

Ministère de l'Enseignement Supérieur, de la
Recherche scientifique
Université de Monastir

--*-**

Institut Supérieur d'Informatique et de Mathématiques de Monastir



Projet de Fin d'Etudes

En vue de l'obtention du Diplôme de Licence Appliquée en Informatique :
Systèmes informatiques et Génie logiciel

**Conception et développement d'une application Android
de pharmacie en ligne avec partie Back-office
(PharmaCity)**

Réalisé par
Hmida Zayneb & Mahfoudh Balkis

Sous la direction de

M. Karamti Walid
Mme. Sagaama Insaf
(ISIMM)

Mr. Ghawar Rami
Mme. Boukhris Marwa
(HelpSoft)

Année universitaire : 2016/2017

Dédicaces

A mes très chers parents

Pour tout l'amour dont vous m'avez entouré, pour tout ce que vous avez fait pour moi. Je ferai de mon mieux pour rester un sujet de fierté à vos yeux avec l'espoir de ne jamais vous décevoir. Que ce modeste travail, soit l'exaucement de vos veux tant formulés et de vos prières quotidiennes.

A mon adorable sœur Hadhami

Je ne peux exprimer à travers ses lignes tous mes sentiments d'amour et de tendresse envers toi. Puisse l'amour et la fraternité nous unissent à jamais. Je te souhaite la réussite dans ta vie, avec tout le bonheur qu'il faut pour te combler.

Ma chère sœur Imène et mon cher frère Abdallah

Vous occupez une place particulière dans mon cœur. Je vous dédie ce travail en vous souhaitant un avenir radieux, plein de bonheur et de succès.

A mes aimables amies Houyem, Zayneb, Amna

En souvenir de nos éclats de rire et des bons moments. En souvenir de tout ce qu'on a vécu ensemble. J'espère de tout mon cœur que notre amitié durera éternellement.

A toute ma famille

A tous mes amis

*Et A tous ceux qui m'aiment, A tous ceux que j'aime, Je dédie le fruit de mon projet
de fin d'études*

Balkis

Dédicaces

A ma très chère mère,

La plus douce et la plus tendre des mères. Aucune dédicace ne saurait exprimer ma reconnaissance, Mon indéfectible attachement et mon grand amour. Ce modeste travail n'est que la consécration de tes grands efforts, et tes immenses sacrifices. Que Dieu vous préserve et vous accorde santé et bonheur.

A mon très cher Père,

Aucune dédicace ne saurait exprimer la profondeur de mon amour et de mon attachement. Vous êtes pour moi un sujet de fierté. Vous avez su m'inculper le sens de la responsabilité et de la confiance en soi, face aux difficultés de la vie.

A ma chère sœur Fatma, A mes chers frères Wael & Mohamed

Vous étiez toujours présents pour moi. Je suis fier de vous. Je vous souhaite un avenir radieux, plein de succès et de réussite.

A mes meilleurs amis,

A tous ceux qui ont contribué à ma formation, A tous ceux qui me connaissent de près ou de loin, Aucun mot ne pourra exprimer mon amour envers vous. Merci pour tout le soutien que vous m'avez offert, vous occuperez pour toujours une partie dans mon cœur

A toute ma famille

A tous mes amis

**Et A tous ceux qui m'aiment, A tous ceux que j'aime, Je dédie le fruit de
mon projet de fin d'études.**

Zayneb

Remerciement

C'est pour nous un plaisir autant qu'un devoir de remercier toutes les personnes qui ont pu contribuer de près ou de loin à l'élaboration de ce projet, qui nous ont aidé, nous ont soutenu et ont fait en sorte que ce travail ait lieu.

Ainsi, nous tenons à exprimer nos sentiments de remerciement et de gratitude à nos encadrants à l'ISIMM Mr Walid Karamti et Mme Insaf Sagaama, pour les conseils qu'ils nous ont prodigués, leur judicieux encadrement et leur assistance pour la rédaction de ce rapport.

Nos remerciements les plus sincères vont à nos tuteurs à Help Soft Mr Rami Ghaouar et Mme Marwa Boukhris, pour leurs conseils directifs et formations qui nous ont été d'un grand aide à l'aboutissement de ce projet. Nous sommes très sensibles à la confiance que vous nous avez témoignée en nous confiant la réalisation de ce travail.

Que tous les membres du jury retrouvent ici l'expression de notre reconnaissance pour avoir accepté d'évaluer notre travail. Nous tenons à vous exprimer notre admiration pour vos très hautes professionnelles et l'étendue de vos connaissances.

Enfin, nos remerciements vont au corps professoral de l'institut supérieur d'Informatique et de mathématique de Monastir pour la formation qu'il nous a prodiguée.

Résumé

Dans le cadre de notre projet de fin d'études à l'Institut Supérieur d'Informatique et de Mathématique de Monastir(ISIMM) en vue de l'obtention du Diplôme de Licence Appliquée en Informatique, nous avons effectué notre stage au sein de la société HelpSoft.

Ainsi, le présent document constitue la synthèse de notre travail dans le cadre de ce projet qui a pour objectif de concevoir et de développer une partie client et une partie Back-office de l'application mobile de pharmacie en ligne (PharmaCity), permettant à l'utilisateur d'accéder aux services offerts par une pharmacie sans aller à une.

Nous avons étudié en premier lieu les applications existantes ainsi que les concepts et les outils sur lesquels se base notre travail. En second lieu, nous avons réparti notre travail sur des phases dans lesquels nous avons capturé les besoins des utilisateurs et conçu l'application avant de passer à la phase de développement.

En ce qui concerne la mise en œuvre de l'application, nous avons utilisé l'Android pour réaliser l'application mobile, ainsi que le PHP pour développer la partie de l'administrateur qui est le pharmacien dans notre cas (Back-office).

A travers ce document, nous allons décrire plus en détails la réalisation de chaque phase de ce projet.

Mots Clés : Back-Office, Android, PHP,

Table des matières

Introduction Générale	14
Chapitre1 : Cadre général	17
Introduction.....	18
1. Présentation du cadre du projet de fin d'études	18
1.1. Les objectifs de projet	20
2. Etude de l'existant.....	20
a) Pharma Express	20
b) Doctipharma.....	21
c) Ma Pharmacie Mobile	22
d) Maladies Dictionnaire Offline	23
2.1. Critique de l'existant	23
2.2. Solution proposée	24
3. Méthodologie de travail.....	26
3.1. Les méthodes existantes	26
3.2. Etude comparative	28
3.3. Choix du 2TUP	29
4. Planning prévisionnel	32
Conclusion	33
Chapitre2 : Capture et spécification des besoins.....	35
Introduction.....	36
1. Capture des besoins	36
1.1. Besoins fonctionnels	36
1.1.1. Partie client.....	36
1.1.2. Partie Back-office.....	37
1.2. Besoins non fonctionnels	38
1.3. Besoins techniques.....	39
1.3.1. Raison du choix du l'UML	39
1.3.2. Raison du choix du l'Android.....	39
1.3.3. Raison du choix du PHP	40
2. Spécification des besoins fonctionnels	40
2.1. Identifications des acteurs	40
a) Le client	40
b) Le super administrateur	40
c) Le sous administrateur	40

2.2.	Diagrammes de spécification	40
2.2.1.	Diagramme de cas d'utilisation	41
i.	Diagramme de cas d'utilisation global	41
ii.	Diagrammes de cas d'utilisation de la partie Back-office	42
a.	Diagramme de cas d'utilisation << ajouter administrateur>>.....	42
b.	Diagramme de cas d'utilisation << Gérer clients >>.....	43
c.	Diagramme du cas d'utilisation << Gérer médicaments >>.....	44
d.	Diagramme du cas d'utilisation << Gérer commandes >>	45
e.	Diagramme du cas d'utilisation << Gérer messages >>.....	46
f.	Diagramme du cas d'utilisation << Gérer Diagnostics >>.....	46
iii.	Diagrammes de cas d'utilisation partie client	47
a.	Diagramme de cas d'utilisation <<s'inscrire>>.....	47
b.	Diagramme du cas d'utilisation << Gérer Profil >>.....	47
c.	Diagramme du cas d'utilisation << Effectuer des commandes >>.....	48
d.	Diagramme du cas d'utilisation << Gérer des favoris >>.....	48
e.	Diagramme du cas d'utilisation << Effectuer de recherche >>	49
f.	Diagramme du cas d'utilisation << Consulter des diagnostics >>	50
g.	Diagramme du cas d'utilisation << Envoyer des messages >>	51
h.	Diagramme du cas d'utilisation << consulter les alertes>>.....	51
i.	Diagramme du cas d'utilisation << Consulter des statistiques >>.....	52
2.2.2.	Diagramme de séquence système	52
a)	Partie Back-office	52
i.	Diagramme de séquence système de cas d'utilisation « ajouter médicament »	52
b)	Partie client	54
i.	Diagramme de séquence de cas d'utilisation « s'authentifier».....	54
ii.	Diagramme de séquence de cas d'utilisation « commander un médicament »	56
iii.	Diagramme de séquence de cas d'utilisation « Consulter diagnostics »	58
3.	Les maquettes de la partie Back-office	60
3.1.	Page de connexion	60
3.2.	Page d'accueil	60
3.3.	Page d'ajout d'un administrateur	61
3.4.	Page de la modification d'un administrateur.....	61
3.5.	Page de la liste des administrateurs	62

