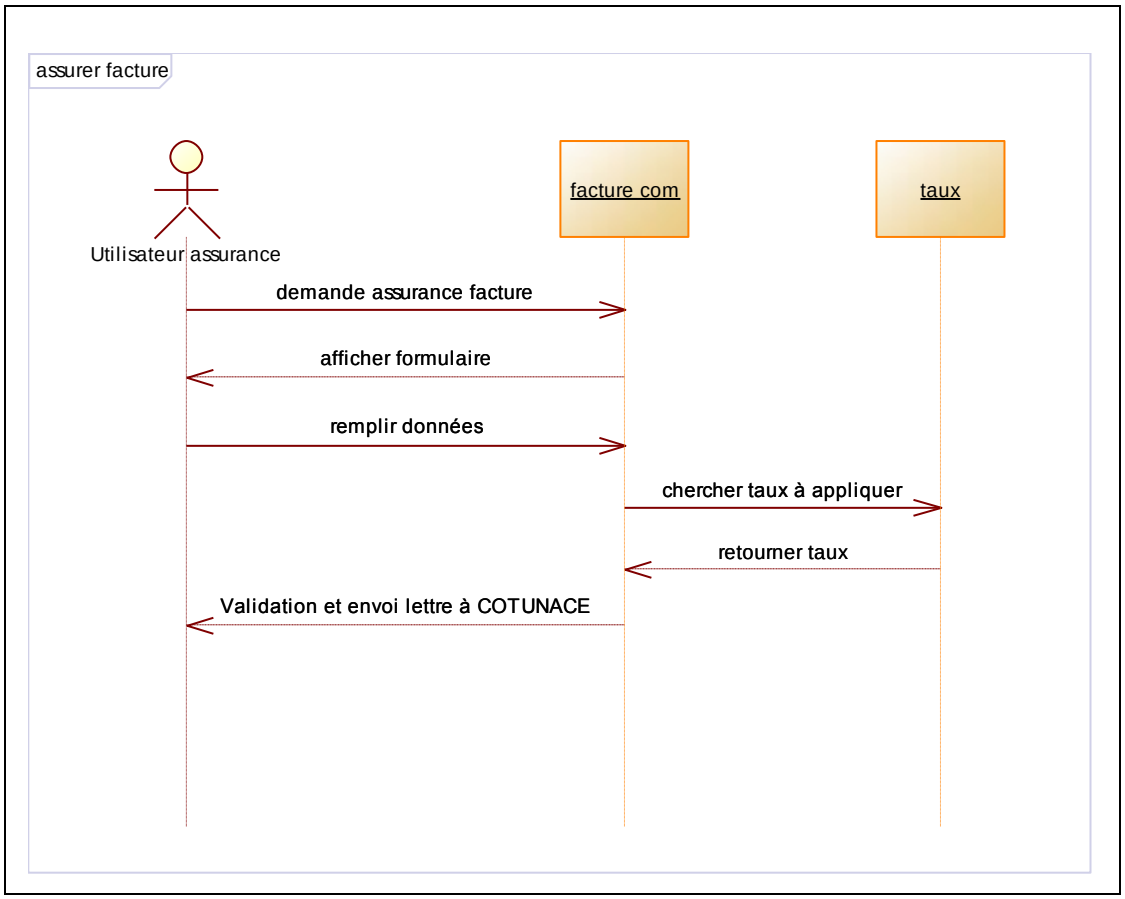


Figure 17: Diagramme de séquence "Assurance facture commerciale"

III. Conception du niveau présentation

Ce paragraphe est consacré pour citer le volet présentation de notre application. Nous commençons par donner le modèle de navigation. Ensuite, nous décrivons la charte graphique.

III.1 Modèle de navigation

La figure N°18 représente le modèle de navigation de notre application :

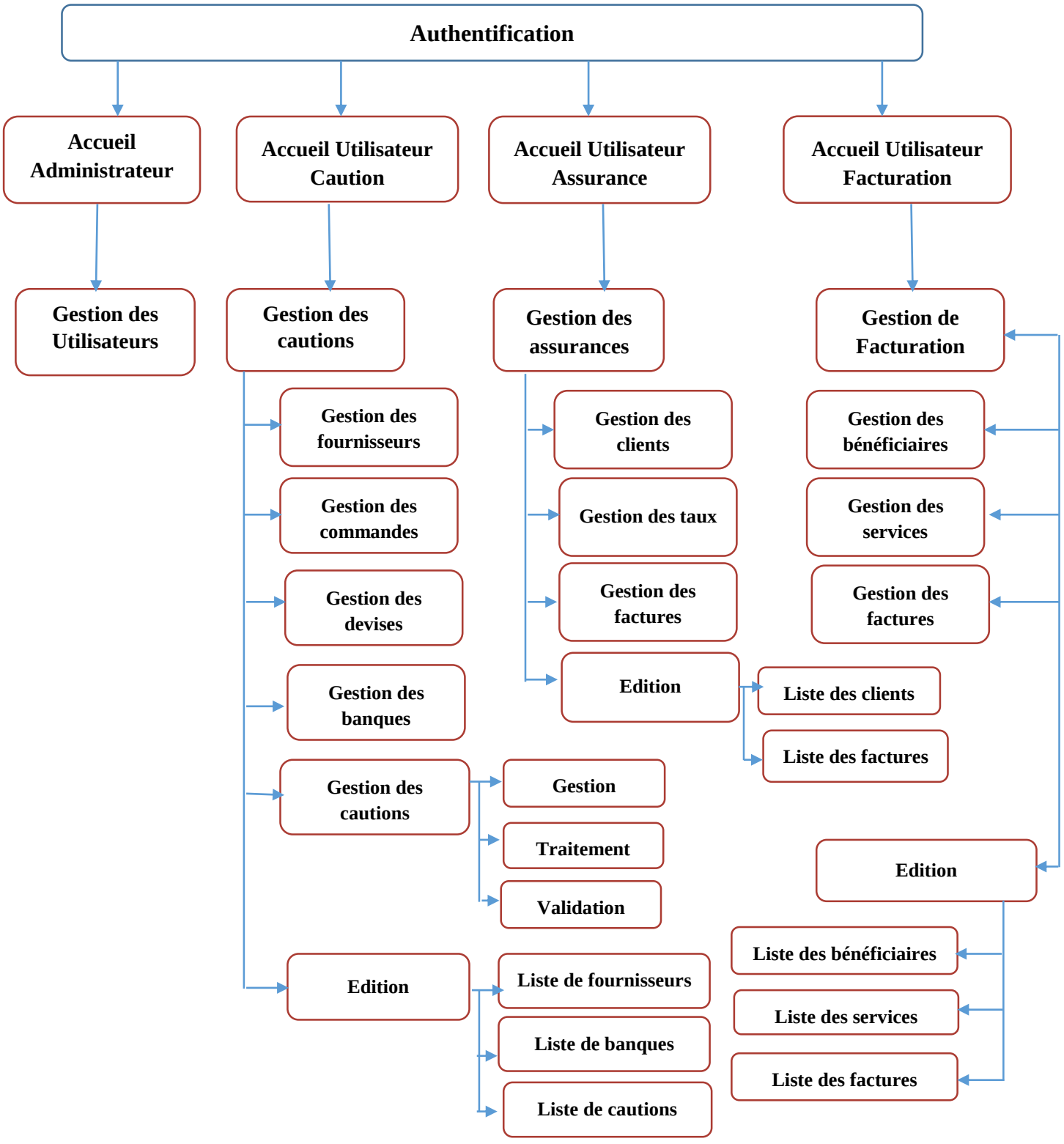


Figure 18: Modèle de navigation

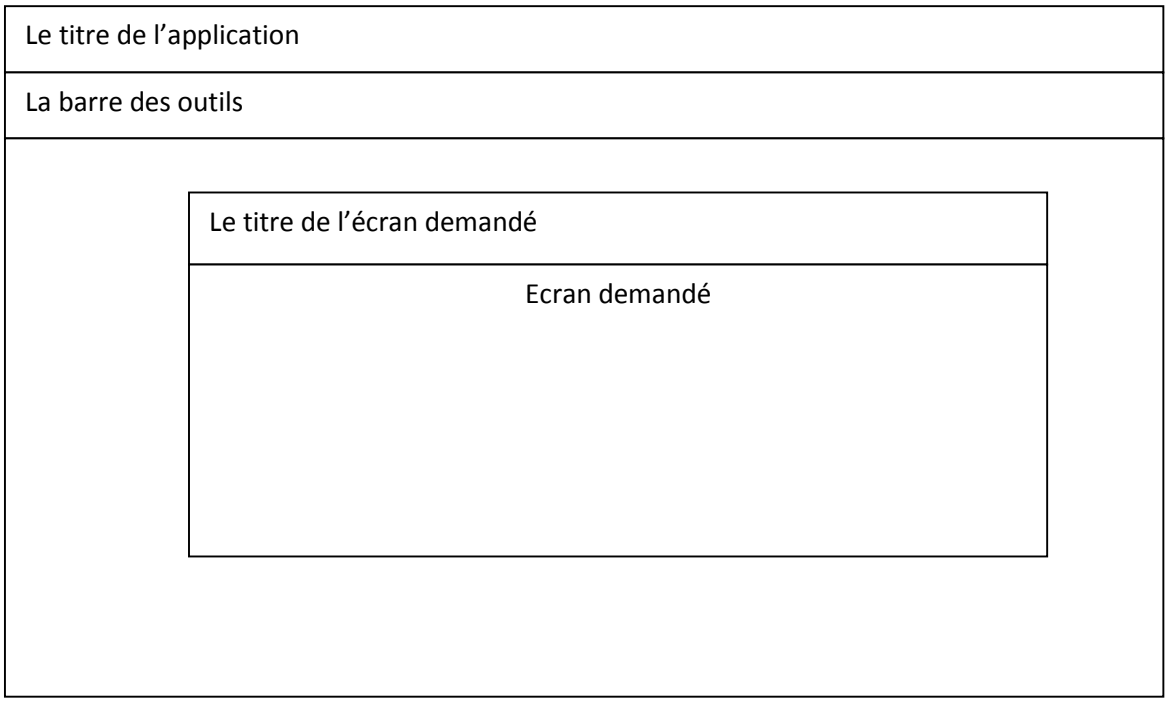
III.2 Charte graphique

La charte graphique d'une application est l'ensemble des symboles et des règles qui définissent son identité graphique. Par extension, on utilise souvent l'expression de charte graphique pour parler de l'ensemble de l'identité visuelle d'une application. L'objectif d'une charte graphique est double. Il s'agit d'une part d'avoir une cohérence graphique quelle que soit l'interface de l'application, et quel que soit le support de communication utilisé, pour renforcer l'image de marque et d'autre part de faciliter la navigation et la lecture grâce à l'utilisation de repères visuels constants. Nous allons utiliser le même fond d'écran, même police dans toute l'application

L'écran d'accueil sera comme suit :

Le titre de l'application
La barre des menus

L’appel à un écran de l’application sera comme suit :



Conclusion

Dans ce chapitre nous avons détaillé les différentes vues conceptuelles de l’application à réaliser. Cette conception est nécessaire à la phase réalisation qui fera l’objet du chapitre suivant.

