

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITÉ A. MIRA-BEJAIA



FACULTÉ DES SCIENCES EXACTES
DÉPARTEMENT D'INFORMATIQUE

Mémoire de fin de cycle

En Vue de l'Obtention du Diplôme de Master

Professionnel en Informatique

Option Génie Logiciel

Thème

Conception et réalisation d'une application web
pour la gestion des produits pharmaceutiques au
niveau du CHU de bejaia

Présenté par :

Mlle CHOULLIT Zohra

Mlle YETTOU Lina

Devant le jury composé de :

Président :	Mr. MEHAOUED Kamal	MCB	Université de Béjaïa
Examinatrice :	Mme. SABRI Salima	MCB	Université de Béjaïa
Encadrante :	Mme. BACHIRI Lina	MCA	Université de Béjaïa

Année Universitaire : 2022/2023

Remerciements

Nous tenons à la fin de ce travail à remercier ALLAH le tout puissant de nous avoir donné la santé et de nous avoir permis de mener à terme ce projet .

En premier lieu, nous tenons à remercier notre encadrante Mme Bachiri et notre encadrant Mr Arab au niveau du CHU de Bejaia pour leur disponibilité, leur patience et leur précieux suivi tout au long de la réalisation de ce travail.

Nous tenons à exprimer notre gratitude aux membres du jury pour avoir accepté d'évaluer notre travail.

Nous remercions chaleureusement tous nos enseignants tout au long de notre cursus universitaire pour leurs conseils, leur gentillesse et les connaissances qu'ils nous ont transmises.

Nous remercions nos très chers parents qui sont notre exemple de réussite que nous aimons tant et respectons pour leur soutien moral et financier pendant toute notre scolarité que Dieu leur apporte la santé et une longue vie.

Merci...

Dédicaces

Je dédie ce modeste travail A mes parents que je respecte beaucoup et auquel je voudrais exprimer ici mon amour et ma reconnaissance pour tout le soutien et l'amour qu'ils m'ont apporté depuis mon enfance ,

A mon très cher frère Aissa et ma très chère petite soeur Sara ,

A la mémoire de mon très cher petit frère Adem ,

A tous mes amies et tous ceux qui m'ont aidé .

Lina

Dédicaces

Je tiens à dédier cet humble travail à toute personne m'ayant soutenu et encouragé pour me voir réussir aujourd'hui, en particulier :

A mes parents, les deux êtres les plus chers à mon coeur qui étaient toujours à mes côtés pour me soutenir et m'encourager à me battre sans jamais m'arrêter à mi-chemin, que dieu vous protège,

A mon cher petit frère Zaid,

A mes chères soeurs Wahiba, Thiziri et Warda,

A mes chères grands-mères, que Dieu veille sur vous,

A tous ceux que j'aime et ceux qui m'ont soutenu.

Zohra

Table des matières

Table des matières	i
Table des figures	vi
Liste des tableaux	viii
Liste des abréviations	ix
Introduction générale	1
1 Généralités et Étude de l'existant	3
1.1 Introduction	3
1.2 Réseaux informatiques	3
1.3 Intérêt des Réseaux	4
1.4 Typologies des réseaux	4
1.4.1 Topologie en Étoile	4
1.4.2 Topologie en bus	5
1.4.3 Topologie en Anneau	6
1.5 Architecture des réseaux	6
1.5.1 Architecture client/serveur	6
1.5.2 Architecture poste à poste	7
1.6 Application informatique	8
1.7 Types d'applications informatiques	8
1.7.1 Application desktop	9
1.7.2 Application mobile	9
1.7.3 Application web	10

1.7.3.1	Historique	10
1.7.3.2	Les types d'applications web	10
1.7.4	Principe de fonctionnement des applications Web	12
1.7.5	Exemples d'application web	12
1.7.6	Les avantages d'une application web	13
1.8	Étude préliminaire et analyse des besoins	14
1.8.1	Présentation de CHU de bejaia	14
1.8.2	Situation géographique	14
1.8.3	Structure CHU	15
1.8.4	Les services du CHU	15
1.8.5	Mission des différentes structures du CHU de bejaia	16
1.8.5.1	La direction des ressources humaines	16
1.8.5.2	La direction des finances et du contrôle	17
1.8.5.3	La direction des moyens matériels	17
1.8.5.4	La direction des activités médicales et paramédicales	17
1.8.6	Les objectifs du CHU	18
1.8.7	Présentation cadre d'étude	18
1.8.8	Organigramme du domaine d'étude	19
1.8.9	Étude des postes de travail	20
1.8.9.1	Définition	20
1.8.9.2	Liste des postes de travail à étudier	20
1.8.9.3	Fiche des postes de travail	20
1.8.9.4	Situation informatique	21
1.8.9.5	Moyens humaines	22
1.8.10	Problématique	23
1.8.11	Objectifs de l'étude	23
1.9	Analyse des besoins	24
1.9.1	Les besoins fonctionnels	24
1.9.2	Les besoins non-fonctionnels	24
1.9.3	Nom de l'application	25
1.9.4	Logo de l'application	25
1.10	Conclusion	25

2	Méthodologie et Conception	26
2.1	Introduction	26
2.2	Méthodologie	26
2.2.1	Méthode de conception	26
2.2.1.1	2TUP (TwoTrackUnifiedProcess)	26
2.2.2	Processus de modélisation avec Unified Modeling Language (UML)	28
2.2.3	Obejctifs du langage UML	28
2.2.4	Historique de l'UML	28
2.2.5	Mise a jour d l'UML	29
2.2.6	Présentation d'UML	29
2.2.6.1	Méta-modèle	29
2.2.6.2	Séréotype	30
2.2.6.3	Valeur marquée	30
2.2.7	Points forts d'UML	30
2.2.8	Points faibles d'UML	30
2.2.9	Différents types de diagrammes	30
2.2.9.1	Diagramme de contexte	32
2.2.9.2	Diagramme de cas d'utilisation	32
2.2.9.3	Diagramme de séquence	33
2.2.9.4	Diagramme de classe	34
2.3	Analyse des besoins	34
2.3.1	Acteurs et modélisation de contexte	34
2.3.1.1	Identification des acteurs	34
2.3.1.2	Identification des messages	35
2.3.1.3	Diagramme de contexte	37
2.4	Analyse et étude conceptuelle	37
2.4.1	Diagramme de cas d'utilisation	37
2.4.1.1	Identification des acteurs	38
2.4.1.2	Diagramme Cas d'utilisation pour l'administrateur	39
2.4.1.3	Diagramme Cas d'utilisation pour Pharmacien chef	40
2.4.1.4	Diagramme Cas d'utilisation pour Pharmacien réactif	41
2.4.1.5	Diagramme de cas d'utilisation globale	42
2.4.1.6	Description de Cas d'utilisation	43
2.4.2	Diagramme de séquence	48

2.4.3	Diagramme de séquence «Authentification»	48
2.4.4	Diagramme de séquence «Gestion des comptes»	49
2.4.5	Diagramme de séquence «Gestion des produits»	50
2.4.6	Diagramme de séquence «Ajouter un bon Commande»	51
2.4.7	Diagramme de séquence «Ajouter un bon Livraison»	52
2.4.8	Diagramme de classe	53
2.4.9	Dictionnaire de données	54
2.4.10	Passage au modèle relationnel	55
2.4.10.1	Règles de passage du diagramme de classe au modèle relationnel	56
2.4.11	Conclusion	57
3	Implémentation et réalisation d'application	58
3.1	Introduction	58
3.2	Environnement de développement	58
3.2.1	Visual Studio code	58
3.2.2	Xampp	59
3.2.3	Visual Paradigm	59
3.3	Les outils de développement d'application	59
3.3.1	HTML(Hyper Text Markup Language)	59
3.3.2	Css (Cascading Style Sheets)	60
3.3.3	JS (JavaScript)	60
3.3.4	PHP(Hypertext Preprocessor)	60
3.3.5	JQuery	61
3.3.6	Bootstrap	61
3.3.7	PHPMYADMIN	61
3.3.8	MYSQL	61
3.4	Présentation de l'application	62
3.4.1	Interface d'authentification	64
3.4.2	Espace Administrateur	64
3.4.2.1	Interface accueil	64
3.4.2.2	Interface utilisateurs	65
3.4.2.3	Interface ajout d'un utilisateur	65
3.4.2.4	Interface service	66
3.4.3	Espace pharmacien	66

3.4.3.1	Interface D'accueil	66
3.4.3.2	Interface Bon Commande	67
3.4.3.3	Interface Ajouter un Bon Commande	67
3.4.3.4	Imprimé de bon Commande	68
3.4.3.5	Interface produit	68
3.4.3.6	Interface Ajouter Produit	69
3.4.3.7	Interface Catégories	69
3.4.4	Espace Agent	70
3.4.4.1	Interface D'accueil	70
3.4.4.2	Interface Liste des bons	70
3.4.4.3	Imprimé de Bon Livrison	71
3.5	Conclusion	71
 Conclusion générale		 72
 Bibliographie		 74

Table des figures

1.1	Topologie en étoile	5
1.2	Topologie en bus	5
1.3	Topologie en anneau	6
1.4	Architecture Client-Serveur	7
1.5	La situation géographique du CHU	14
1.6	La structure du CHU de bejaia	15
1.7	Organigramme de la pharmacie centrale du CHU BEJAIA	19
1.8	Organigramme du domaine d'étude	19
1.9	Logo de l'application	25
2.1	Processus 2TUP	27
2.2	Diagrammes UML	31
2.3	Diagramme de contexte de notre système.	37
2.4	Diagramme de cas d'utilisation de l'administrateur	39
2.5	Diagramme de cas d'utilisation de Pharmacien chef	40
2.6	Diagramme de cas d'utilisation de pharmacien réactif	41
2.7	Diagramme de cas d'utilisation globale	42
2.8	Diagramme de séquence de cas d'utilisation «Authentification»	48
2.9	Diagramme de séquence «Gestion des comptes»	49
2.10	Diagramme de séquence de cas d'utilisation «Gestion des produits»	50
2.11	Diagramme de séquence de cas d'utilisation «Ajouter un bon Commande»	51
2.12	Diagramme de séquence de cas d'utilisation «Ajouter un bon Livraison»	52
2.13	Diagramme de classe	53
3.1	Map de l'application- Administrateur.	62

3.2	Map de l'application- Pharmacien (chef/réactif).	63
3.3	Map de l'application- Agent.	63
3.4	Interface d'authentification	64
3.5	Interface Accueil	64
3.6	Interface utilisateur	65
3.7	Interface Ajouter Utilisateur	65
3.8	Interface Service	66
3.9	Interface d'accueil pharmacien	66
3.10	Interface de Bon	67
3.11	Interface Ajouter Bon Commande	67
3.12	Imprimé de Bon Commande	68
3.13	Interface Produits	68
3.14	Interface Ajouter Produit	69
3.15	Interface Catégories	69
3.16	Interface D'accueil Agent	70
3.17	Interface Liste des Bons	70
3.18	Imprimé de Bon Livrison	71

Liste des tableaux

1.1	Fiche d'étude du poste de pharmacien chef	20
1.2	Fiche d'étude du poste de pharmacien réactif	21
1.3	Fiche d'étude du poste administrateur	21
1.4	Imprimantes	22
1.5	Matériel Informatique	22
2.1	Identification des messages	36
2.2	Identification des acteurs	38
2.3	Description du cas d'utilisation "S'authentifier"	43
2.4	Description du cas d'utilisation «gestion des utilisateurs»	44
2.5	Description du cas d'utilisation «gestion des produits»	45
2.6	Description du cas d'utilisation "Ajouter bon Commande"	46
2.7	Description du cas d'utilisation "Ajouter bon Livraison"	47
2.8	Table Utilisateurs	54
2.9	Table Service	54
2.10	Table Bon	54
2.11	Table Bon-élément	55
2.12	Table Catégorie	55
2.13	Table Produit	55

Liste des abréviations

CHU :Centre Hospitalo Universitaire.

S/D :Sous Direction .

2TUP :TwoTrackUnifiedProces .

HTML :HyperText Markup Language .

HTTP :HyperText Transfer Protocol .

CSS :Cascading Style Sheet .

MySQL :My Structured Query Language .

PHP :Hypertext Preprocessor .

SGBD :Système de Gestion de Base des Données .

SQL :Structured Query Language. .

UML :Unified Modeling Language.

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 .

Avant l'invention de l'ordinateur, nous enregistrions 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.

Le secteur pharmaceutique joue un rôle crucial dans la santé et le bien-être de la population, en fournissant des produits pharmaceutiques essentiels pour la prévention, le traitement des maladies. La gestion efficace des produits pharmaceutiques est d'une importance capitale pour garantir la disponibilité, la qualité, l'accessibilité et l'utilisation appropriée des médicaments.

Nous remarquons ainsi la mauvaise gestion 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, Dans cette optique, le développement d'une application web de gestion des produits pharmaceutiques peut constituer une solution prometteuse pour améliorer les processus et optimiser les opérations liées à ces produits.

Ce mémoire se concentre sur l'étude et la conception d'une application web de gestion de produits pharmaceutiques au niveau du CHU de bejaia, visant à faciliter la gestion de l'ensemble du cycle de vie des produits pharmaceutique, depuis leur approvisionnement jusqu'à

leur distribution en passant par leur stockage et leur suivi .Cette application se veut un outil puissant pour le personnels de la pharmacie du CHU leur permettant de gérer de manière efficace et sécurisée les informations relatives aux produits pharmaceutiques.

Pour mener à bien le développement de ce projet, nous avons utilisé UML comme langage de modélisation et «2TUP » TwoTrackUnifiedProces comme méthode de développement agile, pour l'implémentation, le choix du langage de programmation s'est porté sur le langage HTML ,CSS, JAVASCRIPT ; et la base de données sera implémentée avec MySQL.

Notre mémoire est constitué de trois chapitres :

- Le premier chapitre Généralités et Étude l'existant 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 Méthodologie et conception il regroupe toutes les étapes de notre processus de développement en utilisant le langage de modélisation UML et le processus unifié 2TUP .
- Le troisième 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.

Généralités et Étude de l'existant

1.1 Introduction

Les technologies de l'information et de la communication ont été la révolution la plus importante et innovante qui ont marqué ces dernières décennies. En effet, ces technologies nous ont apporté du confort dans notre vie quotidienne par leurs capacités à traiter l'information dans des délais raisonnables parmi ces technologies les applications informatiques qu'on va définir dans une première partie de ce chapitre et faire part de notre choix en revanche la deuxième partie va contenir l'étude préliminaire .

1.2 Réseaux informatiques

Un réseau (network) est un ensemble d'équipements électroniques (ordinateurs, imprimantes, scanners, modems, routeurs, commutateurs...) interconnectés et capables de communiquer (émettre et recevoir des messages) par l'intermédiaire d'un support de communication. Un réseau informatique permet donc l'échange d'informations (messageries, transfert de fichiers, interrogation de bases de données...) et l'accès aux ressources (ou mise en commun, partage) de certains ordinateurs du réseau (matériel tel qu'une imprimante ou modem, logiciels). Un réseau permet donc une économie de coût, un gain de productivité, une utilisation rationnelle des bases de données (consultations, modifications de chaque utilisateur autorisé sur la même base) et une meilleure stratégie dans le domaine de la sécurité (centralisation et sauvegarde éventuellement automatisée des données). Par contre, il a pour inconvénient sa complexité (d'où le recours à un personnel spécialisé en cas de problème) et les conséquences d'une panne sur l'ensemble du réseau.[1]

1.3 Intérêt des Réseaux

Un réseau informatique peut servir plusieurs buts distincts :

- Le partage de ressources (fichiers, applications ou matériels).
- La communication entre personnes (courrier électronique, discussion en direct, etc.).
- La garantie de l'unicité de l'information (bases de données).
- Le jeu vidéo multi-joueurs.

1.4 Typologies des réseaux

La typologie est la manière dont les ordinateurs et les équipements du réseau sont reliés entre eux, il y a trois topologies des réseaux informatiques qui sont les plus reconnus et les plus utilisées :

1.4.1 Topologie en Étoile

Dans une topologie en étoile, les ordinateurs du réseau sont reliés à un système matériel central appelé concentrateur (HUB en anglais) ou un autre appelé commutateur (switch). Ceux-ci ont pour rôle d'assurer la communication entre les différents ordinateurs connectés à lui.[1] Cette topologie présente plusieurs avantages comme :

- Ajout facile de postes .
- Localisation facile des pannes.
- Le débranchement d'une connexion ne paralyse pas le reste du réseau .
- Simplicité éventuelle des équipements au niveau des nœuds : c'est le concentrateur qui est intelligent.

Mais aussi plusieurs inconvénients parmi eux

- Plus onéreux qu'un réseau à topologie en bus (achat du concentrateur et d'autant de câbles que de nœuds) .
- Si le concentrateur est défectueux, tout le réseau est en panne.
- Utilisation de multiples routeurs ou switches afin de pouvoir communiquer entre différents réseaux ou ordinateur.

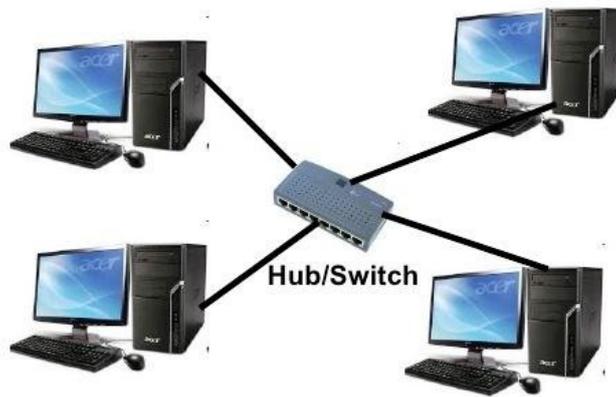


FIGURE 1.1 – Topologie en étoile

1.4.2 Topologie en bus

Dans une topologie en bus, tous les ordinateurs sont connectés sur un même câble. Chaque extrémité du câble est équipée d'un dispositif appelé bouchon de terminaison pour marquer la fin du câble. Dans cette topologie, lorsqu'un ordinateur envoie une information, tous les autres ordinateurs du réseau reçoivent l'information mais seul la machine à qui l'information est destinée va utiliser l'information.[1] Cette structure a plusieurs avantages :

- La défaillance d'un nœud n'entraîne pas l'interruption de la communication avec les autres et il nécessite moins de connexions.
- Il est le plus facile à implémenter

. En revanche, il y a plusieurs inconvénients :

- Deux machines ne peuvent pas communiquer en même temps, En cas de panne du câble, le réseau n'existe plus.
- La vitesse de transmission est faible.

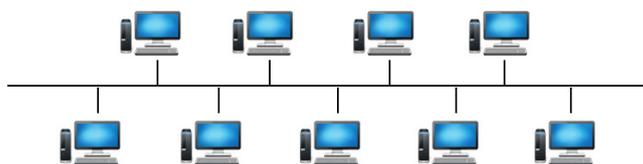


FIGURE 1.2 – Topologie en bus

1.4.3 Topologie en Anneau

Développée par IBM, Dans un réseau possédant une topologie en anneau, les ordinateurs sont connectés à une boucle et communiquent chacun à leur tour. Les informations circulent dans une direction unique, d'un ordinateur à un autre.[1] Un ordinateur n'accepte l'information que si elle lui est destinée. Dans le cas contraire, l'ordinateur en question fait passer l'information à l'ordinateur suivant. Elle présente plusieurs avantages :

- Le signal est régénéré à chaque nœud donc on peut couvrir des distances importantes
- Elle permet de mettre en place des priorités

Mais aussi plusieurs inconvénients :

- On ne peut pas étendre le réseau sans l'interrompre (on rompt le cercle)
- Si l'un des composants ne fonctionne plus, c'est le réseau entier qui est hors service

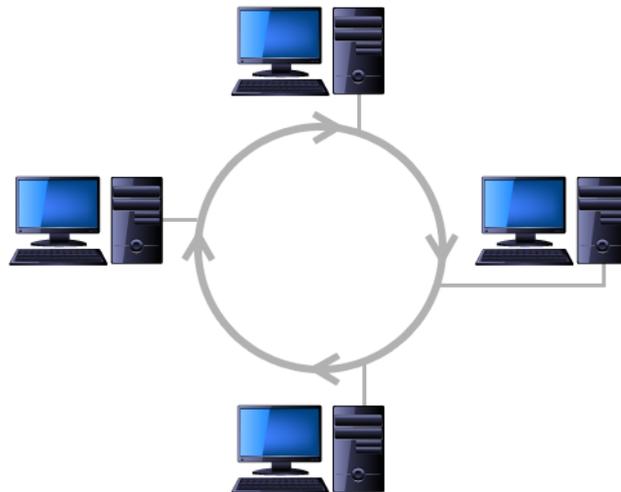


FIGURE 1.3 – Topologie en anneau

1.5 Architecture des réseaux

Il existe deux architectures principales : l'architecture client/serveur et l'architecture poste à poste.

