



**UNIVERSITE CADI AAYAD MARRAKECH**

**Ecole Nationale des Sciences Appliqués**

**RAPPORT DE PROJET JAVA**

---

**ETUDE ET DEVLEOPPEMENT D'UNE  
APPLICATION POUR LA GESTION D'UNE  
PHARMACIE**

---

**Présenté par :**

- MAZALI Omar Abdessalam
- LACHHAB Abir
- OURAKH Ismail
- BENHEBIBI Mohamed Ziad
- AIT SLIMAN Mounir

**Encadré par :**

Mr. NEJEoui Abderrazak

Année universitaire : 2022-2023

## REMERCIEMENTS

Au terme de la rédaction de ce rapport, c'est un devoir agréable d'exprimer en quelques lignes la reconnaissance que nous devons à tous ceux qui ont contribué de loin ou de près à l'élaboration de ce travail, qu'ils trouvent ici nos vifs respects et notre profonde gratitude.

Nous remercions les professeurs chez qui nous avons pu assister durant notre cursus universitaire, nous remercions tout particulièrement notre cher encadreur Mr. NEJEUI ABDERRAZAK pour ses conseils avisés et ses remarques pertinentes.

Nos remerciements s'étendent également au Chef de département ainsi qu'à tous nos professeurs et examinateurs de l'Ecole Nationale des Sciences Appliquées pour la richesse et la qualité de leur enseignement et qui déploient de grands efforts pour assurer à leurs étudiants une formation actualisée.

De façon plus générale, je tiens à adresser mes plus sincères remerciements à l'ensemble du corps professoral de l'Ecole Nationale des Sciences Appliquées (ENSA), et spécialement au Département Informatique pour avoir porté un vif intérêt à ma formation.

Que tous ceux et celles qui ont contribué de près ou de loin à l'accomplissement de ce travail trouvent l'expression de nos remerciements les plus chaleureux.

## LISTE D'ABREVEATIONS

**API** – Application Programming Interface

**CSS** – Cascading Style Sheets

**IT** – Information Technology

**JDK** – Java Development Kit

**UML** – Unified Modeling Language

**SI** – Système d'information

**DB** – Database

**SGBD** – Système de Gestion de Base de Données

# TABLE DES FIGURES

FIGURE 1 : PHARMACIE .....	10
FIGURE 2 : EXEMPLE DE DIAGRAMME DE CAS D'UTILISATION.....	18
FIGURE 3 : TABLEAU DES CAS D'UTILISATION .....	19
FIGURE 4 : DIAGRAMME DE CAS D'UTILISATION « AUTHENTIFICATION » .....	20
FIGURE 5 : DIAGRAMME DE CAS D'UTILISATION « INSCRIPTION » .....	20
FIGURE 6 : DIAGRAMME DE CAS D'UTILISATION « GESTION DE STOCK » .....	21
FIGURE 7 : DIAGRAMME DE CAS D'UTILISATION « GESTION DES VENTES » .....	22
FIGURE 8 : DIAGRAMME DE CAS D'UTILISATION « GESTION DES MEDICAMENTS ».....	23
FIGURE 9 : DIAGRAMME DE SEQUENCE DE L'AUTHENTIFICATION « EMPLOYE » .....	24
FIGURE 10 : DIAGRAMME DE SEQUENCE DE L'AUTHENTIFICATION « PHARMACIEN » .....	25
FIGURE 11 : DIAGRAMME DE SEQUENCE DE LA MISE A JOUR SU STOCK .....	26
FIGURE 12 : DIAGRAMME DES CLASSES .....	27
FIGURE 13 : LISTE DES TABLES DE LA BASE DE DONNEES .....	33
FIGURE 14 : DESCRIPTION ET BASE DE DONNEES DE LA TABLE ADMIN .....	33
FIGURE 15 : DESCRIPTION ET BASE DE DONNEES DE LA TABLE CUSTOMER .....	34
FIGURE 16 : DESCRIPTION ET BASE DE DONNEES DE LA TABLE SELLERS .....	34
FIGURE 17 : DESCRIPTION ET BASE DE DONNEES DE LA TABLE STOCK .....	35
FIGURE 18 : DESCRIPTION ET BASE DE DONNEES DE LA TABLE USERS .....	35
FIGURE 19 : INTERFACE D'AUTHENTIFICATION D'UN EMPLOYE .....	36
FIGURE 20 : MESSAGE D'ERREUR D'AUTHENTIFICATION « EMPLOYE » .....	37
FIGURE 21 : INTERFACE D'INSCRIPTION « EMPLOYE » .....	38
FIGURE 22 : MESSAGE D'ERREUR D'INSCRIPTION « EMPLOYE » .....	38
FIGURE 23 : INTERFACE D'ACCUEIL « EMPLOYE » .....	39
FIGURE 24 : INTERFACE PIN .....	39
FIGURE 25 : INTERFACE D'ACCUEIL « EMPLOYE » .....	40
FIGURE 26 : INTERFACE D'ACCUEIL « PHARMACIEN » .....	41
FIGURE 27 : INTERFACE DE GESTION DES UTILISATEURS .....	41
FIGURE 28 : INTERFACE DE GESTION DES UTILISATEURS .....	42
FIGURE 29 : INTERFACE D'ACHAT DES MEDICAMENTS .....	43
FIGURE 30 : INTERFACE DE VENTE DES MEDICAMENTS .....	43

# TABLE DES MATIERES

<b>Introduction générale</b> .....	<b>7</b>
<b>Contexte générale du projet</b> .....	<b>9</b>
Introduction.....	10
Présentation de la pharmacie .....	10
Enjeux et vision du projet .....	11
Description de l'existant .....	11
Etude de l'existant .....	11
Présentation de l'application .....	12
Critique de l'existant.....	12
Orientations (Solutions).....	13
Methodologie adoptée.....	13
Processus unifié.....	13
Spécification des besoins.....	14
Identification des besoins .....	14
Identification des cas de système .....	15
<b>Conception</b> .....	<b>17</b>
Introduction.....	18
Description du modèle des cas d'utilisation .....	18
Diagramme de cas d'utilisation.....	19
Diagramme de cas d'utilisation « authentification » .....	19
Diagramme de cas d'utilisation « inscription » .....	20
Raffinement des cas d'utilisations .....	21
Gestion des stock .....	21
Gestion des ventes.....	22
Gestion des médicaments .....	23
Diagramme de séquence.....	23
Diagramme de séquence d'authentification « employé » .....	24
Diagramme de séquence d'authentification « pharmacien » .....	24
Diagramme de séquence pour ajouter un nouveau produit .....	25
Construction du diagramme des classes.....	26
Diagramme des classes .....	27
Description des classes.....	27
Descriptions des associations .....	27
Conclusion .....	28
<b>Etude technique et implémentation</b> .....	<b>29</b>

Introduction.....	30
Outils et langes utilise.....	30
Environnement matériel .....	30
Environnement logiciel.....	30
Conception de la base de données.....	33
Réalisation du projet .....	36
Interface authentification « employé ».....	36
Interface d'inscription « employé » .....	37
Interface d'accueil « employé » .....	39
Interface authentification « pharmacien ».....	39
Interface d'inscription « pharmacien ».....	40
Interface d'accueil « pharmacien » .....	40
Interfaces d'achat et ventes .....	42
Conclusion .....	44
<b>Conclusion générale et perspective.....</b>	<b>46</b>

# INTRODUCTION GENERALE

Avant l'invention de l'ordinateur, on enregistrait 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 données des entreprises, ce qui est la partie essentielle dans leur développement aujourd'hui.

Les pharmacies hospitalières et celles des dispensaires publics font partie intégrante des établissements que l'informatique pourra beaucoup aider. En effet, la croissance du nombre des médicaments hospitaliers nécessite la mise en place d'une gestion rationnelle prise et rapide, or et jusqu'à ce jour, la manière de gérer manuellement est encore dominante.

On remarque ainsi la mauvaise organisation du travail dans la pharmacie lors de la recherche d'une information ainsi lors de la création des statistiques l'information n'est pas toujours précise ni disponible d'où la nécessité d'introduire l'informatique dans les pharmacies hospitalières.

Vu cet état de fait, notre projet a pour objectif de concevoir et mettre en place une application web interactive, fiable, conviviale et facile à intégrer dans l'environnement de travail des pharmacies assurant la gestion de ces dernières et de suivre les ordonnances en prenant en considération le type des médicaments sortis à chaque ordonnance et l'état de stock des médicaments, en essayant de trouver les solutions aux problèmes rencontrés lors de l'exécution.

Cette application vise essentiellement à diminuer la complexité des traitements ainsi que le temps perdu lors de la gestion de stock, en particulier.

Dans ce travail, nous avons commencé par faire une présentation de projet, ainsi le choix méthodologique suivi. Nous avons spécifié par la suite les besoins fonctionnels de l'application et le diagramme du cas d'utilisation général du système.

Par la suite nous avons étudié la partie conception qui contiendra une description détaillée de cas d'utilisation, les diagrammes de séquence ainsi que le diagramme de classe détaillé.

Après avoir achevés la partie conception, nous avons définis dans le dernier chapitre l'étude technique en précisant l'outil de travail et l'implémentation du projet.

Nous avons aussi illustré quelques interfaces qui donnent une idée sur les fonctionnalités de l'application.



# CHAPITRE 1

---

## CONTEXTE GENERALE DU PROJET

# **1. Contexte générale du projet**

## **1.1. Introduction**

De nos jours, il n'est plus aisé de définir les enjeux des nouvelles technologies de la gestion dans la professionnalisation de pharmacologie. En effet, elles représentent un véritable aide dans le travail du pharmacien et elles assurent la communication et la collaboration avec ses correspondants et patients. Ce chapitre va présenter le cahier des charges du projet. Dans lequel on va effectuer une étude de l'existant du projet et les solutions proposées et on précisera les besoins fonctionnels et non fonctionnels.

## **1.2. Présentation de la pharmacie**

Dans l'exercice de la pharmacie, tout pharmacien, en fonction du poste qu'il occupe, peut être appelé à utiliser des notions de base en gestion des ressources humaines, financières et matérielles. Planifier, organiser, budgéter, diriger, gérer, contrôler et évaluer représentent des fonctions de gestion nécessaires au maintien d'un environnement professionnel. Ceci permet d'assurer aux patients l'accès aux services et aux produits requis pour répondre à leurs besoins.