3.6.	Page d'ajout d'un médicament	62
3.7.	Page de la modification d'un médicament	63
3.8.	Page de la liste des médicaments	63
3.9.	Page d'ajout des diagnostics	64
3.10.	Page de la liste des médicaments diagnostics	64
3.11.	Page de la liste des symptômes	65
3.12.	Page de la liste des clients.....	65
3.13.	Page de la liste des commandes	66
3.14.	Page d'une ligne de commande	66
	Chapitre3 : Conception	69
	Introduction.....	70
1.	Conception générale de l'application	70
1.1.	MVC	70
1.1.1.	MVC et la partie client.....	71
a)	Le modèle	71
b)	Le contrôleur	72
c)	La vue.....	73
1.1.2.	MVC et la partie Back office PHP.....	74
2.	Conception détaillée	75
2.1.	Les vues de Kruchten.....	75
2.2.	Conception détaillée de la Partie client	76
2.2.1.	Conception de cas d'utilisation « commander un médicament »	76
2.2.2.	Conception de cas d'utilisation « Consulter diagnostics ».....	80
2.2.3.	Vue logique.....	83
a.	Diagramme de classe générale de la partie mobile en MVC.....	84
i.	Explication des classes générées par défaut dans l'application pour la partie modèle.....	86
ii.	Explication des classes et méthodes de l'application de la partie modèle	86
iii.	Explication des classes pour la partie vue.....	89
iv.	Explication des classes pour la partie contrôleur	90
2.3.	Conception de la base des données.....	91
2.3.1.	Dictionnaire des données.....	91
2.3.2.	Modèle conceptuel des données	95
	Conclusion	96
	Chapitre4 : Réalisation	98

Introduction.....	99
1. Environnement de développement	99
1.1. Wamp Server.....	99
1.2. NotePad++.....	99
1.3. Android Studio.....	99
1.4. Genymotion.....	99
2. Architecture du site web	100
3. Composition des modules.....	100
4. Présentation de la solution mobile	103
4.1. Interface de lancement de l'application et de l'authentification	103
4.2. Interface d'ajout d'un compte utilisateur	105
4.3. Interface de menu de l'application	105
4.4. Interface de profil du client.....	106
4.5. Interface de la recherche des médicaments.....	107
4.6. Interface des symptômes	108
4.7. Interface des diagnostics.....	108
4.8. Interface de l'envoi d'un message.....	109
4.9. Interfaces de la boite de réception des messages	110
4.10. Interface du panier des commandes.....	111
4.11. Interfaces des statistiques des commandes des clients	113
4.12. Interface des médicaments favoris pour le client.....	113
4.13. Interface à propos de l'application	114
Conclusion	114
Conclusion Générale	116

Table des figures

Figure 1: Logo de Help Soft	18
Figure 2: Organigramme de Help Soft.....	19
Figure 3: Page d'accueil de l'application Pharma Express [1]	21
Figure 4: Interfaces de l'application Doctipharma [2]	22
Figure 5: Interfaces de l'application « Ma Pharmacie Mobile » [3].....	22
Figure 6: Interfaces de l'application Maladies Dictionnaire Offline [4]	23
Figure 7: Solution proposée.....	25
Figure 8: cycle de développement 2TUP	31
Figure 9: Diagramme de cas d'utilisation global.....	42
Figure 10: Diagramme de cas d'utilisation global.....	43
Figure 11: Diagramme de cas d'utilisation << Gérer client>>.....	44
Figure 12: Diagramme de cas d'utilisation << Gérer médicament>>	45
Figure 13: Diagramme de cas d'utilisation << Gérer commandes >>	45
Figure 14: Diagramme de cas d'utilisation << Gérer messages >>	46
Figure 15: Diagramme de cas d'utilisation << Gérer Diagnostics >>	47
Figure 16: Diagramme de cas d'utilisation << s'inscrire>>	47
Figure 17: Diagramme de cas d'utilisation << Gérer Profil >>	48
Figure 18: Diagramme de cas d'utilisation << Effectuer des commandes >>	48
Figure 19: Diagramme de cas d'utilisation << Gérer favoris >>	49
Figure 20: Diagramme de cas d'utilisation << Effectuer recherche >>	50
Figure 21:Diagramme de cas d'utilisation << Consulter des diagnostics>>	50
Figure 22: Diagramme de cas d'utilisation << Envoyer des messages >>.....	51
Figure 23: Diagramme de cas d'utilisation << consulter les alertes >>.....	52
Figure 24: Diagramme de cas d'utilisation << Consulter des statistiques >>.....	52
Figure 25: Diagramme de séquence << ajouter médicament>>	53
Figure 26: Diagramme de séquence << s'authentifier>>	55
Figure 27: Diagramme de séquence << commander un médicament>>	57
Figure 28: Diagramme de séquence << consulter diagnostics>>	59
Figure 29:Page de connexion de la partie Back-office	60
Figure 30:Page d'accueil de la partie Back-office	61
Figure 31:Page d'ajout d'un administrateur de la partie Back-office	61
Figure 32:Page de la modification d'un administrateur de la partie Back-office	62
Figure 33:Page de la liste des administrateurs de la partie Back-office.....	62
Figure 34:Page d'ajout d'un médicament de la partie Back-office	63
Figure 35:Page de la modification d'un médicament de partie Back-office	63
Figure 36:Page de la liste des médicaments de la partie Back-office	64
Figure 37:Page d'ajout des diagnostics de la partie Back-office	64
Figure 38:Page de la liste des médicaments par diagnostic de la partie Back-office	65
Figure 39:Page de la liste des symptômes de la partie Back-office.....	65
Figure 40:Page de la liste des clients de la partie Back-office	66
Figure 41:Page de la liste des commandes de la partie Back-office.....	66
Figure 42:Page des lignes de commandes de la partie Back-office.....	67
Figure 43:Les interactions entre le modèle, la vue et le contrôleur	71
Figure 44: le modèle de l'application mobile	72
Figure 45: le contrôleur de l'application mobile	73

Figure 46: la vue de l'application mobile	74
Figure 47: Vue 4+1 (Kruchten) [7]	75
Figure 48: diagramme de classe de cas d'utilisation <<commander médicament>>	77
Figure 49:Vue de composants de cas d'utilisation <<commander médicament>>	78
Figure 50: Diagramme de séquence << commander un médicament>>	79
Figure 51: vue de déploiement de cas d'utilisation <<commander un médicament>>....	80
Figure 52: Diagramme de classe de cas d'utilisation << consulter diagnostics>>.....	81
Figure 54: Vue de composants de cas d'utilisation << consulter diagnostics>>.....	82
Figure 55: Diagramme de séquence << consulter diagnostics>>	82
Figure 56: Vue de déploiement de cas d'utilisation << consulter diagnostics>>	83
Figure 57: Diagramme de classes de la partie mobile en MVC	85
Figure 58: modèle conceptuel des données	95
Figure 59: Architecture du site	100
Figure 60:Composition des modules.....	102
Figure 61:Logo de la solution mobile <<PharmaCity>>	103
Figure 62:Interface de lancement de l'application.....	104
Figure 63:Interface de l'authentification	104
Figure 64:Interface d'inscription.....	105
Figure 65:Interface de menu de l'application.....	106
Figure 66:Interface de profil du client	106
Figure 67:Interface de recherche des médicaments	107
Figure 68:interface de détails d'un médicament	107
Figure 69:Interface des symptômes	108
Figure 70:Interface des diagnostics	109
Figure 71:Interface de l'envoi d'un message	110
Figure 72:Interface des messages.....	110
Figure 73:Interface de l'affichage d'un message.....	111
Figure 74:Interface de panier des commandes	112
Figure 75:Interface de l'envoi des ordonnances.....	112
Figure 76:Interface des statistiques	113
Figure 77:Interface des favoris	113
Figure 78:Interface à propos de l'application	114

Table des tableaux

Tableau 1:Comparaison des applications existantes.....	24
Tableau 2:Etude comparative des méthodologies	29
Tableau 3:Planning prévisionnel	33
Tableau 4:Diagramme de séquence <>ajouter médicament>.....	54
Tableau 5:Description du Scénario " s'authentifier "	56
Tableau 6:Description du Scénario " commander un médicament "	58
Tableau 7:Description du Scénario " consulter diagnostics "	59
Tableau 8:Tableau des fonctions de chaque classe	77
Tableau 9:les fonctions de chaque classe.....	81
Tableau 10:Tableau des fonctions de chaque classe	87
Tableau 11:Tableau de définition des méthodes de chaque classe	89
Tableau 12:Tableau de définition des classes de la partie vue	90
Tableau 13:Tableau de définition des classes de la partie contrôleur	90
Tableau 14:table Client.....	91
Tableau 15:table Commande	92
Tableau 16:table message	92
Tableau 17:table diagnostic	93
Tableau 18:table favoris	93
Tableau 19:table médicament	94
Tableau 20:table diag_medicament.....	94
Tableau 21:table ordonnance	94
Tableau 22:table pharmacien.....	94
Tableau 23:table symptome.....	95

Introduction Générale

Introduction Générale

Avec la croissance fulgurante que le segment des smartphones est en train de vivre, le marché s'organise pour fournir un ensemble de services étendant encore plus l'utilisation que nous faisons des smartphones au quotidien. Actuellement, le monde connaît une avance technologique considérable dans tous les secteurs et cela grâce à l'informatique qui est une science qui étudie les techniques du traitement automatique de l'information. Elle joue un rôle important dans le développement de l'entreprise et d'autres établissements.

Non seulement les capacités techniques des machines évoluent, mais la créativité des éditeurs d'applications va dans le même sens. Les tablettes et les smartphones deviennent des seconds écrans omniprésents. Aujourd'hui, de nombreux internautes restent sur les réseaux sociaux tout en regardant la télévision.

Les pharmacies font partie intégrante des établissements que l'informatique pourra beaucoup aider. En effet, l'importance des médicaments dans notre vie quotidienne 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. D'où, apparaît la notion de pharmacie en ligne.