Chapitre 4 : Réalisation

Introduction

Dans ce chapitre, nous présentons au début l'environnement matériel et logiciel du projet. Ensuite, nous abordons les principales fonctionnalités offertes par notre application aux différents types d'utilisateurs en présentant quelques captures d'écrans ainsi que leurs explications.

I. Environnement du travail

I.1 Environnement matériel

Tout le travail est réalisé sur un PC portable DELL caractérisé par :

- Processus Intel® Core™ **i3**
- RAM 4 GO
- Disque dur 500 GO
- Système d'exploitation Microsoft Windows 8.

I.2 Environnement logiciel

I.2.1 Création de la base de données

Pour la création de notre base nous avons installé Oracle 11g Express Edition (Oracle Data base XE). Il s'agit, en fait, d'un système de gestion de base de données relationnelle d'entrée de gamme et à petite empreinte mémoire. Il est libre de développer, déployer et distribuer ; rapide à télécharger ; et simple à administrer.

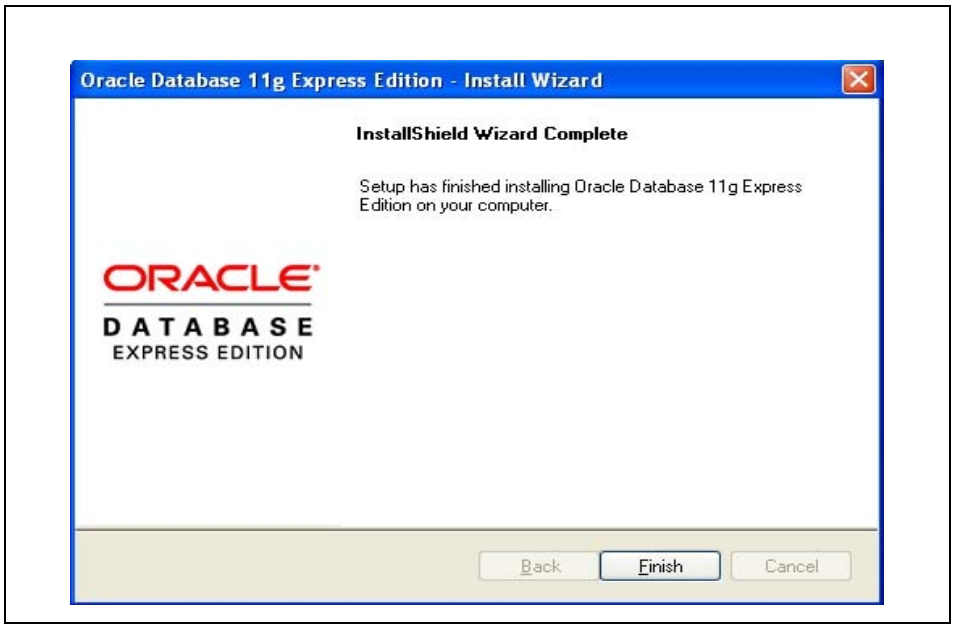


Figure 19: Installation Oracle Data base 11g Express Edition [12]

L'étape suivante est l'installation de PL/SQL Developer. C'est un environnement de développement intégré pour développer, tester, déboguer et optimiser les unités de programme Oracle PL/SQL stockées comme les progiciels, des signaux déclenchant un processus particulier, etc.

Après avoir installé les logiciels nécessaires, nous avons pu créer notre base de données comme la montre la figure N°20 :

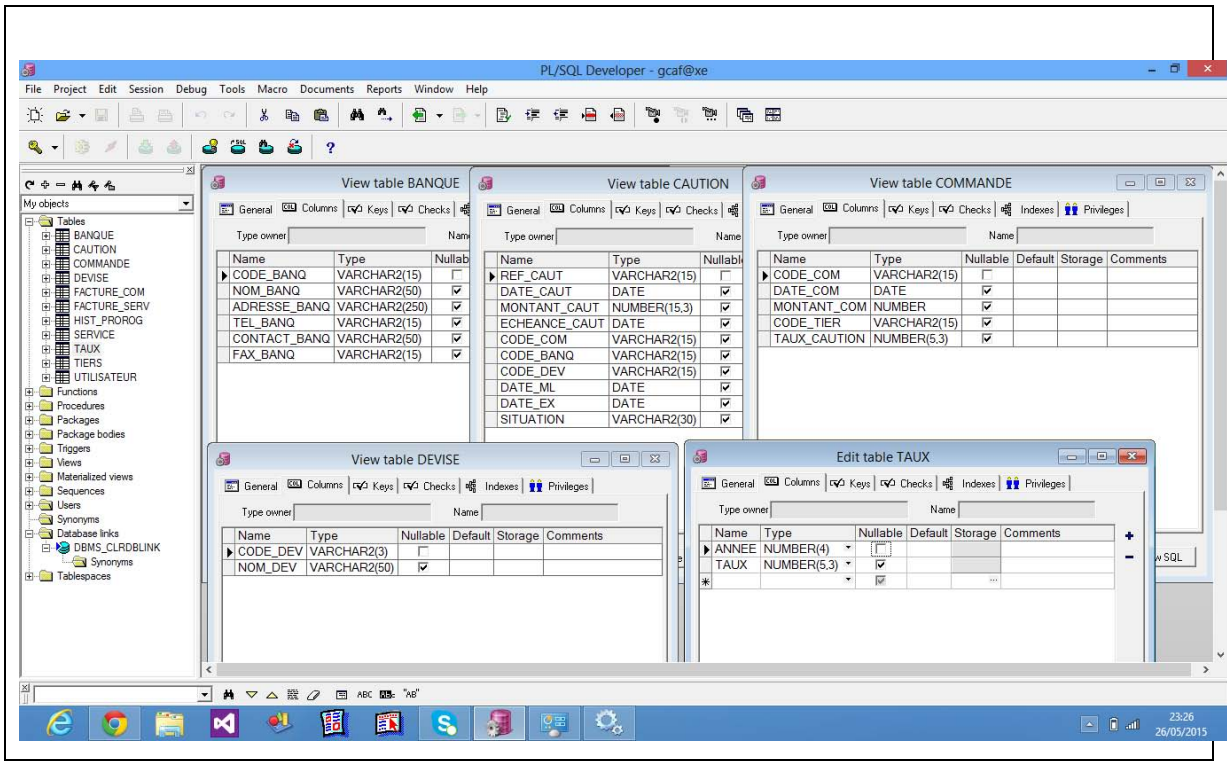


Figure 20: Création de la base de données

I.2.2 Outils de développement

Pour développer notre application, nous avons utilisé Devoloper 2000. C'est un outil de développement front-end d'Oracle. Développeur / 2000 se compose principalement de trois

produits distincts : Oracle Forms, Oracle Reports, et graphiques Oracle. Ces produits peuvent être utilisés individuellement ou en tant que plate-forme de développement d'applications intégrées. Pour notre travail, nous avons utilisé :

- Oracle Forms : C'est un outil pour entrer, accéder, modifier ou supprimer des données à partir d'une base de données Oracle dans un environnement basé sur un formulaire.
- Oracle Reports : il est utilisé principalement pour produire des rapports à partir des données dans la base de données Oracle.

II. Les composantes applicatives réalisées

Ce paragraphe sera réservé pour présenter la majorité des scénarios possibles dans notre application sous forme de captures écrans.

II.1 Interface de connexion

Dès que l'utilisateur tape son code, son nom s'affiche automatiquement si le code est correct il lui suffit de taper le mot de passe correct pour accéder à l'application selon son type d'utilisateur (Administrateur, Utilisateur caution, etc.). Après une authentification correcte, un message de confirmation s'affiche comme le montre la figure ci-dessous :

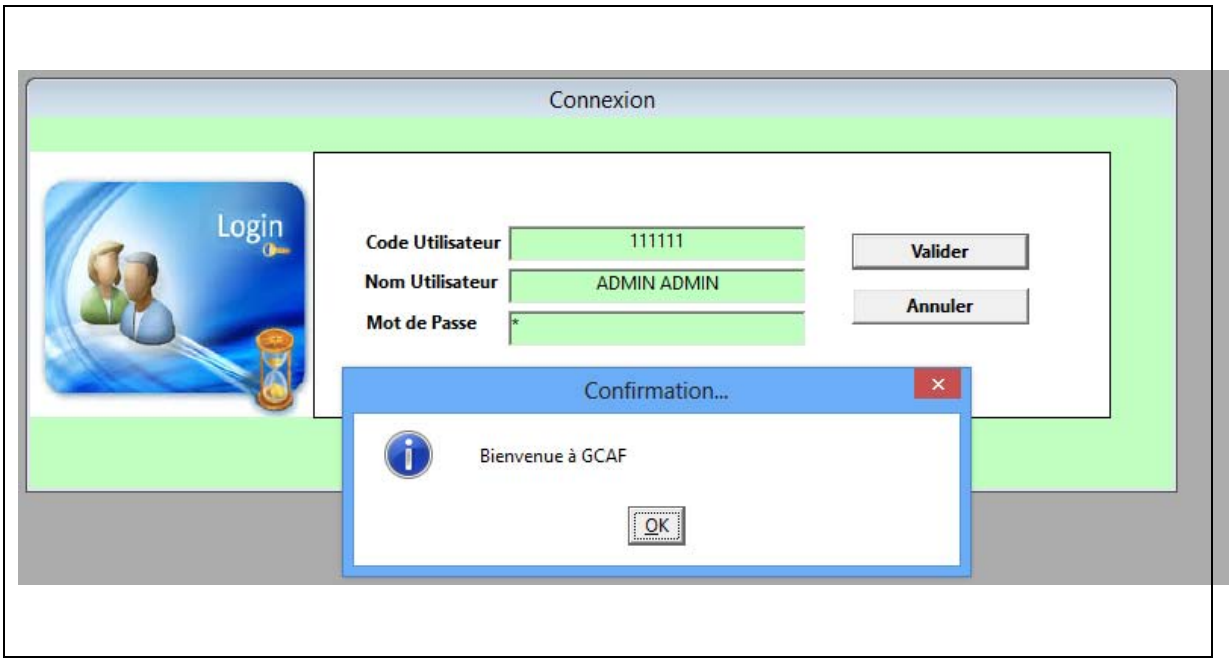


Figure 21: connexion accepté

Les messages suivants s'affichent dans les cas contraires :

- code utilisateur erroné : si l'utilisateur tape un code utilisateur erroné, il ne peut pas terminer à saisir le mot de passe. Le message suivant s'affiche et dans ce cas il doit saisir un code correct ou quitter :

