

République algérienne démocratique et populaire
Ministère de l'enseignement supérieur et
De la recherche scientifique



Université 20 Aout 1955 SKIKDA
Faculté des sciences
Département d'informatique



Mémoire de fin d'études en vue de l'obtention du Diplôme
De Master en Informatique – Spécialité : Réseaux et Systèmes Distribués (RSD)

Thème

Création d'une application mobile pour l'échange
des médicaments entre pharmacies
hospitalières

Réalisé par :

- Aicha Zennir
- Rayane Larabi

Encadré par :

M. Touil Ghassen

Promotion :2022/2023

Remerciements

Nous tenons à remercier tout d'abord ALLAH, le tout puissant de nous a donné le courage et la volonté de mener à terme ce projet.

Nous tenons à remercier notre encadreur, M. Touil Ghassen pour sa disponibilité, son aide, ses conseils précieux et ses critiques constructives, ses explications et suggestions pertinentes.

Nous adressons également nos remerciements à tous les professeurs qui nous ont enseignées durant ces cinq dernières années, pour la formation qu'ils nous ont données, particulièrement.

Que tous ceux qui, de près ou de loin ont contribué, par leurs conseils, leurs encouragements ou leur amitié à l'aboutissement de ce travail, trouvent ici l'expression de notre profonde reconnaissance.

Pour leur encouragement, leur soutien moral et la patience qu'ils nous ont manifestée durant toute l'année, nous remercions fortement tous les membres de nos familles.

Nous remercions également Ahcen Zennir pour tous les conseils et l'assistance qu'il nous a fournies.

Dédicace

Je tien à dédier ce modeste travail à tous ceux qui m'ont encouragée durant toute la période de réalisation de ce travail.

En particulier :

- *A mes chers parents qui se sacrifient pour me voir réussir.*
- *A ma chères sœur « Fatima ».*
- *A mes chères frères « Mohamed », « Ahmed » et « AHCEN ».*
- *A mes amies et toute ma famille, et à ma binôme « Rayane ».*

Aicha

Dédicace

Je tien à dédier ce modeste travail à tous ceux qui m'ont encouragée durant tout la période de réalisation de ce travail.

En particulier :

- *A mes chers parents qui se sacrifient pour me voir réussir « Abdelhak » et « Mahbouba ».*
- *A mes chères sœurs « Sara », « Kaltoum », « Asma » et « Wadjdene »*
- *A toutes mes amies et à toute ma famille, à mon binôme « Aicha ».*

Rayane

Résumé :

Au cours de ce mémoire, nous avons conçu et réalisé une application mobile pour la gestion d'échange de médicaments entre pharmacies hospitalières.

Cette application offre la possibilité pour chaque établissement hospitalier (EPH) pour faire la commande par choisirai médicament pour faire l'échange. Ce travail a été réalisé en utilisant le modèle client-serveur pour application mobile.

Et on utilise le réseau de Pétri et le langage de modélisation UML pour modéliser cette application. Nous avons choisi de programmer l'application avec langage « WLangage » et HFSQL pour la gestion de base de données.

Mots clés : Application mobile, Wlangage, HFSQL, WINDEV, UML.

Summary:

At the heart of this thesis, we have designed and produced a mobile application for the management of drug exchange between hospital pharmacies.

That this application offers the drugs of each establishment hospital pharmacy (EPH) to make the order by will choose drug to make the exchange.

This work was carried out using the client-server model for mobile application. And we use kneaded network and UML modeling language to model this application. We have chosen to program the application with “WLANGAGE” language and HFSQL for database management.

Keywords: Mobile application, WLANGUAGE, HFSQL, WINDEV, UML.

Table de matières

| | |
|---|----|
| Introduction générale | 1 |
| Chapitre 1 : Technologies mobiles | |
| 1.1.Introduction | 4 |
| 1.2.Les application mobiles | 4 |
| 1.2.1.Définition | 4 |
| 1.2.2.Les caractéristiques des application mobiles | 4 |
| 1.2.3.Les types application mobile | 5 |
| 1.2.4.Les avantages et les inconvénient d'application | 7 |
| 1.2.4.1.Les avantages | 7 |
| 1.2.4.2.Les inconvénients | 7 |
| 1.3.Les réseaux informatiques | 8 |
| 1.3.1.Un réseau | 8 |
| 1.3.2.Les catégories | 8 |
| 1.4.Application client-serveur | 8 |
| 1.4.1.Caractéristiques d'un serveur et d'un client | 9 |
| 1.4.2.Le fonctionnement d'un système client-serveur | 9 |
| 1.5.Conclusion | 9 |
| Chapitre 2 : Présentation des concepts médicaux | |
| 2.1.Introduction | 11 |
| 2.2.Les pharmacies hospitaliers | 11 |

| | |
|--|----|
| 2.2.1.Les fonction essentielles de pharmacie hospitalier | 11 |
| 2.2.2.Role des personales sur pharmacie hospitalier | 12 |
| 2.2.3.Organigramme générale de pharmacie hospitalier | 14 |
| 2.2.4.Problématiques et objectifs | 15 |
| 2.2.4.1.Problématique | 15 |
| 2.2.4.2.Objectifs | 15 |
| 2.2.5.Besoins fonctionnel | 16 |
| 2.2.6.Besoins non-fonctionnel | 17 |
| 2.3.Conclusion | 18 |
| Chapitre 3 : Analyse et conception | |
| 3.1.Introduction | 20 |
| 3.2.Modélisation | 20 |
| 3.2.1.Unified modeling language | 20 |
| 3.2.1.1.Diagrammes UML utilisés | 21 |
| 3.3.Les acteurs du système | 21 |
| 3.3.1.Définition | 21 |
| 3.3.2.Identification des acteurs | 21 |
| 3.3.3.Identification de cas d'utilisation | 23 |
| 3.4.Diagramme de cas d'utilisation | 25 |
| 3.4.1.Définition | 25 |
| 3.4.2.Diagramme de cas d'utilisation du pharmacien hospitalier et chef service | 25 |
| 3.4.3.Diagramme de cas d'utilisation de l'assistant administratif | 26 |
| 3.5.Diagramme de séquence | 26 |
| 3.5.1.Description textuelle de diagramme de séquence « s'inscrire » | 26 |
| 3.5.1.1.Désignation de diagramme de séquence « s'inscrire » | 28 |

| | |
|--|----|
| 3.5.2. Description textuelle de diagramme de séquence « authentification » | 29 |
| 3.5.2.1. Désignation de diagramme de séquence « authentification » | 30 |
| 3.5.3. Description textuelle de diagramme de séquence « gérer les utilisateurs » | 30 |
| 3.5.3.1. Désignation de diagramme de séquence « gérer les utilisateurs » | 33 |
| 3.5.4. Description textuelle de diagramme de séquence « échanger médicaments » | 34 |
| 3.5.4.1. Désignation de diagramme de séquence « échange médicaments » | 35 |
| 3.5.5. Description textuelle de diagramme de séquence « gérer les échanges » | 36 |
| 3.5.5.1. Désignation de diagramme de séquence « gérer les échanges » | 37 |
| 3.6. Diagramme de classe | 38 |
| 3.7. Conclusion | 39 |
| Chapitre 4 : Réalisation | |
| 4.1. Introduction | 41 |
| 4.2. L'environnement de développement | 41 |
| 4.2.1. L'environnement matériel | 41 |
| 4.2.2. L'environnement logiciel | 41 |
| 4.3. Définition des logiciels et langages utilisés | 41 |
| 4.4. Présentation quelques interfaces graphiques de l'application | 45 |

| | |
|-----------------------------|----|
| 4.4.1.Configuration serveur | 45 |
| 4.4.2.Inscription | 46 |
| 4.4.3.Authentification | 47 |
| 4.4.4.Page d'accueil | 48 |
| 4.4.5.Page médicaments | 49 |
| 4.4.6.Page recherche | 50 |
| 4.4.7.Page de stock | 51 |
| 4.4.8.Page des commandes | 52 |
| 4.4.9.Page de liste | 53 |
| 4.4.10.Page factures | 54 |
| 4.5.Conclusion | 54 |
| Conclusion générale | 56 |

Listes des figures

| N figure | Figure | page |
|----------|--|------|
| 1.1 | Les types d'application mobile | 5 |
| 2.1 | Organigramme générale de pharmacie hospitalier | 14 |
| 3.1 | Diagramme de cas d'utilisation de pharmacien hospitalier et chef service | 25 |
| 3.2 | Diagramme de cas d'utilisation de l'assistant administratif | 26 |
| 3.3 | Diagramme de séquence « s'inscrire sur l'application » | 28 |
| 3.4 | Diagramme de séquence « authentification » | 30 |
| 3.5 | Diagramme de séquence « gérer les utilisateurs » | 33 |
| 3.6 | Diagramme de séquence « échanger médicaments » | 35 |
| 3.7 | Diagramme de séquence « gérer les échanges » | 37 |
| 3.8 | Diagramme de classe | 39 |
| 4.1 | Page configuration serveur | 45 |
| 4.2 | Page inscription | 46 |
| 4.3 | Page authentification | 47 |
| 4.4 | Page d'accueil | 48 |
| 4.5 | Page médicaments | 49 |
| 4.6 | Page de recherche | 50 |
| 4.7 | Page de stock | 51 |
| 4.8 | Page des commandes | 52 |

| | | |
|------|------------------|----|
| 4.9 | Page de liste | 53 |
| 4.10 | Page de factures | 54 |

Liste des tableaux

| N Tableau | Tableau | page |
|-----------|--|------|
| 3.1 | Descriptif des diagrammes UML | 21 |
| 3.2 | Identification textuelles des cas d'utilisation | 23 |
| 3.3 | Description textuelle de digramme de séquence « inscription » | 27 |
| 3.4 | Description textuelle de digramme de séquence « authentification » | 29 |
| 3.5 | Description textuelle de digramme de séquence « gérer les utilisateurs » | 30 |
| 3.6 | Description textuelle de digramme de séquence « échanger médicaments » | 34 |
| 3.7 | Description textuelle de digramme de séquence « gérer les échanges » | 36 |

L'ordinateur est le moyen le plus sûr de traitement et de stockage de données. Cette invention a permis d'informatiser les systèmes de gestion de données d'entreprise, ce qui est aujourd'hui un élément clé de leur développement.

Le domaine médical est une branche essentielle des sciences de la santé qui englobe les soins destinés à la prévention, au diagnostic, au traitement et à la gestion des maladies et des troubles affectant le corps humain. C'est un domaine basé sur le circuit de médicament qui un processus complexe avec de nombreux interlocuteurs est associé en 2 processus interconnectés : circuit clinique avec la prise en charge médicament du patient hospitalisé et le circuit d'approvisionnement qui concerne le médicament en tant que produit de la santé.

Dans le domaine de la santé, la gestion efficace des pharmacies hospitalières est d'une importance capitale pour assurer la fourniture rapide et sûre des médicaments et dispositifs médicaux nécessaires aux patients. Cependant, la coordination entre les différentes pharmacies hospitalières présente souvent des défis, notamment en ce qui concerne la communication, l'inventaire des médicaments et la coordination des commandes. En effet, une des préoccupations majeures dans ce circuit est la disponibilité des médicaments et dispositifs médicaux pour les malades.

Dans ce mémoire, nous nous intéressons à la conception et au développement d'une application mobile visant à créer un réseau de pharmacies hospitalières.

L'objectif principal de cette application est de faciliter la collaboration, d'améliorer la communication et de simplifier la gestion des médicaments au sein du réseau.

Le présent rapport de fin d'études est organisé comme suit :

- ❖ Tout d'abord, dans l'introduction générale on donne un aperçu général sur la problématique et les objectifs visés,
- ❖ Suivie d'un premier chapitre qui détaille les « **Technologies Mobiles** ».
- ❖ Ensuite nous retrouvons le deuxième chapitre : « **Présentation de concepts médicaux** » où la présentation de la pharmacie hospitalière nous permettra de définir notre problématique et fixer nos objectifs, tout en illustrant les étapes à suivre pour concevoir et mettre en œuvre la solution proposée.
- ❖ Après cela, le troisième chapitre intitulé: « **Analyse et conception** » pour notre application mobile.

- ❖ Dans le quatrième chapitre : « **Réalisation** » nous définirons tous les outils qui nous ont permis de concevoir notre application mobile, quelques interfaces seront présentées.
- ❖ Nous terminons par la "conclusion générale" et nous proposons quelques perspectives pour perfectionner le présent travail.

Chapitre 1 :

Technologies

Mobiles

1.1. Introduction :

Dans Ce chapitre nous allons présenter, les applications mobiles ainsi que ses types, et puis nous allons présenter les réseaux informatiques et les applications client/server.

1.2. Les applications mobiles :

1.2.1. Définition:

Une application mobile est un type de logiciel ou programme conçu pour s'exécuter sur un appareil mobile, tel qu'un smartphone ou une tablette. Les applications mobiles servent souvent à fournir aux utilisateurs des services similaires à ceux du PC [3].

1.2.2. Les caractéristiques des applications mobiles :

Les contraintes techniques qu'il est nécessaire de prendre en compte lors de la conception d'une application mobile :

- Tailles d'écrans variables, pouvant dans certain cas être assez réduites.
- Possibilité limitée de saisie de données.
- Puissance du processeur, pouvant être limitée sur les premiers Smartphones.
- Tailles de mémoires pouvant varier.
- Autonomie du Smartphone.
- Débits variables de la bande passante Internet.

Outre ces aspects techniques, il est également important de prendre en compte l'ergonomie de L'application mobile à réaliser, et c'est un point crucial à ne pas négliger. Une application mobile doit respecter certaines règles :

Utiliser des images petites et légères.

- Utiliser des éléments facilement accessibles.
- Économiser la consommation d'énergie de la batterie.
- Adapter le mode de saisie des informations [7].

1.2.3. Les types d'applications mobiles :

Afin de développer une application mobile il existe 3 différentes solutions comme le montre la figure 1.1.

- Développer une application native.
- Développer une application web.
- Développer une application hybride.