La pharmacie en ligne représente un nouveau concept qui permet aux pharmaciens de vendre librement des produits parapharmaceutiques et médicamenteux sur un site internet officiel qui leur est propre.

Vu cet état de fait, notre projet fin d'études a pour objectif de concevoir et mettre en œuvre une application mobile 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.

L'objectif de notre projet « **application Android de pharmacie en ligne** » est de simplifier en premier lieu la vie du client qui devient capable d'accéder aux différents services offerts par une pharmacie sans devoir aller à une.

Introduction Générale



Chapitre 1 : Cadre général

Introduction

L'étude du cadre du projet est une phase primordiale au démarrage de tout projet. Elle permet une compréhension précise du contexte et une analyse complète de l'existant pour fixer les objectifs à atteindre.

Dans ce chapitre, nous exposons le contexte général de notre projet.

Nous présenterons d'abord le cadre dans lequel s'est déroulé le stage, à savoir la société qui a adopté ce projet. Ensuite, nous entamerons l'étude de l'existant à travers les solutions disponibles sur le marché. Nous clôturons par la proposition de notre solution en montrant les limites des applications et des méthodes existantes et améliorer leurs points forts.

1. Présentation du cadre du projet de fin d'études

Dans le cadre de notre projet, nous avons effectué notre stage au sein de la société Help-Soft qui nous a confié la conception et la réalisation d'une application mobile pour leur projet « PharmaCity ». Nous allons procéder à une présentation générale de l'organisme d'accueil.

➤ Présentation de la société Help-Soft



Figure 1: Logo de Help Soft

La société **Help Soft** est une société à responsabilité limitée (SARL) d'origine tunisienne fondée en janvier 2015 spécialisée dans la conception et le développement des applications spécifiques, sites WEB et d'applications mobiles Android, multimédia. Elle met à la disposition des clients une équipe d'experts pour désigner, réaliser, développer, héberger, maintenir et promouvoir les sites web, etc.

Nos collaborateurs sont pour la plupart des jeunes informaticiens disciplinés en terme de relation client-service et éprouvant une énergie et ambition illimité, dotés d'une large

expérience dans le domaine de développement en Dot-Net, J2EE & ANDROID, d'une solide compétence, que la société n'a pas manqué de la certifier.

➤ Organigramme

La figure 2 illustre l'organigramme de la société **Help Soft**. En fait, c'est une représentation schématique des liens fonctionnels, organisationnels et hiérarchiques de l'entreprise.

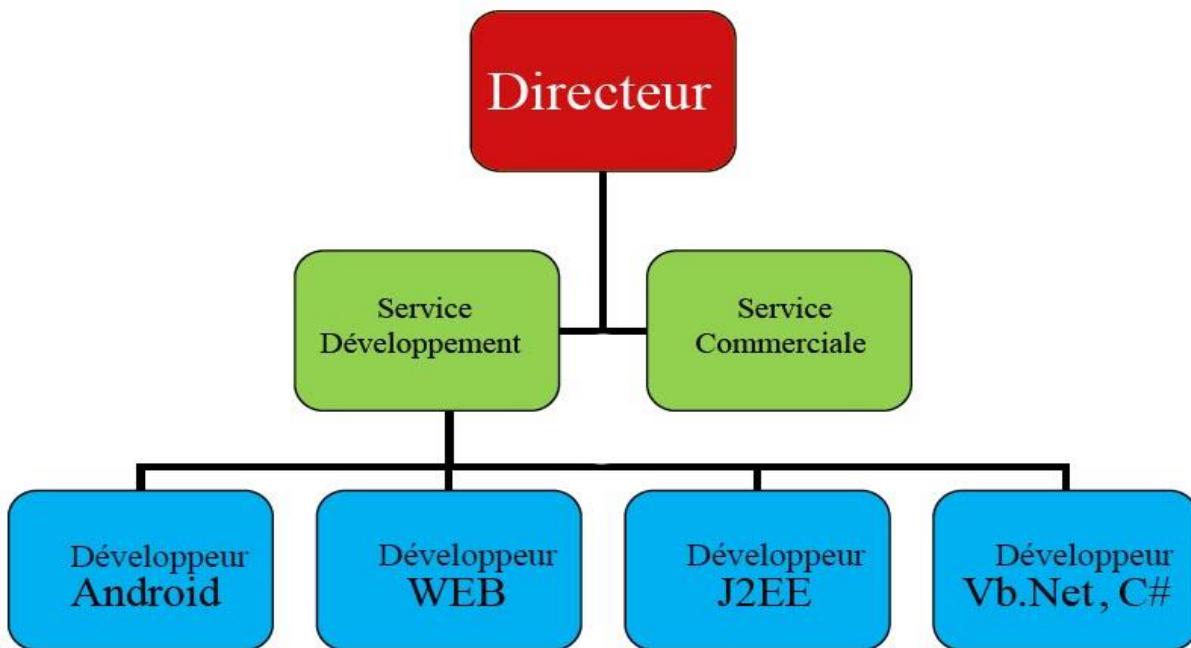


Figure 2: Organigramme de Help Soft

Problématique

Les pharmacies 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.

Nous avons remarqué 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.

Vu cet état de fait, notre projet a pour objectif de concevoir et développer une partie client et une partie Back office. Ces applications doivent être interactives, fiables, conviviales

et faciles à intégrer dans l'environnement de travail des pharmacies. Elles doivent également assurer la gestion des pharmacies à travers la suivie des ordonnances en prenant en considération les types des médicaments associés à chaque ordonnance et leur état de stock.

Notre application vise essentiellement à diminuer la complexité des traitements ainsi que le temps perdu lors de la gestion de stock.

1.1. Les objectifs de projet

L'utilisation actuelle des smartphones devienne une habitude que nous ne pouvons pas l'abandonner. C'est pourquoi nous voulons bien suivre la révolution de la technologie pour simplifier les tâches administratives et celles des services clients pour résoudre les problèmes des livraisons. Par conséquent, l'objectif de notre projet est de développer une partie client et une partie Back office de pharmacie en ligne basées sur les besoins d'utilisateur et la notification des alertes en temps réel. On prévoit de proposer une application compatible avec les systèmes d'exploitation IOS, Android et Windows phone.

2. Etude de l'existant

La réalisation de tout projet doit être précédée par une étude de l'existant qui détermine des observations en détectant les avantages et les inconvénients des applications actuelles et les besoins des utilisateurs.

Dans cette section, nous menons une étude basée sur les observations des différentes applications de pharmacie en ligne et lister leurs points forts puis leurs points faibles. Ces observations permettront de bien définir les grandes lignes de notre solution. Ainsi, le but de cette étude est de déduire les fonctionnalités à intégrer dans notre application.

a) Pharma Express

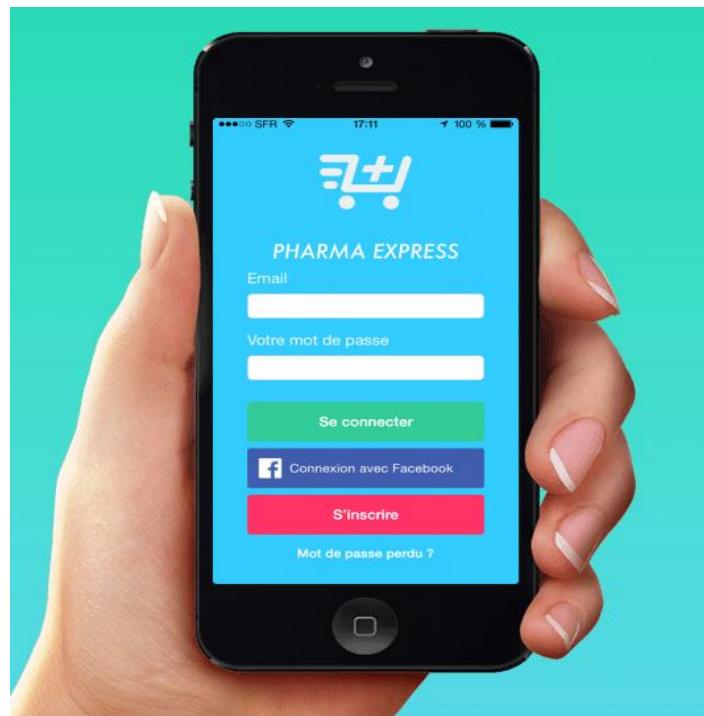


Figure 3: Page d'accueil de l'application Pharma Express [1]

La figure 3 présente l'application Pharma Express qui offre un service de livraison de médicaments sans ordonnance et produits de parapharmacie sur Paris et petite couronne.

L'application Pharma Express présente une interface complètement intuitive, permettant de sélectionner les médicaments et produits de parapharmacie en seulement quelques clics.

