



République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Ben Youcef BENKHEDDA- Alger1

Faculté des Sciences

Département d'Informatique

Mémoire de Licence

En Informatique

Spécialité : Système d'informatique

Thème

Réalisation d'un système de gestion et de génération des cartes bancaires BEA

Encadré par

- SOUAMI Feryel
- Taleb Saïd

Réalisé par

- BENHAMI Abdelhamid
- BOUHADJEB Ilyes
- KHEYAR Abdesslam

Remerciements

C'est avec un immense plaisir que nous écrivons ces quelques phrases en signe de gratitude et de reconnaissance à tous ceux qui ont contribué à la réalisation de ce projet. Nous remercions énormément nos enseignants à l'université d'Alger I pour leurs efforts fournis durant notre formation de licence et nous remercions particulièrement notre encadreur Mme Souami Feryel pour ses recommandations précieuses et pour sa flexibilité. Nous tenons aussi à remercier Mr Taleb Saïd, le directeur général de la Direction des moyens du paiement de la BEA (Banque extérieur d'Algérie) pour la confiance qu'il nous a accordé. Enfin nous tenons à remercier les membres du jury pour avoir évaluer cette mémoire.

Dédicace

Nous dédions ce projet

À nos parents, qui ont toujours été là pour nous. Nous espérons qu'ils seront fiers de notre travail.

À nos familles, pour leurs encouragements et prières.

À nos meilleurs amis, pour leurs souhaits de succès et discours motivants.

À nos enseignants, pour tous leurs efforts fournis durant notre cycle de licence.

À tous ceux qui ont participé de loin ou de près à la réalisation et le bon déroulement de ce travail.

Merci

Résumé

Ce travail s'inscrit dans le cadre du projet de fin d'études à l'université d'Alger I faculté de sciences, département de Mathématiques et Informatique pour l'obtention du diplôme de licence en système d'informations. Dans ce cadre, nous avons envisagé à concevoir et développer une application de gestion des commandes des cartes BEA ainsi que la génération et la mise à jour de ce dernier, tout en réalisant un système pour donner la liberté à le superviseur d'une agence BEA de contrôler tous les utilisateurs de ce système trouvant dans son agence, ainsi qu'à l'équipe d'informaticiens de la direction des moyens des paiements de la BEA de contrôler les utilisateurs et les agences utilisant cette application.

Mots clés : Application bureau, JAVA, JAVAFX, MySQL, base de données, sécurité . . .

Abstract

This work is part of the project of end of studies to the Algiers University I for obtaining the Bachelor's degree Applied information's System. In this context, we have designed and developed a management back office application of a knowledge base within BEA (External Bank of Algeria), this application will also give the possibility for the supervisor of any agency of BEA to control the users of this application who are affiliated to his agency, also the team of computer scientist of the department of payment methods will be able to manage any user of application and any agency.

Key words: Back office application, JAVA, JAVAFX, MySQL, database, security . . .

ملخص

هذا العمل يدخل في إطار مشروع نهاية الدراسات في جامعة الجزائر 1 ، كلية العلوم، قسم الرياضيات و الإعلام الآلي من أجل الحصول على شهادة الليسانس في نظام المعلومات . في هذا السياق ، قمنا بتصميم و تطوير تطبيق لإدارة طلبات بطاقات بنك الجزائر الخارجي إضافة إلى توليد و تحديث هذه الأخيرة ، بإنجاز نظام يسمح بذلك للمشرف على وكالة البنك السابق ذكره بمراقبة جميع مستخدمي النظام الموجودين في وكالته. كما يسمح لفريق تكنولوجيا المعلومات في قسم طرق الدفع بمراقبة مستخدمي التطبيق(مستخدمين أو وكالات).

كلمات مفتاحية : تطبيق ، حماية ، قاعدة بيانات ، JAVA ، JAVAFX ، MySQL

Table des matières

Table des matières	i
Liste des figures	ii
Liste des tableaux	iv
Liste des abréviations.....	v
Introduction générale.....	6
1. Présentation générale de projet :	8
1.1. Concept de projet	8
1.2. Etude de l'existant.....	10
1.3. Problématique	11
1.4. Objectifs et solutions proposés	12
1.5. Conclusion	13
2. Analyse et conception	14
2.1. Méthode d'analyse et de conception.....	14
2.2. Analyse et spécification des besoins.....	15
2.3. Modélisation des besoins	18
2.4. Conception détaillée.....	35
2.5. Architectures utilisées	37
2.6. Conclusion	39
3. Réalisation	40
3.1. Environnement de travail	40
3.2. Description Détailé de l'application	46
3.3. Protection de système.....	60
3.4. Chronogramme de projet	62
3.5. Conclusion	62
Conclusion générale et perspectives	63
Bibliographie et webographie	64

Liste des figures

Figure 1 Logo de la BEA	8
Figure 2 Organigramme de La BEA.....	9
Figure 3 Logo de la SATIM	10
Figure 4 Logo de L'entreprise mondial VISA.....	11
Figure 5 Logo de l'entreprise mondial MASTERCARD	11
Figure 6 Diagramme cas d'utilisation << Authentification >>	19
Figure 7 Diagramme de cas d'utilisation acteur << Superviseur >>	20
Figure 8 Diagramme cas d'utilisation acteur << Administrateur >>.....	21
Figure 9 Diagramme cas d'utilisation acteur << Employé >>	22
Figure 10 Diagramme de séquence << Authentification >>.....	27
Figure 11 Diagramme de séquence << Ajouter un employé >>	28
Figure 12 Diagramme de séquence << Modifier les autorisations des employés >>.....	28
Figure 13 Diagramme de séquence << Saisir une commande >>	29
Figure 14 Diagramme de séquence << Modifier une commande >>	30
Figure 15 Diagramme de séquence << Consulter le délai et le plafond de la carte >>.....	31
Figure 16 Diagramme de séquence << Activation de la carte >>.....	32
Figure 17 Diagramme d'activité << Supprimer un employé >>.....	33
Figure 18 Diagramme d'activité << Modifier les autorisations des employés >>	33
Figure 19 Diagramme des activités << Saisir commande>>.....	34
Figure 20 Diagramme des activités << Suivie commande carte>>.....	34
Figure 21 Diagramme des activités « Activation de la carte »	35
Figure 22 Diagramme de classe de système	36
Figure 23 L'architecture MVC.....	38
Figure 24 Architecture 3 tiers [7]	39
Figure 25 Logo de JAVA	41
Figure 26 Logo de JAVAFX	41
Figure 27 Logo de JDK.....	42
Figure 28 Fonctionnement de JDBC sur une application JAVA	43
Figure 29 Logo de CSS.....	43
Figure 30 Logo de MySQL.....	44
Figure 31 Logo de StarUML	44
Figure 32 Logo de phpMyAdmin	45
Figure 33 Logo de WampServer	45
Figure 34 Logo de Scene Builder 2.0	46
Figure 35 Logo de Netbeans 8	46
Figure 36 Interface de redirection (Splash Screen)	47
Figure 37 Interface d'authentification.....	47
Figure 38 Interface d'ajout d'employé	48
Figure 39 Interface de consultation de liste des employés.....	48
Figure 40 Interface de sélection d'employé acteur << Superviseur>>.....	49

Figure 41 Interface de sélection d'employé acteur << Administrateur >>	49
Figure 42 Interface de modification d'employé acteur << Administrateur >>.....	50
Figure 43 Interface de suppression d'employé	50
Figure 44 Interface de consultation d'historique des activités des employés	51
Figure 45 Interface de modification des droits des employés acteur <<Administrateur>>	52
Figure 46 Figure 47 Interface de modification des droits des employés acteur <<Superviseur >>	52
Figure 48 Interface gestion des agences	53
Figure 49 Interface des statistiques	53
Figure 50 Interface Menu de l'administrateur	54
Figure 51 Interface Menu de superviseur	54
Figure 52 Interface Menu Principal de L'employé.....	55
Figure 53 Interface Menu commande carte	56
Figure 54 Saisir une commande.....	56
Figure 55 Saisir une commande « Suite »	57
Figure 56 Interface Consulter les commandes (par date)	57
Figure 57 Interface Suivi commande carte	58
Figure 58 Interface menu des mises à jour carte	58
Figure 59 Interface de choix de type de carte à modifier	59
Figure 60 interface de modification de carte perdue avec réémission pour carte autorisé.....	59
Figure 61 Interface de modification de carte de type primaire	60
Figure 62 Interface de régénération des codes confidentiel / codes e-paiement.....	60
Figure 63 Exemple sur le hachage des mots de passe dans la base des données	61

Liste des tableaux

Tableau 1 Les besoins fonctionnelles des acteurs de système	17
Tableau 2 Description textuelle cas d'utilisation << Authentification >>.....	19
Tableau 3 Description textuelle cas d'utilisation << Ajout d'employé >>	23
Tableau 4 Description textuelle cas d'utilisation << gestion des autorisations	24
Tableau 5 Description textuelle cas d'utilisation << Modifier une commande >>.....	25
Tableau 6 Description textuel cas d'utilisation<<Mise à jour de la carte perdue avec réémission>>..	27
Tableau 7 Planning de projet.....	62

Liste des abréviations

BEA Banque Extérieur d'Algérie

DMP Direction des moyens des paiements

SATIM Société d'automatisation des Transactions Interbancaires et de Monétique

SPA Société par actions

IHM Interface Homme Machine

UP Unified Processus

UML Unified Modeling Language

MVC Modèle-vue-contrôleur

JDK Java Development Kit

JDBC Java Database Connectivity

CSS Cascading Style Sheets

SGBD Système de gestion de bases de données relationnelles

SQL Structured Query Language

Introduction générale

Les entreprises généralement et les banques précisément ont une énorme quantité d'informations et des données et par conséquent des connaissances qu'ils s'agissent soit des besoins des client ou les échanges de service avec d'autres entreprises ou bien les données venant de l'environnement professionnel de l'entreprise.

La manière dont une entreprise collecte, partage et traite ses connaissances est essentielle pour pouvoir se développer avec succès. Ceci n'est pas applicable uniquement aux sociétés multinationales gigantesque mais la gestion des connaissances peut bénéficier à tous les entreprises d'une simple superette à une grande banque national comme la Banque Extérieur d'Algérie (BEA), la banque à qui notre projet est dédiée.

Le défi dans notre projet consiste à accélérer et faciliter le processus de traitement et exploitation de ces connaissances de façon cohérente et sécurisée en utilisant l'outil informatique et en se basant sur le système utilisé actuellement dans l'entreprise.

À la Banque Nationale d'Algérie (BEA) et plus précisément à la Direction des moyens des paiements (DMP), où s'est déroulé notre stage, les informaticiens ont besoin d'aide pour chercher, connaître et partager, modifier les données des clients, cet aide leurs sera apporté par une entreprise associé nommé SATIM (Société d'Automatisation des Transactions Interbancaires et de Monétique) généralement par des appels téléphoniques. Avec le nombre énorme des commandes des cartes bancaires et le nombre des problèmes répétitifs relatifs aux cartes bancaires des clients, le processus de traitement de ces problèmes devient lent et le risque des anomalies augmente, il est nécessaire de trouver une solution adéquate afin de répondre à ces besoins plus rapidement.

Notre projet sera une solution proposée à la BEA pour répondre à ces besoins rapidement tout en assurant la confidentialité et sécurité des données des clients en offrant la possibilité à chaque agence de traiter les commandes et les demandes des mise à jours des cartes des clients sans communiquer avec la SATIM , cette dernière recevra chaque jours ou semaines les commandes et les demandes des mise a jours reçus par des fichiers textes et appliquera ces opérations sur les cartes en dur uniquement , ils auront aucun droit d'accès à le système réalisé par nous.

Chaque agence de la BEA utilisera ce système et pour chaque entreprise on spécifiera deux rôles principales : le rôle de la saisie qu'on donnera pour n'importe quel employé utilisant le système quel que soit son grade et le rôle de supervision qu'on donnera pour les directeurs des agences pour gérer et contrôler les utilisateurs de système situant dans son agence.

La DMP et ses informaticiens auront un rôle en plus identique au rôle d'un administrateur dans notre système, ils pourront gérer les employés utilisant le système de n'importe quelle agence ainsi que gérer les agences, consulter les statistiques.

Le rapport de notre projet se compose de trois parties :

Le premier chapitre traite le contexte général du projet : Nous donnerons un aperçu général sur l'organisme d'accueil, les problèmes rencontrés au sein de cet organisme et nos solutions proposés pour atteindre l'objectif voulu.

Au chapitre 2, nous identifions les différents types d'utilisateurs qui exploiteront l'application ainsi que les différents cas d'utilisations et scénarios possibles afin de bien clarifier les fonctionnalités de notre application.

Au chapitre 3, nous présentons notre application finale, à cette phase nous expliquerons les différents scénarios possibles pendant l'interaction avec les interfaces de l'application qui seront aussi affiché dans cette partie.

Chapitre 1

1. Présentation générale de projet :

Introduction

Dans ce chapitre, nous avons présenté l'organisme d'accueil qui a ouvert ses portes pour nous accueillir au sein de sa direction centrale informatique ; et nous a permis de découvrir les problèmes rencontrés qui ont donné naissance à notre sujet de stage.

1.1. Concept de projet

1.1.1. Contexte de travail

Ce stage s'inscrit dans le cadre d'un projet de fin d'études pour l'obtention du diplôme Licence Appliquée en Système informatique de l'Université d 'Alger 1. Notre stage a été effectué au sein de la Banque extérieur d'Algérie abrégé BEA dans sa DMP situé à Hydra. Le sujet est intitulé « Système de génération et de gestion des cartes bancaires ».

1.1.2. Présentation de l'organisme d'accueil

La Banque extérieure d'Algérie (BEA) est une banque commerciale algérienne. Créeée en 1967, elle est basée à Bir Mourad Raïs en Algérie. [1]



Figure 1 Logo de la BEA

➤ **Dénomination :**

Banque extérieure d'Algérie (BEA).

Slogan : « La banque à la hauteur de vos aspirations » [1]

- **Activité :** Banque
- **Création :** Le 1er octobre 1967 (par ordonnance N°67-204)
- **Forme juridique :**
 - Société par actions (SPA)
 - Personnalité morale
 - Autonomie financière
- **Actionnaires :** Etat algérien 100 %
- **Structure :**

En 2012, la BEA compte 127 agences distribuées sur plusieurs wilayas de territoire algérien. [1]
- **Siege :**

42 rue des Frères Bouadou, 16005 Bir Mourad Raïs, Alger. [1]

1.1.3. Organigramme de la BEA

La figure ci dessous présente l'organigramme détaillé de la BEA.

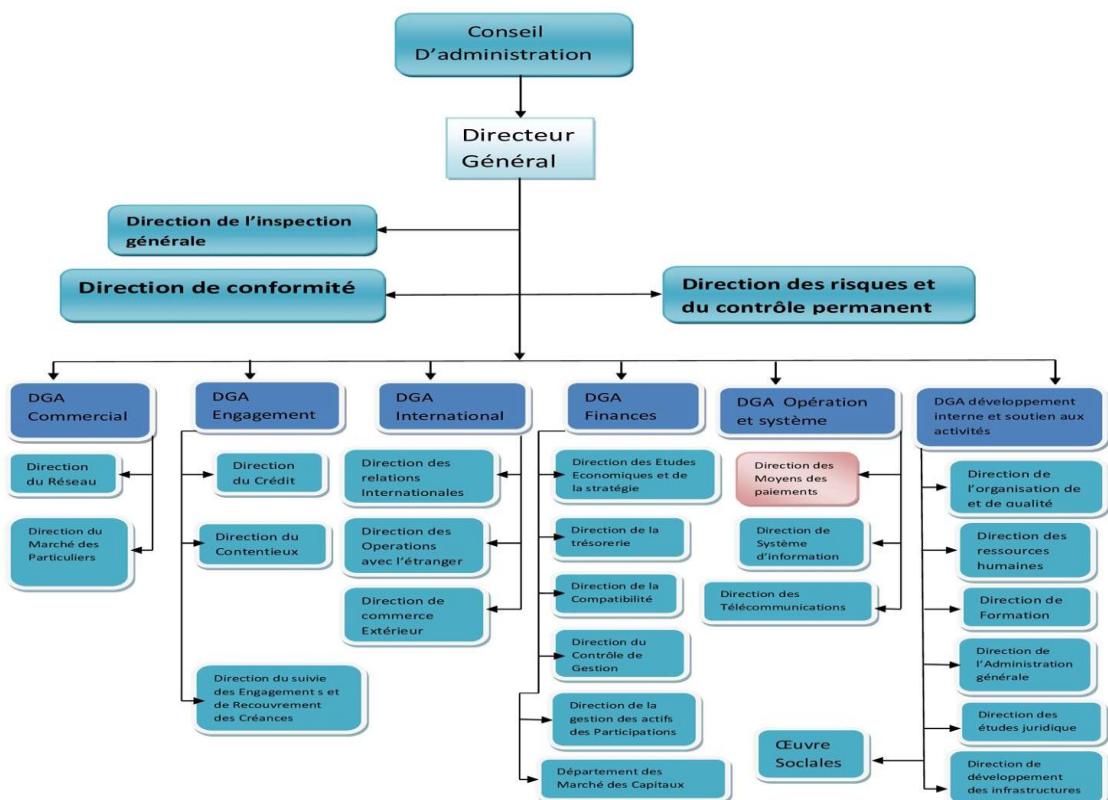


Figure 2 Organigramme de La BEA

1.1.4. La DMP et La SATIM

➤ **La DMP :**

La DMP << Direction des moyens de paiement >> de la BEA est une agence située à Hydra, dans cette agence et grâce à son équipement informatique et à ses informaticiens on traite les différentes commandes et les demandes relatives aux cartes des clients tels que les demandes d'obtention ou suspension des cartes ou bien des mises à jour des cartes etc., c'est dans cette agence que notre stage s'est déroulé.

➤ **La SATIM :**

La société d'automatisation des transactions interbancaires et de monétique est une société qui assure la fabrication et la maintenances des cartes bancaires et les transactions entre eux, cette société est en partenariat avec plusieurs entreprises et banques, parmi les membres de leur réseau monétique on trouve : La CPA, La BNA, Banque BADR, La BEA ...[2]



Figure 3 Logo de la SATIM

La BEA dépend massivement de cette société, au cas d'une anomalie par exemple les employés de la BEA doivent appeler ceux de la SATIM pour régler l'anomalie. Les agences de BEA envoient aussi chaque fois un nombre de demandes et commandes de carte à la SATIM pour les traiter.

1.2. Etude de l'existant

La BEA utilise actuellement sa propre application web locale qui traite seulement les commandes des cartes bancaires, et qui prend en charge les fonctionnalités suivantes :

→ Dès réception d 'une commande (Tous les commandes reçus sont validés par les employés qui étudient les commandes), l'utilisateur de système saisie les données relatives à

la commande de la carte dans le système telles que : Nom et prénom de client, adresse de client, type de carte souhaité etc....

→ Selon la volonté de client, l'utilisateur peut modifier ou mettre à jour les informations relatives à la commande.

→ Enfin l'utilisateur grâce à une interface dédiée peut suivre l'état des commandes des différentes agences.

→ Une fois un quota des commandes est atteint, un employé archivera tous ces demandes dans un dossier qui va être envoyé à la SATIM pour générer la carte associée à la commande.

→ Si un client perd sa carte ou bien souhaite faire des modifications relatives à sa propre carte (augmentation plafond, changer le type de la carte...), il doit soit contacter la SATIM ou passer par son agence BEA qui à son tour contactera la SATIM.

Comme le domaine des banques est très confidentiel, on a constaté uniquement que certaines banques locales ont réduit leurs pourcentages de dépendance à la SATIM qui contrôle le marché au fil de ces dernières années.

Pour l'échelle mondial, les deux gigantesques sociétés sont VISA et MASTERCARD, ces derniers possèdent un système de gestion des cartes bancaires et les transactions puissantes au point où les autres sociétés novices doivent demander la certification chez eux pour assurer le succès de leurs sociétés (la SATIM a été certifié par VISA et MASTERCARD fin 2017 par conséquent la BEA aussi).