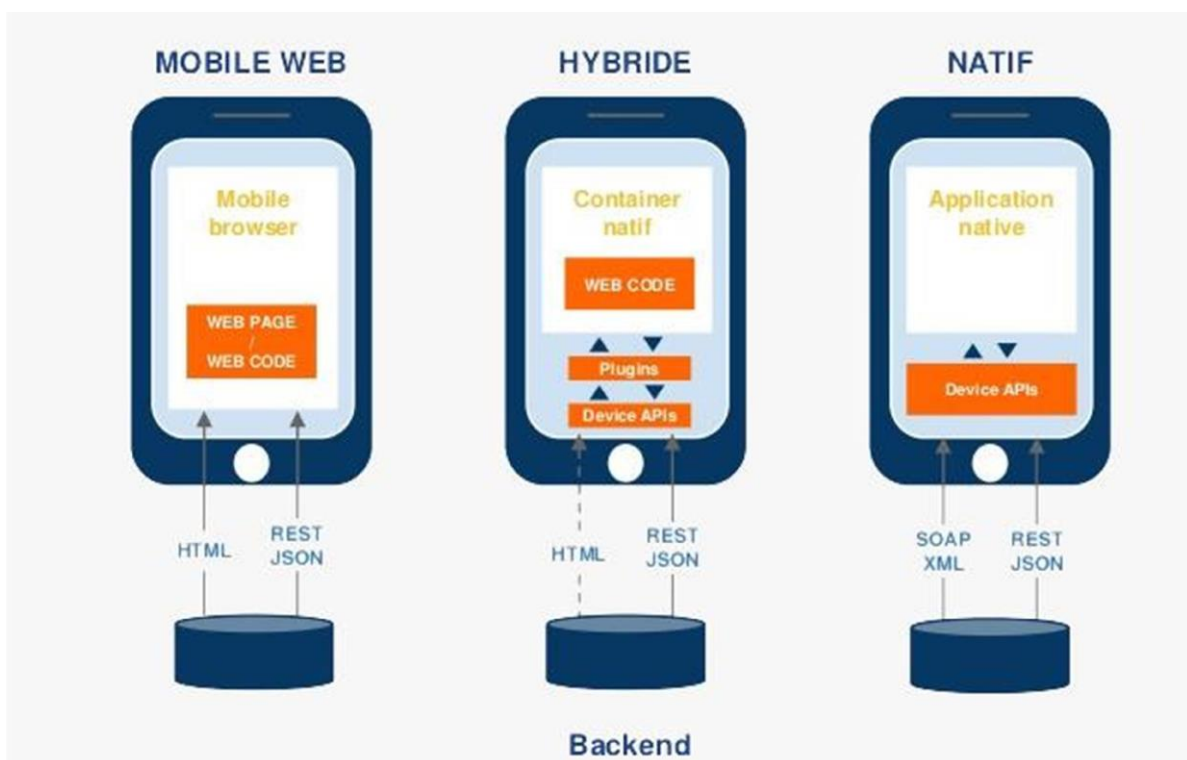


Figure 1.1 Les types d'application mobile [8].

Application native :

Il s'agit des applications conçues pour une plateforme spécifique en se référant à un langage particulier propre au système d'exploitation choisi (Java pour Android, Objective-C ou SWIFT pour iOS et C# pour Windows Phone). Ces applications sont distribuées uniquement par l'intermédiaire de son magasin (Play Store, App Store, etc.).

Le développement natif recourt essentiellement à la mémoire du smartphone ainsi que la facilité d'intégrer toutes les fonctionnalités liées au système d'exploitation visé tel que le GPS et la caméra, cela nous amène à des applications mobiles performantes, bien développées et riches en matière de fonctionnalités.

Le souci du développement natif c'est qu'on doit développer la même application pour chaque plateforme ce qui implique que le temps de développement va être multiplié par le nombre de plateforme visé [4] [5].

Application web (WebApps) :

En opposition à une application native, les applications web sont réellement des sites web développés avec les outils de développement web actuel : HTML5, CSS3 et JavaScript et conçus pour être utilisés sur les tailles des écrans mobiles.

Ils sont accessibles via le navigateur web du smartphone et ne nécessitent pas forcément de télécharger l'application. Le point fort c'est qu'on n'a qu'à développer un seul format qui marche sur toutes les plateformes donc un gain de temps de développement énorme et une facilité de mise à jour.

Cependant l'application web ne sert pas de la mémoire du Smartphone donc si l'application nécessite d'intégrer plusieurs fonctionnalités elle sera longue à charger [4] [5].

Applications hybrides :

Tout comme les applications web, ce type est développé à partir de langages web qui vont par suite s'intégrer à des « WebView » dans la partie native.

Donc c'est des applications multiplateformes qui peuvent en plus utiliser les fonctionnalités natives.

Néanmoins, les applications hybrides peuvent parfois ne pas bien s'adapter au système d'exploitation du smartphone pour des raisons de résolution et d'autres [4] [5].

1.2.4. Les avantages et les inconvénients des applications mobiles :**1.2.4.1. Les avantages :**

- Une parfaite ergonomie est assurée pour les applications mobiles en comparaison aux sites mobiles cela encourage les utilisateurs à demeurer fidèles aux applis. En effet, le développement d'application mobile tient compte la taille du smartphone, le temps de chargement et autres paramètres.
- Les applications mobiles favorisent l'intégration des options de téléphone et ainsi, l'expérience utilisateur devient plus développée.
- Pas besoin d'avoir accès à l'internet pour que l'application fonctionne.
- Facile à trouver sur les stores par rapport aux sites mobiles, les applications mobiles ont connu ainsi un usage plus répandu auprès des jeunes surtout qu'elles notifient sur les événements en cours [8].

1.2.4.2. Les inconvénients :

- Le principal inconvénient d'une application mobile est qu'elle doit respecter les règles définies par les différentes sociétés des plateformes mobiles. Que ce soit l'approbation nécessaire des Apps Store pour diffuser l'application ou ses mises à jour.
- Les conditions tarifaires imposées ou le non compatibilité avec les autres systèmes d'exploitation mobiles.
- Le coût lié au développement d'une application mobile est un frein car généralement plus élevé si elle est portée sur plusieurs plateformes (afin d'être disponible pour un maximum de mobinautes) que le coût d'un site mobile ou d'une Web App. Il faudrait Potentiellement prévoir un développement sur chaque technologie, et donc un coût supplémentaire si l'on souhaite se positionner sur tous les modèles.
- Pour que l'utilisateur ait accès à la dernière version, il faut qu'il la mette à jour depuis le store contrairement aux sites mobiles et Web App qui se mettent à jour directement [6].

1.3. Les réseaux informatiques :

1.3.1. Un réseau :

C'est un ensemble d'équipements reliés entre eux pour échanger des informations. Par analogie avec un filet (un réseau est un « petit rets », c'est-à-dire un petit filet), on appelle nœud l'extrémité d'une connexion, qui peut être une intersection de plusieurs connexions ou équipements (un ordinateur, un routeur, un concentrateur, un commutateur).

Indépendamment de la technologie sous-jacente, on porte généralement une vue matricielle sur ce qu'est un réseau. De façon horizontale, un réseau est une strate de trois couches:

- Les infrastructures.
- Les Fonctions de contrôle et de commande.
- Les services rendus à l'utilisateur.

De façon Verticale, on utilise souvent un découpage géographique : réseau local, réseau D'accès et réseau d'interconnexion [1].

1.3.2. Les catégories :

- Controller Area Network (CAN) : Réseau personnel pour les systèmes électroniques (bus), notamment utilisé dans le secteur automobile.
- Storage Area Network (SAN): Réseau de stockage.
- Local Area Network (LAN): Réseau local.
- Métropolitain Area Network (MAN) : Réseau métropolitain.
- Wide Area Network (WAN): Réseau étendu.
- Personal Area Network(PAN) : Réseau personnel [1].

1.4. Application client/serveur :

L'environnement client-serveur désigne un mode de communication à travers un réseau entre plusieurs programmes: l'un, qualifié de client, envoie des requêtes, l'autre ou les autres, qualifiés de serveurs, attendent les requêtes des clients et y répondent.

1.4.1. Caractéristiques d'un serveur et d'un client :

➤ Caractéristiques d'un programme serveur :

Il attend une connexion entrante sur un ou plusieurs ports réseaux locaux.

- A la connexion d'un client sur le port en écoute, il ouvre un socket local au système d'exploitation.
- A la suite de la connexion, le processus serveur communique avec le client suivant le protocole prévu par la couche application du modèle OSI.

➤ Caractéristiques d'un programme client :

- Il établit la connexion au serveur à destination d'un ou plusieurs ports réseaux.
- Lorsque la connexion est acceptée par le serveur, il communique comme le prévoit la couche application du modèle OSI.

1.4.2. Le fonctionnement d'un système client/serveur :

- Le client émet une requête vers le serveur grâce à son adresse et son port, Qui désigne un service particulier du serveur.
- Le serveur reçoit la commande et répond à l'aide de l'adresse de la machine client et de son port [2].

1.5. Conclusion :

En conclusion, la technologie mobile, les réseaux informatiques et les applications client-serveur sont des éléments essentiels de l'environnement technologique moderne.

Chapitre 2 :

Présentation des concepts

médicaux

2.1. Introduction :

Dans ce chapitre, nous donnerons une explication générale des pharmacies hospitalières. Nous définirons également le rôle de chaque acteur agissant dans cette organisation pour répondre aux problèmes rencontrés par chacune des parties. Nous définirons les objectifs à atteindre pour assurer une meilleure gestion.

2.2. Les pharmacies hospitalières :

Les pharmacies hospitalières sont des services pharmaceutiques qui fonctionnent au sein des établissements de santé tels que les hôpitaux, les cliniques ou les centres de soins spécialisés. Elles sont responsables de la gestion des médicaments et des produits de santé destinés aux patients hospitalisés ou traités en ambulatoire.

2.2.1. Les fonctions essentielles d'une pharmacie hospitalière :

- **Approvisionnement** : Elles sont chargées d'acquérir, de stocker et de gérer les médicaments, les dispositifs médicaux, les produits de santé et les fournitures nécessaires aux soins des patients.
- **Préparation des médicaments** : Elles préparent les médicaments sous différentes formes, notamment les doses unitaires, les solutions injectables, les pommades, les gels, les suspensions, etc., en respectant les bonnes pratiques de préparation et d'hygiène.
- **Dispensation** : Elles délivrent les médicaments aux patients hospitalisés ou aux services de soins ambulatoires, en veillant à l'exactitude de la prescription, à la posologie appropriée et aux éventuelles interactions médicamenteuses.
- **Contrôle de qualité** : Elles s'assurent de la qualité, de la sécurité et de l'intégrité des médicaments et des produits de santé en effectuant des contrôles réguliers, notamment sur les dates de péremption, les conditions de stockage et les conditions d'hygiène.
- **Conseil pharmaceutique** : Elles fournissent des conseils aux professionnels de santé, aux patients et à leurs familles sur l'utilisation appropriée des médicaments, les effets secondaires possibles, les interactions médicamenteuses et les précautions à prendre.

- **Gestion des déchets pharmaceutiques** : Elles sont responsables de la collecte, du traitement et de l'élimination des déchets pharmaceutiques conformément aux réglementations en vigueur.

Les pharmacies hospitalières jouent un rôle crucial dans la prise en charge globale des patients, en assurant l'approvisionnement sécurisé en médicaments et en fournissant un soutien pharmaceutique de qualité au sein de l'établissement de santé.

2.2.2. Rôle du personnel dans une pharmacie hospitalière :

Le personnel de la pharmacie hospitalière joue un rôle crucial dans la gestion des médicaments et la fourniture de soins pharmaceutiques aux patients. Voici les rôles clés du personnel de la pharmacie hospitalière :

- **Pharmacien hospitalier:**
 - ✓ Participe activement à la mise à jour de la liste national des médicaments et dispositif médicaux.
 - ✓ Il doit assurer la dispensation des médicaments dans l'établissement.
 - ✓ Il travaille en collaboration avec le chef service de l'hôpital (médecins, infirmières) pour d'assurer que les médicaments prescrits sont appropriés et sécuritaires pour les patients.
 - ✓ De contribuer aux opérations liées aux opérations liées à l'approvisionnement au contrôle de la distribution et de l'utilisation des médicaments et dispositif médicaux dans l'hôpital.
 - ✓ Superviser toutes les activités de la pharmacie hospitalière.
- **Chef service :**
 - ✓ Superviser les activités de préparation des médicaments, y compris le dosage, l'étiquetage et l'emballage appropriés.
 - ✓ Il est responsable de la gestion de la pharmacie hospitalière (gestion des dossiers des patients).
 - ✓ Il est responsable de la prescription et distribution des médicaments aux patients hospitalisés.
 - ✓ Il travaille en collaboration avec le pharmacien pour assurer que les médicaments répondant aux besoins des malades et pour administré les médicaments de manière approprié et surveille les effets secondaires éventuelles.

- **Techniciens en pharmacie (l'employé) :**
 - ✓ Assister les pharmaciens dans les tâches quotidiennes de la pharmacie, notamment la préparation des médicaments, la gestion des stocks, l'étiquetage et la distribution.
 - ✓ Travailler en collaboration avec le pharmacien hospitalier pour assurer la gestion quotidienne de la pharmacie.
 - ✓ Veiller à ce que les procédures de préparation des médicaments respectent les normes de sécurité et de stérilité.
 - ✓ Collaborer avec les pharmaciens pour répondre aux demandes des professionnels de la santé concernant les médicaments et fournir des conseils appropriés.
- **Assistant administratif :**
 - ✓ Gérer les tâches administratives et de bureau de la pharmacie, y compris la gestion des dossiers des patients, la facturation et la coordination des communications internes et externes.
 - ✓ Assurer la gestion des commandes de médicaments, des fournisseurs et des inventaires.
 - ✓ Soutenir les processus de documentation et de conformité aux réglementations en vigueur.

Chaque membre du personnel de la pharmacie hospitalière joue un rôle essentiel dans la fourniture de soins pharmaceutiques sécuritaires et efficaces. Ils collaborent également étroitement avec d'autres professionnels de la santé pour assurer une utilisation appropriée et rationnelle des médicaments et améliorer les résultats pour les patients.

2.2.3. Organigramme général de la pharmacie hospitalière :

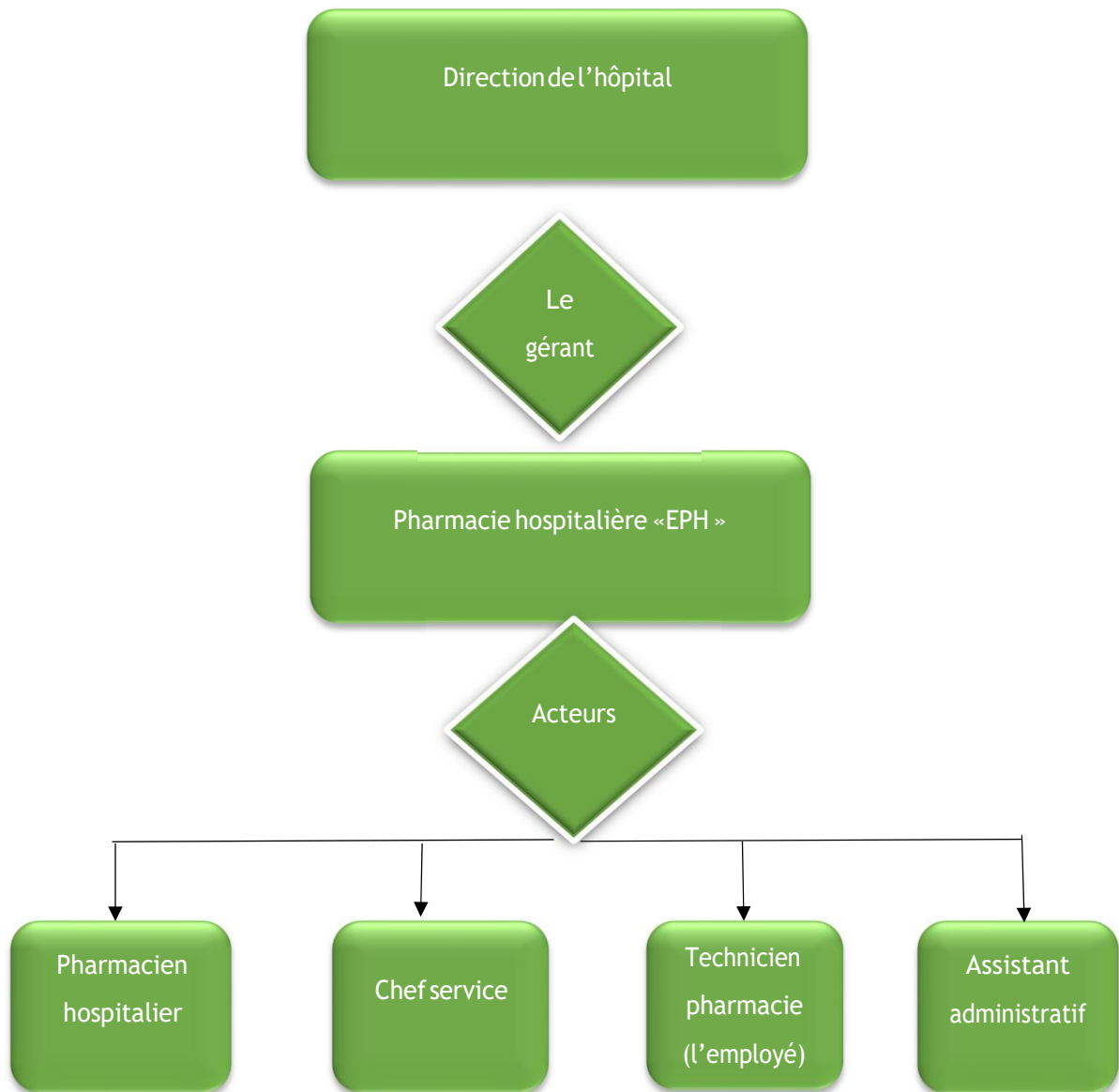


Figure 2.1 Organigramme général de pharmacie hospitalière

2.2.4. Problématique et objectifs:

Après avoir étudié l'organisation pharmaceutique hospitalière, nous analyserons les éventuels problèmes auxquels elle est confrontée, puis nous essaierons d'apporter des solutions adaptées.