L'application, dans son milieu d'exercice, des principes de base et des habiletés de gestion permet au pharmacien d'optimiser les soins pharmaceutiques et les relations interprofessionnelles. Elle contribue également à assurer et à maintenir l'hygiène, la sécurité, l'efficacité et l'efficience du milieu de travail



FIGURE 1 : PHARMACIE

### **1.3. Enjeux et vision du projet**

Comme de nombreuses autres entreprises, les pharmacies sont en face de la contrainte de gestion des stocks. De ce fait, afin de satisfaire au mieux la clientèle en quantité, qualité, délais, ... Et de réduire son coût d'exploitation, les stocks constituent un maillon essentiel pour le bon fonctionnement de la structure. D'où l'incontournable problème de gestion de stocks. En effet, la gestion de stocks a pour but de maintenir à un seuil acceptable, le niveau de service pour lequel le stock considéré existe. Ceci est fait grâce à des coûts relativement faibles. En effet, la gestion de stocks n'a pas d'objectifs absolus valables pour toutes les entreprises, pour tous les produits ou pour toutes les catégories de stocks. L'objectif correspondra toujours à un contexte particulier, de plus il ne sera pas figé mais évoluera dans le temps. Dans le cas de la pharmacie, elle a pour objectif la satisfaction continue de la demande de sa clientèle en produits pharmaceutiques, cela à moindre coût.

### **1.4. Description de l'existant**

#### **1.4.1. Etude de l'existant :**

Ce projet consiste à créer une application permettant de gérer une pharmacie, il s'agit de définir les responsabilités de la gestion, mettre à jour les données, organiser des données collectées, de renforcer le contrôle et la confrontation, assurer une meilleure gestion et une cohérence de l'information et enfin faciliter le travail des responsables.

#### **1.4.2. Présentation de l'application :**

Cette application que nous nous sommes proposés de développer pour l'automatisation de la gestion de la majorité des activités existantes dans la pharmacie, doit permettre de répondre aux exigences de cette dernière. Pour le développement de cette application, nous avons jugé nécessaire d'utiliser les différents outils et méthodes qui sont les suivants :

- Pour la Programmation, nous utilisons **Langage java**.
- Pour le traitement de texte, nous travaillons avec **Microsoft Word**.
- Nous utilisons la base de données **MySQL**.

### **1.4.3. Critique de l'existant :**

L'analyse de l'existant met l'accent sur plusieurs difficultés telles que :

- Le travail de certaines pharmacies publiques se fait encore manuellement.
- Négligence du facteur temps : le facteur temps est un facteur fondamental pour toutes activités dans le centre médical et vu que les tâches destinées au responsable de pharmacie, pour bien gérer le stock des médicaments, il sera difficile de réussir cette tâche manuellement, aussi bien pour les différentes ordonnances que pour les statistiques qui lui sont associées.
- Mal organisation du travail dans la pharmacie.
- Les documents (fiche de produit, le stock, bon de livraison, etc...) ne sont pas bien détaillés.
- Volume important des informations traitées manuellement, ce qui provoque parfois des erreurs dans l'établissement des documents.
- Recherche difficile sur les registres qui engendre une perte de temps
- Insécurité des informations.
- Possibilité d'erreur dans le remplissage des différents documents et registres.
- Possibilité d'erreur dans les calculs des statistiques.
- Nombre important des archives qui engendre une difficulté de stockage. (Détérioration des archives à force de leur utilisation trop fréquente.
- Mauvaise codification sur quelques objets dans la gestion d'information.

### **1.4.4. Orientations (Solutions) :**

Afin de corriger les problèmes présentés ci-dessus, nous sommes appelés à réaliser cette application qui assure les points suivants :

- Automatiser les tâches qui se traitent manuellement.
- Faciliter la recherche et l'accès aux informations.
- Sauvegarder toutes les données relatives à la gestion des ordonnances sur des supports informatiques ce qui assurera leur sécurité.
- Minimiser les supports papiers utilisés.

- Faire toute modification (ajout, suppression, modification) automatiquement.
- Plus d'organisation dans le travail du responsable de pharmacie.
- Faciliter la recherche de l'information.
- Rapidité dans l'établissement des différents documents.
- Gain de temps dans les calculs des statistiques.1 Proposer une bonne codification.

## **1.5. Méthodologie adoptée**

L'objectif de toute approche de conduite de projet est d'obtenir des résultats fiables. En fait, la fiabilité d'un système dépend de l'approche utilisée. Nous avons adopté pour un processus de développement logiciel appelé Processus Unifié.

### **1.5.1. Processus unifié :**

Le processus unifié est un processus de développement logiciel qui regroupe les activités à mener pour transformer les besoins d'un utilisateur en système logiciel. Mais c'est plus qu'un simple processus. C'est un Framework de processus générique pouvant être adapté à une large classe de systèmes logiciels, à différents domaines d'application, à différents types d'entreprises, à différents niveaux de compétence et à différentes tailles de projets.

Ce processus simplifié aux caractéristiques suivantes :

- Piloté par les cas d'utilisation d'UML
- Ne néglige pas l'analyse et la conception
- Utilise 20% d'UML pour modéliser 80% du système

Le processus simplifié est composé des phases suivantes :

- Étude des besoins (qui est détaillé en ce qui va suivre)
- Analyse
- Conception – Implémentation

### **1.5.2. Spécification des besoins :**

Cette phase consiste à comprendre le contexte du système. Il s'agit de déterminer les fonctionnalités et les acteurs les plus pertinents, de préciser les risques les plus critiques et d'identifier les cas d'utilisation initiaux

### **1.5.3. Identification des besoins :**

#### **1.5.3.1. Les besoins fonctionnels**

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

Les fonctionnalités que doit intégrer l'application à développer peuvent être décrites comme suit :

- **Gestion des sécurités :** Le Système permet de gérer les droits d'accès de chaque utilisateur en associant un ID et un mot de passe pour chaque utilisateur (pharmacien ou employé).
- **Gestion des médicaments :** Cette opération consiste à ajouter un nouveau médicament dans le stock, modifier ou supprimer un médicament qui existe déjà, à savoir les mouvements réalisés sur le stock (entrée /sortie de médicament, quantité des médicaments dans le stock)
- **Gestion des ventes :** Lorsqu'un client lui présente une ordonnance ou veut acheter un médicament précis, le pharmacien ou un de ses employés doit vérifier la disponibilité de chaque médicament dans le stock, le logiciel doit être capable d'indiquer au pharmacien la quantité de stock restante dans le stock pour chaque médicament demandé par le client ainsi que leur prix. Ce n'est qu'après cette étape de vérification que le pharmacien peut effectuer la vente en saisissant dans le logiciel le nom et la quantité de médicament vendu, ce dernier doit lui retrouver le prix de chaque médicament ainsi que le montant total qui sera encaissé par le pharmacien.
- **Gestion de stock :** Lorsqu'un fournisseur lui livre les médicaments commandés, le propriétaire de la pharmacie doit implémenter la base de données du logiciel par le nombre de médicaments livrés, et vérifier leurs dates de péremption.

#### **1.5.3.2. Les besoins non fonctionnels**

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

- **Authentification** : le système doit permettre à l'utilisateur de saisir son login et son mot de passe pour accéder au système. Cette opération assure la sécurité du système et limite le nombre des utilisateurs.
- **Ergonomie** : le système devra offrir aux utilisateurs une interface qui soit le plus riche possible afin de limiter le nombre d'écrans. Par ailleurs, l'interactivité devra être adaptée (usage du clavier, menu, etc..).

## 1.6. Identification des acteurs du système

Un acteur représente un rôle joué par une personne ou une chose qui interagit avec un système. En réponse à l'action d'un acteur, le système fournit un service qui correspond à son besoin.

Les différents acteurs définis pour notre système sont les suivants

### PHARMACIEN (PRINCIPAL) :

- Possède des droits sur la gestion du stock pharmaceutique. Possède des droits sur la gestion des sécurités.
- Possède des droits sur la gestion des médicaments.
- Possède des droits sur la gestion des ventes.

### EMPLOYE (SECONDAIRE) :

- Possède des droits sur la gestion des médicaments. Possède des droits sur la gestion des ventes.
- Possède des droits sur la gestion du stock pharmaceutique.





# CHAPITRE 2

---

## CONCEPTION

-

## 2. Conception

### 2.1. Introduction

Dans ce chapitre, le diagramme global des cas d'utilisation du système ainsi que les autres cas d'utilisation en particulier seront présentés. Cela est fait en utilisant les acteurs qui interagissent avec le système. On couvre une vue dynamique du système à travers les différents diagrammes de séquences relatifs aux cas d'utilisations. Enfin, on dégage les différentes tables de la base de données via le diagramme de classe.

### 2.2. Description du modèle d'utilisation

Dans un diagramme de cas d'utilisation (Figure 2), elles peuvent être reliées par une relation de généralisation ou de dépendance (inclusion ou extension).

La relation de généralisation entre cas d'utilisation signifie que le cas d'utilisation père peut être substitué par le cas d'utilisation fils. La relation de dépendance d'inclusion est utilisée pour éviter de décrire plusieurs fois les mêmes collaborations d'objets

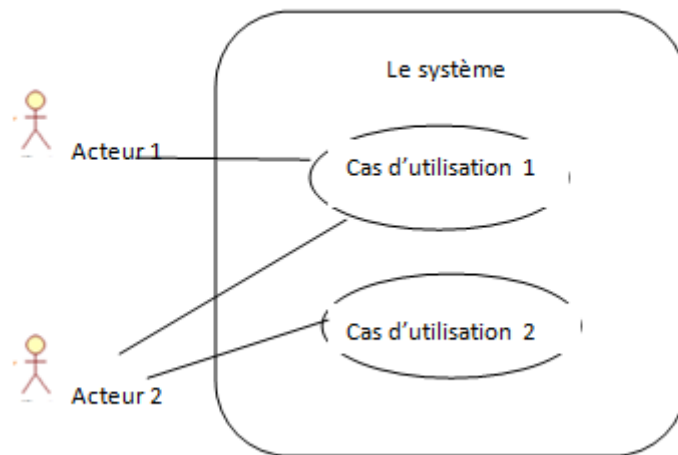


FIGURE 2 : EXEMPLE DE DIAGRAMME DE CAS D'UTILISATION

Cette relation peut aussi être vue comme un exemple de délégation, c'est à dire que le cas d'utilisation qui est inclus représente une responsabilité particulière dans le S.I. qui pourra être utilisée par les autres cas d'utilisation qui en auront besoin.