Pharma Express se positionne en tant que pionnier dans la recherche et le développement de l'ordonnance numérique en France.

b) Doctipharma



Figure 4: Interfaces de l'application Doctipharma [2]

La figure 4 illustre l'application Doctipharma qui permet à l'utilisateur d'envoyer la photo de son ordonnance à un pharmacien qui va lui préparer une prescription. Le client doit venir la chercher quand elle est prête dans la pharmacie, munie de l'ordonnance originale. Cette application permet de trouver la pharmacie ouverte la plus proche de client, même un dimanche ou à minuit et de recevoir des alertes pour des rappels de prises de médicaments.

c) Ma Pharmacie Mobile

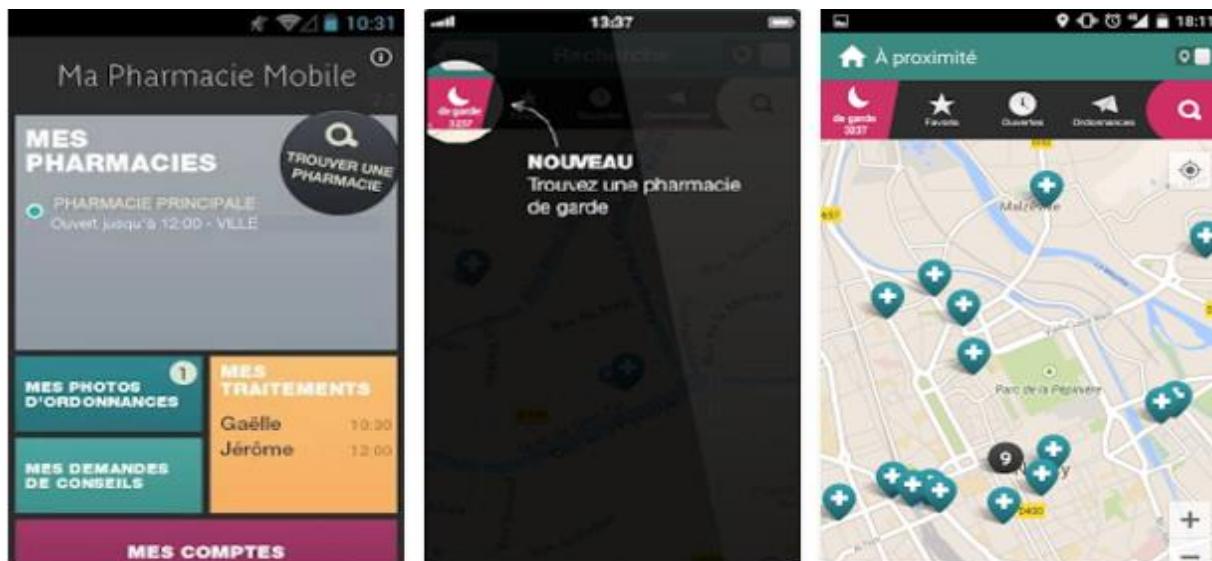


Figure 5: Interfaces de l'application « Ma Pharmacie Mobile » [3]

La figure 5 présente l'application « Ma pharmacie Mobile » qui est une application santé gratuite qui permet de localiser la pharmacie ouverte la plus proche de client dont les pharmacies de garde, trouver les horaires de n'importe quelle pharmacie, envoyer la photo d'ordonnance à pharmacien et recevoir des alertes pour le rappel de prises et de posologie.

d) Maladies Dictionnaire Offline

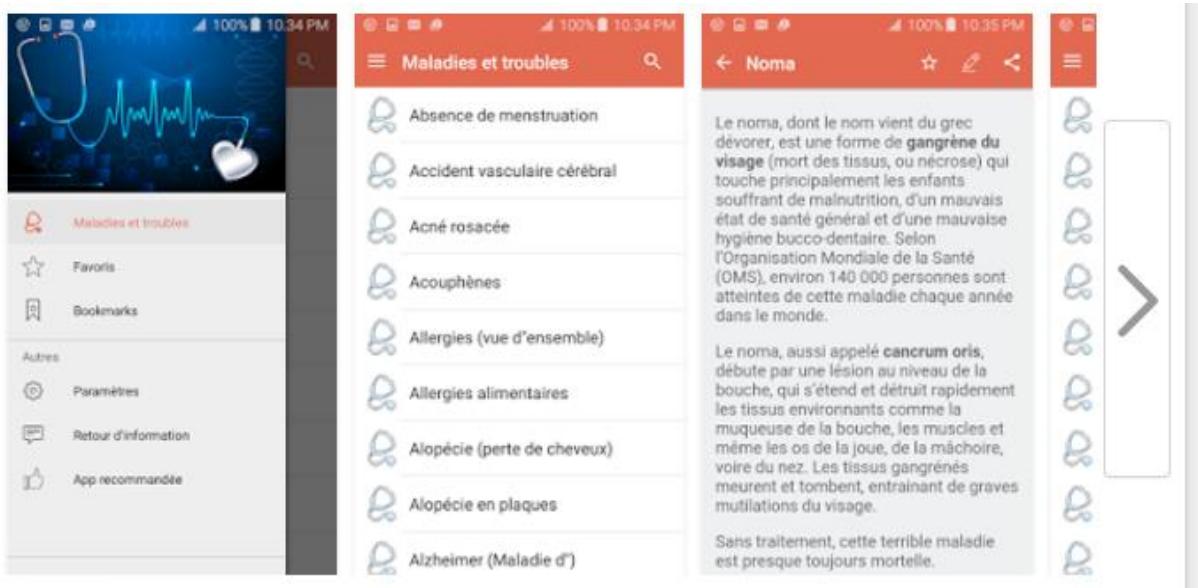


Figure 6: Interfaces de l'application Maladies Dictionnaire Offline [4]

La figure 6 présente l'application Maladies Dictionnaire hors ligne qui est OFFLINE complète et application gratuite liste des troubles et maladies médicales qui fournit toutes les informations sur les symptômes, le désordre et le traitement et beaucoup de la terminologie médicale contenant ces maladies et troubles .Le Dictionnaire peut agir comme un conseiller clinique pour l'autodiagnostic et peut également être utilisé pour rechercher les symptômes, les maladies et le traitement. Médicale téléchargement gratuit dictionnaire est comme un docteur GRATUIT à la maison pour les maladies courantes et les codes de traitement.

2.1. Critique de l'existant

La critique de l'existant doit mettre en évidence les imperfections des solutions existantes sur le marché. En s'intéressant sur les applications présentées précédemment, on constate qu'elles n'offrent ses services qu'aux clients habitants en France et elles ne sont pas

accessibles pour le reste du monde. Ainsi qu'elles ne permettent pas une interaction avec le client qui est incapable de donner un avis ou une proposition, etc.

Aussi, elles ne disposent pas de rappel de prise des médicaments. Concernant la dernière application, elle sert seulement à donner une idée sur les symptômes des maladies sans proposer une solution ou un diagnostic au client.

	Interface intuitive	interaction avec le client	rappel de prise des médicaments
Pharma Express	+	-	-
Doctipharma	+	-	-
Ma Pharmacie Mobile	+	-	-
Maladies Dictionnaire Offline	+	-	-

Tableau 1:Comparaison des applications existantes

Le tableau précédent illustre la comparaison des applications existantes au niveau de d'interface et d'interaction avec le client

2.2. Solution proposée

Dans ce projet, nous allons proposer une application composée d'une partie mobile et d'une partie Back office qui englobe tous les services offerts précédemment ainsi que des nouvelles fonctionnalités. Elle présente une interface complètement intuitive. De plus, elle permet à l'utilisateur final de sélectionner ses médicaments et ses produits en seulement quelques clics.

Notre application se positionne en tant que pionnier dans la recherche et le développement de l'ordonnance numérique. Elle permet d'envoyer la photo de l'ordonnance du client .Puis, elle le permet également de recevoir immédiatement une alerte sur son téléphone à propos de la disponibilité du médicament désiré. Enfin, il sera livré chez le client.

Cette application permet au client d'envoyer la photo d'ordonnance au pharmacien et d'effectuer une commande des médicaments existants dans cette ordonnance.

Le client peut recevoir des alertes pour ses rappels de prises de médicaments pour qu'il n'oublie pas.

En outre, notre application offre l'opportunité de consulter des statistiques sur les médicaments les plus vendus et plus efficaces à travers les avis des autres utilisateurs. Enfin, et comme nouvelle initiative, nous allons proposer un nouveau service qui donne au client des diagnostics approximatifs en se basant sur ses propres symptômes.

La figure 7 présente l'architecture de notre application.