2.2.4.1. Problématique:

La pharmacie hospitalière fait face à diverses problématiques liées à la gestion des médicaments et des produits de santé au sein de l'établissement de santé. Ces problématiques peuvent inclure :

- **Gestion efficace des stocks :** Assurer un approvisionnement adéquat en médicaments tout en évitant les pénuries ou les surstocks, en optimisant la rotation des stocks et en contrôlant les dates de péremption.
- **Sécurité des patients :** Garantir la sécurité des patients en évitant les erreurs de médication, en identifiant et en prévenant les interactions médicamenteuses potentielles, et en assurant une manipulation et une administration appropriées des médicaments.
- **Coût et rentabilité :** Optimiser les ressources financières en recherchant des solutions rentables pour l'achat et la gestion des médicaments et des produits de santé, tout en maintenant un niveau de qualité élevé.
- **Qualité et conformité réglementaire :** Respecter les normes de qualité et les réglementations en vigueur en matière de préparation, de stockage, de dispensation et de contrôle des médicaments et des produits de santé.
- **Sécurité des données :** comment sécuriser les données des patients et les informations pharmaceutiques ?

2.2.4.2. Objectifs:

La phase d'analyse vise à décrire les besoins et les exigences de l'organisation pharmaceutique hospitalière de manière précise, concise, correcte et compréhensible. Il s'agit de fournir un cahier des charges pour permettre la conception de la solution.

La phase d'analyse permet de s'accorder sur « que doit faire l'application mobile ? »

L'organisation pharmaceutique hospitalière souhaite disposer d'une application mobile qui lui permettra de faire ce qui suit :

- **Optimisation de la gestion des stocks :** L'application mobile peut offrir une visibilité en temps réel sur les niveaux de stock, les dates d'expiration, les réapprovisionnements nécessaires, etc. Cela permet aux pharmaciens de gérer leurs stocks de manière plus précise, d'éviter les ruptures de stock et de réduire les coûts liés à la sur stockage.
- **Amélioration de la coordination des commandes et des approvisionnements :** L'application peut simplifier le processus de commande de médicaments et de fournitures médicales en permettant aux pharmaciens de passer des commandes directement depuis leur appareil mobile. Cela facilite la communication avec les fournisseurs et réduit les délais de traitement des commandes.

2.2.5. Communication interne efficace : L'application peut intégrer des fonctionnalités de messagerie instantanée et de partage d'informations, ce qui facilite la communication entre les membres de l'équipe pharmaceutique au sein de l'hôpital. Cela permet de transmettre rapidement des informations importantes, de partager des mises à jour et de résoudre les problèmes plus rapidement. Besoins fonctionnel:

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

- **Échange d'informations médicales :** Les pharmacies hospitalières doivent pouvoir communiquer rapidement et de manière sécurisée avec d'autres pharmacies, les médecins et les fournisseurs de soins de santé. Cela inclut la transmission des ordonnances, la vérification des interactions médicamenteuses, et la consultation des dossiers médicaux des patients.
- **Gestion des stocks :** Les pharmacies hospitalières doivent être en mesure de gérer efficacement leurs stocks de médicaments, de dispositifs médicaux et de fournitures connexes. Cela comprend
- Le suivi des niveaux d'inventaire, les commandes auprès des fournisseurs, et la gestion des dates d'expiration.

- **Suivi des médicaments** : Les réseaux doivent permettre le suivi précis des médicaments depuis leur réception jusqu'à leur administration aux patients. Cela implique la gestion des codes-barres, des numéros de lot et des informations sur les fabricants.
- **Intégration avec d'autres systèmes hospitaliers** : Les réseaux doivent être capables de s'intégrer avec d'autres systèmes hospitaliers tels que les dossiers médicaux électroniques (DME), les systèmes de gestion des patients et les systèmes de facturation pour assurer une coordination et une continuité des soins efficaces.

2.2.6. Besoins non-fonctionnel:

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

- **Sécurité des données** : Les réseaux entre les pharmacies hospitalières doivent garantir la confidentialité, l'intégrité et la disponibilité des données sensibles, notamment les informations médicales des patients. Cela nécessite la mise en place de mesures de sécurité telles que le cryptage des données, l'authentification des utilisateurs et la protection contre les accès non autorisés.
- **Fiabilité et disponibilité** : Les réseaux doivent être fiables et disponibles en tout temps pour assurer un accès continu aux informations nécessaires à la dispensation des médicaments. Les pannes de réseau ou les temps d'arrêt peuvent avoir un impact direct sur les soins aux patients.
- **Performance** : Les réseaux doivent être suffisamment rapides et réactifs pour permettre une utilisation fluide des applications et des systèmes utilisés par les pharmacies hospitalières. Des performances médiocres peuvent entraîner des retards dans la dispensation des médicaments et des erreurs potentielles.
- **Scalabilité** : Les réseaux doivent être évolutifs pour répondre aux besoins croissants des pharmacies hospitalières. Ils doivent être capables de gérer efficacement une augmentation du volume de données et du nombre d'utilisateurs sans compromettre les performances.

- **Conformité réglementaire :** Les réseaux doivent être conformes aux réglementations et aux normes vigueurs, telles que la Health Insurance Portability and Accountability Act (HIPAA) aux États-Unis, qui protège la confidentialité et la sécurité des informations de santé.

Ces besoins fonctionnels et non fonctionnels sont essentiels pour garantir un fonctionnement optimal des réseaux entre les pharmacies hospitalières, en veillant à ce que les médicaments soient correctement gérés et administrés dans un environnement sûr et efficace.

2.3. Conclusion :

Dans ce deuxième chapitre, nous avons présenté l'organisation de la pharmacie hospitalière et défini le rôle de chaque membre du personnel de la pharmacie. Ensuite, nous avons identifié les différentes problématiques auxquelles ils étaient confrontés dans la réalisation de leurs tâches quotidiennes, ce qui nous a permis de dessiner les besoins fonctionnels et non fonctionnels du système à concevoir, puis nous avons choisi une application mobile.

Chapitre 3 :

Analyse et conception

3.1. Introduction:

Dans ce chapitre dédié à la conception, nous allons définir le langage de modélisation UML et les diagrammes qu'on a utilisés dans notre system puis le rôle de chaque acteur qui Interagit avec le système. Nous allons aussi modéliser les rôles des acteurs principaux sous forme de diagrammes de cas d'utilisation et on utilise réseau de Pétri pour modéliser l'architecture globale du système puis nous définirons les cas d'utilisation et nous les modéliserons sous forme de diagramme de séquence. Nous finirons par le diagramme de classe de cette application.

3.2. Modélisation :

Notre modélisation de la solution se fera en utilisant des diagrammes UML.

3.2.1. Unified Modeling Language (UML):

UML (Unified Modeling Language) est un langage de modélisation objet né de la fusion de trois Langages de modélisation objet (Booch, OMT, OOSE) qui ont révolutionnés la modélisation dans les années 90. UML est dit universel car il est indépendant des langages de programmation, des domaines d'application et aussi du processus de développement adopté.

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

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

3.2.1.1. Diagrammes UML utilisées :

Pour modéliser notre system on a utilisé les trois diagrammes suivants :

Tableau 3.1 Descriptif des diagrammes UML

| Diagramme | Objectif | Type |
|--------------------------------|--|-------------|
| Diagramme de cas d'utilisation | <ul style="list-style-type: none"> • Décrire les interactions entre les différents acteurs externes (utilisateurs du cas) et les fonctionnalités du système. • Identifier les acteurs, leur responsabilité respective et de décrire leurs besoins. • Mettre en évidence les services rendus par le système. | Fonctionnel |
| Diagramme de séquence | <ul style="list-style-type: none"> • Validation des cas d'utilisation pour comprendre la logique de l'application. • Complete le diagramme de cas d'utilisation en mettant en évidence les objets et leurs interactions d'un point de vue temporelle. | Dynamique |
| Diagramme de classe | <ul style="list-style-type: none"> • Point centrale de la modélisation du système pour décrire ce que le système doit faire (analyse) et avec quoi il va faire (conception). • Représenter la structure en terme de classes et relations entre elles. | Statique |

3.3. Les acteurs du système:

3.3.1. Définition :

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

3.3.2. Identification des acteurs :

▪ Les acteurs principaux:

La pharmacie hospitalière emploie chef service, pharmacien hospitalier, Assistant administratif et l'employé, dans notre system le rôle de l'employé et le pharmacien hospitalier presque ont le même rôle et doivent avoir les mêmes privilèges.

1. Chef service :

Le rôle du chef service est :

- ❖ S'inscrire.
- ❖ S'identifier.
- ❖ Échanger médicaments.
- ❖ Gérer la réception.

2. Pharmacien hospitalier :

Le rôle du pharmacien hospitalier est :

- ❖ S'inscrire.
- ❖ S'identifier.
- ❖ Gérer stock médicaments.
- ❖ Échanger médicaments.
- ❖ Gérer la réception.

3. Assistant Administratif :

Le rôle de l'assistant administratif est :

- ❖ S'inscrire.
- ❖ Gérer les utilisateurs.
- ❖ Gérer les échanges.
- ❖ Règlement facturation.

3.3.3. Identification de cas d'utilisation :

Dans l'application à développer, nous avons définis les cas d'utilisation suivante :

Tableau 3.2 Identification des cas d'utilisations

| N | Acteurs | Cas d'utilisation |
|---|--|---|
| 1 | Pharmacien hospitalier /chef service /assistant administratif | Authentification |
| 2 | Pharmacien hospitalier | S'inscrire S'identifier Gérer stock de médicaments (ajouter, enregistrer, demande on attend, modifier, rechercher médicaments, consulter, supprimer) Échanger médicaments (Commander (ajouter, modifier, annuler) demander (ajouter, annuler, valider)) Gérer la réception (vérifier, rechercher médicaments) |
| 3 | Chef service | S'inscrire S'identifier Échanger médicaments (Commander (ajouter, modifier, annuler) demander (ajouter, annuler, valider)) Gérer la réception (vérifier, rechercher médicaments) |
| 4 | Assistant administratif | S'inscrire S'identifier Gérer les utilisateurs (ajouter, modifier, rechercher un utilisateur, consulter, supprimer) Gérer les échanges (gestion des transactions (choisir, rechercher bon échange, modifier, annuler) liste des critères |

Cas d'utilisation :

- **Authentification** : cette opération permet à l'utilisateur de s'authentifier avant d'accéder à l'application, elle permet aussi d'assurer de l'identité de l'utilisateur.
- **Gérer stock médicaments** : permet à l'utilisateur de gérer le stock de médicaments et il inclut des fonctionnalités telles que l'ajout, la modification, de rechercher et d'annuler l'enregistrement des médicaments.
- **Gérer les utilisateurs** : permet de l'assistant administratif d'ajouter, modifier ces fonctionnalités et supprimer les utilisateurs.
- **Échanger médicaments** : permet au pharmacien hospitalier et chef service de commander le médicament en cas d'existe déjà ou de demander le médicament en cas n'est pas existé.
- **Gérer les échanges** : permet à l'assistant administratif de faire gestion des transactions à travers la liste des critères pour choisir le bon échange.
- **Gérer de la réception** : en collaboration avec le chef service et pharmacien hospitalier de vérifier les médicaments livrés et assurer la correspondance entre ce qui est commander et ce qui est livré, tout au niveau de la quantité et la qualité.
- **Règlement facturation** : permet au assistant administratif accéder à la liste de facturations et de faire règlement de la facturation au coure du période du temps.

3.4. Diagramme de cas d'utilisation :

3.4.1. Définition :

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

3.4.2. Diagramme de cas d'utilisation du pharmacien hospitalier et chef service :

Le diagramme ci-dessous représente les cas d'utilisations identifiées pour le pharmacien hospitalier et chef service.

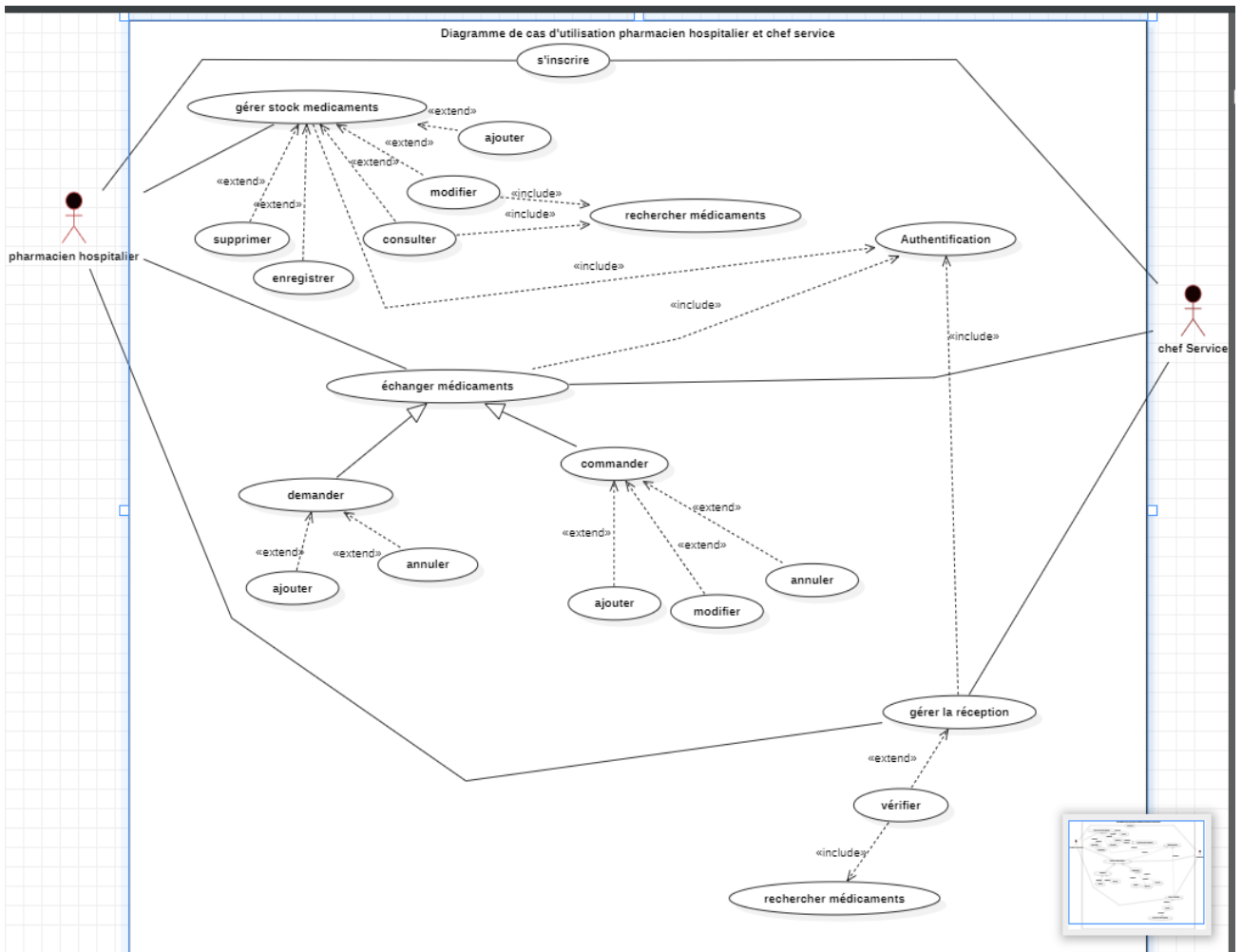


Figure3.1 Diagramme de cas d'utilisation du pharmacien hospitalier et chef service.