Figure 4 Logo de L'entreprise mondial VISA



Figure 5 Logo de l'entreprise mondial MASTERCARD

1.3. Problématique

Pour arriver à mettre en place un système d'information, il est indispensable de faire une étude complète pour les processus de travail existants dans l'établissement.

Cette étude doit détecter les insuffisances et les difficultés qui peuvent mener à des défaillances dans le processus du travail et toucher à la qualité de la gestion des ressources d'informations.

Alors la réalisation du projet se basera sur les défaillances qui existent dans la méthode de travail suivie.

Durant notre stage dans la DMP de la banque extérieure d'Algérie nous avons constaté que l'entreprise avec le système existant subit au cours de ces dernières années un ralentissement en ce qui concerne le processus de gestion des cartes bancaires et le traitement des commandes des cartes.

Parmi les difficultés détectées dans l'existant, on peut citer :

- Manque d'une application qui facilite la création et génération des cartes bancaires avec leurs mises à jour.
- L'entreprise dépend d'autre entreprise pour la création et la gestion des cartes.
- Le délai de prise en charge des commandes des cartes bancaires est lent.
- Quelques ressources d'information sont sous forme de supports papier ; difficulté dans la recherche, l'archivage et le suivi.
- La charge du travail devienne très importante vu l'absence d'un vrai système qui centralise les informations et qui ne dépend pas d'autre société.

1.4.Objectifs et solutions proposés

1.4.1. Objectifs :

L'objectif principale de notre étude consiste à accélérer le processus de la gestion et la génération des cartes bancaires en réalisant un système qui permettra à n'importe quel agence BEA de traiter les différents tâches sur les cartes de clients sans passer par la SATIM (Ex : si un client souhaite changer sa carte BEA classique vers un autre type ou bien souhaite augmenter le plafond de sa carte ou même bloquer sa carte il a qu'à se présenter à l 'agence BEA la plus proche de lui et sa demande sera réglé sans passer par la SATIM) , toute fois quelques besoins clientèles tels que les nouveaux commandes Ola génération des cartes bancaires) ou bien la mise à jours de carte (récréation de carte) seront sauvegardés dans des fichiers textes et envoyés à la SATIM qui possède le matérielle nécessaire pour les traiter en dur.

- La BEA va gérer et générer les cartes bancaires en interne.
- La SATIM va s'occuper de la création ou mise à jour des cartes en dur.
- La SATIM va assurer les transactions interbancaires.

Pour atteindre cet objectif principal d'autres objectifs secondaires sont visés :

- Centraliser les informations sous un seul système.
- Minimiser les traitements des supports en papier.
- Faciliter l'accès aux documents mis à la disposition de l'employé.
- Assurer un niveau de sécurité lors du traitement des informations.

1.4.2. Solutions proposés :

Pour atteindre les objectifs mentionnés précédemment, on est mené à réaliser une application qui peut réaliser ces objectifs, et ceci en offrant :

- Une application Back-Office permettant d'effectuer l'ensemble des traitements disponibles dans le cycle de vie de processus de gestion des cartes bancaire (De l'ajout d'une commande jusqu'à la génération de carte ou de la mise à jour de la carte et la génération de fichier texte qui va être envoyé à la SATIM).
- Séparer les utilisateurs de cette application selon leurs tâches (voir le chapitre concernant).
- Une base de données centralisées, dans laquelle on pourra sauvegarder et traiter les informations.
- Une base d'informations accessibles à tous les futurs utilisateurs du système, en garantissant un niveau de sécurité et confidentialité pour les informations personnelles et les informations propres aux clients.

1.5. Conclusion

Ce chapitre a été consacré à la présentation de banque extérieure d'Algérie (*BEA*), son historique, ses objectifs, Les lacunes qu'elle rencontre, ainsi que les solutions que nous proposons pour pallier ces problèmes.

Dans le chapitre suivant nous allons aborder l'étude conceptuelle de notre application, en mentionnant tous les scénarios possibles, les acteurs, les diagrammes ...

Chapitre 2

2.Analyse et conception

Introduction

Avant de commencer à coder l'application, il est nécessaire de s'intéresser à la phase de spécification pour bien clarifier les grandes fonctionnalités de notre application.

Dans ce chapitre, nous citons d'abord la méthode d'analyse et de conception choisie, le choix de cette méthode sera justifié également, à la suite nous procéderons à l'identification de toutes les fonctionnalités de notre système pour chaque type d'utilisateur identifié et ceci en recensant les besoins fonctionnels et d'appréhender la liste des exigences traduites par les besoins non fonctionnels, nous modélisons ainsi ces besoins par les diagrammes appropriés.

Enfin, nous étudierons l'architecture logicielle et le modèle logique de l'application illustré par le diagramme de classe.

2.1.Méthode d'analyse et de conception

Une méthode d'analyse et de conception est un procédé qui permet de formaliser les principales étapes du développement d'un système afin d'augmenter sa fidélité aux besoins des clients. Pour ce faire, on part d'un énoncé informel (le besoin exprimé par le client au départ, complété par des recherches et des collectes d'informations auprès des experts du domaine fonctionnel, comme les futurs utilisateurs d'un logiciel), ainsi que de l'analyse de l'existant éventuel (la manière dont les processus à traiter par le système se déroulent chez les clients). La phase d'analyse permet de lister les résultats espérés et attendus, en termes de fonctionnalités, de performance, de robustesse, de maintenance, de sécurité et d'extensibilité, etc. La phase de conception permet de décrire le fonctionnement futur du système, afin d'en faciliter la réalisation. [3]

Méthode choisie (UP) :

Le processus uniifié (PU), ou « unified process (UP) » en anglais, ou « Unified Software Development Process (USDP) » est une famille de méthodes de développement de logiciels orientés objets. Elle se caractérise par une démarche itérative et incrémentale, pilotée par les cas d'utilisation, et centrée sur l'architecture et les modèles UML. Elle définit un processus intégrant toutes les activités de conception et de réalisation au sein de cycles de développement composés d'une phase de création, d'une phase d'élaboration, d'une phase de construction et d'une phase de transition, comprenant chacune plusieurs itérations. [3]

Dans notre sujet de stage, la méthode la plus adéquate est la méthode de processus

unifie (« UP ») déjà défini précédemment et cela est du aux nombreux avantages qu'elle offre :

- Limiter les coûts, en termes de risques, aux strictes dépenses liées à une itération.
- Limiter les risques de retard de mise en mains du produit développé (identification des problèmes dès les premiers stades de développement et non en phase de test comme avec l'approche « classique »).
- Prendre en compte le fait que les besoins des utilisateurs et les exigences correspondantes ne peuvent être intégralement définis à l'avance et se dégagent peu à peu des itérations successives.
- Accélérer le rythme de développement grâce à des objectifs clairs et à court terme. [3]

2.2.Analyse et spécification des besoins

2.2.1. Acteurs de système

Les trois différents acteurs de système sont :

Employé :

Son rôle principal s'accumule dans la partie de saisie et de gestion des commandes de cartes et ses différentes demandes de mise à jour.

Superviseur :

Le superviseur en plus d'être capable de faire le travail d'un employé il peut gérer et contrôler les employés (le premier acteur) de son agence utilisant le système, cet acteur est destiné aux directeurs des agences de la BEA.

Administrateur :

C'est la personne qui possédera tout le privilège permettant la gestion et le contrôle des deux autres acteurs trouvant dans n'importe quelle agence en plus d'avoir la possibilité de gérer les agences, consulter les statistiques de n'importe quelle agence.

Pour notre stage on a décidé de réserver cet acteur pour l'équipe d'informaticiens de la DMP.

2.2.2. Besoins fonctionnels

Les besoins fonctionnels de notre application dépendent de la nature de l'acteur. Pour cela, nous avons spécifié pour chaque acteur les besoins fonctionnels qui lui sont associé.

Notre future application doit permettre les actions suivantes pour chaque utilisateur :

Acteur principal	Cas d 'utilisation
Superviseur	<ul style="list-style-type: none"> -Authentification -Consulter la liste d'employés (Les employés ayant le grade employé trouvant dans son agence uniquement) -Imprimer la liste des employés (Les employés ayant le grade employé trouvant dans son agence uniquement) -Ajouter un employé (Les employés ayant le grade employé trouvant dans son agence sont les seuls qui peuvent être ajouté) -Modifier un employé (Les employés ayant le grade employé trouvant dans son agence sont les seuls qui peuvent être modifié) -Supprimer un employé (Les employés ayant le grade employé trouvant dans son agence sont les seuls qui peuvent être supprimé) -Modifier les autorisations des employés (Les employés ayant le grade employé trouvant dans son agence uniquement) -Consulter l'historique des activités des employés (Les employés ayant le grade employé trouvant dans son agence uniquement) -Déconnexion
Administrateur	<ul style="list-style-type: none"> -Authentification - Consulter la liste d'employés (Tous les employés et superviseur utilisant le système dans n'importe quelle agence peuvent être affiché avec la possibilité de filtrer la table d'employé par le numéro de l'agence) -Imprimer la liste des employés (Tous les employés et superviseur utilisant le système dans n'importe quelle agence peuvent être affiché avec la possibilité de filtrer le fichier PDF généré contenant la liste d'employé par le numéro de l'agence). -Ajouter un employé (L'ajout peut être dans n'importe quel agence, l'employé peut avoir n'importe quelle garde). -Modifier un employé (La modification peut être dans n'importe quel agence, l'employé peut avoir le grade employé ou superviseur).

	<ul style="list-style-type: none"> -Supprimer un employé (La suppression peut être dans n'importe quel agence, l'employé peut avoir le grade employé ou superviseur). -Modifier les autorisations des employés (Les employés ayant le grade employé ou superviseur trouvant dans n'importe quelle agence). - Modifier les autorisations de supervision des superviseurs (Les employés ayant le grade superviseur trouvant dans n'importe quelle agence). -Consulter l'historique des activités des employés (Les employés ayant n'importe quel grade trouvant dans n'importe quelle agence uniquement) -Gestion des agences (Ajout, modification, suppression des agences de système. - Consulter les statistiques (général ou par agence). -Déconnexion
Employé	<ul style="list-style-type: none"> -Authentification - Ajouter une commande -Modifier une commande -Consulter les commandes des cartes -Mise à jour des et modification des cartes -Renouveler un contrat -Générer ou maître à jour les fichier texte relative aux commandes ou aux demandes de modification qui sera envoyé à la SATIM -Déconnexion

Tableau 1 Les besoins fonctionnelles des acteurs de système

2.2.3. Besoins non fonctionnels

Après avoir déterminé les besoins fonctionnels, nous présentons ci-dessous l'ensemble des contraintes non fonctionnelles à respecter pour garantir la performance du système tout en respectant les exigences de l'utilisateur.

- **Compatibilité :** L'application est compatible avec n'importe quel système d'exploitation installé sur n'importe quel ordinateur.
- **Performance et rapidité :** Notre application assurera un temps tout en répondant aux besoins d'utilisateur.
- **La simplicité :** Chaque utilisateur (acteur) avec un niveau informatique pourra utiliser cette application facilement.
- **L'ergonomie de l'interface :** Les interfaces sont simples et conviviales.
- **La modularité de l'application :** L'application a un code simple, facile à maintenir et à comprendre en cas de besoin.
- **La confidentialité des données :** Les données des utilisateurs et des clients sont protégées et sont accessibles que pour les personnes possédant les autorisations nécessaires.

2.3.Modélisation des besoins

Langage de modélisation (UML) :

UML (Unified Modeling Language, « langage de modélisation unifié ») est un langage graphique de modélisation objet des données et des traitements. Elle est héritée de plusieurs autres méthodes telles que OMT (Object Modeling Technique) et OOSE (Object Oriented Software Engineering) et Booch. Les principaux auteurs de la notation UML sont Grady Booch, Ivar Jacobson et Jim Rumbaugh.). Dans UML, il existe plusieurs formalismes ou « modèles » : le modèle des classes, le modèle des états, le modèle des cas d'utilisation, le modèle d'interaction, le modèle de réalisation, le modèle de déploiement. [4]

2.3.1. Les diagrammes des cas d'utilisations

Définition :

Les diagrammes de cas d'utilisation sont des diagrammes UML utilisés pour clarifier le comportement fonctionnel d'un système logiciel. Ils sont utiles pour des présentations auprès de la direction ou des acteurs d'un projet, mais pour le développement, les cas d'utilisation sont plus sollicités. Un cas d'utilisation représente un ensemble de séquences d'actions qui sont réalisés par le système et qui produisent un résultat observable intéressant pour un acteur particulier. [5]

La figure 6 présente le cas d'utilisation qui correspond à l'authentification.

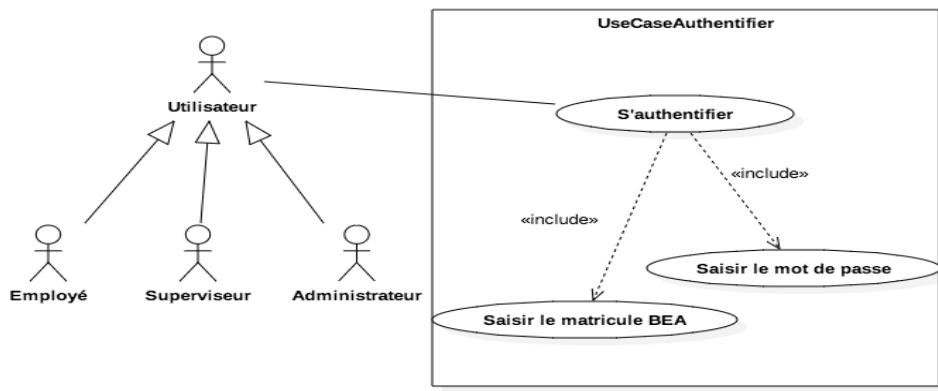


Figure 6 Diagramme cas d'utilisation << Authentification >>

Le tableau 2 ci dessous présente la description textuelle qui correspond au cas d'utilisation d'authentification.

Titre	S'authentifier
Acteur(s)	Employé, Superviseur et l'Administraticeur
But	Authentification et redirection vers le menu principal de l'acteur
Résumé	L'acteur introduit son matricule BEA et son mot de passe pour accéder au système
Pré conditions	L'acteur non authentifié ouvre l'application
Scénario normal	1-L'acteur non authentifié ouvre l'application. 2-Le système affiche un formulaire d'authentification. 3-l'acteur remplit le formulaire avec ses identifiants. 4-Le système vérifie les identifiants. 5-Le système affiche un message de succès et redirige l'acteur à son menu
Scénario échéant	Le système affiche un message d'erreur si les identifiants sont incorrects ou ne sont pas saisis

Tableau 2 Description textuelle cas d'utilisation << Authentification >>

Le superviseur est l'un des trois acteurs de système il peut ajouter, modifier et supprimer les employés (qui ont le grade d'employé) affiliés à son agence uniquement, il peut

consulter l'historique des activités et modifier les autorisations de ces derniers, le superviseur peut également accéder à l'interface de saisie comme tous employés (voir dans le diagramme d'utilisation général).

L'administrateur est l'acteur qui a tous les droits dans le système, cet acteur est destiné aux informaticiens de la direction des moyens de paiement situé à Hydra, l'administrateur peut ajouter, modifier supprimer les employés possédant le grade : employé ou superviseur (administrateur uniquement pour l'ajout) trouvant dans n'importe quelle agence il consulter l'historique des activités de ces derniers et modifier leurs autorisations, l'administrateur a le droit de ajouter, modifier, supprimer les agences dans le système et de consulter les statistiques. Tout comme le superviseur il peut accéder à tout moment aux interfaces de saisie et des gestions de commandes et cartes comme tout employé.

N.B : Les employés qui sont visibles pour le superviseur (c'est à dire les employés qui peuvent être consultés, modifiés, supprimés, consultés leurs historiques) sont ceux qui ont un grade inférieur (grade d'employé) à lui et qui appartiennent à son agence, la seule exception c'est lors de l'ajout, il peut ajouter des superviseurs ainsi que des employés. Les employés qui sont visibles pour les administrateurs sont ceux qui ont un grade inférieur à lui (les employés possédant le grade employé ou superviseur), la seule exception c'est lors de l'ajout il peut ajouter des administrateurs. Les différents diagrammes des cas d'utilisation des trois acteurs ainsi que leurs principales descriptions textuelles sont :

