

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

**MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA
RECHERCHE SCIENTIFIQUE**



Université de Béjaia



Faculté Des Sciences Exacte

Département d'informatique

En vue de l'obtention du diplôme de master

Option : Génie logiciel

Thème

*Conception et réalisation d'une application web pour
la gestion de pharmacie*

Présenté par :

Mr OUAKKOUCHE Lamine Badri

Devant le jury composé de :

Présidente : BOULFEKHAR Samra	MCA	UNIVERSITE DE BEJAIA.
Encadreuse: AIT HACENE Souhila	MAA	UNIVERSITE DE BEJAIA.
Examinatrice: OUADA Sarah Farah	MCB	UNIVERSITE DE BEJAIA.
Examinatrice: OUAZINE Kahina	Doctorante	UNIVERSITE DE BEJAIA.

Année universitaire 2016/2017

Remerciement

Tout d'abord à ALLAH l'unique dieu.

Je tiens à exprimer mes remerciements avec un grand plaisir et un grand respect à notre encadreuse Madame AIT HACENE Souhila, pour Ses conseils, Sa disponibilité et ses encouragements qui m'ont permis de réaliser ce travail dans les meilleures conditions.

J'adresse aussi mes remerciements à la pharmacienne et au gérant de la pharmacie OUAKKOUCHE qui m'a beaucoup aidé en me dirigeant sur la bonne voie.

Je voudrai aussi exprimer ma gratitude envers tous ceux qui nous ont accordé leur soutien, tant par leur gentillesse que par leur dévouement.

Je ne peux pas nommer ici toutes les personnes qui de près ou de loin m'a aidé et encouragé mais Nous les remercions vivement.

Enfin je tiens à dire combien le soutien quotidien de ma famille a été important tout au long de ces quelques années, je leur dois beaucoup.

Dédicace

A mes très chers parents

*Pour tout ce que vous m'avez donné, pour tout ce que
vous avez fait pour moi.*

Tables des matières

Table des matières.....	I
Liste des figures.....	IV
Liste des tableaux	V
Liste des abréviations.....	VI
Introduction général	01
1 Étude et critique de l'existant.	
1.1 Introduction.....	02
1.2 Présentation de l'organisme d'accueil.....	02
1.2.1 Rôle du personnel de la pharmacie.....	02
1.2.3 Organigramme de la pharmacie.....	03
1.3 Situation informatique.....	04
1.3 .3 Chifaa	04
1.4 Problématique et objectif.....	06
1.4.1 Problématique.....	06
1.4.2 Objectifs.....	06
1.5 Besoins fonctionnel.....	06
1.6 Besoins non-fonctionnel.....	07
1.7 Méthodologie de développement.....	07
1.7.1 Définition du Processus Unifié	07
1.7.2 Phases du processus unifié.....	07
1.7.3 Activités du processus unifié.....	08
1.8) Langage de modélisation	08
1.8.1 Unified Modeling Language	08
1.8.2 Diagramme UML a utilisé.....	09
1.8.3 Liens entre les diagrammes.....	10
1.9 MVC (Model, Vue, Contrôleur)	10
1.9.1 Définition de la couche Modèles.....	10
1.9.2 Définition de la couche vue.....	11
1.9.3 Définition de la couche contrôleur.....	11
1.9.4 Illustration du modèle MVC.....	11
1.10 Conclusio.....	11
2 Conception	
2.1 introduction.....	12
2.2 Les acteurs du système.....	12
2.2.1 Définition.....	12
2.2.2 Identification des acteurs.....	12

2.2.2.1 Les acteurs principaux.....	12
2.2.2.2 Les acteurs secondaire.....	13
2.3 Identification des cas d'utilisation	13
2.4 Cas d'utilisation	14
2.5 Diagramme de cas d'utilisations	14
2.5.1 Diagramme de cas d'utilisation de l'employer.....	15
2.5.2 Diagramme de cas d'utilisation du gérant.....	16
2.5.3 Diagramme de cas d'utilisation global.....	17
2.6 Description textuelle des cas d'utilisations	18
2.7 Réalisation des diagrammes de séquence.....	21
2.8 Réalisation du diagramme de classe.....	28
2.8.1 Diagramme de classe	28
2.8.2 Dictionnaire de données	29
2.8.3 Le modèle relationnel	29
2.8.3.1 Règles de passage au modèle relationnel	29
2.8.3.2 Modèle relationnel	30
2.9 Conclusion.....	31
3 Étude technique et implémentation	
3.1 Introduction.....	32
3.2 Application web	32
3.3 Avantage d'une application web.....	32
3.4 Outil de développement	32
3.4.1 Développement des diagrammes.....	32
3.4.2 NetBeans	33
3.4.3 PhpMyAdmin	33
3.4.4 MYSQL	33
3.4.5 Mozilla Firefox	33
3.5 Langage de programmation.....	33
3.5.1 PHP	33
3.6 Framwork.....	34
3.6.1 YII framwork.....	34
3.7 Architecture global de l'application.....	34
3.8 Représentation des interfaces de l'application.....	35
3.8.1 Interface d'Authentification	35
3.8.2 Interface de création de Produit	36
3.8.3 Interface de recherche d'un produit	37
3.8.4 Interface de création d'échange	38
3.8.5 Interface de recherche d'échange	39
3.9 Conclusion.....	39
Conclusion général	40

Référence bibliographique41
Webographie.....42

Liste des figures

1.1 : Organigramme de la pharmacie.....	03
1.2 : Capture d'écrans de l'interface vente chifaa.....	05
1.3: Capture d'écrans de la facture chifaa.....	05
1.4 Lien entre les diagrammes UML.....	10
1.5 : Illustration du paterne MVC.....	11
2.1 Diagramme de cas d'utilisation de l'employer.....	15
2.2 Diagramme de cas d'utilisation du gérant.	16
2.3 Diagramme global de cas d'utilisations.....	17
2.4 Diagramme de séquence cas d'utilisation (Authentification).....	21
2.5 Diagramme de séquence cas d'utilisation (Gérer la vente).....	22
2.6 Diagramme de séquence cas d'utilisation (Gérer le stock).....	23
2.7 Diagramme de séquence cas d'utilisation (Gestion des utilisateurs).....	24
2.8 Diagramme de séquence cas d'utilisation (gérer les ruptures).....	25
2.9 Diagramme de séquence cas d'utilisation (gérer les échanges).....	26
2.10 Diagramme de séquence cas d'utilisation (État des ventes).....	27
2.11 Diagramme de classes.....	38
3.1 Représentation de l'architecture globale de l'application.....	34
3.2 Interface d'authentification.....	35
3.3 Interface d'ajout d'un produit.....	36
3.4 Représentation de l'interface de recherche de produit.....	37
3.5 Interface de création d'échange.....	38
3.6 Interface de recherche d'échange.....	39

Liste des tableaux

1.1 Représentatif du matériel existant au sein de la pharmacie.....	04
1.2 Représentatif du logiciel qui est utilisé au sein de la pharmacie	04
1.3 Descriptif des diagrammes UML.....	10
2.1 Identification des cas d'utilisation.....	13
2.2 Description textuelle du cas d'utilisation Authentification.....	18
2.3 Description textuelle du cas d'utilisation gérer le stock.	18
2.4 Description textuelle du cas d'utilisation gérer la vente.	19
2.5 Description textuelle du cas d'utilisation gérer les utilisateur.....	19
2.6 Description textuelle du cas d'utilisation gérer les ruptures.....	20
2.7 Cas d'utilisation consulté l'état des ventes.....	20
2.8 Description textuelle du cas d'utilisation consulter l'état des ventes	29

Liste des abréviations

- BDD :Base De Données.
- HTML :HyperText Markup Language.
- HTTP :HyperText Transfer Protocol.
- IDE :Integrated Development Environment..
- MVC :Modèle Vue Contrôleur.
- MySql :My Structured Query Language.
- PHP :Personal Home Page..
- SGBD :System de Gestion de Base de Données.
- SQL :Structured Query Language.
- UP :Unified Process.
- UML: Unified Modeling language
- CNAS : Caisse Nationa des Assurances social.

Introduction générale

Actuellement, le monde connaît une avancée technologique considérable dans tous les secteurs et cela grâce à l'informatique qui est une science étudiant les techniques du traitement automatique de l'information. Elle joue un rôle important dans le développement de l'entreprise et d'autres établissements [19].

Avant l'invention de l'ordinateur, nous enregistrons toutes les informations manuellement sur des supports en papier ce qui engendrait beaucoup de problèmes tel que la perte de temps considérable dans la recherche de ces informations ou la dégradation de ces dernières.

Ainsi, jusqu'à présent, l'ordinateur reste le moyen le plus sûr pour le traitement et la sauvegarde de l'information. Cette invention a permis d'informatiser les systèmes de gestion de données des entreprises, ce qui est la partie essentielle dans leur développement aujourd'hui.

Les pharmacies hospitalières et celles des dispensaires publiques font partie des établissements que l'informatique pourra beaucoup aider [19]

En effet, la croissance du nombre des médicaments hospitaliers nécessite la mise en place d'une gestion rationnelle efficace et rapide, or et jusqu'à ce jour, la gestion manuelle est encore la plus dominante.

Nous remarquons ainsi la mauvaise organisation du travail dans la pharmacie lors de la recherche d'une information ou lors de la création des statistiques, l'information n'est pas toujours précise et disponible d'où la nécessité d'introduire l'informatique dans les pharmacies.

Vu cet état de fait, notre projet de fin de cycle a pour objectif de concevoir et mettre en œuvre une application web interactive, fiable, conviviale et facile à intégrer dans l'environnement de travail des pharmacies.

Notre mémoire est organisé en trois chapitres principaux :

Le premier chapitre Étude et critique de l'existant est consacré à la présentation de l'organisme d'accueil. La spécification des besoins nous permettra de délimiter notre problématique et les objectifs visés, tout en introduisant le choix du cycle de vie et les étapes à suivre pour concevoir une application web de gestion.

Le deuxième chapitre porte sur la conception, il regroupe toutes les étapes de notre processus de développement en utilisant le langage de modélisation UML et le processus unifié UP

Le troisième et dernier chapitre est consacré à la réalisation où nous allons définir tout les outils qui nous ont permis de concevoir notre application web, quelques interfaces y seront présentées.

Notre travail s'achèvera par une conclusion générale.

1.1) Introduction :

Dans ce chapitre, nous allons présenter l'organisme d'accueil qui est la Pharmacie OUAKKOUCHE. Nous allons aussi déterminer le rôle de chaque acteur agissant dans cet organisme pour répondre aux problèmes rencontrés par chacun. Nous déminerons les objectifs à atteindre dans le but d'assurer une meilleure gestion, ensuite nous allons choisir le processus à suivre et le langage de modélisation adéquat.

1.2) Présentation de l'organisme d'accueil:

La pharmacie OUAKKOUCHE du Dr OUAKKOUCHE Salima fut ouverte le 16/03/1993 aux 40 cités tobbal Béjaia.

La pharmacie OUAKKOUCHE a pour objectif de :

- Assurer le bon approvisionnement des médicaments à la population.
- Bien orienter le malade dans son processus de guérison.
- Sensibiliser le citoyen.
- Conseiller le malade.
- Veiller à la préservation de la santé de la population.
- Elle joue le rôle d'intermédiaire entre la CNAS et l'assuré via chifa un logiciel de la CNAS.

Pour assurer le bon fonctionnement de la pharmacie, le gérant doit tout d'abord gérer le stock et donner une formation aux employé sous l'encadrement du pharmacien, qui eux accueillent le client, le conseillent dans certains cas et vendent le médicament.

1.2.1) Rôle du personnel de la pharmacie :

Dans ce qui suit, nous allons énumérer le rôle et la fonction de chaque membre actif au sein de la pharmacie:

Le Pharmacien:

Le pharmacien est un livre parlant, cette métaphore veut tout dire:

- Encadrer les employés dans leurs rôle de conseiller ;
- Donner des cours de chimie au employé et les suivre dans leurs travail d'apprentie chimiste ;
- Assister aux séminaires et aux conférences pour apporter du nouveau à la science et à la populations ;
- Suivre le gérant dans son travail de gestionnaire.

Le gérant: le gérant assure plusieurs tâches à savoir :

- L'approvisionnement journalier en termes de médicaments et produits parapharmaceutiques chez le fournisseur ;
- Règlement des factures ;
- Formation des vendeurs ;
- Vérifications des arrivages (factures des produits pharmaceutique et parapharmaceutique) ;
- Vérification des factures Chifaa (bordereau de la semaine, tarification, extra) ;
- Paiement des charges fiscales ;
- Versement hebdomadaire à la banque.

L'employé:

Assure plusieurs tâches:

- Réceptionne la clientèle;
- Conseille le malade (en cas de doute le vendeur doit systématiquement demander conseil au pharmacien ou au gérant);
- Mise en place des produits;
- Établir les factures client (Bon de commande, Facture chifaa) ;
- Établir la liste journalière des produits en rupture;
- Manipulation de l'outil informatique.

1.2.3) Organigramme de la pharmacie:

La pharmacie OUAKKOUCHE emploie deux personnes qui sont le gérant et l'employé plus le pharmacien :

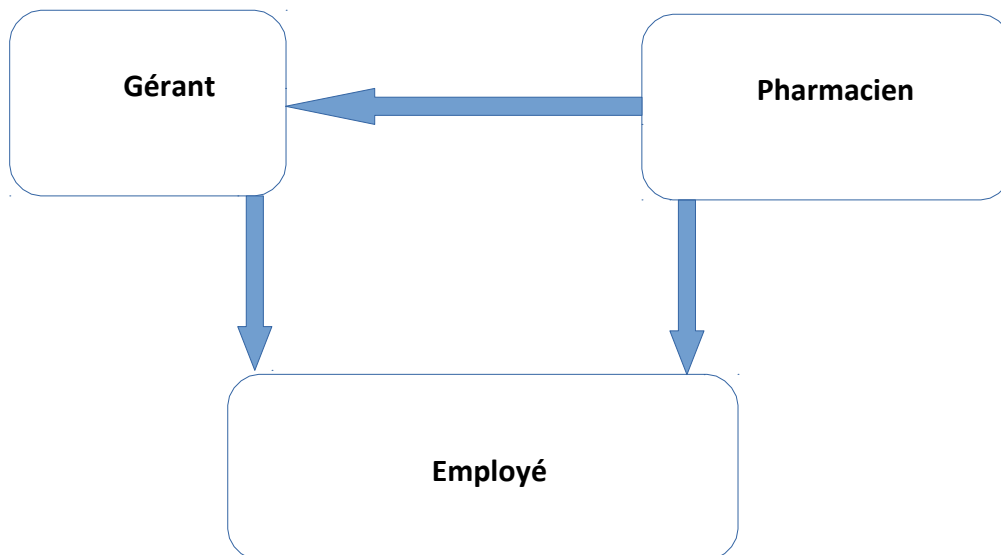


Figure 1.1 : Organigramme de la pharmacie.