3.4.3. Diagramme de cas d'utilisation de l'assistant administratif:

Le diagramme ci-dessous représente les cas d'utilisations identifiées pour l'assistant administratif.

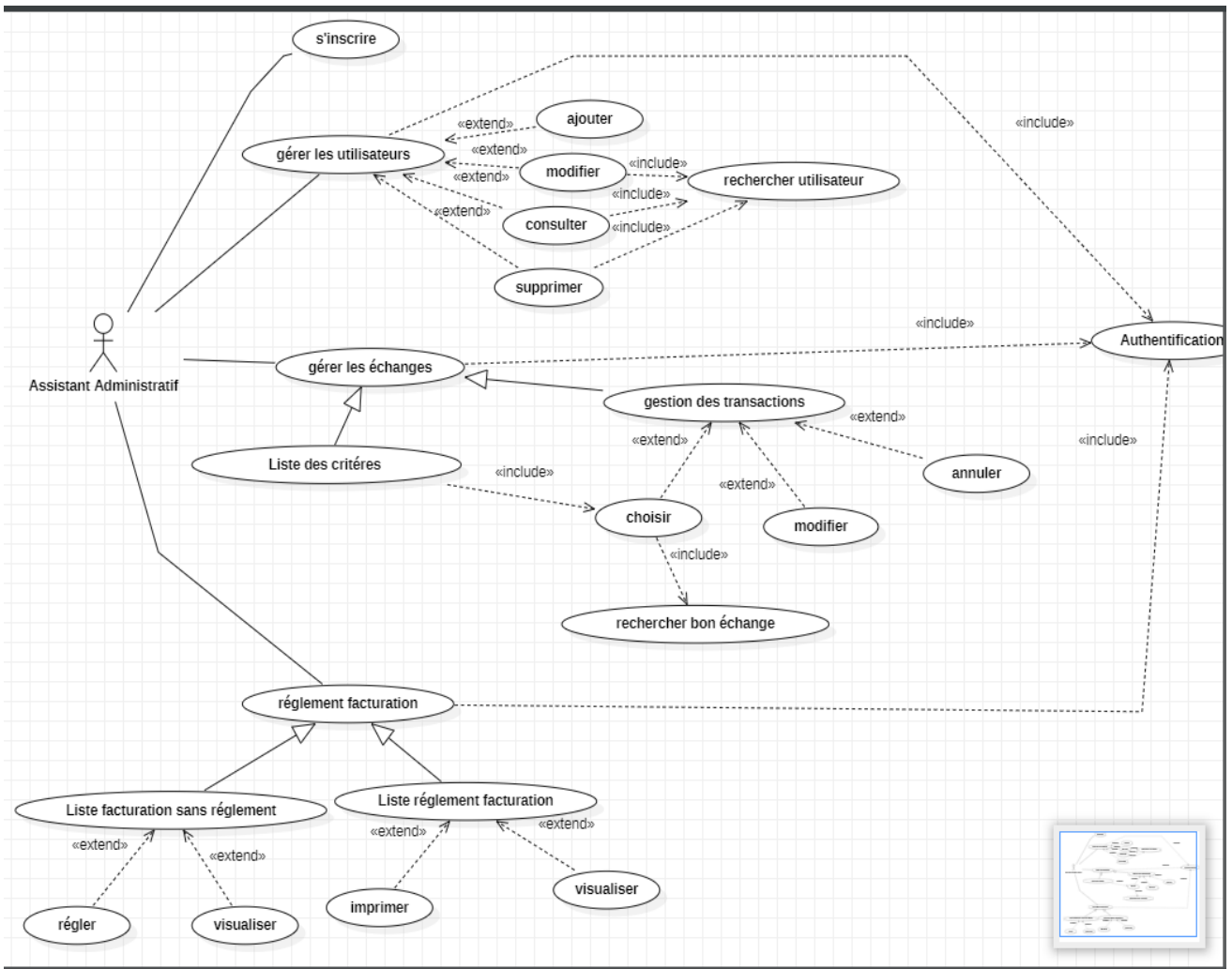


Figure 3.2 Diagramme de cas d'utilisation de l'assistant administratif

3.5. Diagramme de séquence :

Le diagramme de séquence est la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation UML. On montre ces interactions dans le cadre d'un scénario d'un diagramme des cas d'utilisation. Le but étant de décrire comment se déroule les actions entre les acteurs objets. Dans ce qui suit, on va présenter les différents diagrammes de séquence de notre application [30].

3.5.1. Description textuelle du diagramme de séquence « s'inscrire » :

Tableau 3.3 Description textuelle du diagramme de séquence « inscription »

| Sujet | Description |
|----------------|--|
| Objectif | Faire une inscription d'utilisateur |
| Acteur | Pharmacien hospitalier, chef service, assistant administratif |
| Pré-condition | Accès à l'application dans leur téléphone mobile |
| Post condition | Inscription d'acteur |
| Scénarios | <ul style="list-style-type: none"> - L'utilisateur demande d'inscrire. - L'interface d'inscription renvoie le formulaire d'inscription. - L'utilisateur remplit le formulaire. - L'interface d'inscription vérifie les informations saisies. - Elle renvoie une requête d'enregistrement vers le serveur. - Le serveur fait la vérification et il envoie une requête d'enregistrement vers la base de données. - La base de données fait le chargement, il enregistre les informations et envoie un message de succès vers serveur. - Serveur envoie un message de succès vers l'interface. - L'interface d'inscription affiche le profil créé par l'utilisateur. |
| Alternative | <ul style="list-style-type: none"> - Si la page d'inscription affiche un message d'erreur (s'il y a un champ vide ou l'email existe déjà) sinon elle effectuée l'opération. - Si on a erreur interne commit dans le serveur il passe un message d'erreur sinon l'opération et fait avec succès. - Si l'utilisateur déjà existe s'affiche message erreur sinon l'opération effectuée. |

3.5.1.1. Désignation du diagramme de séquence « s’inscrire » :

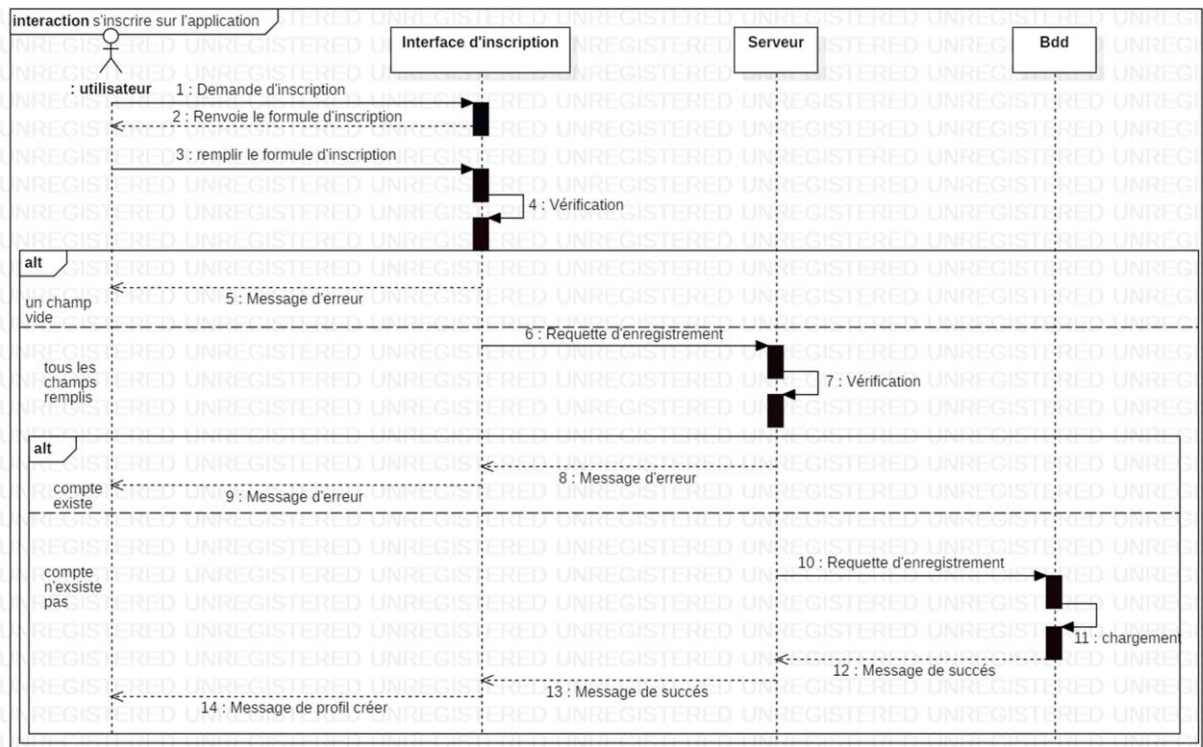


Figure 3.3 Diagramme de séquence « s’inscrire sur l’application »

3.5.2. Description textuelle de diagramme de séquence « authentification » :

Tableau 3.4 Description textuelle du diagramme de séquence « authentification »

| Sujet | Description |
|-------------------|--|
| Objectif | s'authentifier |
| Acteur principale | utilisateur |
| Pré-condition | L'utilisateur est déjà enregistré dans la base de données |
| Post condition | L'utilisateur est connecté |
| Scénarios | <ul style="list-style-type: none"> - L'utilisateur demande d'authentification. - La page d'authentification renvoie le formulaire. - L'utilisateur remplit le formulaire. - La page d'authentification fait une vérification des champs. - Elle envoie une requête vers le serveur. - Ce dernier fait une vérification, il envoie une requête vers BDD et elle renvoie une réponse. - Le serveur fait vérification, il passe un message de succès et autorise l'accès. - La page d'authentification envoie l'utilisateur à la page demandée. |
| Alternative | <ul style="list-style-type: none"> - Si la page trouve erreur (champ non rempli, les informations incorrectes) elle effectue une erreur sinon elle renvoie les informations vers le serveur pour l'autorisation d'accès. - Le serveur affiche un message d'erreur s'il y a une erreur de connexion sinon la connexion est faite. - Si l'utilisateur n'existe pas, le serveur envoie un message d'échec sinon elle charge la page demandée. |

3.5.2.1. Désignation du diagramme de séquence « authentification » :

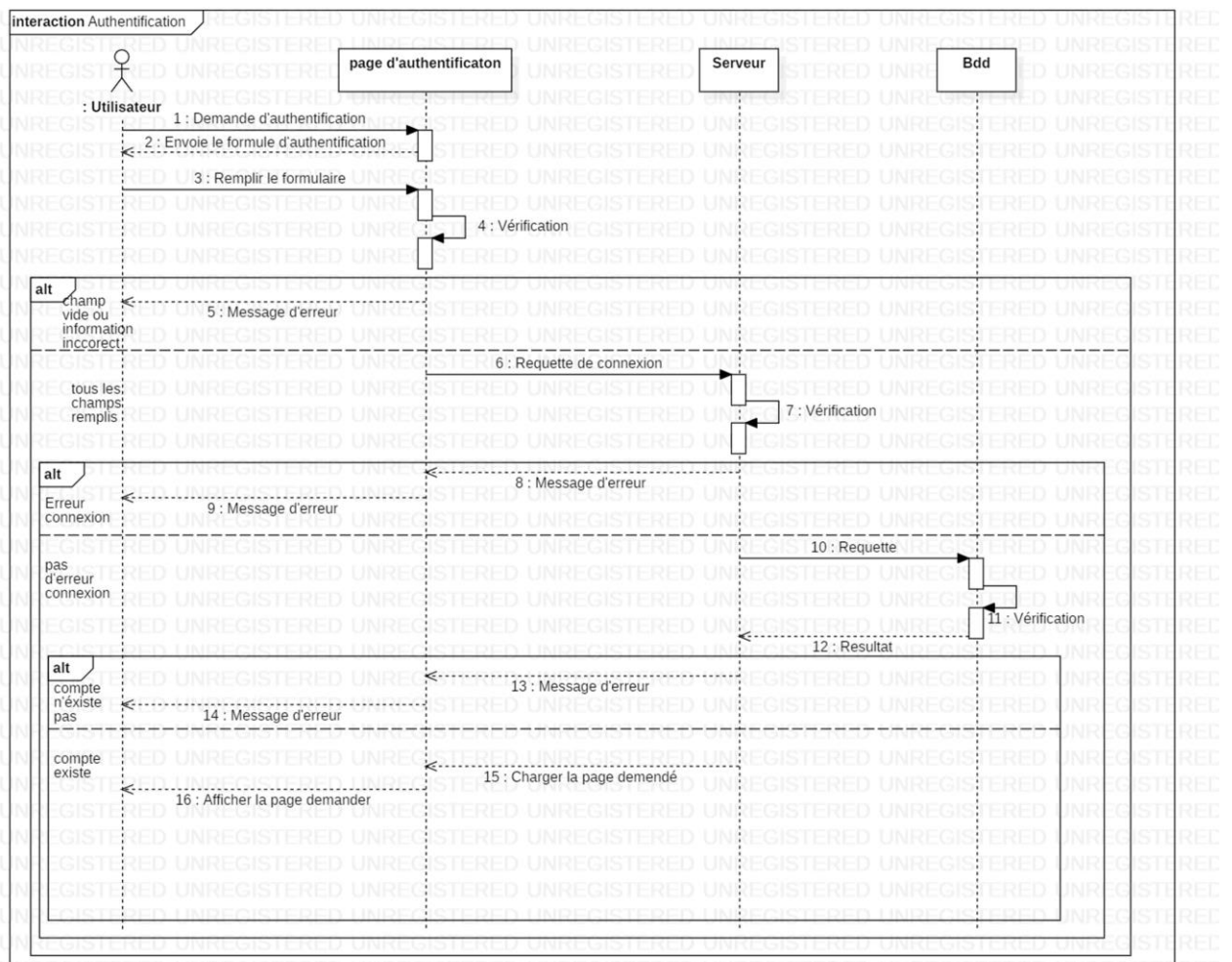


Figure 3.4 Diagramme de séquence « authentification »

3.5.3. Description textuelle de diagramme de séquence « gérer les utilisateurs » :

Tableau 3.5 Description textuelle de diagramme de séquence « gérer les utilisateurs »

| Sujet | Description |
|-------------------|---|
| Objectif | Permet à l'acteur d'ajouter, supprimer, modifier ou consulter les informations du patient. |
| Acteur principale | Assistant administratif |
| Pré-condition | L'acteur doit être identifié |
| Post condition | L'acteur gérer l'utilisateur |
| Scénarios | <ul style="list-style-type: none"> - L'acteur consulte la liste des utilisateurs. - L'interface passe requête vers serveur. - Ce dernier passe requête vers BDD. |

- Elle fait le chargement et renvoie les données vers le serveur.
- Ce dernier envoie les données vers l'interface.
- L'interface affiche la liste des utilisateurs.