1.5.1 Architecture client/serveur

L'architecture client/serveur désigne un mode de communication entre plusieurs ordinateurs d'un réseau qui distingue un ou plusieurs clients du serveur : chaque logiciel client peut envoyer

des requêtes à un serveur. Un serveur peut être spécialisé en serveur d'applications, de fichiers ou encore de messagerie électronique.[2]

Les postes de travail n'ont seulement besoin que d'un petit bout de logiciel (appelé client) pour se connecter au serveur et ce quel que soit le système d'exploitation installé sur les postes de travail. Cette architecture présente plusieurs avantages comme :

- Des ressources centralisées : étant donné que le serveur est au centre du réseau, il peut gérer des ressources communes à tous les utilisateurs, comme par exemple une base de données centralisée, afin d'éviter les problèmes de redondance et de contradiction.
- Une meilleure sécurité : car le nombre de points d'entrée permettant l'accès aux données est moins important.
- Une administration au niveau serveur : les clients ayant peu d'importance dans ce modèle, ils ont moins besoin d'être administrés.

Mais aussi des inconvénients parmi eux :

- Un coût élevé dû à la technicité du serveur.
- Un maillon faible : le serveur est le seul maillon faible du réseau client/serveur, étant donné que tout le réseau est architecturé autour de lui.

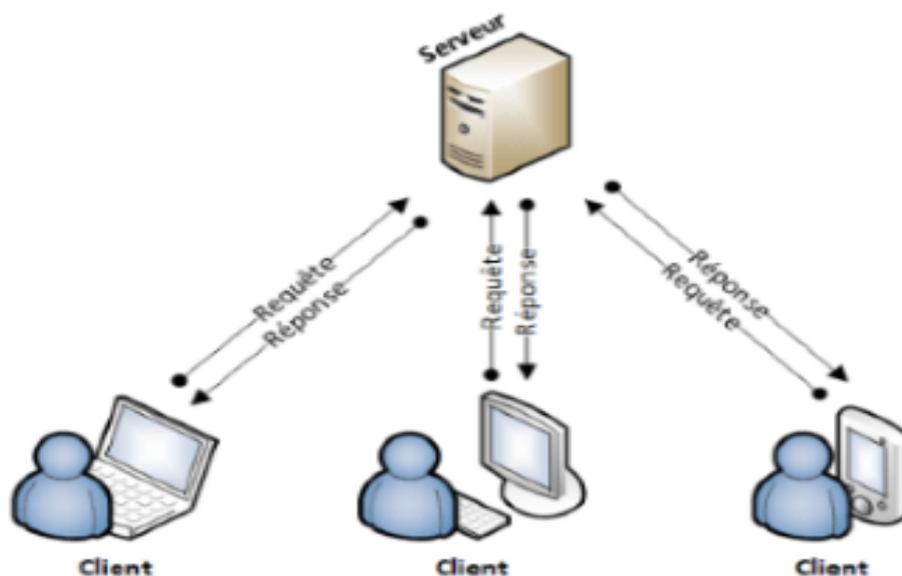


FIGURE 1.4 – Architecture Client-Serveur

1.5.2 Architecture poste à poste

Dans une architecture d'égal à égal (appelée aussi « poste à poste », en anglais peer to peer, notée P2P), contrairement à une architecture de réseau de type client/serveur, il n'y a pas de

serveur dédié. Ainsi chaque ordinateur dans un tel réseau joue à la fois le rôle de serveur et de client. Cela signifie notamment que chacun des ordinateurs du réseau est libre de partager ses ressources. Un ordinateur relié à une imprimante pourra donc éventuellement la partager afin que tous les autres ordinateurs puissent y accéder via le réseau. [3]

les avantages de cette architecture sont :

- Un coût réduit (les coûts engendrés par un tel réseau sont le matériel, les câbles et la maintenance).
- Une simplicité à toute épreuve.

Les inconvénients de cette architecture sont :

- Ce système n'est pas du tout centralisé, ce qui le rend très difficile à administrer
- La sécurité est moins facile à assurer, compte tenu des échanges transversaux ;
- Aucun maillon du système ne peut être considéré comme fiable.

Ainsi, les réseaux d'égal à égal sont préférentiellement utilisées pour des applications ne nécessitant pas un haut niveau de sécurité ni une disponibilité maximale (il est donc déconseillé pour un réseau professionnel avec des données sensibles).

1.6 Application informatique

Une application informatique est un programme informatique conçu pour être exécuté sur un ordinateur, un smartphone, une tablette ou tout autre appareil électronique. Elle est développée pour accomplir une tâche spécifique ou fournir une fonctionnalité particulière à l'utilisateur. Les applications informatiques peuvent être de différents types. Elles sont généralement créées en utilisant des langages de programmation et des outils adaptés à la plateforme sur laquelle elles seront utilisées. [4]

1.7 Types d'applications informatiques

De nos jours, on voit que la technologie ne cesse de se développer, et ce n'est que pour la satisfaction de la population. Un simple exemple sur ça est l'évolution des applications de différents types : personnel, bureau, local et universel.

1.7.1 Application desktop

Une application desktop est une application qui s'exécute complètement sur un seul PC et donc n'a pas besoin d'un accès Web pour fonctionner. L'un de ses avantages est qu'elle est plus rapide et plus stable car ne dépend pas des performances d'une autre application. Par contre, elle a besoin d'être installée sur chaque machine pour être accessible. De plus, elles font face aux problèmes de compatibilité de système d'exploitation.[5]

1.7.2 Application mobile

Une application mobile est un programme téléchargeable qui a été conçu pour fonctionner sur un appareil mobile tel qu'un assistant personnel, un téléphone portable, un smartphone, une tablette ou encore sur certains ordinateurs fonctionnant avec un système d'exploitation Windows phone ou Chrome OS. L'un des avantages d'une application mobile réside sur le fait que votre application une fois installée, L'accès à cette application (via le logo) devient alors simple et rapide. Outre, disposer d'une application mobile, c'est l'occasion d'exploiter des outils inédits comme la géo-localisation. Son principal inconvénient est que, la plupart du temps, elle est difficile à développer car elle doit respecter certaines règles définies par différentes sociétés (Apple pour les applications iOS, Google pour les applications Androïde, Windows pour les applications Windows phone, etc. . .). Un autre inconvénient est le coût lié à son développement qui est assez élevé.[33]

1. Applications mobile natives

Ce type d'application représente la majorité des applications que nous téléchargeons chaque jour. Une application native est une application mobile qui est développée spécifiquement pour un des systèmes d'exploitation utilisé par les Smartphones et tablettes (IOS, Android, etc).[6]

2. application mobile de type hybride

Situé entre l'application mobile et web, l'application hybride ou dite application cross-plateformes est accessible sur toutes les plateformes d'application. Il s'agit d'une application mobile qui fusionne entre les caractéristiques d'une application web (développement en langage HTML 5) et celles de l'application native citées ci-dessus. C'est une technologie multiplateforme, ce qui est un gain de temps considérable si le projet n'est pas trop complexe et ne nécessite pas le développement de plugins spécifiques. Ce genre d'application mobile minimise les charges et la durée de son développement même si cela sera au

détriment du perfectionnement et de la qualité qui caractérise l'application native. Les applications cross-plateformes sont réputées moins performantes et moins fluides que les applications natives.[6]

1.7.3 Application web

Une application web désigne un programme disponible sur un serveur à distance qui réalise des fonctions spécifiques pour son utilisateur. Elle permet aux utilisateurs de gérer des travaux et des documents grâce à leur capacité à fonctionner sur tous les types de navigateurs. Étant donné que les applications web se chargent sur les serveurs web et fonctionnent dans le navigateur, elles ne sont pas à télécharger. Vous y avez donc accès à tout moment et depuis n'importe quel appareil tant que vous avez une connexion internet.[7]

1.7.3.1 Historique

Le World Wide Web est un système de documentation hypertexte créé en 1993 pour les besoins du Centre européen pour la recherche nucléaire (CERN). Le premier navigateur Web (NCSA Mosaic) a été créé la même année par le National Center for Supercomputing Applications (NCSA). Le World Wide Web a permis aux utilisateurs de se partager des documents et des images plus rapidement que via le courrier électronique et plus facilement que via le partage de fichiers. Le nombre grandissant de documents publiés a rendu rapidement les moteurs de recherche nécessaires pour les retrouver : l'utilisateur entre un mot clé, le serveur Web effectue la recherche, puis envoie le résultat sous forme d'un document. Les moteurs de recherche ont été mis en œuvre par extension du serveur Web. En 1995, le NCSA publie la norme industrielle CGI, qui spécifie quelles sont les modalités d'extension d'un serveur Web, dans le but de le brancher avec un logiciel applicatif - par exemple un moteur de recherche. La technologie des applications Web a évolué très rapidement entre 1994 et 2000, où plusieurs logiciels de serveurs Web réservés à cet usage sont sortis sur le marché un après l'autre, ainsi que des améliorations des navigateurs Web.[8]

1.7.3.2 Les types d'applications web

Les applications web peuvent être de différents types tel que :

1. Application web statique

On désigne par application web statique, les applications web de base qui renferment peu de contenu et offrent peu de possibilités de mouvement. Ces applications web sont souvent

conçues en utilisant du CSS et du HTML et peuvent supporter du contenu animé comme les GIFS et les vidéos. Elles se chargent sur le navigateur d'un utilisateur exactement comme elles sont enregistrées sur le serveur web. Ce sont des applications qui manquent de flexibilité et proposent peu ou pas d'interaction. Changer le contenu statique stocké sur les applications web statiques peut être délicat et nécessite d'effectuer de grands changements : télécharger le code HTML, le modifier sur le serveur, etc.

2. Application web dynamique

Les applications web dynamiques constituent des outils qui, à l'inverse des applications web statiques, utilisent des bases de données pour charger leurs informations. Elles changent continuellement et peuvent fournir beaucoup d'informations en fonction de l'interaction avec l'utilisateur. Ce style d'application est techniquement plus compliqué que l'application statique et est mis en place pour faire en sorte que les bases de données ou les forums soient régulièrement capables de se mettre à jour ou de changer l'information disponible. Ceci est généralement possible grâce à l'utilisation d'un CMS ou d'un système de gestion de contenu.[9]

3. Application e-commerce

Si vous vous êtes déjà procuré quelque chose en ligne, vous connaissez éventuellement les applications web les plus répandues, car elles font partie du e-commerce. Une application web qui prend la forme d'un magasin est considérée comme du e-commerce. Le processus de développement de cette application est plus compliqué que les deux précédemment mentionnés, car une application de e-commerce doit accepter des paiements électroniques par carte de crédit, PayPal ou autre mode de paiement. Il est donc conseillé de confier votre projet à des entreprises de développeurs web spécialisées comme Globalis qui se chargeront de réaliser convenablement votre projet d'application web.

4. Application web portail

Une application de portail web renvoie simplement à une application dans laquelle différentes sections ou catégories sont accessibles par l'intermédiaire d'une page d'accueil. Ce type de page se compose à la fois des forums, des chats, des courriels, des navigateurs et ainsi de suite.[9]

5. Application web animée

Les applications web animées doivent obligatoirement être compatibles avec la technologie FLASH. Cette dernière peut être complexe pour quiconque n'ayant pas d'expérience avec ce type de technologie. Toutefois, si vous êtes à la recherche d'une application web

relativement moderne offrant la possibilité d'ajouter du mouvement, il est préférable de travailler avec la technologie FLASH.[9]

1.7.4 Principe de fonctionnement des applications Web

L'architecture des applications Web décrit les interactions entre les applications, les bases de données et les systèmes middleware sur le Web. Les premières applications Web souffraient d'une maniabilité et d'une ergonomie inférieure aux applications client. Est appelée Rich Internet Application une application Web qui offre une maniabilité et une ergonomie équivalente à une application client. Le terme Rich Internet Application est apparu la première fois dans une publication de Macromedia en 2002. Dès que l'utilisateur clique sur le bouton OK après avoir tapé une URL dans la barre d'adresse d'un navigateur Web, il demande cette adresse Web particulière. Le serveur envoie des fichiers au navigateur en réponse à la demande effectuée. Le navigateur exécute ensuite ces fichiers pour afficher la page demandée. Enfin, l'utilisateur peut interagir avec le site Web. La chose la plus importante à noter ici est le code analysé par le navigateur Web. Une application Web fonctionne de la même manière. Ce code peut avoir ou non des instructions spécifiques indiquant au navigateur comment répondre aux différents types d'entrées utilisateur.[9]

1.7.5 Exemples d'application web

La technologie des applications web permet de nombreux usages. Les usages populaires sont les moteurs de recherche, le web mail, le commerce électronique et les jeux en ligne.

- Un moteur de recherche est une application web qui recherche des documents
- Un web mail est une application web pour recevoir et envoyer du courrier électronique
- Un jeu par navigateur est un jeu vidéo réalisé sous la forme d'une application web.
- Facebook est une application web qui permet à chaque utilisateur de se constituer un réseau social (amis, associés, personnes qui partagent les mêmes centres d'intérêt).

1.7.6 Les avantages d'une application web

1. Maîtrise de votre budget et diminution des coûts :

- La mise de départ est inexistante.
- Aucune mise à niveau de votre infrastructure.
- Le budget est prévisible par un abonnement mensuel fixe.
- Investissement étalé dans le temps.
- Vous ne payez que pour les fonctionnalités que vous utilisez.
- La facturation est proportionnelle à votre consommation réelle.
- Vous faites une économie sur le matériel informatique.

2. Gain de temps :

- La mise en œuvre et le déploiement sont plus rapides.
- La circulation et le partage des données entre utilisateurs sont optimisés.
- Vous évitez les sollicitations de personnels informatiques.
- Utilisation des applications plus intuitive et plus facile.

3. Accessibilité optimisée :

- Accès universel depuis n'importe quel type de poste : PC, portables, téléphone
- Aucune incompatibilité de système d'exploitation (il suffit d'avoir un navigateur)
- Vous pouvez travailler depuis n'importe quel endroit de la planète
- Vos données sont centralisées

4. Meilleure gestion de la sécurité :

- Vous profitez des moyens des grandes infrastructures.
- Vos accès aux données sont contrôlés par identification et certificats.
- Sauvegardes automatiques.
- Hébergement de vos données dans un cadre contractuel de confidentialité.

5. Évolution et innovation continue :

- La mise à niveau des applications est automatique et transparentes.
- Votre retour d'expérience est directement pris en compte pour des améliorations et innovations permanentes.
- Vous bénéficiez toujours de la version la plus récente. [9]

1.8 Étude préliminaire et analyse des besoins

1.8.1 Présentation de CHU de bejaia

Le Centre Hospitalier universitaire (CHU) est un établissement public à caractère administratif, doté de la personnalité morale et de l'autonomie financière. Il a été créé par décret exécutif N 09-319 du 06.10.2009, sur proposition conjointe du ministre chargé de la santé et du ministre chargé de l'enseignement supérieur et de la recherche scientifique. Le CHU est chargé, en relation avec l'établissement d'enseignement et/ou de formation supérieur en sciences médicale concerné, des missions : de diagnostic, d'exploration, de soin, de prévention, de formation, d'étude et de recherche.[10]

Le CHU de Bejaia est composé de :

- Hôpital Khalil Amrane.
- Hôpital Frantz Fanon.
- Hôpital Targua Ouzemmour.

1.8.2 Situation géographique



FIGURE 1.5 – La situation géographique du CHU

1.8.3 Structure CHU

Le CHU de Bejaia est composé de plusieurs services médicaux spécialisés répartis dans les trois hôpitaux :

Frantz fanon : Frantz fanon sise à l'ancienne ville, bordj moussa sa capacité d'accueil est de 85 lits, avec un effectif de 242 personnes .

Targa Azemmour : L'unité Targa Ouzemour (clinique maternité), située au village tala merkha. Sa capacité d'accueil est de 85 lits, avec un effectif de 235 personnes, et 03 mis en disponibilité

Khelil Amrane : L'administration générale du CHU se trouve à cet hôpital. Il à une capacité de 240 lits.

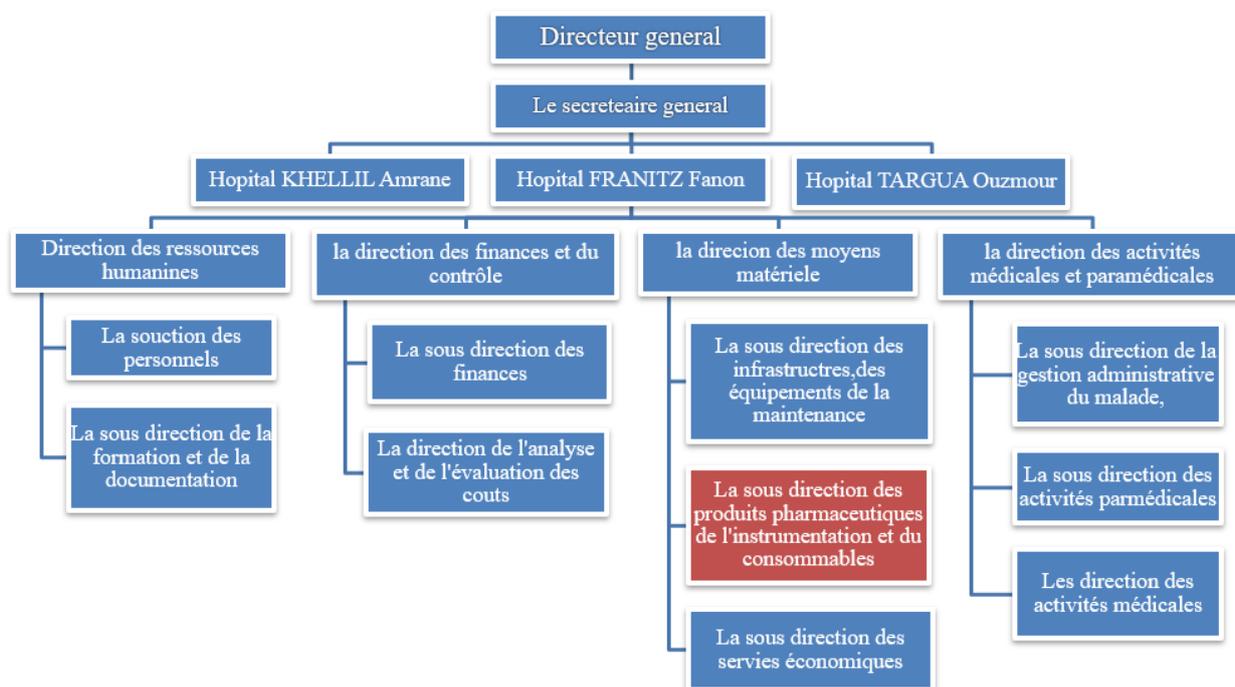


FIGURE 1.6 – La structure du CHU de bejaia

1.8.4 Les services du CHU

Le CHU se compose des services suivant :

- Service de médecine interne.
- Service de chirurgie.

- Service de réanimation.
- Pavillon des urgences médico-chirurgicales et pédiatries.
- Le bloc opératoire.
- Laboratoire d'analyse médicale.
- Service de radiologie.
- Une pharmacie centrale.

1.8.5 Mission des différentes structures du CHU de bejaia

Le CHU de bejaia comprend une direction générale, un secrétariat général et quatre directions. Chaque direction comprend un ensemble de sous directions et de bureaux.

- La direction générale est chargée d'assurer la gestion de l'hôpital.
- Le secrétariat général est chargé de gérer les différentes sous directions et les différents bureaux.

Le CHU se compose des différentes directions et sous directions :

1.8.5.1 La direction des ressources humaines

La direction des ressources humaines comporte deux sous-direction des personnels et la sous-direction de la formation et de la documentation.

— La sous-direction des personnels

La sous-direction des personnels se compose de trois bureaux :

- Le bureau de la gestion des carrières des personnels administratifs, techniques et de service.
- Le bureau de la gestion des personnels médicaux, paramédicaux et psychologues.
- Le bureau des effectifs, de la régulation et de la solde.

— La sous-direction de la formation et de la documentation

La sous-direction de la formation et de la documentation endosse plusieurs activités de formation et de formation continues : médicale, paramédicale, administrative, technique, perfectionnement, recyclage, formation de court durée à l'étranger, encadrement et suivi des stagiaires de différents instituts, suivi et formation des enfants hospitalisée, organisation des manifestation scientifiques.