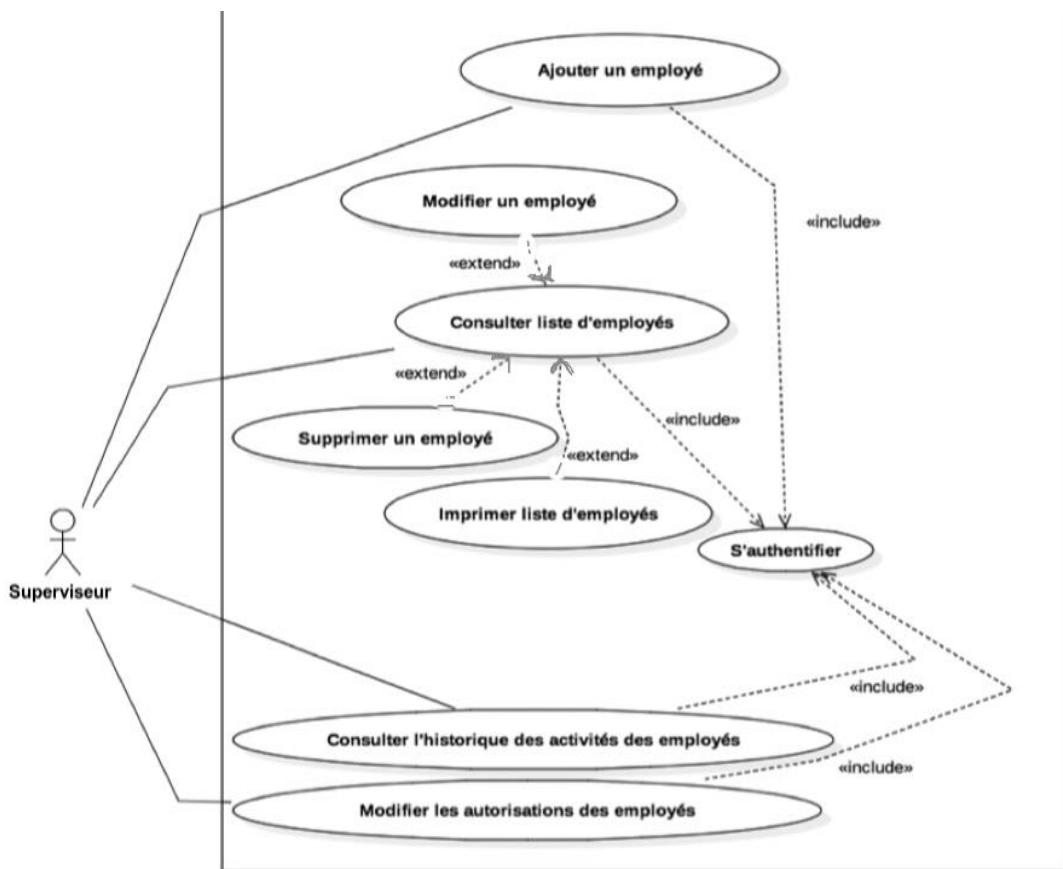


Figure 7 Diagramme de cas d'utilisation acteur << Superviseur >>

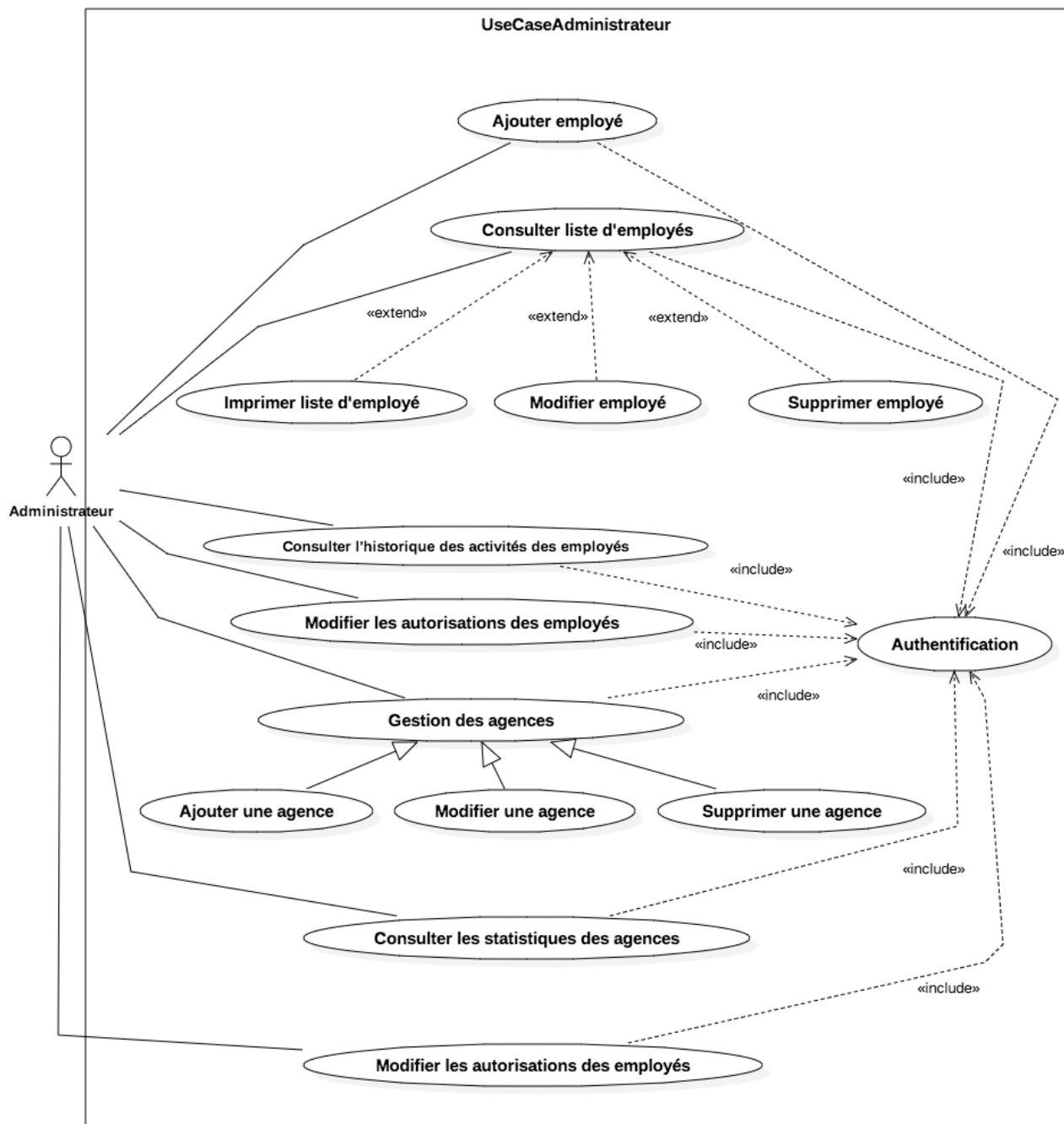


Figure 8 Diagramme cas d'utilisation acteur <> Administrateur >>

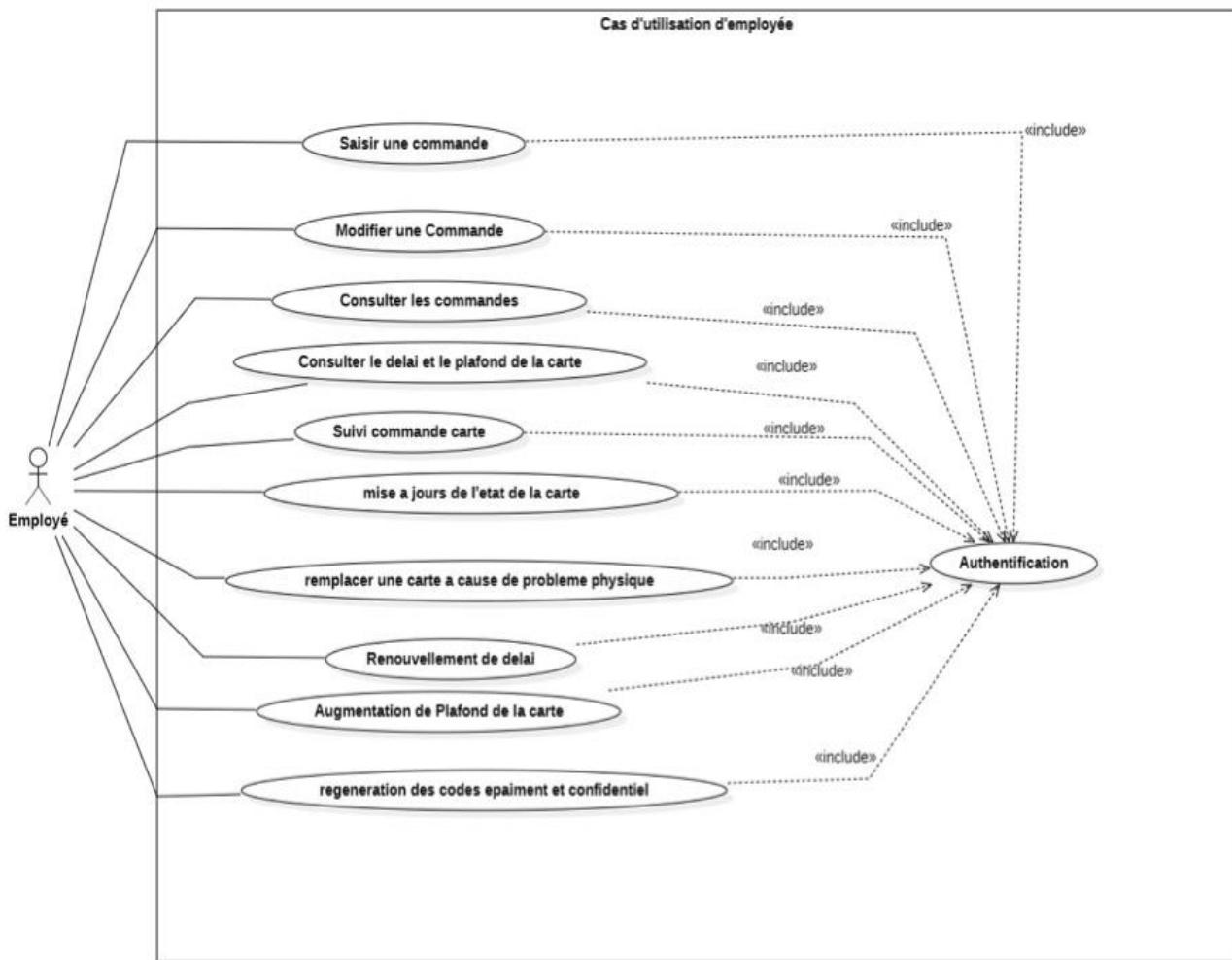


Figure 9 Diagramme cas d'utilisation acteur << Employé >>

Les principales descriptions textuelles :

Les tableaux ci dessous présentent les descriptions textuelles des principaux diagrammes des cas d'utilisation.

Titre	Ajouter un employé
Acteur(s)	Superviseur et l'Administrateur
But	Ajouter un utilisateur dans le système

Résumé	L'acteur introduit les données relatives à l'employé qu'il souhaite ajouter dans le et valide l'ajout
Pré conditions	L'acteur authentifié clique sur le bouton qui redirige vers l'interface d'ajout d'employé depuis son menu
Scénario normal	<p>1-L'acteur clique sur le bouton qui redirige vers l'interface d'ajout d'employé.</p> <p>2-Le système affiche un formulaire pour saisir les données de l'employé ce formulaire contiendra les champs suivants : le matricule BEA de l'employé , le mot de passe choisi par l'employé , le nom , prénom de l'employé et son adresse et sa date de naissance , le numéro téléphonique et son email et le type de contrat qu'il a signé et pour les champs grade et numéro d'agence d'employé, le système offre la possibilité de choisir un numéro d'agence parmi les agences disponibles dans le système et un grade (Employé , Superviseur , Administrateur) pour l'employé si l'acteur est un administrateur sinon si c'est un superviseur il choisira par défaut le numéro de l'agence à qu'il est affilié ce superviseur et choisira par défaut un grade inférieur à lui (Employé).</p> <p>3-l'acteur remplit le formulaire avec ses les données.</p> <p>4-Le système vérifie les données.</p> <p>5-Le système demande à l'acteur de confirmer l'ajout</p> <p>6-Le système affiche un message de succès et demande de rediriger l'acteur à l'interface de gestion des autorisations pour fixer les autorisations de l'employé ajouté, si l'acteur accepte il sera redirigé vers cet interface, s'il refuse il sera redirigé vers son menu principal et l'employé possédera par défaut aucune autorisations.</p>
Scénario échéant	Le système affiche un message d'erreur si les données ne sont pas saisies ou si l'employé existe déjà dans le système

Tableau 3 Description textuelle cas d'utilisation << Ajout d'employé >>

Titre	Modifier les autorisations d'un employé.
--------------	--

Acteur(s)	Superviseur et l'Administrateur.
But	Changer les autorisations d'un employé dans le système.
Résumé	L'acteur choisi l'utilisateur à supprimer parmi les la liste des utilisateurs de système avec la possibilité d'utiliser les barres de recherche et valide la suppression.
Pré conditions	L'acteur authentifié clique sur le bouton qui redirige vers l'interface de modifications des autorisations (droits d'accès) d'employé depuis son menu et sélectionne les autorisations d'employés à modifier.
Scénario normal	<p>1-L'acteur clique sur le bouton qui redirige vers l'interface de modification d'employé.</p> <p>2-Le système affiche pour le superviseur un tableau contenant tous les employés possédant le grade d'employé affilié dans la même agence que ce superviseur ainsi que l'état de leurs autorisations, l'état d'une autorisation est présenté par une check box qui sera coché si l'employé possède cette autorisation sinon il ne sera pas coché, l'utilisateur peut utiliser la barre de recherche pour rechercher un employé dans la table affichée. Sinon pour l'administrateur le système affiche un tableau contenant les employés de tous les agences avec le grade (employé ou superviseur) ainsi que l'état de leurs autorisations qui peuvent être soit des droits d'employés soit des droits de supervision (un filtre sera fourni dans l'interface pour filtrer entre les deux types de droits ainsi qu'un filtre pour filtrer les employés par agence et une barre de recherche pour rechercher un employé directement).</p> <p>3-l'acteur coche les autorisations à ajouter ou décoche les autorisations à enlever.</p> <p>4-Le système demande à l'acteur de confirmer la modification ou les modifications</p> <p>5-Le système affiche un message de succès et redirige l'acteur vers son menu principal et</p>
Scénario échéant	Si l'acteur ne confirme pas la modification, aucun changement ne sera produit.

Tableau 4 Description textuelle cas d'utilisation << gestion des autorisations**D'employé >>**

Titre	Modifier une commande
Acteur(s)	Employé Superviseur Administrateur
But	Modifier une commande carte en cas d'erreur
Résumé	L'acteur corrige les données relatives à la commande qu'il a ajouté, et valide la modification, si l'état de la commande est Non-Traiter le système va modifier les champs erronés seulement, sinon si la commande est traitée le système régénère la carte à nouveau.
Pré conditions	L'acteur authentifié clique sur le bouton qui redirige vers l'interface de la modification de la commande carte depuis son menu
Scénario normal	<p>1-L'acteur clique sur le bouton qui redirige vers l'interface de la modification de la commande carte.</p> <p>2-Le système affiche pour l'employé un champ de recherche par numéro de contrat et un formulaire pour modifier les données de la commande ce formulaire contiendra les mêmes champs du formulaire commande carte, on grise les champs inchangeables suivant : le type de la carte (Primaire, Autoriser), le type du porteur (Nouveau porteur, Porteur existant), le numéro de la carte primaire (en cas d'une carte autoriser), et le numéro du compte du porteur.</p> <p>3-l'acteur corrige les champs erronés.</p> <p>4-Le système demande à l'acteur de confirmer les modifications</p> <p>5-Le système vérifie les données.</p> <p>6- Le système génère ou maître à jour un fichier texte « BA_date_de_jour .txt » pour les commandes carte envoyer à la SATIM.</p> <p>7-Le système affiche un message de succès.</p>
Scénario échéant	Le système affiche un message d'erreur si les données ne sont pas saisies, ou si l'acteur a commis une faute de saisie.

Tableau 5 Description textuelle cas d'utilisation << Modifier une commande >>

Titre	Mise à jour de la Carte Volée Avec réémission
Acteur(s)	Employé, Superviseur, Administrateur
But	Changer l'état de carte dans la base de données à carte perdue avec réémission, avec la régénération de nouveaux :
	<ul style="list-style-type: none"> 1- Numéro de carte 2- Code confidentiel de la carte 3- Code E paiement de la carte
Pré conditions	L'acteur authentifié clique sur le bouton approprié au cas d'utilisation dans le menu
Scénario normal	<p>1-le système affiche une interface contenant les deux types de carte possible.</p> <p>2-l'utilisateur choisit le type de carte</p> <p>3-le système affiche une interface avec un champ de numéro de carte (TextField), un bouton valider et un bouton changer type carte.</p> <p>4-l'utilisateur insère le numéro de carte et clique sur valider.</p> <p>5-l'état de carte se change à carte volée avec réémission</p> <p>6-le code confidentiel de la carte est régénéré et modifié dans la base de données.</p> <p>7-le code e-paiement de la carte est régénéré et modifié dans la base de données.</p> <p>8-e système vérifie l'existence du fichier texte créé aujourd'hui.</p> <p>8-a- si le fichier n'est pas encore créé : le système génère un fichier texte sous le nom de (BC+dated'aujourd'hui.txt) avec les informations de l'action faite.</p> <p>8-b- si le fichier est déjà créé il sera mis à jour.</p> <p>9-le système affiche un message de succès.</p>
Scénario échéant	<p>4-1 si le taille de numéro de carte inséré n'égale pas la taille voulue, ou bien le numéro contient des caractères non numérique un message d'erreur sera affiché par le système « vérifier bien le numéro de carte svp ».</p> <p>4-2 si le numéro n'existe pas dans la base de données, le système affiche un message d'erreur « il faut chercher l'existence de numéro de carte le système a échoué ».</p>

	4-3 si la carte existe mais son état actuel ne permet pas à se changer, un message d'erreur sera affiché par le système.
--	--

Tableau 6 Description textuel cas d'utilisation<<Mise à jour de la carte perdue avec réémission>>

2.3.2. Les diagrammes de séquences

Les diagrammes de séquences sont une représentation graphique des interactions entre les objets (le système dans le cas d'une application informatique) et ses acteurs selon un ordre chronologique dans la formulation Unified Modeling Language. [5] Le but étant de décrire comment se déroulent les actions entre les acteurs ou objets.

Les principaux diagrammes de séquences de notre système sont :

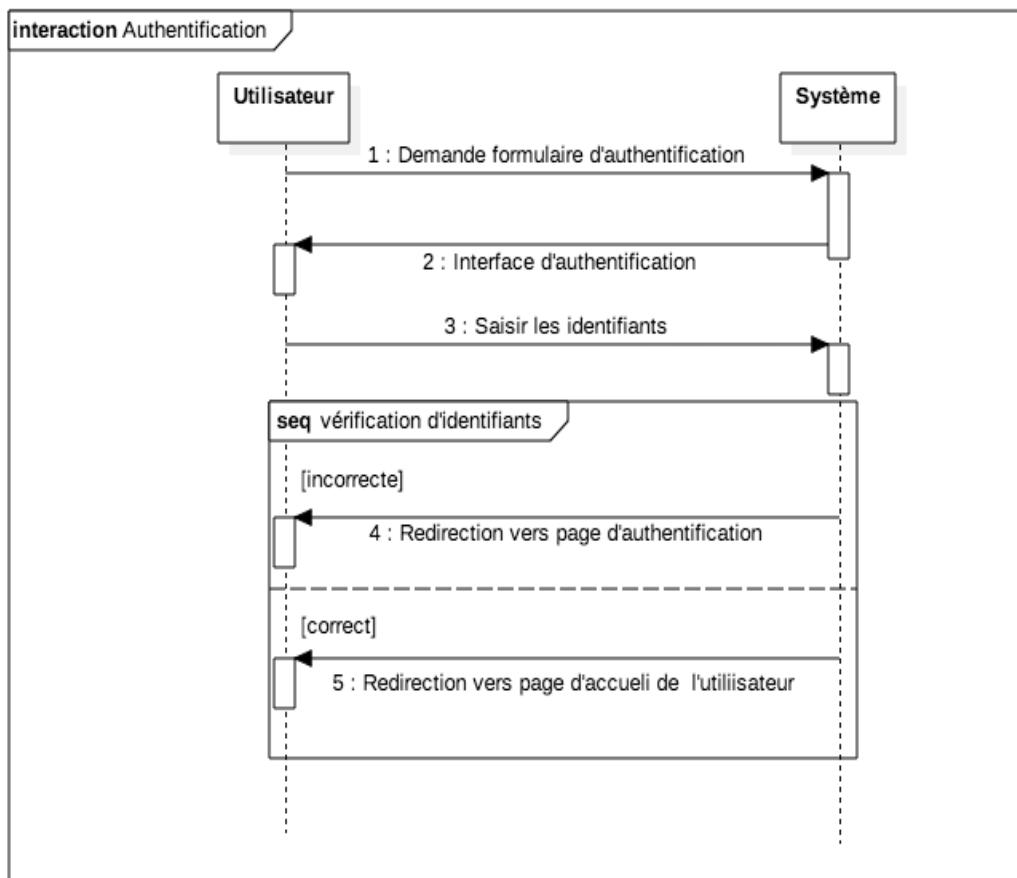
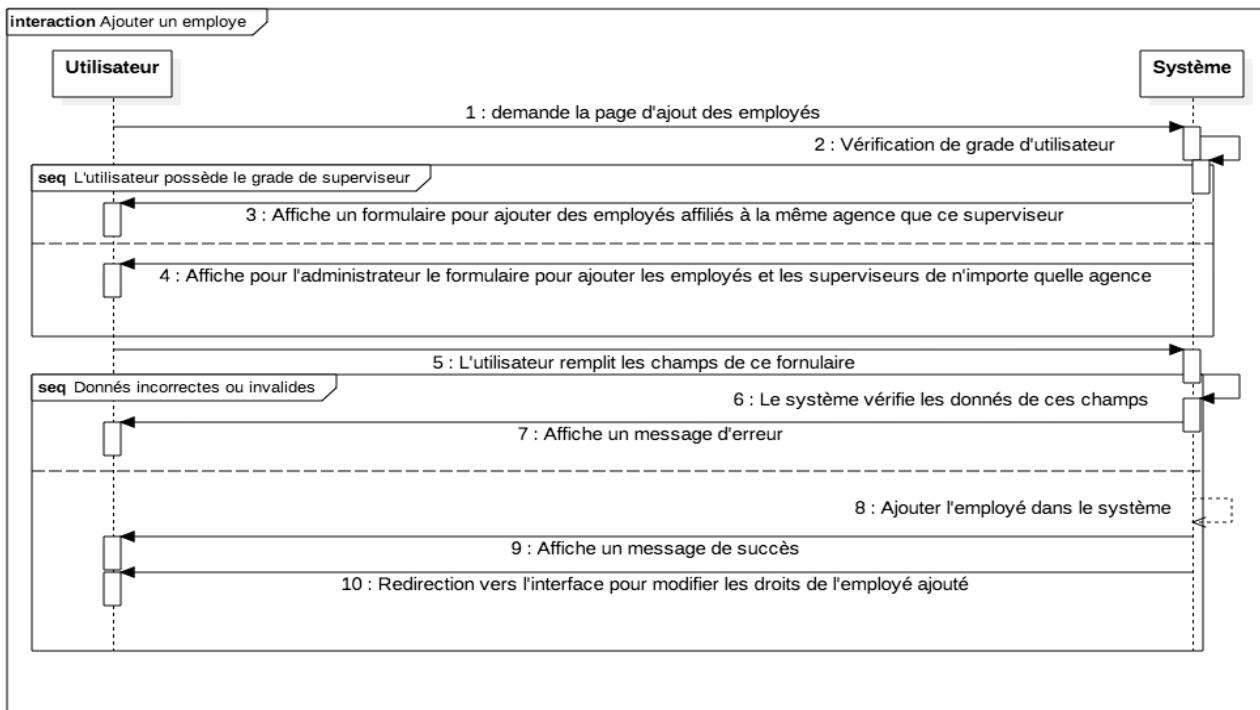
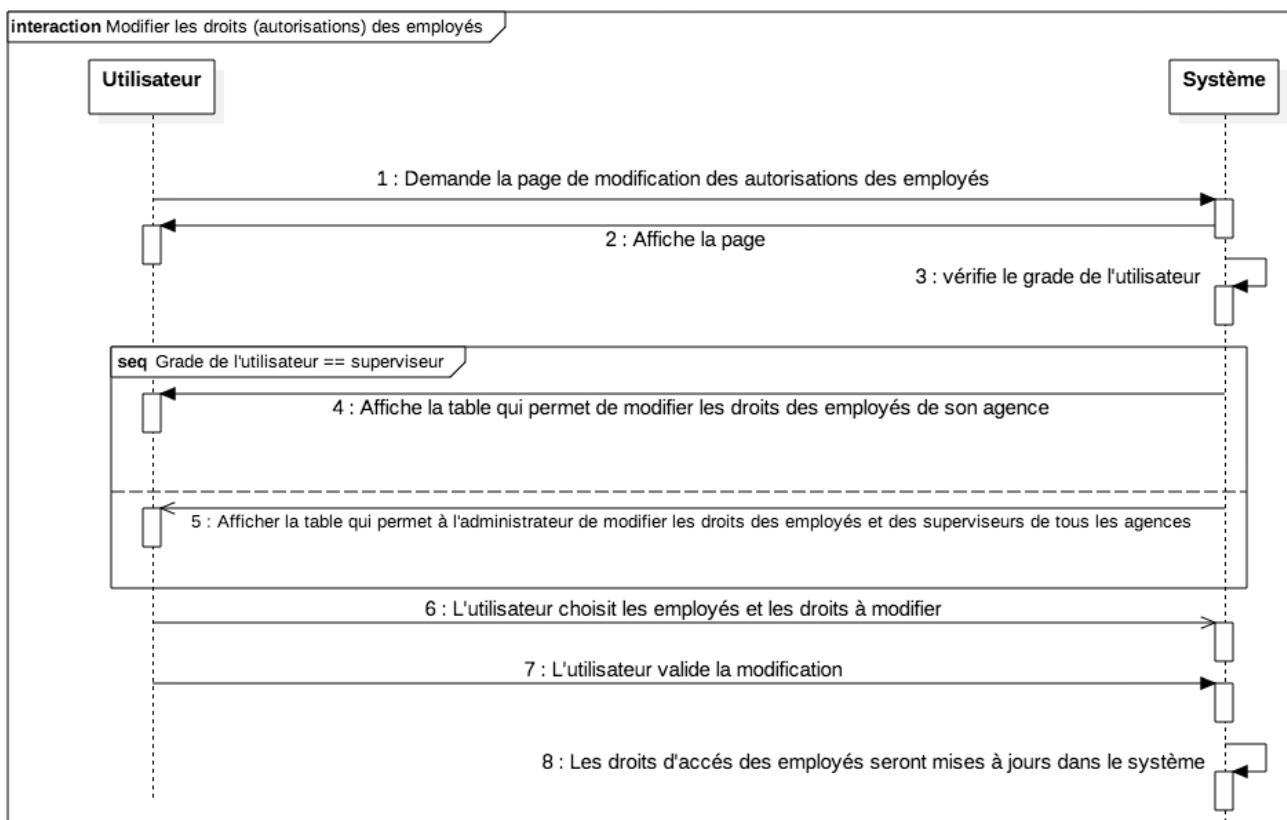