La relation de dépendance d'extension entre cas d'utilisation indique que le cas d'utilisation de base va éventuellement être étendu par le cas d'utilisation à la base de la relation. Ce cas d'utilisation peut correspondre à la description du traitement d'une exception.

L'acteur ici est le pharmacien, le diagramme des cas d'utilisation est montré par le tableau des cas d'utilisation (Figure 3)

Acteurs	Description
Pharmacien	Possède des droits sur la gestion de stock pharmaceutique.
	Possède des droits sur la gestion des médicaments.
	Possède des droits sur la gestion des sécurités
	Possède des droits sur la gestion des ventes.

FIGURE 3 : TABLEAU DES CAS D'UTILISATION

Le nombre des acteurs et des cas change selon les besoins de chaque projet. La représentation de cas d'utilisation en diagrammes est décrite par les figures ci- dessous :

## **2.3. Diagramme des cas d'utilisation**

### **2.3.1. Diagramme des cas d'utilisation « Authentification »**

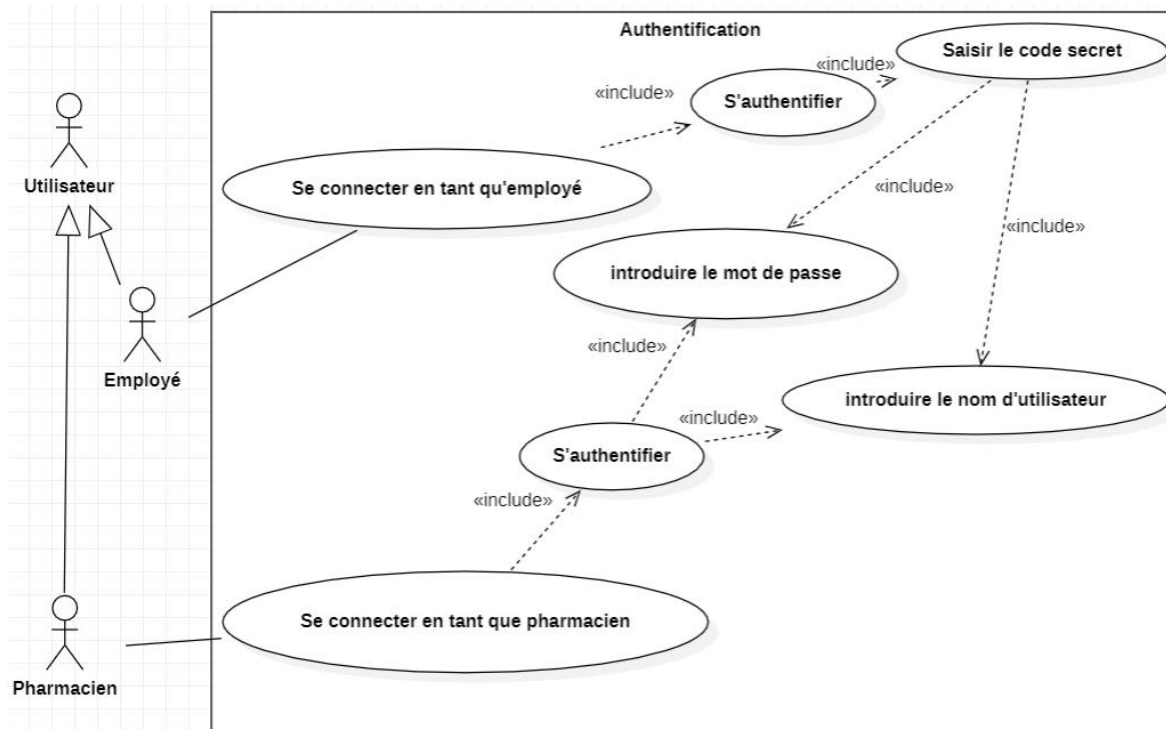


FIGURE 4 : DIAGRAMME DE CAS D'UTILISATION « AUTHENTIFICATION »

### 2.3.2. Diagramme de cas d'utilisation « Inscription »

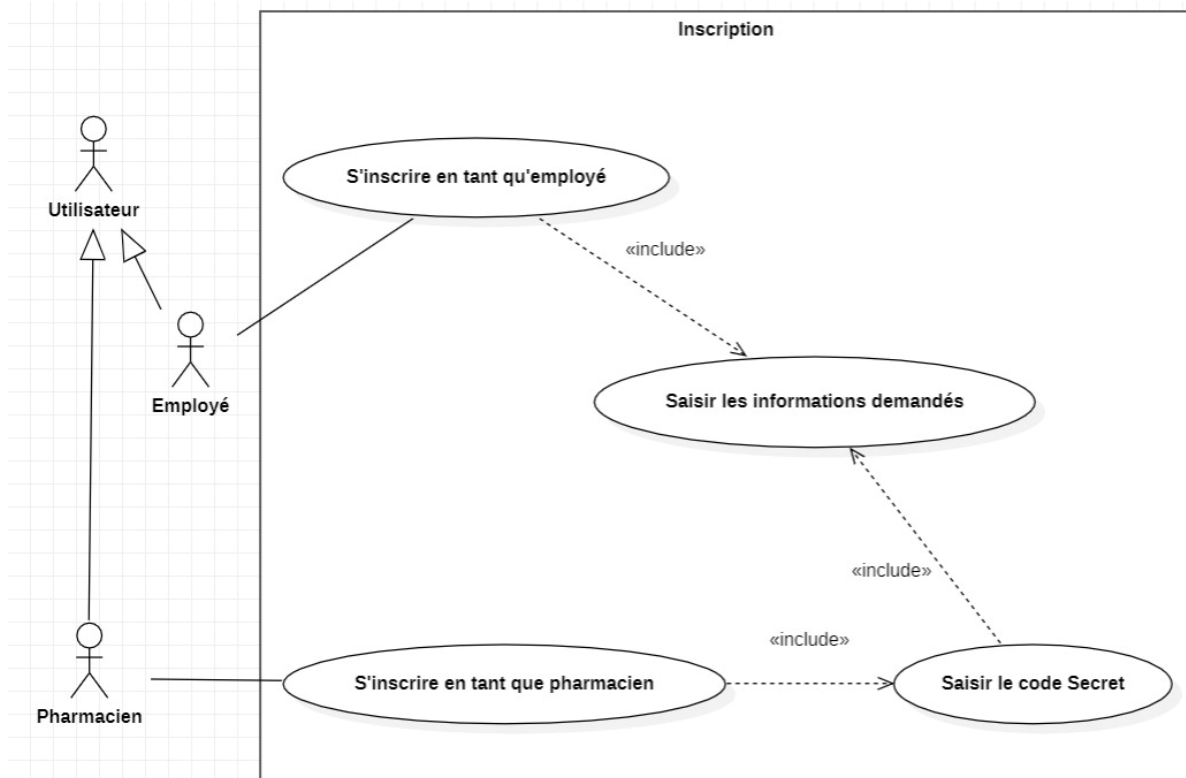


FIGURE 5 : DIAGRAMME DE CAS D'UTILISATION « INSCRIPTION »

## 2.4. Raffinement des cas d'utilisation

### 2.4.1. Gestion de stock

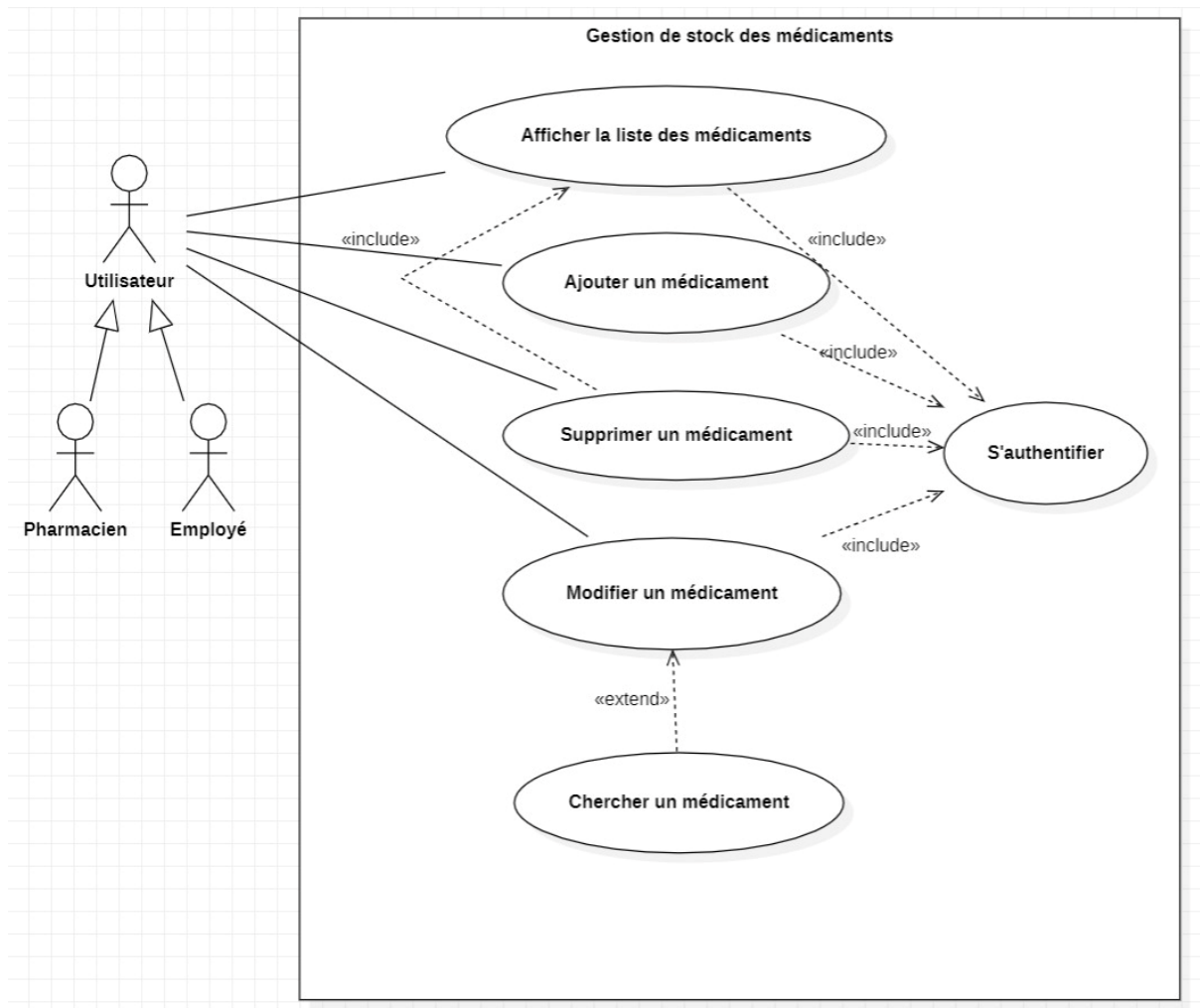


FIGURE 6 : DIAGRAMME DE CAS D'UTILISATION « GESTION DE STOCK »