1.3) Situation informatique:

Nous présentons dans le tableau ci-dessous la situation informatique en terme de matériels et de logiciels.

Matériels :

Matériels	Type	État	Marque	Quantité
Ordinateur	Bureau	Neuf	HP	2
Photocopieur	Photocopieur classique	Très bon	Koyera	1
Fax	Téléphone, fax	Bon	Panasonic	1
Modem	Sans fils	Neuf	Tp-link	1
Imprimante	Matriciel	Bon	Canon	2

Tableau 1.1 Représentatif des matériels existant au sein de la pharmacie.

Logiciel :

Logiciel	Rôle
Logpharm	Gère que 20% des produits sortant, mis à l'oublié.
Chifaa	Intermédiaire entre la CNAS, la pharmacie et le client.
Office Word & Exel	Support d'écriture pour Word, inventaire facture extra pour Exel.

Tableau 1.2 Représentatif des logiciels utilisés au sein de la pharmacie.

1.3.3) Chifaa :

Chifaa est un logiciel de la Caisse National des Assurances Sociales, il assure aux bénéficiaires de la carte Chifaa quelques avantages.

L'avantage qu'on doit souligner est que l'assuré ne se préoccupe plus du remboursement de ses médicaments il ne paye rien ou presque, car en effet le logiciel Chifaa assure une traçabilité via les factures des assurés et les bordereaux hebdomadaires remis par le gérant à la CNAS.

La CNAS après vérification du bordereau et des factures rembourse le pharmacien. Chifaa dispose d'une base de données de médicaments remboursés ou semi remboursés (médicaments dépassant leur tarif de référence imposé par la CNAS), la base de données est mise à jours chaque mois ou presque via une clé USB.

Pharmacien : Code Pharmacien : Code Centre :

N° Assuré : Nom Prescripteur :
 Nom Assuré : Code Prescripteur :
 N° Facture : N° Bordereau :
 N° Centre Payeur : Etablissement de soins: Structures hospitalière(CHU-EHS-EPH-SHM) Autres structures
 Type de Consultation : Spécialité :
 Bénéficiaire : Date de Naissance :
 Taux de Prise en Charge :
 Date de Fin Droits :

Nom Commercial du médicament :

Nom Commercial	Conditionnement	Dosage	Arret
ZYTEC	B/10	10MG	
VIRLIX	B/15	10MG	
CERIN	B/10	10MG	

Tarif de Reference: à partir de :
 Générique : Pays de Fabrication :
 Durée : Qté : PPA : Substitution Enregistrer Annuler
 Quantité :
 PPA :
 Ajouter Aperçu
 Modifier Valider
 Supprimer

Montant Assuré : DA
 Montant Officine : DA
 Montant Total : DA
 Montant Majoration : DA

Montant FAE : DA Majoration :
 Nouvelle Facture Signer Annuler Lire la Carte Lister l'Historique Fermer

Figure 1.2 : Capture d'écrans de l'interface vente chifaa.

edition facture :

Medicament Servis	Qt	Prix U	Montant	Montant assuré	Montant officine
CODOLIPRANE	1	124.43DA	124.43DA	0DA	124.43DA
VENTOLINE	1	238.53DA	238.53DA	0DA	238.53DA
SERETIDE DISKUS	1	2145.14DA	2145.14DA	0DA	2145.14DA
BIOCALYPTOL	1	202.71DA	202.71DA	0.71DA	202DA
Montant facture :				2 710.81	DA
Montant Réglé par l'assuré :				0.71	DA
Montant à Rembourser à l'officine :				2 710.10	DA
Formalités administratives et électroniques :				5.00	DA
Majoration :				Aucune	
Montant de la majoration :				0.00	DA
Montant global à Rembourser à l'officine :				2 715.10	DA

0% Page 1 sur 1

Figure 1.3: Capture d'écrans de la facture chifaa.

1.4) Problématique et objectifs:

Après avoir fait l'étude de l'organisme d'accueil, nous allons analyser des problèmes rencontrés par les membres du personnel de la pharmacie puis nous allons tenter d'apporter des solutions.

1.4.1) Problématique:

La gestion de cette pharmacie se fait manuellement ou presque sachant que 80 % du chiffre d'affaire de cette pharmacie viens exclusivement de la vente par chifaa et que Logpharm le logiciel existants ne gère pas, ce qui engendre plusieurs problèmes tels que :

- Une perte de temps dans la recherche des médicaments périmés ou proche de leur date de péremptions bien des produits en rupture.
- Les erreurs de calcul engendrées par la gestion manuelles (Bon, Facture, Facture Chifaa...).
- Une dispersion du stock pour certain médicaments.
- Utilisation de plusieurs documents (facture, bon de livraison, liste des produit en ruptures) ce qui entraine une mauvaise organisation de ces dernier.

Afin de palier à ces problèmes, nous proposons de développer une application web permettant à la pharmacie OUAKKOUCHE d'atteindre les objectifs présentés dans le point suivant.

1.4.2) Objectifs:

La phase d'analyse a pour objectif de décrire de manière précise, concise, correcte et compréhensible les besoins et les exigences du client. Il s'agit de livrer des spécifications pour permettre la conception de la solution.

La phase d'analyse permet de s'accorder sur « ce que doit faire le système ? »

La pharmacie OUAKKOUCHE veut se doter d'un logiciel qui pourra lui permettra de:

- Gérer le stock des médicaments.
- Gérer la vente de produit.
- Gérer les échanges de produits entre pharmacie.
- Visualiser les factures chifaa à distance et en temps réel.
- Automatiser la vente chifaa dans la gestion de la pharmacie.
- Connaître les recettes de la pharmacie (recettes journalières, hebdomadaires et mensuels).
- Avoir une vue d'ensemble sur la pharmacie à distances.

1.5) Besoins fonctionnel:

Les besoins fonctionnels se rapportent aux fonctionnalités que l'application doit offrir pour satisfaire les utilisateurs.

Les fonctionnalités que doit intégrer l'application à développer sont :

- **Gestion des médicaments:** Cette opération consiste à suivre l'état du stock à savoir les mouvements réalisés sur le stock (entrée /sortie de médicament, quantité des médicaments dans le stock, liste de médicaments en vois de rupture ou de péremption).
- **Gestion des commandes :** cette opération est établie lorsqu'il y a un besoin de renouveler le stock des médicaments. L'utilisateur peut créer un bon de commande correspondant à ses besoins ou se référer directement à la liste des produits en rupture dans le stock.
- **Gestion des ventes:** cette opération consiste à réaliser une vente sur l'application.

L'utilisateur peut même consulter la liste des ventes (journalières, hebdomadaires et mensuelles).

- **Gestion chifaa:** cette opération est actionnée automatiquement après avoir enregistré une vente sur le logiciel de la caisse national d'assurance social (CNAS), la vente en elle même s'effectuera automatiquement sur l'application, les médicaments vendu seront déduit du stock et la facture s'enregistrera automatiquement sur l'application pour que le gérant puisse la visualiser à distance.
- **Gestion des échanges:** cette opération consiste à enregistrer les échanges entre les pharmacies, cette action permettra d'enregistrer l'échange et déduire le ou les produits échangés du stock existant.

1.6) Besoins non-fonctionnel:

Les besoins non fonctionnels sont indispensables et permettent l'amélioration de la qualité logicielle de notre système. Ils agissent comme des contraintes sur les solutions, mais leur prise en considération fait éviter plusieurs incohérences dans le système. Ce dernier doit répondre aux exigences suivantes :

- **Authentification :** le système doit permettre à l'utilisateur de saisir son login et son mot de passe pour accéder au système. Cette opération assure la sécurité du système et limite le nombre des utilisateurs.
- **Ergonomie :** le système devra offrir aux utilisateurs une interface qui soit la plus riche possible afin de limiter le nombre d'écrans. Par ailleurs, l'interactivité devra être adaptée (usage du clavier, menu, etc..).
- **Fiabilité :** le système doit être fiable (l'utilisateur doit avoir confiance en la qualité de son produit, pour mieux s'occuper du malade tant le domaine est sensible).
- **Accessibilité:** l'application doit être mobile c'est à dire que le gérant où le pharmacien peuvent accéder à cette dernière et avoir le même service en dehors de la pharmacie.

1.7) Méthodologie de développement :

Selon la nature de notre projet, nous avons jugés que PU (processus unifié) serait le plus adapté pour sa réalisation.

1.7.1) Définition du Processus Unifié :

PU ou UP (Unified Process), est une méthode orientées objets pour le développement de logiciel, c'est une méthodes générique itérative et incrémental, piloté par les cas d'utilisation centré sur l'architecture.

Le processus unifié fournit un cadre au développement logiciel pour la construction de systèmes orientés objet [1].

1.7.2) Phases du Processus Unifié :

La méthode UP se base sur quatre phases :

- **Analyse des besoins :** Établir une vision globale du projet où on spécifie les besoins et on étudie la faisabilité du projet.
- **Élaboration :** On reprend les éléments de l'analyse des besoins et on développe une

architecture de référence, les risques et la plupart des besoins sont identifiés.

- **Construction** : Finaliser l'analyse, la conception, l'implémentation et les tests puis transformer l'architecture de référence en produit exécutable tout en veillant à respecter son intégrité.
- **Transition** : Livraison du produit au client afin d'effectuer des essais pour détecter d'éventuelles anomalies [1].

1.7.3) Activités du processus unifié :

Chaque phase est constituée d'une succession d'activités. Les activités de la méthode UP sont :

- **Expression des besoins** : Compréhension et expression des besoins et des exigences du client qu'elle soit fonctionnelle ou non fonctionnelle.
- **Analyse et Conception** : Permet d'acquérir une compréhension approfondie des contraintes liées aux outils de réalisation en prenant en compte le choix d'architecture technique retenu pour le développement et l'exploitation système.
- **Implémentation** : On implémente le système sous forme de composants, bibliothèques et de fichiers. Elle a pour objectif de planifier l'intégration.
- **Tests** : Permettent de vérifier les résultats de l'implémentation de toutes les exigences et de s'assurer de la bonne intégration de tous les composants dans le logiciel.
- **Déploiement** : Livraison et exploitation du produit [1].

1.8) Langage de modélisation :

Sachant que le processus unifié exige l'utilisation d'UML, notre modélisation de la solution se fera en utilisant des diagrammes UML.

1.8.1) Unified Modeling Language (UML):

UML (Unified Modeling Language) est un langage de modélisation objet né de la fusion de trois langages de modélisation objet (Booch, OMT, OOSE) qui ont révolutionnés la modélisation dans les années 90.

UML est dit universel car il est indépendant des langages de programmation, des domaines d'application et aussi du processus de développement adopté.

Il permet de représenter un système sous forme de schémas. Sa notation graphique permet d'exprimer visuellement une solution objet, ce qui facilite la comparaison et l'évaluation de la solution.

Sa véritable force repose sur un méta modèle qui normalise la sémantique des concepts, qu'il véhicule. Notez qu'UML est ouvert et n'est la propriété de personne [2].

1.8.2) Diagrammes UML utilisés :

Pour modéliser notre système nous avons utilisés les trois types de diagrammes suivants:

Diagramme	Objectifs	Type
Diagramme de classes	<ol style="list-style-type: none"> 1. Point central de la modélisation du système pour décrire ce que le système doit faire (analyse) et avec quoi il va le faire (conception) . 2. Représentation de la structure statique du système d'information. 3. Modélisation des classes et de leurs relations. 	Statique
Diagramme de cas d'utilisation	<ol style="list-style-type: none"> 1. Décrire la manière dont une organisation ou un système externe doivent interagir avec le système. 2. décrire ce que doit faire le système. 3. Mettre en évidence les services rendus par le système. 	Fonctionnel
Diagramme de séquence	<ol style="list-style-type: none"> 1. validation des cas d'utilisation pour comprendre la logique de l'application. 2. Complète le diagramme des cas d'utilisation en mettant en évidence les objets et leurs interactions d'un point de vue temporel. 	Dynamique

Tableau 1.3 Descriptif des diagrammes UML.

1.8.3) Liens entre les diagrammes :

La figure suivante représente les liens entre les diagrammes à concevoir dans le chapitre suivant :

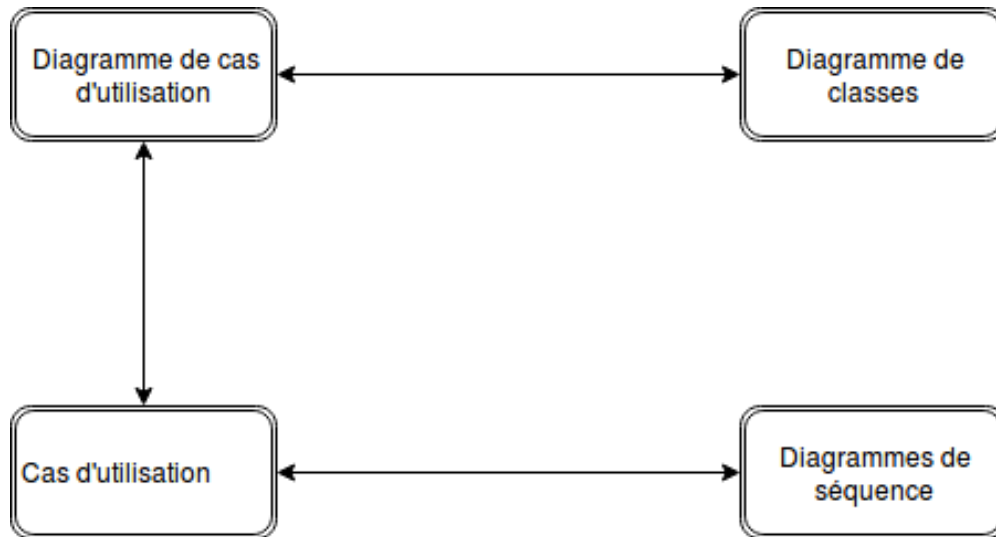


Figure 1.4 Liens entre les diagrammes UML.

1.9) MVC (Modèle, Vue, Contrôleur) :**Définition :**

MVC est un modèle d'architecture qui cherche à séparer nettement les couches de présentation (UI : User Interface), métier (BLL : Business Logic Layer) et d'accès aux données (DAL : Data Access Layer). Le but étant d'avoir une dépendance minimale entre les différentes couches de l'application ainsi les modifications effectuées sur n'importe quelle couche de l'application n'affectent pas les autres couches [3].

1.9.1) Définition de la couche Modèles:

La couche modèle représente la partie de l'application qui exécute la logique métier. Cela signifie qu'elle est responsable de récupérer les données, de les convertir selon des concepts chargés de sens pour l'application, tel que le traitement, la validation, l'association et beaucoup d'autres tâches concernant la manipulation des données.