Figure 10 Diagramme de séquence << Authentication >>

**Figure 11 Diagramme de séquence << Ajouter un employé >>****Figure 12 Diagramme de séquence << Modifier les autorisations des employés >>**

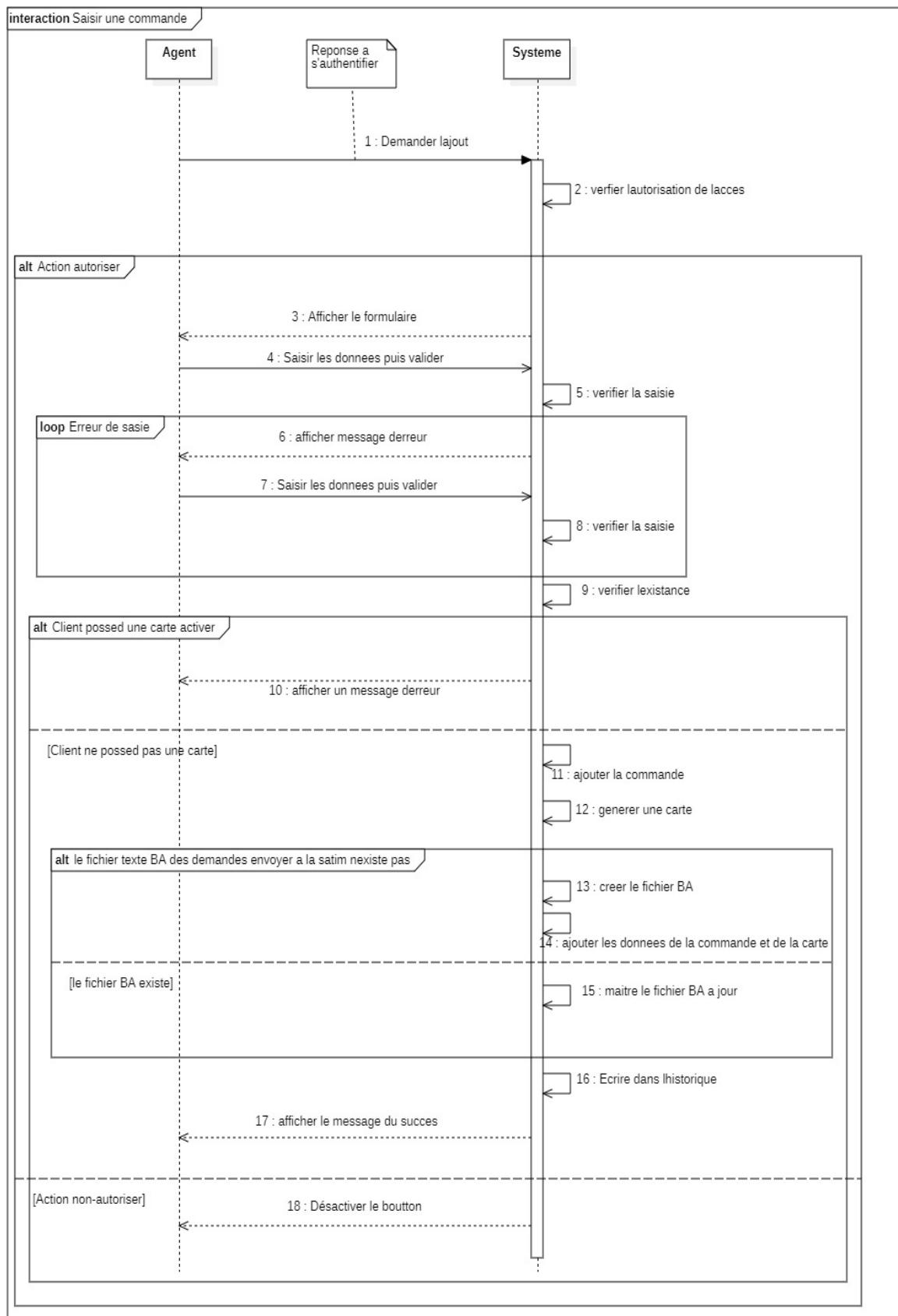


Figure 13 Diagramme de séquence << Saisir une commande >>

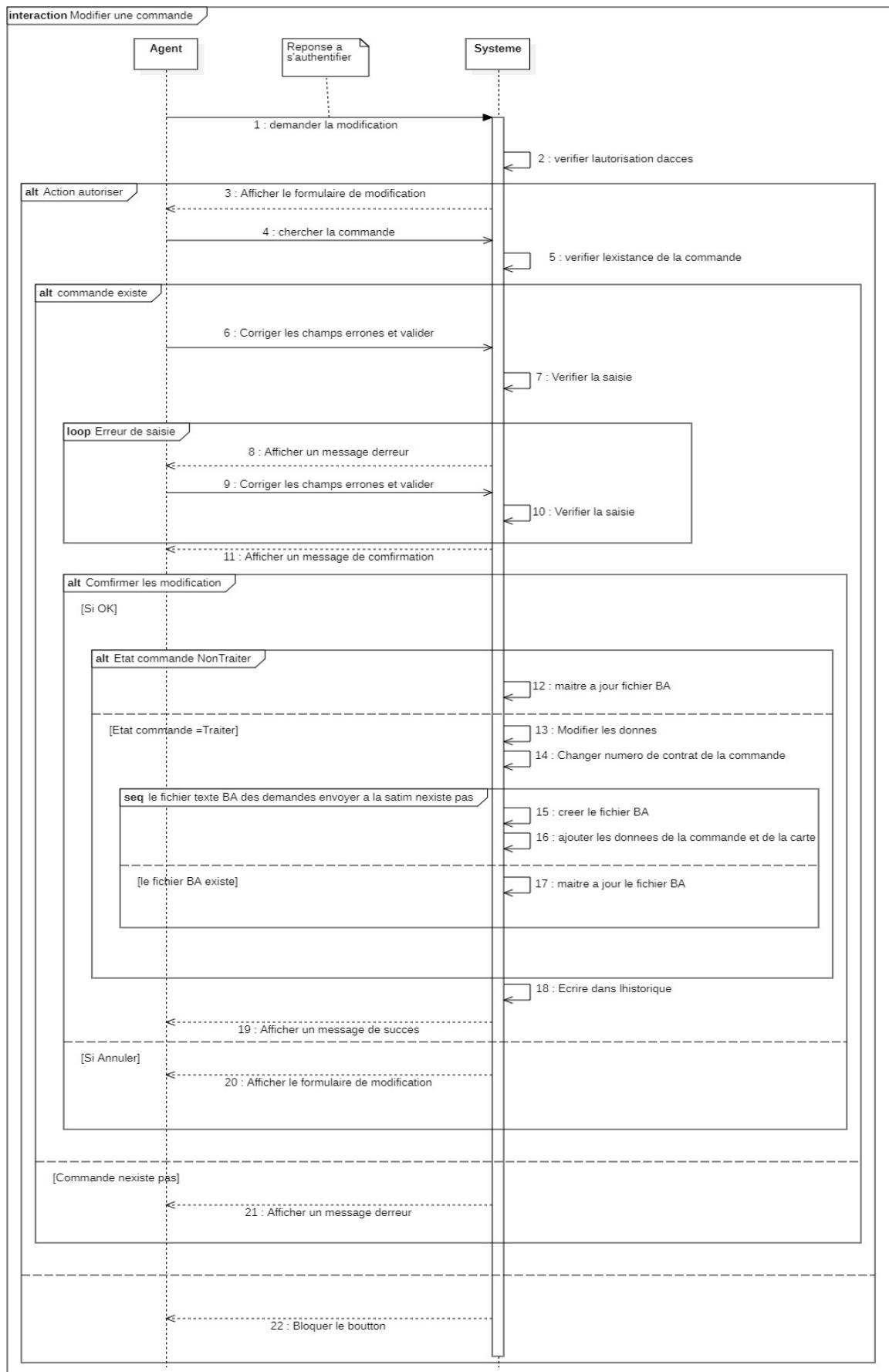


Figure 14 Diagramme de séquence << Modifier une commande >>

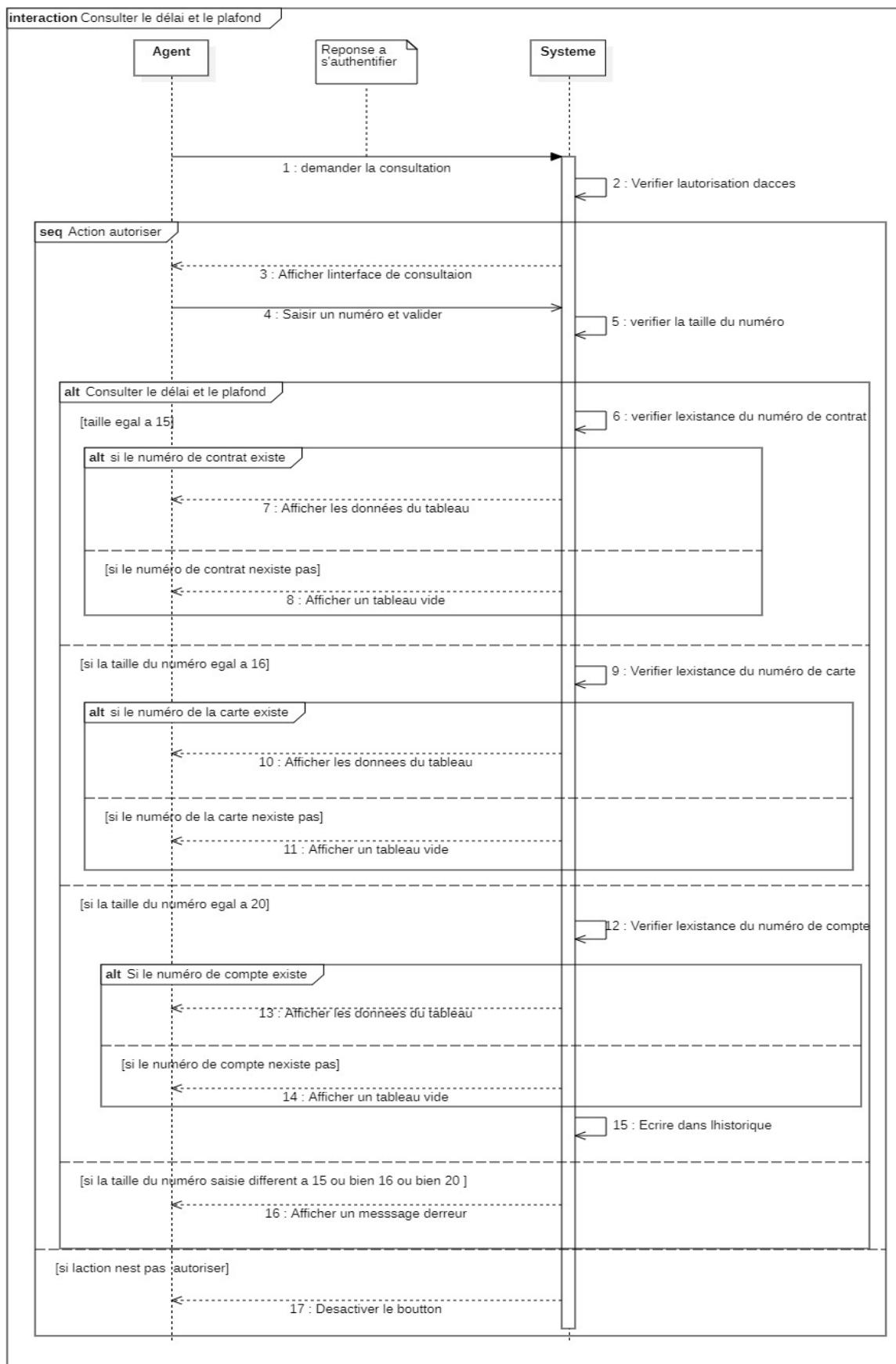


Figure 15 Diagramme de séquence << Consulter le délai et le plafond de la carte >>

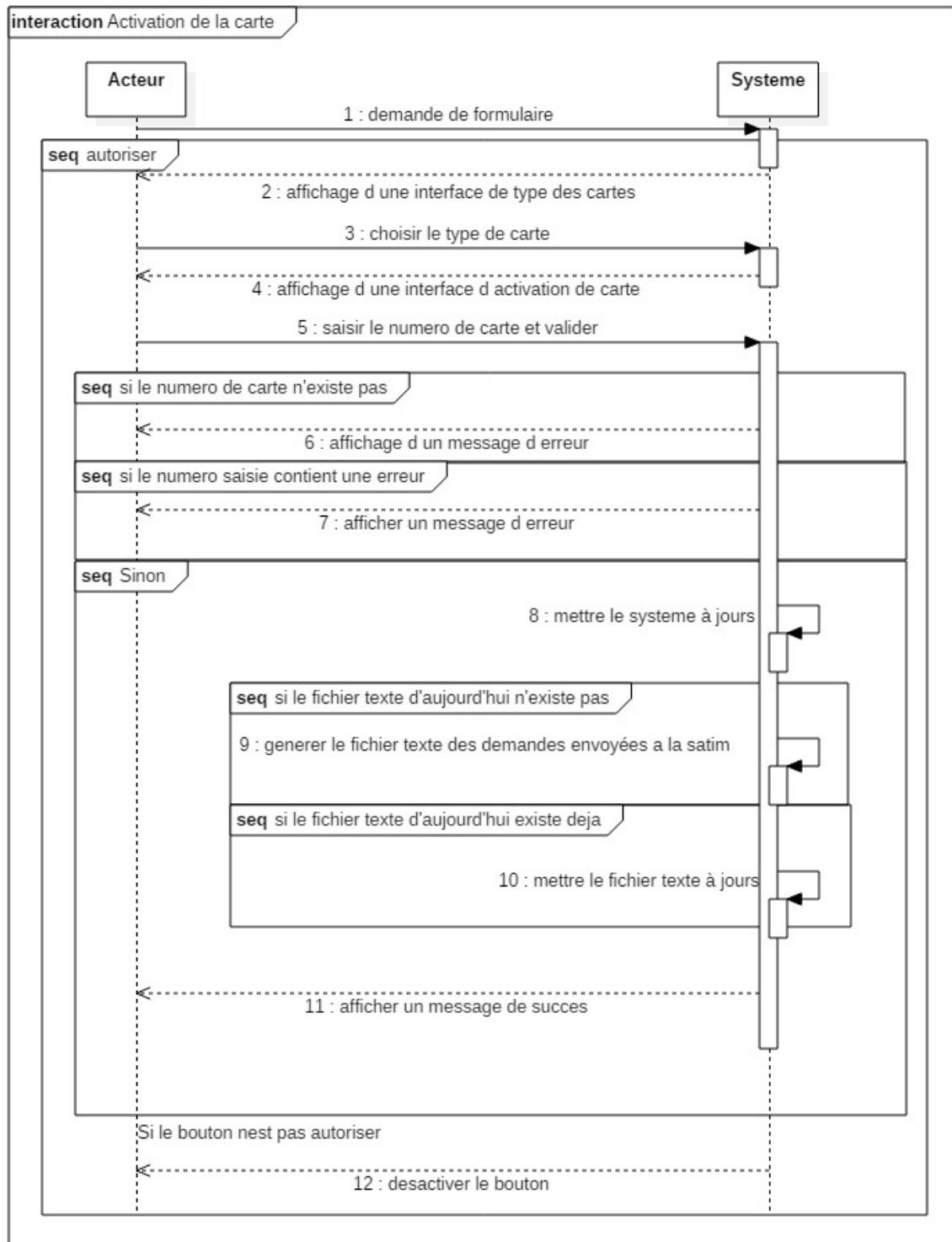


Figure 16 Diagramme de séquence << Activation de la carte >>

2.3.3. Les diagrammes d'activité

Le diagramme d'activité est un diagramme comportemental d'UML, permettant de spécifier le « comment » doit faire le système pour accomplir ses fonctionnalités. Un diagramme d'activité permet de représenter un comportement du système (sa dynamique). Ce comportement peut être celui d'un cas d'utilisation, une méthode d'une classe, un processus métier... [5]

Les principaux diagrammes d'activités de notre système sont :