## 2.4.2. Gestion des ventes

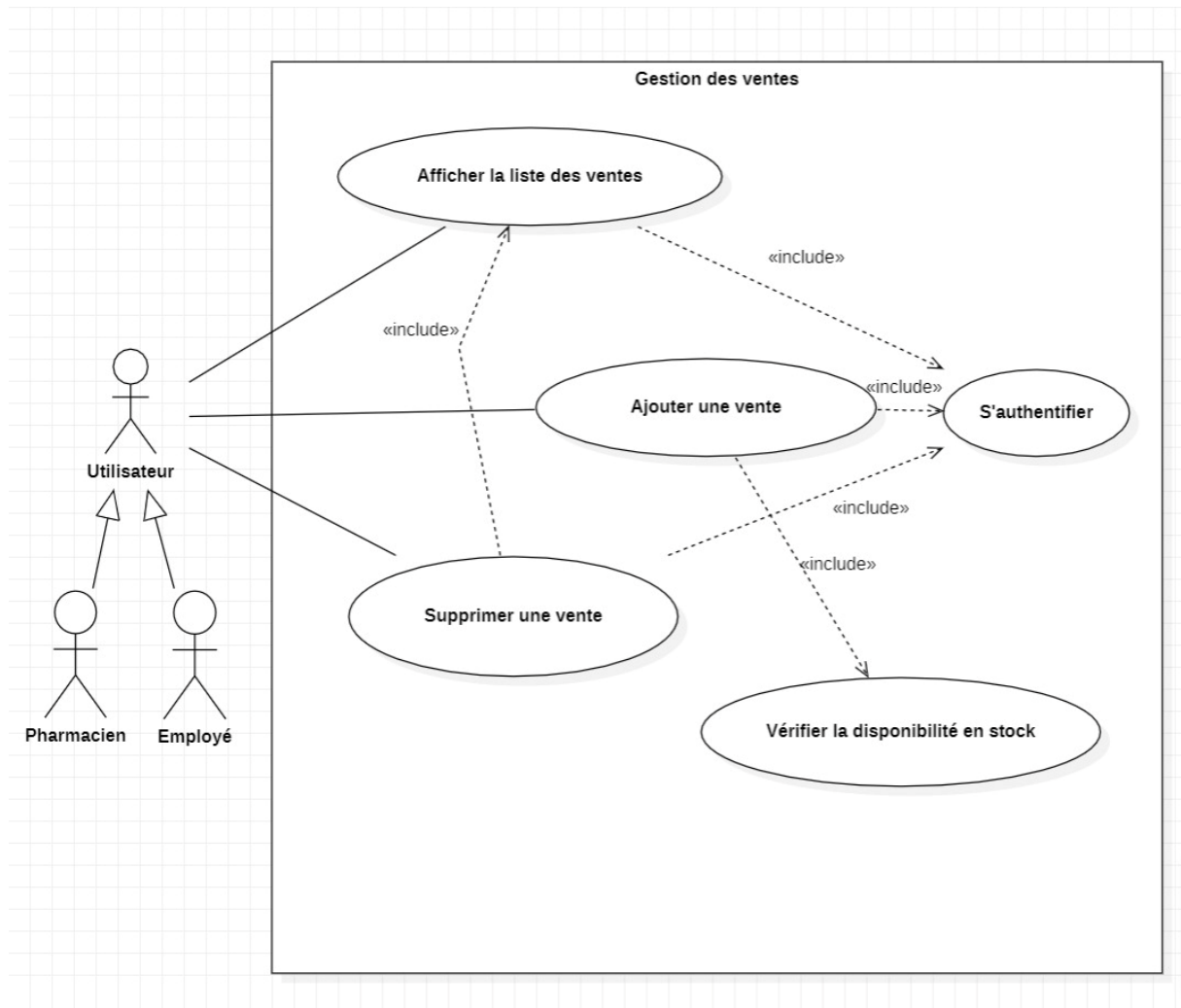


FIGURE 7 : DIAGRAMME DE CAS D'UTILISATION « GESTION DES VENTES »

### 2.4.3. Gestion des médicaments

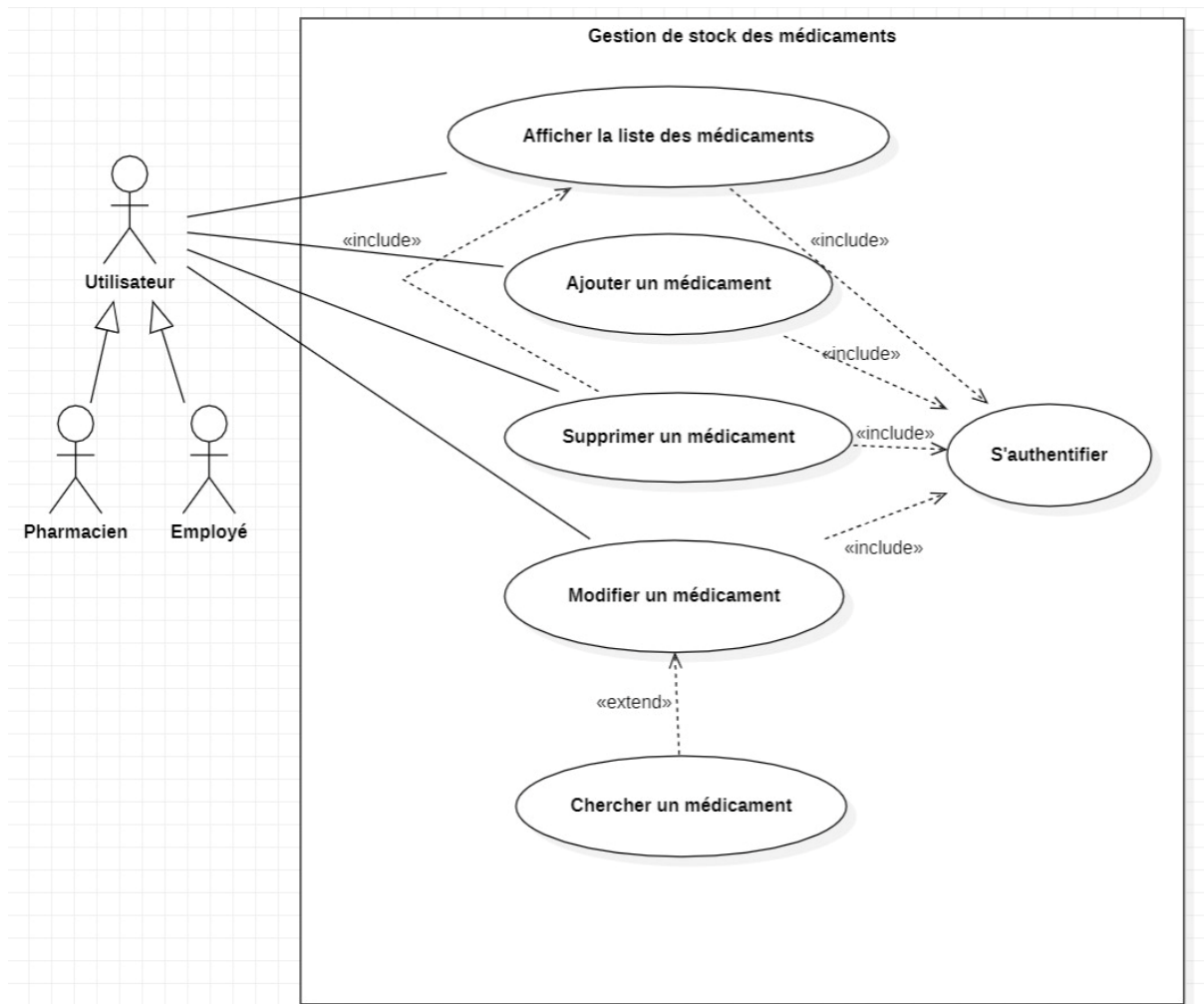


FIGURE 8 : DIAGRAMME DE CAS D'UTILISATION « GESTION DES MEDICAMENTS »

### 2.5. Diagramme de séquence

Le diagramme de séquence est une description graphique des opérations d'un système sous un angle chronologique. C'est une vue dynamique qui contient les symboles d'objets (instances de classe), d'acteurs et de messages qu'ils échangent. La dimension verticale est l'axe temporel : les messages y sont représentés par ordre chronologique. La dimension horizontale montre des objets et des acteurs qui échangent des informations, le diagramme de séquence permet de représenter les échanges entre les composants et les objets du système, dans le cadre d'exécution des cas d'utilisation, de point de vue temporel.