1.8.5.2 La direction des finances et du contrôle

La direction des finances et du contrôle comprend :

— **La sous-direction des finances**

Elle comporte les bureaux suivants :

- Le bureau du budget et de la comptabilité évactions budgétaires ;
- Le bureau des recettes et des caisses

— **La sous-direction de l'analyse et de l'évaluation des coûts**

Elle comporte :

- Le bureau de facturation

1.8.5.3 La direction des moyens matériels

Cette direction comprend :

— **La sous-direction des services économiques** Elle comporte les bureaux suivant :

- Bureau des approvisionnements.
- Bureau de la gestion des magasins,des inventaires et des reformes.
- Bureau de la restauration et de l'hôtellerie.

— **La sous-direction des produits pharmaceutiques**

Elle comporte les bureaux suivants :

- Bureau de produits pharmaceutiques.

— **La sous-direction d'infrastructures,des équipements et de la maintenance**

Elle comporte les bureaux suivants :

- Bureau des infrastructures.
- Bureau des équipements.
- Bureau de la maintenance.

1.8.5.4 La direction des activités médicales et paramédicales

Cette direction est composée de trois(03) sous directions :

— **La sous-direction de la gestion administrative du malade**

Elle est composée de deux(02) bureaux :

- Bureau des entrées(admissions)

- Le bureau de l'accueil, de l'orientation et des activités socio- thérapeutiques.

— **La sous-direction des activités médicales**

Elle est composée de trois(03)bureaux :

- Bureau de l'organisation et de l'évaluation des activités médicales.
- Bureau de la programmation et du suivi des étudiants.
- Le bureau des gardes et des urgences.

— **La sous-direction des activités paramédicales**

Elle se compose de trois (3)bureaux :

- Bureau de la programmation et du suivi des étudiants
- Bureau des soins infirmiers
- Bureau de l'évaluation et de l'organisation de l'activité paramédicale.

1.8.6 Les objectifs du CHU

- Soins a haut niveau
- Formation médicale et soins infirmier
- Recherches en science de la santé
- Soutenir la mise en oeuvre des schémas régionaux d'organisation et de suivis, d'accompagner les recompositions internes, le regroupement de plateaux techniques, les partenariats entre les établissement publics et privés.
- Développer les systèmes d'informations, de la communication et audio-visuel.
- Soutenir les opérations répondant aux critères d'efficience.
- Assurer les mises aux normes de sécurité .

1.8.7 Présentation cadre d'étude

Nous avons effectué notre stage au niveau de la S/D produits pharmaceutiques du CHU qui se charge de commander, réceptionner, stocker et distribuer les produits pharmaceutique. La S/D produits pharmaceutique assure la distribution des produits pharmaceutiques pour les unités TARGA OUZMMOUR ,Hopital FRANTZ FANON et les services de KHELIL AM-RANE, comme le montre l'organigramme suivant :

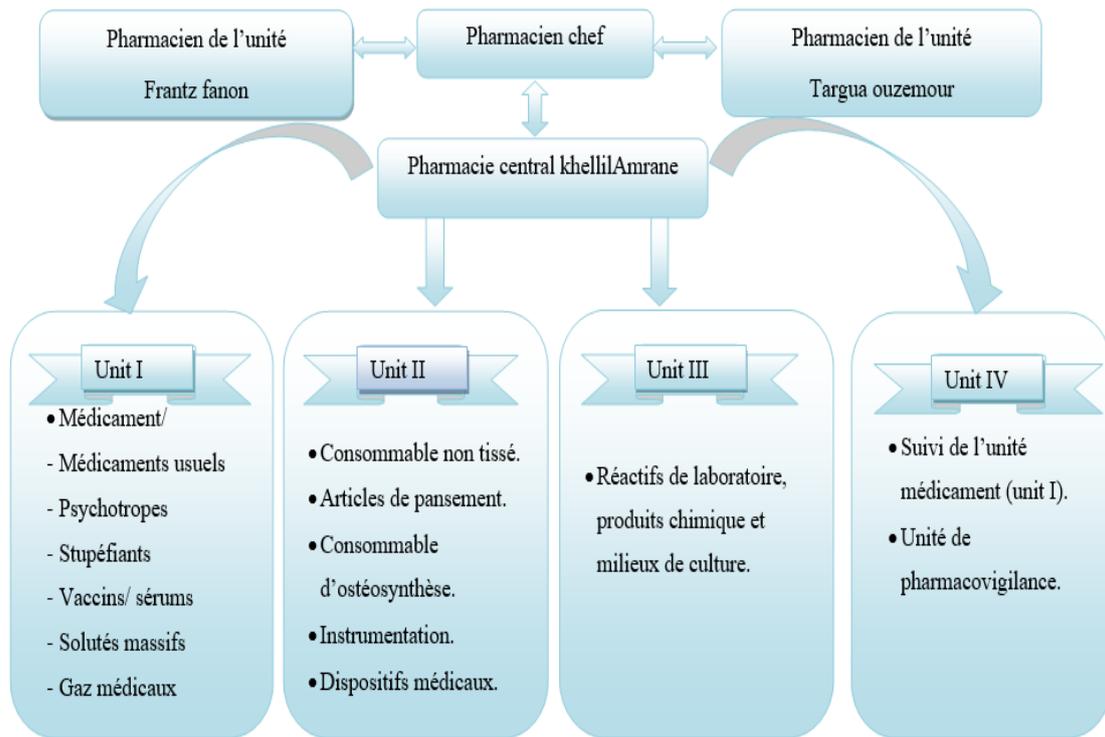


FIGURE 1.7 – Organigramme de la pharmacie centrale du CHU BEJAIA

1.8.8 Organigramme du domaine d'étude

Les tâches effectuées au niveau de la S/D produits pharmaceutiques (commande, réception, distribution et stockage) sont assurées par les postes suivant :

- Pharmacien chef.
- Les pharmaciens.
- L'administrateur.



FIGURE 1.8 – Organigramme du domaine d'étude

1.8.9 Étude des postes de travail

1.8.9.1 Définition

Le poste de travail est un emplacement où s'effectue une phase d'un travail avec des équipements nécessaires à l'accomplissement de ce travail.

1.8.9.2 Liste des postes de travail à étudier

- Pharmacien chef
- Pharmaciens (consommable, médicaments, réactif)
- Administrateur.

1.8.9.3 Fiche des postes de travail

— Pharmacien chef

Fiche d'étude du poste de travail N° 01		
Désignation du poste : pharmacien chef Service de rattachement : pharmacien centrale Effectif : 01 Objectif du poste : Assurer la bonne gestion des produits pharmaceutiques.		
N°	Taches	Fréquence
01	Commande, livre et réceptionne les produits pharmaceutique	Journalière
02	Assure la gestion du stock et l'organisation des inventaires	Journalière
03	Assure les états de fiche de stock	Journalière

TABLE 1.1 – Fiche d'étude du poste de pharmacien chef

— **Pharmacien réactif**

Fiche d'étude du poste de travail N° 02		
Désignation du poste :pharmacien		
Service de rattachement :pharmacien centrale		
Effectif :02 (un pour médicaments et un pour réactif)		
Objectif du poste :Gestion des produit pharmaceutiques.		
N°	Taches	Fréquence
01	Il assure la commande et la livraison des produits du médicament,réactifs.	Journalière
02	Il assure la gestion du stock et l'organisation des inventaires	Journalière
03	Assure les états de fiche de stock	Journalière

TABLE 1.2 – Fiche d'étude du poste de pharmacien réactif

— **Administrateur**

Fiche d'étude du poste de travail N° 03		
Désignation du poste :Administrateur		
Service de rattachement :pharmacien centrale		
Effectif :03		
Objectif du poste :Enregistrement des informations concernant les produits pharmaceutiques.		
N°	Taches	Fréquence
01	Enregistrer les informations concernant les produits pharmaceutique(entrés /sorties et facteur)	Journalière

TABLE 1.3 – Fiche d'étude du poste administrateur

1.8.9.4 Situation informatique— **Le matériel**

La S/D produits pharmaceutique du CHU dispose de :

Les imprimantes

imprimantes :
03 canon LBP6020B
01EPSON FX-2190

TABLE 1.4 – Imprimantes

Les ordinateurs

Type d'ordinateur	Nombre	Les caractéristiques
Lenovo	3	Processeur :Intel Core i5-8250U Ram :02go Disque dur :500go
Maxipower	1	processeur :pentien Ram :2.59GhZ,0.99go Disque dur :500go
hp	1	Processeur Intel Core i7-8565U Ram :4 Go disque dur :500 Go SATA

TABLE 1.5 – Matériel Informatique

Les logiciels

- Système d'exploitation : Windows 7 professionnel
- Microsoft office 2007(word,excel,access)
- Epifarm : la gestion des entres et sorties des produits pharmaceutiques

1.8.9.5 Moyens humaines

- Administrateur
- Pharmacien

1.8.10 Problématique

Après avoir consulté le logiciel utilisé au niveau de la pharmacie centrale du CHU nous avons constaté plusieurs insuffisances :

- La redondance de donnée : Les informations sont stockées ou saisies de manière répétée ou en double dans le système. Cela peut se produire pour plusieurs raisons, telles que des erreurs de programmation, des processus de saisie manuelle répétitifs La redondance de données peut entraîner des problèmes tels que la confusion des utilisateurs, des incohérences dans les informations, une augmentation de la taille de la base de données et des problèmes de performance.
- Pour chercher un médicament on prend beaucoup de temps ce qui engendre la perte de temps pour le pharmacien réactif
- Manque d'informations pour assurer une bonne distribution des produits pharmaceutique
- La perte de documents :la situation où des informations importantes ou critiques sont supprimées ou deviennent inaccessibles de manière involontaire ou imprévue.Elle peut avoir des conséquences graves.
- Les informations sont non sécurisées .
- Erreurs de remplissage dans les registres.
- Traitement manuel de l'information.
- Mauvaise codification.

1.8.11 Objectifs de l'étude

Notre étude consiste à remédier au mieux possible au défaillance citée en apportant les améliorations suivantes :

- Concevoir une base de données qui englobera toutes les informations et permettra d'éviter la redondance et de faciliter leur recherche.
- Optimisation du temps d'accès au donnée.
- Sécuriser les informations par l'administrateur de la base de données.
- Simplifier le travail administratif par l'automatisation des tâches effectuées manuellement.

1.9 Analyse des besoins

1.9.1 Les besoins fonctionnels

Ce sont les exigences du client spécifiant un comportement d'entrée et de sortie du système.

- **Pour l'administrateur**

- Création des comptes pour les utilisateurs .
- Gestions des utilisateurs .
- Gestion des unités de services .

- **Pour le pharmacien chef** Nous devons regrouper toutes les fonctionnalités nécessaires de gestion pour faciliter et organiser le travail du pharmacien chef. Parmi ces fonctionnalités :

- Possibilité de modifier les informations de son propre compte qui va être créé par l'administrateur.
- Possibilité de faire l'ajout, la modification des informations et la suppression d'un produit.
- Possibilité d'établir un bon de commande et de visualiser les modifications possibles.
- Possibilité d'établir un bon de livraison.

- **Pour le pharmacien réactif** Par rapport aux fonctionnalités du pharmacien réactif elles sont presque les mêmes que le pharmacien chef .

1.9.2 Les besoins non-fonctionnels

Les besoins non fonctionnels décrivent les objectifs liés aux performances du système et aux contraintes de son environnement. Ses exigences techniques sont souvent exprimées sous forme d'objectifs spécifiques que doit atteindre le système. Ainsi, notre application doit pouvoir couvrir les besoins non fonctionnels suivants :

- **Maintenabilité** : Le code source doit être compréhensible par simple lecture, notamment en respectant les règles de gestion et les normes de développement.
- **Exploitabilité** : Les impacts en termes de performances doivent être pris en compte lors du développement, ainsi que la consommation des ressources (i.e., CPU, mémoire, etc.) qui doit être minimisée.

- **Convivialité et ergonomie** : Le système doit être facilement utilisable et disposer d'interfaces conviviales ; le design doit permettre une identification immédiate de ses différents éléments pour permettre à l'utilisateur d'accéder de manière intuitive à ce qu'il cherche, dès la première utilisation.
- **Sécurité** : L'obligation pour accéder à l'application de s'authentifier par un nom d'utilisateur et un mot de passe affectés par l'administrateur.

1.9.3 Nom de l'application

On a décidé de choisir PharmaCHU comme nom pour cette application puisque elle est destinée à la pharmacie du CHU .

1.9.4 Logo de l'application

Le choix du logo a été fait selon les critères du laboratoire :



FIGURE 1.9 – Logo de l'application

1.10 Conclusion

Dans ce chapitre, nous avons défini les applications informatiques et nous avons présenté l'organisme d'accueil encadrant le cadre d'étude, la problématique ainsi que les objectifs recherchés que nous utiliserons comme référence dans le chapitre suivant.

Méthodologie et Conception

2.1 Introduction

Dans ce chapitre, nous présentons notre travail en commençant par la définition des étapes du processus de développement logiciel 2TUP, et la présentation du langage de modélisation UML(Unified Modeling Language), ensuite nous entamerons la modélisation de notre système en mettant en avant les diagrammes des cas d'utilisation , les diagrammes de séquence. Nous finirons par le diagramme de classe et le modèle relationnel.

2.2 Méthodologie

2.2.1 Méthode de conception

En ingénierie, une méthode d'analyse et de conception est un procédé qui a pour objectif de permettre de formaliser les étapes préliminaires du développement d'un système afin de rendre ce développement plus fidèle aux besoins du client.

Pour la réalisation de notre application, notre choix a été porté sur le TwoTrackUnifiedProcess (2TUP).

2.2.1.1 2TUP (TwoTrackUnifiedProcess)

2TUP est un processus de développement logiciel qui implémente le processus unifié (c.à.d. itératif, incrémental, basé sur UML).

Il propose un cycle de développement qui sépare les aspects techniques des aspects fonctionnels en partant du constat que toute évolution peut se traiter parallèlement, suivant un axe fonctionnel et un axe technique. Ensuite, et en fusionnant les résultats de ces deux axes (branches),

on arrive à réaliser le système désiré ; ce qui nous donne un cycle de développement sous forme de Y.

Le processus de développement en Y articule autour de trois branches :

- **Branche fonctionnelle** : (la branche gauche du Y) qui consiste en la modélisation et le maquettage, dans le but de clarifier les besoins fonctionnels. Ceci permet d'étudier d'une manière pointue la spécification fonctionnelle afin d'obtenir une idée de ce que va réaliser le système en termes de métier. Le fruit de cette branche ne dépend d'aucune technologie.
- **Branche technique** : (la branche droite du Y) qui recense toutes les contraintes à respecter pour réaliser le système. Elle définit ensuite les composants nécessaires à la construction de l'architecture technique. Cette branche dépend au minimum des aspects fonctionnels.
- **Phase conception-réalisation** : (la branche du milieu) : En premier lieu, elle intègre le modèle d'analyse dans l'architecture technique de manière à tracer la cartographie des composants du système à développer et étudie ensuite comment réaliser chaque composant. Arrive par la suite l'étape de codage, qui produit ces composants et teste au fur et à mesure les unités de codes réalisées. Et enfin l'étape de recette, qui consiste à valider les fonctions du système développé.[11]

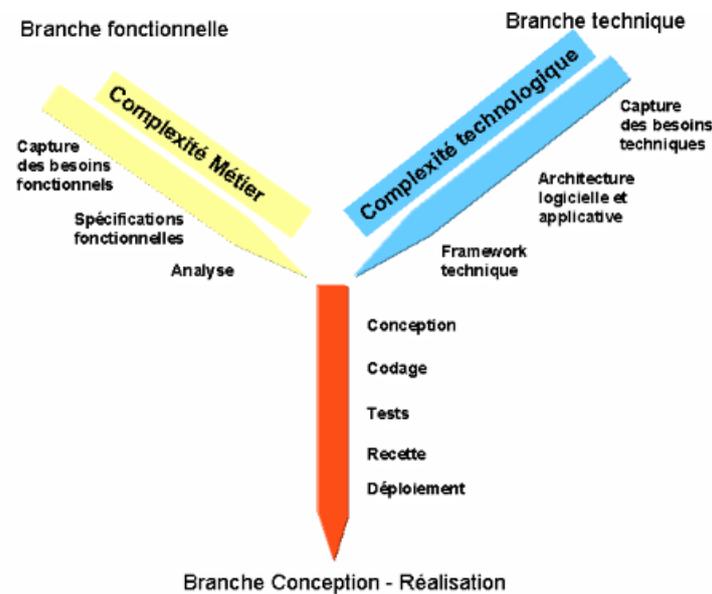


FIGURE 2.1 – Processus 2TUP

2.2.2 Processus de modélisation avec Unified Modeling Language (UML)

Le processus unifié utilise le langage UML (Unified Modeling Language). Ce langage de modélisation est une partie intégrante du processus unifié.

2.2.3 Obejctifs du langage UML

Créer un langage visuel commun dans le monde complexe du développement de logiciels, un langage qui serait également compris par les utilisateurs professionnels et tous ceux qui veulent comprendre un système. Acquérir des connaissances de base sur les diagrammes UML, et découvrir leurs origines, usages, concepts et types.

2.2.4 Historique de l'UML

Par rapport à la cinquantaine de méthodes d'analyse et de conception objet qui existaient au début des années 90, seulement trois d'entre elles se sont détachées nettement au bout de quelques années. En effet, la volonté de converger vers une méthode unifiée était déjà bien réelle et c'est pour cette raison que les méthodes OMT, BOOCH et OOSE se sont démarquées des autres. OMT (Object Modeling Technique) de James Rumbaugh et BOOCH de GradyBooch ont été les deux méthodes les plus diffusées en France et partout ailleurs durant les années 90. Par ailleurs, OOSE de Ivar Jacobson s'est aussi imposée dans le monde objet pour la partie formalisation des besoins. Pour aller plus loin dans le rapprochement, James Rumbaugh et GradyBooch se sont retrouvés au sein de la société Rational Software et ont été ensuite rejoints par Ivar Jacobson en se donnant comme objectif de fusionner leurs méthodes et créer UML (UnifiedMethodLanguage). Il est important de noter que contrairement à ce qui avait été envisagé au départ, le processus de développement a été sorti du champ couvert par le projet de norme. UML est donc une norme du langage de modélisation objet qui a été publiée, dans sa première version, en novembre 1997 par l'OMG (Object Management Group), instance de normalisation internationale du domaine de l'objet. En quelques années, UML s'est imposée comme standard à utiliser en tant que langage de modélisation objet. Aujourd'hui, au milieu de la deuxième décennie des années 2000, nous avons déjà une dizaine d'années de recul sur l'enseignement et la pratique d'UML en entreprise.[12]

2.2.5 Mise a jour d l'UML

L'UML fait l'objet d'améliorations continues. Sa version 2.0 étend ses spécifications pour couvrir davantage d'aspects du développement, y compris le développement agile. L'objectif était ici de restructurer et de perfectionner l'UML afin que son utilisabilité, sa mise en œuvre et son adaptation soient simplifiées. Voici quelques-unes des mises à jour apportées aux diagrammes UML :

- Une plus grande intégration entre les modèles structurels et comportementaux.
- La possibilité de définir une hiérarchie et de décomposer un système logiciel en éléments et sous-éléments.
- UML 2.0 augmente le nombre de diagrammes, qui passe de 9 à 13

2.2.6 Présentation d'UML

Afin d'assurer un bon niveau de cohérence et d'homogénéité sur l'ensemble des modèles, UML propose d'une part un certain nombre de règles d'écriture ou de représentations graphiques normalisées et d'autre part des mécanismes ou des concepts communs applicables à l'ensemble de ces diagrammes dont nous parlerons plus tard. Certains éléments, comme les stéréotypes, sont spécifiquement prévus pour assurer une réelle capacité d'adaptation et d'évolution de la notation notamment pour prendre en compte les particularités des différentes situations à modéliser. Les principaux éléments généraux d'UML que nous présentons sont : le stéréotype, la valeur marquée, la note, la contrainte, et la relation de dépendance.