En cas d'ajouter utilisateur :

- L'acteur demande d'ajouter utilisateur.
- L'interface renvoie le formulaire.
- L'acteur saisie et valide le formulaire.
- L'interface fait la vérification et elle renvoie une requête d'ajouter l'utilisateur vers le serveur.
- Ce dernier fait la vérification puis il envoie la requête vers BDD.
- Cette dernière renvoie réponse vers serveur.
- Serveur fait la vérification et renvoie requête d'enregistrement vers BDD.
- BDD fait le chargement et passe message de succès vers serveur.
- Serveur envoie message de succès
- L'interface envoie message de succès.
- L'utilisateur est ajouté.

En cas de modifier utilisateur :

- L'acteur choisit utilisateur à modifier et demande de modification.
- L'interface renvoie de modification.
- L'acteur saisie et valide le formulaire.
- L'interface fait la vérification.
- L'interface envoie une requête de modification vers serveur.
- Ce dernier fait la vérification et envoie requête de modification vers BDD.
- BDD fait modification et envoie message de succès.
- Serveur envoie message de succès vers l'interface.
- Cette dernière envoie message de succès.
- L'utilisateur est modifié.

En cas de supprimer :

- L'acteur choisit patient à supprimer et demande de la suppression.
- L'interface renvoie le message de confirmation.
- L'acteur confirme la suppression.
- L'interface renvoie une requête de suppression vers serveur.
- Ce dernier fait la vérification et envoie une requête de suppression vers BDD.

| | |
|-------------|--|
| | <ul style="list-style-type: none">- Elle fait la suppression et envoie message de succès.- Serveur envoie message d'accès.- L'interface envoie message d'accès.- L'utilisateur est supprimé. |
| Alternative | <p>En cas d'ajouter :</p> <ul style="list-style-type: none">- Si erreur commit dans le remplissage du formulaire (exemple champ vide) l'interface envoie message d'erreur si non l'utilisateur est créé.- Si l'utilisateur est déjà existé serveur passe message d'erreur sinon il continu l'opération. <p>En cas modifier :</p> <ul style="list-style-type: none">- Si une erreur commit dans le formulaire de modification l'interface envoie message d'erreur si non le patient est modifié. <p>En cas supprimer :</p> <ul style="list-style-type: none">- Si on doit annuler suppression l'interface envoie message d'annulation sinon le patient est supprimer. |

3.5.3.1 Désignation du diagramme de séquence « gérer les utilisateurs » :

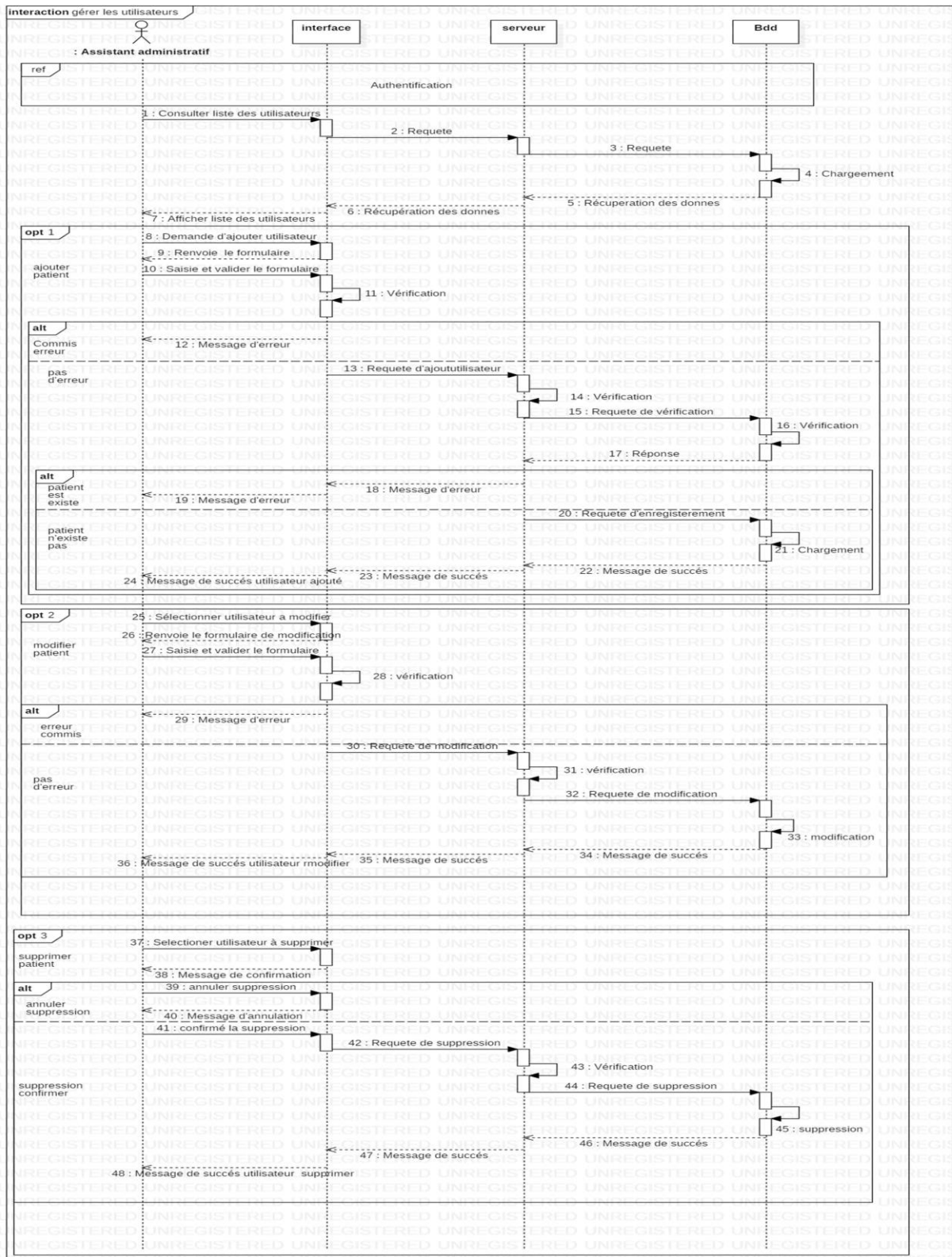


Figure 3.5 Diagramme de séquence « gérer les utilisateurs »

3.5.4. Description textuelle de diagramme de séquence « échanger médicaments » :

Tableau 3.6 Description textuelle de diagramme de séquence « échanger médicaments »

| Sujet | Description |
|---------------|---|
| Objectif | échanger des médicaments |
| Acteurs | Pharmacien hospitalier, chef service |
| Pré-condition | Authentification |
| Scénarios | <p>En début:</p> <ul style="list-style-type: none"> - L'acteur demande de formulaire de gestion d'échange. - L'interface passe requête d'affichage vers serveur. - Ce dernier fait le traitement et envoie le formulaire. <p>En cas de commander :</p> <ul style="list-style-type: none"> - L'acteur demander formule de la commande. - L'interface renvoie requête d'affichage vers serveur. - Ce dernier fait le traitement et envoie formulaire. - L'acteur remplir formulaire de la commande. - L'interface fait la vérification et envoie la requête d'enregistrement vers serveur. - Ce dernier fait la vérification et envoie la requête vers BDD. - BDD fait l'enregistrement et envoie message de succès vers serveur. - Ce dernier envoie message de succès vers interface. - Commande enregistrée. <p>En cas de demander :</p> <ul style="list-style-type: none"> -L'acteur demander le formuler de la demande. - L'interface passe la requête d'affichage vers serveur. - Ce dernier fait le traitement et envoie le formulaire. - L'acteur remplit formulaire de la demande. - L'interface fait la vérification d'existence de médicament puis envoie la requête d'enregistrement vers serveur. - Ce dernier fait traitement et envoie vers BDD - BDD fait l'enregistrement et envoie message de succès vers serveur. - Serveur envoie message de succès vers l'interface. - La demande est enregistrée. |
| Alternative | <p>En cas demander :</p> <ul style="list-style-type: none"> -si erreur commit dans la vérification du l'existence de médicaments l'interface envoie message demande d'attend vers serveur et BDD sinon demande enregistrer. |

3.5.4.1. Désignation de diagramme de séquence « échanger médicaments » :

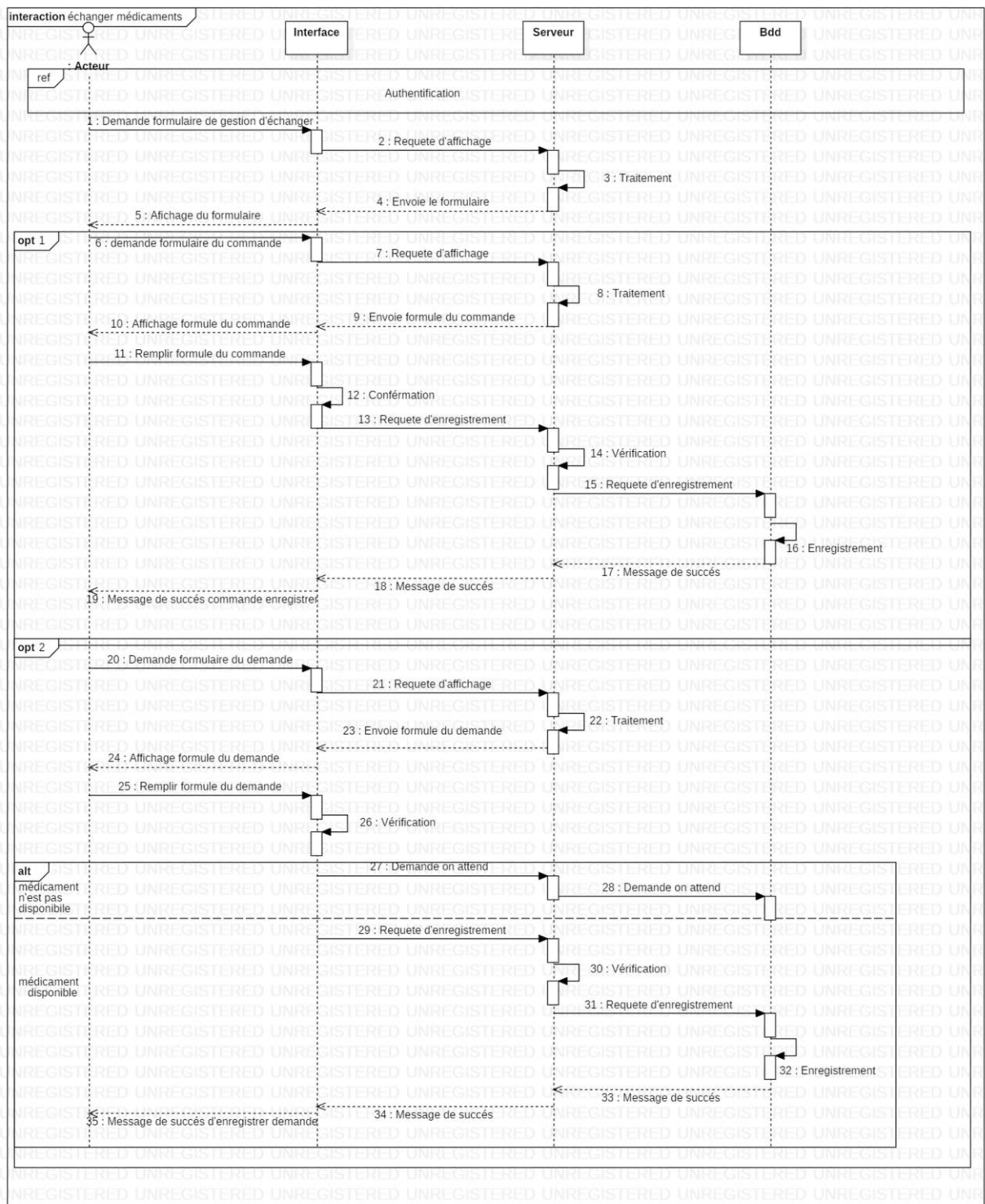


Figure 3.6 Diagramme de séquence « échanger médicaments »

3.5.5. Description textuelle de diagramme de séquence « gérer les échanges » :

Tableau 3.7 Description textuelle de diagramme de séquence « gérer les échanges »

| Sujet | Description |
|-------------------|---|
| Objectif | Gérer les échanges |
| Acteur principale | Assistant administratif |
| Pré-condition | Authentification |
| Scénarios | <ul style="list-style-type: none"> - L'acteur demande formulaire de gestion des transactions. - L'interface envoie requête d'affichage vers serveur. - Ce dernier fait traitement et envoie le formulaire. - Dans ce formulaire l'acteur demande formulaire de choisir soit une commande ou demande à travers la liste des critères (date de péremption de médicament, quantité...) . - L'interface fait la vérification de liste de critères et envoie requête de choisir bon échange vers serveur. - Ce dernier fait vérification de liste de critères et envoie la requête vers BDD. - BDD fait le traitement et envoie message de succès vers serveur. - Serveur envoie message de succès vers l'interface. - Bon échange à choisie. |