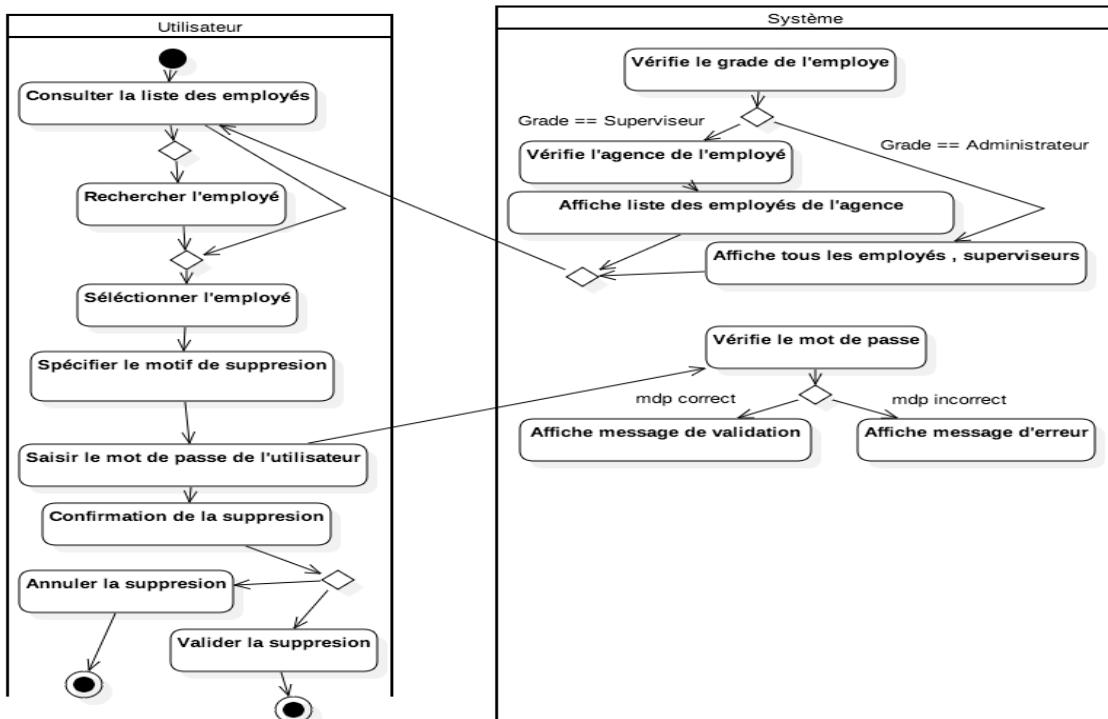


Figure 17 Diagramme d'activité << Supprimer un employé >>

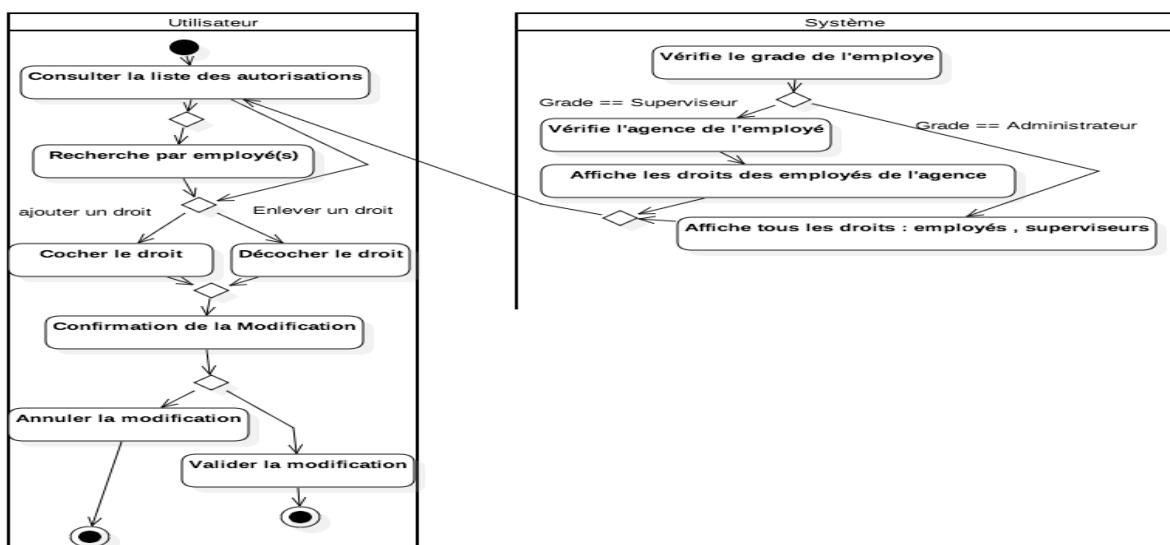


Figure 18 Diagramme d'activité << Modifier les autorisations des employés >>

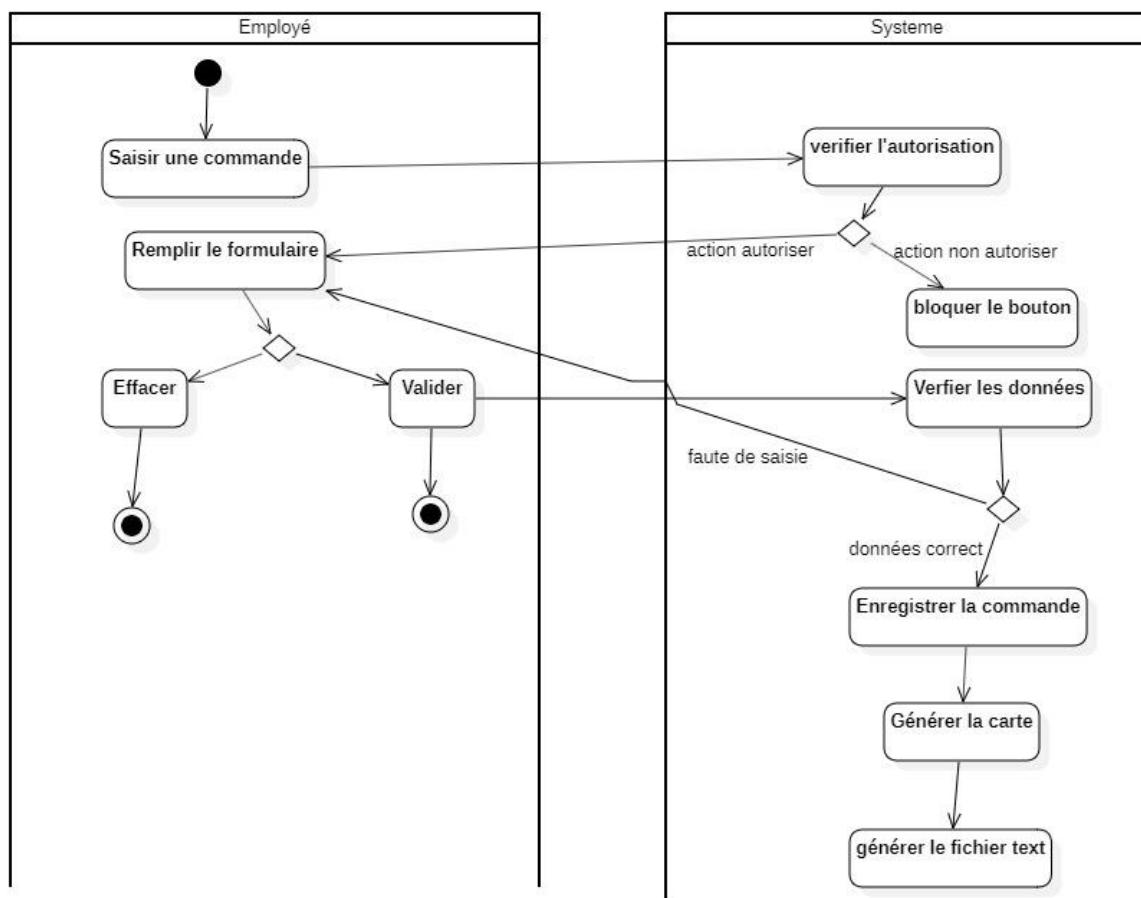


Figure 19 Diagramme des activités << Saisir commande>>

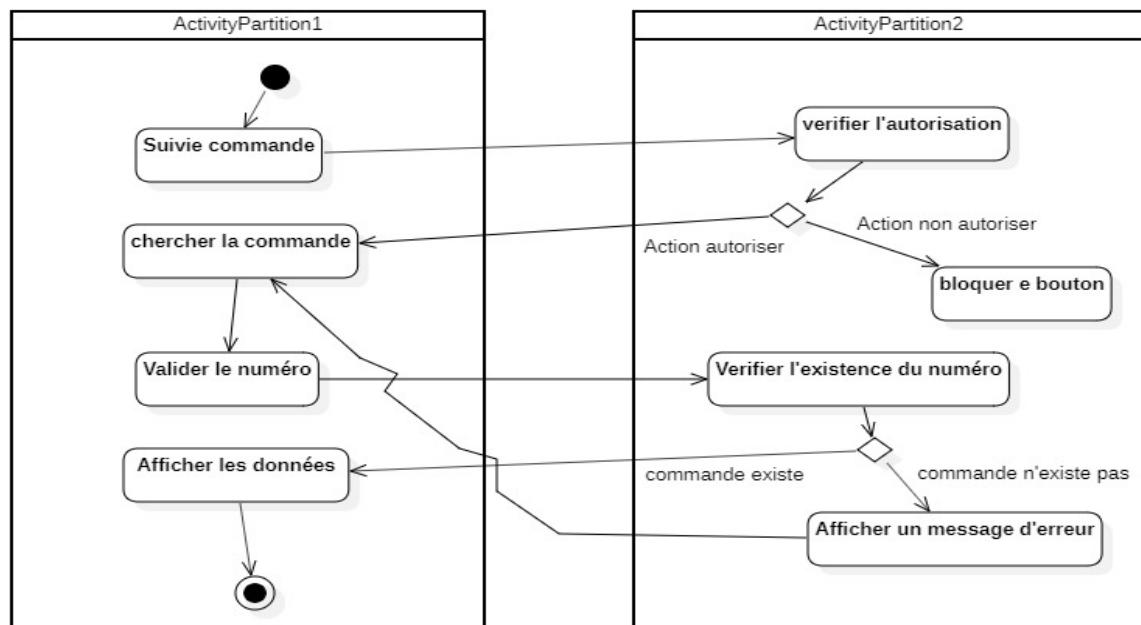


Figure 20 Diagramme des activités << Suivie commande carte>>

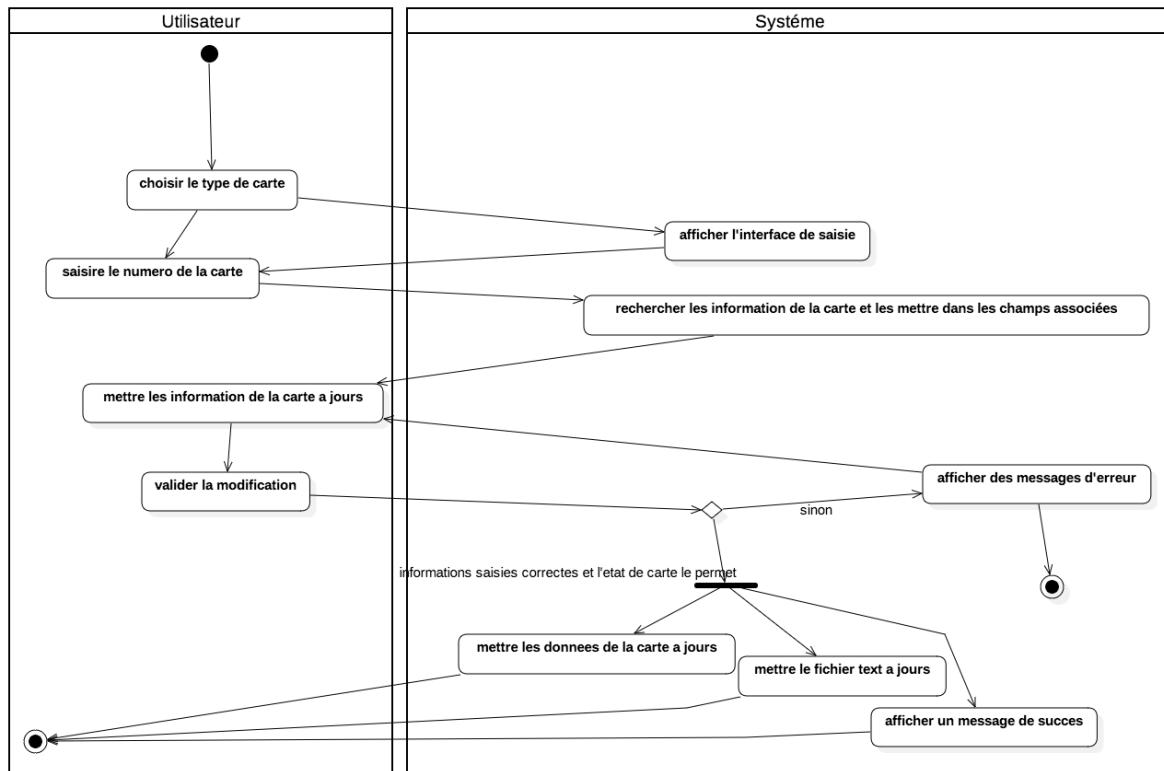


Figure 21 Diagramme des activités « Activation de la carte »

2.4. Conception détaillée

Dans cette partie nous présentons les différentes classes de système ainsi que le diagramme de classes les associations entre elles et finalement le modèle relationnel des données enregistrés dans la base de données de l'application.

2.4.1. Diagramme de classe

Les diagrammes de classes expriment de manière générale la structure statique d'un système, en termes de classes et de relations entre elles. De même qu'une classe décrit un ensemble d'objets, une association décrit un ensemble de liens ; les objets sont des instances de classes et les liens sont des instances de relations. Le diagramme de classes exprime la structure statique du système en termes de classes et de relations entre ces classes. L'intérêt du diagramme de classe est de modéliser les entités du système d'information. Le diagramme de classe permet de représenter l'ensemble des informations finalisées qui sont gérées par le domaine. [3]

Le diagramme des classes de notre système :

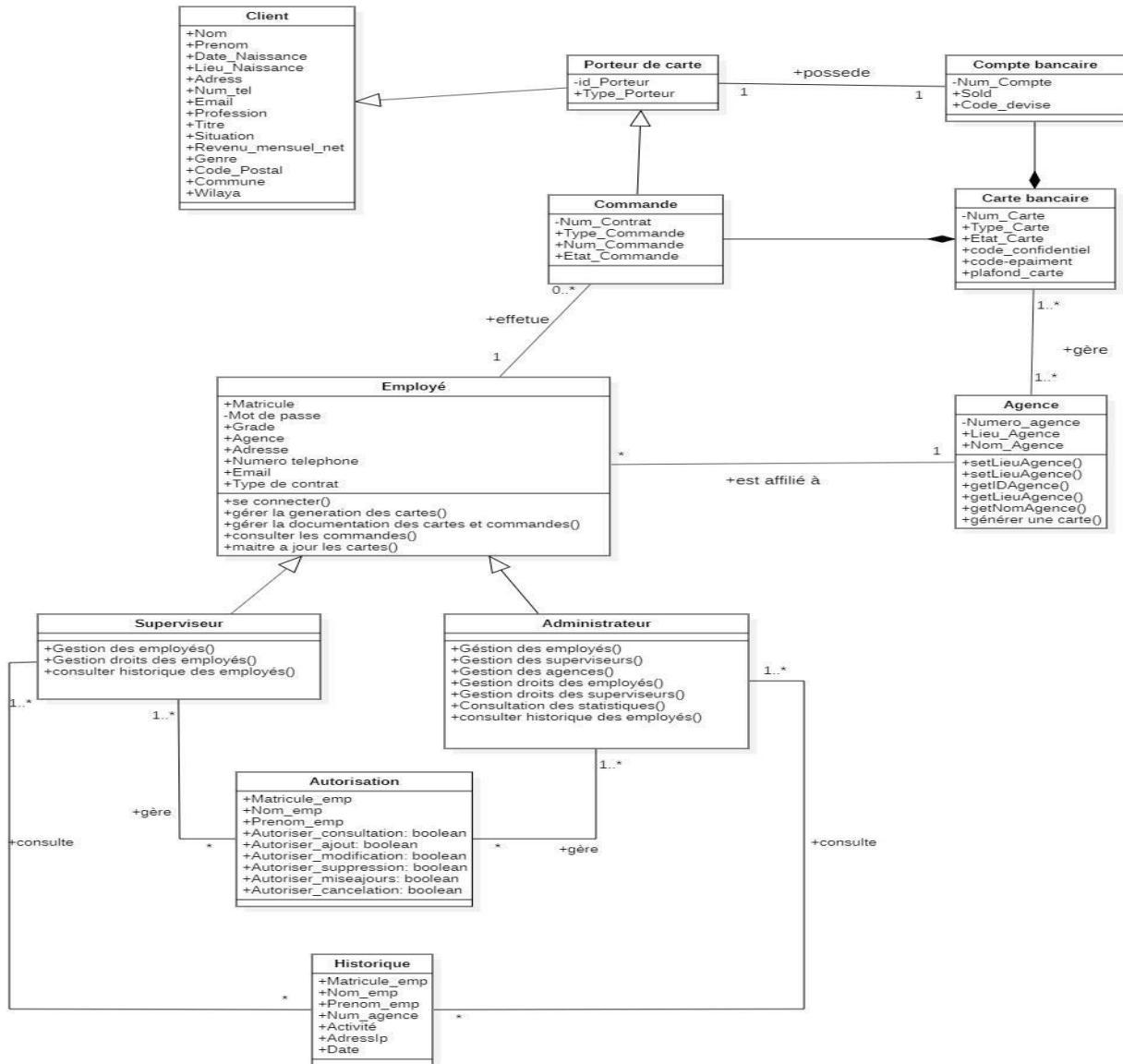


Figure 22 Diagramme de classe de système

2.4.2. Modèle relationnel

Le modèle relationnel est une manière de modéliser les relations existantes entre plusieurs informations, et de les ordonner entre elles. Cette modélisation qui repose sur des principes mathématiques mis en avant par E.F. Codd est souvent retranscrite physiquement (« implémentée ») dans une base de données. [3]

Le modèle relationnel associé au diagramme de classe de l'application :

Après l'application de règles de passage sur notre diagramme de classe, on a obtenu le modèle relationnel suivant :

-Porteur_card (id_porteur, type_porteur, nom_porteur, prenom_porteur, date_N, lieu_N, adress_Porteur, num_tele, email, profession, situation, revenuNet, genre, code_postale, commune, daira, wilaya)

-Commande (Num_Contrat, Type_commande, Etat_commande, Num_carteprimaire, #id_proteur, type_porteur, nom_porteur, prenom_porteur, date_N, lieu_N, adress_Porteur, num_tele, email, profession, situation, revenuNet, genre, code_postale, commune, daira, wilaya)

-Compte_bancaire (NumCompte, Sold)

-Carte_bancaire (NumCarte, TypeCarte, EtatCarte, codeConfidentiel, codeEpalement, plafond, #NumContrat, #NumCompte)

-Employé (Matricule, Mot_de_passe, Grade, Adresse, Num-télé, Email, TypeContrat, #Numagence)

2.5. Architectures utilisées

Dans ce titre nous abordons la définition de l'architecture technique qui consiste à faire les choix de technologies et d'organisation de composants logiciels les plus adaptés aux besoins et aux contraintes afin de résoudre les problématiques de l'organisation d'accueil (**BEA**). Ces choix sont ensuite utilisés dans notre projet, guidant la conception et permettant la transformation d'un modèle fonctionnel en application performante et robuste.

2.5.1. L'architecture logique MVC

Modèle-vue-contrôleur ou MVC est un motif d'architecture logicielle destiné aux interfaces graphiques lancé en 1978 et très populaire pour les applications web. Le motif est composé de trois types de modules ayant trois responsabilités différentes : les modèles, les vues et les contrôleurs. [6]

Un modèle (Model) contient les données à afficher.

Une vue (View) contient la présentation de l'interface graphique.