À première vue, l'objet Modèle peut être vu comme la première couche d'interaction avec n'importe quelle base de données de l'application. Mais plus globalement, il fait partie des concepts majeurs autour desquels l'application est exécutée [4].

1.9.2) Définition de la couche vue :

La vue retourne une présentation des données venant du modèle. Étant séparée par les Objets Modèle, elle est responsable de l'utilisation des informations dont elle dispose pour produire une interface de présentation de l'application.

Par exemple, de la même manière que la couche Model retourne un ensemble de données, la Vue utilise ces données pour fournir une page HTML. Ou un résultat XML formaté.

La couche Vue n'est pas seulement limitée au HTML ou à la représentation en texte de données. Elle peut aussi être utilisée pour offrir une grande variété de formats en fonction de vos besoins, comme les vidéos, la musique, les documents et tout autre format auquel vous pouvez penser [4].

1.9.3) Définition de la couche contrôleur :

La couche contrôle gère les requêtes des utilisateurs. Elle est responsable de retourner une réponse avec l'aide mutuelle des couches Modèle et Vue.

Les contrôleurs peuvent être imaginés comme des managers qui ont pour mission de vérifier que toutes les ressources souhaitées pour accomplir une tâche soient déléguées aux travailleurs de façon appropriée. Il attend des requêtes des clients, vérifie leur validité selon l'authentification et les règles d'autorisation, délègue les données récupérées et traitées par le Modèle et sélectionne les types de présentation correcte que le client accepte, pour finalement déléguer le processus d'affichage à la couche Vue [4].

1.9.4) Illustration du modèle MVC :

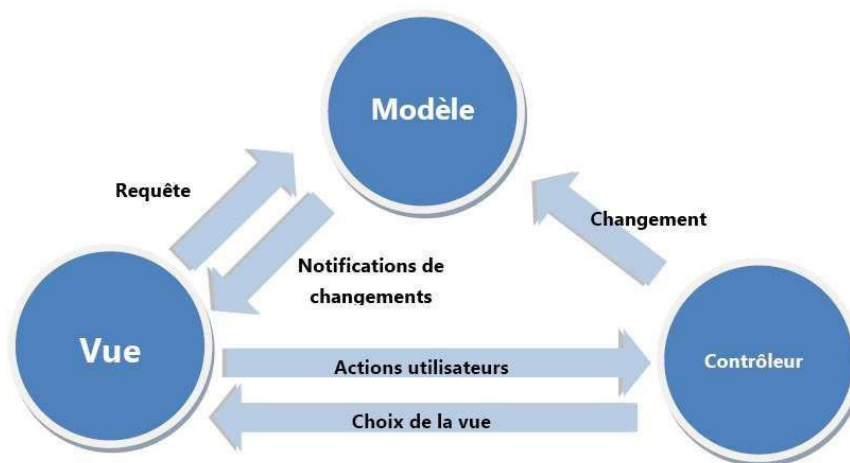


Figure 1.5 : Illustration du paterne MVC. [5]

1.10) Conclusion :

Dans ce premier chapitre, nous avons présenté l'organisme d'accueil et identifié le rôle de chacun des membres du personnel de la pharmacie. Ensuite, nous avons déterminé les différents problèmes rencontrés par ces derniers dans la réalisation de leurs tâches quotidiennes ce qui nous a permis de tirer les besoins fonctionnel et non-fonctionnel du système à concevoir, puis nous avons opté pour une application web.

2.1) Introduction:

Dans ce chapitre dédié à la conception, nous allons définir le rôle de chaque acteur qui interagit avec le système. Nous allons aussi modéliser leurs rôles sous forme de diagrammes de cas d'utilisation, puis nous définirons les cas d'utilisation et nous les modéliserons sous forme de diagramme de séquence. Nous finirons par le diagramme de classe et le modèle relationnel.

2.2) Les acteurs du système:

2.2.1) Définition :

Un acteur est une entité externe qui interagit avec le système (Utilisateur, dispositif matériel, ou autre système...). En réponse à l'action d'un acteur, le système fournit un service qui correspond à son besoin (modification du système ou simple consultation). Les acteurs peuvent être classés hiérarchiquement [6].

2.2.2) Identification des acteurs:

2.2.2.1) Les acteurs principaux :

La pharmacie OUAKKOUCHE emploie un employeur, un gérant et un pharmacien.

Dans notre système le pharmacien et le gérant ont le même rôle et doivent avoir les mêmes privilèges.

1. Le gérant :

Le rôle du gérant est de :

- Gérer la vente.
- Gérer le stock.
- Gérer les échanges de produits entre pharmacie.
- Gérer les utilisateurs.
- Formuler le bon de commande.
- Consulter l'état des ventes et factures chifaa.

2. L'employé:

Le rôle de l'employé est limité :

- Gérer la vente.
- Gérer le stock (l'employeur enregistre l'arrivée de produit).
- Consulter le bon de commande.
- Consulter la liste des périmés.

3. L'administrateur :

Le seul acteur ayant le droit d'accès au code source, la mise à jour de l'application et à sa maintenance en cas de panne.

2.2.2.2) Les acteurs secondaires:

Dans notre système, nous avons qu'un acteur logiciel qui va interagir avec notre système il s'agit bien sur du logiciel chifaa.

A chaque fois qu'une vente s'effectue sur ce dernier, la même vente s'enregistre automatiquement sur notre application et une copie de la facture chifaa est faite de la même occasion, ce qui va permettre au gérant de visualiser les ventes chifaa sans être sur l'ordinateur ou le logiciel et de faire une soustraction du stock automatiquement sans refaire l'opération à nouveau sur notre application (Appharm).

2.3) Identification des cas d'utilisation :

Dans l'application a développé, nous avons définis les cas d'utilisation suivant :

N°	Acteurs	Cas d'utilisation
1	Gérant/employer	Authentification
2	Gérant	Gérer le stock (Enregistrer, modifier, Rechercher produit, annuler, consulter) Gérer les ventes (Enregistrer, Modifier, Rechercher une vente, annuler, consulter) Gérer les échanges (Enregistrer, modifier, rechercher un échange, annuler, consulter) Gérer les utilisateurs (Ajouter, Modifier, Supprimer, rechercher un utilisateur, consulter) gérer les ruptures (Enregistrer, Modifier, Rechercher un bon de commande, annuler, Visualiser, visualiser la liste des produits en vois de ruptures) Consulter l'état des ventes : Liste de vente (Imprimer, Visualiser) Consulter les Factures chifaa (Imprimer, Visualiser) Liste des produit en vois de péremption (Imprimer, Visualiser) Liste des produit en rupture (Imprimer, Visualiser)
3	Employer	Géré les ventes (Enregistrer, modifier, Rechercher une vente, annuler, consulter) Gérer le stock (Enregistrer, Modifier, Rechercher un produit, annuler, consulter). Consulté le bon de commande (Visualiser, rechercher bon de commande)
4	Logiciel chifaa	Après enregistrement des factures émise par les utilisateurs dans le disque dur de l'ordinateur où est installer le logiciel chiffa, notre application détecte la facture dans le disque dur et l'enregistre à son tour dans la base de donnée du serveur au quel est connecté notre application.

Tableau 2.1 : Identification des cas d'utilisation.

2.4) Cas d'utilisation :

A) Authentification : Cette opération permet à un utilisateur de s'authentifier avant d'accéder à l'application, elle permet aussi de s'assurer de l'identité de l'utilisateur.

B) Gérer les ventes : Donne la possibilité à l'utilisateur d'enregistrer une vente ou de l'annuler, de rechercher une vente pour la modifier ou pour la consulter.

C) Gérer le stock : Permet à l'utilisateur d'enregistrer ou d'annuler l'enregistrement d'un produit dans le stock, de rechercher un produit pour consulter ou modifier ses caractéristique.

D) Gestion des échanges : Permet au gérant d'enregistrer un échange ou annuler l'enregistrement, de rechercher un échange pour le modifier ou le consulter.

E) Gérer les ruptures : Le gérant peut enregistrer ou annuler un bon de commande, il peut aussi rechercher un bon de commande pour le modifier ou le visualiser ou l'annuler. Elle permet aussi au gérant de visualiser la liste des produits en rupture. L'employé peut seulement rechercher un bon de commande pour le visualiser.

F) Consulter l'état des ventes: permet au gérant d'accéder aux listes de vente et recettes (Journalières, Hebdomadières, Mensuels), aux ventes chifaa et aux factures chifaa, pour les visualiser ou les imprimer.

G) Gérer les utilisateurs : permet au gérant d'ajouter un utilisateur de rechercher un utilisateur pour le supprimer, modifier ses caractéristique ou le consulter.

H) Enregistrement des factures chifaa : A chaque fois que notre application Appharm détecte une nouvel facture dans le disque dur de l'ordinateur où est installé le logiciel chifaa , elle enregistre tel qu'elle est dans base de donnée du serveurs au quel est connecté notre application et déduis les médicament vendu sur la facture automatiquement du stock.

2.5) Diagramme de cas d'utilisations :

Définition :

Les diagrammes de cas d'utilisation décrivent les utilisations requises d'un système, ou ce qu'un système est supposé faire. Les principaux concepts de ces diagrammes sont les acteurs, cas d'utilisation et sujets. Un sujet représente un système avec lequel les acteurs et autres sujets interagissent. Le comportement requis du sujet est décrit par les cas d'utilisation [7].

Le diagramme de cas d'utilisation modélise à QUOI ? Sert le système.

2.5.1) Diagramme de cas d'utilisation de l'employeur :

Le diagramme ci-dessous représente les cas d'utilisations identifiés pour l'employeur.

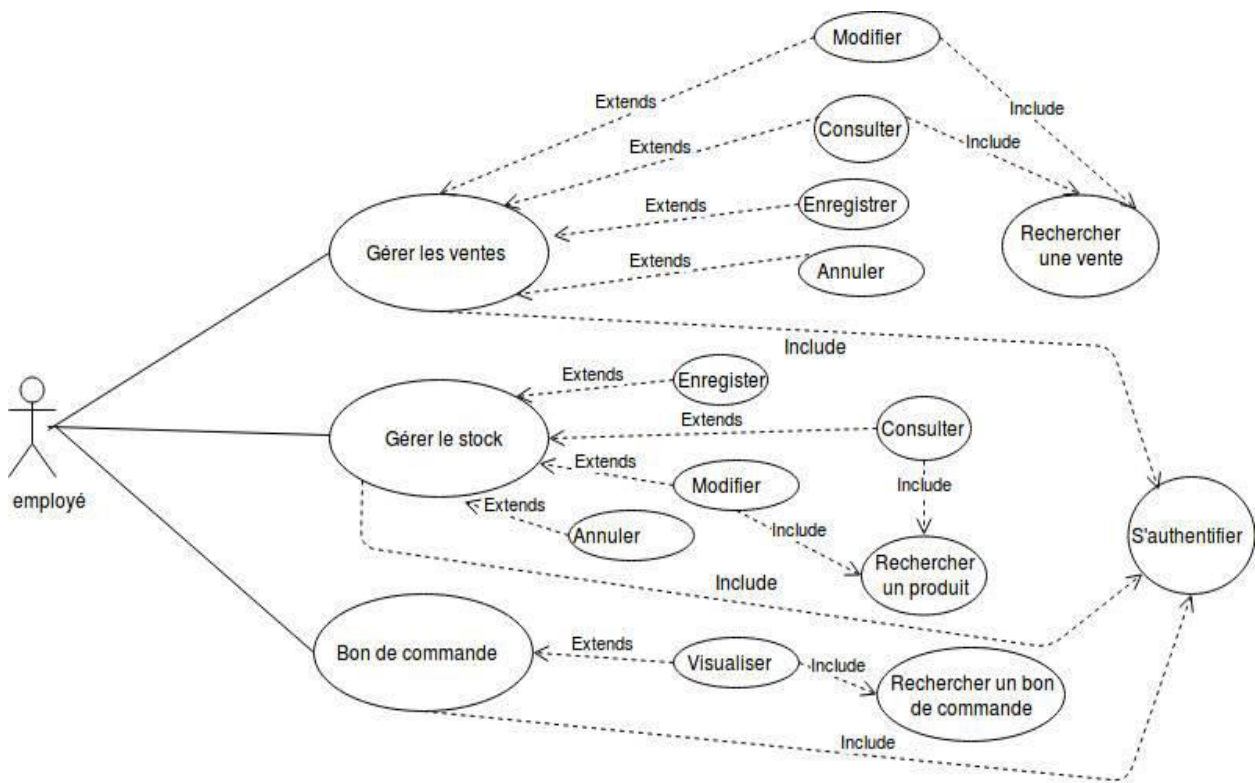


Figure2.1 : Diagramme de cas d'utilisation de l'employeur.

2.5.2) Diagramme de cas d'utilisation du gérant :

Le diagramme ci-dessous représente les cas d'utilisations identifiés pour le gérant.