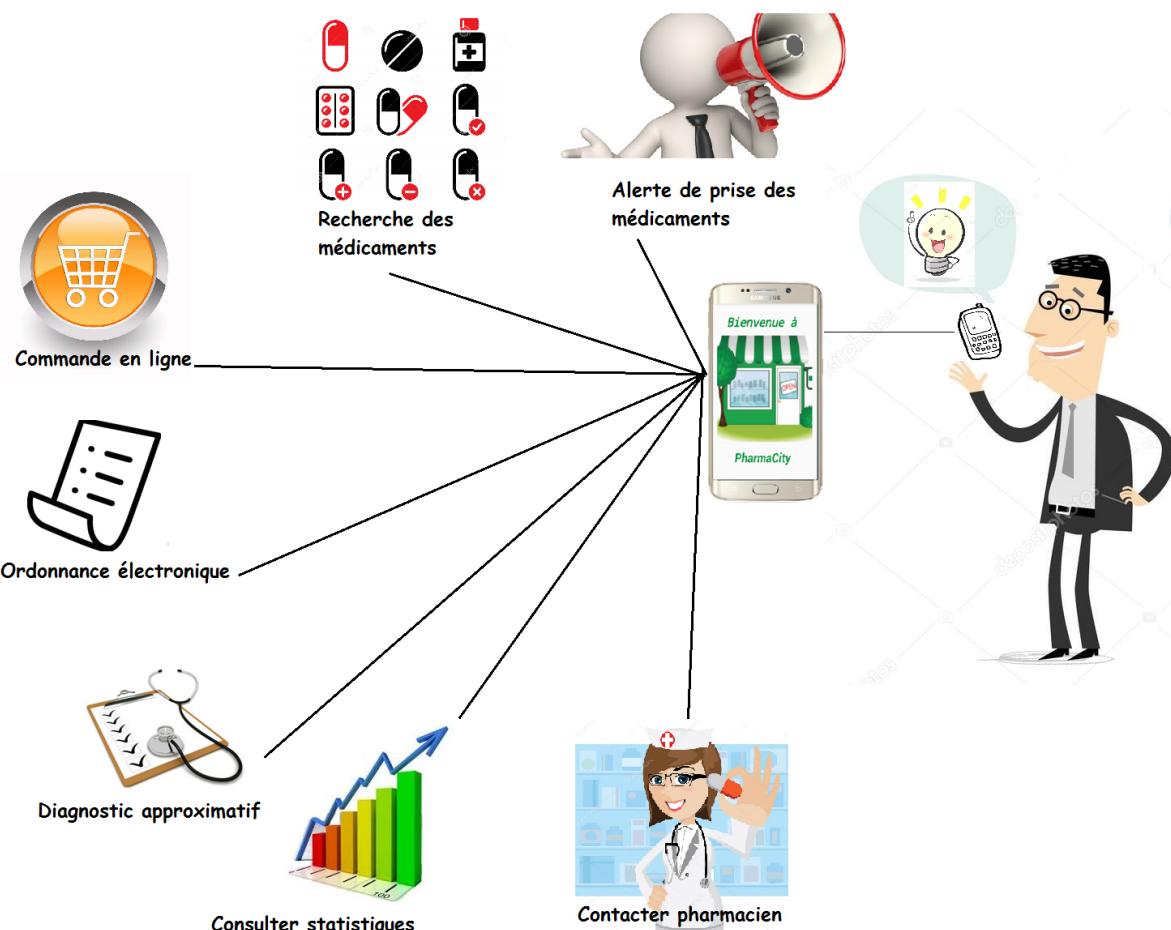


Figure 7: Solution proposée

Fonctionnalités

- Notre application devrait être compatible avec tous les systèmes d'exploitation mobile et compréhensible par tous les utilisateurs.
- Elle assure le gain de temps relatif au traitement de données et la simplification des tâches de l'administrateur et des utilisateurs.
- Elle est disponible pour tout le monde.
- Elle doit être simple à utiliser.

3. Méthodologie de travail

Un processus de développement logiciel est un ensemble d'activités qui conduit à la production d'un logiciel ou à l'évolution d'un système existant.

- Il n'existe pas de processus idéal.
- La plupart des entreprises adaptent les processus existants à leurs besoins.
- Ces besoins varient en fonction du domaine, des contraintes de qualité, des personnes impliquées.

« La qualité du processus de fabrication est garantie de la qualité du produit »

Les méthodologies de développement ont pour but d'aider à produire un produit de meilleure qualité :

- Logiciel mieux documenté.
- Plus facile côté utilisation.
- Plus facile à maintenir/à entretenir.
- Logiciel plus homogène.
- Aider à assurer le suivi des spécifications des utilisateurs.
- Aider le manager du projet à contrôler le projet.
- Réduire les coûts de développement.
- Encourager la communication entre les personnes.

La complexité croissante des systèmes informatiques a conduit les concepteurs à s'intéresser aux méthodes de développement. Ces dernières ont toujours essayé d'apporter un contrôle continu sur un projet tout au long de son processus de vie.

Plusieurs méthodes de développement sont disponibles .Mais, pour pouvoir choisir la plus convenable, il est nécessaire d'étudier les caractéristiques de chaque méthode et de les comparer.

3.1. Les méthodes existantes

La méthode Scrum

Dans le cas d'un développement d'application, il est nécessaire de la compléter avec des pratiques de qualité du logiciel. Scrum est un schéma d'organisation de développement de produits complexes.

La méthode ASD

Adaptive Software Development : Le développement logiciel adaptif.

ASD est une méthode agile de développement logiciel. Elle considère que l'imprévu est inévitable, et qu'on peut en tirer parti pour atteindre de meilleures solutions, qui n'auraient pas été a priori envisagées. Elle comporte 3 phases (parallèles et non linéaires) :

- Spéculation
- Collaboration
- Apprentissage

La méthode XP

La méthode XP (eXtremeProgramming) définit un certain nombre de bonnes pratiques permettant de développer un logiciel dans des conditions optimales en plaçant le client au cœur du processus de développement, en relation étroite avec le client. Le client (maîtrise d'ouvrage) pilote lui-même le projet, et ce de très près grâce à des cycles itératifs extrêmement courts (1 ou 2 semaines).

Les développeurs améliorent sans cesse la structure interne du logiciel pour que les évolutions y restent faciles et rapides.

La méthode UP

Méthode générique de développement logiciel construite sur UML. Le processus unifié est un processus de développement logiciels orientés objets, centré sur l'architecture, guidé par des cas d'utilisation et orienté vers la diminution des risques.

La méthode RUP

Le Rational UnifiedProcess (RUP) est l'une des plus célèbres implémentations de la méthode PU, livrée clés en main, permettant de donner un cadre au développement logiciel, répondant aux exigences fondamentales préconisées par les créateurs d'UML :

- Une méthode de développement doit être guidée par les besoins d'utilisateurs.
- Elle doit être centrée sur l'architecture logicielle.
- Elle doit être itérative et incrémentale.

La méthode 2TUP

« 2 Trucs UnifiedProcess », ou 2TUP, est un processus de développement logiciel qui implémente le Processus Unifié. Le 2TUP propose un cycle de développement en Y, qui dissocie les aspects techniques des aspects fonctionnels.

Il commence par une étude préliminaire qui consiste essentiellement à identifier les acteurs, qui vont interagir avec le système à construire, les messages qu'échangent les acteurs et le système, à produire le cahier des charges et à modéliser le contexte (le système est une boîte noire, entourée d'acteurs et reliés à lui. Sur l'axe qui lie un acteur au système, on met les messages échangés avec le sens). Le processus s'articule ensuite autour de trois phases essentielles [5]

- Une branche technique
- Une branche fonctionnelle
- Une phase de réalisation

3.2. Etude comparative

Après avoir défini les différentes méthodes de développement disponibles, nous avons constaté la difficulté du choix de méthode qui convient. Pour cela, nous allons présenter une étude comparative des méthodologies de développement les plus connues dans le tableau 1.

Méthodes	Forces	Faiblesses
Scrum	<ul style="list-style-type: none">- Léger- Simple à comprendre- Répond à des problèmes complexes et changeants	<ul style="list-style-type: none">- Scrum ne résout pas les problèmes de développement, il les rend visibles- Manque de pratique le contexte logiciel- Difficile à maîtriser
ASD	<ul style="list-style-type: none">- Souplesse au changement	<ul style="list-style-type: none">- ASD se concentre plus sur le concept et

	<ul style="list-style-type: none"> - Respect des délais - Implication de la maîtrise d'ouvrage 	la culture que la pratique logicielle.
XP	<ul style="list-style-type: none"> - Flxibilité aux changements. - Contrôle continu. - Simplicité des processus. 	<ul style="list-style-type: none"> - Très difficile à mettre en place avec une équipe virtuelle - Peu, voir pas, de documentation écrite. - Violation de responsabilité (pas de hiérarchie).
RUP	<ul style="list-style-type: none"> - Léger - Simple à comprendre - Répond à des problèmes complexes et changeants. 	<ul style="list-style-type: none"> - Difficile à maîtriser <p>Scrum ne résout pas les problèmes de développement, il les rend visibles</p> <ul style="list-style-type: none"> - Manque de pratique dans le contexte logiciel.
2TUP	<ul style="list-style-type: none"> - Itératif - Cible de projets de différentes tailles une large place à la technologie et à la gestion du risque. - Définit les profils des intervenants, les livrables, les plannings, les prototypes. 	<ul style="list-style-type: none"> - Plutôt superficiel sur les phases situées en amont et en aval du développement : capture des besoins, support, maintenance, gestion du changement, etc. - Ne propose pas de documents types.

Tableau 2:Etude comparative des méthodologies

3.3. Choix du 2TUP

La meilleure façon d'aborder un problème, c'est de s'y attaquer de front. C'est ce que propose le 2TUP en faisant une place à part entière à la technologie dans le processus de développement.

Et donc notre choix s'est porté sur la méthode 2TUP du fait de son approche nouvelle et originale.

Ce processus se base lui-même sur l'UP (UnifiedProcess) Processus Unifié qui est devenu un standard général réunissant les meilleures pratiques de développement. Cette méthode ne se base aucunement sur un processus linéaire mais bien, sur un développement itératif et incrémental.

« 2 Track » signifie littéralement que le processus suit deux chemins. Il s'agit des « chemins fonctionnels » et « d'architecture technique », qui correspondent aux deux axes de changement imposés au système d'information.

Le 2TUP propose un cycle de développement en Y, qui dissocie les aspects techniques des aspects fonctionnels. Le processus en Y s'articule autour de 3 phases

- Une branche technique.
- Une branche fonctionnelle.
- Une phase de réalisation (conception et développement).

La figure 8 présente cycle de développement 2TUP.