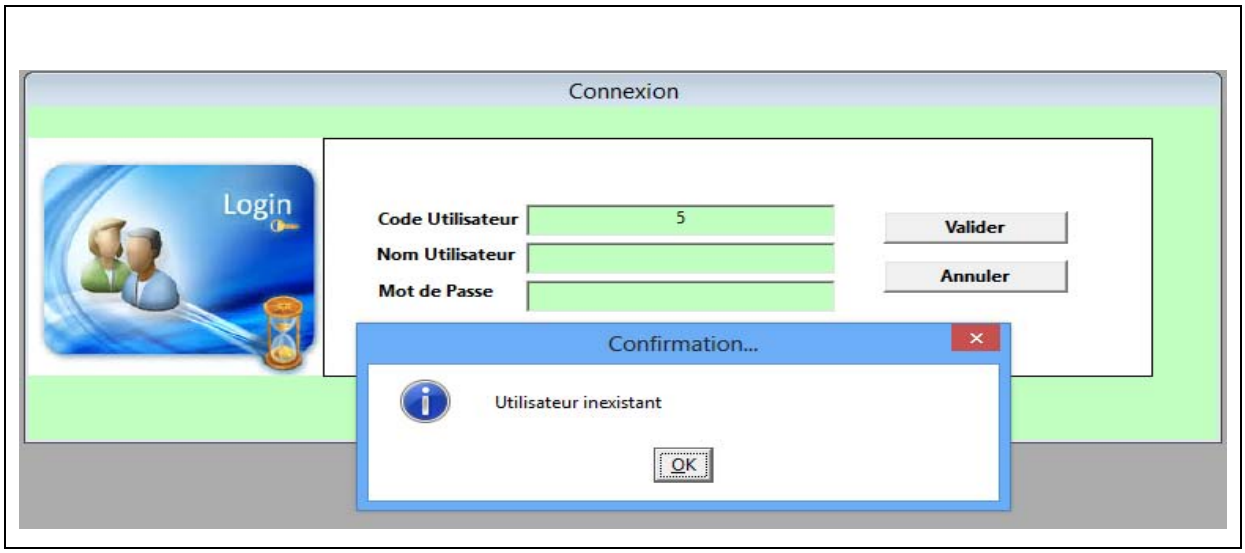


Figure 22: Connexion refusée 1er cas

- mot de passe erroné : le deuxième cas est celui de saisir un mot de passe erroné. Dans ce cas, un message s'affiche comme le montre la capture d'écran suivant :

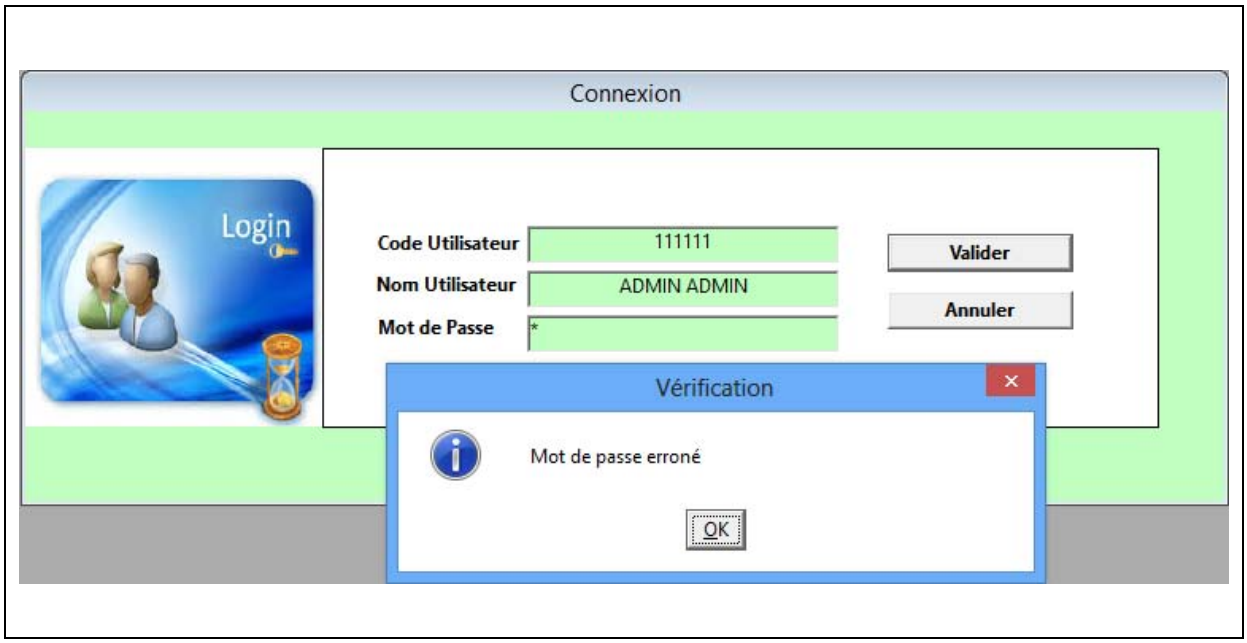


Figure 23: Connexion refusée 2éme cas

- Champs vides : le dernier cas est celui quand l'utilisateur veut accéder sans remplir les champs, dans ce cas ce message lui sera affiché :

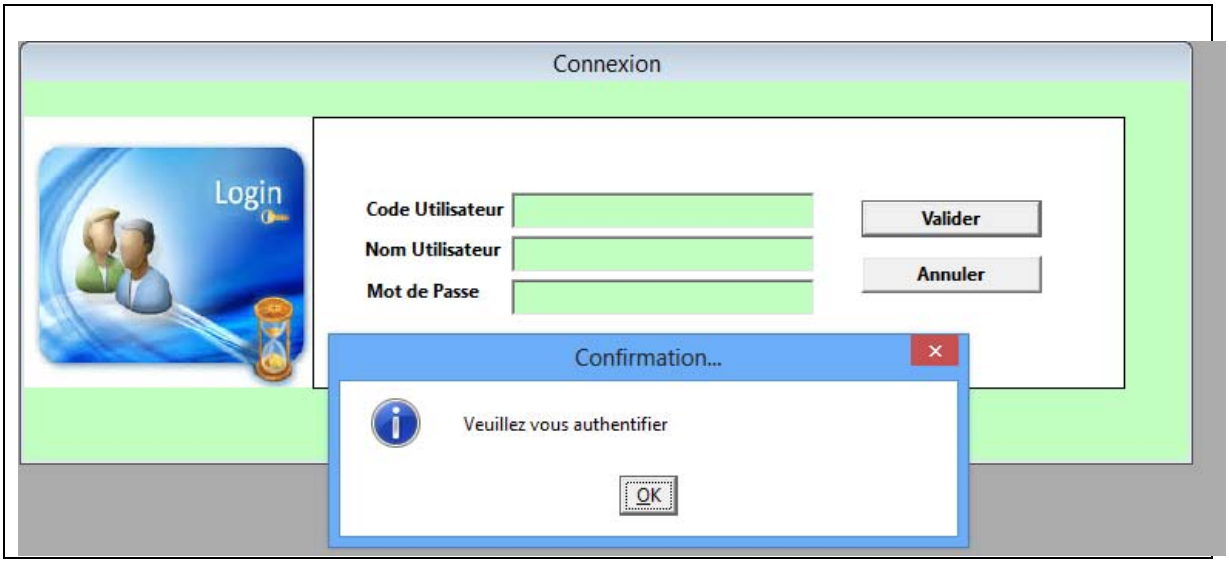


Figure 24: Connexion refusée 3éme cas

II.2 Menu Gestion des cautions

Après avoir réussi à s'authentifier, l'utilisateur caution ne peut accéder qu'au menu : Gestion des cautions :

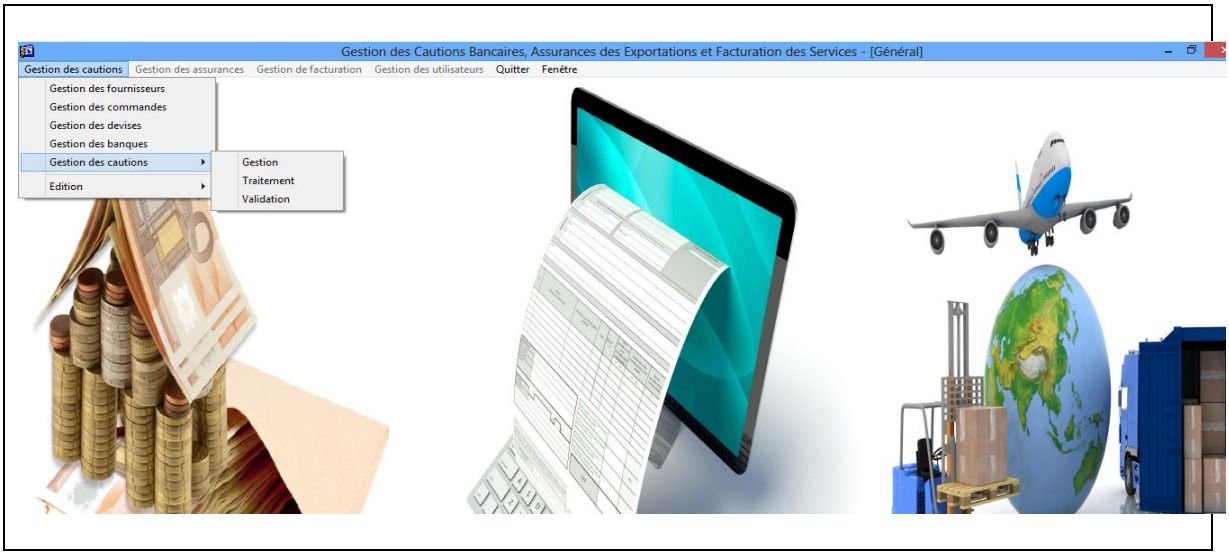


Figure 25: Menu Gestion des cautions

Si l'utilisateur choisit de gérer une caution, il accède au sous menu « Gestion », une liste des cautions enregistrées s'affiche avec tous les détails ; il peut la parcourir et y faire des recherches aussi. Il peut ajouter, modifier ou supprimer une caution.

En cas de saisie d'une nouvelle caution, l'utilisateur remplit les trois premiers champs, puis il sélectionne les trois suivants dans des listes. Le champ « Etat Caution » se met automatiquement à l'état « normal ». Quand l'utilisateur saisie le montant de la caution, le système le compare avec le résultat de la formule suivante : **[*montant de la commande*taux caution/100*]**. Si les deux montants sont égaux il peut faire l'enregistrement dans la base de données. Si non, le message suivant s'affiche :

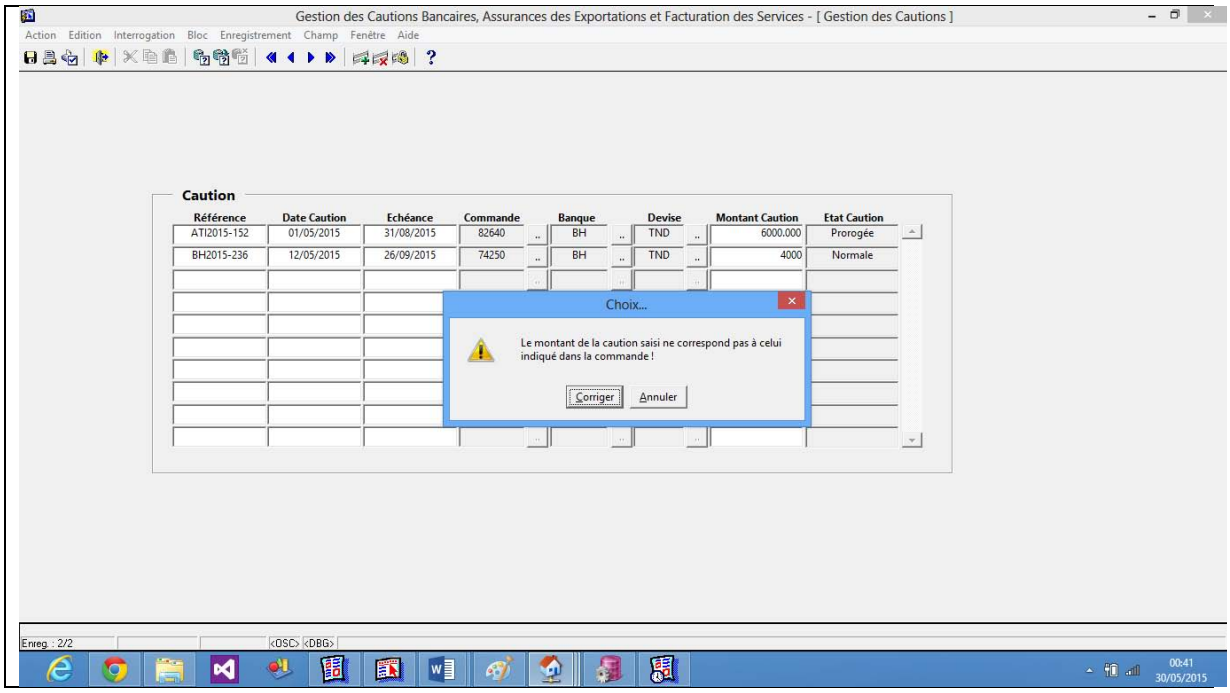


Figure 26: Saisie d'une nouvelle caution

Si l'utilisateur souhaite libérer, exécuter ou proroger une caution, il peut le faire en cliquant sur le sous menu « Traitement » et la fenêtre ci-dessous s'affiche :