Un contrôleur (Controller) contient la logique concernant les actions effectuées par l'utilisateur.). [6]

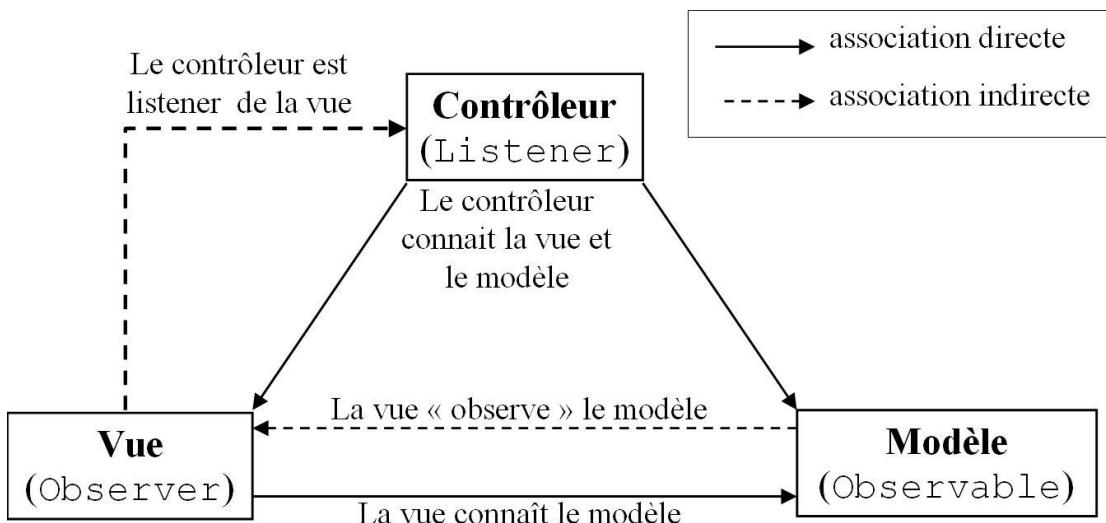


Figure 23 L'architecture MVC

L'avantage de cette architecture est la séparation de la couche interface graphique d'utilisateur (GUI) des autres parties du système (car les interfaces utilisateurs sont plus susceptibles de changer que la base de connaissances du système). [6]

2.5.2. L'architecture physique 3-tiers

L'architecture trois tiers, aussi appelée architecture à trois niveaux ou architecture à trois couches, est l'application du modèle plus général qu'est le multi-tiers. L'architecture logique du système est divisée en trois niveaux ou couches :

- Couche de présentation : la présentation des données, correspondant à l'affichage, la restitution sur le poste de travail, le dialogue avec l'utilisateur.
- Couche de traitement : le traitement métier des données, correspondant à la mise en œuvre de l'ensemble des règles de gestion et de la logique applicative.
- Couche d'accès aux données : l'accès aux données persistantes : correspondant aux données qui sont destinées à être conservées sur la durée, voire de manière définitive.

C'est une architecture basée sur l'environnement client–serveur. Dans cette approche, les couches communiquent entre elles au travers d'un « modèle d'échange », et chacune d'entre elles propose un ensemble de services rendus. Les services d'une couche sont mis à disposition de la couche supérieure. On s'interdit par conséquent qu'une couche invoque les services d'une couche plus basse que la couche immédiatement inférieure ou plus haute que la couche immédiatement supérieure (chaque couche ne communique qu'avec ses voisins immédiats). [7]

Dans notre projet on a constaté que

La partie client : c'est la partie qui englobe toutes les interfaces graphiques destinés à l'utilisateur.

La partie serveur : elle correspond à la partie fonctionnelle de l'application, celle qui implémente la logique métier, et qui décrit les opérations que l'application opère sur les données en fonction des requêtes des utilisateurs, effectuées au travers de la couche de présentation, les différentes règles de gestion et de contrôle du système sont mises en œuvre dans cette couche.

Partie interconnexion client-serveur : Elle correspond à la partie gérant l'accès aux données de l'application. Ces données peuvent être sauvegardés dans une base de données (certains sont hachés) ou bien dans des fichiers (les fichiers PDF, textes générés).

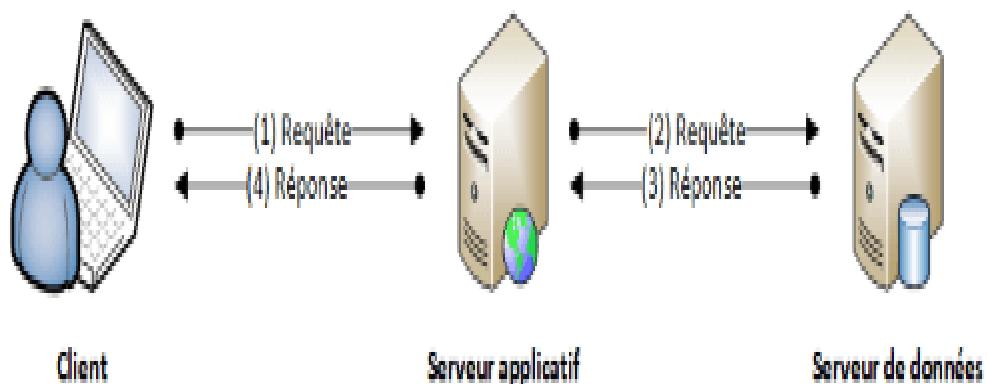


Figure 24 Architecture 3 tiers [7]

2.6.Conclusion

La partie d'analyse a permis de donner une spécification complète et précise des besoins issus du diagramme de cas d'utilisation, de les structurer sous une forme compréhensible et de définir les différents scenarios possibles, ce que facilitera la partie de réalisation.

Chapitre 3

3.Réalisation

Introduction

Dans ce chapitre nous présentons les différents outils matériels et logiciels ainsi que les différents langages de programmation utilisés durant la conception et le développement de notre application Back office.

3.1.Environnement de travail :

3.1.1. Configuration matérielle

Les différents matériels utilisés dans la réalisation de l'application sont :

Un pc portable HP (Compaq Presario CQ60-205EF), sa fiche technique :

- Processeur AMD Sempron SI-42 2.1GHz
- Mémoire vive (RAM) 2 Go DDR2-SDRAM
- Capacité de stockage principal 160 Go 5400tr/mn Serial ATA
- Carte graphique Nvidia GeForce 8200M
- Ecran 15,6 " 1366 x 768 pixels
- Système d'exploitation : Windows Vista Edition Familiale Basique

Un pc portable Apple Mac (MacBook air 2013), sa fiche technique :

- Processeur Intel Core i5-4250U 1.3GHz
- Mémoire vive (RAM) 4 Go DDR3-SDRAM
- Capacité de stockage principal 120 Go SSD Serial ATA
- Carte graphique Intel HD Graphics 5000
- Ecran 13,1 " 1440 x 900 pixels
- Système d'exploitation : Mac OS X 10.8 (Mountain Lion)

Un pc portable Acer (TravelMate P253), sa fiche technique :

- Processeur Core i3-3120M 2,5 GHz –
- Mémoire vive (RAM) 4Go
- Capacité de stockage principal 500 Go 5400tr/mn S-ATA II
- Carte graphique Intel HD Graphics 4000
- Ecran 15 "1366 x 768 pixels

- Système d'exploitation : Windows 7

Un disque dur TOSHIBA interne de capacité 500 GO

Un flash ADATA disque de taille 8 GO

Une imprimante EPSON L3310

3.1.2. Outils et langages de développement

Dans la réalisation de notre système ainsi que note application nous avons utilisés les langages et les bibliothèques (Framework) ainsi que les formats de données suivants :

- JAVA :



Figure 25 Logo de JAVA

Java est un langage de programmation orienté objet créé par James Gosling et Patrick Naughton, employés de Sun Microsystems, avec le soutien de Bill Joy (cofondateur de Sun Microsystems en 1982), présenté officiellement le 23 mai 1995 au SunWorld. La société Sun a été ensuite rachetée en 2009 par la société Oracle qui détient et maintient désormais Java. Une particularité de Java est que les logiciels écrits dans ce langage sont compilés vers une représentation binaire intermédiaire qui peut être exécutée dans une machine virtuelle Java (JVM) en faisant abstraction du système d'exploitation.

- JAVA FX :



Figure 26 Logo de JAVA FX

JavaFX est un framework et une bibliothèque d'interface utilisateur issue du projet OpenJFX, qui permet aux développeurs Java de créer une interface graphique pour des applications de bureau, des applications internet riches et des applications smartphones et tablettes tactiles. Créé à l'origine par Sun MicroSystems, puis développé par Oracle après son rachat et ce, jusqu'à la version 11 du JDK, c'est depuis lors à la communauté OpenJFX que revient la poursuite de son développement. Cette bibliothèque a été conçue pour remplacer Swing et AWT, qui ont été développés à partir de la fin des années 90, pour pallier les défauts de ces derniers et fournir de nouvelles fonctionnalités (dont le support des écrans tactiles). Le

cycle de sortie d'une nouvelle version de JavaFX correspond à celui de Java, soit tous les 6 mois.

-JDK :

Le Java Development Kit (JDK) désigne un ensemble de bibliothèques logicielles de base du langage de programmation Java, ainsi que les outils avec lesquels le code Java peut être compilé, transformé en byte code destiné à la machine virtuelle Java. Il existe plusieurs éditions de JDK, selon la plate-forme Java considérée (et bien évidemment la version de Java ciblée) :

JSE pour la Java 2 Standard Edition également désignée J2SE ;

JEE, sigle de Java Enterprise Edition également désignée J2EE ;

JME 'Micro Edition', destinée au marché mobile ; etc.

À chacune de ces plateformes correspond une base commune de Development Kits, plus des bibliothèques additionnelles spécifiques selon la plate-forme Java que le JDK cible, mais le terme de JDK est appliqué indistinctement à n'importe laquelle de ces plates-formes.



Figure 27 Logo de JDK

- FXML :

FXML est un format de données textuelles, dérivé du format XML, qui permet de décrire une interface utilisateur pour des applications conçus avec JavaFX. Il s'agit d'une alternative à la conception d'interfaces réalisées par l'écriture de lignes de codes, en découpant l'interface graphique du code source qui le contrôle. Cette technologie a été intégrée dans JavaFX à partir de sa version 2.0, pour remplacer JavaFX Script. Pour faciliter la manipulation de ce format de données, le logiciel Scène Builder a été développé par Oracle, permettant ainsi de décrire rapidement des interfaces.

- Java Database Connectivity (JDBC) :

JDBC (Java Database Connectivity) est une interface de programmation créée par Sun Microsystems — depuis racheté par Oracle Corporation — pour les programmes utilisant la plateforme Java. Elle permet aux applications Java d'accéder par le biais d'une interface commune à des sources de données pour lesquelles il existe des pilotes JDBC. Normalement, il s'agit d'une base de données relationnelle, et des pilotes JDBC sont disponibles pour tous les systèmes connus de bases de données relationnelles.

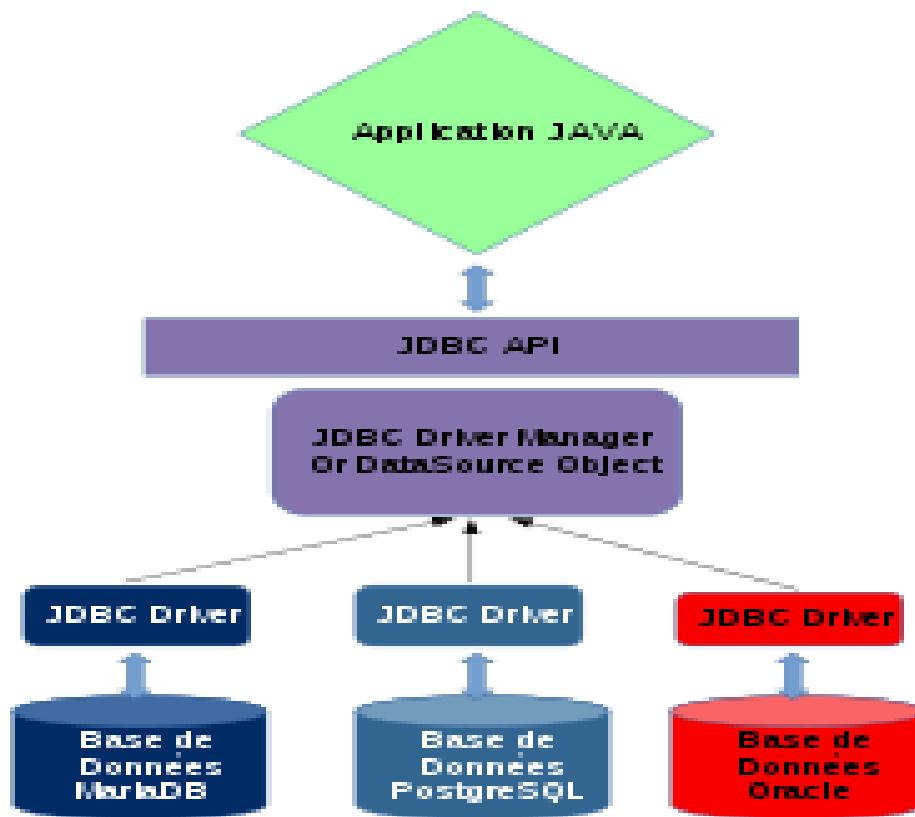


Figure 28 Fonctionnement de JDBC sur une application JAVA

- CSS:

Les feuilles de style en cascade1, généralement appelées CSS de l'anglais Cascading Style Sheets, forment un langage informatique qui décrit la présentation des documents HTML et XML. Les standards définissant CSS sont publiés par le World Wide Web Consortium (W3C). Introduit au milieu des années 1990, CSS devient couramment utilisé dans la conception de sites web et bien pris en charge par les navigateurs web dans les années 2000.



Figure 29 Logo de CSS

-Sql :

SQL (Structured Query Language, en français langage de requête structurée) est un langage informatique normalisé servant à exploiter des bases de données relationnelles. La partie langage de manipulation des données de SQL permet de rechercher, d'ajouter, de modifier ou de supprimer des données dans les bases de données relationnelles. Outre le langage de manipulation des données, la partie langage de définition des données permet de créer et de modifier l'organisation des données dans la base de données, la partie langage de contrôle de transaction permet de commencer et de terminer des transactions, et la partie langage de contrôle des données permet d'autoriser ou d'interdire l'accès à certaines données à certaines personnes. Créé en 1974, normalisé depuis 1986, le langage est reconnu par la grande majorité des systèmes de gestion de bases de données relationnelles (abrégé SGBDR) du marché.

-MySQL :

MySQL est un système de gestion de bases de données relationnelles (SGBDR). Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels. Son nom vient du prénom de la fille du corrélateur Michael Widenius, My, (SQL fait référence au Structured Query Language, le langage de requête utilisé).



Figure 30 Logo de MySQL

3.1.3. Configuration logicielle :

-StarUML :

StarUML est un logiciel de modélisation UML, qui a été "cédé comme open source" par son éditeur, à la fin de son exploitation commerciale (qui visiblement continue ...), sous une licence modifiée de GNU GPL. Aujourd'hui la version StarUML V3 n'existe qu'en licence propriétaire. StarUML gère la plupart des diagrammes spécifiés dans la norme UML 2.0. StarUML est écrit en Delphi, et dépend de composants Delphi propriétaires (non open-source). [8]



Figure 31 Logo de StarUML

-phpMyAdmin :

PhpMyAdmin (PMA) est une application Web de gestion pour les systèmes de gestion de base de données MySQL réalisée principalement en PHP et distribuée sous licence GNU GPL.

Il s'agit de l'une des plus célèbres interfaces pour gérer une base de données MySQL sur un serveur PHP. De nombreux hébergeurs, gratuits comme payants, le proposent ce qui évite à l'utilisateur d'avoir à l'installer. Cette interface pratique permet d'exécuter, très facilement et sans grandes connaissances en bases de données, des requêtes comme les créations de table de données, insertions, mises à jour, suppressions et modifications de structure de la base de données, ainsi que l'attribution et la révocation de droits et l'import/export. Ce système permet de sauvegarder commodément une base de données sous forme de fichier .SQL et d'y transférer ses données, même sans connaître SQL. Les requêtes SQL restent possibles, ce qui permet de les tester interactivement avant de les incruster dans notre code.



Figure 32 Logo de phpMyAdmin

-Wamp Server :

WampServer (anciennement WAMP5) est une plateforme de développement Web de type WAMP, permettant de faire fonctionner localement (sans avoir à se connecter à un serveur externe) des scripts PHP. WampServer n'est pas en soi un logiciel, mais un environnement comprenant trois serveurs (Apache, MySQL et MariaDB), un interpréteur de script (PHP), ainsi que phpMyAdmin pour l'administration Web des bases MySQL.



Figure 33 Logo de WampServer

-Scene builder:

Scene Builder est un outil interactif de conception d'interface graphique pour JavaFX. Créé par Oracle, il permet de construire rapidement des interfaces utilisateurs sans avoir besoin de coder. Le logiciel est décliné en deux versions : l'une destiné à JavaFX 8 et l'autre pour JavaFX 9 et plus.



Figure 34 Logo de Scene Builder 2.0

-NetBeans 8 :

NetBeans est un environnement de développement intégré, placé en open source par Sun en juin 2000 sous licence CDDL et GPLv2. En plus de Java, NetBeans permet la prise en charge native de divers langages tels le C, le C++, le JavaScript, le XML, le Groovy, le PHP et le HTML, ou d'autres par l'ajout de greffons. [9]

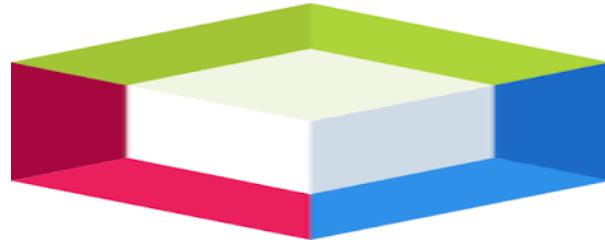


Figure 35 Logo de Netbeans 8

3.2. Description Détailé de l'application

Notre application contient plusieurs interfaces et chaque interface peut avoir plusieurs scenarios, on peut distinguer quatre types des interfaces : Interfaces dédiées uniquement aux superviseur et administrateurs, interfaces dédiées aux administrateurs uniquement, Interfaces dédiées aux agents de saisie (employés), interfaces partagées par tous les acteurs qu'on appelle interfaces générales.

3.2.1. Interfaces générales

Interface de redirection :

L'interface s'affiche pendant 3 secondes avant de rediriger l'utilisateur vers son menu ou vers la page d'authentification s'il n'est pas connecté.



Figure 36 Interface de redirection (Splash Screen)

Interface d'authentification :

L'interface permet à l'utilisateur non authentifié de se s'authentifier avant de le rediriger vers son menu principal.

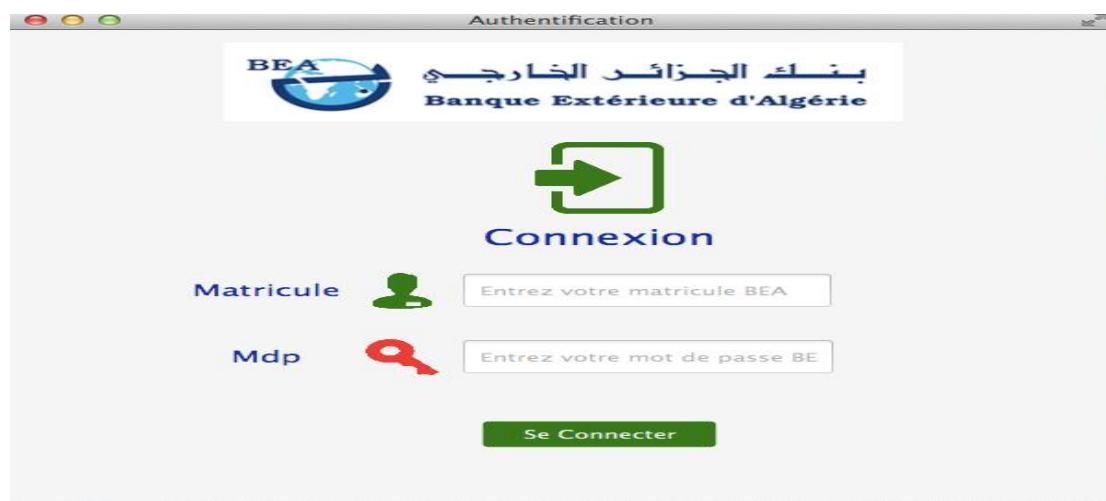


Figure 37 Interface d'authentification

3.2.2. Interfaces des administrateurs et superviseurs

Interface d'ajout d'employé :

L'interface permet à :

L'administrateur : d'ajouter un employé avec n'importe quel grade (employé, superviseur, administrateur) dans n'importe quelle agence.