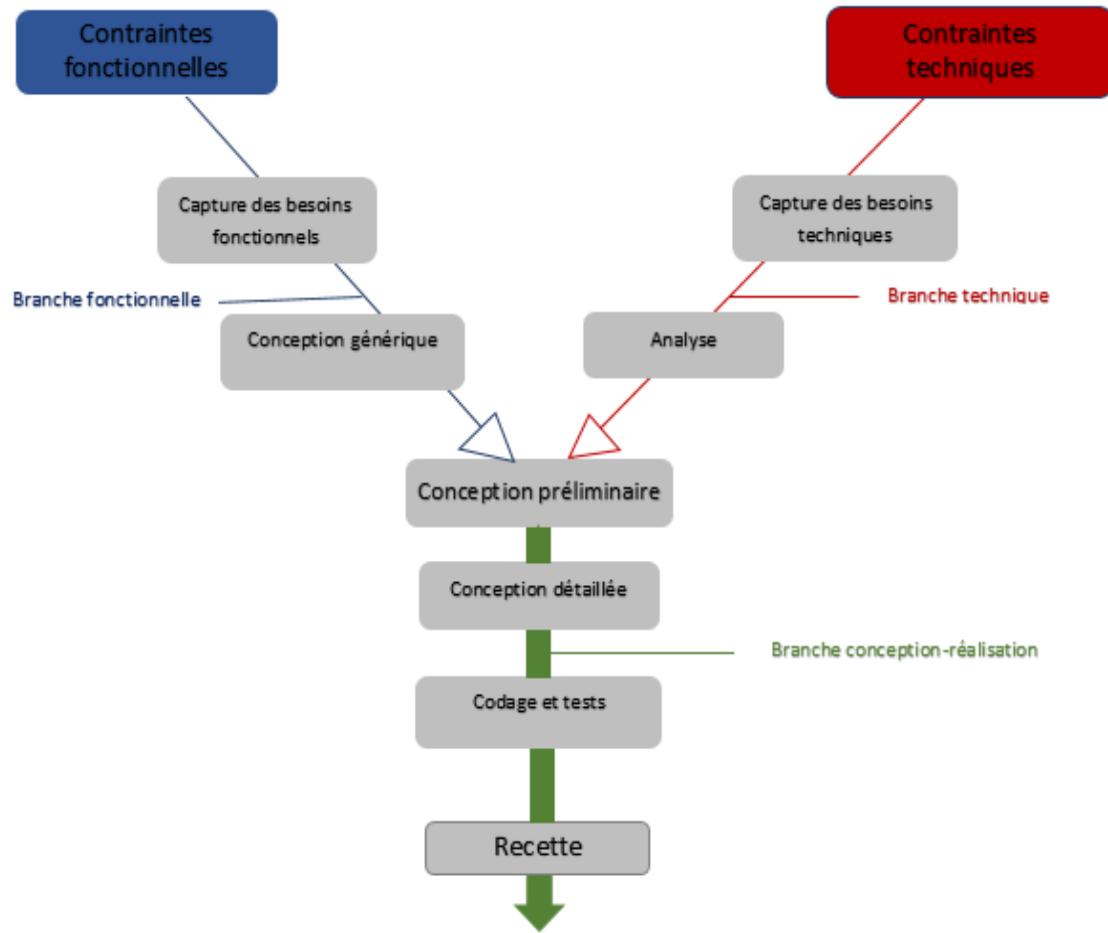


Figure 8: cycle de développement 2TUP

La branche gauche (fonctionnelle)

Elle capitalise la connaissance du métier de l'entreprise. Elle constitue généralement un investissement pour le moyen et le long terme. Les fonctions du système d'information sont en effet indépendantes des technologies utilisées.

Cette branche comporte les étapes suivantes

- **La capture des besoins fonctionnels** qui produit un modèle des besoins focalisé sur le métier des utilisateurs.
- **L'analyse** consiste à étudier précisément les spécifications fonctionnelles.

La branche droite (technique)

Elle capitalise un savoir-faire technique. Elle constitue un investissement pour le court et le moyen terme. Les techniques développées pour le système peuvent l'être en effet indépendamment des fonctions à réaliser. Cette branche comporte les étapes suivantes

- **La capture des besoins techniques** qui permet de définir le modèle d'analyse technique.
- **La conception générique** qui permet de définir les composants nécessaires à la construction de l'architecture technique.

La branche du milieu (conception-réalisation)

A l'issue des évolutions du modèle fonctionnel et de l'architecture technique, la réalisation du système consiste à fusionner les résultats des 2 branches. Cette fusion conduit à l'obtention d'un processus en forme de Y. Cette branche comporte les étapes suivantes

- **La conception préliminaire** est une étape qui permet de produire le modèle de conception système. Ce dernier organise le système en composants, délivrant les services techniques et fonctionnels. Ce modèle regroupe donc les informations de ces branches techniques et fonctionnelles.
- **La conception détaillée** permet d'étudier comment réaliser chaque composant. Cette étape produit le modèle de conception des composants.
- **Le codage** consiste à effectuer la production des composants et les tests des unités de code au fur et à mesure de leur réalisation.
- **La recette** consiste à valider les fonctionnalités du système développé.

4. Planning prévisionnel

Le tableau 3 montre le planning prévisionnel que nous avons suivi tout au long de notre projet.

	Février				Mars				Avril				Mai			
	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4
Etude de l'existant	■	■														
Documentation			■													
Conception				■	■	■	■									
Réalisation								■	■	■	■					

Rédaction de rapport																			
Test																			

Tableau 3:Planning prévisionnel

Conclusion

Au cours de ce chapitre, nous avons présenté le contexte du projet avant de citer et critiquer les méthodes et les applications qui sont encore utilisables malgré les points faibles qui les empêchent parfois de remplir leurs tâches convenablement.

Nous présentons dans le chapitre suivant un aperçu sur l'analyse et la spécification des besoins.



Chapitre 2 : Capture et spécification des besoins

Introduction

La spécification est la phase primordiale pour la conception des systèmes informatiques. Elle sert à identifier les acteurs réactifs du système et leur associer chacun l'ensemble d'actions avec lesquelles il intervient dans l'objectif de donner un résultat optimal et satisfaisant au client. Dans ce chapitre, nous commençons en premier lieu par une analyse des besoins auxquels doit répondre notre application. Nous enchaînons ensuite par une présentation de la vue dynamique du système à travers l'introduction des acteurs, les diagrammes de cas d'utilisation et les différents diagrammes de séquences relatifs à ces cas d'utilisations.

1. Capture des besoins

1.1. Besoins fonctionnels

Dans cette sous-section, nous allons exposer l'ensemble des besoins fonctionnels auxquels devrait répondre impérativement la partie client et la partie Back-office de notre application à développer.

1.1.1. Partie client

- Gestion des profils

Le client n'aura accès à son profil que lorsque l'administrateur lui accorde une suite des données propre à lui (login et mot de passe). Le profil de chaque client est unique, il contient ses favoris, ses commandes et ses informations personnels.

- Réalisation des commandes

L'utilisateur final doit créer un bon de commande correspondant à ses besoins.

- Gestion des favoris

Chaque client aura une liste des médicaments favoris qui contient les médicaments commandés souvent par lui.

- Effectuer la recherche sur le médicament souhaité

Cette opération permet au client de faire une recherche sur le médicament dont il a besoin.

- Consultation des diagnostics

En cas d'absence d'une ordonnance, ce module permet au client de trouver un médicament correspondant à ses symptômes.

- Envoi des messages

Cette fonction permettra au client d'envoyer un message à l'administrateur qui peut contenir un avis, une suggestion ou une interrogation concernant le fonctionnement de l'application.

- Consultation des statistiques

Cette fonction permettra de suivre les différentes statistiques possibles des médicaments commandés par d'autres clients.

1.1.2. Partie Back-office

- Gestion des clients

Ce module est réservé pour la gestion des utilisateurs qui vont exploiter notre application. Ce dernier nous permet de consulter, ajouter, modifier ou supprimer un client.

- Gestion des médicaments

Cette opération consiste à consulter la liste des médicaments existants dans le stock ainsi qu'elle permet au client d'ajouter, de modifier ou de supprimer un médicament.

- Gestion des commandes

Les commandes envoyées par le client seront gérées par l'administrateur (pharmacien) ayant accès à la liste des commandes.

- Gestion des messages

Ce module présente une communication directe entre le client et le pharmacien de façon que le message envoyé par l'utilisateur final soit traité par le pharmacien. Ce dernier envoie également une relance, une alerte ou encore une promotion par SMS.

- Gestion des diagnostics

Ce modèle permet à l'administrateur de proposer au client une suite des diagnostics approximatifs. Ce dernier indique ses symptômes à travers une liste proposée par la partie Back-office. Une fois le choix effectué, un ou plusieurs médicaments seront suggérés au client.

- Gestion des ordonnances

Cette opération permet de valider les bons des médicaments et de consulter la liste des ordonnances.

- Gestion de sécurité

Le Système permet de gérer les droits d'accès de chaque utilisateur ainsi les menus qui seront affichés selon le privilège.

1.2. Besoins non fonctionnels

Les besoins non fonctionnels décrivent toutes les contraintes auxquelles est soumis le système pour sa réalisation et son bon fonctionnement.

- Sécurité

Protection de la confidentialité en assurant la validité de l'identité de l'utilisateur. C'est ce que se fait par le moyen d'un mot de passe assurant le contrôle d'accès déterminé selon des priviléges.

- Maintenabilité et extensibilité

Le système doit être facile à utiliser et manipuler. De plus, il doit être maintenable et la description de données facile à mettre à jour. L'application doit fournir la possibilité d'intégration des nouvelles composantes.

➤ Ergonomie et souplesse

L'application doit offrir une interface conviviale et ergonomique exploitable par l'utilisateur en envisageant toutes les interactions possibles à l'écran du support tenu.

➤ Contraintes temps réels

L'application doit optimiser les traitements pour avoir un temps de génération de schéma raisonnable.

1.3. Besoins techniques

On souhaite utiliser une approche itérative et incrémentale, fondée sur le processus en Y. Nous avons affranchies le choix d'un certain nombre de technique clés pour ce projet.

- La modélisation objet avec **Unified Modeling Language**(**UML**).
- Android
- PHP

1.3.1. Raison du choix du l'UML

L'UML est un langage de modélisation objet. Il est apparu dans le monde du génie logiciel, dans le cadre de la « conception orientée objet ». Couramment utilisé dans les projets logiciels, il peut être appliqué à toutes sortes de systèmes ne se limitant pas au domaine informatique. Ainsi que l'UML permet de faire des simulations avant de construire un système et de générer automatiquement la partie logiciel d'un système. [6]

1.3.2. Raison du choix du l'Android

Pour répondre aux besoins du client et s'assurer du bon déroulement du notre projet et s'assurer que notre application est disponible pour autant de gens que possible, nous avons choisi de travailler sur Android, qui est conçue pour des appareils clients au sens large. Nullement restreinte aux téléphones, elle ouvre d'autres possibilités d'utilisation des tablettes, des ordinateurs portables, etc. [7]

1.3.3. Raison du choix du PHP

PHP est tout d'abord un langage de script interprété gratuit, Open Source et distribué sous une licence autorisant la modification et la redistribution.