Dans ce qui suit, nous présentons quelques fonctions de l'utilisateur à travers des différents diagrammes de séquences :

### 2.5.1. Diagramme de séquence d'authentification « employé »

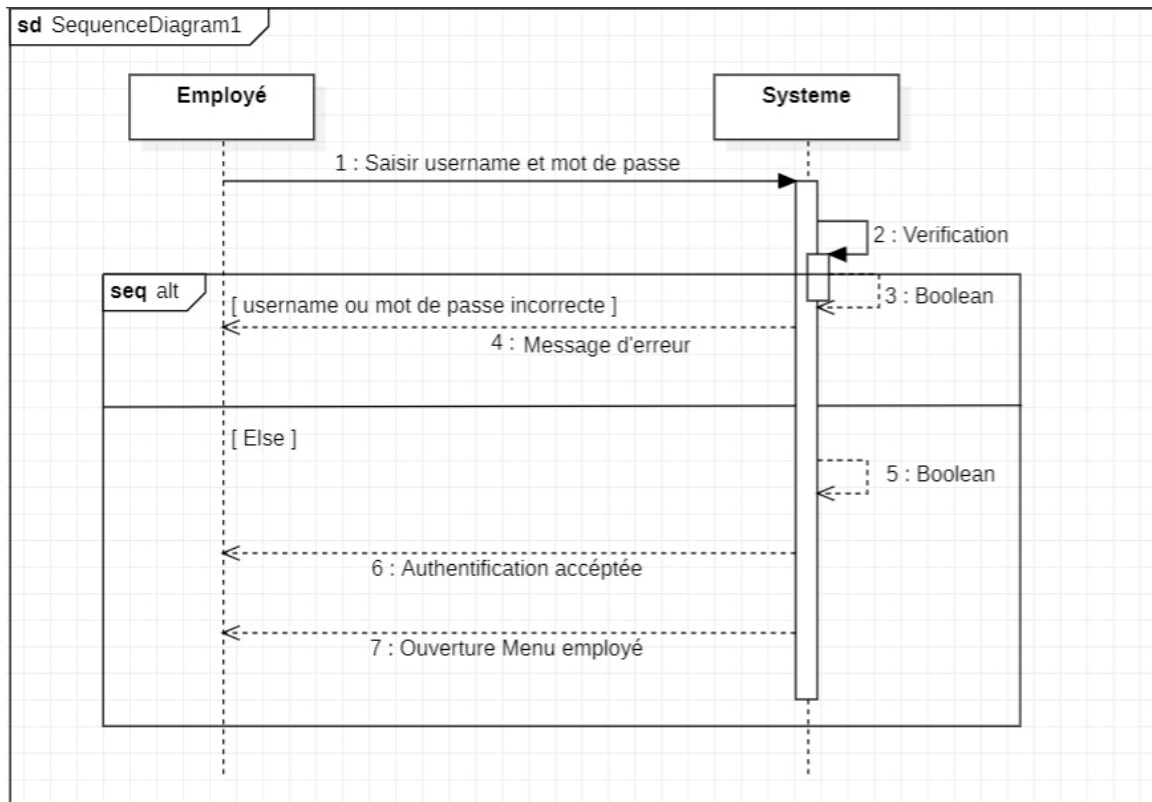


FIGURE 9 : DIAGRAMME DE SEQUENCE DE L'AUTHENTIFICATION « EMPLOYE »

**Description textuelle :** L'utilisateur qui veut y accéder tape son id et son mot de passe, le système vérifie la concordance des données et confirme l'authentification, l'affichage de l'interface principale se suit directement.

### 2.5.2. Diagramme de séquence d'authentification « pharmacien »



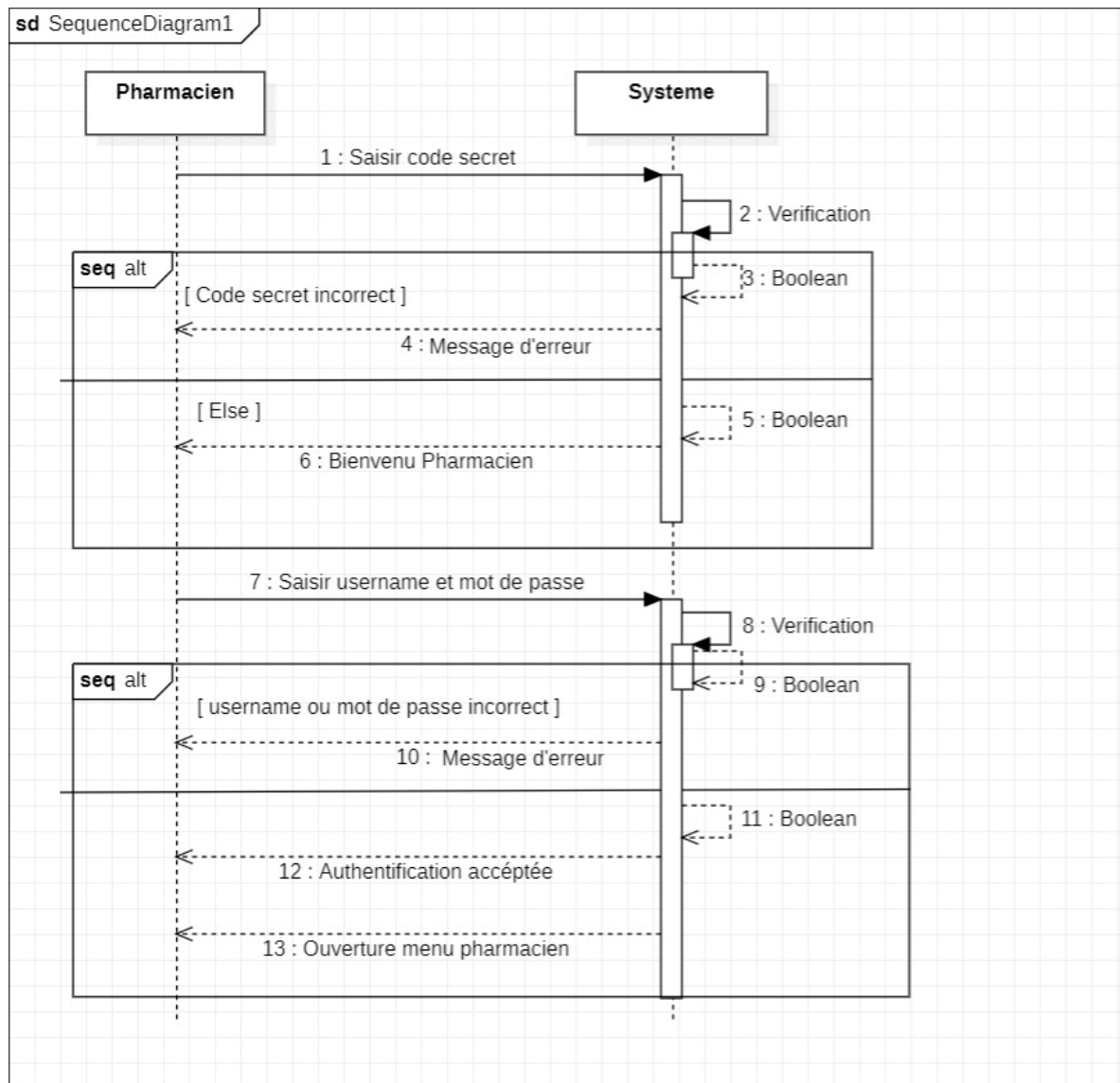


FIGURE 10 : DIAGRAMME DE SEQUENCE DE L'AUTHENTIFICATION « PHARMACIEN »

### 2.5.3. Diagramme de séquence pour ajout d'un nouveau produit

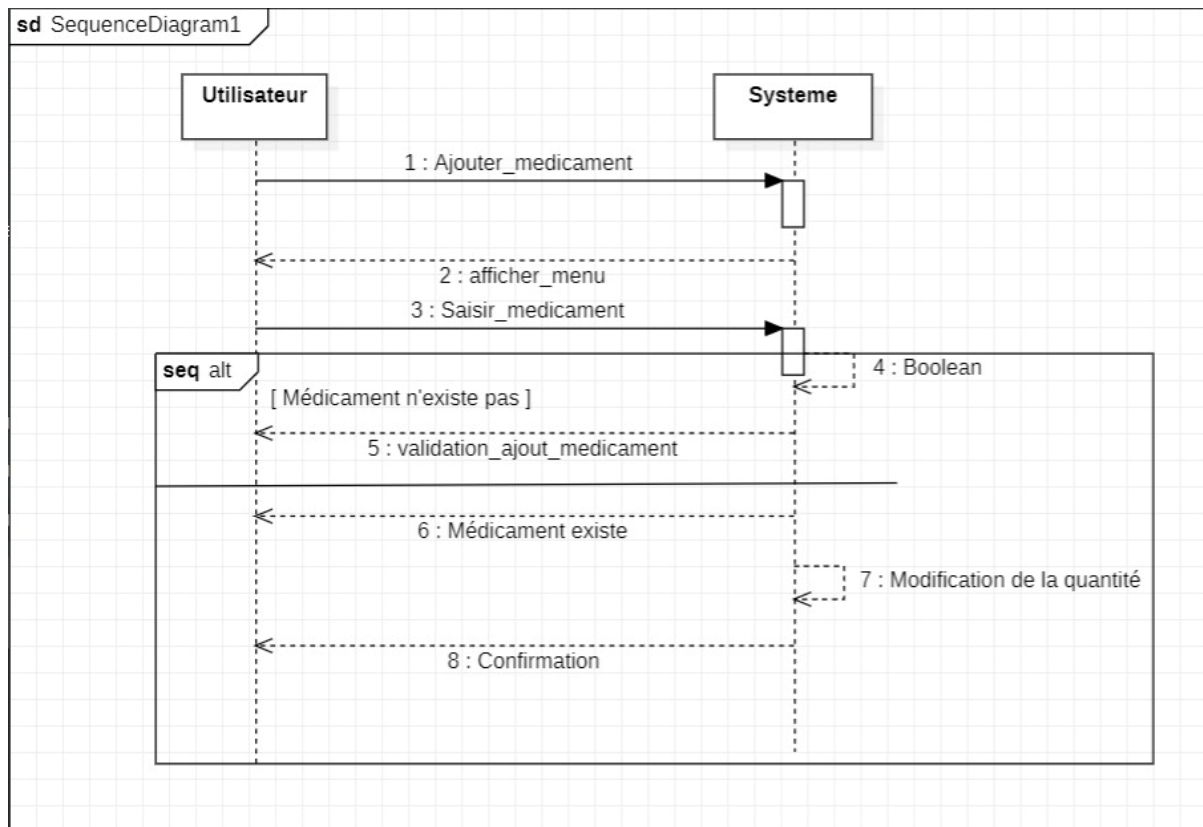


FIGURE 11 : DIAGRAMME DE SEQUENCE DE LA MISE A JOUR SU STOCK

**Description textuelle :** L'utilisateur mentionne les informations du nouveau médicament, le système valide, sinon l'utilisateur modifie seulement la quantité si le produit existe déjà dans le stock, le système confirme.