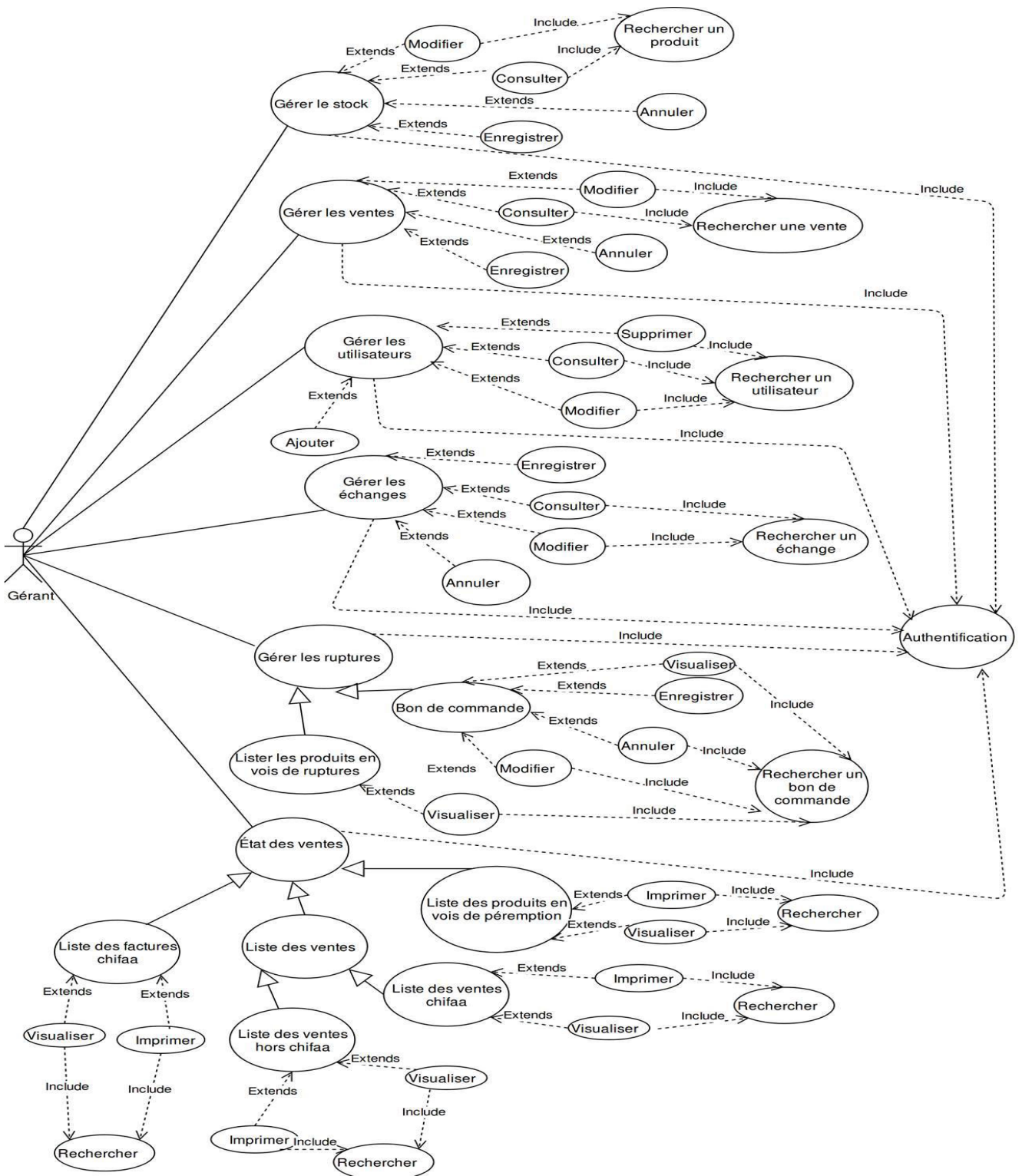


Figure 2.2 : Diagramme de cas d'utilisation du gérant

2.5.3) Diagramme de cas d'utilisation global : Le diagramme de cas d'utilisation global représente les différentes fonctions de notre application autour des quelles, sont érigées les besoins et les exigences des différents acteurs qui interagissent au sein même du système.

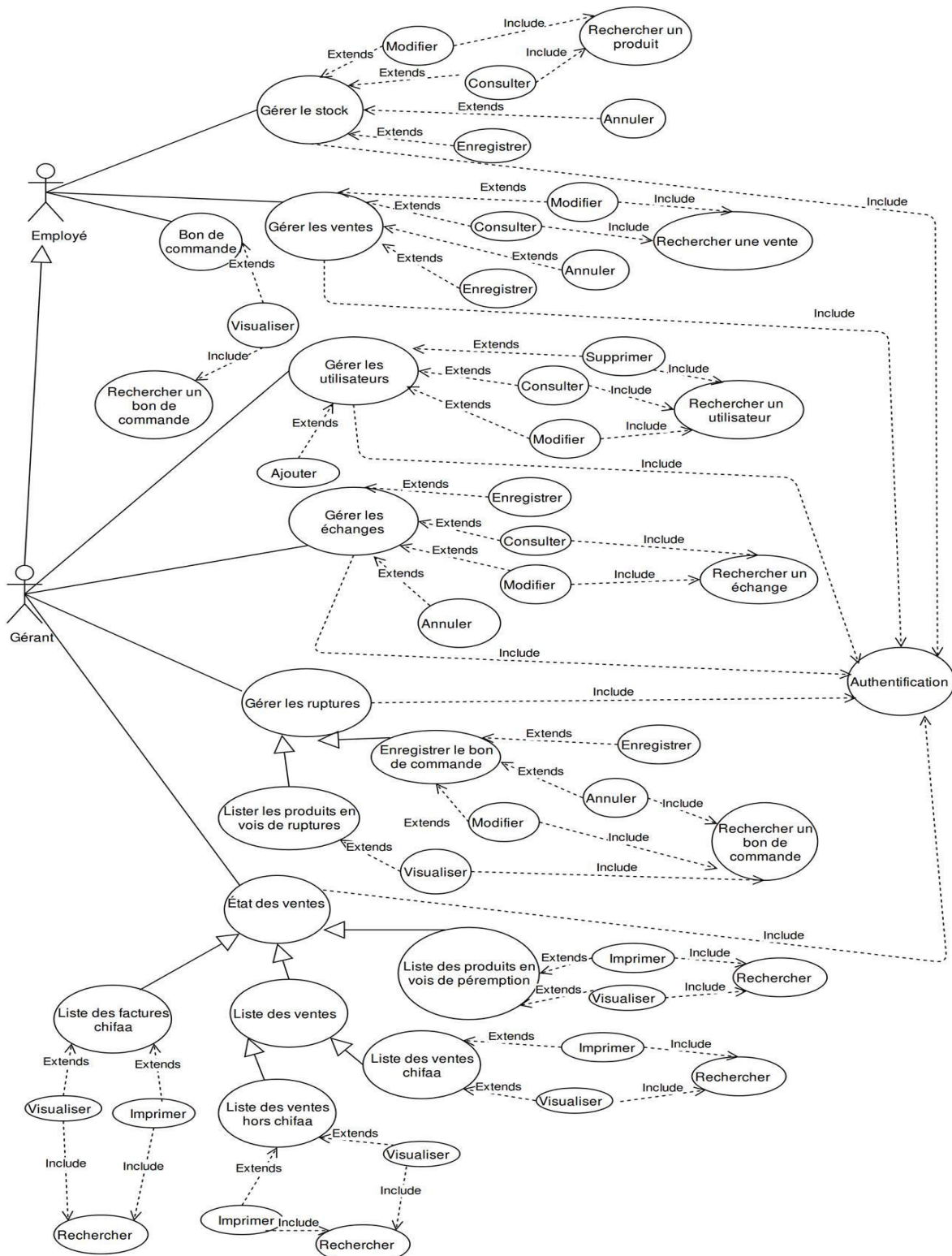


Figure 2.3 : Diagramme global de cas d'utilisations.

2.6) Description textuelle des cas d'utilisations :**Cas d'utilisation N°1 : Authentification**

Acteur principal	Gérant/ Employer
Objectif	S'authentifier avant d'accéder a la page d'accueil de l'application.
Pré-conditions	Avoir une connexion internet et un navigateur.
Scénarios	L'utilisateur se connecte à internet, lance l'application web via un navigateur web. Le système demande à l'utilisateur de s'authentifier. L'utilisateur saisit son nom et son mot de passe. Le système vérifie la conformité des informations saisies en envoyant une requête aux serveurs. La requête est vérifié par le serveur et envois une réponse favorable. L'utilisateur accède au menu principal.
Alternative	En cas de réponse défavorable du serveur, le système affiche un message d'erreur en cas d'erreur de saisie ou bien d'un champ incomplet (retour à 2).

Tableau 2.2 : Description textuelle du cas d'utilisation Authentification.**Cas d'utilisation N°2: Gérer le stock**

Acteur principal	Gérant/ Employer
Objectif	Enregistrer un produit dans le stock existant.
Pré-conditions	Authentification
Scénario	L'utilisateur saisie le nom du produit et tout ces caractéristiques (numéros de lot, date d'expiration, quantité...), puis clique sur enregistrer. Le système envois la requête au serveur. Après le traitement des données par le serveur, il envois un message au système. Le système affiche à l'écran la réponse du serveur
Alternative	Le système affiche un message d'erreur en cas d'une erreur de saisie ou bien d'un champ incomplet (retour a -1-).

Tableau 2.3 : Description textuelle du cas d'utilisation gérer le stock.

Cas d'utilisation N°3: Gérer les ventes

Acteur principal	Gérant/Employer
Objectif	Enregistrer une vente de produit/Imprimer la facture si besoin.
Pré-conditions	Authentification.
Scénario	<ol style="list-style-type: none"> 1. L'utilisateur saisie le nom du produit dans la barre de recherche de produit puis sélectionne le bon produit ajoute la quantité et clique sur enregistrer. 2. Une requête est envoyée au serveur pour traitement. 3. Le serveur envois un message de succès à l'interface de l'utilisateur.
Alternative	Le système affiche un message d'erreur en cas d'une erreur de saisie ou bien d'un champ incomplet (retour a -1-).

Tableau 2.4 : Description textuelle du cas d'utilisation gérer la vente.**Cas d'utilisation N°4 : Gérer les utilisateurs**

Acteur principal	Gérant.
Objectif	Enregistrer un utilisateur.
Pré-conditions	Authentification.
Scénario	<ol style="list-style-type: none"> 1. Le gérant accède a l'interface gérer les utilisateurs puis clique sur ajouter un utilisateur et enfin saisit le formulaire d'ajout et clique sur enregistrer. 2. Une requête est envoyée au serveur pour traitement. 3. Le serveur envois un message de succès a l'interface de l'utilisateur.
Alternative	Le système affiche un message d'erreur en cas d'une erreur de saisie ou bien d'un champ incomplet (retour a -1-).

Tableau 2.5 : Description textuelle du cas d'utilisation gérer les utilisateurs.

Cas d'utilisation N°5 : Gérer les ruptures

Acteur principal	Gérant
Objectif	Formuler un bon de commande/ accéder a la liste de produit en rupture.
Pré-conditions	Authentification
Scénario	<ol style="list-style-type: none"> 1. Le gérant accède a l'interface gérer les ruptures. 2. Le gérant demande a accéder a la liste des produit en rupture. 3. Une requête est envoyée au serveur pour traitement. 4. Le serveur envoie la liste des produits en voie de rupture et l'interface l'affiche a l'écran. 5. Le gérant peut formuler le bon de commande : <ul style="list-style-type: none"> • Saisie des produits et leurs quantités par le gérant qui ajoute au bon de commande. • A la fin de sa saisie il peut enregistrer • Une requête est envoyée au serveur pour traitement. • Le serveur répond par un message de succès.
Alternative	Le système affiche un message d'erreur en cas d'une erreur de saisie ou bien d'un champ incomplet (retour a -5-).

Tableau 2.6 : Description textuelle du cas d'utilisation gérer les ruptures.

Cas d'utilisation N°6 : consulter l'état des ventes

Acteur principal	Gérant
Objectif	Accéder aux nombreuses listes (liste des produits vendus)
Pré-conditions	Authentification
Scénario	<ol style="list-style-type: none"> 1. Le gérant demande le formulaire de consultation. 2. Le système affiche le formulaire. 3. Le gérant demande de consulter une des listes (listes des produits, périmés et vendus ou au facture chifaa). 4. Une requête est envoyée au serveur pour traitement. 5. Le serveur répond et le système affiche la liste demandée.
Alternative	Coupe internet/ou problème technique (réactualiser la page/ou vérifié connexion internet).

Tableau 2.7: Description textuelle du cas d'utilisation consulter l'état des ventes.

2.7) Réalisation des diagrammes de séquence :

Diagramme de séquence N°1 : cas d'utilisation Authentification :

Lorsque l'utilisateur (Gérant, Employer) veut accéder à notre application web, il sera obligé de s'authentifier avant d'y accéder en saisissant son identifiant et mot de passe, après la saisie le système envoie une requête au serveur pour traiter les informations envoyées, si les informations sont correcte l'utilisateur accèdera à sa session sinon un message d'erreur sera affiché et reconduira l'utilisateur à la page authentification.

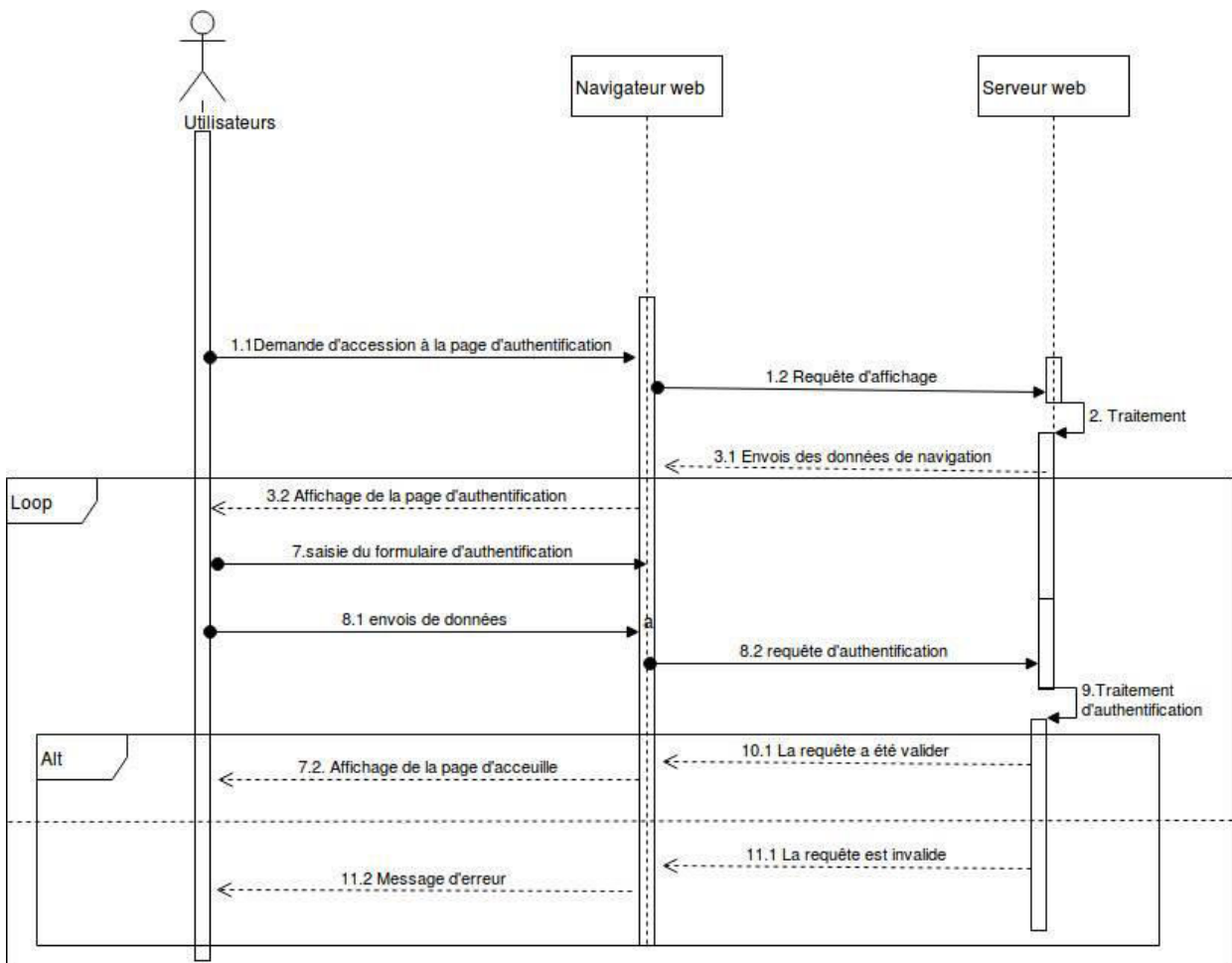


Figure 2.4 : Diagramme de séquence du cas d'utilisation (Authentification).