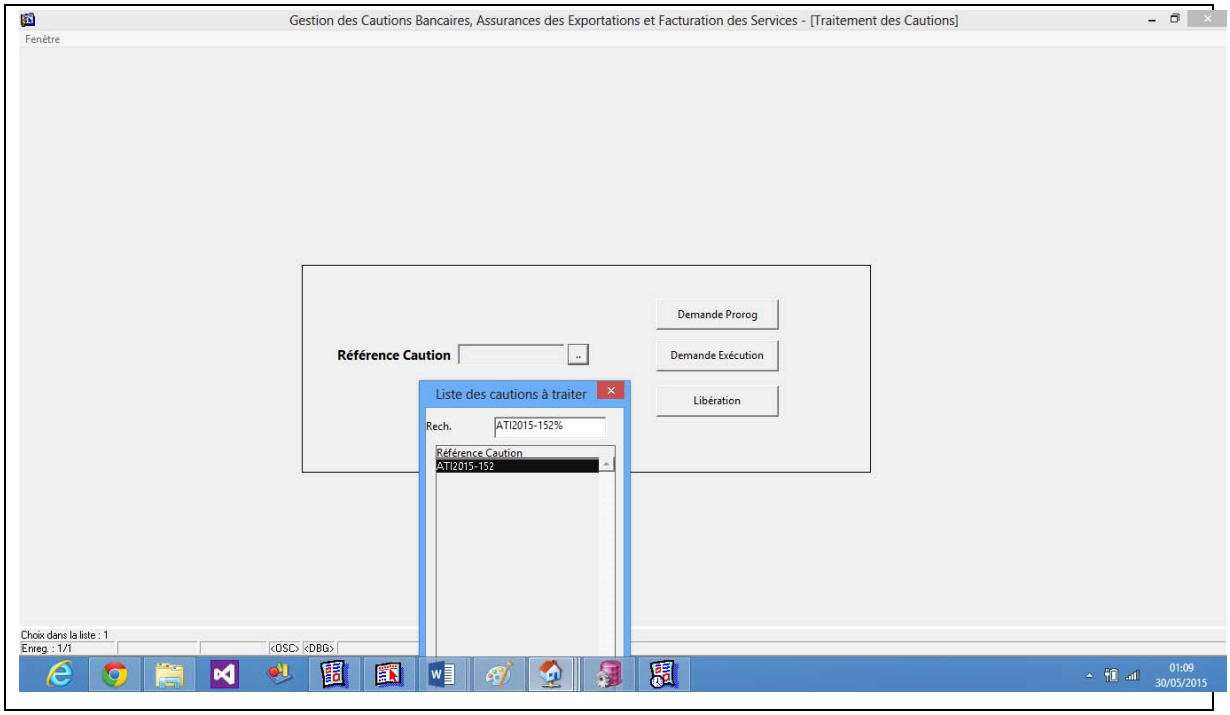


Figure 27: Traitement des cautions

Dès que l'utilisateur clique sur le bouton « Demande Exécution » ou « Libération » de la caution sélectionnée de la liste, la lettre ci-dessous s'affiche comportant toutes les informations nécessaires et l'utilisateur choisit de l'imprimer et l'envoyer avec l'original de la caution à la banque. La banque choisit de proroger et d'affecter une nouvelle échéance ou bien elle procède à l'exécution de la caution. Dans les deux cas, elle doit retourner une réponse écrite comportant la décision.

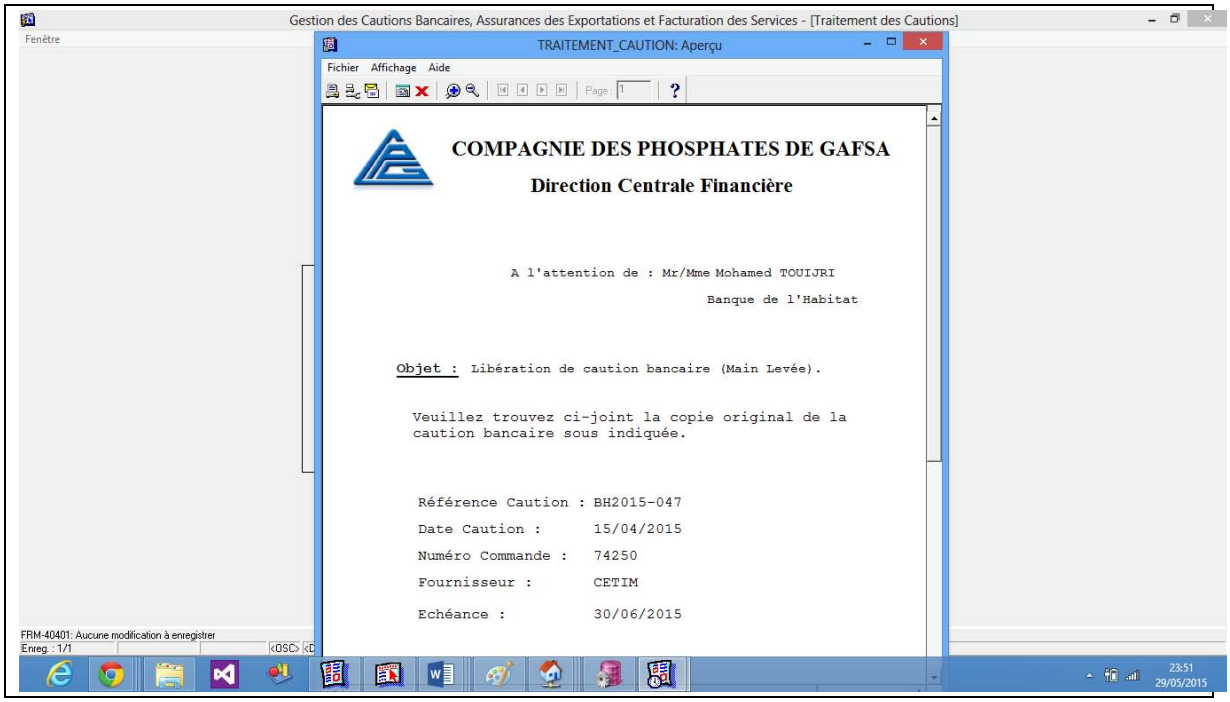


Figure 28: Lettre de main levée

Dans le cas de prorogation, une demande sera éditée et envoyée à la banque et l'état de la caution dans la fenêtre « Gestion » change de « normal » à « Prorogation demandée ». Dès la réception de la validation de prorogation de la banque comportant la nouvelle échéance, l'utilisateur doit la saisir et cette opération se fait dans la fenêtre ci-dessous qui s'affiche en passant par le sous menu « Validation ». Une liste des cautions s'affiche ne comportant que les cautions à l'état « Prorogation demandée » :

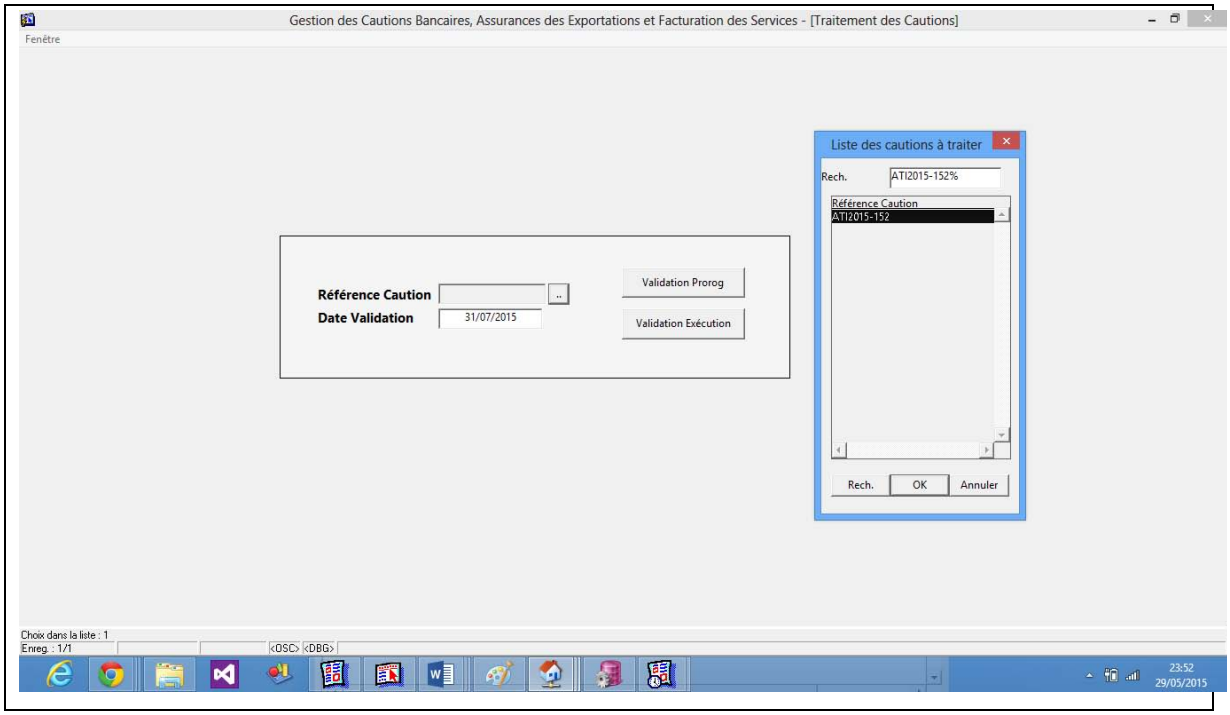


Figure 29: Traitement des cautions

L'utilisateur peut aussi faire des autres éditions en cliquant sur le sous menu « Edition ». La figure suivante présente la liste des fournisseurs :