## 2.6. Construction du diagramme des classes

Le diagramme de classes est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que les différentes relations entre celles-ci. Ce diagramme fait partie de la partie statique d'UML car il fait abstraction des aspects temporels et dynamiques.

Une classe décrit les responsabilités, le comportement et le type d'un ensemble d'objets. Les éléments de cet ensemble sont les instances de la classe.

Une classe est un ensemble de fonctions et de données (attributs) qui sont liées ensemble par un champ sémantique. Les classes sont utilisées dans la programmation orientée objet. Elles permettent de modéliser un programme et ainsi de découper une tâche complexe en plusieurs petits travaux simples.

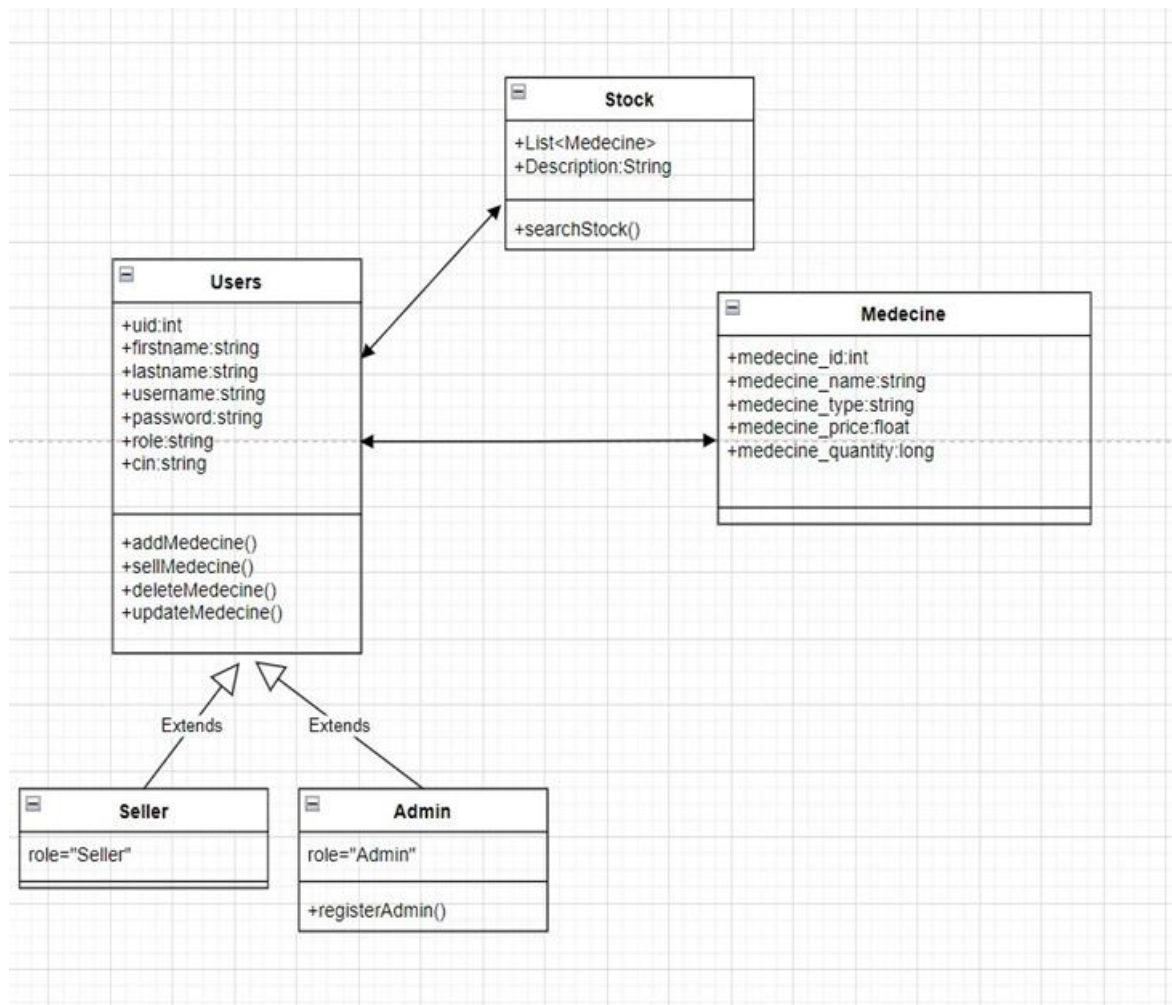


FIGURE 12 : DIAGRAMME DES CLASSES

### 2.6.1. Description des classes

Après une analyse de l'existant, nous avons dégagé les classes nécessaires pour une bonne gestion de la pharmacie.

N	Nom de classe
1	Médicament
2	Stock
3	Utilisateur
4	Admin/Vendeur

### 2.6.2. Description des associations

Numéro	Association	Classes Participantes	Description
01	Vendre médicament	Médicament, Vente	Un médicament peut être vendu plusieurs fois
02	Appartient	Stock, Médicament	Le stock contient plusieurs médicaments
03	Hérite	Admin/Seller, User	Admin/user hérite les propriétés de la classe user

## 2.7. Conclusion

Durant cette phase, nous avons achevé l'analyse de tous les cas d'utilisations via la description des différents diagrammes de séquences, de cas d'utilisation et l'exposition du diagramme de classe. Nous entamerons dans la prochaine phase, la construction du système en présentant l'environnement technique, et une description des différentes interfaces de l'application.

## CHAPITRE 3

---

# ETUDE TECHNIQUE ET IMPLEMENTATION

### **3. Etude technique et implémentation**

#### **3.1. Introduction**

Dans ce chapitre nous allons s'intéresser à la présentation de l'environnement matériel et logiciel utilisés pour la réalisation de l'application.

Il s'agit en plus de décrire les étapes de mise en œuvre de l'application ainsi que les différentes interfaces permettant l'interaction entre l'utilisateur et le système à développer et décrivant les différentes phases suivies pour la réalisation.

#### **3.2. Outils et langage utilisés**

##### **3.2.1. Environnement de réalisation**

Dans cette partie, nous allons présenter :

- *L'environnement matériel*
- *L'environnement logiciel*

##### **3.2.1.1. L'environnement matériel**

Pendant la phase de documentation, de spécification des besoins, de conception et de développement, nous avons utilisé un PC ayant les caractéristiques suivantes :

- Processeur Ryzen 7 5700, carte graphique RTX 3050.
- 16Gb MB de mémoire vive.
- Disque dur de capacité 500 Go.
- Système d'exploitation Microsoft Windows 11 Professionnel.
- Ecran 15 pouce.

### **3.2.1.2. L'environnement logiciel**

#### **3.2.1.2.1. IntelliJ 2022.3**



Pour la réalisation de notre application JAVA en utilisant l'environnement de développement IntelliJ. IntelliJ est un environnement de développement intégré (IDE) destiné au développement de logiciels informatiques reposant sur la technologie Java. Il est développé par JetBrains (anciennement « IntelliJ ») et disponible en deux versions, l'une communautaire, open source, sous licence Apache 2 et l'autre propriétaire, protégée par une licence commerciale. Tous deux supportent les langages de programmation Java, Kotlin, Groovy et Scala.

#### **3.2.1.2.2. Langage JAVA**



Java est un langage de programmation informatique orienté objet, la particularité et l'objectif central de java est que les logiciels écrit dans ce langage doivent être très facilement portables sur plusieurs systèmes d'exploitation. Java a donné naissance à un système d'exploitation (Java OS), a des environnements de développements (JDK), des machines virtuelles (MSJVM, JRE) applications multiplateformes (JVM), une

déclinaison pour les périphériques mobiles/embarqués (J2ME), une bibliothèque de conception d'interface graphique (AWT/Swing), des applications lourdes( Oracle SQL...) , des technologies web(applets, servlets) et une déclinaison pour l'entreprise (La J2EE) .Le point essentiel en java c'est la portabilité de ses applications , elles peuvent être exécuté sur n'importe quelle machine.

### **3.2.1.2.3.Base de données**

Une base de données (en anglais : database) est un outil permettant de stocker et de retrouver l'intégralité de données brutes ou d'informations en rapport avec un thème ou une activité ; celles-ci peuvent être de natures différentes et plus ou moins reliées entre elles.

Dans la très grande majorité des cas, ces informations sont très structurées, et la base est localisée dans un même lieu et sur un même support. Ce dernier est généralement informatisé.

La base de données est au centre des dispositifs informatiques de collecte, mise en forme, stockage, et utilisation d'informations. Le dispositif comporte un système de gestion de base de données (abr. SGBD) : un logiciel moteur qui manipule la base de données et dirige l'accès à son contenu. De tels dispositifs — souvent appelés base de données — comportent également des logiciels applicatifs, et un ensemble de règles relatives à l'accès et l'utilisation des informations.

### **3.2.1.2.4.Langage de manipulation de base de données**



Le langage de requête spatiale a été conçue comme une extension minimale à la partie interrogative de SQL et distingue des extensions précédemment conçus SQL par : la préservation des concepts SQL ; le traitement de haut niveau des objets spatiaux ; et



l'intégration des opérations spatiales et les relations. Il se compose de deux éléments, un langage de requête pour décrire les informations à récupérer et une langue de présentation pour spécifier comment afficher les résultats de la requête. Les utilisateurs peuvent demander des requêtes SQL standard pour récupérer des données non spatiales sur la base de contraintes non spatiales.