Diagramme de séquence N°2 : cas d'utilisation Gérer la vente :

Lorsque l'utilisateur (Gérant, Employeur) a accédé a l'application il lui sera possible d'effectu  une vente en cliquant sur gestion de vente, apr s le clique il pourra enregistrer une vente de produit en introduisant les champs requis et en cliquant sur enregistrer le syst me envoie la requ te au serveur qui lui enregistre les donn es et envoie un message de succ s ou sinon un message d'erreur sera affich  si il manque un champ.

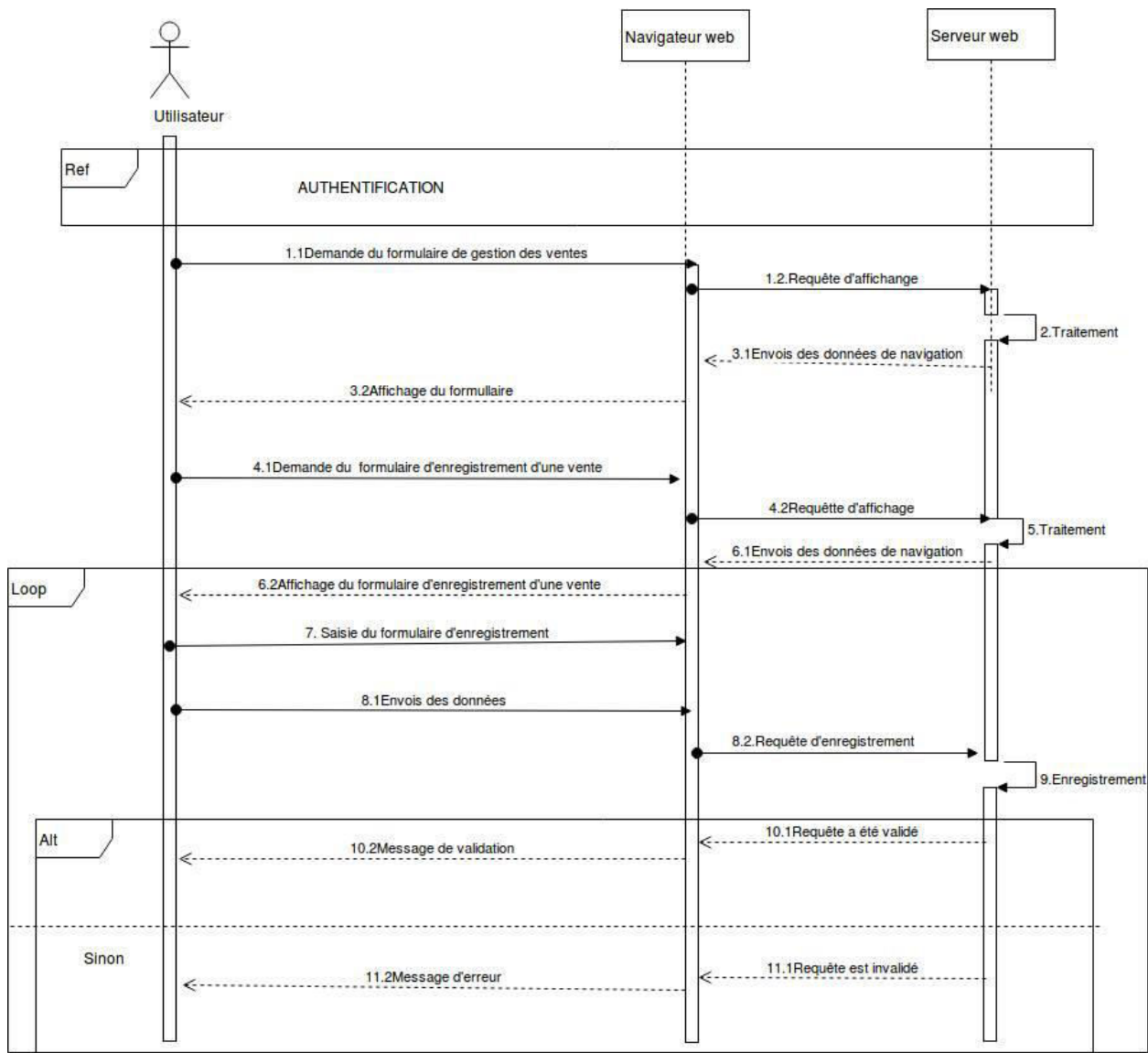


Figure 2.5 : Diagramme de s quence du cas d'utilisation (G rer la vente).

Diagramme de séquence N°3: cas d'utilisation Gérer le stock

Lorsque l'utilisateur (Gérant,Employer) veut gérer le stock c'est a dire enregistrer un arrivage de médicament, il doit tout d'abord cliquer sur gestion de stock puis ajouter un produit au stock, pour ensuite remplir les champ requis et enfin envoyer les données au serveur pour les enregistrer. Après traitement du serveur un message de succès ou d'erreur sera envoyé.

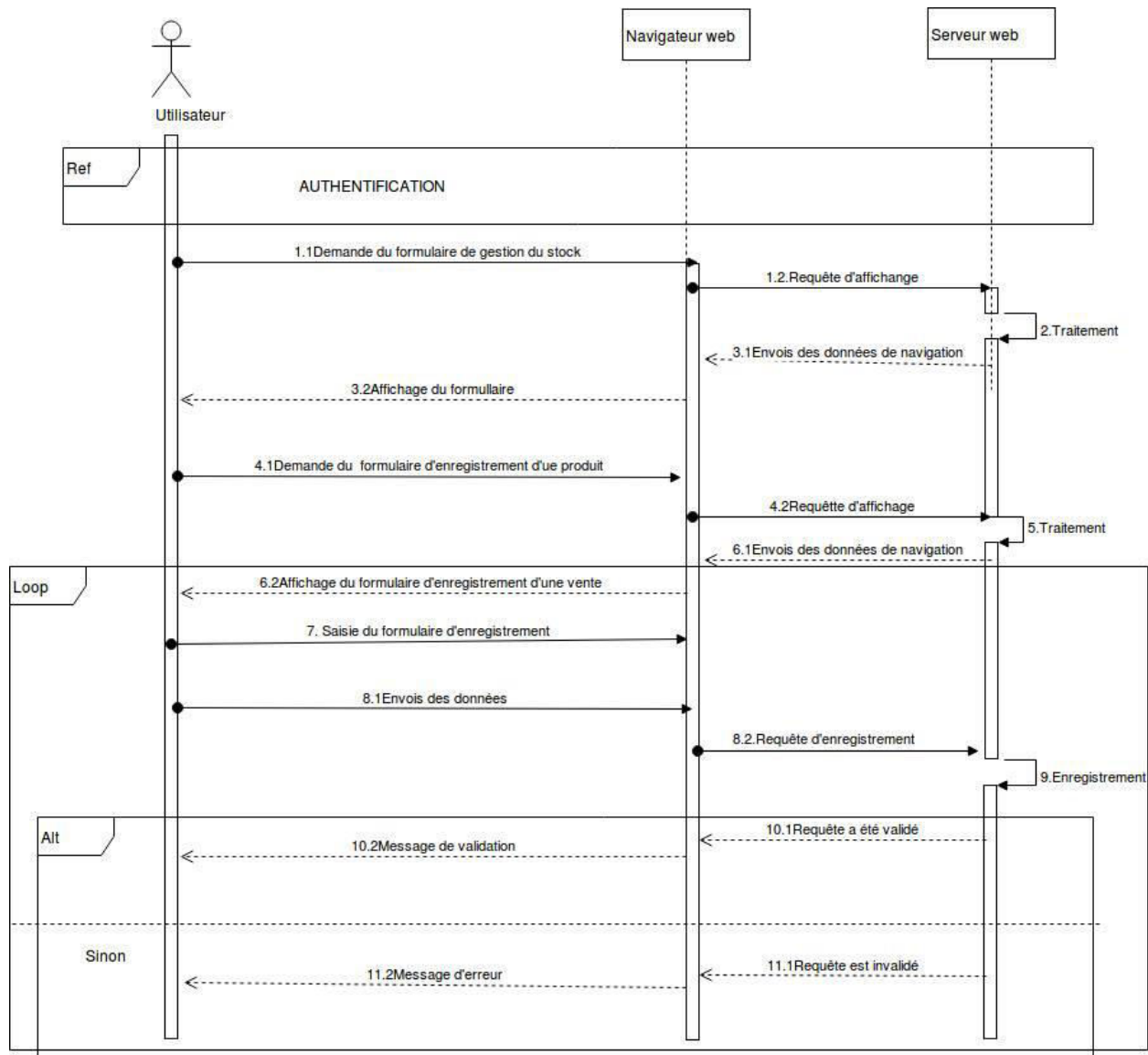


Figure 2.6 : Diagramme de séquence du cas d'utilisation (Gérer le stock).

Diagramme de séquence N°4: cas d'utilisation Gérer les utilisateurs :

Lorsque le gérant veut gérer les utilisateurs il doit tout d'abord accéder à la gestion des utilisateurs puis rechercher un utilisateur donné pour le modifier.

Après ça recherche le gérant n'a qu'à cliquer sur modifier et remplir les champs requis et envoyer au serveur pour traitement et enregistrement. Un message d'erreur est envoyé en cas de champ incomplet sinon un message de succès.

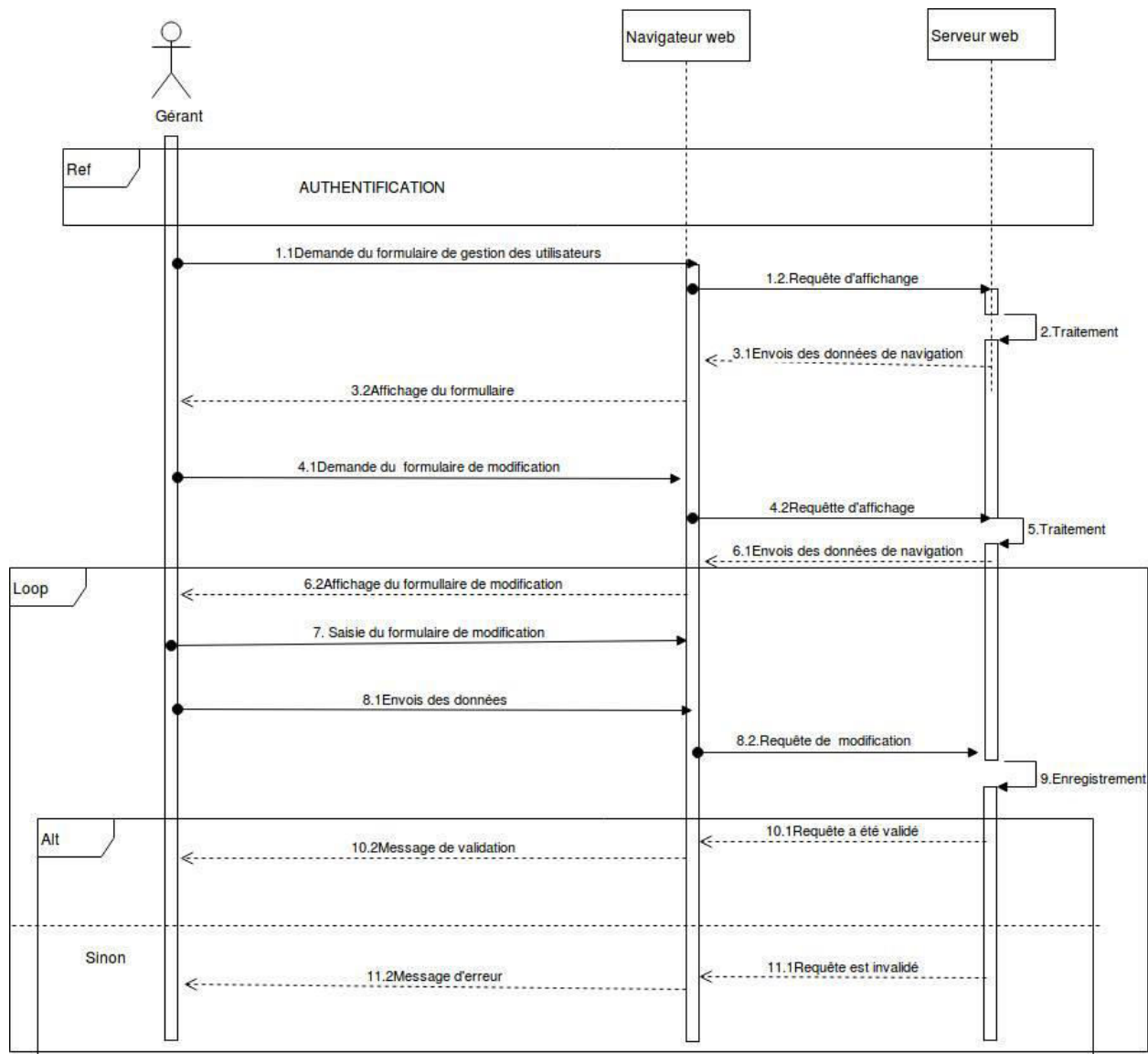


Figure 2.7 : Diagramme de séquence du cas d'utilisation (Gestion des utilisateurs).

Diagramme de séquence N°5: cas d'utilisation Gérer les ruptures :

Lorsque le gérant veut passer une commande au fournisseur il pourra formuler un bon de commande via notre application web. En effet, il doit accéder au formulaire gestion des ruptures et cliquer sur formuler le bon de commande puis remplir les champs requis et envoyé les données saisie au serveur pour traitement. Un message de validation serra affiché, l'employé pourra consulter le bon de commande et commander par téléphone sinon un message d'erreur serra envoyé par le serveur et le gérant devra corriger son erreur.