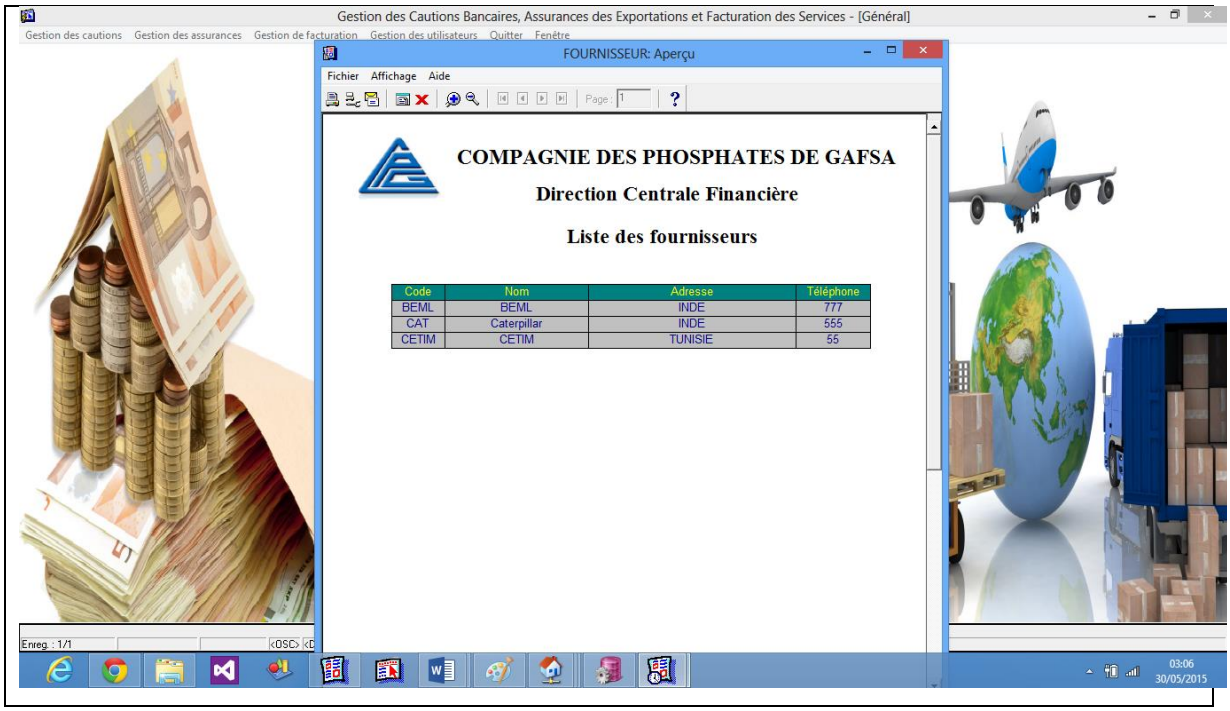


Figure 30: Liste des fournisseurs

II.2 Menu Gestion des assurances

Après une authentification correcte auprès du système, l'utilisateur Assurance peut uniquement accéder au menu Gestion des assurances :

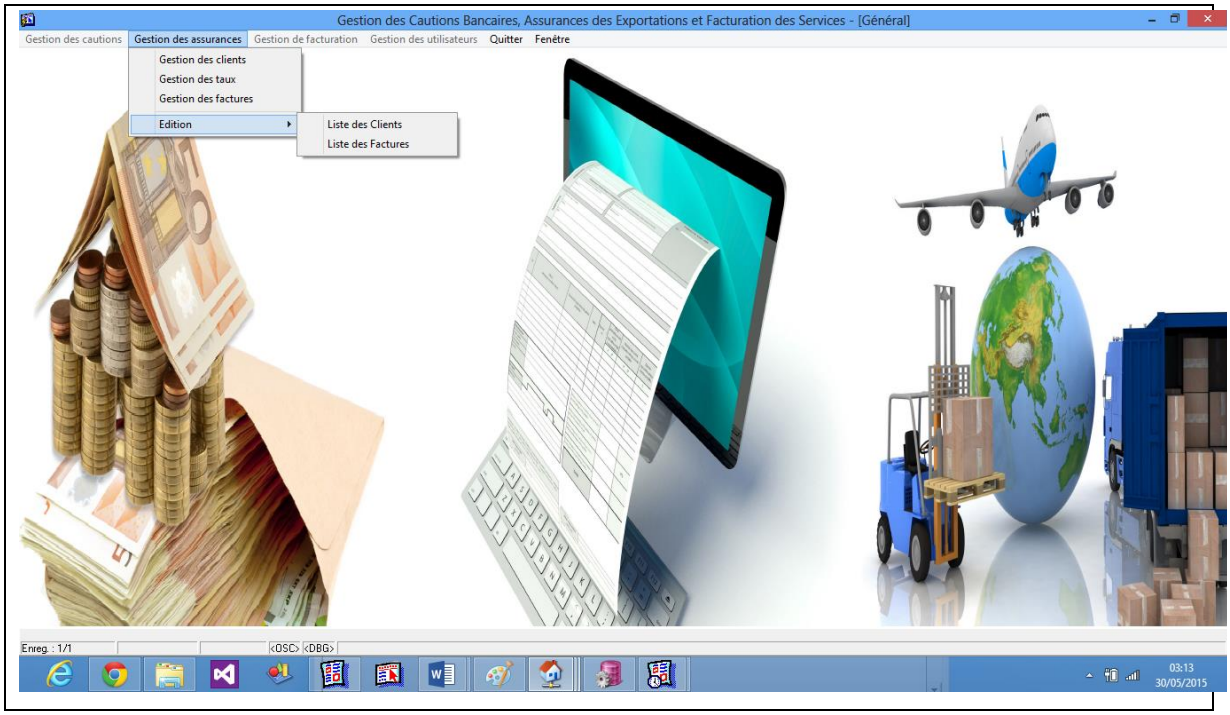


Figure 31: Menu Gestion des assurances

La gestion des clients se fait en cliquant sur le menu « Gestion des clients » pour accéder à la fenêtre suivante dans laquelle l'utilisateur peut ajouter, modifier ou supprimer des clients :

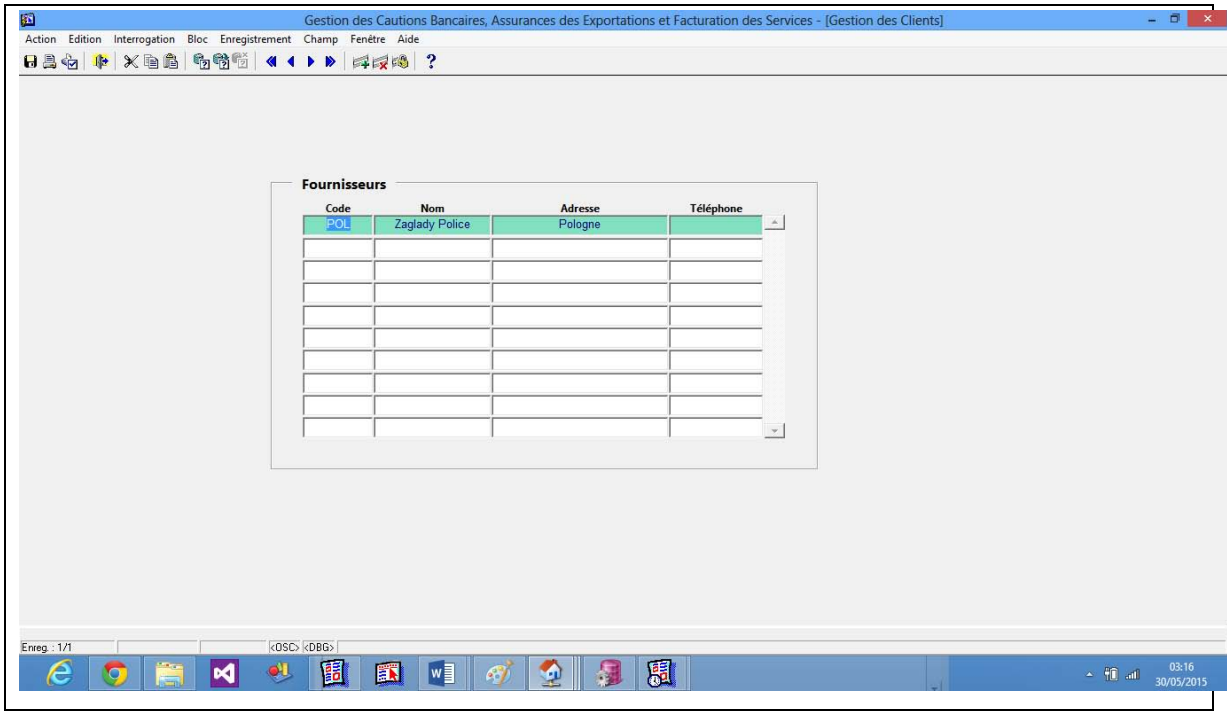


Figure 32: Gestion des clients

A chaque année, La COTUNACE attribut un taux à appliquer sur toutes les factures des assurances des exportations. L'utilisateur doit donc saisir le nouveau taux à chaque année comme le montre la figure suivante :

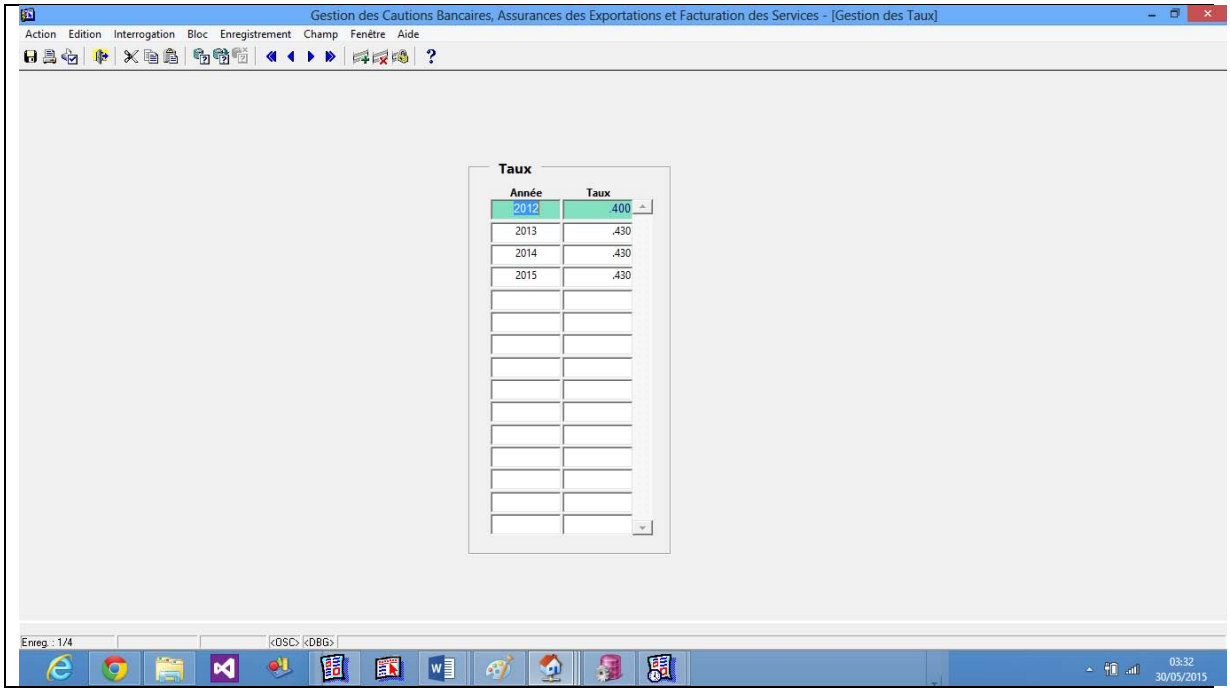


Figure 33: Gestion des taux

Afin de pouvoir gérer les factures, nous avons développé l'écran suivant :