3.5.5.1. Désignation de diagramme de séquence « gérer les échanges » :

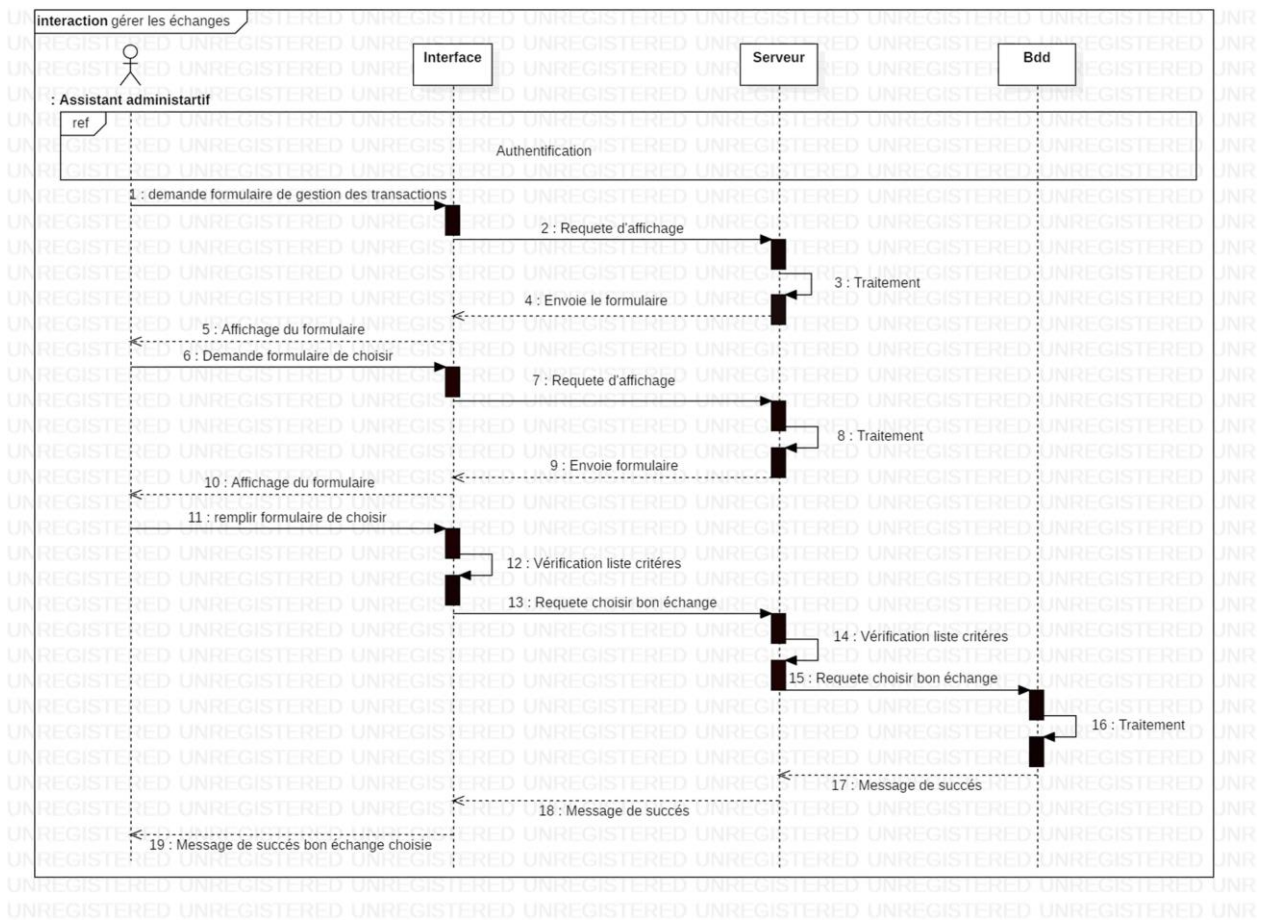


Figure 3.7 Diagramme de séquence « gérer les échanges »

➤ **Algorithme d'arbitrage :**

C'est un algorithme facilite le travail dans le serveur pour choisir le bon échange à travers des critères sont :

- **Abonner** : c'est le type d'abonnement du client (pharmacie hospitalier).
- **Consommer** : quantité des consommations des médicaments de client.
- **Régler facturation** : les dates de règlement la facturation de client.

Algorithme d'arbitrage :

```

Var a,c,r : entier
    i,n : entier
    M : réel
Debut
Lire(a,c,r)
    
```

M → 0

Pour i de 1 à n faire

M → (a+c+r) /3

M → M+i

Si (M > M+i)

Ecrire (échange valider)

Sinon

Ecrire (échange n'est pas valide)

Fin Si

Fin Pour

Fin

3.6. Diagramme de classe :

Le diagramme de classes est sans doute le diagramme le plus important à représenter pour les méthodes d'analyse orientées objet. En effet, il permet de spécifier QUI intervient à l'intérieur du système. Un diagramme de classes fait abstraction des aspects dynamiques et temporels du système, il va permettre de représenter une vue statique du système d'information. Il s'agit plutôt des relations entre les classes, des services rendus et utilisés par chacune d'elles et de l'articulation de l'ensemble [8].

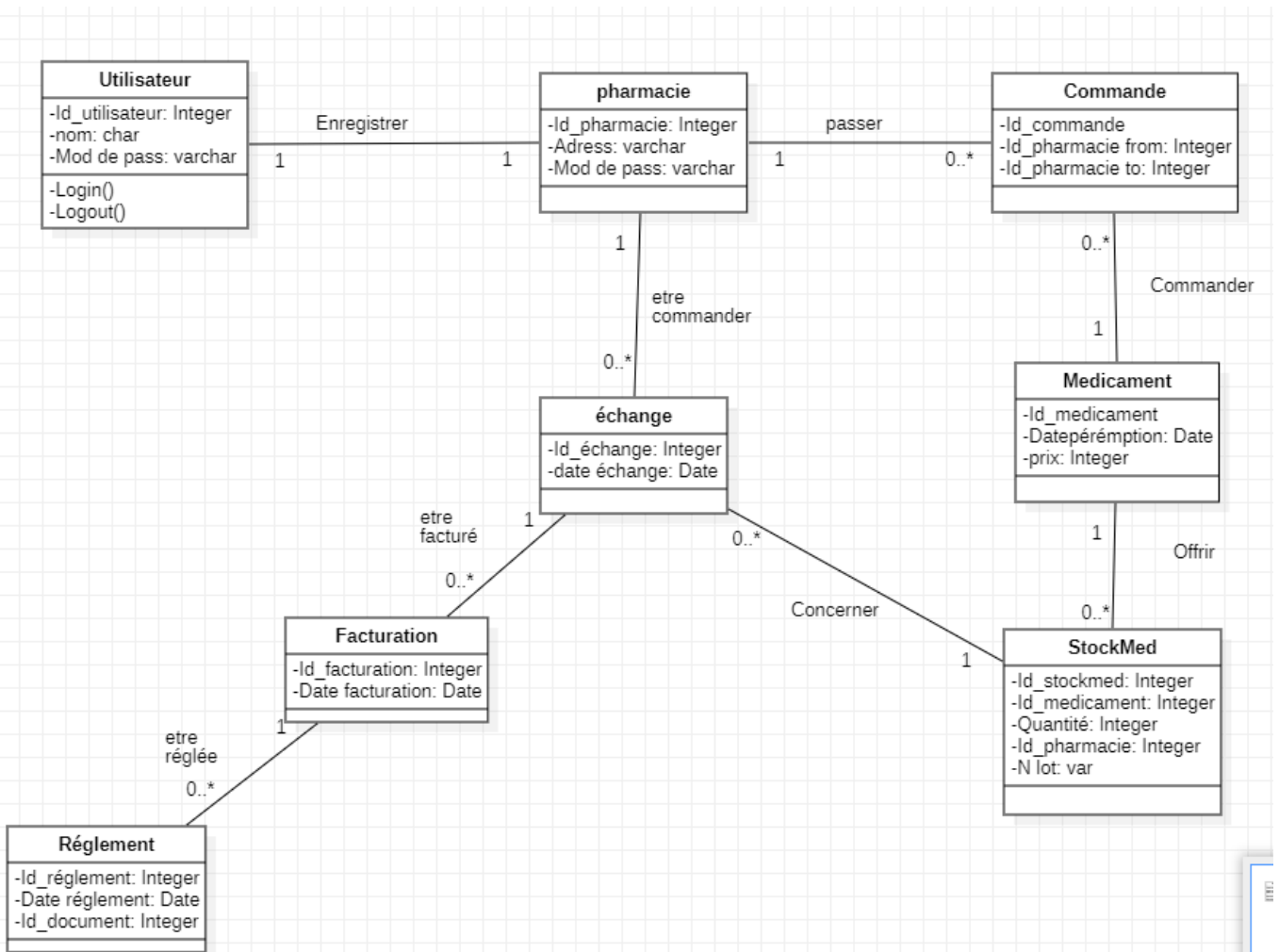


Figure 3.8 Diagramme de classe

3.7. Conclusion :

Dans ce chapitre, nous avons modélisé notre application en apportant des réponses à nos questions de modélisation et de conception. En s'appuyant sur l'analyse des besoins de notre application. Nous avons modélisé sous forme de diagrammes de cas d'utilisation puis en utilisons réseau pétri pour modéliser structure globale, puis en utilisons digramme de séquence, enfin on concevoir notre application sous forme de diagrammes de classe et schémas relationnelle, reste à concevoir et a créé notre application.

Chapitre 4 :

Réalisation

4.1. Introduction :

Ce chapitre a pour objectif majeur de présenter le produit final. C'est la phase de réalisation de notre application mobile qui utilise des technologies spécifiques.

4.2. L'environnement de développement :

4.2.1. L'environnement matériel :

Pour développer ce site on a utilisé une machine, qui est configurée comme suit :

- ✓ Mémoire Vive : 4 Go.
- ✓ Disque Dur : 500 Go.
- ✓ Processeur: AMD E1-2100APU with Radeon (TM) HD Graphics 1.00 GHZ.
- ✓ Système d'exploitation : Windows 10.

4.2.2. L'environnement logiciel :

Lors du développement de ce site, on a utilisé les outils logiciels et les langages suivants :

- WinDev mobile 25
- W langage
- HFSQL server
- Star UML
- Canva

4.3. Définition des logiciels et langages utilisés :

WinDev mobile 25 :

WinDev Mobile 25 est une version spécifique du logiciel de développement d'applications WinDev, développé par la société française PC SOFT. WinDev Mobile 25 est conçu pour faciliter la création d'applications mobiles



Multiplateformes, c'est-à-dire des applications pouvant fonctionner sur différents systèmes d'exploitation mobiles tels qu'Android, iOS et Windows.

Avec WinDev Mobile 25, les développeurs peuvent créer des applications mobiles en utilisant un langage de programmation de haut niveau appelé WLanguage, qui est spécialement conçu pour être convivial et intuitif. Le logiciel offre une variété de fonctionnalités pour accélérer

le processus de développement, telles que des assistants de création d'interfaces graphiques, des bibliothèques d'objets prédéfinis et des outils de gestion des données.

WinDev Mobile 25 offre également des fonctionnalités avancées pour l'intégration de services web, l'accès aux capteurs du téléphone (comme l'appareil photo ou le GPS), la gestion des notifications push et la connectivité avec des bases de données locales ou distantes.

L'objectif principal de WinDev Mobile 25 est de simplifier et d'accélérer le processus de développement d'applications mobiles, en permettant aux développeurs de se concentrer sur la logique métier de leurs applications plutôt que sur les détails techniques de chaque plateforme mobile.

W langage :

Le langage W est un langage de programmation spécifique à l'environnement de développement intégré (EDI) appelé WinDev.



Le langage W est conçu pour être utilisé avec WinDev et offre un ensemble de fonctionnalités spécifiques pour le développement d'applications Windows. Il s'agit d'un langage de haut niveau qui permet de développer des applications graphiques avec une interface utilisateur conviviale.

Le langage W offre des fonctionnalités telles que la gestion des formulaires, des fenêtres, des contrôles et des événements. Il permet également d'accéder aux fonctionnalités avancées du système d'exploitation Windows, telles que les fichiers, les bases de données, les communications réseau, etc.

En utilisant le langage W, les développeurs peuvent créer des applications Windows complètes et personnalisées, en exploitant les fonctionnalités offertes par l'EDI WinDev. Le langage W est connu pour sa simplicité et sa facilité d'utilisation, ce qui en fait un choix populaire parmi les développeurs utilisant WinDev.

HFSQL server :

HFSQL Server permet de stocker, organiser et gérer efficacement les données utilisées dans une application. Il prend en charge les fonctionnalités de base d'un SGBDR telles que la création de tables, l'indexation, les requêtes SQL, les transactions, la gestion des utilisateurs et des droits d'accès, etc.



L'une des caractéristiques clés de HFSQL Server est sa capacité à fonctionner en mode client-serveur, ce qui signifie que plusieurs utilisateurs peuvent accéder simultanément à la base de données à partir de différents postes de travail. Il offre également des mécanismes de sauvegarde et de restauration pour assurer la sécurité des données.

HFSQL Server est largement utilisé dans le développement d'applications métier, notamment dans les secteurs de la gestion commerciale, de la gestion des ressources humaines, de la finance, de la logistique, de la santé, etc. Il offre une intégration étroite avec les outils de développement de PC SOFT, ce qui facilite la création d'applications robustes et évolutives.

Star UML :

Star UML est un logiciel de modélisation UML (Unified Modeling Language) largement utilisé par les développeurs et les concepteurs de logiciels pour créer des diagrammes et des modèles de conception. Il fournit un environnement graphique convivial pour la création de modèles UML, ce qui facilite la communication et la



Compréhension des différentes parties prenantes impliquées dans le développement de logiciels.

Star UML offre un large éventail de fonctionnalités pour la création de diagrammes UML, notamment des diagrammes de cas d'utilisation, des diagrammes de classes, des diagrammes de séquence, des diagrammes d'activités, des diagrammes de composants, des diagrammes de déploiement, etc. Il permet aux utilisateurs de créer des modèles détaillés en utilisant les notations et les concepts standardisés fournis par UML.

En utilisant Star UML, les développeurs peuvent représenter visuellement la structure d'un système logiciel, les interactions entre les différents composants, les flux de données, les relations d'héritage, les dépendances, etc. Les modèles créés avec Star UML peuvent être

artagés avec d'autres membres de l'équipe de développement, ce qui facilite la collaboration et la compréhension commune des spécifications du système.

Canva :

Canva est une plateforme de conception graphique en ligne qui permet aux utilisateurs de créer facilement des designs et des visuels de haute qualité.



Il offre une large gamme d'outils et de fonctionnalités conviviales, ce qui en fait un outil populaire pour les professionnels du marketing, les créateurs de contenu, les entrepreneurs et les particuliers.

Canva offre également des fonctionnalités de collaboration en temps réel, ce qui permet à plusieurs utilisateurs de travailler sur un projet ensemble, en partageant et en modifiant les designs. Il est possible de télécharger vos propres images et de les intégrer dans vos créations, ou d'utiliser les images et les illustrations fournies par Canva.

4.4. Présentation interfaces graphique de l'application :

Après avoir justifié le choix des différents outils qui sont utilisés dans la réalisation de notre projet, nous détaillons ci-dessous quelques interfaces de notre application mobile.

4.4.1. Configuration server :

Cette page permet d'entrer et de sauvegarder le serveur en cours d'utilisations.

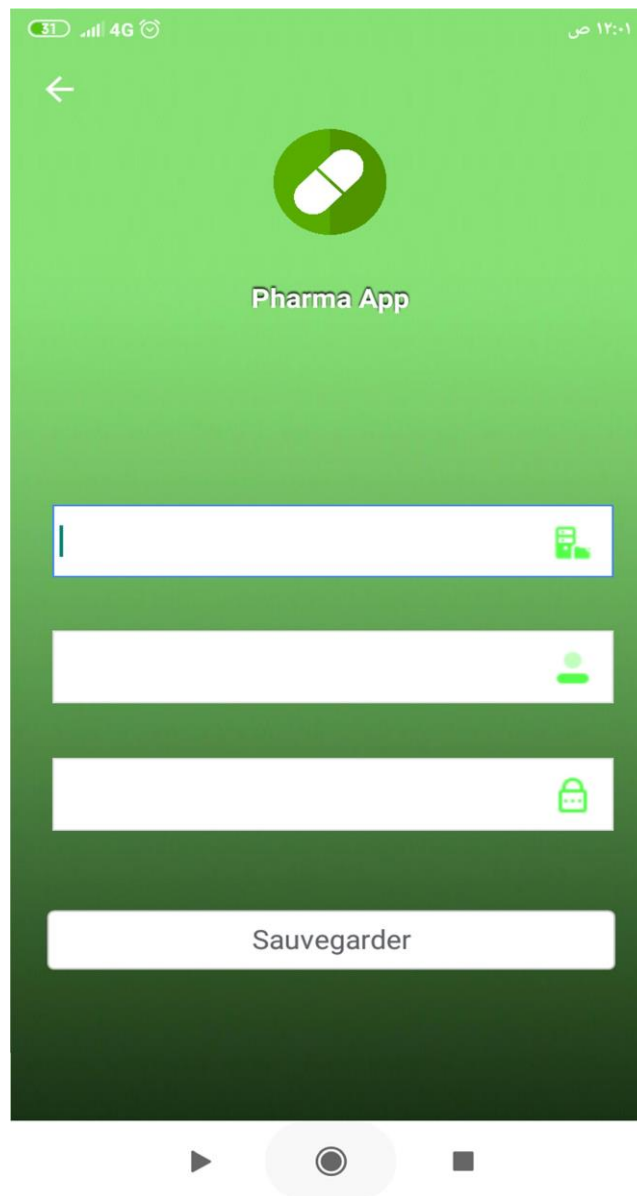


Figure4.1 Page configuration server

4.4.2. Inscription:

Cette page permet à toutes les pharmacies hospitalières d'inscrire dans la base de données pour pouvoir, par la suite, échanger les informations médicales et suivre les médicaments.