En outre UML propose un méta-modèle de tous les concepts et notations associées utilisés dans les treize diagrammes de ce langage de modélisation.[12]

2.2.6.1 Méta-modèle

Le langage de modélisation UML respecte un certain nombre de règles sur les concepts manipulés (classes, attributs, opérations, paquetages, ...) ainsi que sur la syntaxe d'écriture et le formalisme de représentation graphique. L'ensemble de ces règles constitue en soi un langage de modélisation qui a fait l'objet d'un méta-modèle UML. L'intérêt de disposer d'un méta-modèle UML permet de bien maîtriser la structure d'UML et de faciliter son évolution.[12]

2.2.6.2 Stéréotype

Un stéréotype constitue un moyen de classer les éléments de la modélisation. Un certain nombre de stéréotypes sont déjà définis dans UML, mais d'autres valeurs de stéréotypes peuvent être ajoutées si cela est nécessaire soit à l'évolution générale d'UML, soit à la prise en compte de situations particulières propres aux entreprises. Les stéréotypes peuvent s'appliquer à n'importe quel concept d'UML. Nous nous utiliserons dans ce travail un certain nombre d'entre eux que nous présenterons au niveau des diagrammes lorsque leur utilisation nous paraîtra pertinente.[12]

2.2.6.3 Valeur marquée

UML permet d'indiquer des valeurs particulières au niveau des éléments de modélisation et en particulier pour les attributs de classe. Une valeur marquée se définit au niveau méta-attribut.[12]

2.2.7 Points forts d'UML

UML est un langage formel et normalisé :

- Gain de précision.
- Gage de stabilité.
- Encourage l'utilisation d'outils.
- UML est un support de communication performant.
- La facilite la compréhension de représentations abstraites complexes.[13]

2.2.8 Points faibles d'UML

- La mise en pratique d'UML nécessite un apprentissage et passe par une période d'adaptation.
- L'intégration d'UML dans un processus n'est pas triviale, et améliorer un processus est une tâche complexe et longue.[13]

2.2.9 Différents types de diagrammes

UML s'articule maintenant autour de 13 diagrammes différents, dont quatre nouveaux diagrammes introduits par UML 2.0. Chacun d'eux est dédié à la représentation d'un système

logiciel suivant un point de vue particulier. Par ailleurs, UML modélise le système logiciel suivant deux modes de représentation l'un concerne la structure du système pris "au repos", l'autre concerne sa dynamique de fonctionnement. Les deux représentations sont nécessaires et complémentaires pour schématiser la façon dont est composé le système et comment ses composants fonctionnent entre elles. On visualise ces modèles de système grâce à deux types distincts de diagrammes : structurel et comportemental. [36]

la figure suivant présente les différents types de diagrammes de l'UML :

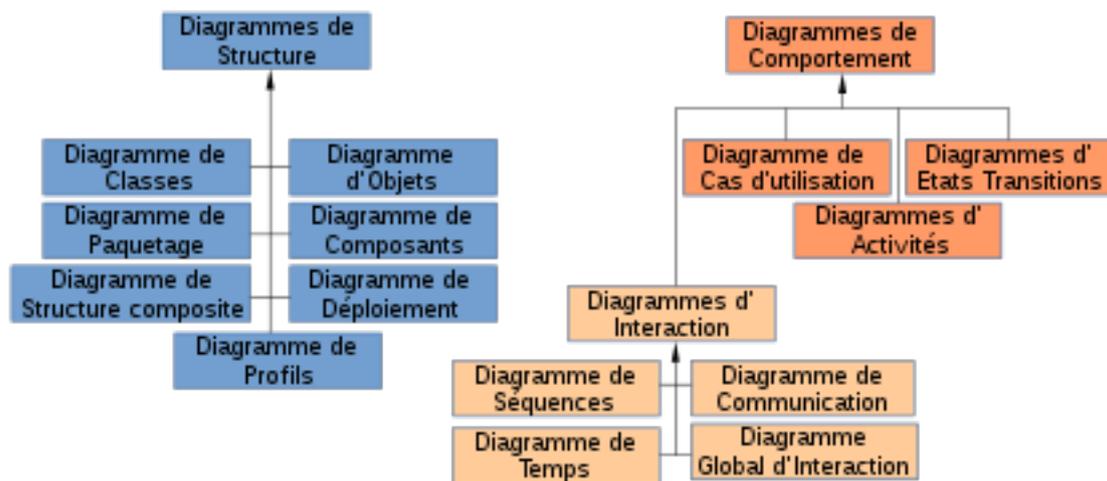


FIGURE 2.2 – Diagrammes UML

- **Fonctionnel** : ce sont des diagrammes de cas d'utilisation, qui décrivent la fonctionnalité du système du point de vue de l'utilisateur.
- **Objet** : ce sont des diagrammes de classes qui décrivent la structure d'un système en termes d'objets, attributs, associations et opérations.
- **Dynamique** : ce sont des diagrammes d'interaction, diagrammes états-transitions et diagrammes d'activités utilisés pour décrire le comportement interne du système.

Dans notre projet nous allons utiliser juste les diagrammes suivant :

2.2.9.1 Diagramme de contexte

Le diagramme de contexte permet de définir les limites de l'étude. Il place le diagramme dans son contexte en listant les acteurs ou éléments qui agissent ou interagissent avec lui. Le degré de raffinement de ce diagramme dépend de ce que l'on veut montrer.

2.2.9.2 Diagramme de cas d'utilisation

En langage UML, les diagrammes de cas d'utilisation modélisent le comportement d'un système et permettent de capturer les exigences du système. Ils décrivent les fonctions générales et la portée d'un système. Ces diagrammes identifient également les interactions entre le système et ses acteurs. Les cas d'utilisation et les acteurs dans les diagrammes de cas d'utilisation décrivent ce que le système fait et comment les acteurs l'utilisent, mais ne montrent pas comment le système fonctionne en interne. [14]

1. L'utilité des diagrammes de cas d'utilisation

- Avant de commencer un projet, vous pouvez créer des diagrammes de cas d'utilisation pour modéliser une entreprise, afin que tous les participants au projet visualisent bien les travailleurs, clients et activités de l'entreprise.
- Lors du recueil des exigences, vous pouvez créer des diagrammes de cas d'utilisation pour capturer les exigences du système et pour présenter aux autres ce que le système doit faire.
- Lors des phases d'analyse et de conception, vous pouvez utiliser les cas d'utilisation et les acteurs de vos diagrammes de cas d'utilisation pour identifier les classes nécessaires.

2. **Cas d'utilisation** : définissent l'activité sous la forme d'actes et de réponses, ou simplement ce qui se passe. Le diagramme de cas d'utilisation décrit toutes les fonctions du système. Il permet également à chaque acteur de remplir ses exigences professionnelles

3. **Acteur** : l'acteur représente un ensemble cohérent de rôles que les utilisateurs de cas d'utilisation jouent dans l'interaction avec ces cas. En règle générale, un acteur joue un rôle qu'un homme, une machine ou éventuellement un autre système interagissant avec le système modélisé. Il existe quatre principaux types d'acteurs :

- acteurs principaux.
- acteurs secondaires.
- matériel externe.

- Les interactions du système avec d'autres systèmes.
4. **Relation entre les cas d'utilisation** : L'approche UML décrit trois types différents de relations structurées entre les cas d'utilisation, qui sont répertoriés ci-dessous :
- **La relation d'inclusion** : Le cas d'utilisation principal ne peut être fait qu'en spécifiant un autre cas d'utilisation de manière obligatoire. Cette dépendance est symbolisée par le stéréotype "include"
 - **La relation d'extension** : un cas d'utilisation de base comprend spécifiquement un autre cas d'utilisation, de manière optionnelle, représentée par le mot-clé "extend"
 - **La relation de généralisation ou de spécialisation** : un cas d'utilisation peut hériter d'un autre cas d'utilisation [15]

2.2.9.3 Diagramme de séquence

Les diagrammes de séquence sont utilisés pour montrer les interactions temporelles entre les acteurs et le système selon un ordre chronologique dans la formulation Source spécifiée non valide. Les diagrammes de séquence contiennent :

1. **L'objet** : Un objet est une instance de la classe ou de l'interface responsable de la réalisation des opérations définies. Il est représenté graphiquement dans le diagramme de séquence par un carré comportant son identifiant
2. **La relation** : est un lien sémantique entre objets, ou une association en général. Il y aura une connexion entre les instances de deux classes si une classe est connectée à une autre par une interaction, et chaque fois qu'une relation existe entre deux objets, le premier objet peut envoyer un message au second.
3. **Le message** : est la spécification d'une correspondance entre des objets qui transmet des informations et est affiché afin d'initier une action. La réception d'une instance de message peut être appelée instance d'une occurrence.[15]

Dans un diagramme de séquence, deux types de messages peuvent être distingués :

- **Message synchrone** : Dans ce cas l'émetteur reste en attente de la réponse à son message avant de poursuivre ses actions. La flèche avec extrémité pleine symbolise ce type de message. Le message retour peut ne pas être représenté car il est inclus dans la fin d'exécution de l'opération de l'objet destinataire du message.
- **Message asynchrone** : Dans ce cas, l'émetteur n'attend pas la réponse à son message, il poursuit l'exécution de ses opérations. C'est une flèche avec une extrémité non pleine qui symbolise ce type de message.

4. **Fragment d'interaction** : Un fragment d'interaction, dit " combiné ", correspond à un ensemble d'interactions auquel on applique un opérateur. Un fragment combiné se représente globalement comme un diagramme de séquence avec indication dans le coin à gauche du nom de l'opérateur. 13 opérateurs ont été définis dans UML : alt, opt, loop, par, strict/weak, break, ignore/ consider, critical, négative, assertion et ref.

2.2.9.4 Diagramme de classe

C'est un diagramme qui montre une collection d'éléments statiques (classes), leur contenu et les relations entre eux, En analyse, le diagramme de classe représente la structure des informations manipulées par les utilisateurs .En conception, il représente la structure d'un code orienté objet.Source spécifiée non valide. Les éléments suivants se trouvent généralement dans les diagrammes de classes :

Une classe : est une classification d'un groupe d'objets qui ont les mêmes attributs, opérations, relations et sémantiques, Un rectangle est utilisé pour représenter une classe.

Attribut : Une propriété nommée d'une classe qui définit une collection de valeurs que les instances de cette propriété peuvent prendre est appelée un attribut. Une classe peut avoir un ou plusieurs attributs, mais elle peut également n'en avoir aucun.

Opération : une opération est une représentation de ce qu'un objet peut faire et de ce que tous les objets de la classe peuvent faire. Puisqu'une classe peut avoir plusieurs opérations, elle peut ne pas en avoir.

Relation entre agrégation et composition

- Une relation sémantique entre deux groupes est représentée par une association.
- Une relation de confinement est exprimée par une agrégation, qui est une relation non symétrique.
- Une composition est une forme d'agrégation plus efficace.[15]

2.3 Analyse des besoins

2.3.1 Acteurs et modélisation de contexte

2.3.1.1 Identification des acteurs

La première étape de la modélisation consiste à définir le périmètre du système le contour de l'organisation.La seconde étape sert à modéliser et identifier les entités qui interagissent avec

le système qui sont les acteurs .

- **Administrateur** : c'est l'informaticien de la pharmacie qui gère les comptes des utilisateurs et les services.
- **Pharmacien chef** : c'est le chef de la pharmacie il fait plusieurs tâches.
- **Pharmacien réactif** : c'est le pharmacien qui s'occupe de la gestion des produits réactifs.
- **Agent** : c'est l'agent d'autres service qui fait la livraison des produits.

2.3.1.2 Identification des messages

Un message représente la spécification d'une communication unidirectionnelle entre objets et qui transporte de l'information avec l'intention de déclencher une activité chez le récepteur un message est normalement associé a deux occurrences d'événement : un événement d'envoi et un évènement de réception .

Acteurs	Messages émis par l'acteur	Message émis par le système
Administrateur	<p>M1 :S'authentifier</p> <p>M2 :Demande l'accès a l'un des gestions</p> <p>M3 :Ajouter/Supprimer/Modifier /Consulter au niveau des gestions</p>	<p>MS :Interface d'authentification</p> <p>MS2 :interface de la gestion demandée</p> <p>MS3 :Valider le formulaire</p>
Pharmacien chef	<p>M1 :S'authentifier</p> <p>M2 :Demande l'accès a l'un des gestions</p> <p>M3 :Ajouter/Supprimer/Modifier /Consulter au niveau des gestions</p>	<p>MS :Interface d'authentification</p> <p>MS2 :interface de la gestion demandée</p> <p>MS3 :Valider les résultats</p>
Pharmacien réactif	<p>M1 :S'authentifier</p> <p>M2 :Demande l'accès a l'un des gestions</p> <p>M3 :Ajouter/Supprimer/Modifier /Consulter au niveau des gestions</p>	<p>MS :Interface d'authentification</p> <p>MS2 :interface de la gestion demandée</p> <p>MS3 :Valider les résultats</p>
Agent	<p>M1 :S'authentifier</p> <p>M2 :Demande l'accès a l'un des gestions</p> <p>M3 :Modifier/Valider /Consulter au niveau des gestions</p>	<p>MS :Interface d'authentification</p> <p>MS2 :interface de la gestion demandée</p> <p>MS3 :Valider le formulaire</p>

TABLE 2.1 – Identification des messages

2.3.1.3 Diagramme de contexte

Le diagramme de contexte c'est un diagramme qui intervient au début et il représente l'ensemble des différents acteurs qui interagissent avec notre système.

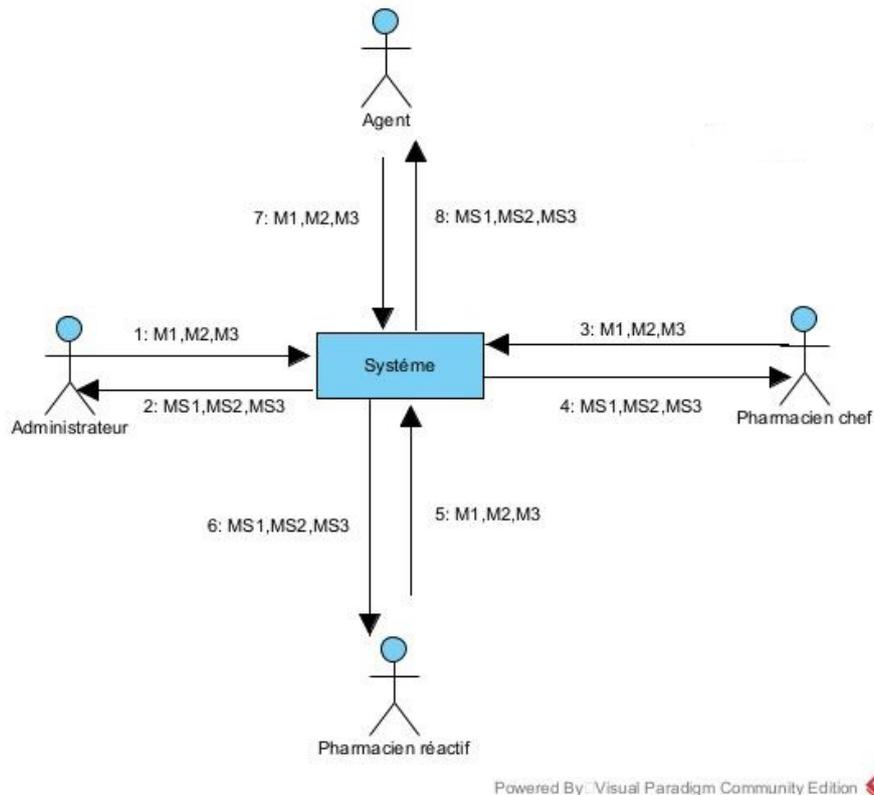


FIGURE 2.3 – Diagramme de contexte de notre système.

2.4 Analyse et étude conceptuelle

Cette partie nous permettra de faire l'analyse de notre système. Cette analyse consiste dans un premier temps à rechercher et à décrire les cas d'utilisation de notre système à travers les diagrammes de cas d'utilisation, les diagrammes de séquences, cette analyse nous conduira aussi à la production de diagramme de classes et à la conception de notre application.

2.4.1 Diagramme de cas d'utilisation

Dans ce qui suit, nous allons présenter les différents diagrammes cas d'utilisation mis en oeuvre dans notre application.

2.4.1.1 Identification des acteurs

Ce tableau est une représentation des différents acteurs du système et leurs cas d'utilisation.

Acteurs	Cas d'utilisation
Administrateur	S'authentifier Gérer les utilisateurs Gérer les Services
Pharmacien chef	S'authentifier Gérer les produits Gérer les bon Entrée et Sortie Gérer les catégories
Pharmacien réactif	S'authentifier Gérer les produits Gérer les bon Entrée et Sortie Gérer les catégories
Agent	S'authentifier Gérer les bons livraison

TABLE 2.2 – Identification des acteurs

2.4.1.2 Diagramme Cas d'utilisation pour l'administrateur

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

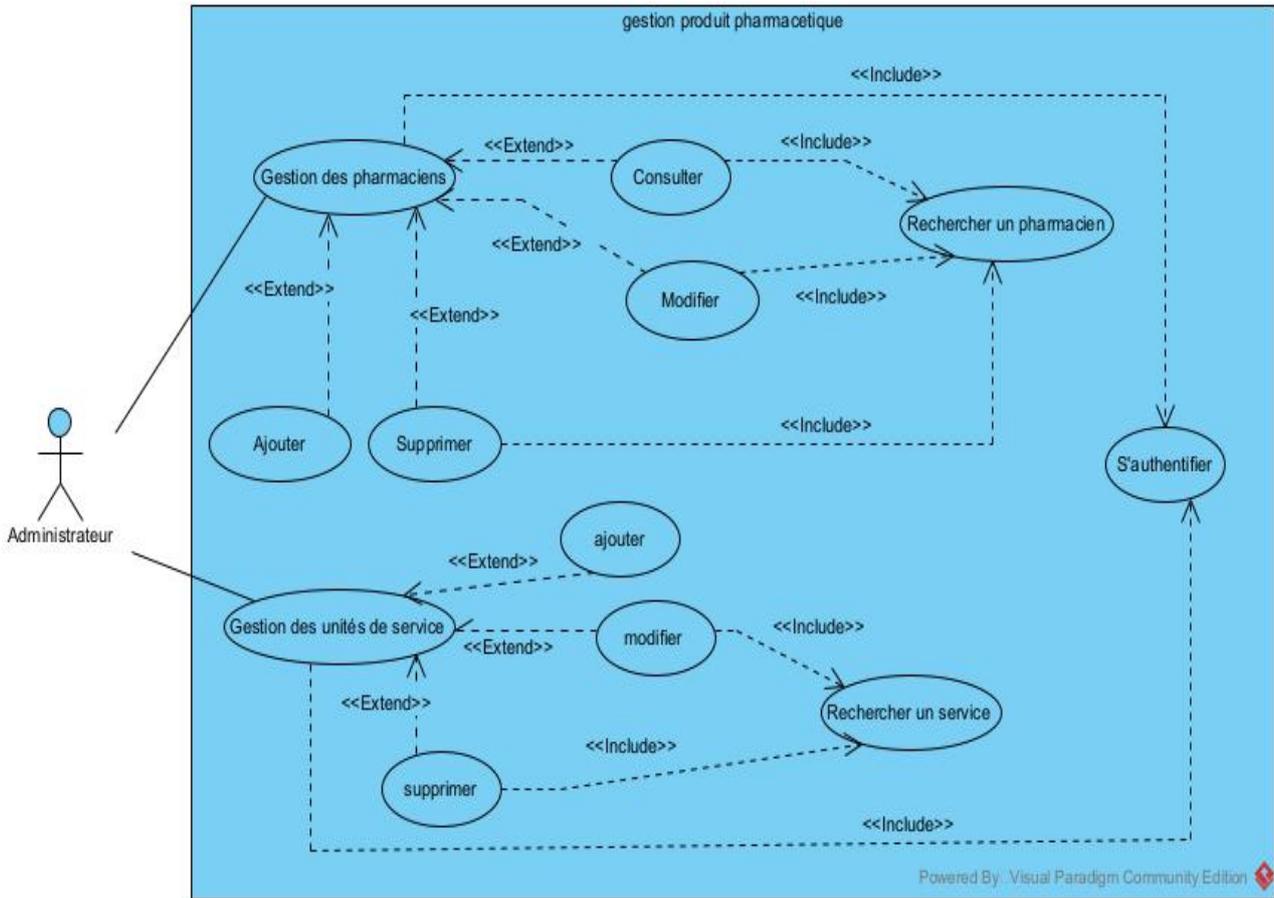


FIGURE 2.4 – Diagramme de cas d'utilisation de l'administrateur

2.4.1.3 Diagramme Cas d'utilisation pour Pharmacien chef

Le diagramme ci-dessous représente les cas d'utilisations identifiés pour Pharmacien chef.