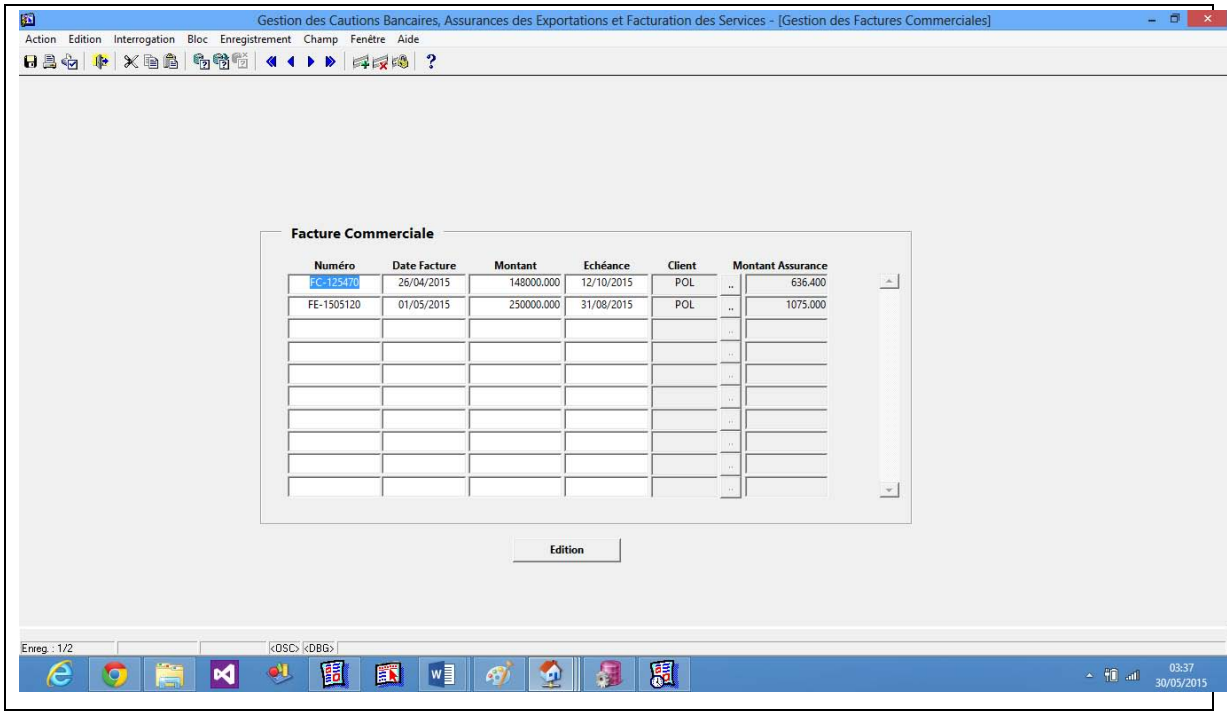


Figure 34: Gestion des factures

Le Montant de l'assurance est calculé automatiquement en appliquant la formule suivante [***Montant*taux***]. L'utilisateur édite ainsi la lettre d'assurance de la facture commerciale qui sera envoyée à la COTUNACE, en cliquant sur le bouton « Edition » :

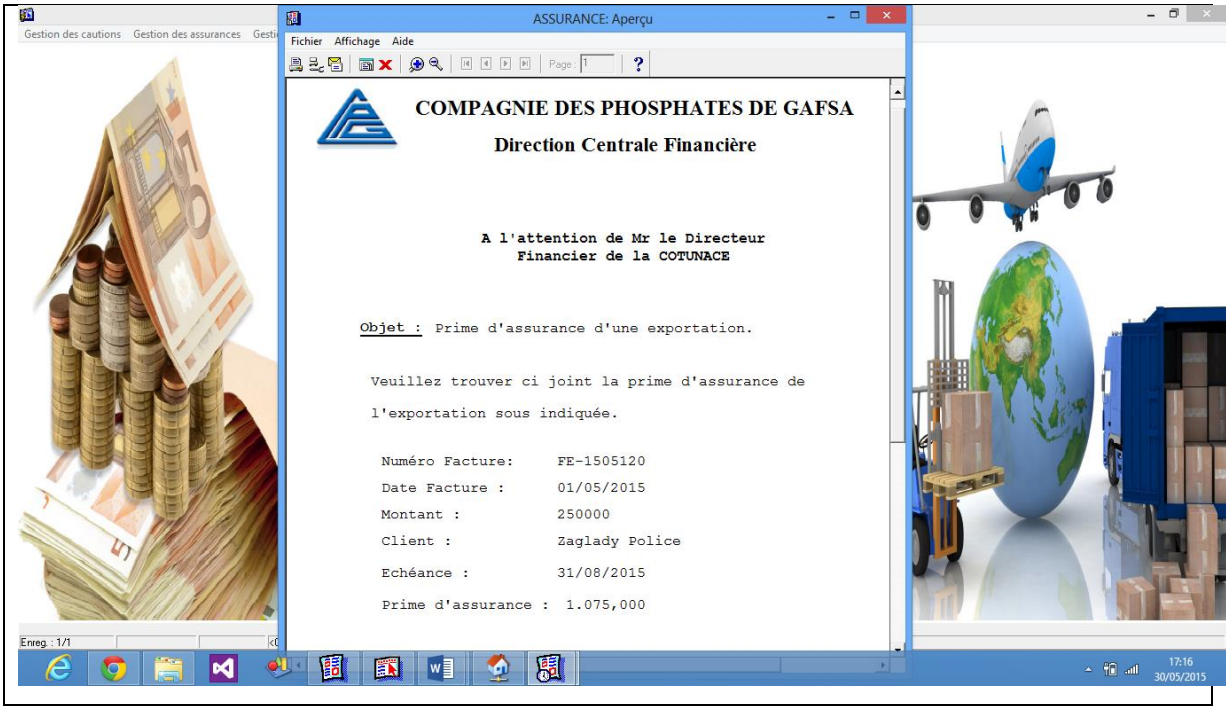


Figure 35: Lettre d'assurance

Si l'utilisateur souhaite, par exemple, éditer la liste des factures, il passe par le sous menu « Editer » --> « Liste des factures » et la liste sera affichée comme suit :

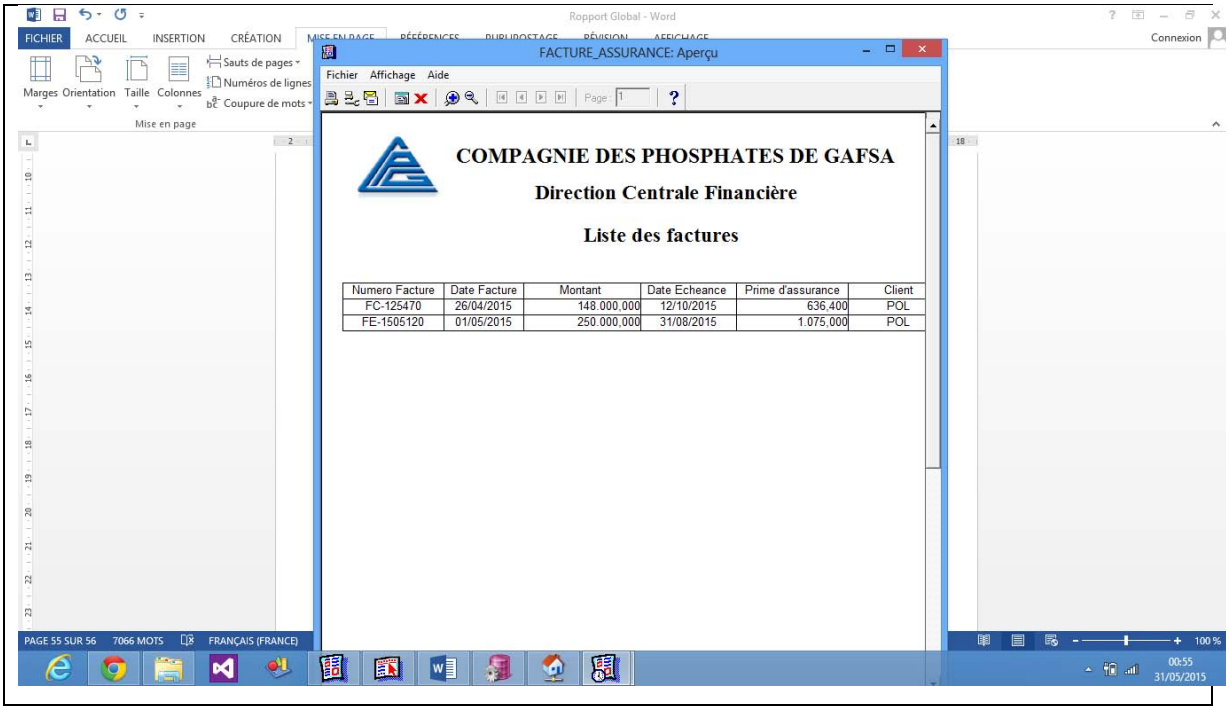


Figure 36: Liste des factures d'assurance

II.3 Menu Gestion de facturation

Quand l'utilisateur facturation réussit à s'authentifier, il accède uniquement au menu qui lui est réservé : « Gestion de facturation » comme l'indique la figure suivante :

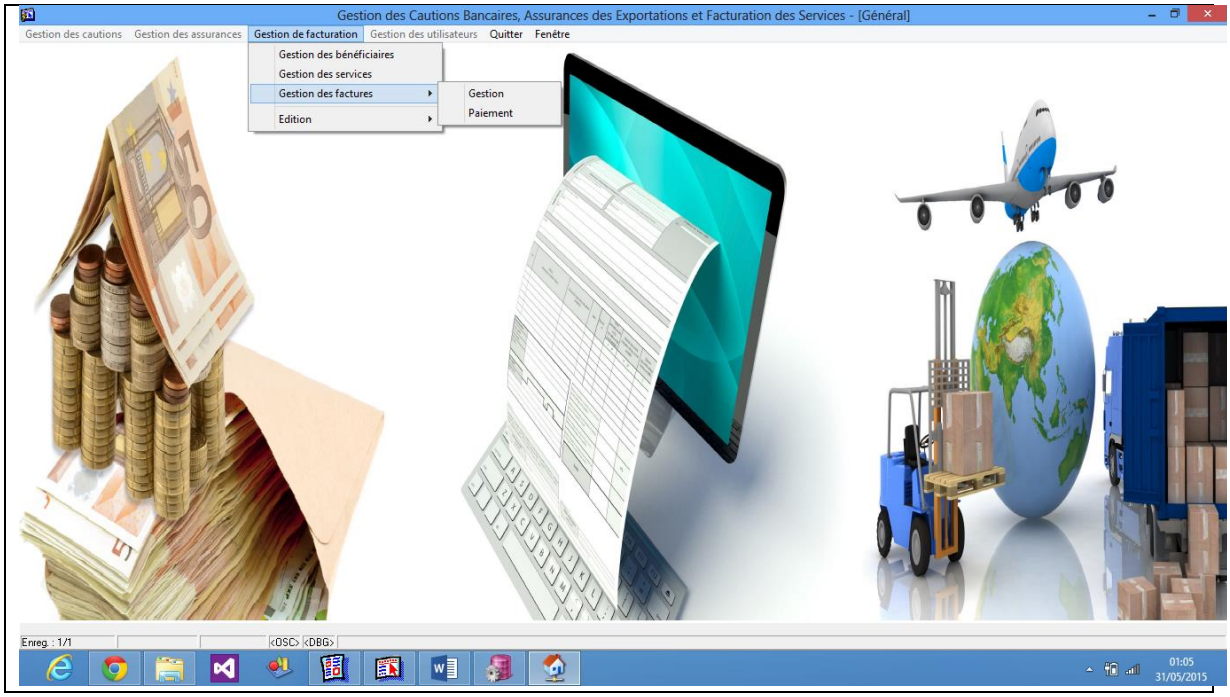


Figure 37: Gestion de facturation

La gestion des bénéficiaires se fait en accédant au menu « Gestion des bénéficiaires ». En cas d'ajout d'un nouveau bénéficiaire, l'utilisateur doit remplir tous les champs pour qu'il puisse enregistrer les nouvelles données :