Pharmacie

Mot de passe

Adresse

Ajouter

Figure4.2 Page inscription

4.4.3. Authentification :

Cette page permet au les pharmacies hospitalières d'insérer le nome de pharmacie et le mot de passe après avoir fait la procédure d'inscription.

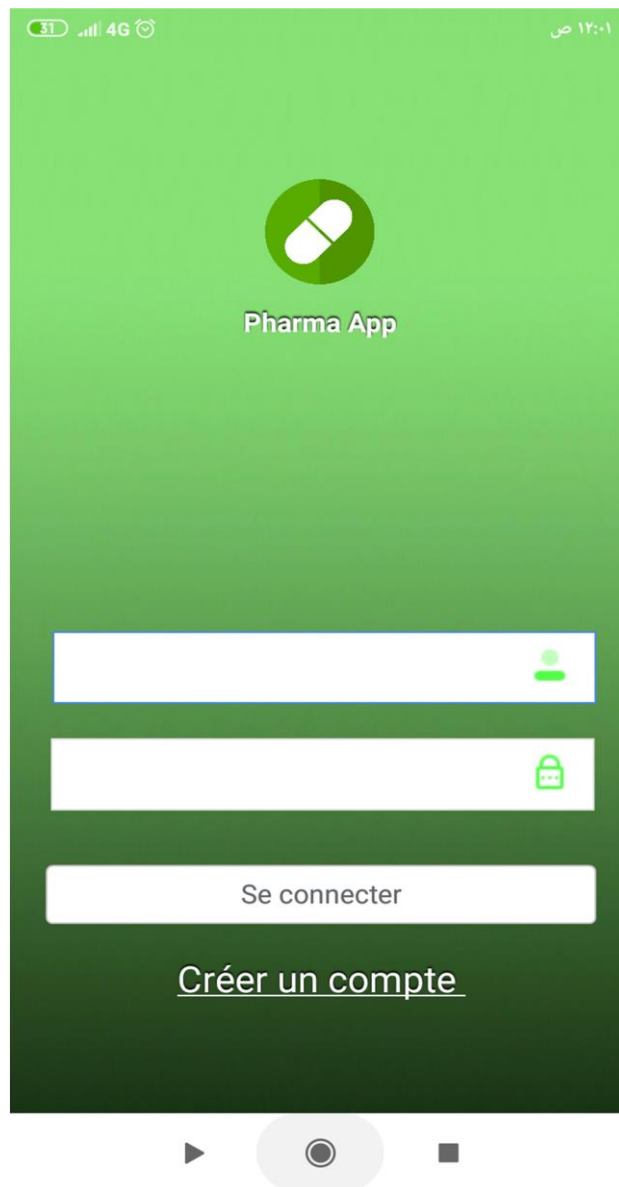


Figure4.3 Page authentication.

4.4.4. Page d'accueil:

Cette page présente les différents services fournis par l'application.

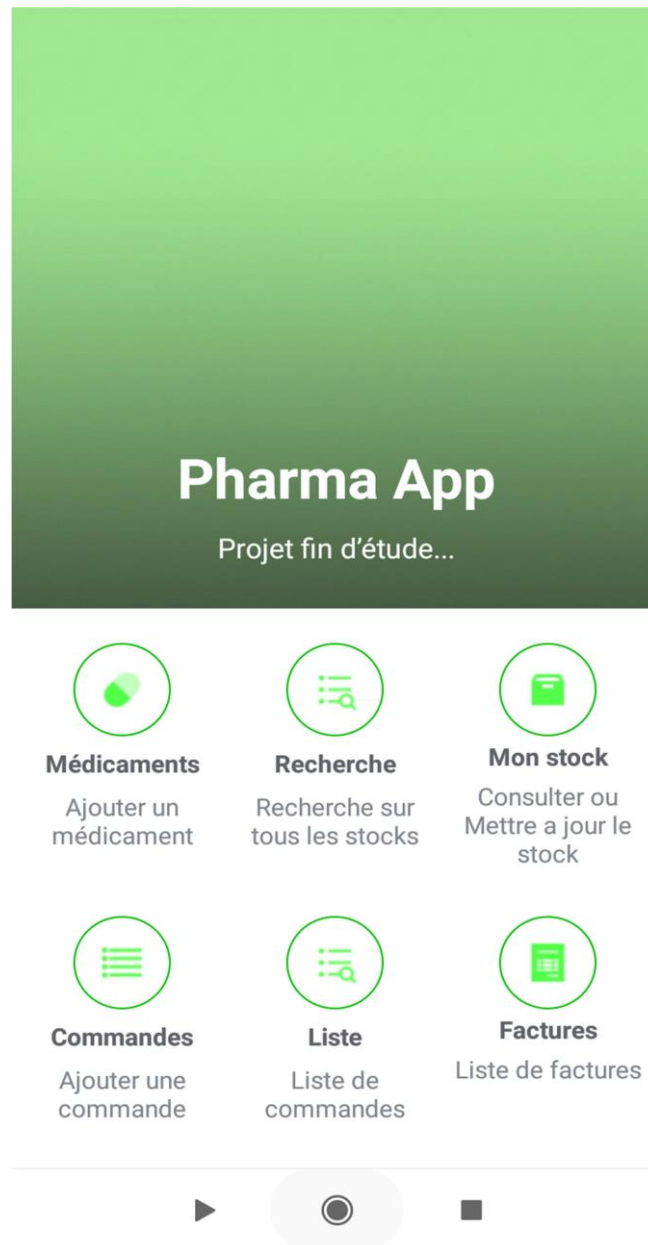


Figure4.4 Page d'accueil

4.4.5. Page médicaments:

Cette page permet d'ajouter des médicaments.

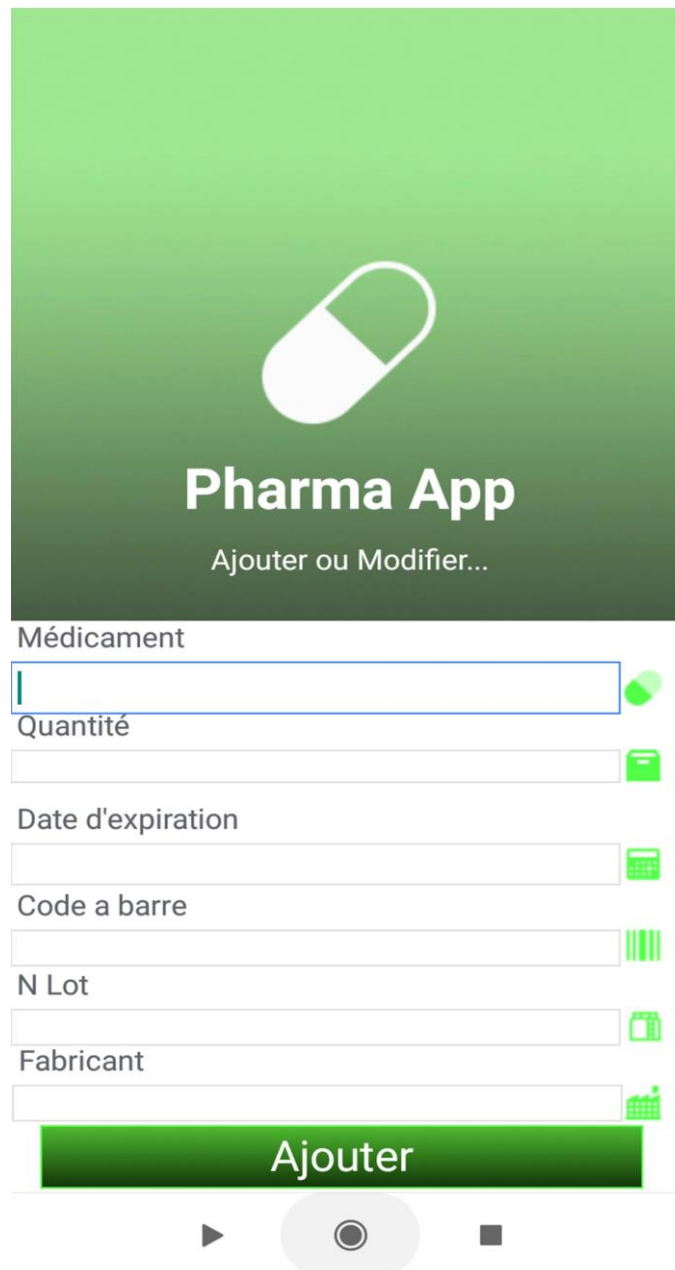


Figure4.5 Page médicaments.

4.4.6. Page recherche:

Cette page permet de recherche sur tous les stocks avec (le nom de médicament, pharmacie, code a bar, N lot, fabricant).

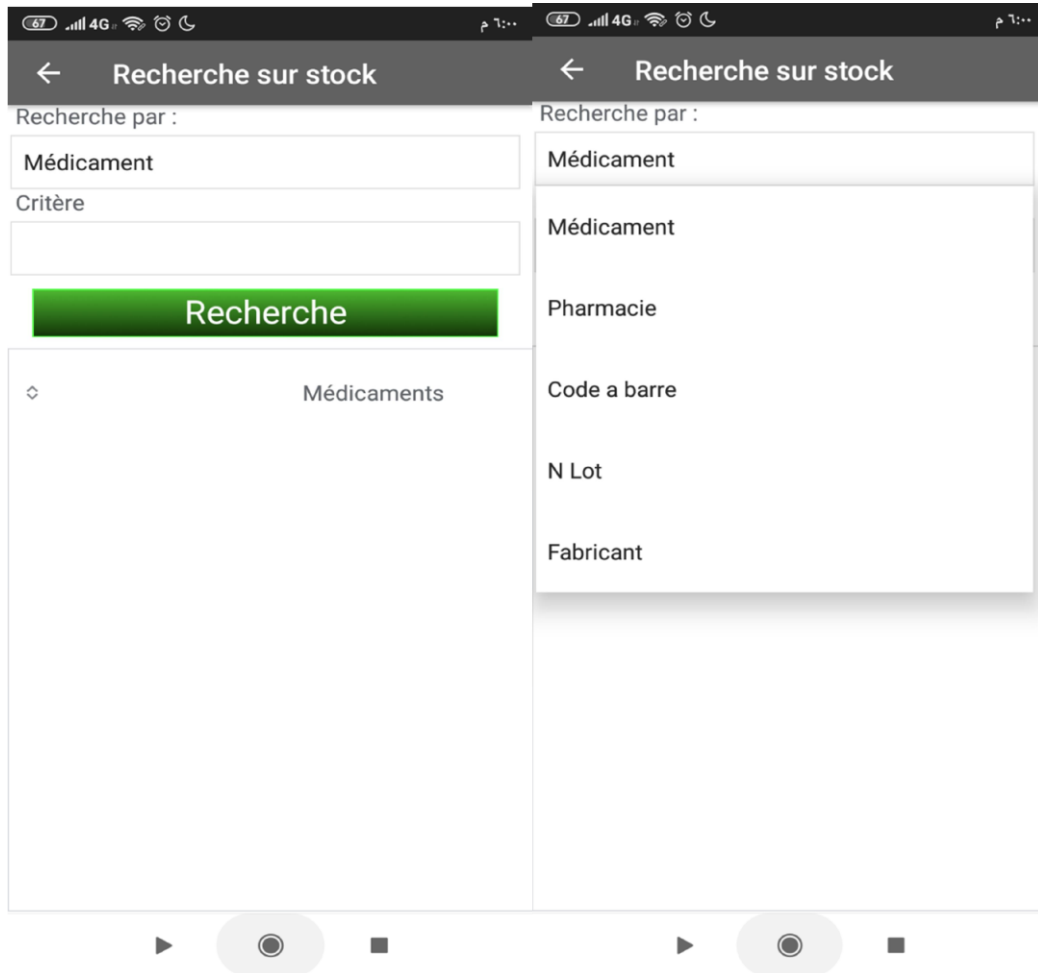


Figure4.6 Page recherche.

4.4.7. Page de stock:

Cette page et le stock de médicaments.

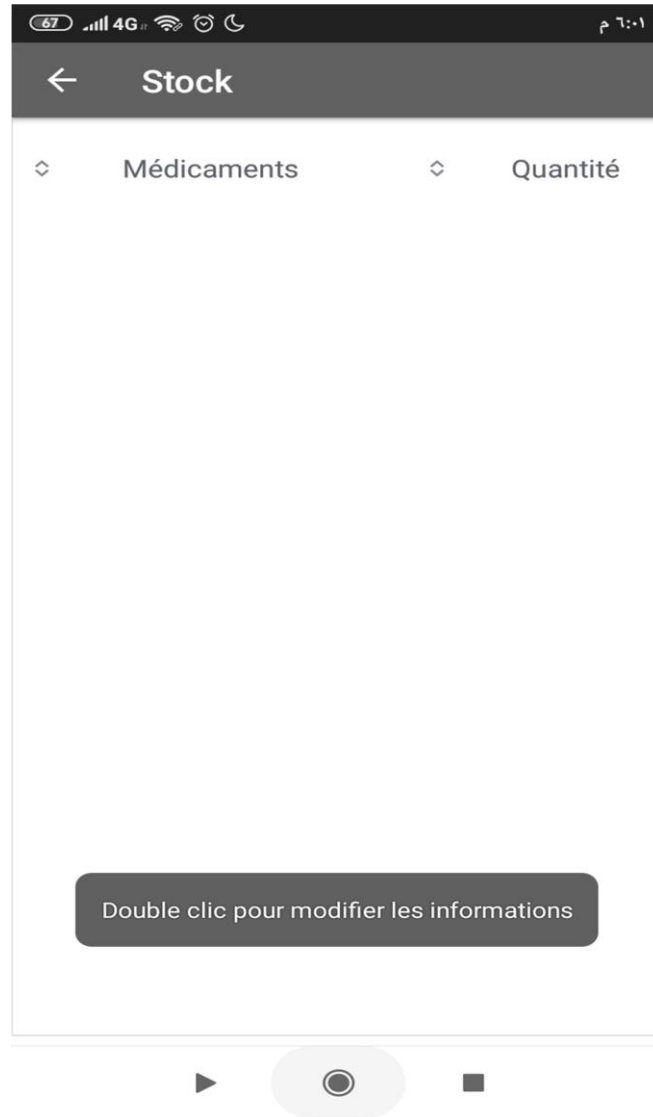


Figure4.7 Page de stock.

4.4.8. Page des commandes:

Cette page permet d'ajouter une commande.