### 3.3. Conception de base de données

Notre base de données est composée de 5 tables, et voici les listes des tables de la base de données :

Tables_in_pharmacy
admins
customer
sellers
stock
users

FIGURE 13 : LISTE DES TABLES DE LA BASE DE DONNEES

#### TABLE ADMIN

Field	Type	Null	Key	Default	Extra
aid	int	NO	PRI	NULL	auto_increment
firstname	varchar(45)	NO		NULL	
lastname	varchar(45)	NO		NULL	
username	varchar(45)	NO		NULL	
password	varchar(45)	NO		NULL	
cin	varchar(45)	NO		NULL	

aid	firstname	lastname	username	password	cin
12	Omar	Mazali	omar	12345	HH125869

FIGURE 14 : DESCRIPTION ET BASE DE DONNEES DE LA TABLE ADMIN

## TABLE CUSTOMERS

Field	Type	Null	Key	Default	Extra
cid	int	NO		NULL	
name	varchar(45)	NO		NULL	
type	varchar(45)	NO		NULL	
price	float	NO		NULL	
quantity	mediumtext	NO		NULL	
total	float	NO		NULL	
date	date	YES		NULL	

cid	name	type	price	quantity	total	date
3	Ibuprofène 400 mg	Antalgique	30.01	20	600.2	2022-12-18
4	Amoxicilline	Antibiotique	55	5	275	2022-12-18
1	Doliprane	Anti-inflammatoire	16	20	320	2022-12-18
2	Rinomicine	Anrti-rhume	27	10	270	2022-12-18
4	Amoxicilline	Antibiotique	55	5	275	2022-12-18
1	Doliprane	Anti-inflammatoire	16	20	320	2022-12-18
2	Rinomicine	Anrti-rhume	27	10	270	2022-12-18
1	Doliprane	Anti-inflammatoire	16	5	80	2023-01-06
5	Lyvotirox 25mg	Médicament de la thyroïde	6	875	5250	2023-01-09

FIGURE 15 : DESCRIPTION ET BASE DE DONNEES DE LA TABLE CUSTOMER

## TABLE SELLERS

Field	Type	Null	Key	Default	Extra
sid	int	NO	PRI	NULL	auto_increment
firstname	varchar(45)	NO		NULL	
lastname	varchar(45)	NO		NULL	
username	varchar(45)	NO	UNI	NULL	
password	varchar(45)	NO		NULL	
cin	varchar(45)	NO		NULL	

sid	firstname	lastname	username	password	cin
7	Ziad	Benhebibi	ziad	123	J457832

FIGURE 16 : DESCRIPTION ET BASE DE DONNEES DE LA TABLE SELLERS

## TABLE STOCK

Field	Type	Null	Key	Default	Extra
mid	int	NO	PRI	NULL	
name	varchar(100)	YES		NULL	
type	varchar(100)	YES		NULL	
price	float	YES		NULL	
quantity	int	YES		NULL	

mid	name	type	price	quantity
1	Doliprane	Anti-inflammatoire	16	955
2	Rinomicine	Anrti-rhume	27	480
3	Ibuprofène 400 mg	Antalgique	30.01	200
4	Amoxicilline	Antibiotique	55	140

FIGURE 17 : DESCRIPTION ET BASE DE DONNEES DE LA TABLE STOCK

## TABLE USERS

Field	Type	Null	Key	Default	Extra
uid	int	NO	PRI	NULL	auto_increment
firstname	varchar(45)	NO		NULL	
lastname	varchar(45)	NO		NULL	
username	varchar(45)	NO	UNI	NULL	
password	varchar(45)	NO		NULL	
role	varchar(45)	NO		NULL	
cin	varchar(45)	NO		NULL	

uid	firstname	lastname	username	password	role	cin
16	Omar	Mazali	omar	12345	Admin	HH125869
17	Ziad	Benhebibi	ziad	123	Seller	J457832

FIGURE 18 : DESCRIPTION ET BASE DE DONNEES DE LA TABLE USERS

### 3.4. Réalisation du projet

Notre application de gestion de pharmacie permet de gérer les produits, la vente, etc.... et d'offrir à l'utilisateur quelques accessoires à savoir la date actuelle.

La multitude des tâches que notre application est capable de faire engendrer un grand nombre de fenêtres.

Nous allons essayer de sélectionner quelques fenêtres qui nous paraissent importantes pour les intégrer dans le présent mémoire.

#### 3.4.1. Interface d'authentification « employé »

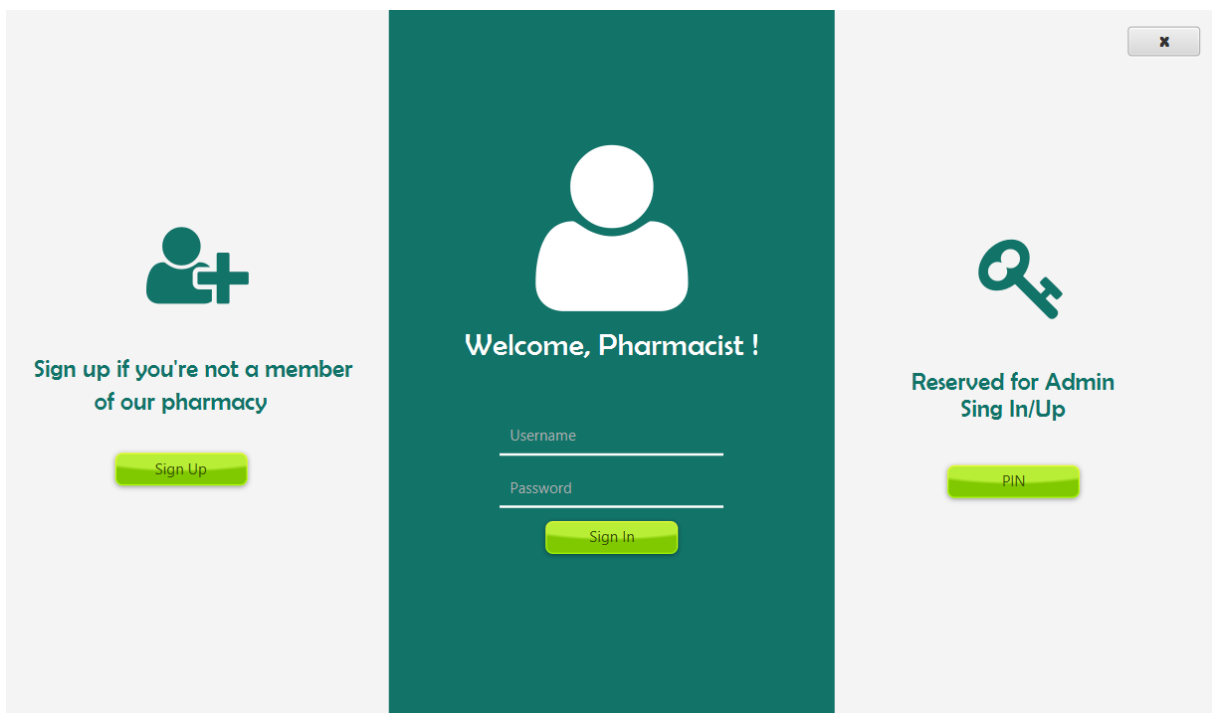


FIGURE 19 : INTERFACE D'AUTHENTIFICATION D'UN EMPLOYE

Cette première capture présente l'interface d'authentification dans laquelle on doit choisir le nom d'utilisateur et entrer le mot de passe pour commencer à utiliser notre application. A travers cette fenêtre l'utilisateur s'authentifie pour utiliser l'application.

Cette étape met en valeur l'aspect sécurité.

Si ces données sont fausses, un message d'erreur s'affiche :

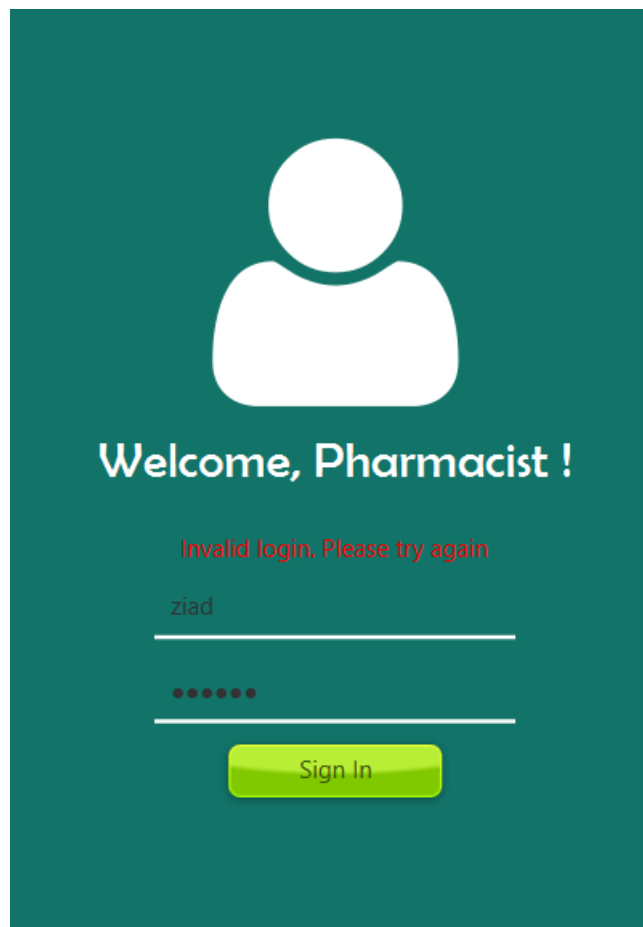


FIGURE 20 : MESSAGE D'ERREUR D'AUTHENTIFICATION « EMPLOYE »

### **3.4.2. Interface d'inscription « employé »**

Cette interface représente la page d'inscription de notre application offrant à l'utilisateur des champs à remplir. Nous avons veillé à ce que les interfaces soient assez conviviales : le choix des couleurs, l'ergonomie et la clarté du contenu permettent de faciliter son exploitation par l'utilisateur.

## Personal Informations

First name :

Last Name :

Last Name

Username :

Username

CIN :

CIN

Password :

Password

Confirm Password :

Confirm Password

Register

Welcome sir/ madam to our pharmacy, please enter your personal details

Back

FIGURE 21 : INTERFACE D'INSCRIPTION « EMPLOYE »

Please enter your pesonal informations

First name :

First Name

Last Name :

Last Name

Username :

Username

CIN :

CIN

Password :

Password

Confirm Password :

Confirm Pass

First name :

Abir

Last Name :

Lachhab

Username :

Username \*

CIN :

CIN \*

Password :

Password \*

Confirm Password :

Confirm Password

Register

FIGURE 22 : MESSAGE D'ERREUR D'INSCRIPTION « EMPLOYE »

### 3.4.3. Interface d'accueil « employé »

Cette partie va lui permettre d'accéder à plusieurs choix possibles mais qui sont restreints par rapport au pharmacien qui lui a plusieurs options de gestions de sa pharmacie par rapport à l'employé.