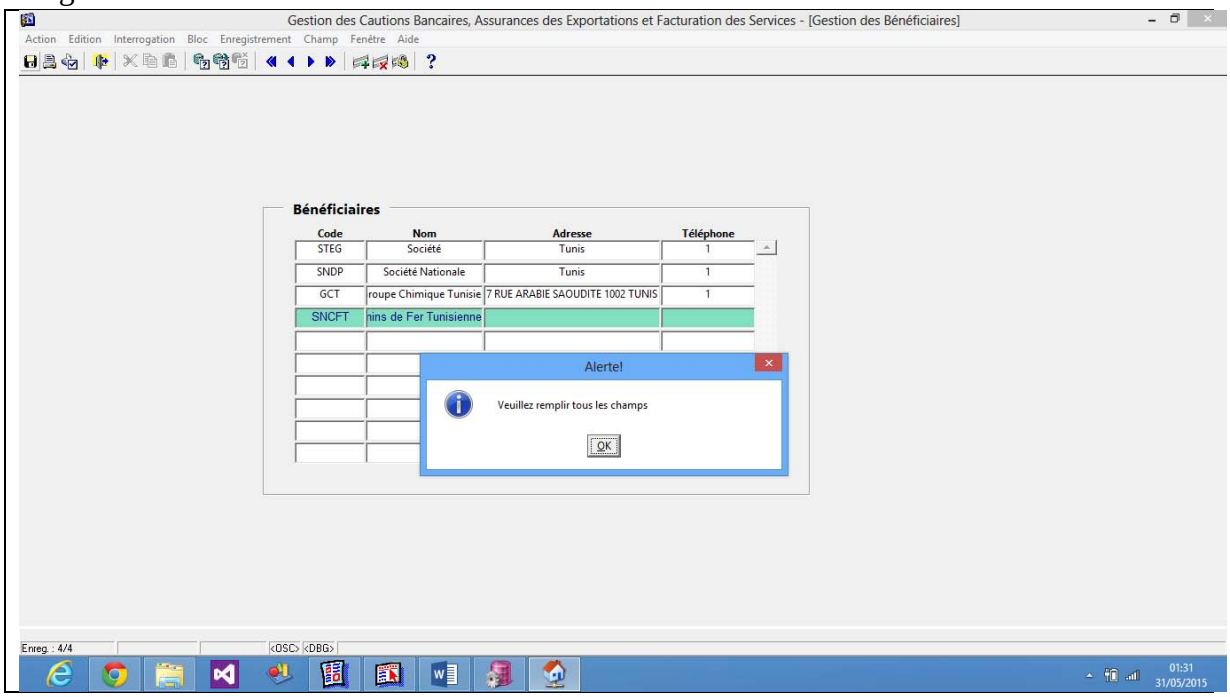


Figure 38: Ajout d'un bénéficiaire

En ce qui concerne la gestion des factures, cette tâche se fait en deux étapes :

➔ La première étape consiste à taper toutes les informations relatives à cette facture, ensuite il suffit de cliquer sur le bouton « Editer » pour avoir une facture détaillée et prête à être imprimée :

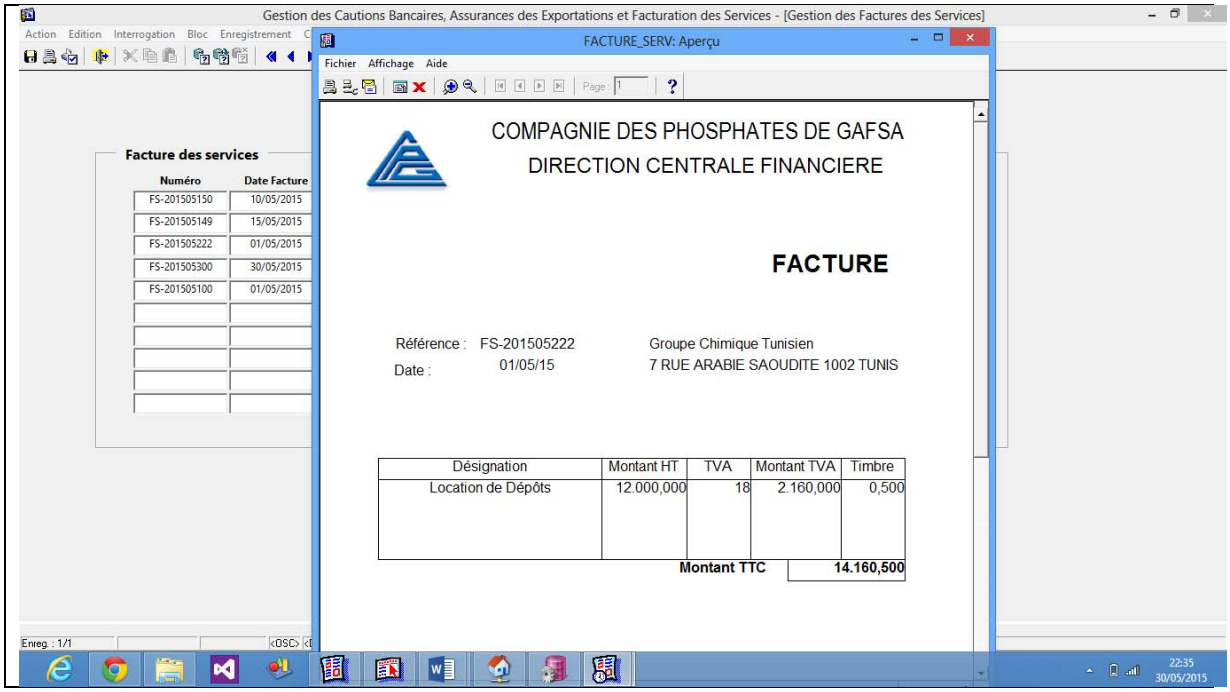


Figure 39: Liste des factures

➔ La seconde étape se présente lors du paiement de la facture par le client. Dans ce cas, l'utilisateur accède au sous menu « Paiement », il se trouve en face de cette interface :

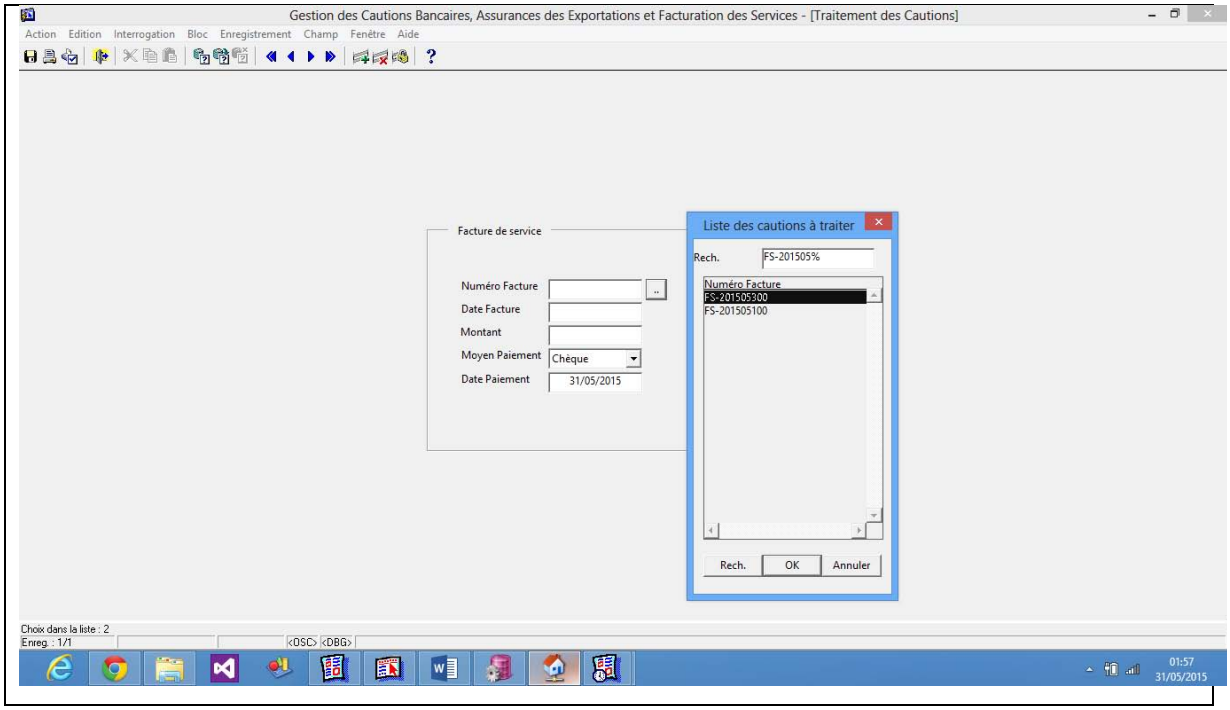


Figure 40: Paiement de facture

L'utilisateur choisit la facture à payer parmi une liste qui ne comporte que les factures non payées. La date paiement sera affichée automatiquement (date du même jour de paiement). Il choisit le moyen de paiement : chèque ou espèce. Enfin il enregistre les données et quitte l'interface.