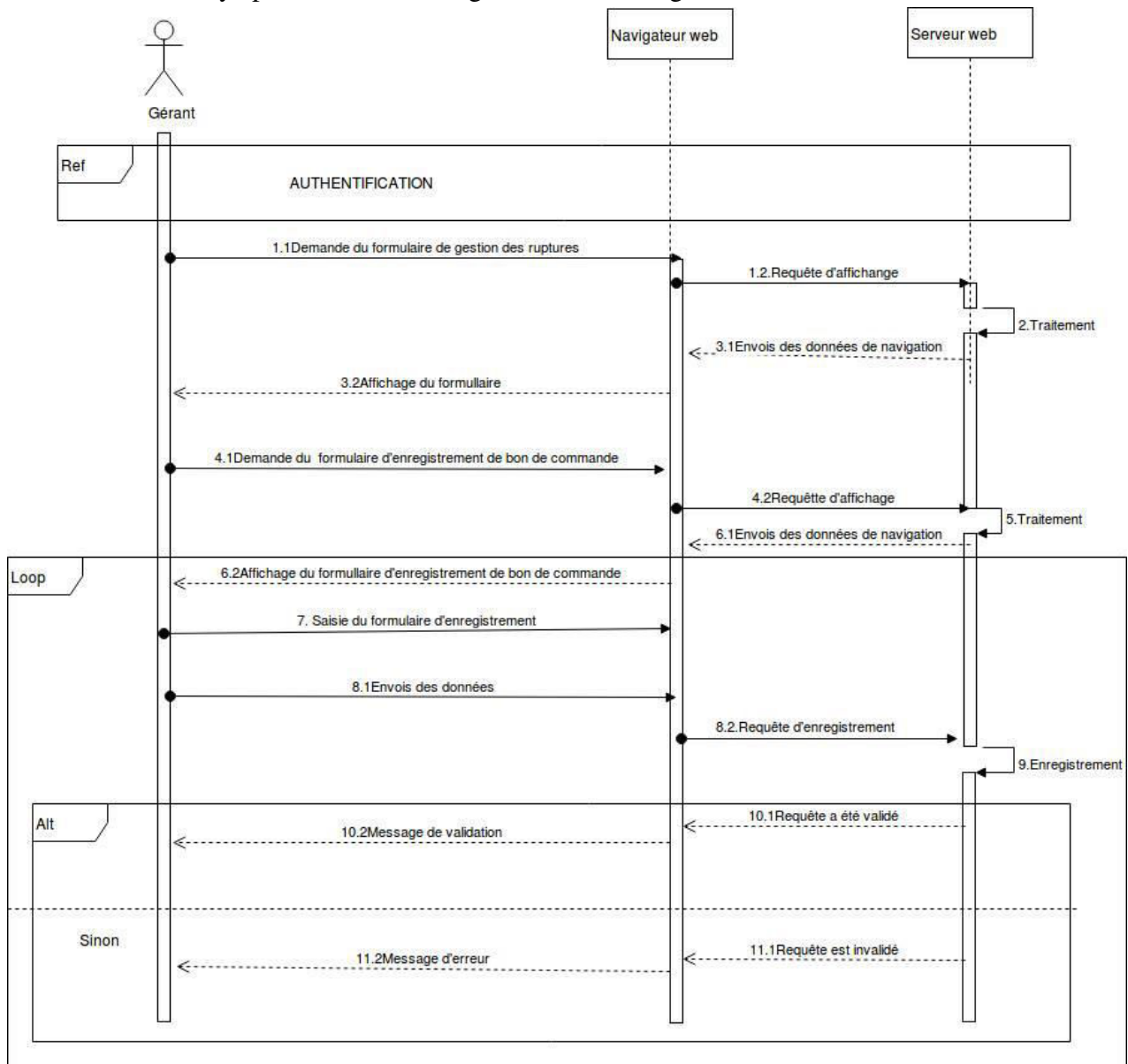


Figure 2.8 : Diagramme de séquence du cas d'utilisation (gérer les ruptures).

Diagramme de séquence N°6: cas d'utilisation Gérer les échanges :

Quand le gérant veut rechercher un échange de produit, il doit accéder à la rubrique gérer les échange et demander le formulaire de recherche puis remplir les champs requis et envoyé les donnée au serveur pour qu'il puisse rechercher et envoyer le but de la rechercher au gérant sinon un message d'erreur sera envoyé.

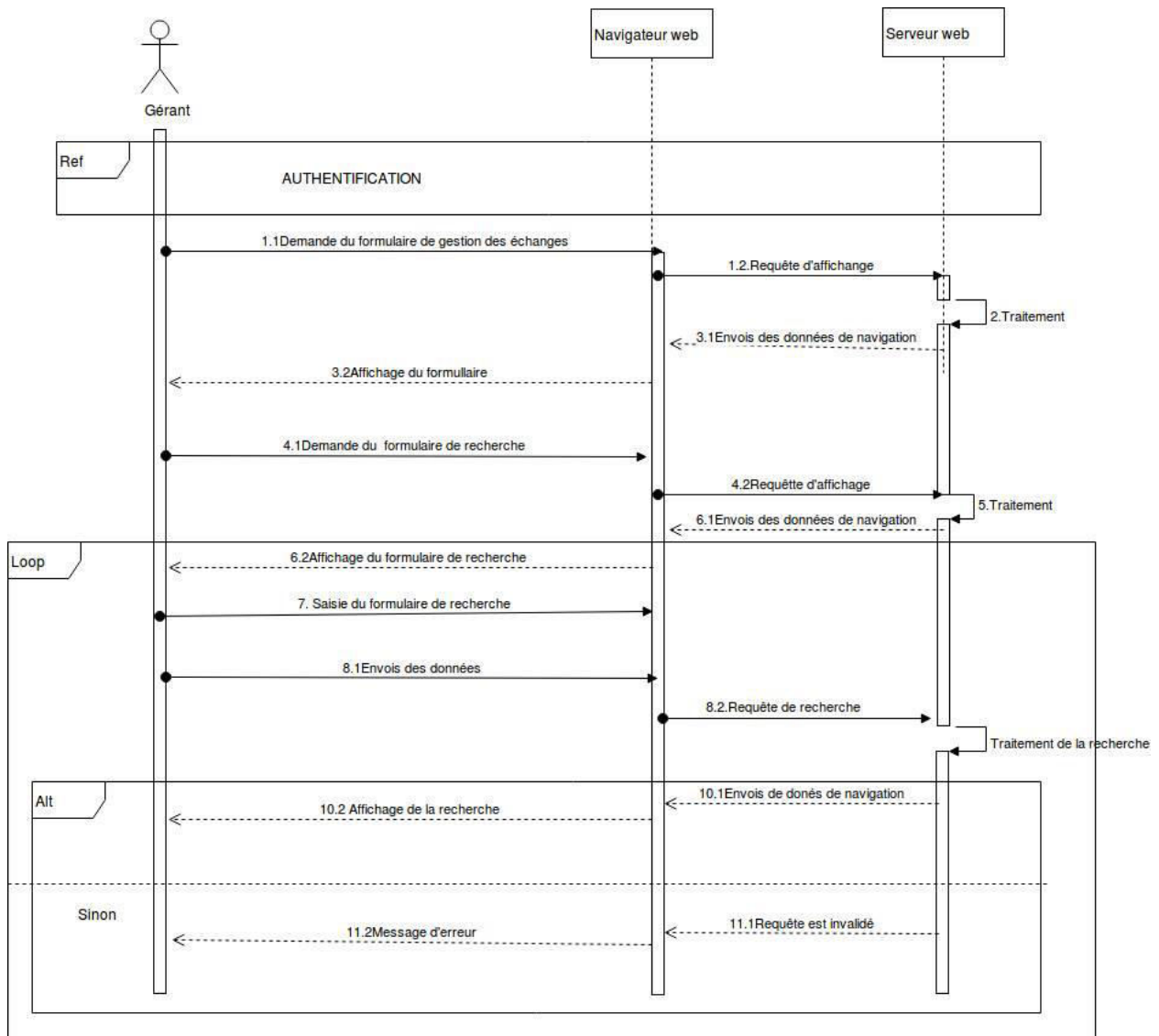


Figure 2.9 : Diagramme de séquence du cas d'utilisation (gérer les échanges).

Diagramme de séquence N°7: cas d'utilisation consulter l'état des ventes :

Lorsque le gérant veut consulter l'une des nombreuses listes de notre application, il doit tout d'abord accéder à la rubrique état des ventes ou il pourra sélectionner une des listes (liste des ventes). Le serveur envoie les données requises dès que la requête d'affichage fini son traitement.

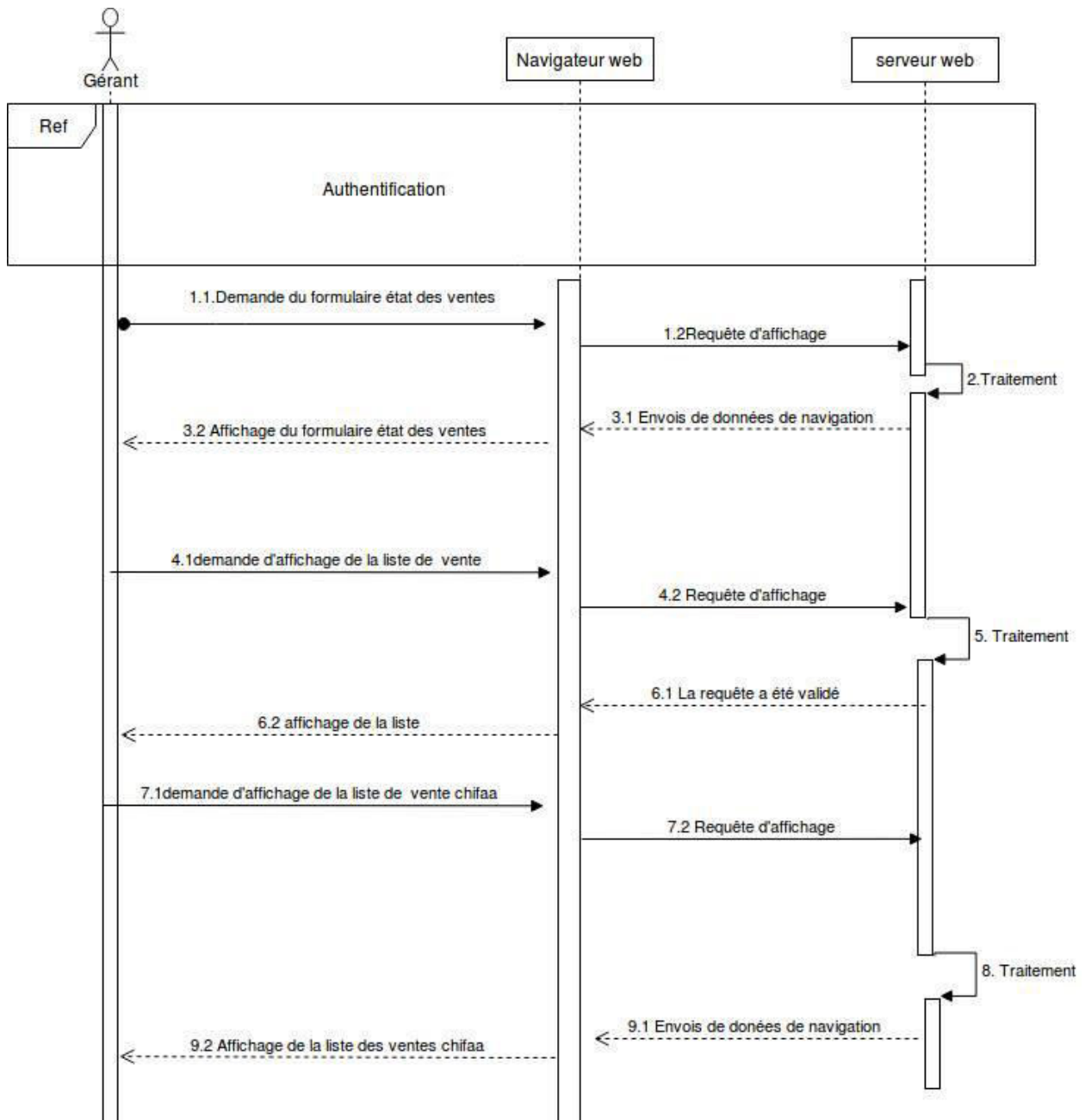


Figure 2.10 : Diagramme de séquence du cas d'utilisation (État des ventes).

2.8) Réalisation du diagramme de classe

2.8.1) Diagramme de classes :

Le diagramme de classes est sans doute le diagramme le plus important à représenter pour les méthodes d'analyse orientées objet. En effet, il permet de spécifier QUI intervient à l'intérieur du système.

Un diagramme de classes fait abstraction des aspects dynamiques et temporels du système, il va permet de représenter une vue statique du système d'information. Il s'agit plutôt des relations entre les classes, des services rendus et utilisés par chacune d'elles et de l'articulation de l'ensemble [8].

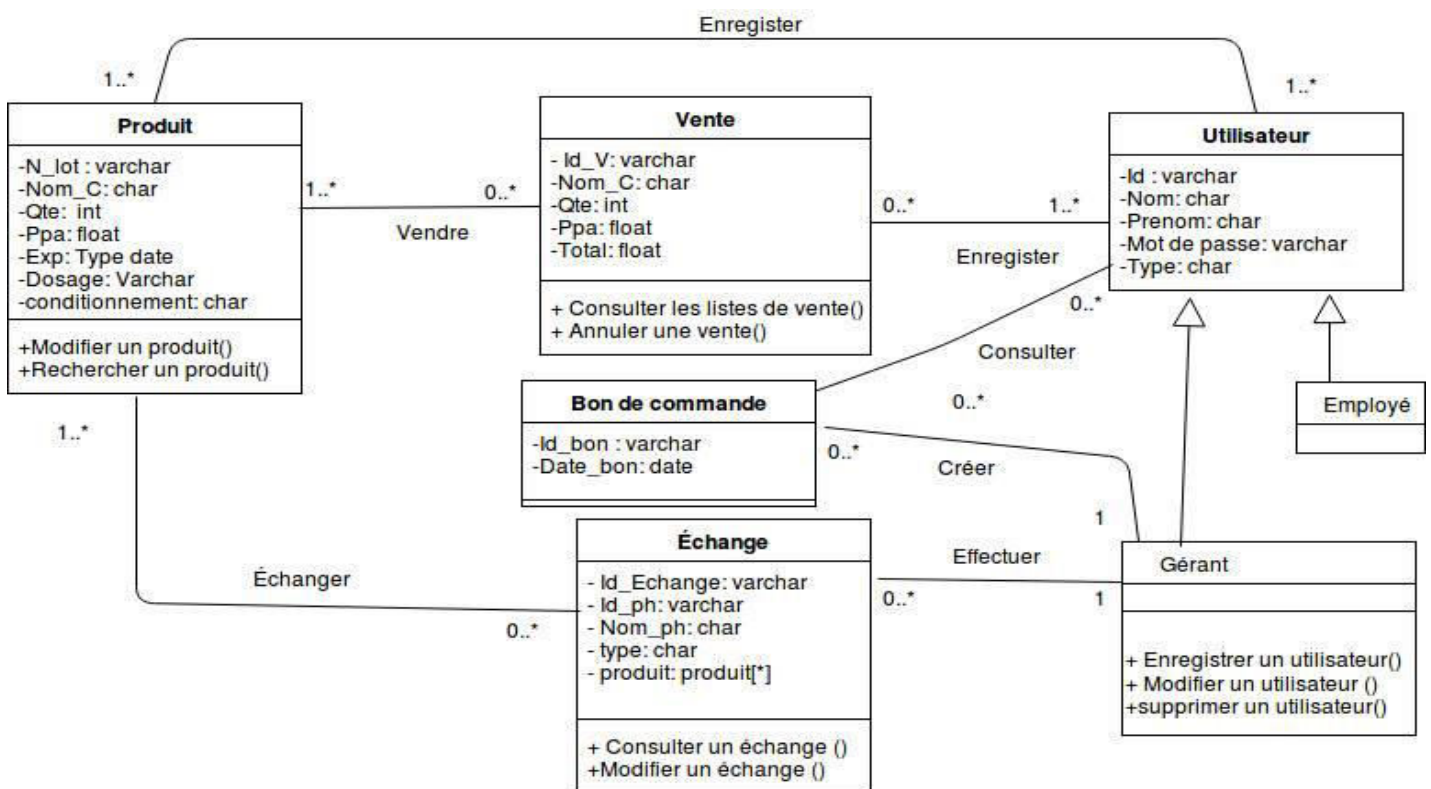


Figure 2.11 : Diagramme de classes.