Nous avons choisi de développer avec le langage PHP vu qu'il est supporté sur plusieurs systèmes d'exploitation.

Par conséquent, il sera alors très facile de déplacer une application écrite en PHP d'un serveur Windows d'origine vers un serveur Linux sans avoir à la modifier. [8]

2. Spécification des besoins fonctionnels

Nous allons modéliser et concevoir les différents traitements gérés par notre application en utilisant la méthodologie UML.

2.1. Identifications des acteurs

Le système interagit avec trois acteurs à savoir

a) Le client

C'est l'acteur principal auquel les services de notre système sont destinés. Il peut être soit un employé (identifié par son matricule), soit un dispensaire appartenant au service médical soit un client.

b) Le super administrateur

C'est l'acteur secondaire de notre système. Il joue le rôle de l'administrateur pour la partie Back-office. Il s'occupe à la fois de la partie commande, de la gestion des administrateurs, de la gestion de médicaments, de la gestion des clients et de la spécification des diagnostics approximatifs.

c) Le sous administrateur

Il permet de faire toutes les tâches du super administrateur sauf la gestion des administrateurs.

2.2. Diagrammes de spécification

2.2.1. Diagramme de cas d'utilisation

Le diagramme des cas d'utilisations met en évidence de quelle façon les acteurs utiliseront le logiciel ainsi que les utilisations principales du système.

i. Diagramme de cas d'utilisation global

Le diagramme global est illustré par la Figure9. Il permet d'identifier les possibilités d'interaction entre la partie Back-office et la partie client de notre système, ainsi qu'entre notre système et les acteurs (intervenants extérieurs au système). Il englobe tous les cas d'utilisation.

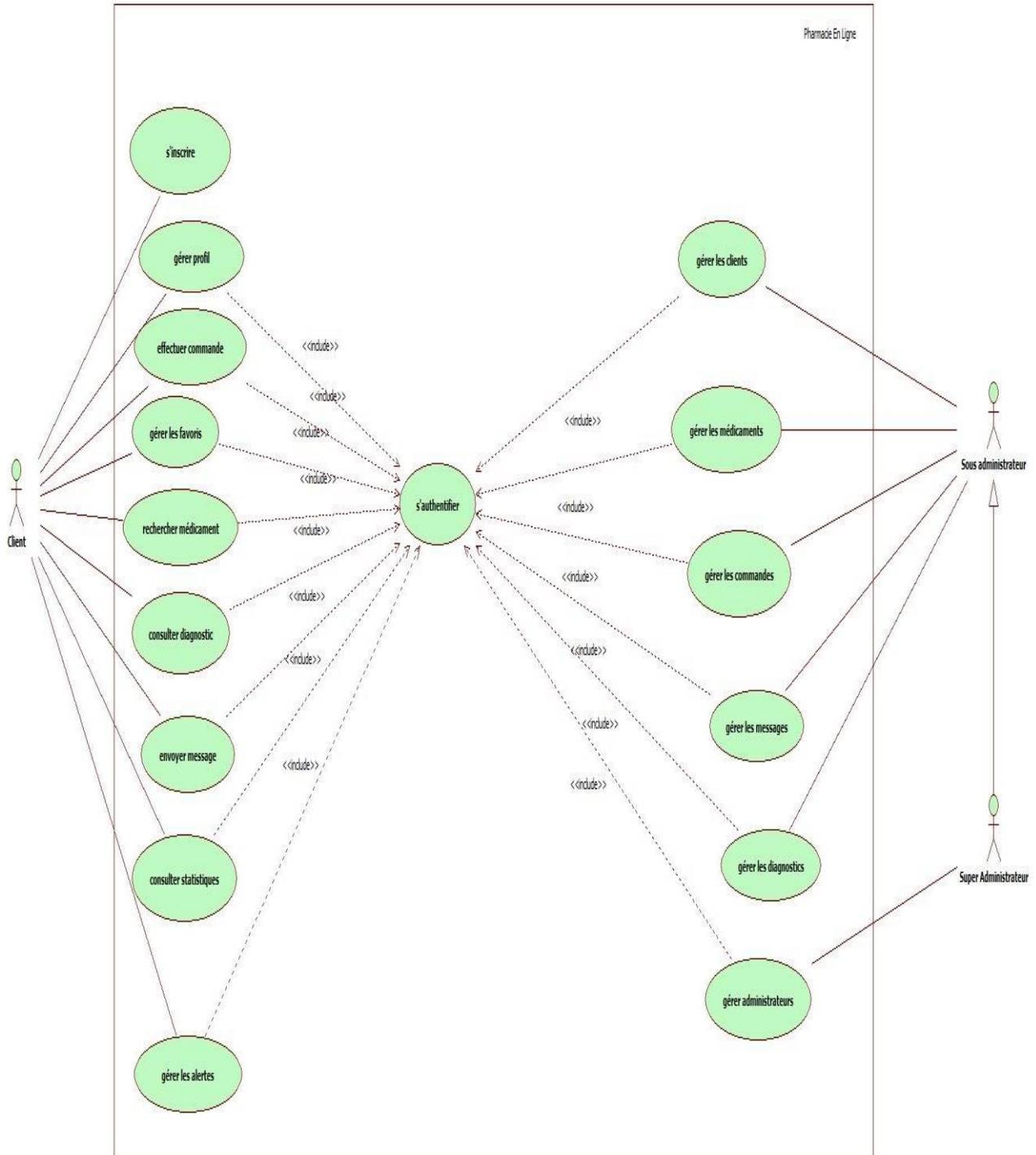


Figure 9: Diagramme de cas d'utilisation global

ii. Diagrammes de cas d'utilisation de la partie Back-office

a. Diagramme de cas d'utilisation << ajouter administrateur>>

La figure 10 représente le diagramme de cas d'utilisation d'ajout d'un administrateur. Cette tache permet d'ajouter un nouvel administrateur qui peut faire des taches similaires au super administrateur mais celui-ci ne peut pas gérer les administrateurs.

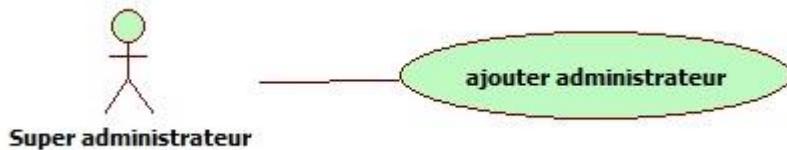


Figure 10: Diagramme de cas d'utilisation global

b. Diagramme de cas d'utilisation << Gérer clients >>

La figure 11 représente le diagramme de cas d'utilisation de gestion clients. Cette tache permet de consulter la liste des clients utilisant l'application Android ou de supprimer un client qui n'est pas actif depuis longtemps.

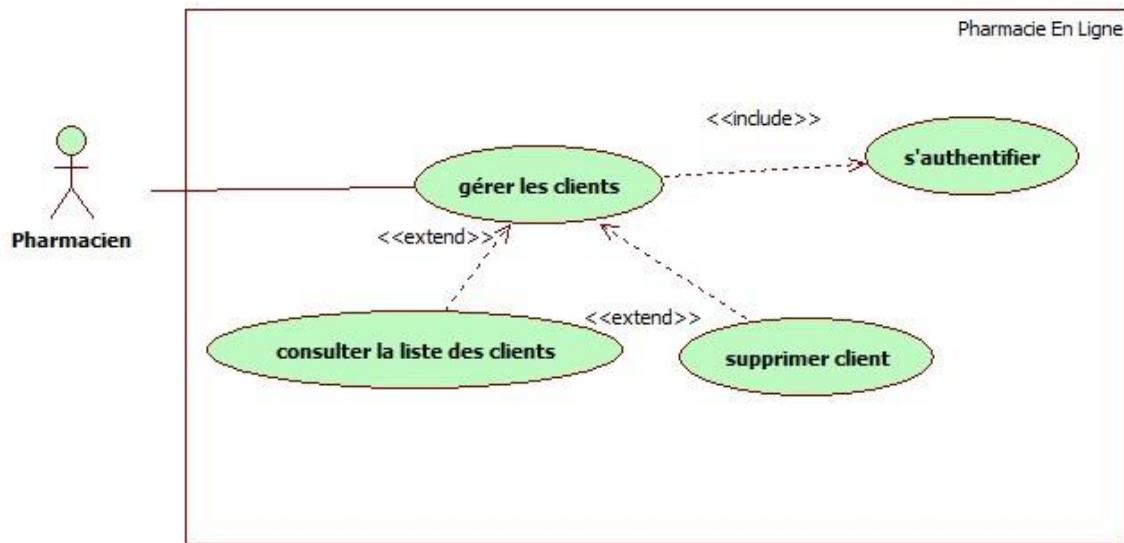


Figure 11: Diagramme de cas d'utilisation << Gérer client>>

c. Diagramme du cas d'utilisation << Gérer médicaments >>

Cette fonctionnalité illustrée dans la figure 12 a pour rôle d'effectuer le traitement qui touche directement au stock : demandes de liste des produits, suivi des mouvements et l'état du stock.