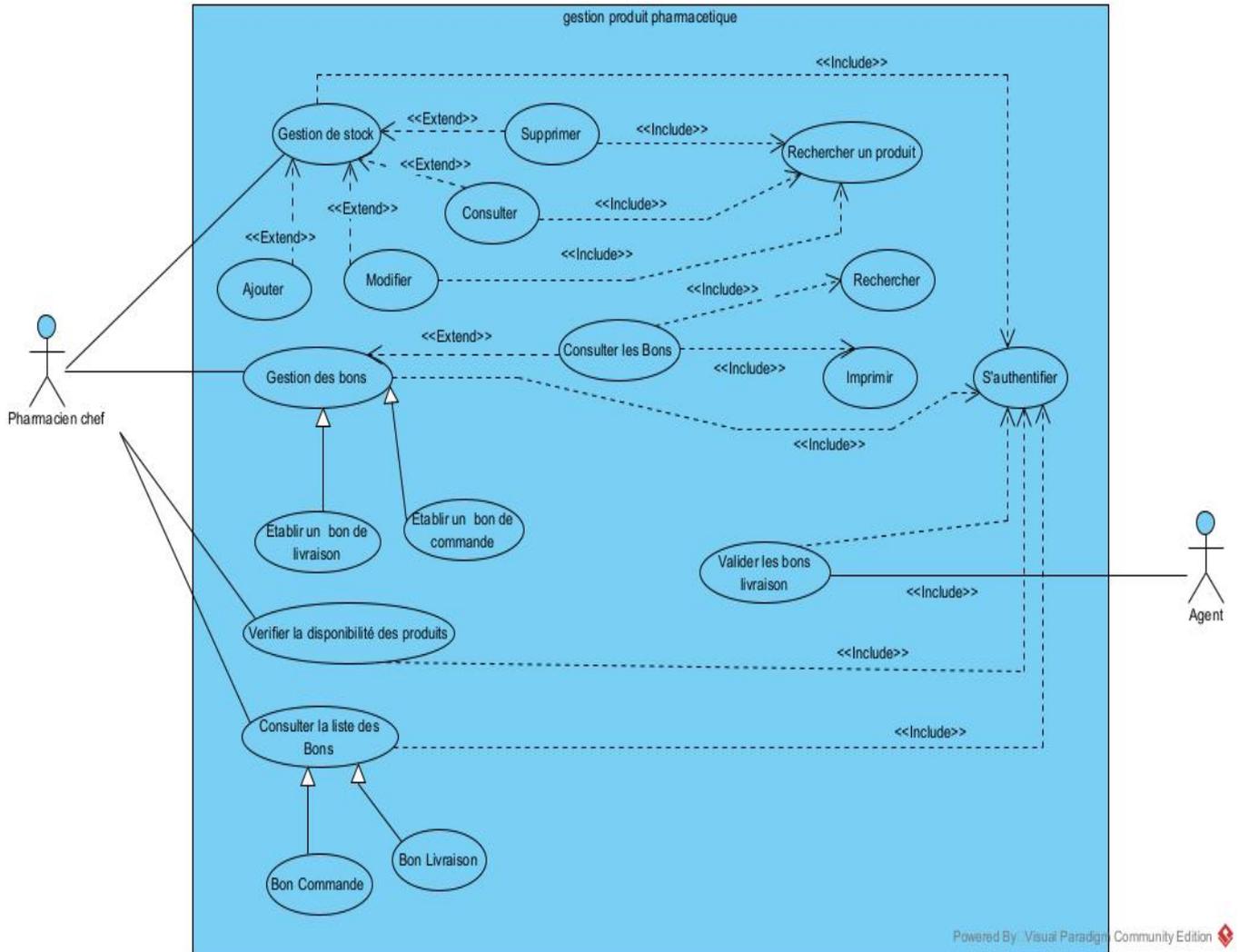


FIGURE 2.5 – Diagramme de cas d'utilisation de Pharmacien chef

2.4.1.4 Diagramme Cas d'utilisation pour Pharmacien réactif

Le diagramme ci-dessous représente les cas d'utilisations identifiés pour Pharmacien réactif.

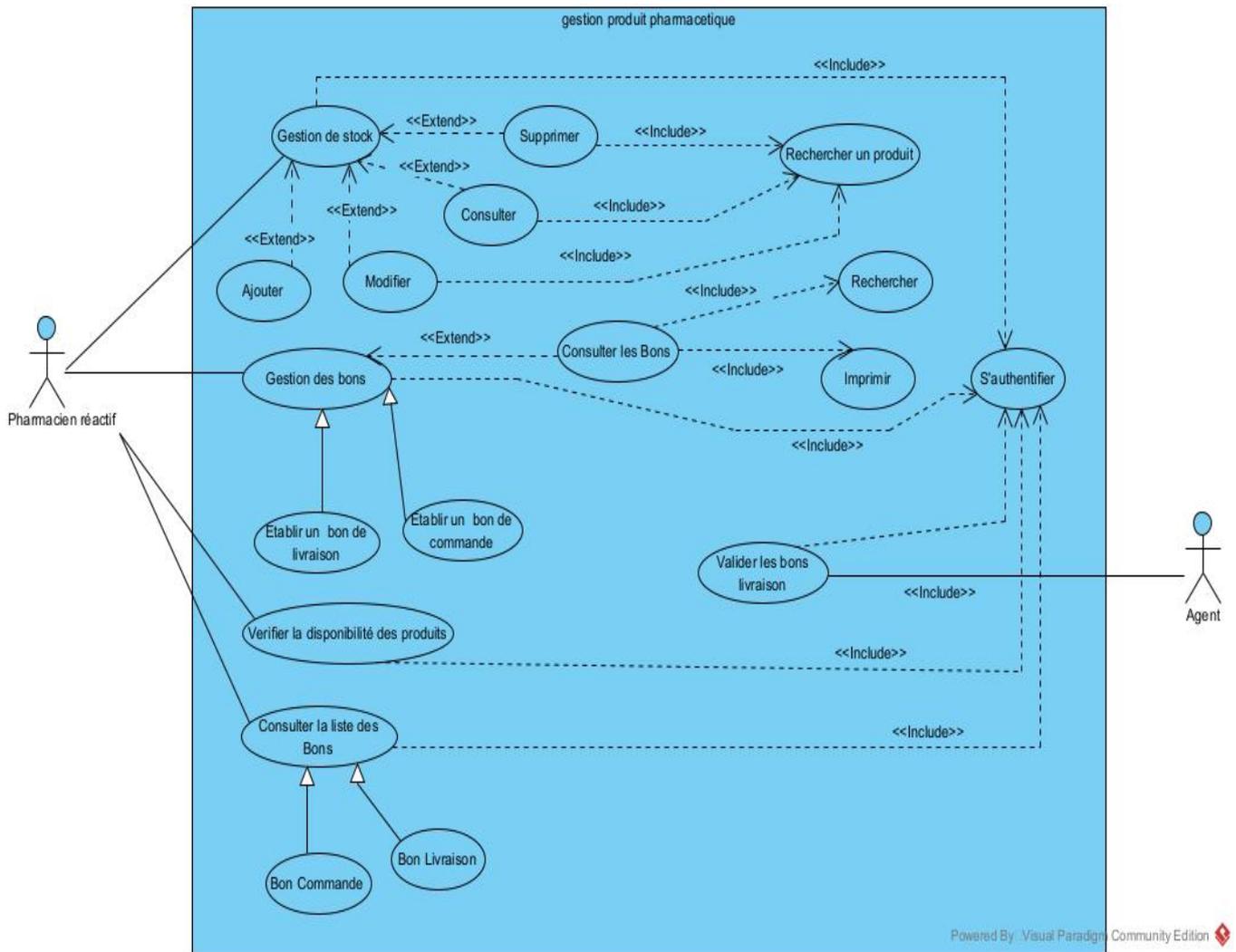


FIGURE 2.6 – Diagramme de cas d'utilisation de pharmacien réactif

2.4.1.5 Diagramme de cas d'utilisation globale

La figure suivant présente le digramme global des cas d'utilisation de notre application :

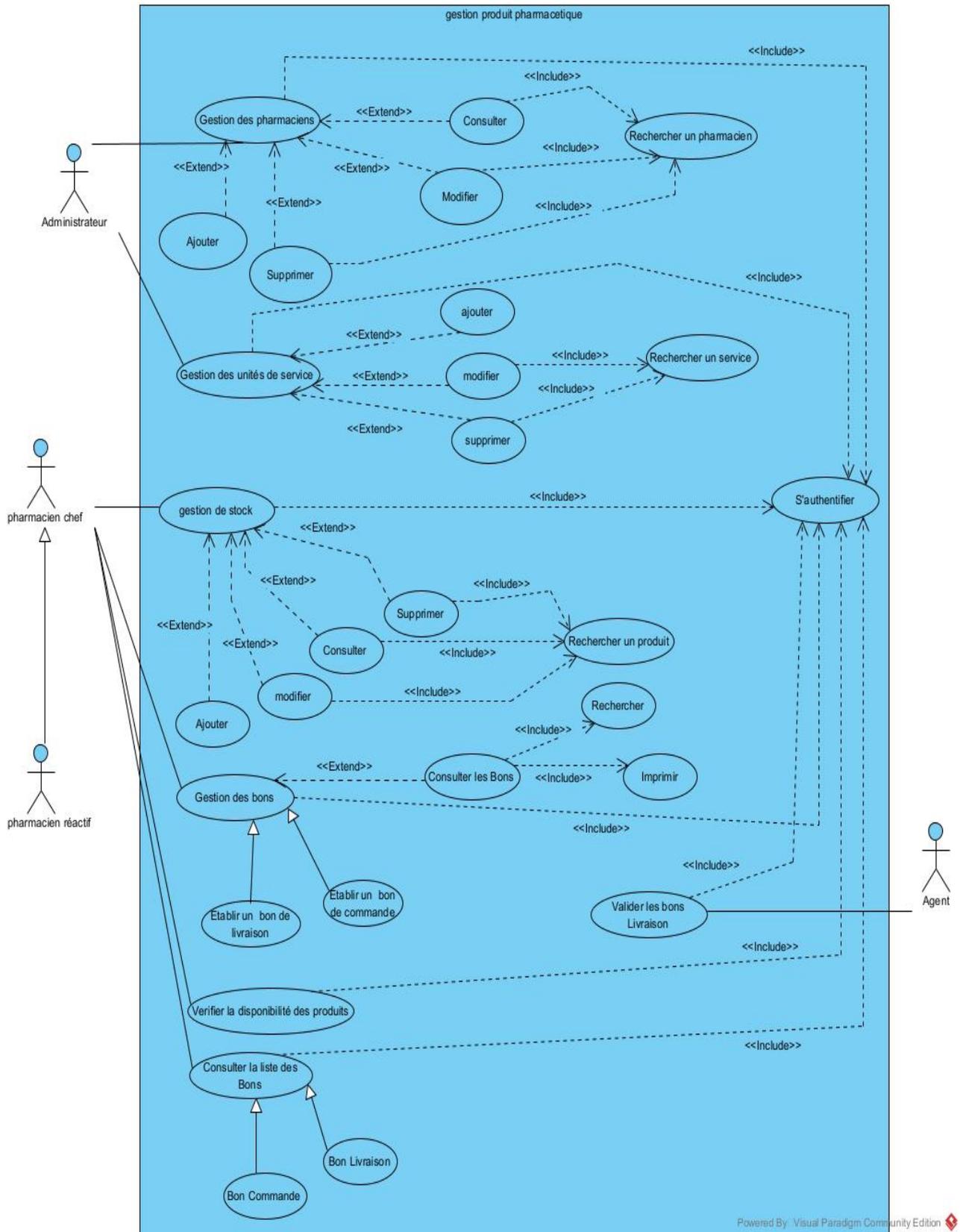


FIGURE 2.7 – Diagramme de cas d'utilisation globale

2.4.1.6 Description de Cas d'utilisation

- La description du cas d'utilisation « S'authentifier » :

Description
Acteur Administrateur, chef pharmacien, pharmacien réactif, Agent
Objectif Vérifier les possibilités d'accès au système
Scénario nominal L'utilisateur lance l'application Le système affiche l'interface d'authentification L'utilisateur saisit le code d'utilisateur et le mot de passe Si l'utilisateur est identifié, le système affiche l'interface de "Accueil"
Scénario alternatif Si le nom d'utilisateur ou le mot de passe n'est pas valide Le système affiche un message d'erreur

TABLE 2.3 – Description du cas d'utilisation "S'authentifier"

• La description du cas d'utilisation « Gestion des utilisateurs »

description
Acteur Administrateur
Objectif pouvoir ajouter, modifier, supprimer
Scénario nominal 1er cas : ajouter un utilisateur L'administrateur choisi d'ajouter un utilisateur. Le système affiche le formulaire d'ajouter un utilisateur. L'administrateur du système saisit les informations nécessaires en indiquant le rôle de l'utilisateur (administrateur, pharmacien chef, pharmacien réactif, Agent) et valide la création. Le système crée un compte pour le nouvel utilisateur. 2ème cas : Modifier les informations L'administrateur choisi la fenêtre utilisateur Le système affiche de la liste des utilisateurs L'administrateur clique sur le champ modifier pour modifier les informations. 3ème cas : Supprimer le compte d'un utilisateur l'administrateur choisi la fenêtre utilisateur le système affiche la liste des utilisateurs l'administrateur clique sur le champ supprimer un utilisateur Le système affiche un message de confirmation de la suppression l'administrateur confirme la suppression de l'utilisateur.
Scénario alternatif cas1 : le compte existe déjà ou bien champs non conforme aux type ou bien formulaire vide : un message d'erreur s'affiche. cas2 : Modification avec champs vides : un message d'erreur s'affiche. cas3 : le système n'arrive pas à identifier l'administrateur.

TABLE 2.4 – Description du cas d'utilisation «gestion des utilisateurs»

- La description du cas d'utilisation «Gestion des produits» :

description
Acteur chef pharmacien, pharmacien réactif
Objectif Ajouter ,rechercher,modifier et supprimer un produit
Scénario nominal 1er cas :ajouter un produit permet d'ajouter un produit à la liste des produits selon l'enchaînement suivant : L'utilisateur choisit Ajouter un produit. Le système affiche le formulaire à remplir. l'utilisateur rempli et valide le formulaire Le système actualise la liste des produits. 2ème cas :Rechercher un produit l'utilisateur saisir le nom ou le code d'un produit la système afficher tous les information. 3ème cas :Modifier un produit L'utilisateur choisi le produit a modifier puis il clique sur le bouton modifier. e système affiche le formulaire de modification. l'utilisateur modifier les champs souhaitées 4ème cas :Supprimer un produit L'utilisateur choisi le produit a supprimer puis il clique sur le bouton supprimer. Le système demande la confirmation. L'utilisateur confirme la suppression . Le produit sera supprimé.
Scénario alternatif cas1 produit existe déjà ou champs non conforme aux types :message d'erreur sera affiché cas3 modifier avec champs vide ou bien champs non conforme aux types : un message d'erreur sera affiché

TABLE 2.5 – Description du cas d'utilisation «gestion des produits»

- La description du cas d'utilisation «Ajouter bon Commande» :

Description
Acteur pharmacien chef ,pharmacien réactif
Objectif Commander les produits
Scénario nominal Le pharmacien demande l'interface des bons Le système afficher l'interface des bons Le pharmaciens demande l'interface d'ajouter un bon Le système affiche l'interface d'ajout Le pharmacien remplir le formulaire et valider Le système affiche le bon dans la liste des bons
Scénario alternatif Un champ vide : message d'erreur

TABLE 2.6 – Description du cas d'utilisation "Ajouter bon Commande"

- La description du cas d'utilisation «Ajouter bon Livraison» :

Description
Acteur Agent
Objectif livrée les produits
Scénario nominal L'agent demande l'interface des bons Le système afficher l'interface des bons L'agent sélectionner un bon puis clique sur Valider Le système affiche les information de cette bon L'agernt remplir le champ quantité livrée de chaque produit et valider le système affiche le bon valider l'agent demande l'imprimer l'impression de bon

TABLE 2.7 – Description du cas d'utilisation "Ajouter bon Livraison"

2.4.2 Diagramme de séquence

Dans ce qui suit, nous allons présenter les différents diagrammes de séquence des cas d'utilisation mis en oeuvre dans notre application.

2.4.3 Diagramme de séquence «Authentification»

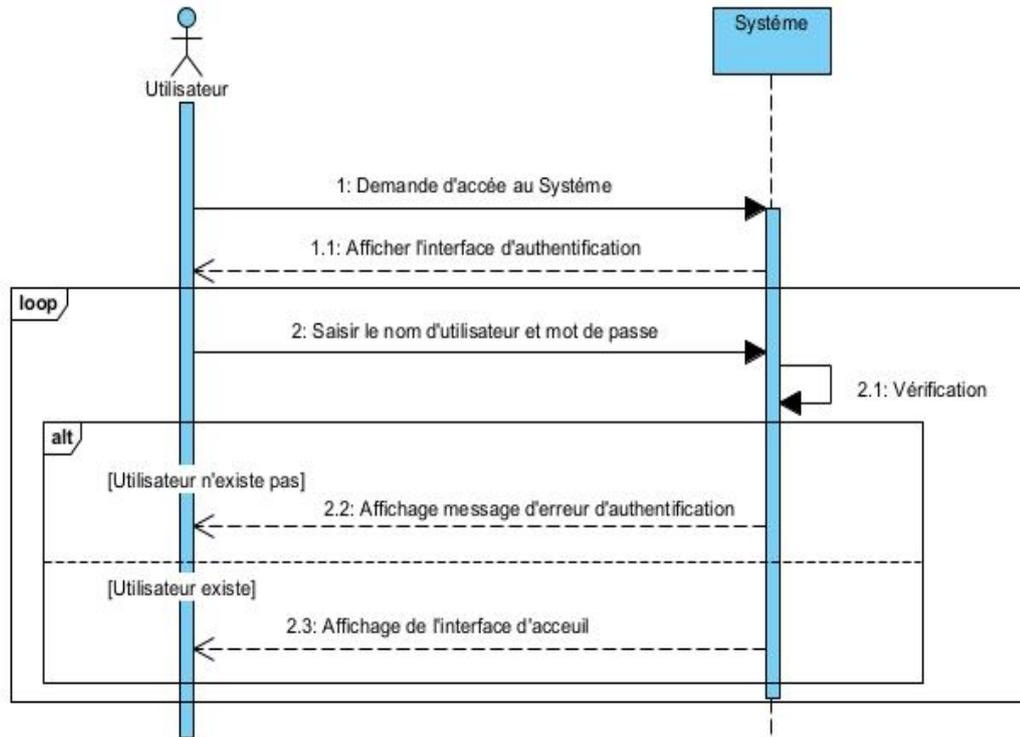


FIGURE 2.8 – Diagramme de séquence de cas d'utilisation «Authentification»