2.8.2) Dictionnaire de données :

Classe	Attributs	Signification	Type	Taille
Utilisateur	-Nom -Prénom -Id -Mot de passe -Type	-Nom de l'utilisateur. -Prénom de l'utilisateur. - Identifiant -Clé de sécurité -Gérant/Employer	-Caractère -Caractère -Alpha-num -Alpha-num -Caractère	20
Produit	-Nom_C -N_Lot -EXP -PPA -Qte -Dosage -Conditionnement	-Nom commercial. -Numéro de lot -Date d'expiration - Prix populaire Algérien -Quantité -Dosage -Conditionnement	-Caractère -Alpha-num -Date - réel -Entier -Caractère -Caractère	20
Echanges	-Nom_Ph -Non_C -Qte -PPA -Id_Ph	-Nom pharmacie -Nom commercial -Quantité -Prix populaire Algérien -Identifiant pharmacie	-Caractère -Caractère -Entier -Réel -Alpha-num	20
Vente	Id_v Qte Ppa Total	Identifiant vente Quantité Prix populaire Algérien Total d'une vente	Alpha-numérique Entier Réel Réel	20
Bon de commande	Id_Bon Date_bon	Identifiant Bon Date de bon de commande	Alpha-numérique Date	20

Tableau 2.8 : Représentatif du dictionnaire de données.

2.8.3) Le modèle relationnel :

Le concepteur d'une base de données relationnelle doit élaborer un schéma relationnel de la base de données.

Cette activité consiste à définir toutes les relations de la base de données et leurs attributs.

2.8.3.1) Règles de passage au modèle relationnel:

Les règles utilisées pour le passage du diagramme de classes de notre application web au modèle Relationnel sont les suivantes :

- Toute entité devient une relation ayant pour clé primaire son identifiant [9].
- Chaque propriété se transforme en attribut [9].
- Toute association non hiérarchique (de type [n, n] ou de dimension > 2) devient une relation. La clé primaire est formée par la concaténation (juxtaposition) de l'ensemble des identifiants des entités reliées. Toutes les propriétés éventuelles deviennent des attributs qui ne peuvent pas faire partie de la clé [9].

2.8.3.2) Modèle relationnel:

Après avoir appliqué les règles de passage cité précédemment, nous avons abouti au schéma relationnel de la base de données suivant :

Utilisateur (Id, Nom, Prénom, Mot de passe, Type),

Produits (N lot, Nom_c, EXP, PPA, Qte, Dosage, Conditionnement),

Vente (Id V, Nom_C, Qte, Ppa, Total),

Échange (Id Echange, Id ph, Nom_ph, Type, Produit, #Id_gérant),

Bon de commande (Id_bon, date_bon, #Id_gérant)

Échanger (#N lot, #Id Echange, #d ph, Produit, Type),

Vendre (#N lot, #Id V),

Enregistrer (#id, #Id v),

Consulter (#Id, #Id_bon)

Enregistrer (#Id, #N lot).

2.9) Conclusion:

Dans ce chapitre, nous avons modélisé notre application en apportant des réponses à nos questions de modélisation et de conception. En s'appuyant sur l'analyse des besoins de notre application. Nous avons pu là modéliser sous forme de diagrammes de cas d'utilisation puis de séquence, puis concevoir notre application sous forme de diagrammes de classe et schémas relationnelle, reste à concevoir et à créer notre application.

3.1) Introduction :

Dans ce chapitre dédié à l'étude technique et à l'implémentation, nous avons commencé à définir les outils de développement utiliser pour l'implémentation de notre application. Ensuite nous passerons à la présentation de l'application puis on finira par une conclusion.

3.2) Application web :

Une application web désigne un logiciel applicatif hébergé sur un serveur et accessible via un navigateur web. Contrairement à un logiciel traditionnel, l'utilisateur d'une application web n'a pas besoin de l'installer sur son ordinateur. Il lui suffit de se connecter à l'application à l'aide de son navigateur [10].

3.3) Avantages d'une application web :

1. Accès universel depuis n'importe quel type de poste : PC, portables, téléphone mobile, tablette ;
2. Aucune incompatibilité de système d'exploitation (il suffit d'avoir un navigateur) ;
3. Travailler depuis n'importe quel endroit de la planète ;
4. Les données sont centralisées ;
5. Les données sont disponibles 24h sur 24 et 7j sur 7 ;
6. Aucun risque de perte de données.

3.4) Outils de développement :

3.4.1) Développement des diagrammes :

Pour Réaliser les diagrammes UML qui ont servis a modélisé notre application web, nous avons utilisé un logiciel et une application web de développement de diagramme, qui sont:

- **Umbrello** est un outil de modélisation de langage de modélisation unifiée (UML) et un générateur de code. Développer par KDE sous linux.
Il peut créer des diagrammes de logiciels et d'autres systèmes dans le format UML standard de l'industrie, et peut également générer du code à partir de diagrammes UML dans une variété de langages de programmation.

Draw.io est une application web de création de diagramme compatible avec google drive et complètement gratuite, elle permet de dessiner des diagrammes UML en ligne via un navigateur (sou protocole http).

3.4.2) NetBeans :

NetBeans est un environnement de développement intégré (EDI), placé en Open Source par Sun en juin 2000. En plus de Java, NetBeans permet également de supporter différents autres langages, comme C, C+, JavaScript, PHP, HTML. Il comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages Web).

Conçu en Java, NetBeans est disponible sous Windows, Linux, Solaris, Mac OS X ou sous une version indépendante des systèmes d'exploitation (requérant une machine virtuelle Java) [12].

3.4.3) PhpMyAdmin :

PhpMyAdmin est un outil logiciel gratuit écrit en PHP, destiné à gérer l'administration de MySQL sur le Web. PhpMyAdmin prend en charge une large gamme d'opérations sur MySQL et MariaDB. Les opérations fréquemment utilisées (gestion des bases de données, des tableaux, des colonnes, des relations, des index, des utilisateurs, des autorisations, etc.) peuvent être effectuées via l'interface utilisateur, alors que vous avez toujours la possibilité d'exécuter directement une instruction SQL [13].

3.4.4) MYSQL :

MySQL est une base de données relationnelle libre qui a vu le jour en 1995 et très employée sur le Web, souvent en association avec PHP (langage) et Apache (serveur web). MySQL fonctionne indifféremment sur tous les systèmes d'exploitation (Windows, Linux, Mac OS notamment). Le principe d'une base de données relationnelle est d'enregistrer les informations dans des tables qui représentent des regroupements de données par sujets (table des produits, table d'utilisateur par exemple). Les tables sont reliées entre elles par des relations [14].

3.4.5) Mozilla Firefox :

Mozilla Firefox est un navigateur web libre et gratuit, développé et distribué par la Mozilla Foundation avec l'aide de milliers de bénévoles grâce aux méthodes de développement du logiciel libre (open source) et à la liberté du code source [15].

3.5) Langage de programmation :**3.5.1) PHP :**

PHP signifie Personal Home Page, c'est un langage incrusté au HTML et interprété ou compilé côté serveur. Il dérive du C et du Perl dont il reprend la syntaxe.

Ce langage est principalement utilisé pour produire un site web dynamique. Il est courant que ce langage soit associé à une base de données, tel que MySQL.

Exécuté du côté serveur (l'endroit où est hébergé le site) il n'y a pas besoin aux visiteurs d'avoir des logiciels ou plugins particulier.

Comme il supporte tous les standards du web et qu'il est gratuit, il s'est rapidement répandu sur la toile. PHP peut être installé sur les principaux serveurs web du marché.

Néanmoins, les web masters qui souhaitent développer un site en PHP doivent s'assurer que l'hébergeur prend en compte ce langage. Lorsqu'une page PHP est exécutée par le serveur, alors celui-ci renvoie généralement au client (aux visiteurs du site) une page web qui peut contenir du HTML, XHTML, CSS [16].

3.6) Framework :

Un framework est, comme son nom l'indique en anglais, un "cadre de travail". L'objectif d'un framework est généralement de simplifier le travail des développeurs informatiques, en leur offrant une architecture "prête à l'emploi" et qui leur permette de ne pas repartir de zéro à chaque nouveau projet. Il constitue une base cohérente et regroupe en général les fondations d'un logiciel informatique ou d'une application web [17].

3.6.1) Yii framework

Yii signifie (Yes it is) est un framework PHP basé sur des composants ultra performants qui a été développé pour créer des applications Web de grande qualité. Yii a été conçu autour de composants dans le but d'accélérer le développement d'applications Web en simplifiant la réutilisation de code existant [18].

3.7) Architecture global de l'application:

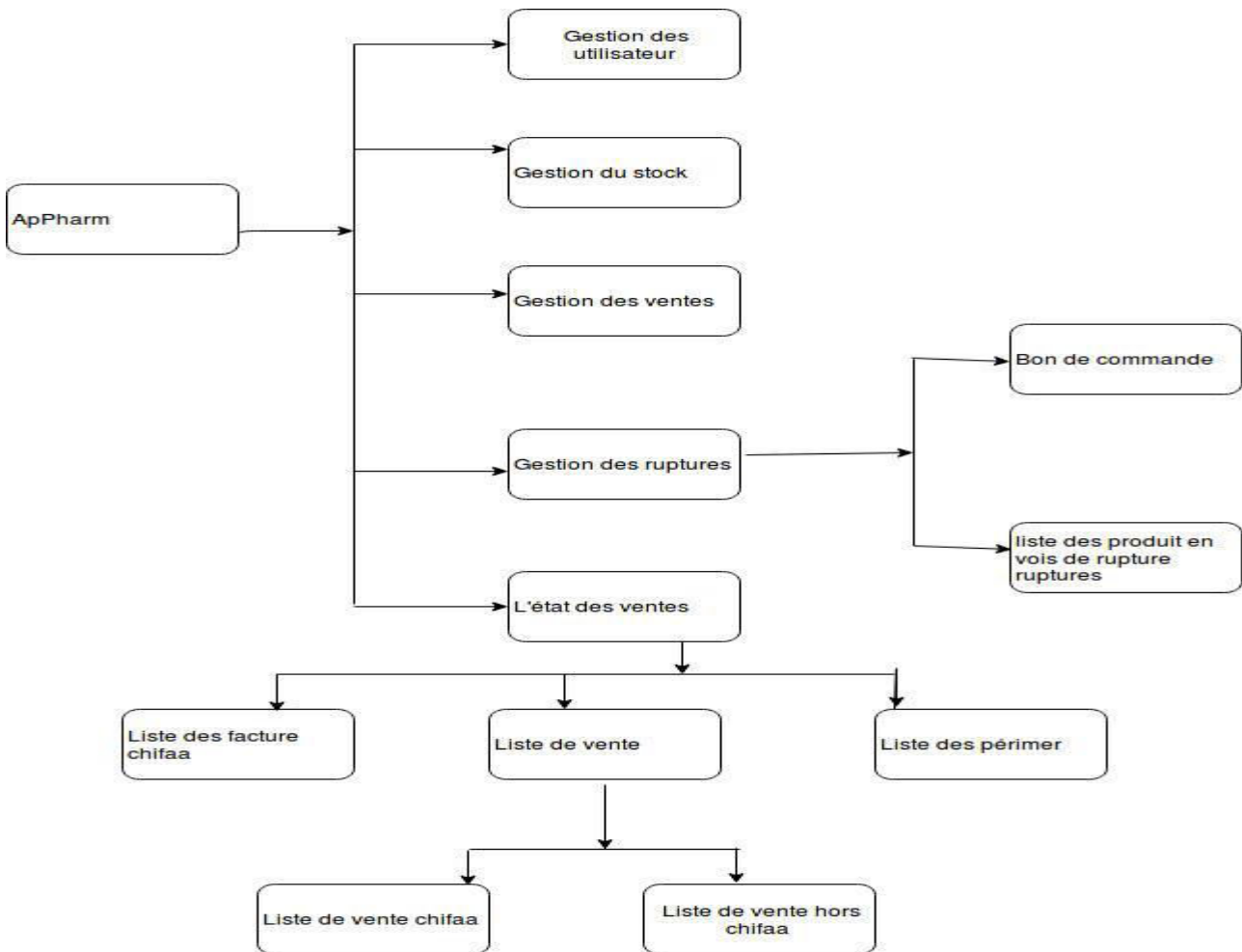


Figure 3.1: Représentation de l'architecture globale de l'application.

3.8) Représentation des interfaces de l'application :

3.8.1) Interface d'Authentification :

Les membres du personnel doivent impérativement se connecter à leur session via cette interface.