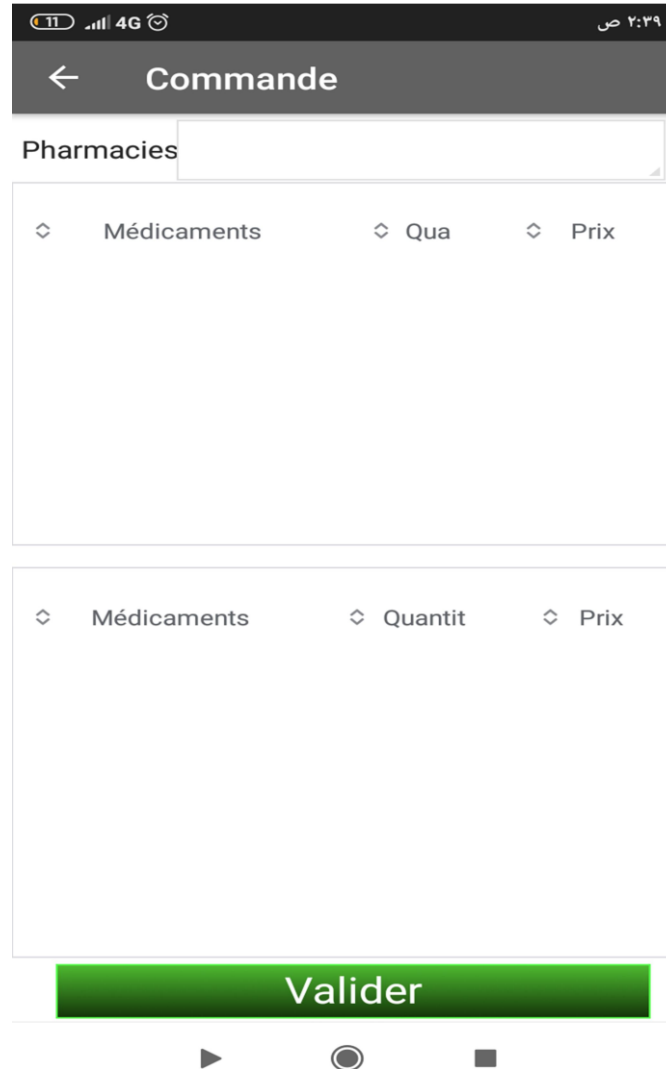


Figure4.8 Page des commandes.

4.4.9. Page de liste :

Dans cette fenêtre, vous trouvez les commandes que vous avez créées et les commandes qui vous ont été soumises.

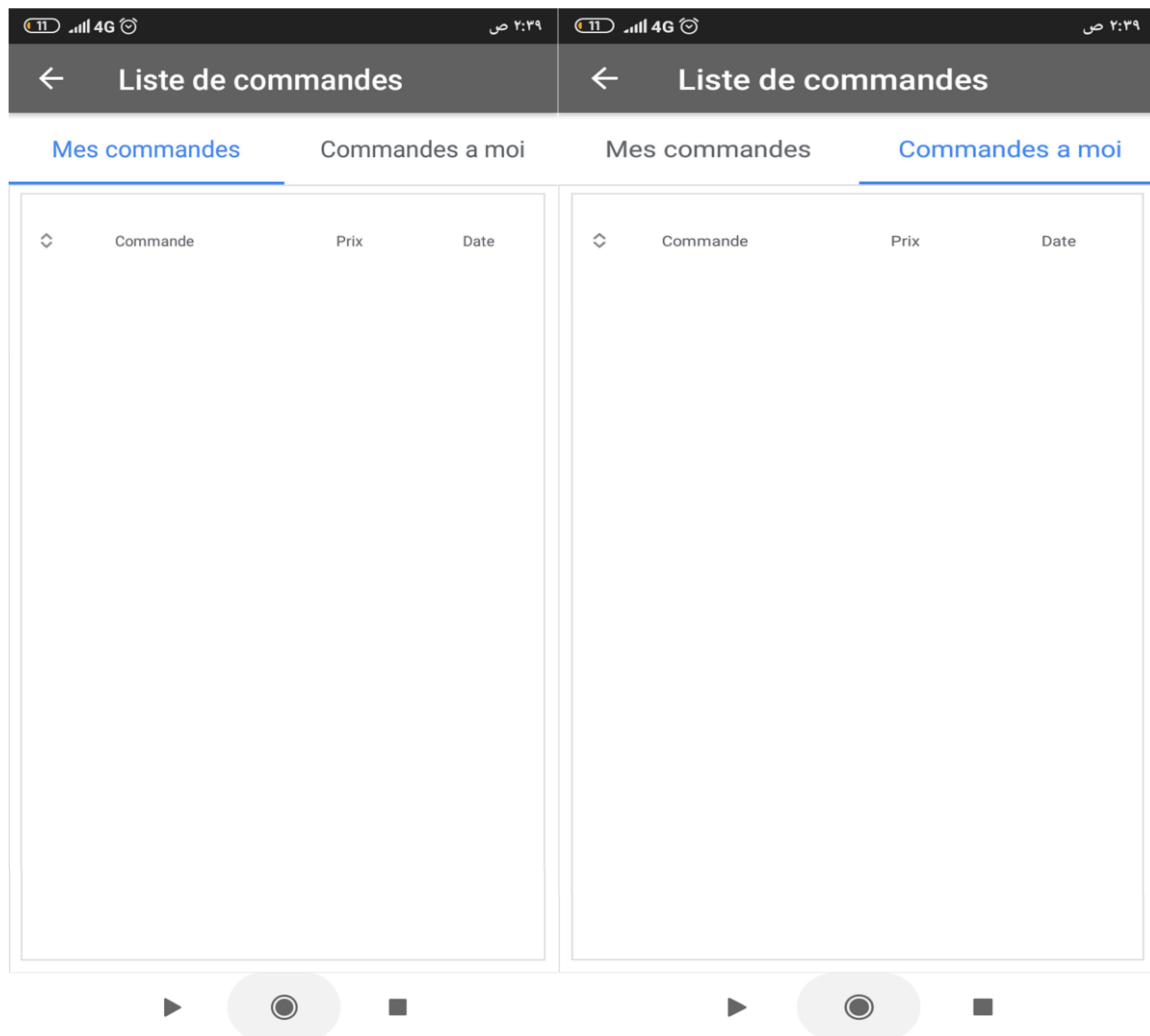


Figure4.9 Page de liste.

4.4.10. Page factures:

Dans cette fenêtre, vous trouvez les factures que vous avez créées et les factures qui vous ont été soumises.

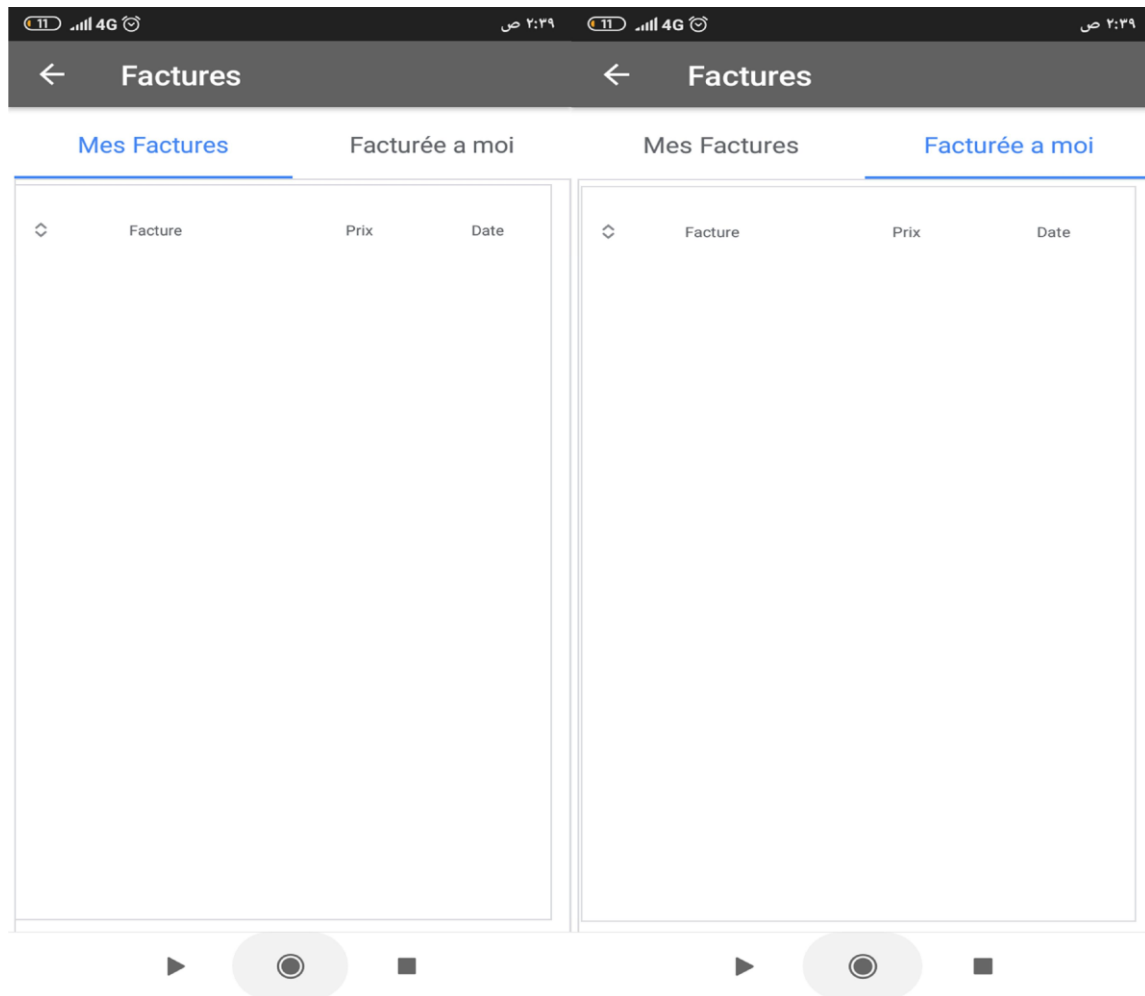


Figure4.10 Page factures.

4.5. Conclusion :

Dans ce chapitre nous avons d’écrit d’une façon détaillée les composants essentiels de l’environnement de notre système, les outils d’implémentation et de mise en œuvre ainsi que les différentes interfaces de notre application mobile.

Conclusion

générale

Conclusion générale

Au cours de ce travail, nous avons présenté les différentes étapes ayant conduit à la mise en œuvre d'une application mobile pour la gestion de l'échange de médicaments entre pharmacies hospitalières.

Nous avons commencé par présenter les technologies mobiles, les réseaux et le type de la solution envisagée. Nous avons ensuite présenté les concepts liés au fonctionnement des pharmacies hospitalières puis nous avons discuté le problème rencontré par ces pharmacies hospitalières avant de fixer les objectifs et présenter la solution développée.

Nous avons utilisé la notation UML dans la capture des besoins de notre système. En ce qui concerne l'analyse des besoins, nous nous avons servi des diagrammes de séquence en parallèle avec le réseau de Pétri. La conception de notre application mobile est illustré à travers différents diagrammes, convenant les aspects fonctionnels, dynamique et statiques de tous le développement.

Enfin pour réaliser l'application, nous avons utilisé « HFSQL » pour implémenter la base de données et le langage « WLANGAGE » via le Framework WinDev pour l'application mobile.

Ce projet a fait l'objet d'une expérience intéressante, très bénéfique pour nous. En effet, il nous a permis d'enrichir nos connaissances sur le domaine de la conception et de programmation.

Bibliographies et Webographie

Webographie

- [1] : <http://www.wikipédia.net>
- [2] : www.CommentçaMarche.net
- [3] : <https://www.techopedia.com>
- [4] : <https://www.supinfo.com>
- [5] : <https://www.taktilcommunication.com>
- [6] : <http://www.contenus-en-ligne.com>
- [13] : <https://theses-algerie.com/1915672692349149/memoire-de-master/centre-universitaire-abdel-hafid-boussouf-mila/application-mobile-pour-la-recherche-des-pharmacies-au-niveau-national>
- [14] : <https://www.memoireonline.com/03/19/10645/Conception-et-realisation-d-une-application-de-gestion-des-stocks-des-produits-pharmaceutiques.html>
- [15] : <https://ch-bassindethau.fr/structures/pharmacies-2/>
- [16] : <https://pharmed.datapharma.ch/la-pharmacie-hospitaliere/>
- [17] : <https://www.opq.org/protection-du-public/votre-pharmacien-et-vous/role-du-pharmacien/pharmacien-en-etablissement-de-sante/>
- [18] : <https://www.pharmashopi.com/pharmacie-hospitaliere-pxl-942.html>
- [19] <https://www.nucleodoconhecimento.com.br/sante/pharmacies-hospitalieres>
- [20] : <https://medicalguidelines.msf.org/fr/viewport/EssDr/francais/organisation-et-gestion-d-une-pharmacie-16689290.html>
- [21] : <https://elearn.univ-tlemcen.dz/mod/page/view.php?id=36941>
- [22] : <https://www.cismef.org/page/mesh-descripteur/pharmacie-hospitaliere>
- [23] : <https://www.ibm.com/docs/fr/rsar/9.5?topic=diagrams-class>
- [24] : <https://www.ibm.com/docs/fr/rsm/7.5.0?topic=uml-sequence-diagrams>
- [25] : <https://www.ibm.com/docs/fr/rational-sof-arch/9.5?topic=diagrams-use-case>
- [26] : <https://www.24hdz.com/medwak-une-plateforme-dechange-de-medicament-lancee-par-une-etudiante/>

[27] : <https://trocmed.com/>

[28] : <https://www.techopital.com/story?ID=3224>

[29] : <https://www.google.com/search?q=windev&oq=windev&aqs=chrom>

[30] : <https://pcsoft.fr/hfsq/presentation-de-hfsq.html>

[31] : <https://staruml.sourceforge.net/v1/about.PHP>

[32] : <https://doc.pcsoft.fr/?9000196&name=wlangagewlangage>

[32] : https://doc.pcsoft.fr/?1410086964&name=concepts_windev

Bibliographies

- [7] : Djouama Zohir. Conception et réalisation d'une compétition scientifique sous Android. Mémoire de fin d'étude pour l'obtention du diplôme de Master en Informatique, Ingénieur des Systèmes Distribués (I.S.D). Skikda : Université 20 Août -1955-SKIKDA Faculté des Sciences.
- [8] : Hamimed Imen. Application mobile guide de voyage. Mémoire de fin d'étude pour l'obtention du diplôme de Master en Informatique, Ingénierie Systèmes distribuée. Skikda : Université 20 Août -1955-SKIKDA Faculté des Sciences.
- [9] : G.picard SMA/G2/ENS Mines, définition d'UP Saint-Etienne, 2009. .
- [10] : P. Roques, UML2 par la pratique, Eyrolles, 2006.
- [11] : Référence RAD Studio UML 2.0 2015.
- [12] : sylvain Cherrier Diagramme de classes UML 2015.
- [33] : BOUAGACHE Djamel Eddine, TAIRI Hassane. Développement d'une application mobile client/serveur dédiée à la logistique transitaire sur windev mobile cas d'étude : BMT (Bejaia Mediteranean Terminal). Mémoire de fin d'étude pour l'obtention du *diplôme de Master en informatique Ingénierie Administration et Sécurité des réseaux* : Université A/Mira de Béjaïa Faculté des Sciences.