Le système permet aussi à l'utilisateur facturation de faire quelques éditions en cas de besoin telles que celle de la liste des bénéficiaires :

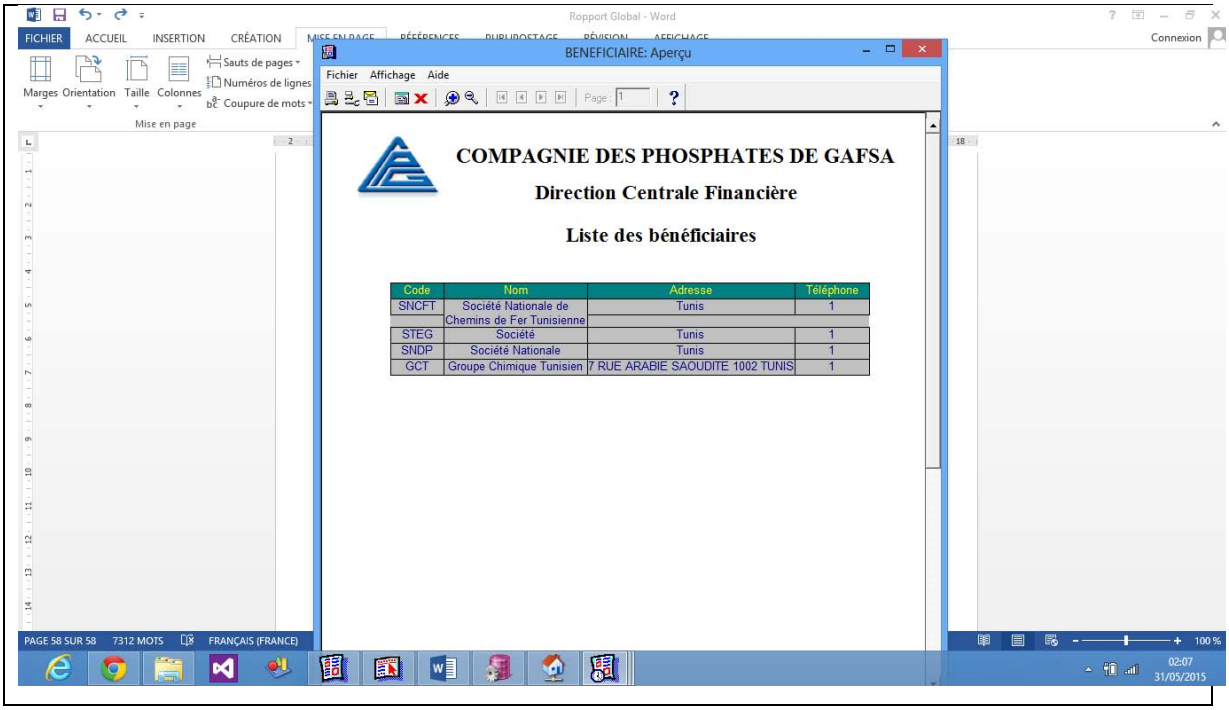


Figure 41: Liste des bénéficiaires

II.4 Menu Gestion des utilisateurs

Le dernier module traité par notre application est celui de gestion des utilisateurs qui est la mission de l'administrateur. Cet administrateur ne pourra accéder aux autres menus :

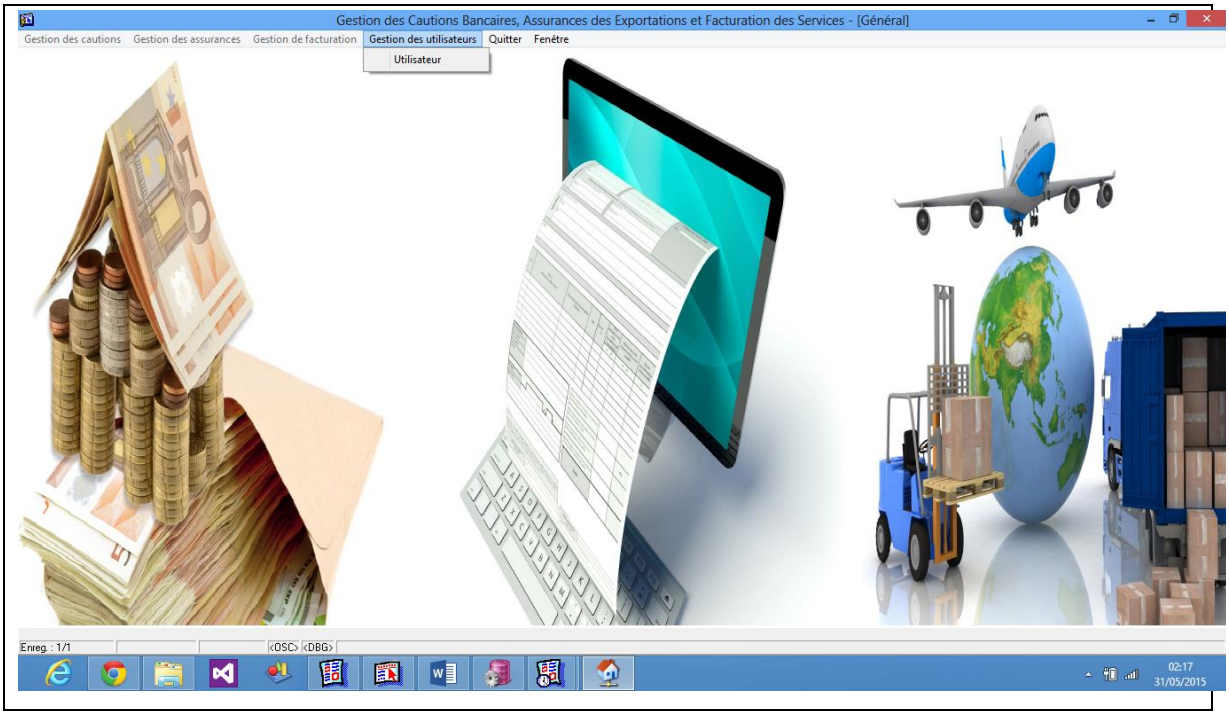


Figure 42: Menu administrateur

A travers l'interface ci-dessous, l'utilisateur peut supprimer, modifier ou ajouter des utilisateurs tout en affectant à chacun son droit d'accès :

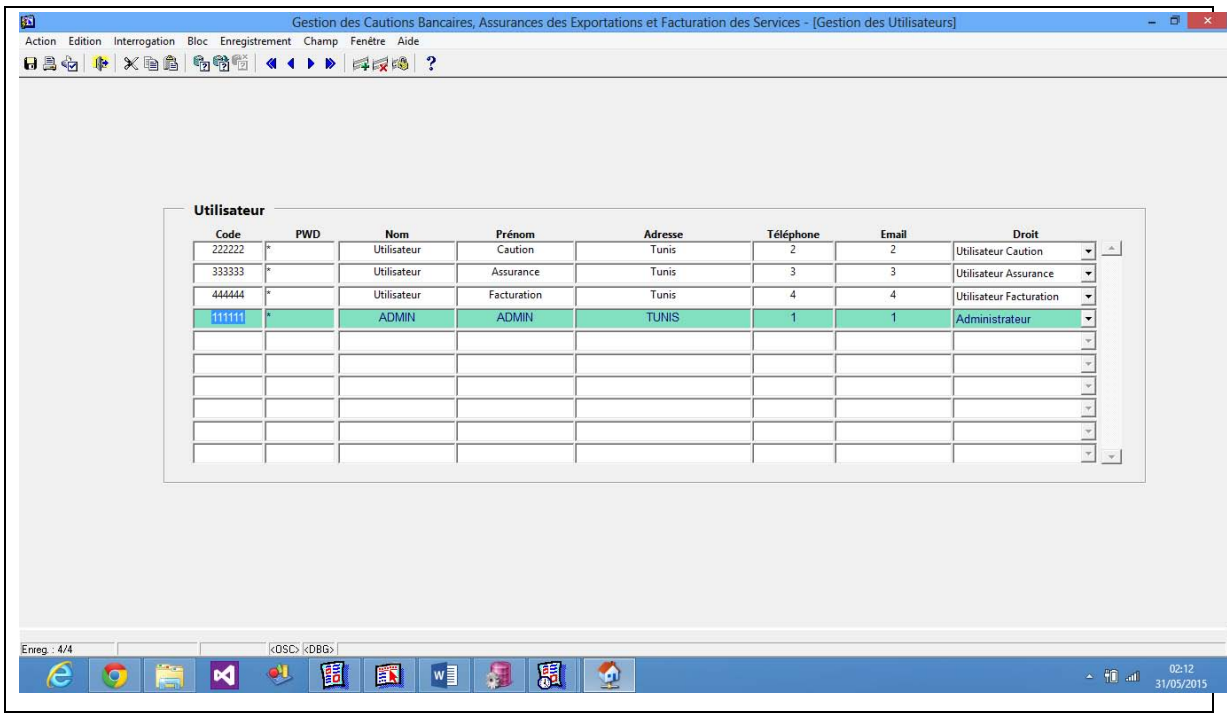


Figure 43: Gestion des utilisateurs

Conclusion

Au cours de ce chapitre, nous avons défini l'environnement du travail matériel et logiciel avec lequel nous avons pu implémenter cette application. Nous avons essayé par la suite d'expliquer le fonctionnement de notre système en présentant quelques imprimes écrans de chaque module.

En achevant ce chapitre, nous nous approchons de la fin de ce mémoire qui sera clôturée par une conclusion générale.

Conclusion Générale

Le But de ce travail a consisté à concevoir et développer une application comportant trois modules différents à savoir : La gestion des cautions bancaires, des assurances des exportations et la facturation des services.

La réalisation de ce travail a nécessité au début de faire une collecte d'informations qui nous a permis à son tour de préparer une étude théorique au cours de laquelle nous avons étudié et critiqué l'existant. Cette étude nous a abouti à proposer quelques solutions et en choisir une.

La solution retenue était étudiée profondément et nous avons pu grâce à cette étude déterminer les besoins fonctionnels et non fonctionnels de l'application. Arrivant à cette étape nous étions capables de passer à concevoir notre système.

La phase conception détaillée, a couvert tous les aspects conceptuels : statique et physique de l'application. Nous avons aussi, au cours de ce chapitre, défini l'architecture physique et logique pour mettre en place notre système. Enfin, nous avons clôturé cette phase en traçant le modèle de navigation de l'application et en définissant la charte graphique du projet.

Toutes les étapes sus indiquées nous ont abouti finalement à la dernière étape à savoir : la réalisation. Nous avons débuté ce chapitre par la présentation de l'environnement matériel et logiciel utilisé pour l'implémentation du système. Puis, nous avons enchainé avec une série de captures d'écran illustrant le fonctionnement de l'application.

Ce nouveau système sera mise en place dans la direction centrale financière, pour être exploité par le responsable des cautions bancaires, le responsable des assurances des exportations et le responsable de la facturation des services. Ce système facilitera énormément leurs travaux qui manquaient beaucoup d'organisation et de sécurité de données.

Ce travail a représenté pour nous une véritable occasion pour approfondir nos connaissances tout en rassemblant nos acquis théoriques à l'environnement pratique en utilisant l'UML comme langage de modélisation, PL/SQL Developer pour la création de la base de données dans le SGBD ORACLE et Form Builder/Report Builder pour la création des formulaires, des menus et des états.

Webographie

Nom du site	Date de visite	Référence
http://www.labri.fr	01/04/2015	[1]
www.infres.enst.fr	03/04/2015	[2]
users.polytech.unice.fr	03/04/2015	[3]
fr.slideshare.net	03/04/2015	[4]
http://fredericmirman.blogspot.com	03/04/2015	[5]
itilfrance.com	06/04/2015	[6]
http://bpesquet.developpez.com	10/04/2015	[7]
http://mbf-iut.i3s.unice.fr	12/04/2015	[8]
http://security.stackexchange.com	12/04/2015	[9]
http://uml.free.fr	15/04/2015	[10]
http://saoudyihab.voila.net	16/04/2015	[11]
http://www.oracle.com	01/05/2015	[12]