Le superviseur : ajouter un employé avec le grade d'employé ou superviseur dans l'agence ou ce superviseur est affilié uniquement.

Ajout d'employé

BEA Banque Extérieure d'Algérie

Ajouter un employé

Nom Nom d'employé Prenom Prenom d'employé

Date de naissance Ex - 18/07/1999 Adresse Adresse d'employé

Matricule Matricule BEA de l'en Mdp Mot de passe choisi

Numero de mobile numero mobile de l'emp E-mail Saissez l'email de l'emp

Grade Agence

Type contrat e de contrat de l'emp

Reinitialiser Valider

Agence : 16031

Figure 38 Interface d'ajout d'employé

Interface de consultation de liste d'employés :

L'interface permet à :

L'administrateur : de consulter et imprimer la liste des employés de n'importe quelle agence

Le superviseur : de consulter et imprimer la liste des employés de son agence uniquement.

Consultation de liste d'employés

BEA Banque Extérieure d'Algérie

Liste d'employés

16031

Nom	Prenom	Agence	Grade	Matricule
Achour	Mouloud	16031	Employé	123455689
slimani	islam	16031	Employé	123455789
Benhami	Said	16031	Superviseur	123456788
Benhami	Samir	16031	Administrateur	191919199
Sadok	Montasser	16031	Administrateur	511111115
Benhami	Abdelhamid	16031	Administrateur	511111120

Afficher Imprimer

Agence : 16031

Figure 39 Interface de consultation de liste des employés

Interface de consultation de liste d'employés :

L'interface permet à :

L'administrateur : de sélectionner un employé possédant le grade employé ou superviseur parmi liste des employés de n'importe quelle agence pour modifier ses données ou le supprimer.

Le superviseur : de sélectionner un employé possédant le grade employé uniquement parmi la liste des employés de son agence uniquement.

Nom	Prenom	Agence	Grade	Matricule
Achour	Mouloud	16031	Employé	12345689
slimani	islam	16031	Employé	123455789

Modifier **Supprimer**

Figure 40 Interface de sélection d'employé acteur << Superviseur>>

Nom	Prenom	Agence	Grade	Matricule
wewew	eewwe	28000	Employé	123121221
werty	efgh	25000	Employé	123123456
Achour	Mouloud	16031	Employé	123455689
slimani	islam	16031	Employé	123455789
Mohamed	Ahmed	16001	Employé	123456543
asds	dssd	31081	Employé	123456777
mohmi	ramzi	31082	Employé	123456781
Benhami	Said	16031	Superviseur	123456788
Khebzi	Mohssen	16004	Employé	125678965
Bennacer	Saud	16001	Superviseur	145412881
Mahmudi	Oussama	0000	Superviseur	123456789

Modifier **Supprimer**

Figure 41 Interface de sélection d'employé acteur << Administrateur >>

Interface de modification d'employé :

L'interface permet à :

L'administrateur : de modifier les données d'un employé sélectionné dans l'interface de sélection, possédant le grade employé ou superviseur.

Le superviseur : de modifier les données d'un employé possédant le grade employé avec l'interdiction de modifier l'agence et le grade de cet employé.

Nom	Mohamed	Prenom	Ahmed
Adresse	boulevard mohamed	Type contrat	CDI
Numero	598235179	Email	mogaahmed@gmail.c
Service	Employé	Agence	16001

Mot de passe

Valider

Agence : 16031

Figure 42 Interface de modification d'employé acteur << Administrateur >>

Interface de suppression d'employé :

L'interface permet à :

L'administrateur : de supprimer un employé sélectionné dans l'interface de sélection, possédant le grade employé ou superviseur.

Le superviseur : de supprimer un employé sélectionné dans l'interface de sélection, possédant le grade employé uniquement.

NB : il sera impossible de modifier le matricule sélectionné depuis l'interface de sélection.

Saisissez le matricule de l'employé

Saissez votre mot de passe BEA

Spécifiez le motif

Valider

Agence : 16031

Figure 43 Interface de suppression d'employé

Interface de consultation d'historique des employés :

L’interface permet à :

L’administrateur : de consulter les activités des employés de n’importe quelle agence possédante n’importe quel grade.

Le superviseur : de consulter les activités d’un ou plusieurs employés possédant le grade employé de son agence.

Matricule	Nom	Prenom	Agence	Activité	Date / Heure	IP Local
511111120	Benhami	Abdelhamid	16031	Modification d’agence : 16003	2020/08/25 22:2...	192.168.100.3
511111120	Benhami	Abdelhamid	16031	Modification d’agence : 16003	2020/08/25 22:2...	192.168.100.3
511111120	Benhami	Abdelhamid	16031	Suppression d’employé : 897654321	2020/08/27 17:4...	192.168.100.3
511111120	Benhami	Abdelhamid	16031	Ajout d’utilisateur : 123123456	2020/08/27 18:3...	192.168.100.3
511111120	Benhami	Abdelhamid	16031	Ajout d’utilisateur : 123552212	2020/08/28 15:1...	192.168.100.3
511111120	Benhami	Abdelhamid	16031	Ajout d’utilisateur : 123121221	2020/08/28 15:1...	192.168.100.3
511111120	Benhami	Abdelhamid	16031	Ajout d’utilisateur : 1233333444	2020/08/28 15:2...	192.168.100.3
511111120	Benhami	Abdelhamid	16031	Ajout d’utilisateur : 123456777	2020/08/28 15:2...	192.168.100.3
511111120	Benhami	Abdelhamid	16031	Ajout d’utilisateur : 298328121	2020/08/31 19:5...	192.168.100.4
511111120	Benhami	Abdelhamid	16031	Ajout d’utilisateur : 234455771	2020/08/31 20:0...	192.168.100.4
511111120	Benhami	Abdelhamid	16031	Ajout d’utilisateur : 982121219	2020/08/31 20:0...	192.168.100.4
511111120	Benhami	Abdelhamid	16031	Ajout d’utilisateur : 178902333	2020/08/31 20:1...	192.168.100.4
511111120	Benhami	Abdelhamid	16031	Ajout d’utilisateur : 171723331	2020/08/31 21:5...	192.168.100.4
123456788	Benhami	Said	16031	Ajout d’utilisateur : 123455689	2020/08/31 22:1...	192.168.100.4

Figure 44 Interface de consultation d’historique des activités des employés

Interface de modification des droits des employés :

L’interface permet à :

L’administrateur : de modifier les droits d’accès des employés de n’importe quelle agence possédante n’importe quel grade, il peut aussi modifier les droits de supervision des superviseurs.

Le superviseur : de modifier les droits d’accès des employé possédant le grade employé de son agence.

Gestion des droits d'employés

Tous les agences | Banque Extérieure d'Algérie

Droit d'employé | Utilisateur

Matricule	Nom	Prenom	Autorisation sur la gestion des commandes				Autorisation sur les mises à jour des cartes et des contrats			
			Consulter	Ajouter	Modifier	Supprimer	Mise à jours de carte	Mise à jours carte perdu /vole	Activation/désactivation de carte	Canc
123456781	mohmi	ramzi	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
123456788	Benhami	Said	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
125678965	Khebzi	Mohssen	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
171723331	said	benhamid	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
178902333	Mustapha	Maddi	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
182723110	Medjdi	Larbi	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
234455771	mohamed	henni	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
234568819	El mourabet	Bachir	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Valider les changements

Agence : 16031

Figure 45 Interface de modification des droits des employés acteur <<Administrateur>>

Gestion des droits d'employés

16031 | Banque Extérieure d'Algérie

Droit d'employé | Utilisateur

Matricule	Nom	Prenom	Autorisation sur la gestion des commandes				Autorisation sur les mises à jour des cartes et des contrats			
			Consulter	Ajouter	Modifier	Supprimer	Mise à jours de carte	Mise à jours carte perdu /vole	Activation/désactivation de carte	Canc
123455689	Achour	Mouloud	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
123455789	slimani	islam	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Valider les changements

Agence : 16031

Figure 46 Figure 47 Interface de modification des droits des employés acteur <<Superviseur >>

Interface gestion des agences :

L’interface permet à l’administrateur d’ajouter ou de modifier ou supprimer une agence dans le système.

The screenshot shows the 'Gestion des agences' (Agency Management) interface. At the top right is a blue house icon. The main area has two tabs: 'Ajouter une agence' (Add a branch) on the left and 'Gestion des agences' (Management of branches) on the right. The right tab contains a search bar with placeholder text 'Tapez l'identifiant de l'agence' (Enter the branch identifier). Below the search bar is a table with columns: Num, Lib, Adresse, Codewilaya, LibWilaya, Region, Bin, and Banque. The table lists several branches, including BEA EL MAKKAH, BEA BLIDA, BEA DIDOUCH, BEA BELOUZDAD, BEA CHERAGA, BEA MOHAMMED, BEA DELY IBRAHIM, BEA EL HARRACH, BEA ROUIBA, and BEA HYDRA. The row for BEA CHERAGA is selected, highlighted in blue. At the bottom of the table are buttons for 'Ajouter' (Add), 'Modifier' (Edit), and 'Supprimer' (Delete). A status message 'Agence : 16031' is at the bottom right.

Figure 48 Interface gestion des agences

Interface des Statistiques :

L’interface permet à l’administrateur de consulter les statistiques de chaque agence

The screenshot shows the 'Statistiques' (Statistics) interface. At the top right is a blue house icon. The main area features five cards with icons and numerical values: 'Commandes approuvés' (Approved orders) with 7, 'Comptes créés' (Accounts created) with 6, 'Clients ajoutés' (New clients) with 6, 'Cartes' (Cards) with 6, and 'Cartes autorisés' (Authorized cards) with 1. Above these cards is a dropdown menu labeled 'Tous les agences' (All branches) and a small bank icon. A status message 'Agence : 16031' is at the bottom right.

Figure 49 Interface des statistiques

Interface Menu d’administrateur :

C’est le panneau de control de l’administrateur, cette interface sert comme une redirection vers les autres interfaces de l’administrateur.



Figure 50 Interface Menu de l'administrateur

Interface Menu de superviseur :

C'est le panneau de control de superviseur, cette interface sert comme une redirection vers les autres interfaces de superviseur.



Figure 51 Interface Menu de superviseur

3.2.3. Interfaces des employés

Interface Menu principal :

Cette interface sert comme une redirection vers le menu commande carte ou bien le menu mise à jours carte.

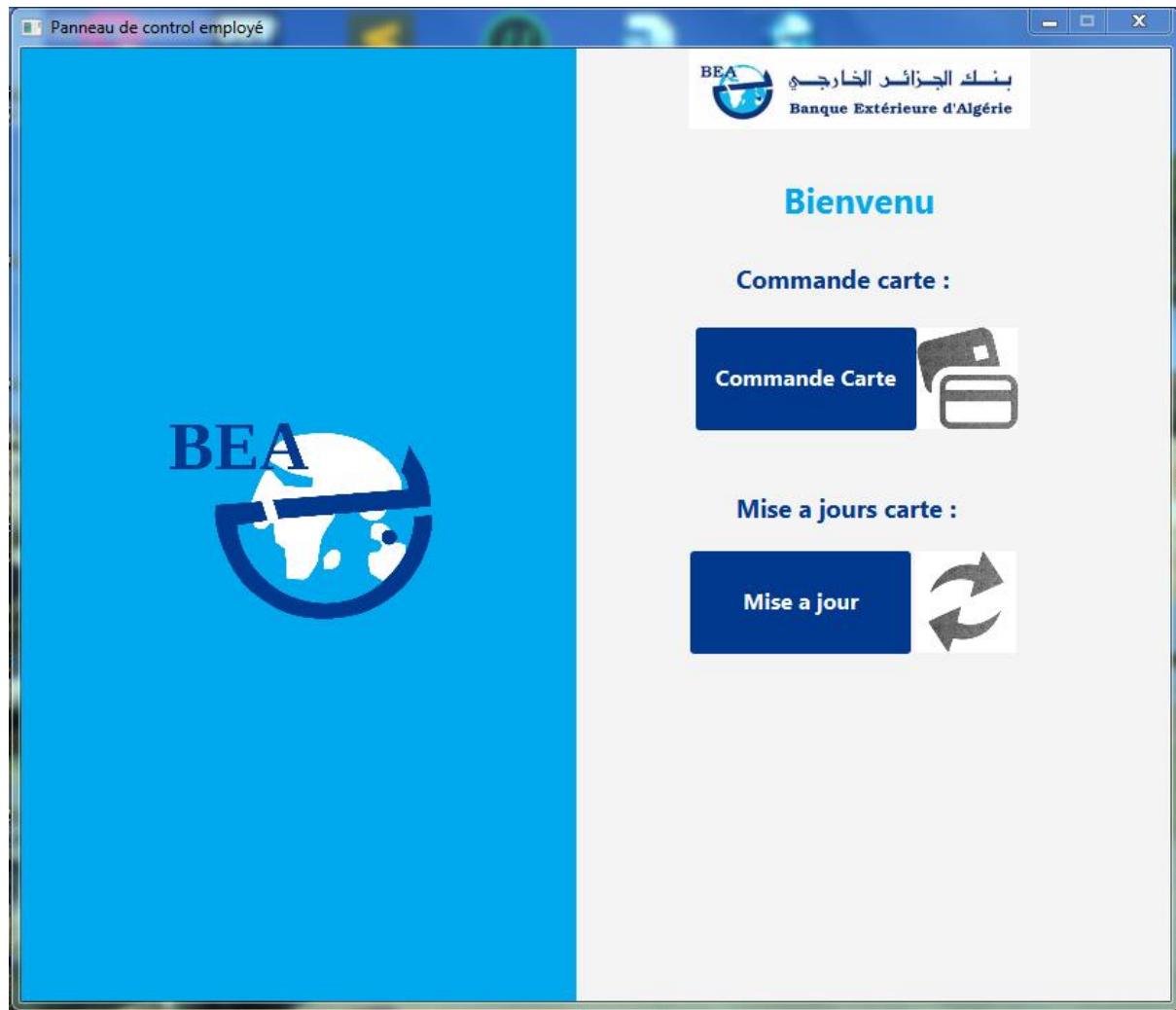


Figure 52 Interface Menu Principal de L'employé

Interface Menu commande carte :

C'est le panneau de control d'employé, cette interface sert comme une redirection vers les autres interfaces de la commande carte.

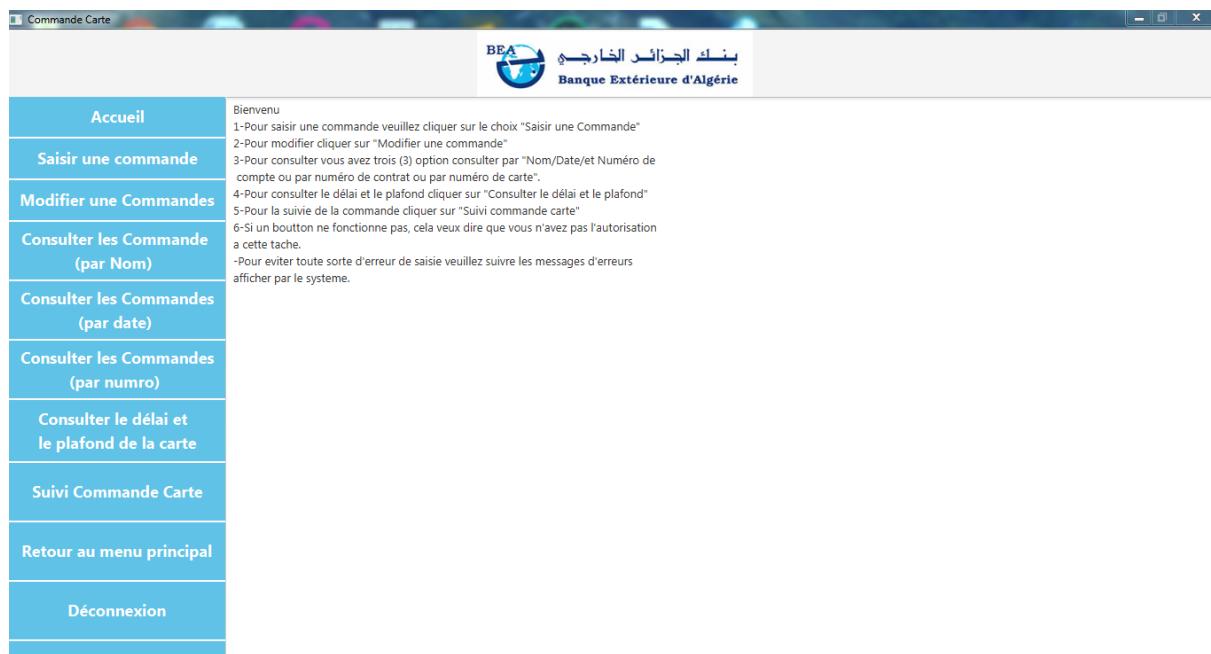


Figure 53 Interface Menu commande carte

Interface Saisir une commande :

L’interface permet de :

- Enregister les coordonnées de la commande carte.
- générer la carte une fois la commande est ajouté.
- générer ou bien maître à jour un fichier texte des commande cartes.

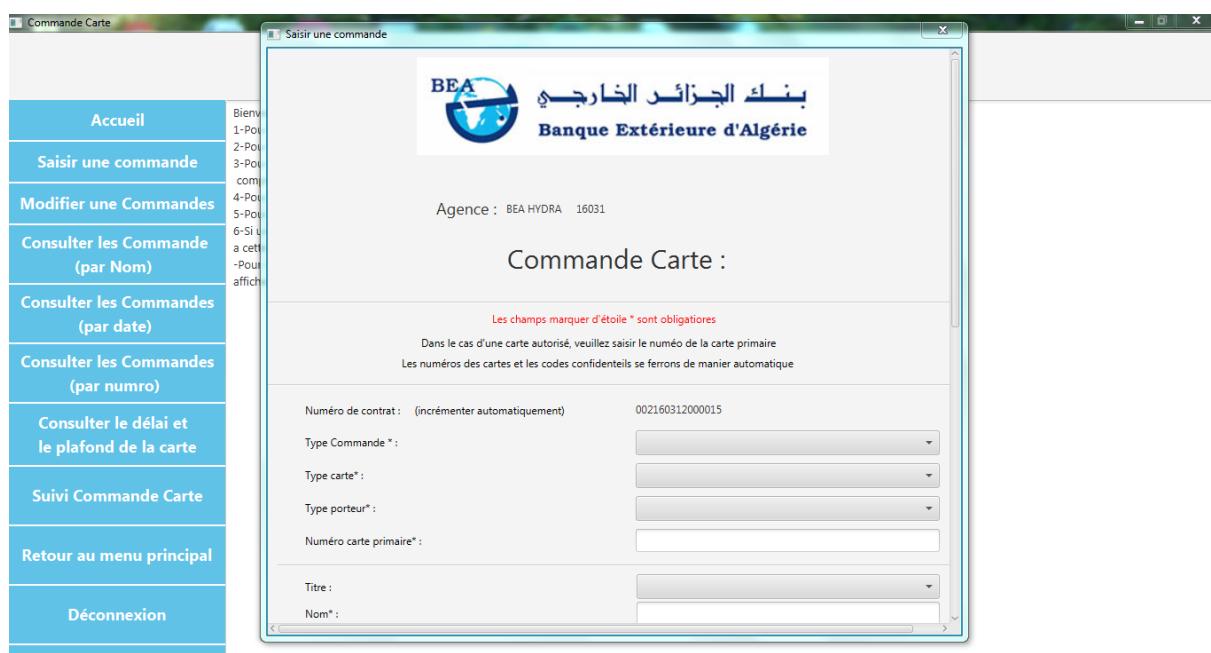


Figure 54 Saisir une commande

The screenshot shows a software interface for entering a command. The main window is titled "Saisir une commande". It contains various input fields for personal information such as date of birth, place of birth, address, sex, commune, daira, wilaya, postal code, situation, phone number, profession, company name, monthly net income, and account number. At the bottom of the window are three buttons: "Valider" (green), "Effacer" (red), and "Retourne à la page d'accueil" (grey).