2.4.4 Diagramme de séquence «Gestion des comptes»

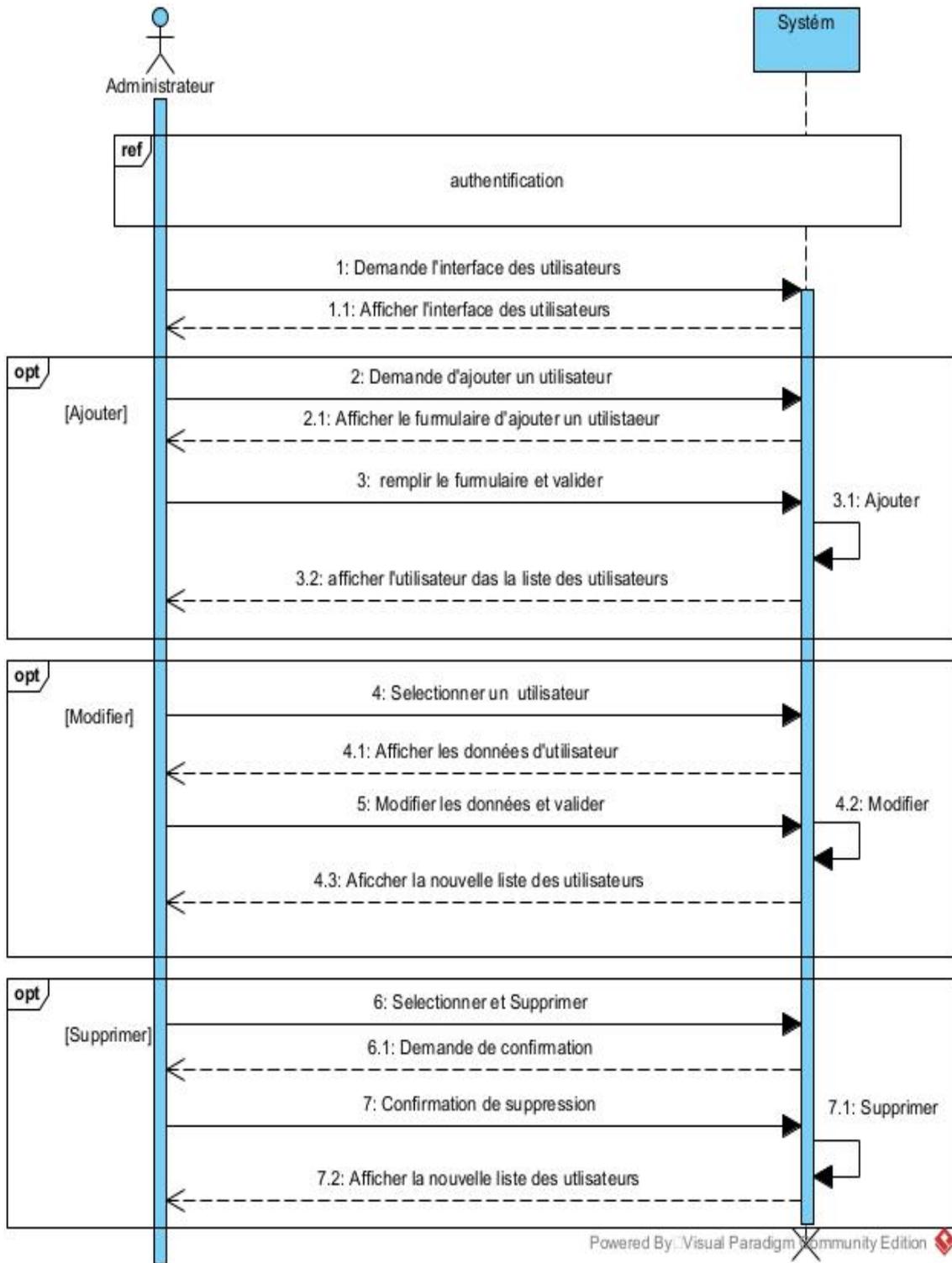


FIGURE 2.9 – Diagramme de séquence «Gestion des comptes»

2.4.5 Diagramme de séquence «Gestion des produits»

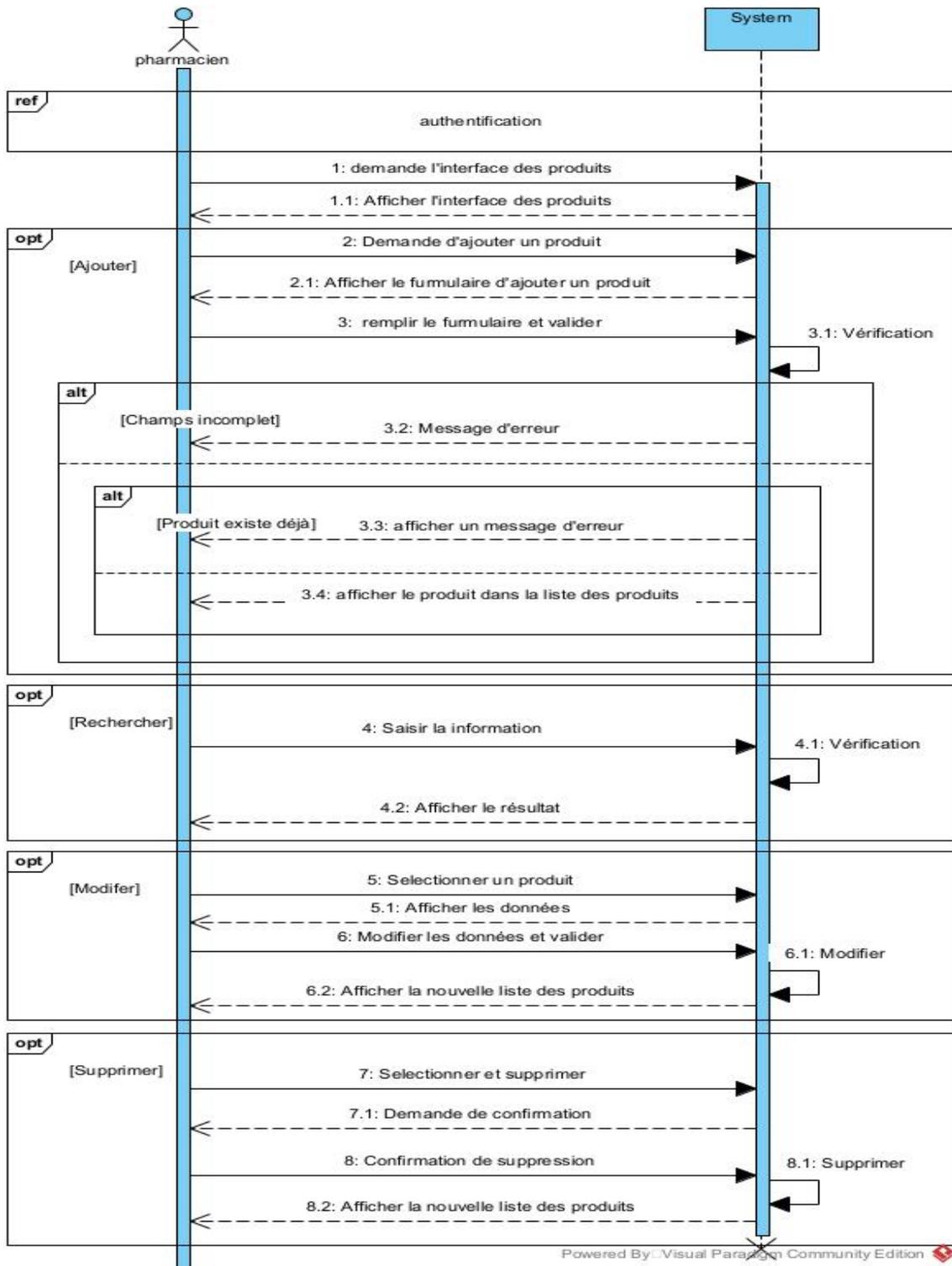


FIGURE 2.10 – Diagramme de séquence de cas d’utilisation «Gestion des produits»

2.4.6 Diagramme de séquence «Ajouter un bon Commande»

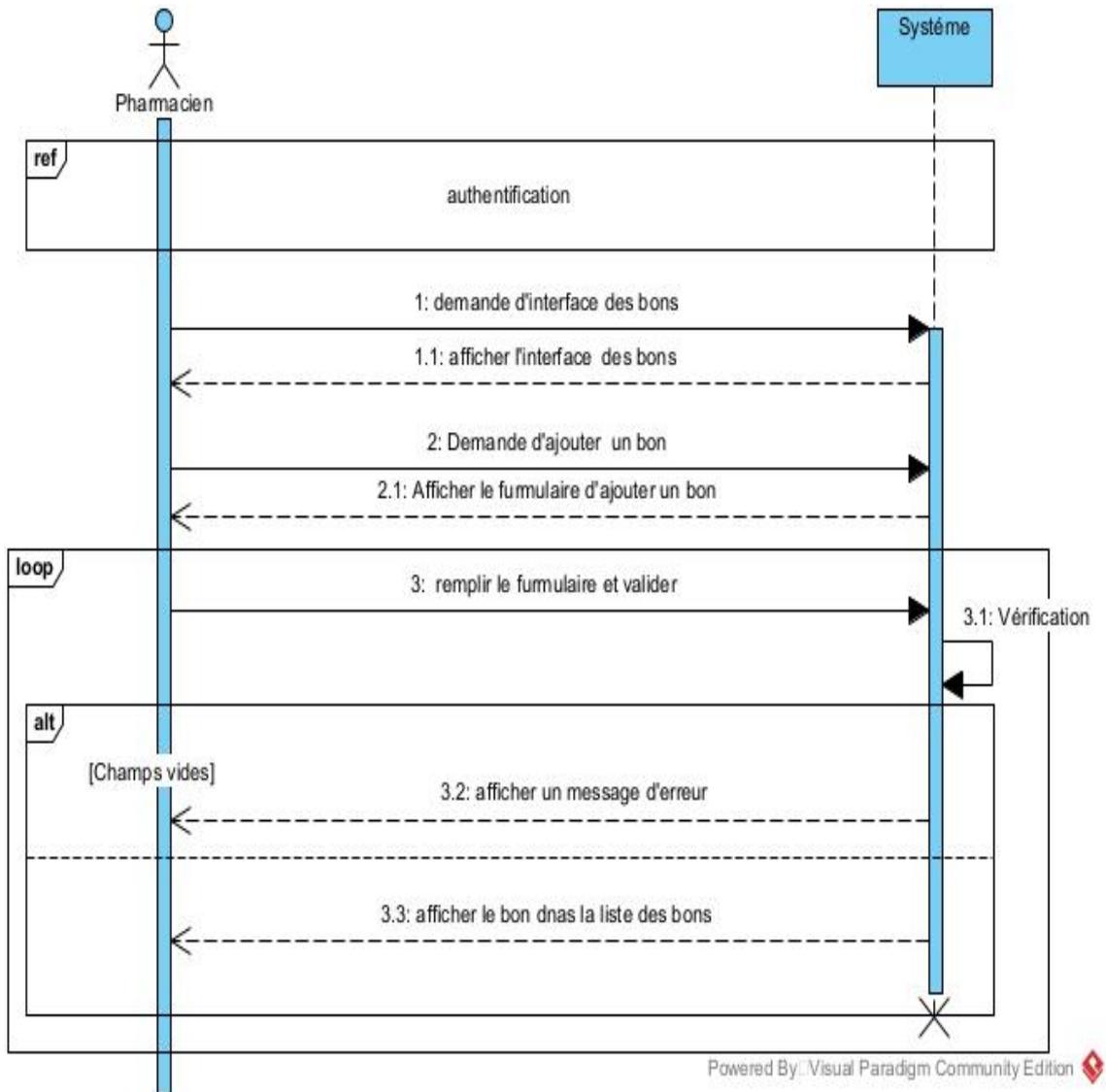
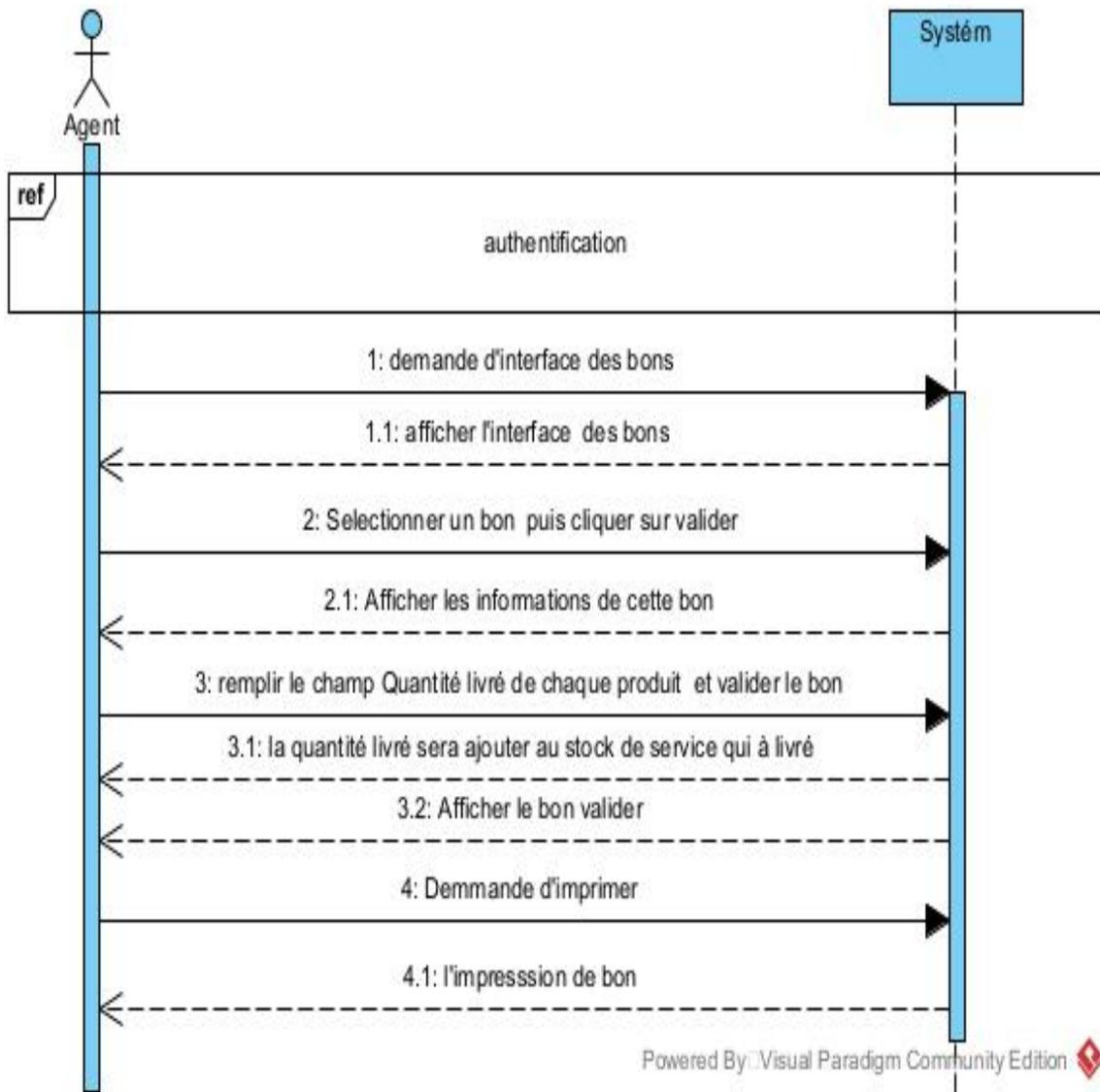


FIGURE 2.11 – Diagramme de séquence de cas d’utilisation «Ajouter un bon Commande»

2.4.7 Diagramme de séquence «Ajouter un bon Livraison»



Powered By Visual Paradigm Community Edition

FIGURE 2.12 – Diagramme de séquence de cas d’utilisation «Ajouter un bon Livraison»

2.4.8 Diagramme de classe

Le diagramme de classe est généralement considéré comme le plus important dans la conception d'une application.

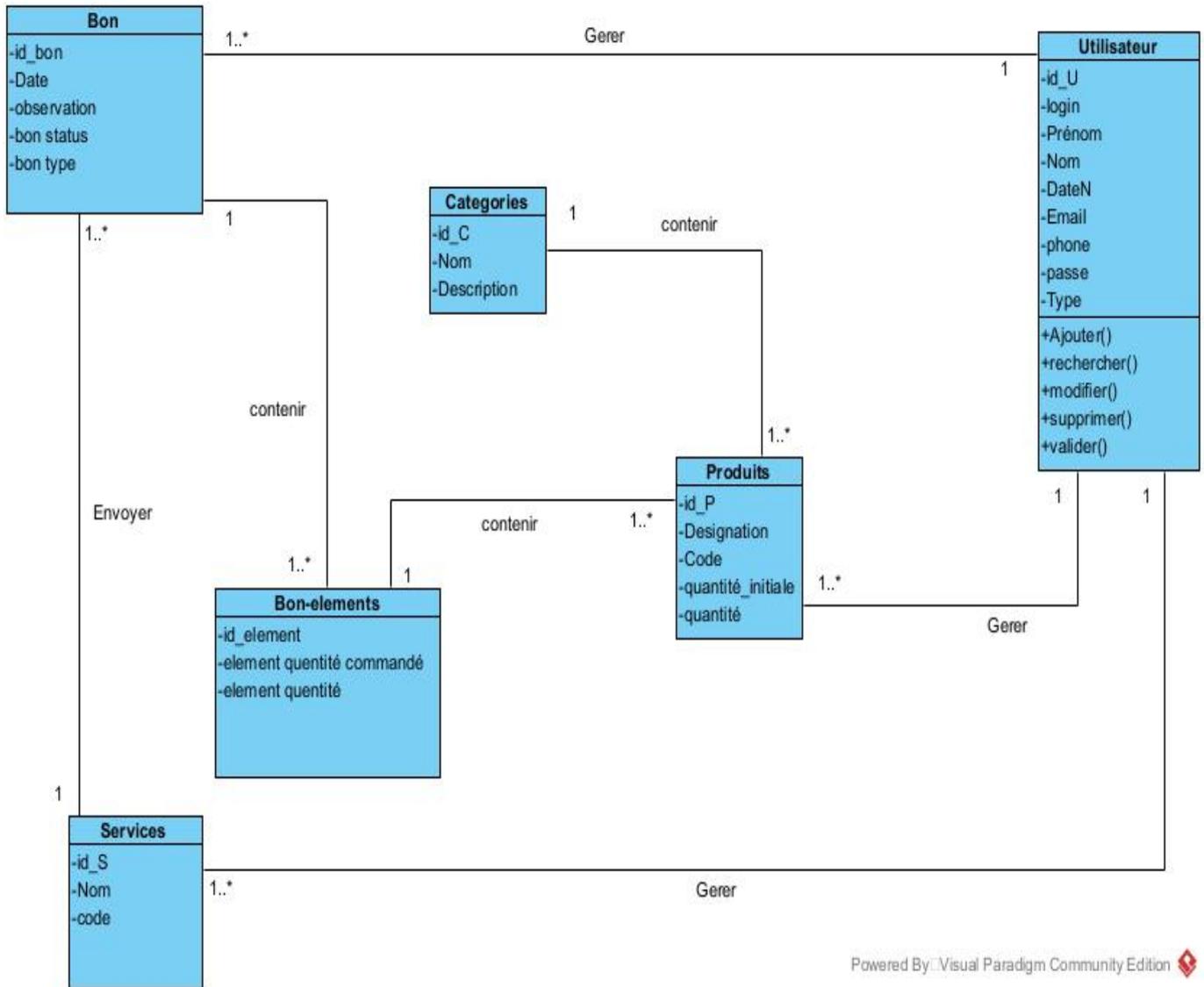


FIGURE 2.13 – Diagramme de classe

2.4.9 Dictionnaire de données

Table Utilisateur

Class	Attributs	Définition de l'attribut	Type
Utilisateur	id-U	L'identifiant de l'utilisateurs	Integer
	Nom	Le nom de l'utilisateur	String
	Prénom	Le prénom de l'utilisateur	String
	Date	La date naissance de l'utilisateur	String
	Email	Le Email de l'utilisateur	String
	phone	Numéro de téléphone de l'utilisateur	String
	type	Le grade d'utilisateur	String

TABLE 2.8 – Table Utilisateurs

Table Service

Class	Attributs	Définition de l'attribut	Type
Service	id-S	L'identifiant de service	Integer
	Nom	Le nom du service	String
	Code	Le code du service	Integer

TABLE 2.9 – Table Service

Table Bon

Class	Attributs	Définition de l'attribut	Type
Bon	id-bon	l'identifiant de bon	Integer
	Date	La date du bon	Date
	Observation	L'observation(la remarque)	String
	Bon status	Statut du bon(entrée ou bien sortie)	Enums
	Bon type	Le type du bon	Enums

TABLE 2.10 – Table Bon

Table Bon-élément

Class	Attributs	Définition de l'attribut	Type
Bon-élément	id-élément	L'identifiant	Integer
	élément quantité commandé	La quantité commandée	integer
	élément quantité	La quantité	integer

TABLE 2.11 – Table Bon-élément

Table Catégorie

Class	Attributs	Définition de l'attribut	Type
Catégorie	id-C	L'identifiant du Catégorie	Integer
	Nom	Le nom du Catégorie	String
	Description	La description du catégorie	String

TABLE 2.12 – Table Catégorie

Table Produit

Class	Attributs	Définition de l'attribut	Type
Produit	id-P	l'identifiant du produit	Integer
	Désignation	La désignation du produit	String
	Code	Le code du produit	Integer
	Quantité-initiale	La quantité initial du produit	Integer
	Quantité	La quantité du produit	Integer

TABLE 2.13 – Table Produit

2.4.10 Passage au modèle relationnel

Après avoir élaboré le diagramme de classe, ce dernier peut être traduit en modèle relationnel ,c'est a dire en un ensemble de relations en troisième forme normale. Cet ensemble de relations peut être implémenté directement,sous forme d'une base de donnée.