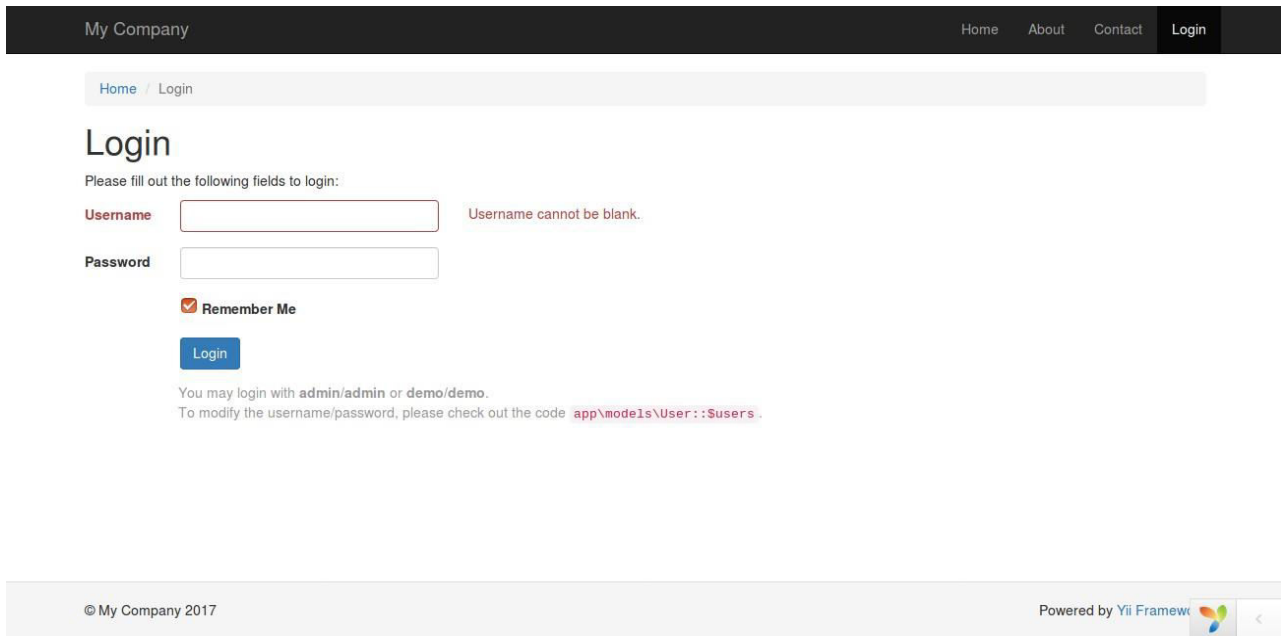


Figure 3.2 : Interface d'authentification.

3.8.2) Interface de création de Produit :

Cette figure représente l'interface de création d'un produit c'est a dire l'ajout au stock d'un produit pharmaceutique.

The screenshot shows a web application interface for creating a product. At the top, there is a dark navigation bar with 'My Company' on the left and 'Home', 'About', 'Contact', and 'Login' on the right. The main content area is titled 'Create Produit'. Below the title, there are seven input fields, each with a label to its left: 'N Lot', 'Non C', 'Qte', 'Ppa', 'Exp', 'Dosage', and 'Conditionnement'. At the bottom left of the form is a green 'Create' button. At the bottom right, there is a small icon with four colored dots (red, yellow, green, blue) and a grey arrow pointing left.

Figure 3.3 : Interface d'ajout d'un produit.

3.8.3) Interface de recherche d'un produit :

Cette capture représente l'interface de recherche de produit existant dans le stock.

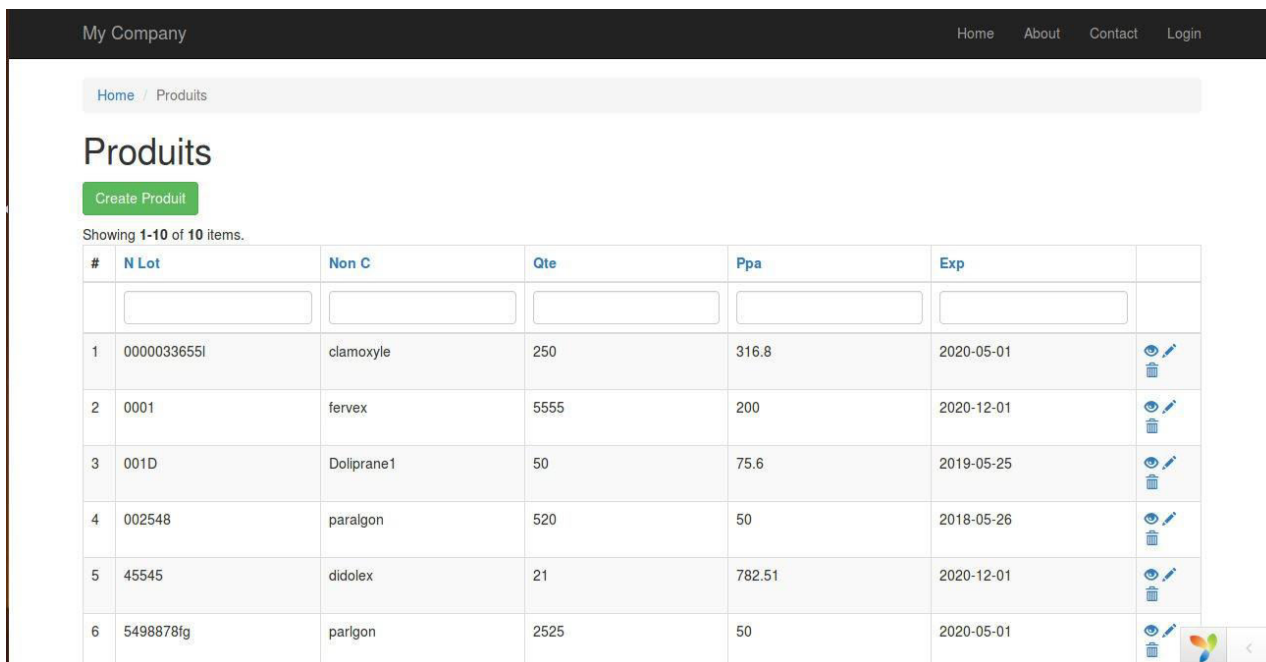


Figure3.4 : Représentation de l'interface de recherche de produit.

3.8.4) Interface de création d'échange :

Cette capture représente l'interface de création d'un échange de produit entre pharmacie

My Company Home About Contact Login

Home / Deals / Create Deal

Create Deal

Id Echange

Id Ph

Nom Ph

Type

Non C

Ppa

Qte

Figure3.5 : Interface de création d'échange.

3.8.5) Interface de recherche d'échange :

Cette figure représente l'interface de recherche d'échange de produits entre pharmacie.

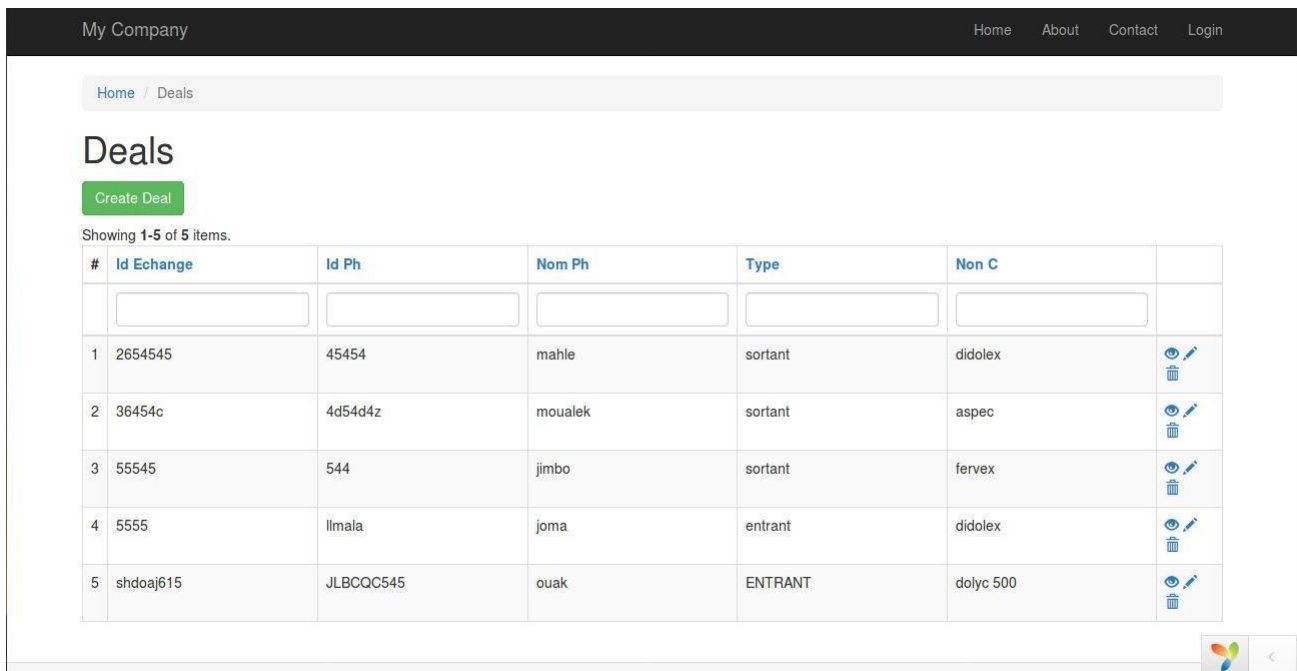


Figure3.6 : Interface de recherche d'échange.

3.9) conclusion :

Au cours de ce chapitre dédié à l'implémentation, nous sommes passés de la théorie à la pratique, elle nous a permis d'atteindre nos objectifs décrits dans les étapes précédentes.

Nous avons présenté les outils qui nous ont aidé à concevoir notre application, le langage de programmation et le framework en passant par la représentation graphique de l'architecture globale de l'application en question et les interfaces pour bien illustrer le travail qui a été fait.

Conclusion général

Au cours de ce travail, nous avons présenté les différentes étapes ayant conduit à la mise en œuvre d'une application web dédiée à la gestion (vente, stock, échange, produit, chifaa) au sein de ' la Pharmacie OUAKKOUCHE' à Bejaia.

Nous avons commencé par recenser les difficultés que rencontre le personnel de la pharmacie afin d'apporter la solution adéquate et spécifier ainsi les besoins.

Le langage de modélisation UML et le processus unifié UP ont constitué le support de l'analyse des besoins et la conception de notre application web via les différents diagrammes UML couvrant les aspects fonctionnels, dynamiques et statiques de tout le développement.

Pour enfin réaliser l'application, nous avons utilisé le 'PhpMyAdmin MYSQL' pour implémenter a base de données et le langage Php via le framwork yii qui est basé sur MVC (Modèles, vue, contrôleurs) pour les interfaces.

Ce projet a fait l'objet d'une expérience intéressante, très bénéfique pour nous. En effet, il nous a permis d'enrichir nos connaissances théoriques et compétences dans le domaine de la conception et de la programmation. Ajoutant à ceci, la mise en application des connaissances acquises tout au long de nos études.

En plus, c'était une bonne occasion pour réaliser un travail concret avec des objectifs clairs et bien définis. Ce projet nous a également permis de nous familiariser avec l'environnement de travail et de la vie professionnelle.

Cependant des perspectives restent envisageables, telles que l'enrichissement de l'application par la fonctionnalité permettant d'établir des statistiques ainsi que l'amélioration de la qualité des renseignements via une application android et ios qui viendra compléter l'application web.

Bibliographie

- [1] PIECHOCKI Laurent UML en français définition d'UML du dictionnaire conception, 2005. Consulter (15 aout 2017).
- [2] G.picard SMA/G2/ENS Mines, définition d'UP Saint-Etienne, 2009. Consulter (15 aout 2017).
- [6] P. Roques, UML2 par la pratique, Eyrolles, 2006. Consulter (20 aout 2017).
- [7] Référence RAD Studio UML 2.0 2015. Consulter (22 aout 2017).
- [8] sylvain Cherrier Diagramme de classes UML 2015. Consulter (25 aout 2017).

Webographie

- [3] <http://dico.developpez.com/html/3020-Conception-MVC-Model-View-Controller.php>. Consulter (16 aout 2017).
- [4] <https://book.cakephp.org/2.0/fr/cakephp-overview/understanding-model-view-controller.html>. Consulter (16 aout 2017).
- [5] <https://www.developpez.net/forums/d751216/general-developpement/alm/architecture/autres/conception-trois-tiers-3-tier/> Consulter (18 aout 2017).
- [9] <http://stephanie.laporte.pagesperso-orange.fr/Pdf/passageMCDMR.pdf> Consulter (26 aout 2017).
- [10] www.ideematic.com. Consulter (27 aout 2017).
- [11] www.larousse.fr/dictionnaires/francais. Consulter (06 septembre 2017).
- [12] <http://dictionnaire.sensagent.leparisien.fr/NETBEANS/fr-fr/> (04 septembre 2017).
- [13] <https://doc.ubuntu-fr.org/phpmyadmin>. (06 septembre 2017).
- [14] <http://www.mosaique-info.fr/glossaire-web-referencement-infographie-multimedia-informatique/m-glossaire-informatique-et-multimedia/448-mysql-definition.html>. (05 septembre 2017).
- [15] <http://dictionnaire.sensagent.leparisien.fr/Mozilla%20Firefox/fr-fr/>. (08 septembre 2017).
- [16] <http://glossaire.infowebmaster.fr/php/>. (11 septembre 2017).
- [17] <https://www.1min30.com/dictionnaire-du-web/framework>. (07 septembre 2017).
- [18] <http://www.yiiframework.com/doc/guide/1.1/fr/quickstart.what-is-yii>. (07 septembre 2017).
- [19] <http://www.memoireonline.com/02/13/7039/Application-web-Gestion-de-pharmacie-en-Java.html> (25 septembre 2017).

Résumé

Au cours de ce mémoire, nous avons conçu et réalisé une application web dédiée à la gestion de pharmacie. Cette application web permet de faciliter le travail du gérant en automatisant la gestion du stock, des ventes et les produit vendus via le logiciel de la Caisse National des Assurances Social (CNAS) "chifaa". Elle permet aussi la création d'une plateforme d'échange entre pharmacies.

Notre application web offre aussi la possibilité au gérant de lister les ventes et les produits en voie de péremption ou de rupture.

Ce travail a été réalisé en utilisant le processus de développement "Processus Unifier" (UP) et le langage de modélisation Unified Modeling Language (UML) afin de schématiser la solution. Nous avons choisi de programmer l'application avec le langage Personal Home Page (PHP) Objet et MYSQL PHPMyAdmin comme Système de Gestion de Base de données (SGBD).

Mots clés : Application Web, PHP, MYSQL PHPMyAdmin, SGBD, UP, UML

Abstract

summary In this brief, we designed and realized a web application dedicated to pharmacy management. This web application makes it possible to facilitate the work of the manager by automating the management of the stock, the sales and the products sold through the software of the National Social Insurance Fund (CNAS) "chifaa". It also allows the creation of a platform for exchange between pharmacies.

Our web application also offers the manager the ability to list sales and products that are expiring or breaking.

This work was carried out using the development process "Process Unifier" (UP) and the modeling language Unified Modeling Language (UML) in order to schematize the solution.

We chose to program the application with the Personal Home Page (PHP) Object and MYSQL PHPMyAdmin as the Data Databases Management System (DBMS).

Keywords: Web application, PHP, MYSQL PHPMyAdmin, DBMS, UP, UML