Figure 55 Saisir une commande « Suite »**Interface de consulter les commandes :**

L’interface permet à :

- Consulter les données relatives aux commandes cartes.
- L’employé peut consulter les commandes par le Nom du client, entre deux dates de saisie, et par numéro (de contrat, compte, et carte).

The screenshot shows a software interface for querying commands. The main window is titled "Consulter les Commandes". It features the BEA logo and the text "بنك الجزائر الخارجي" and "Banque Extérieure d'Algérie". Below this, it displays "N° Agence : 16031" and "Nom Agence : BEA HYDRA". A section titled "Consulter les Commandes" includes fields for "Entre" and "Et" with dropdown menus. At the bottom is a "Valider" button. A table header is visible at the very bottom: NumContrat, NumCompte, TypeCommande, TypeCarte, TypePorteur, NumCartePrimaire, EtatCommande, Nom, Pren.

Figure 56 Interface Consulter les commandes (par date)**Interface de Suivi commande carte :**

L’interface permet à :

- Consulter la date de saisie et la date de traitement de la commande ainsi que le numéro de contrat et le numéro de compte, le nom et le prénom du client.
- L’employé peut faire le suivie de la commande avec le numéro de contrat, compte, et carte.

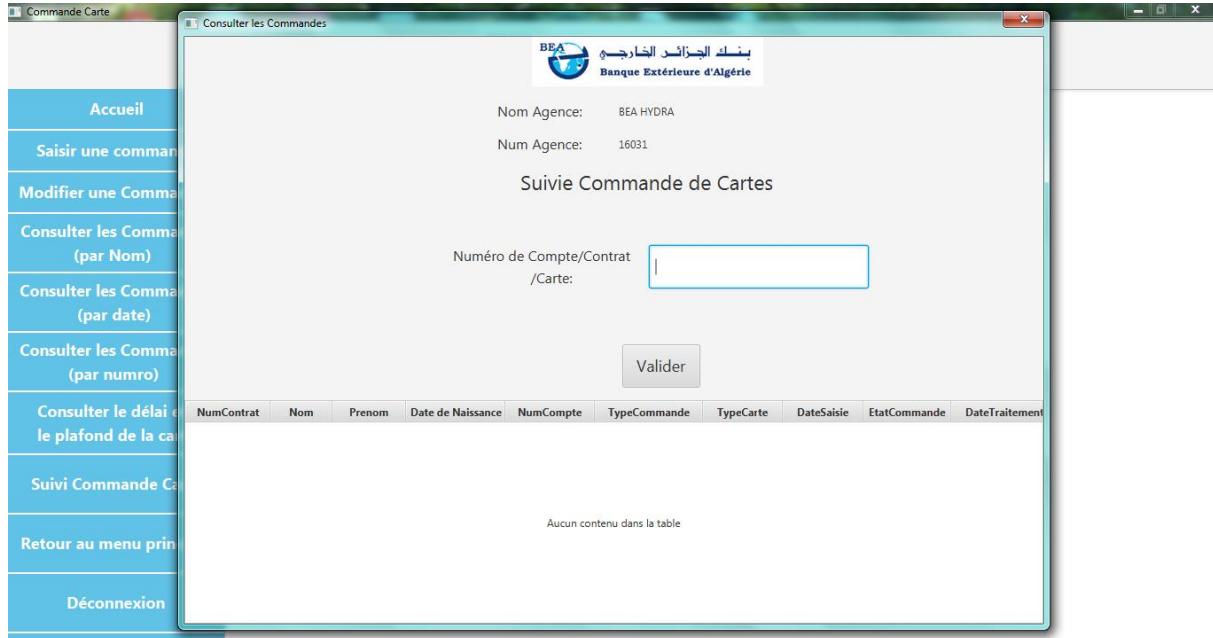


Figure 57 Interface Suivi commande carte

L’interface de menu des mises à jour de la carte :

C’est le menu des toutes les opérations que l’employé peut effectuer sur les cartes (de type Carte Primaire ou Carte Autoriser).

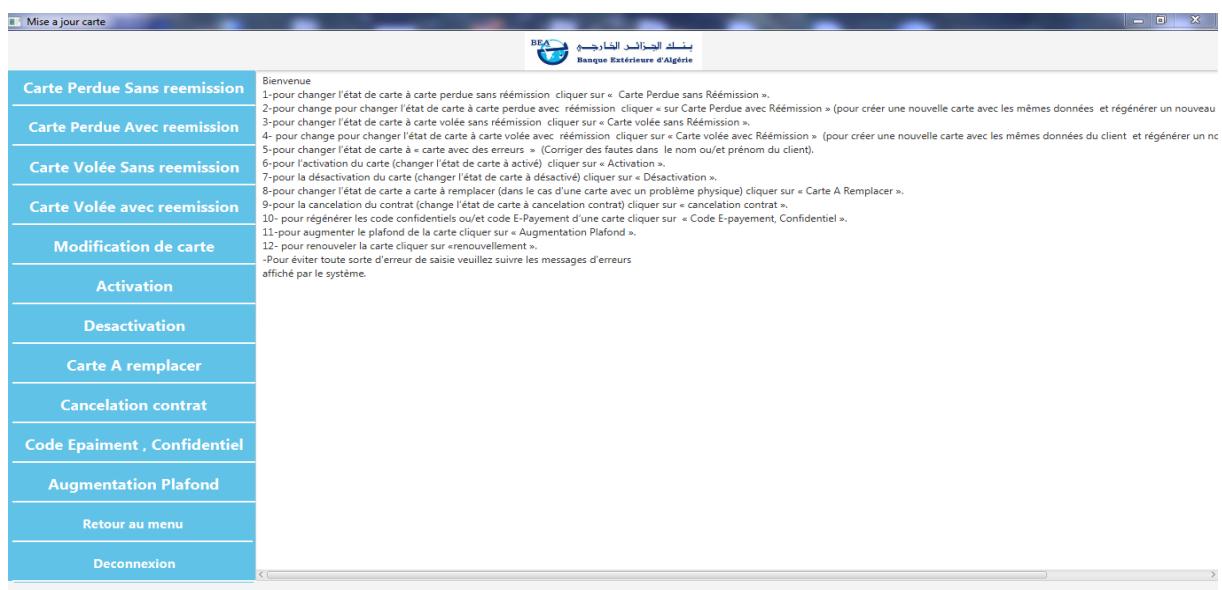


Figure 58 Interface menu des mises à jour carte

L'interface de type de carte :

On trouve ce type d'interface dans toutes les fonctionnalités, il permet à l'employé de choisir le type de carte qui sera mis à jour.



Figure 59 Interface de choix de type de carte à modifier

L'interface d'insertion des informations de la carte :

Elle permet à effectuer les mises à jour à la carte en saisissant les informations nécessaires et en cliquant sur le bouton de validation (Valider, Corriger).

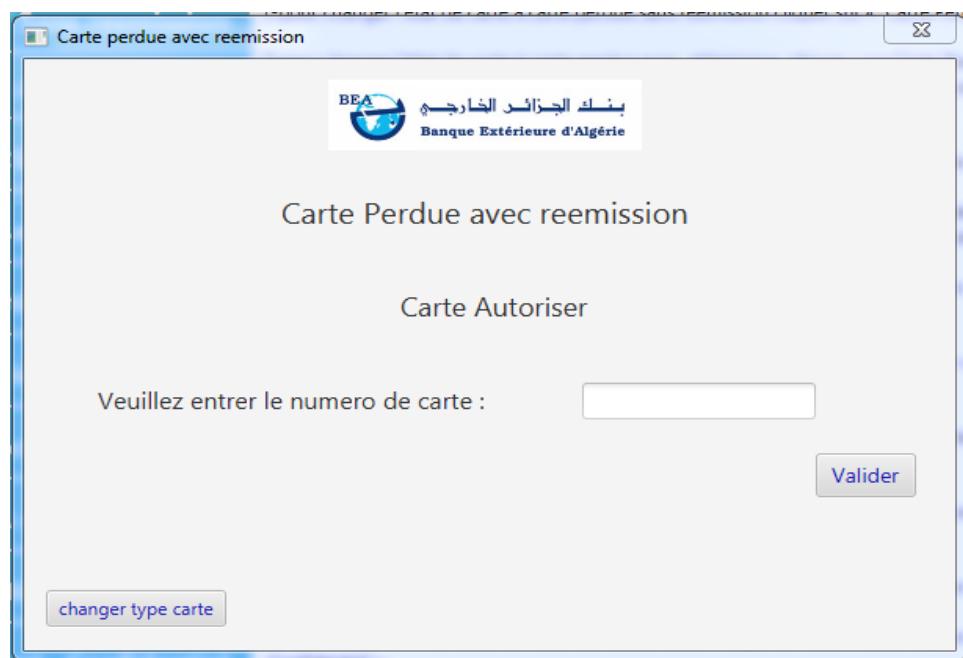


Figure 60 interface de modification de carte perdue avec réémission pour carte autorisé

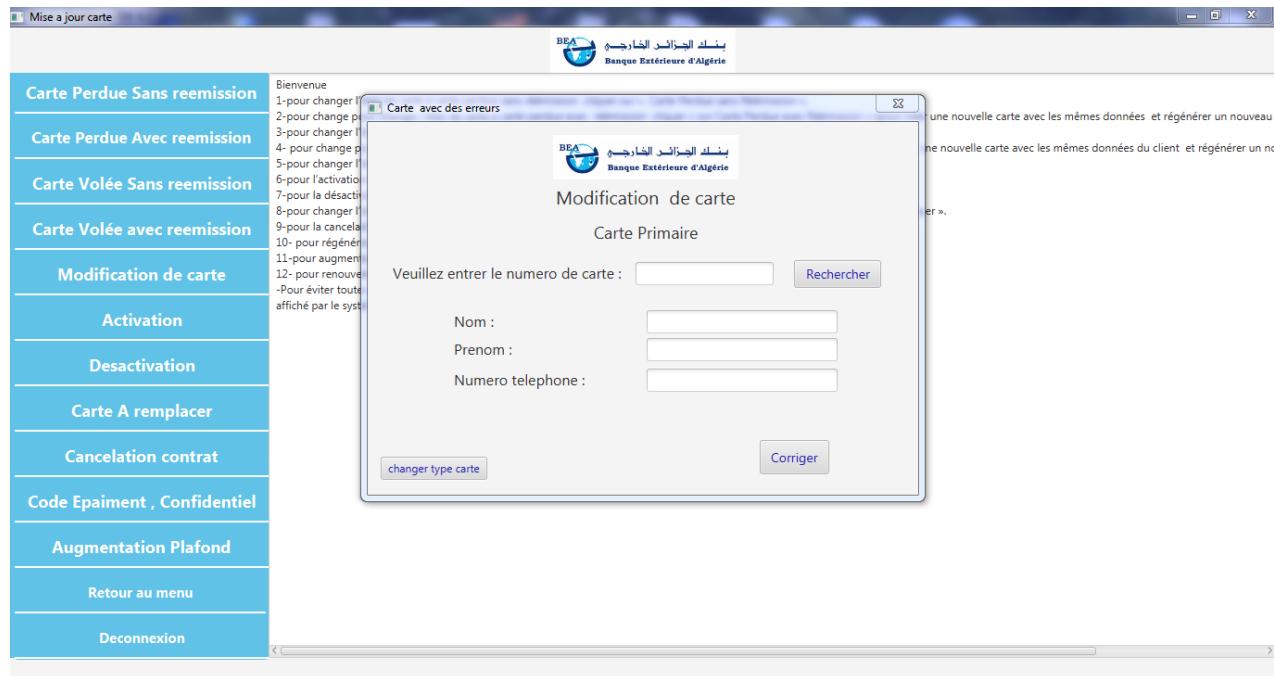


Figure 61 Interface de modification de carte de type primaire



Figure 62 Interface de régénération des codes confidentiel / codes e-paiement

3.3.Protection de système

Pour assurer la confidentialité, l'intégrité et la disponibilité de notre système, nous avons appliqué les concepts suivants pour faire face aux différentes menaces :

3.3.1. Protection logique

Prenant en compte l'importance des données traitées dans le système et l'architecture proposée, les procédures de contrôle (système de sécurité) doivent permettre d'assurer la sécurité au niveau des applications.

Filtrer l'accès au système : Tout utilisateur doit s'authentifier pour se connecter avec un mot de passe privé codé en algorithme d'hachage SHA256 en ajoutant un préfixe et un suffixe (SALT) au résultat d'hachage. Comme ça, les données cryptées des cartes et les mots de passe des utilisateurs de système ne pourront pas être décrypter si la base des données fuitera. De plus, aucun utilisateur quel que soit son grade ne pourra pas avoir le mot de passe d'un autre utilisateur.

Matricule_Employe	Mdp_Employe
123121221	3596a38f28745af0aab5ce43678fdab17a5a275fae32263214...
123123456	88f0ddf1a71140bf242a7c2fbe0ed863d41a9de0f0fcfde01d...
123455689	acd16318696b71cb01763e520e7ffbc1ae4ca00b68e841e11c...
123455789	c7a84dcd2636ff8116866f9178be4e9b66048ea2ed9fe9f87e...
123456543	b32600305fe032cf18b9cc20f82d429befdbfd7c899ba4a50a...
123456777	ca343060531313e50975cfea26c6afd087ea163ab93eeba19f...
123456781	2ebc5410d452a20d4f093c497ab7a04db77b4d3b98a5bf3509...
123456788	3be2c8566e8d5fc10f01081ce4706629a4a48f269f94a662f7...
125678965	7898db685f880f94cd0679813592c7ba05b200022b5796a197...
165233211	1007810280a0f6bf6937ff22156079887bec119d19e3bbc5f0...
171723331	4bcc9cda36aa164302110ad16ee91729817e047fbfb211ee34...
178902333	a3a94727ba4aa8e3f2a3dd358ddfe88677dd8d8ea14e71d419...
182723110	1f69a0b0694ad44b19a06c05032ca5cd0ffa673a091fddf3c3...

Figure 63 Exemple sur le hachage des mots de passe dans la base des données

Filtrer l'accès aux données : Nous avons défini des privilèges d'accès. Chaque utilisateur aura un matricule et un mot de passe pour l'authentification et il lui sera affecté selon son rôle dans l'agence (ou banque), un ensemble de privilèges sur les fonctionnalités de l'application.

Filtrer l'accès à l'application : Chaque utilisateur (qu'il soit membre du personnel ou client) est affecté à un profil et ce dernier à son propre espace de travail.

3.3.2. Protection physique

Au niveau de la sécurité physique, nous avons les points suivants ont été prévu :

- La base de données devra être périodiquement sauvegardée sur des supports externes, ces derniers doivent être à leurs tours conservés dans des lieux sûrs, des importations et exportations ont été créées dans le système pour faciliter la sauvegarde et la restauration.

- Protéger le matériel en le mettant dans un lieu sûr dans la direction des moyens de payement de l'agence.
- Protéger le serveur par un onduleur au cas des coupures d'électricité et un Antivirus au cas d'injection de matériel intrus.

3.4.Chronogramme de projet

Le projet a été réalisé pendant 5 mois et demi entre mi-avril et fin Septembre. L'organigramme ci-dessous représente le diagramme de Gantt des différentes tâches importantes de notre travail.

Planning de Projet									
Tâche	Début	Fin	Durée	Avril	Mai	Juin	Juillet	Août	Septembre
Etude de l'existant	10/04/2020	09/05/2020	29 Jours						
Analyse et conception	16/05/2020	30/06/2020	35 Jours						
Réalisation	20/04/2020	10/09/2020	142 Jours						
Rédaction de la mémoire	10/04/2020	20/09/2020	162 Jours						

Tableau 7 Planning de projet

3.5.Conclusion

Dans ce dernier chapitre, l'environnement matériel et logiciel de notre projet a été présenté, ainsi que le choix des langages et outils de développements. Ensuite on a illustré les principales interfaces de ce travail à travers des captures d'écran.

Conclusion générale et perspectives

Le back office est l'ensemble des activités de soutien, de contrôle, d'administration d'une entreprise. Elles sont facilement reconnaissables et peuvent être fonctionnellement et spatialement séparées du reste des opérations. Dans le secteur bancaire par exemple, les tâches de back office désignent l'ensemble des opérations de gestion administratives ou techniques réalisées en agence ou dans les services centraux

L'objectif de notre projet de fin d'étude était de réaliser et développer une application back office qui permet la gestion des commandes et des cartes bancaire BEA et la génération de ces derniers, avec la prise en considération de certains critères.

La première phase dans la réalisation de ce projet était la récolte des informations nécessaires pour avoir un état de l'existant, et détecter la problématique ainsi que l'architecture.

La deuxième phase était l'analyse et la spécification des besoins qui nous a permis de distinguer les différents acteurs qui interagissent avec le système, ainsi que la conception détaillée, dans laquelle on a eu la structure globale de l'application.

La dernière phase de notre projet était la partie de réalisation qui a été consacrée pour la présentation des outils du travail et les interfaces les plus importants de notre application.

L'apport que ce projet nous a importé a été très important, en effet, il nous a permis d'apprendre une méthodologie de travail bien organisé et d'améliorer nos connaissances dans le monde de développement des applications et dans le langage java et son Framework java fx. Cette expérience nous a permis de maîtriser les fondamentaux de la modélisation dans l'informatique (le langage UML), ainsi que l'environnement de développement java à savoir l'IDE Netbeans 8.2.

En perspectives notre application pourrait être améliorée par plusieurs fonctionnalités qui à cause de manque de moyens et de la crise sanitaire actuel de pays, on n'a pas pu les réaliser durant notre période de réalisation de projet tel que la génération en dur des cartes bancaires et l'intégration de l'application avec le site web de la banque ainsi que la réalisation d'une version mobile de l'application.

Bibliographie et webographie

1. Banque Extérieur d'Algérie. Banque Extérieure d'Algérie. **[En ligne]**. Disponible sur :< <https://www.bea.dz/> > (Consulté le 20/04/2020).
2. SATIM. SOCIÉTÉ D'AUTOMATISATION DES TRANSACTIONS INTERBANCAIRES ET DE MONÉTIQUE. [En ligne]. Disponible sur :< <https://www.satim.dz/>> (Consulté le 01/09/2020).
3. BERGOUGNAUX, Patrick. Modélisation Conceptuelle de Données - Une Démarche Pragmatique. Bergougoux Pat. Lille 15 octobre 2019. 320 pages. 979-1069930872.
4. Université Paris Descartes, UNIFIED MODELING LANGUAGE (UML), In : Web mail de département MI de l'université **[en ligne]**. Disponible sur :< <http://helios.mi.parisdescartes.fr/~bouzy/Doc/UML-NotesCours.pdf> > (Consulté le 17/06/2020).
5. TAIBOUNI, N et BENGHALIA, A. Génie Logiciel LG1. Alger : Université d'Alger I. Publications Faculté des sciences Université d'Alger I Département M.I 2eme année informatique, Cours, 2016, 201 pages.
6. Université de Angers, Architecture MVC, In : Site officiel de l'université d ; Angers **[en ligne]** (Modifié le 24/01/2011.). Disponible sur :< http://www.info.univ-angers.fr/pub/icher/ens/l3info/dw_crs4.pdf > (Consulté le 22/07/2020).
7. ARAR Mohamed Akram et AGABI Rayane Younes. Conception et Réalisation d'un Système d'Information pour la gestion des Identifiants Numériques des clients d'Algérie Poste. Systèmes Informatiques. Alger : Université d'Alger I, 2018,63 pages.
8. StarUML. StarUML 3 A sophisticated software modeller for agile and concise modelling. **[En ligne]**. Disponible sur :< <http://staruml.io/> > (Consulté le 25/04/2020).
9. Netbeans. Welcome to fr.netbeans.org! **[En ligne]**. Disponible sur :< <https://fr.netbeans.org/> > (Consulté le 17/05/2020).
10. Bibliothèque de l'INSA de Toulouse. Guide pour la rédaction des références bibliographiques In : Site de la Bibliothèque de l'institut national des sciences appliquées de Toulouse **[en ligne]**. (Modifié le 16/07/2012.) Disponible sur :<http://bib.insa-toulouse.fr/_resources/formation/guide_redaction_biblio.pdf > (Consulté le 21/09/2020).