2.4.10.1 Règles de passage du diagramme de classe au modèle relationnel

Les règles utilisées pour le passage du modèle des classes au modèle relationnel sont les suivantes :

- **Règle 1 : Transformation de classe** Chaque classe du diagramme UML devient une relation. Il faut choisir un attribut de la classe pouvant jouer le rôle de l'identifiant. Dans le cas où aucun attribut ne convient en tant qu'identifiant, il faut en ajouter un de telle sorte que la relation dispose d'une clé primaire.
- **Règle 2 : Association un à plusieurs** Il faut ajouter un attribut de type clé étrangère dans la relation fils de l'association. L'attribut porte le nom de la clé primaire de la relation père de l'association.
- **Règle 3 : Association plusieurs à plusieurs** L'association (classe-association) devient une relation dont la clé primaire est composée par la concaténation des identifiants des classes connectés à l'association (classe-association). Les attributs de l'association (classe-association) doivent être ajoutés à la nouvelle relation. Ces attributs ne sont ni clé primaire, ni clé étrangère.

L'application des règles de passage énumérées précédemment, nous permet d'avoir le modèle relationnel de la base de données de l'application à mettre en œuvre qu'est le suivant :

Utilisateur(id-U, Nom, Prénom, Date, Émail, phone, type, #id-bon, #id-P, #id-S)

Service(id-S, nom, Code, #id-bon)

Produit(id-P, Désignation, Code, quantité initiale, quantité)

Bon(id-bon, Date, Observation, bon status, bon type, #id-élément)

Bon-élément(id-élément, élément quantité commandé, élément quantité, #id-P)

Catégorie(id-C, nom, description, #id-P)

2.4.11 Conclusion

Dans ce chapitre, nous avons définie le langage de modélisation UML et le processus de développement qui est la méthodologie 2TUP , En s'appuyant sur l'analyse des besoins de notre application. Nous avons pu là modéliser sous forme de diagrammes de contexte puis de cas d'utilisation et de séquence, puis concevoir notre application sous forme de diagrammes de classe et schémas relationnelle, Dans le chapitre suivant, nous allons présenter l'implémentions de notre application et les outils et les environnements utilisés. Des copies d'écran sont ajoutées pour montrer les fonctionnalités de notre application.

Implémentation et réalisation d'application

3.1 Introduction

Dans ce chapitre, nous présentons la partie réalisation qui est la phase finale de notre projet. Nous décrirons l'environnement ainsi que les outils qui nous ont servis pour la réalisation de l'application et nous terminerons enfin par la présentation de l'application.

3.2 Environnement de développement

3.2.1 Visual Studio code

Visual Studio Code est un éditeur de code extensible développé par microsoft pour Windows, Linux et macOS. Le meilleur choix pour JavaScript et les développeurs web, avec des extensions pour prendre en charge à peu près n'importe quel langage de programmation. Il est basé sur Electron, une structure utilisée pour déployer des applications Node.js pour le bureau exécuté sur le moteur Blink. Le code source est fourni sous la licence libre MIT sur le site du projet sur Github .[16]



3.2.2 Xampp

est un ensemble de logiciels permettant de mettre en place un serveur Web local, un serveur FTP et un serveur de messagerie électronique. Il s'agit d'une distribution de logiciels libres (X (cross) Apache MariaDB Perl PHP) offrant une bonne souplesse d'utilisation, réputée pour son installation simple et rapide. Ainsi, il est à la portée d'un grand nombre de personnes puisqu'il ne requiert pas de connaissances particulières et fonctionne, de plus, sur les systèmes d'exploitation les plus répandus.



Il est distribué avec différentes bibliothèques logicielles qui élargissent la palette des services de façon notable : OpenSSL, Expat (analyseur syntaxique de fichiers XML), PNG, SQLite, zlib... ainsi que différents modules Perl et Tomcat.

Officiellement, XAMPP permet de configurer un serveur de test local avant la mise en œuvre d'un site Web, et son usage n'est pas recommandé pour un serveur dit de production²[17]

3.2.3 Visual Paradigm

Visual Paradigm est une plateforme de développement logiciel qui fournit des outils pour créer, modéliser et concevoir des systèmes logiciels. Elle inclut des fonctionnalités telles que des diagrammes UML, des outils de gestion de projet Agile et des outils de collaboration. Le lien de téléchargement est :

<https://www.visual-paradigm.com/download/>

3.3 Les outils de développement d'application

3.3.1 HTML(Hyper Text Markup Language)

Le langage HTML est le langage universel utilisé sur les pages web lisibles par tous les navigateurs web (Internet Explorer, Netscape, Mozilla, etc...). Ce langage fonctionne suivant l'assemblage et la combinaison de balises permettant de structurer et donner l'apparence voulue aux données textes, images et multimédias suivant la mise en page voulue. C'est un langage qui permet de créer des hyperliens, à savoir des liens d'un document à un autre ou d'un endroit d'un document à un autre endroit du même document (identificateur de fragment)[29]



3.3.2 Css (Cascading Style Sheets)

CSS est l'acronyme de « Cascading Style Sheets » ce qui signifie « feuille de style en cascade ». Le CSS correspond à un langage informatique permettant de mettre en forme des pages web (HTML ou XML). Ce langage est donc composé des fameuses « feuilles de style en cascade » également appelées fichiers CSS. [18]



3.3.3 JS (JavaScript)

JavaScript (souvent abrégé en « JS ») est un langage de script léger, orienté objet, principalement connu comme le langage de script des pages web. Mais il est aussi utilisé dans de nombreux environnements extérieurs aux navigateurs web tels que Node.js, Apache CouchDB voire Adobe Acrobat. Le code JavaScript est interprété ou compilé à la volée (JIT). C'est un langage à objets utilisant le concept de prototype, disposant d'un typage faible et dynamique qui permet de programmer suivant plusieurs paradigmes de programmation fonctionnelle, impérative et orientée objet.[6]



3.3.4 PHP(Hypertext Preprocessor)

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 [19]



3.3.5 JQuery

JQuery est une bibliothèque JavaScript libre et multi-plateforme créée pour faciliter l'écriture de scripts côté client dans le code HTML des pages web.[20]

3.3.6 Bootstrap

Bootstrap est une collection d'outils utile à la création de sites et d'applications web. Développé par l'équipe du réseau social Twitter. Proposé en open source (sous licence MIT)Ce langage utilise les principaux langages de développement web (HTML, CSS Javascript). Il s'agit d'un code qui raccourcit différentes fonctionnalités (plugins utilisant la bibliothèque JQuery), permettant au développeur de gagner du temps et de réaliser simplement des codes complexes (animation, carrousel, tableau, histogramme, ...) et tout en réduisant la quantité de caractères requis, et donc le poids du site web , c'est l'un des projets les plus populaires sur la plateforme de gestion de développement GitHub. [21]



3.3.7 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.[22]



3.3.8 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.[23]

3.4 Présentation de l'application

Notre projet est composé d'une application web PharmaCHU qui permet aux personnel du laboratoire de faire la gestion des produits pharmaceutiques ainsi que la gestion des entrées et sorties.

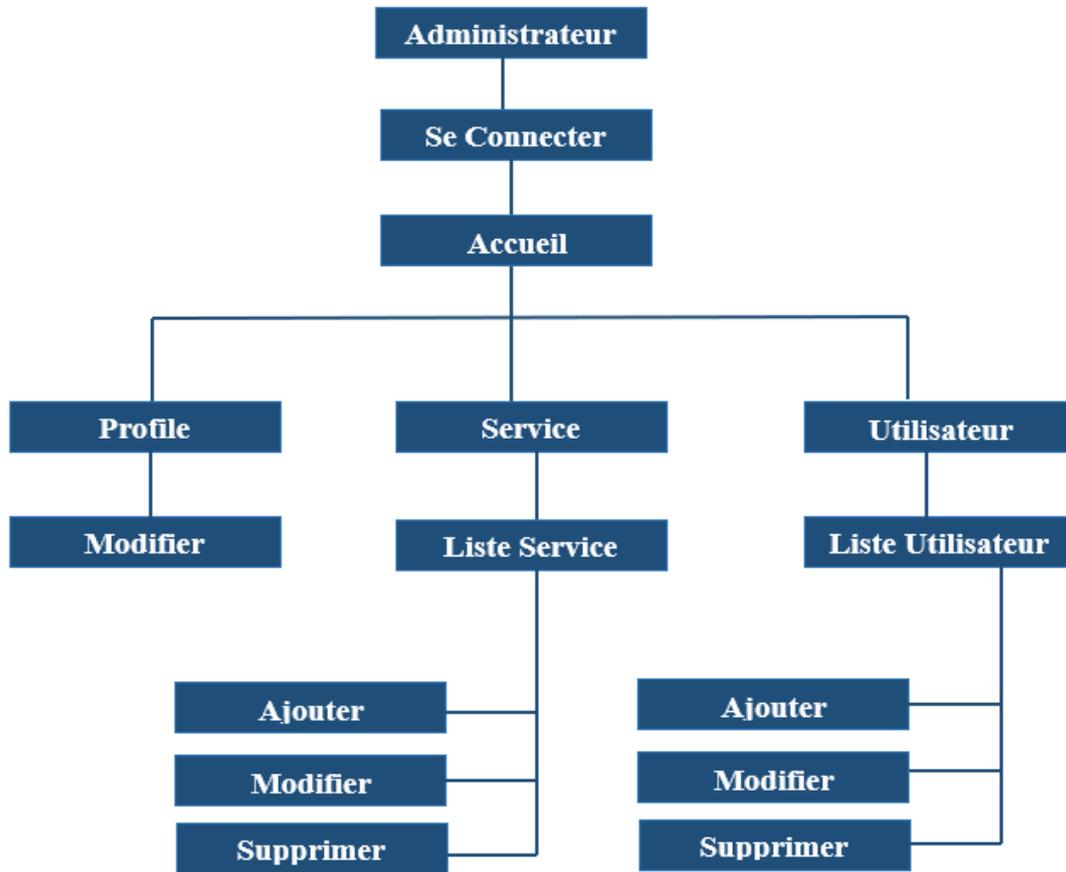


FIGURE 3.1 – Map de l'application- Administrateur.

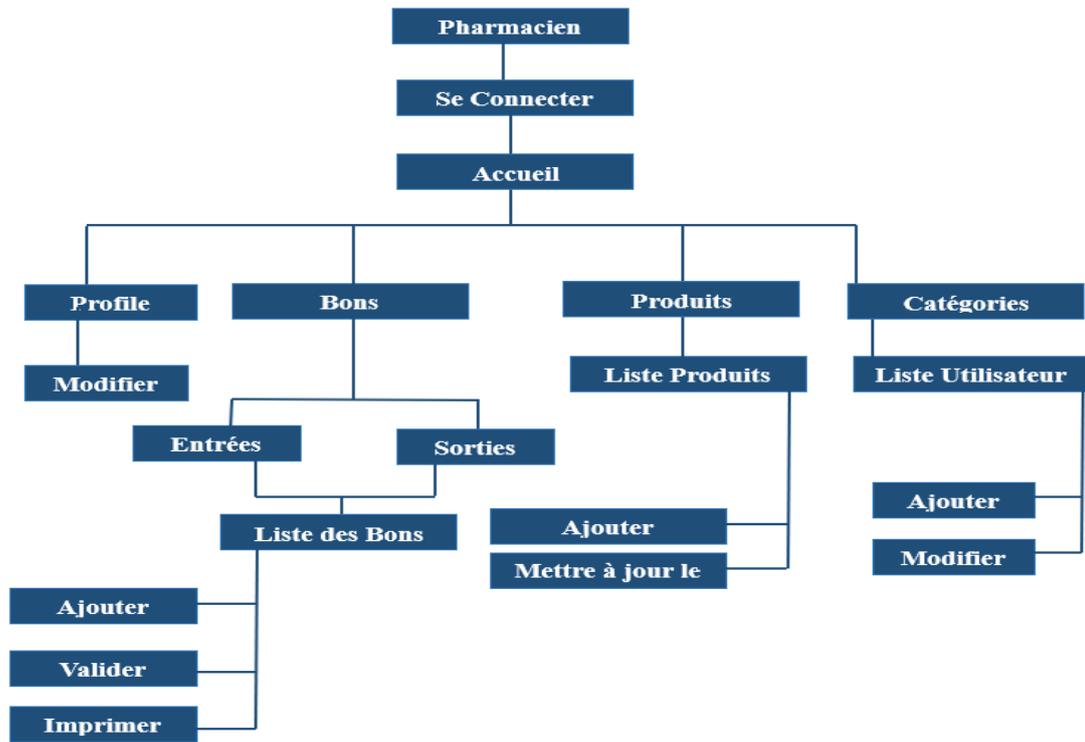


FIGURE 3.2 – Map de l'application- Pharmacien (chef/réactif).

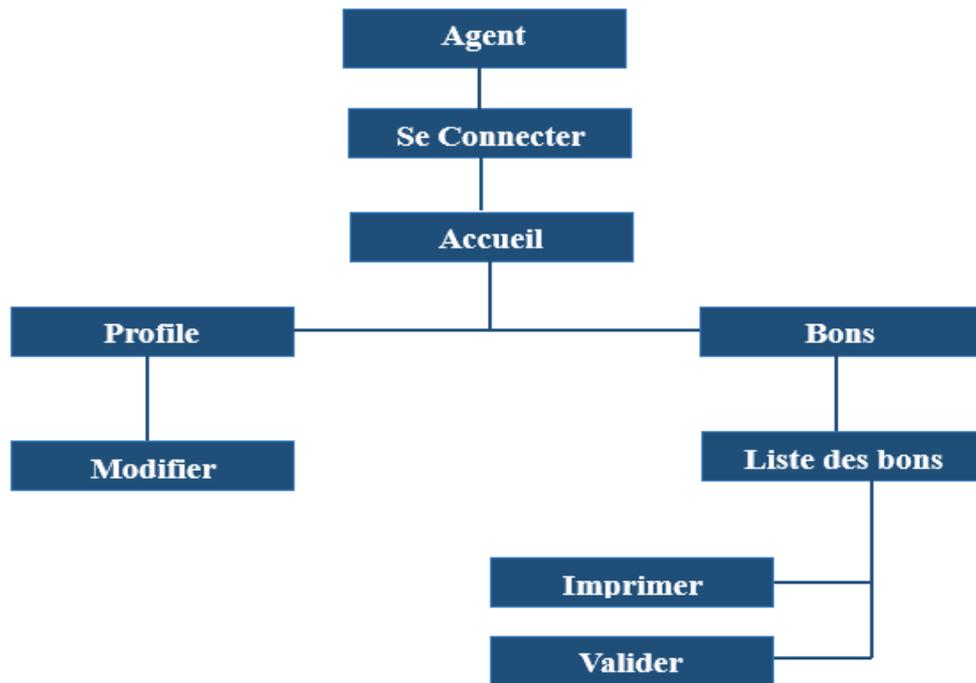


FIGURE 3.3 – Map de l'application- Agent.

3.4.1 Interface d'authentification

L'utilisateur (Administrateur/ pharmacien réactif/ pharmacien chef/ agent) doit remplir les deux champs d'authentification en saisissant les coordonnées correctes au qu'il puisse accéder a la page d'accueil.

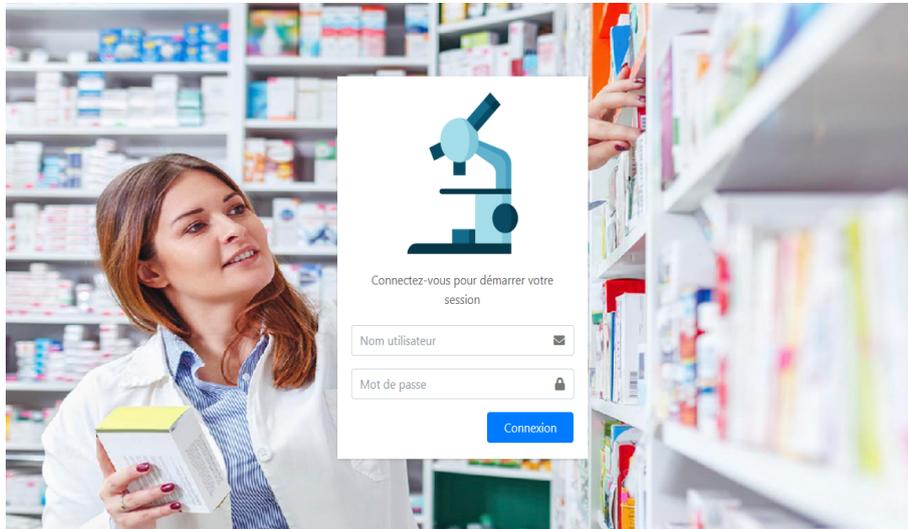


FIGURE 3.4 – Interface d'authentification

3.4.2 Espace Administrateur

3.4.2.1 Interface accueil

C'est la première interface qui s'affiche après l'authentification de l'administrateur



FIGURE 3.5 – Interface Accueil

3.4.2.2 Interface utilisateurs

Cette interface permet à l'administrateur de l'application de visualiser les comptes utilisateurs, avec comme options : ajouter ,supprimer un compte et modifier les informations d'un compte .

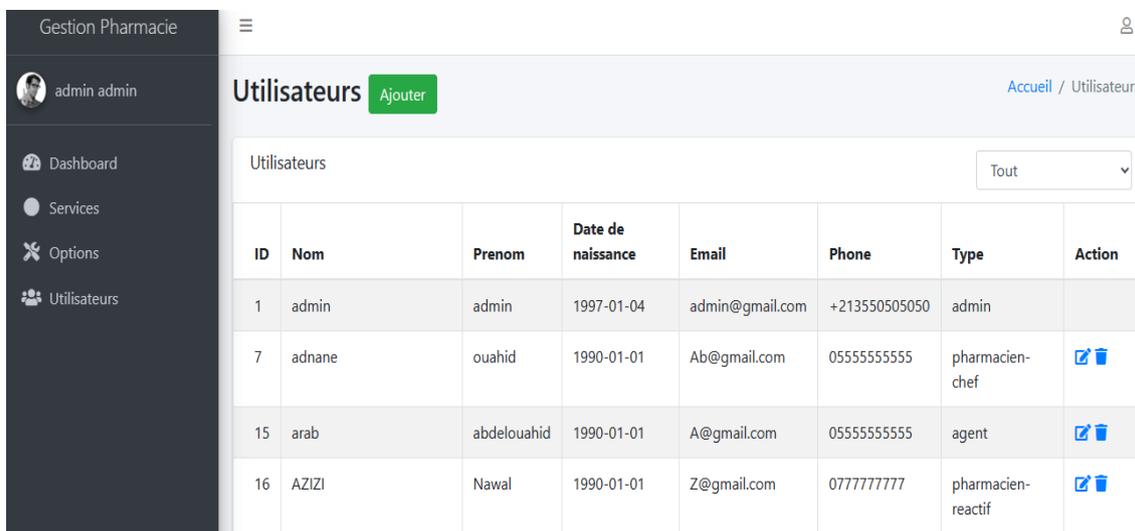


FIGURE 3.6 – Interface utilisateur

3.4.2.3 Interface ajout d'un utilisateur

Cette page permet a l'administrateur d'ajouter un compte utilisateur (pharmacien chef/pharmacien réactif/ agent) en saisissant les informations personnel de chaque utilisateur .

FIGURE 3.7 – Interface Ajouter Utilisateur

3.4.2.4 Interface service

Cette interface permet a l'administrateur de visualiser la listes des services avec comme options :ajouter,supprimer un service ou bien modifier les informations d'un service .