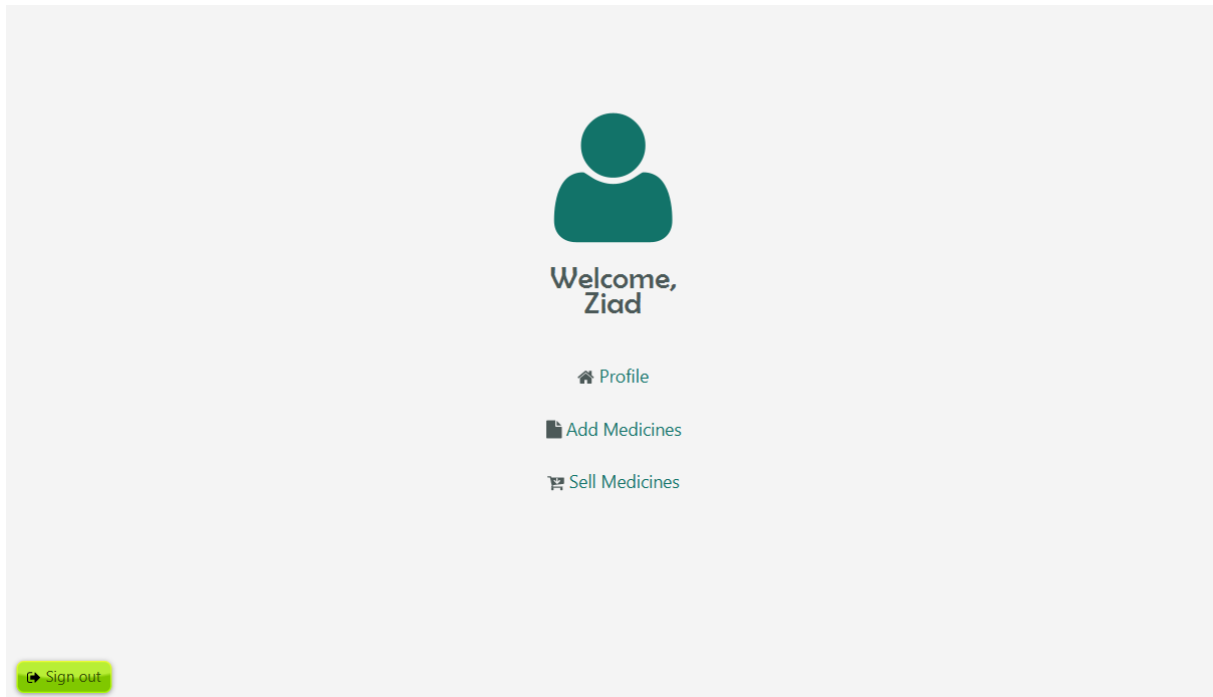


FIGURE 23 : INTERFACE D'ACCUEIL « EMPLOYE »

### 3.4.4. Interface d'authentification « pharmacien »

Avant même d'y accéder le pharmacien doit obligatoirement taper un PIN donnée par son collègue pour pouvoir par la suite soit se connecter ou s'inscrire en tant que pharmacien.

Si on accède par la suite en tapant bien sûr le PIN correct, on change de scène et le pharmacien est passé à la page de connexion

**PS : Traitement d'erreur est mise en place pour les deux interfaces**

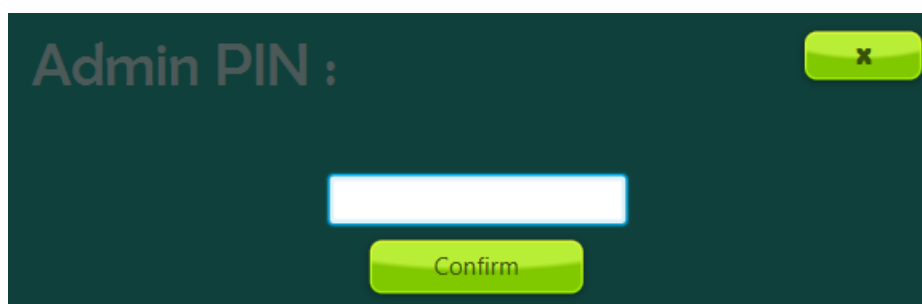


FIGURE 24 : INTERFACE PIN

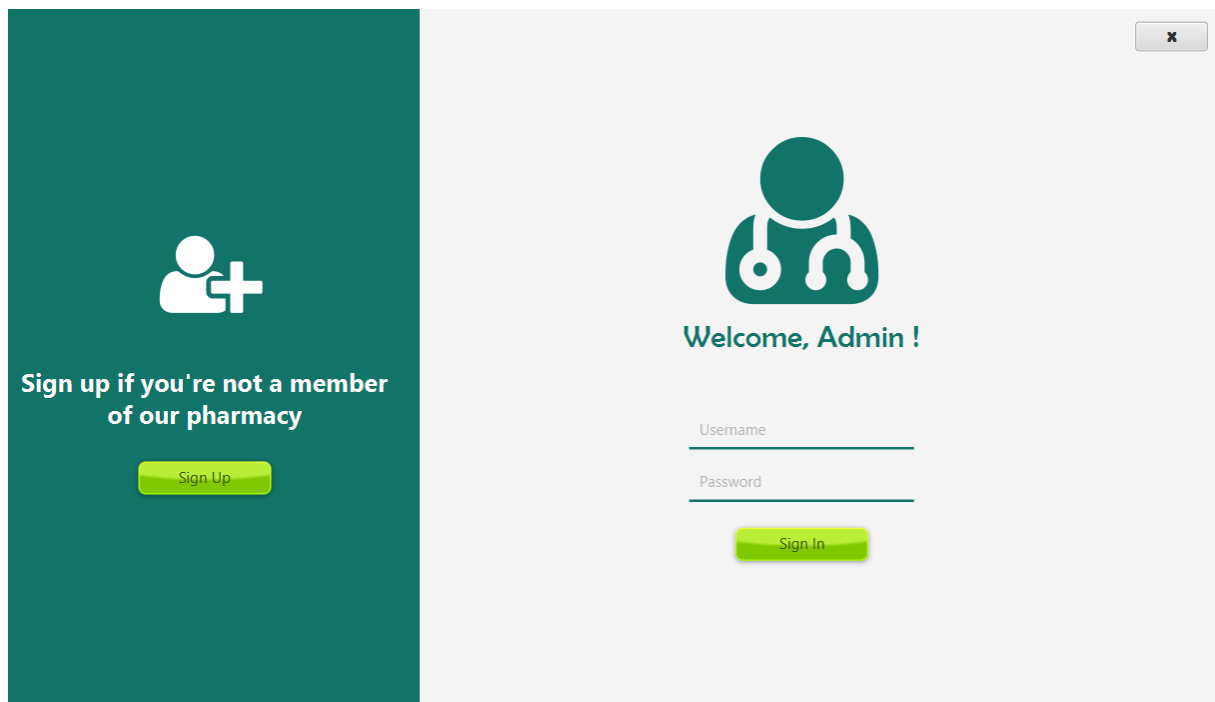


FIGURE 25 : INTERFACE D'ACCUEIL « EMPLOYE »

### 3.4.5. Interface d'inscription « pharmacien »

Cette interface est identique à celle de l'employé, la seule différence se montre dans la base de données. Ce dernier est insérer à la fois dans la table ADMIN vu qu'il est un pharmacien et dans la table USERS en tant qu'ADMIN dans son rôle par contre un simple employé aura dans son rôle SELLER et il sera insérer dans ses tables correspondantes aussi.

### 3.4.6. Interface d'accueil « pharmacien »

Vous remarquez la large différence entre deux personnes (pharmacien et employé). Un pharmacien à un accès aux stats de sa pharmacie, sa gestion, son historique, gestion des employés mais aussi l'achat et la vente.









### **3.5. Conclusion**

Dans ce chapitre, nous avons présentés notre environnement de travail matériel et logiciel, les différentes techniques utilisées pour implémenter le système ainsi que les principales interfaces de notre application avec leurs descriptions.

## CHAPITRE 4

---

# CONCLUSION GENERALE ET PERSPECTIVE

#### **4. Conclusion générale et perspective**

Rappelons que l'objectif de ce travail était d'informatiser l'activité de gestion du système d'informations des pharmacies hospitalières et des dispensaires publiques. Pour cela, on a réalisé une application interactive permettant de gérer les différents traitements de cette activité et de satisfaire les besoins des différents utilisateurs impliqués dans ce processus de gestion.

Le travail est débuté par la compréhension du contexte de notre projet. Ensuite, on a réalisé une étude de l'existant concernant les applications de gestion des activités de la pharmacie, ce qui nous a permis de fixer les anomalies à éviter et les objectifs à réaliser pour avoir un système satisfaisant. Puis, on a passé à la l'étude conceptuelle de notre application selon une approche orientée objet tout en se basant sur le langage UML. Par la suite, on a effectué le codage et l'implémentation de l'application. Enfin on a effectué les tests nécessaires pour valider l'application.

Ce projet a été très bénéfique pour nous car il nous a permis de renforcer et enrichir nos connaissances théoriques dans le domaine de la conception, et de mettre en application nos connaissances acquises le long de nos études. Il nous a encore donné l'occasion de maîtriser le langage de programmation Java, le système de gestion de la base de données MySQL manager et de nous familiariser avec la conduite des projets informatiques.

En plus, ce projet était une bonne occasion pour réaliser un travail très concret, avec des objectifs clairs et bien définis et de se familiariser avec l'environnement du travail et de la vie professionnelle.

En perspective, notre application peut être améliorée en ajoutant d'autres fonctionnalités comme (Gestion des fournisseurs, des commandes, gestion des ordonnances ...).

## RESUME

Depuis le début de construction des pharmacies, les pharmaciens s'orientaient à un objectif identique, celui-ci se résumait à comment arriver à gérer la pharmacie à l'aide de moyens facile et rapides. Mais malgré tous les efforts faites par les pharmaciens, ils restent des difficultés. Dans ce but qu'on a développé une application qui permet au pharmacien de gérer le stock des médicaments, les utilisateurs ainsi que le processus de vente.

Afin d'implémenter les différentes parties décrites ci-dessus, nous avons opté pour :

- L'environnement de programmation utilisé est IntelliJ 2022.3.
- Ce travail a été réalisé en utilisant le processus de développement "Processus Unifier" (UP) et le langage de modélisation Unified Modeling Language (UML) afin de schématiser la solution.
- Nous avons choisi de programmer l'application avec le langage Java.
- MYSQL comme Système de Gestion de Base de données (SGBD).