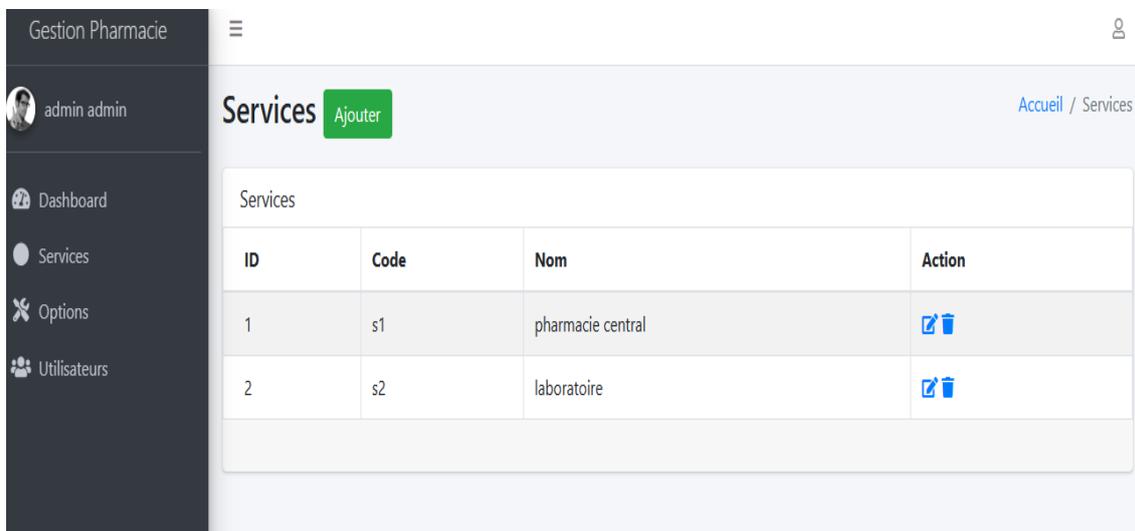


FIGURE 3.8 – Interface Service

3.4.3 Espace pharmacien

3.4.3.1 Interface D'accueil

Cette page représente la page d'accueil dont l'utilisateur (pharmacien chef/pharmacien réactif) accède après avoir fait son authentification puis il sélectionne une rubrique pour la fonctionnalité à gérer :

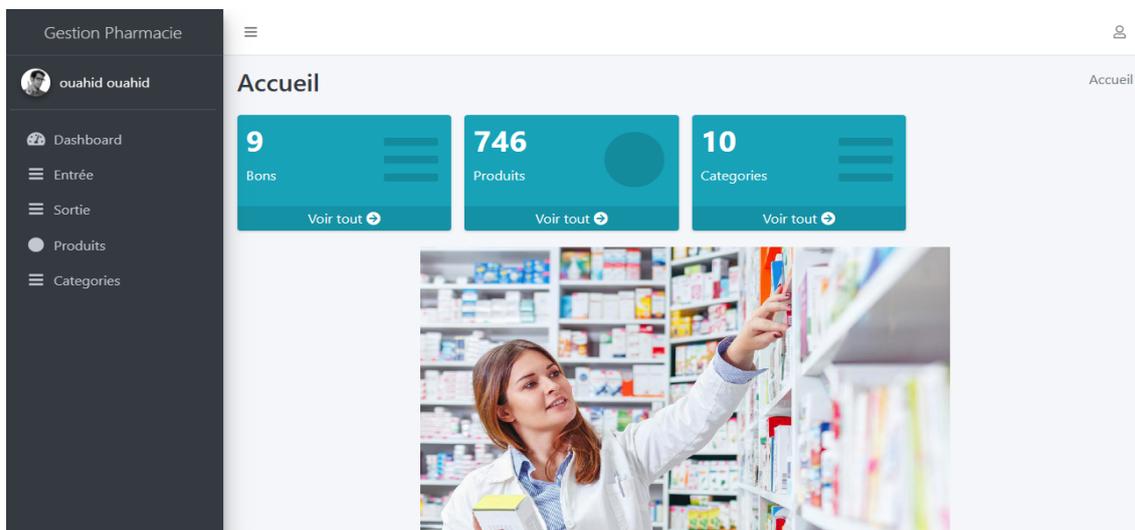


FIGURE 3.9 – Interface d'accueil pharmacien

3.4.3.2 Interface Bon Commande

Cette page permet a l'utilisateur(pharmacien chef/pharmacien réactif) de visualiser la liste des bons et aussi d'effectuer des bons avec plusieurs options comme :supprimer ,modifier ,imprimer.

The screenshot shows a web interface for 'Gestion Pharmacie'. The main content area is titled 'Bon commandes' and features a table with the following data:

ID	Utilisateur	Code	Date	Status	Imprimer	Action
18	adnane ouahid	33	2023-06-11 11:40:09	valide	Imprimer	
19	adnane ouahid	23	2023-06-11 11:41:02	non-valide	Imprimer	
20	adnane ouahid	333	2023-06-11 11:42:06	valide	Imprimer	
34	AZIZI Nawal	1	2023-06-26 12:32:04	valide	Imprimer	
35	AZIZI Nawal	2	2023-06-26 12:43:35	valide	Imprimer	
36	AZIZI Nawal	23	2023-06-26 12:46:13	valide	Imprimer	
37	AZIZI Nawal	4	2023-06-26 12:50:53	valide	Imprimer	

FIGURE 3.10 – Interface de Bon

3.4.3.3 Interface Ajouter un Bon Commande

Cette page montre comment ajouter un bon et les informations a remplir .

The screenshot shows the 'Ajouter' form for creating a new order. The form includes the following fields and sections:

- Code**: Text input field.
- Type**: Dropdown menu with 'sortie' selected.
- Status**: Dropdown menu with 'non-valide' selected.
- Service**: Dropdown menu with 'pharmacie central' selected.
- Observation**: Text area for notes.
- Produits**: A table for adding items to the order.

Categorie	Produit	Quantité	Action
Anatomie Pathologie	RECEPTEUR ESTROGENE,PREDILUE	1	

At the bottom of the form, there are buttons for '+', 'Cancel', and 'Ajouter'.

FIGURE 3.11 – Interface Ajouter Bon Commande

3.4.3.4 Imprimé de bon Commande

Après avoir cliquer sur imprimer le bon on aura cet affichage .

Service: laboratoire
Code: 22
Date: 2023-06-26

Bon Commande

N°	Code	Designation	Quantité demandé
01	731521	EGFR	16
02	731602	FUSHINE BASIQUE PHENIQUEE	5
03	731669	HBS	4
04	731971	ANA screen anti corps anti-antigènes soluble	10

signature médecin

FIGURE 3.12 – Imprimé de Bon Commande

3.4.3.5 Interface produit

Cette page montre la liste des produits réactifs avec les options rechercher un produit ,ajouter ,mettre a jour la liste de stock et aussi modifier les informations d'un produit .

☰
👤

Produits
Ajouter
Miss a jour stock
Accueil / Produits

Produits

🔍
Tout
▼

ID	Code	Categorie	Designation	Date peremption	Quantite initial	Quantite	Action
1	731231	biochimie	UREE SG	2024-07-09	15	41	✎ 🗑
2	731232	biochimie	CREATININE	2024-07-09	0	30	✎ 🗑
3	731233	biochimie	GLUCOSE	2024-07-09	0	0	✎ 🗑
4	731234	biochimie	CALCEMIE	2024-07-09	0	0	✎ 🗑
5	731235	biochimie	CALCIUM 0.025ML	2024-07-09	0	0	✎ 🗑
6	731236	biochimie	CHOLESTEROL	2024-07-09	0	2	✎ 🗑
7	731237	biochimie	TRIGLYCERIDE	2024-07-09	0	0	✎ 🗑

FIGURE 3.13 – Interface Produits

3.4.3.6 Interface Ajouter Produit

Après avoir choisi de cliquer sur Ajouter un produit cette page sera afficher elle contient un formulaire a remplir qui va permettre de saisir les informations d'un produit puis l'ajouter a la liste .

The screenshot shows a web interface titled 'Produits' with a breadcrumb 'Accueil / Produits / Nouveau'. Below the title is a form labeled 'Ajouter'. The form contains the following fields:

- Categorie:** A dropdown menu with 'Anatomie Pathologie' selected.
- Designation:** A text input field with the placeholder 'Designation ...'.
- Produit code:** A text input field with the placeholder 'Produit code ...'.
- Produit Quantité:** A text input field with the placeholder 'Produit Quantité ...'.
- Date péremption:** A date input field with the placeholder 'jj/mm/aaaa' and a calendar icon.

At the bottom right of the form are two buttons: 'Cancel' (grey) and 'Ajouter' (green).

FIGURE 3.14 – Interface Ajouter Produit

3.4.3.7 Interface Catégories

Cette page montre les différentes catégories des produits réactifs .

The screenshot shows a web interface titled 'Categories' with a breadcrumb 'Accueil / Categories' and a green 'Ajouter' button. Below the title is a table with the following data:

ID	Nom	Description	Action
1	Anatomie Pathologie		✎ 🗑
2	autres		✎ 🗑
3	biochimie		✎ 🗑
4	colorants et produits chimiques		✎ 🗑
5	consommables laboratoires		✎ 🗑
6	Hemobiologie		✎ 🗑
7	immunologie		✎ 🗑
8	microbiologie		✎ 🗑

FIGURE 3.15 – Interface Catégories

3.4.4 Espace Agent

3.4.4.1 Interface D'accueil

Cette interface montre les fonctionnalités qu'un agent peut faire .

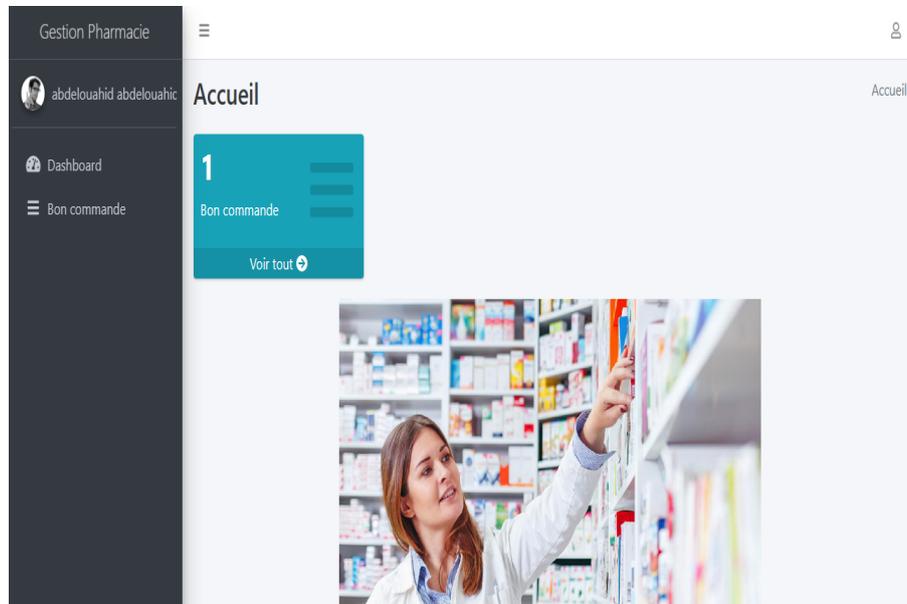


FIGURE 3.16 – Interface D'accueil Agent

3.4.4.2 Interface Liste des bons

Cette page affiche la liste des bons de commande envoyer par le laboratoire vers d'autres services.

ID	Utilisateur	Code	Date	Status	Imprimer	Action
40	adnane ouahid	22	2023-06-26 18:17:34	non-valide	Imprimer	Valider
18	adnane ouahid	33	2023-06-11 11:40:09	valide	Imprimer	
20	adnane ouahid	333	2023-06-11 11:42:06	valide	Imprimer	
34	AZIZI Nawal	1	2023-06-26 12:32:04	valide	Imprimer	
35	AZIZI Nawal	2	2023-06-26 12:43:35	valide	Imprimer	
37	AZIZI Nawal	4	2023-06-26 12:50:53	valide	Imprimer	
38	AZIZI Nawal	44	2023-06-26 12:54:28	valide	Imprimer	

FIGURE 3.17 – Interface Liste des Bons

3.4.4.3 Imprimé de Bon Livrison

Après avoir cliquer sur imprimer le bon on aura cet affichage.

Service: pharmacie central
Code: 33
Date: 2023-06-11

Bon livraison

N°	Code	Designation	Quantité demandé	Quantité livrée
01	731236	CHOLESTEROL	8	2
02	731508	RECEPTEUR ESTROGENE,PREDILUE	1	1
03	731520	DOG1	3	1

signature médecin

FIGURE 3.18 – Imprimé de Bon Livrison

3.5 Conclusion

Ce chapitre a été consacré à la phase de réalisation. Cette phase est le fruit des nos efforts tout au long de la durée du projet. Dans ce chapitre, nous avons présenté les outils et langages utilisés pour la réalisation de notre application, montré les interfaces que nous avons jugé les plus importantes.

Conclusion générale

En conclusion, ce mémoire s'est penché sur le développement d'une application web de gestion de produits pharmaceutiques au sein du CHU de Béjaia qui offre une solution innovante pour améliorer les processus liés à la gestion des médicaments. Nous avons souligné l'importance de l'informatique dans le secteur pharmaceutique, en particulier pour optimiser la gestion de l'information, améliorer l'organisation du travail et faciliter l'accès aux données précises et fiables.

L'introduction a mis en évidence les défis auxquels est confrontée la gestion traditionnelle des produits pharmaceutiques, tels que la perte de temps et la dégradation de l'information due à des processus manuels. L'application web de gestion de produits pharmaceutiques a été présentée comme une solution prometteuse pour résoudre ces problèmes en offrant un moyen sûr et efficace de traiter et de sauvegarder l'information.

Nous avons également abordé les objectifs de ce mémoire, qui consistent à étudier et à concevoir une application web de gestion de produits pharmaceutiques en utilisant UML comme langage de modélisation et la méthode agile "2TUP" pour le processus de développement. Le choix des langages de programmation HTML, CSS et JavaScript, ainsi que la base de données MySQL, a été justifié pour l'implémentation de l'application.

Les chapitres ont détaillé la méthodologie et la conception du projet, en mettant l'accent sur les étapes du processus de développement et en utilisant le langage de modélisation UML et le processus unifié UP. Nous avons également présenté les résultats de notre réalisation, en montrant quelques interfaces de l'application web développée.

Et pour finir, ce mémoire a permis d'explorer les possibilités offertes par une application web de gestion de produits pharmaceutiques, en mettant en évidence les avantages tels que l'op-

timisation des processus, l'organisation du travail, l'accès facile aux informations et la fiabilité des données. Cette solution technologique peut contribuer à l'amélioration globale de la gestion des produits pharmaceutiques et à l'efficacité opérationnelle des pharmacies et des acteurs de l'industrie pharmaceutique.

Il est important de souligner que ce mémoire constitue une base solide pour d'éventuelles améliorations et développements futurs de l'application web de gestion de produits pharmaceutiques. La mise en œuvre réussie de cette solution nécessitera une attention particulière à la sécurité des données, à la conformité réglementaire et à l'adaptation aux besoins spécifiques de chaque entreprise pharmaceutique. Des perspectives d'amélioration de notre application restent toutefois indispensables. Nous envisageons ainsi d'ajouter de nouvelles fonctionnalités telles que :

- Faire une sorte d'affichage à chaque fois que la date de péremption approche .
- Rajouter la fonctionnalité qui permet de calculer le prix des produits pharmaceutiques .

Bibliographie

- [1] ARCHITECTURE D'UN RÉSEAU INFORMATIQUE <https://sti.ac-versailles.fr/IMG/pdf/reseau.pdf> / consultation le 20/03/2023.
- [2] Client-serveur - Définition <https://www.techno-science.net/definition/3743.html> / consultation le 10/2/2023.
- [3] Réseaux - Architecture d'égal à égal <https://web.maths.unsw.edu.au/~lafaye/CCM/initiation/peer.htm/> / consultation le 10/2/2023.
- [4] Applications informatique : <https://www.ideematic.com/dictionnaire-digital/> / consultation le 14/03/2023.
- [5] Application desktop : https://www.streetdirectory.com/travel_guide/114448/programming/desktop_applications_vs_web_applications.html / consultation le 14/03/2023.
- [6] Javascript <https://developer.mozilla.org/fr/docs/Web/JavaScript/> / consultation le 10/2/2023.
- [7] ApplicationsWeb <https://www.ideematic.com/dictionnaire-digital/application-web/> / consultation le 14/03/2023.
- [8] ApplicationsWeb <https://www.techno-science.net/glossaire-definition/Application-Web.html> / consultation le 14/03/2023.
- [9] Type Applications Web <https://fr.yeeply.com/blog/5-types-developpement-d-applications-web/> / consultation le 14/03/2023.
- [10] Définition CHU <https://www.chubejaia.dz/> / consultation le 10/2/2023.

- [11] 2TUP <https://www.rapport-gratuit.com/description-de-2tup-et-du-processus-de-developpement-en-y/> consultation le 28/05/2023.
- [12] Historiquedel'UML
https://www.memoireonline.com/12/19/11386/m_Conception-et-realisation-d-une-application-web-de-la-gestion-de-recrutement-militaire-pour-le-cas9.html consultation le 14/03/2023.
- [13] Les Points forts et faibles d'UML <http://dspace.univtlemcen.dz/bitstream/112/5500/5/chapitre1.pdf> consultation le 10/2/2023.
- [14] Diagramme de cas d'utilisation <https://www.ibm.com/docs/fr/rational-soft-arch/9.5?topic=diagrams-use-case/> consultation le 10/2/2023.
- [15] Les diagrammes UML <http://e-biblio.univ-mosta.dz/bitstream/handle/123456789/20464/MINF292.pdf?sequence=1&isAllowed=y> consultation le 14/03/2023.
- [16] Visual code <https://www.blogdumoderateur.com/tools/visual-studio-code/> consultation le 15/4/2023.
- [17] Définition de XAMPP
<https://desgeeksetdeslettres.com/xampp-plateforme-pour-heberger-son-propre-site-web/> consultation le 20/04/2023.
- [18] CSS <https://www.atinternet.com/glossaire/css/> consultation le 10/2/2023.
- [19] PHP <https://www.ionos.fr/digitalguide/serveur/know-how/quest-ce-que-mysql/> consultation le 10/04/2023.
- [20] JQuery <https://fr.wikipedia.org/wiki/JQuery> consultation le 10/04/2023.
- [21] Bootstrap :
<https://agency-inside.com/2016/06/01/definition-webmarketing-bootstrap/> consultation le 10/04/2023.
- [22] Phpmyadmin :
<https://www.ionos.fr/digitalguide/serveur/know-how/quest-ce-que-mysql/> consultation le 10/04/2023.
- [23] MYSQL :
<https://www.ionos.fr/digitalguide/serveur/know-how/quest-ce-que-mysql/> consultation le 10/2/2023.

- [24] Diagramme de cas d'utilisation https://atefsd.weebly.com/uploads/5/0/3/6/503639/csi_02_chap02.pdf consultation le 23/04/2023.
- [25] langage de modélisation unifié <https://www.lucidchart.com/pages/fr/> consultation le 23/04/2023.
- [26] Application <https://www.numidev.fr/une-application-mobile-cest-quoi/> consultation le 10/2/2023.
- [27] Processus2TUP <https://www.rapport-gratuit.com/description-de-2tup-et-du-processus-de-dveloppement-en-y/> consultation le 12/03/2023.
- [28] BEN ALI, K., ET AL. *REALISTION ET CONCEPTION DE GESTION DE L'HOPITAL*. 2016.
- [29] CHARNAY, D., AND CHALÉAT, P. *HTML et Javascript*. Eyrolles, 1998.
- [30] DE NICE, I. Concepts et langages des bases de données relationnelles.
- [31] KHALED, B. A. *REALISTION ET CONCEPTION DE GESTION DE RENDEZ-VOUS DE L'HOPITAL*. 2016.
- [32] MANSOURI, B. *Un système de prise de rendez-vous en ligne pour une clinique*. PhD thesis, 2017.
- [33] MR.YAKOUBEN DJAAFAR, M. M. S. *Conception et réalisation d'une application Androïde : Localisation des boutiques, hôtels....* 2015.
- [34] NAIMA, K. *Conception et Réalisation d'une Application web pour la Gestion des Rendez-vous Médicaux*. 2021.
- [35] ROQUES, P. *Modéliser une application web*. 2008.
- [36] VALLÉE, P. R. F. *De l'analyse des besoins à la conception*. 2007.
- [37] Z.LAAREDJ. *cours génie logiciel*. 2020.

Résumé

Dans ce mémoire, nous avons conçu et réalisé une application web de gestion des produits pharmaceutiques pour le laboratoire de la pharmacie centrale du CHU de Bejaia dans le but de faciliter certaines tâches au profit des employés de la pharmacie

Pour ce travail nous avons utilisé le processus de conception 2TUP et UML comme langage de modélisation, nous avons choisi de programmer l'application avec la collection d'outils design BOOTSTRAP qui contient des codes HTML et CSS et le langage PHP Objet avec MySQL comme SGBD.

Mots clés : *Application Web, CHU bejaia, Gestion des produits pharmaceutiques, HTML, CSS, UML, PHP*

Abstract

In this thesis, we designed and implemented a web application for managing pharmaceutical products for the central pharmacy laboratory at the Bejaia University Hospital (CHU). The purpose of this application is to facilitate certain tasks for the pharmacy employees.

For this project, we used the 2TUP design process and UML as the modeling language. We chose to program the application using the BOOTSTRAP design toolkit, which includes HTML and CSS code, and the PHP Object language with MySQL as the database management system (DBMS).

Keywords : *WEB APPLICATION, CHU bejaia, pharmaceutical product management, HTML, CSS, UML, PHP.*