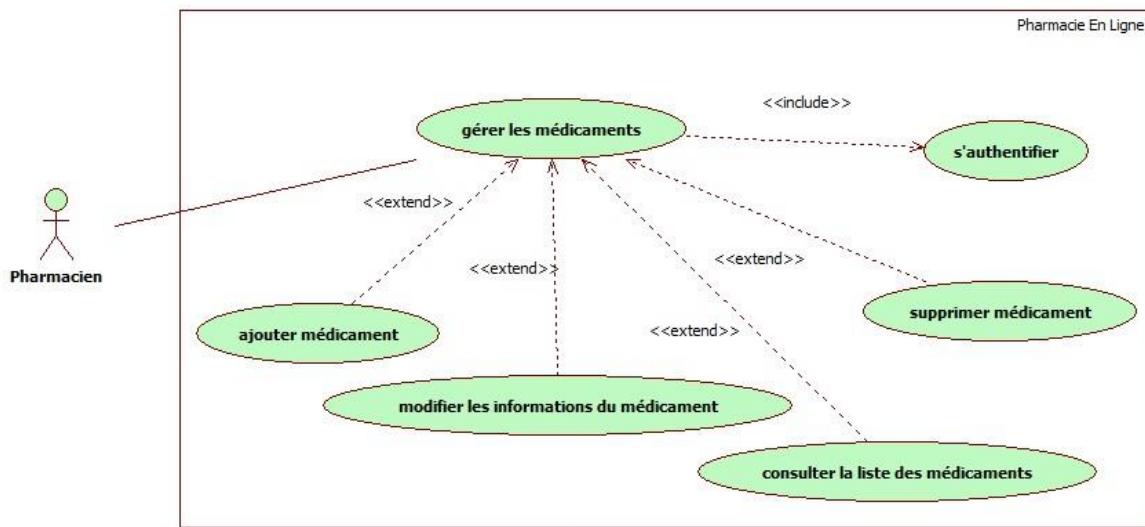


Figure 12: Diagramme de cas d'utilisation << Gérer médicament >>

d. Diagramme du cas d'utilisation << Gérer commandes >>

La gestion de commande représentée dans la figure 13 permet au pharmacien de consulter la liste des commandes ainsi que l'annulation de celles qui sont périmées.

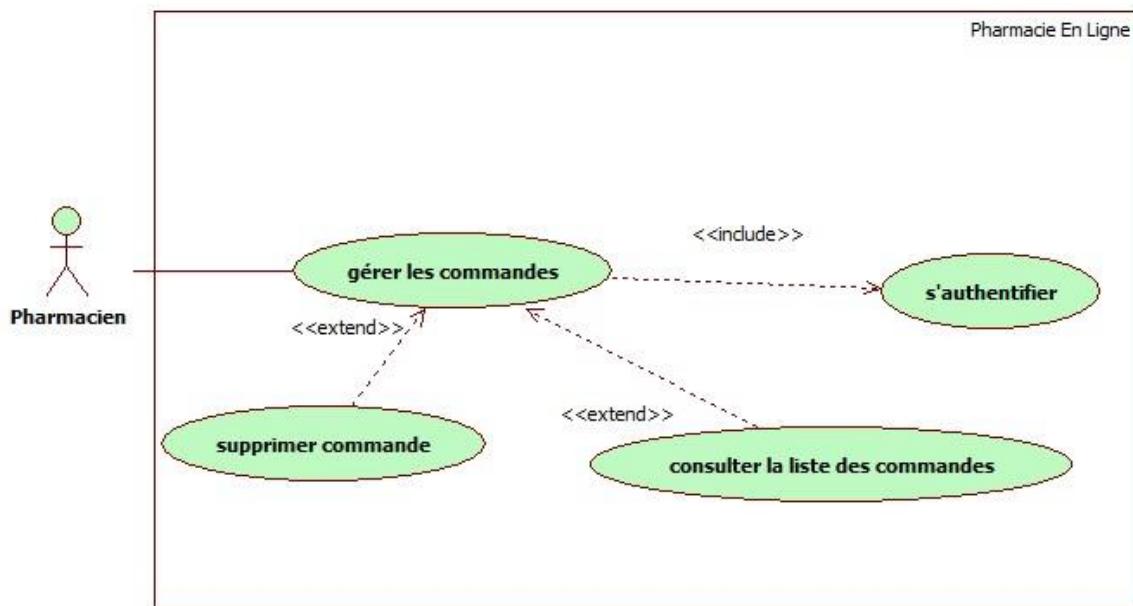


Figure 13: Diagramme de cas d'utilisation << Gérer commandes >>

e. Diagramme du cas d'utilisation << Gérer messages >>

La figure 14 montre que l'administrateur est capable d'accéder à la liste des messages envoyés par le client depuis l'application client, lui répond ou supprime les messages de cette liste.

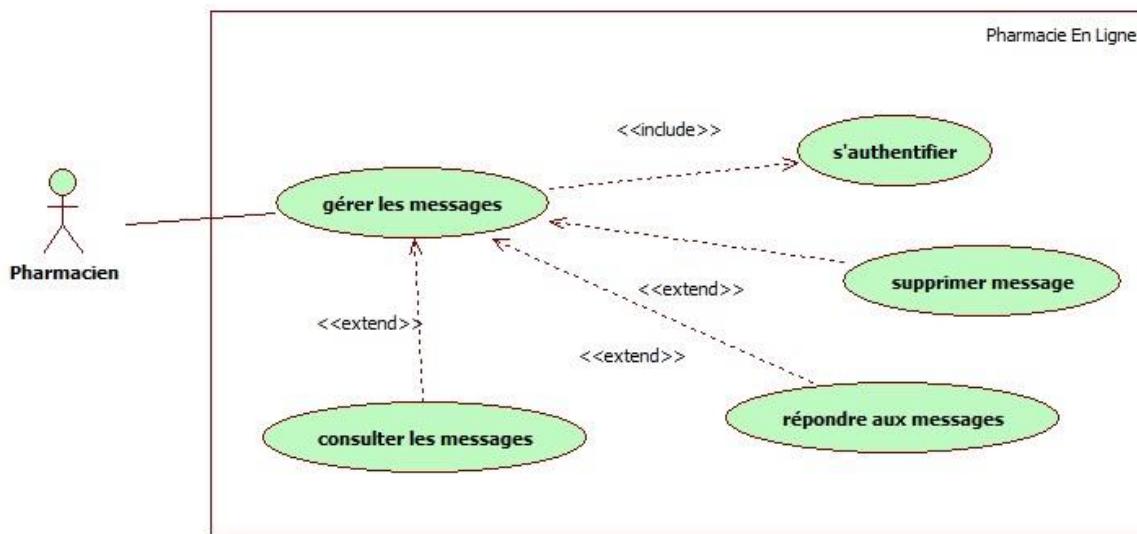


Figure 14: Diagramme de cas d'utilisation << Gérer messages >>

f. Diagramme du cas d'utilisation << Gérer Diagnostics >>

La figure 15 illustre la gestion des diagnostics qui donne au pharmacien le pouvoir de définir un diagnostic approximatif ,de modifier un ancien ou de supprimer un existant ainsi que la consultation des diagnostics définis en avance.

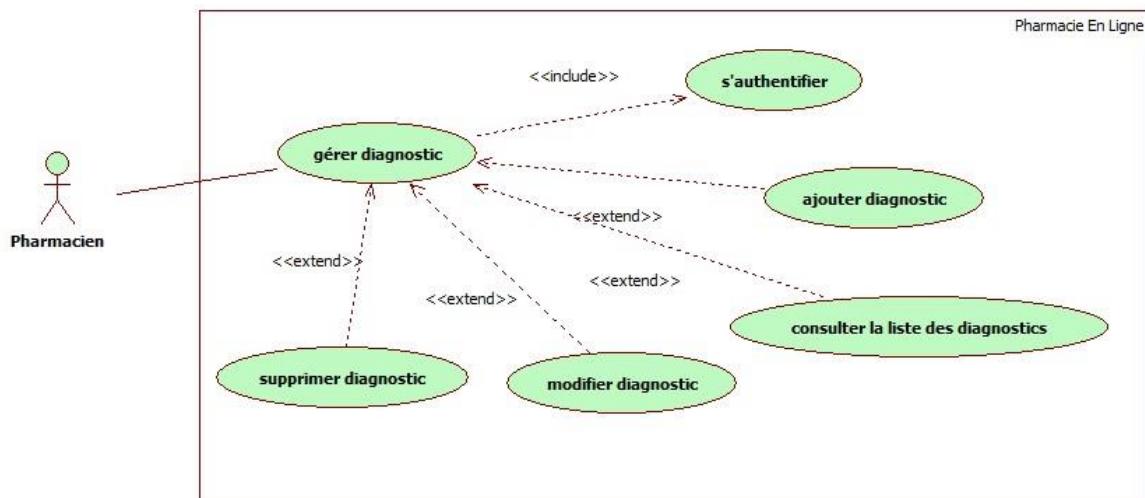


Figure 15: Diagramme de cas d'utilisation << Gérer Diagnostics >>

iii. Diagrammes de cas d'utilisation partie client

a. Diagramme de cas d'utilisation <<s'inscrire>>

La fonctionnalité illustrée dans la figure 16 offre au client la possibilité de créer un nouveau compte.

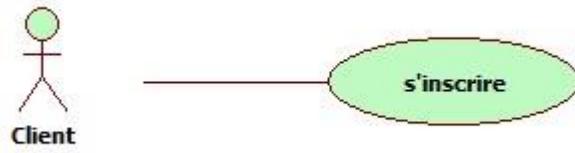


Figure 16: Diagramme de cas d'utilisation << s'inscrire>>

b. Diagramme du cas d'utilisation << Gérer Profil >>

La fonctionnalité illustrée dans la figure 17 permet au client de consulter son profil et de modifier ses informations personnelles s'il veut après la phase de l'authentification.

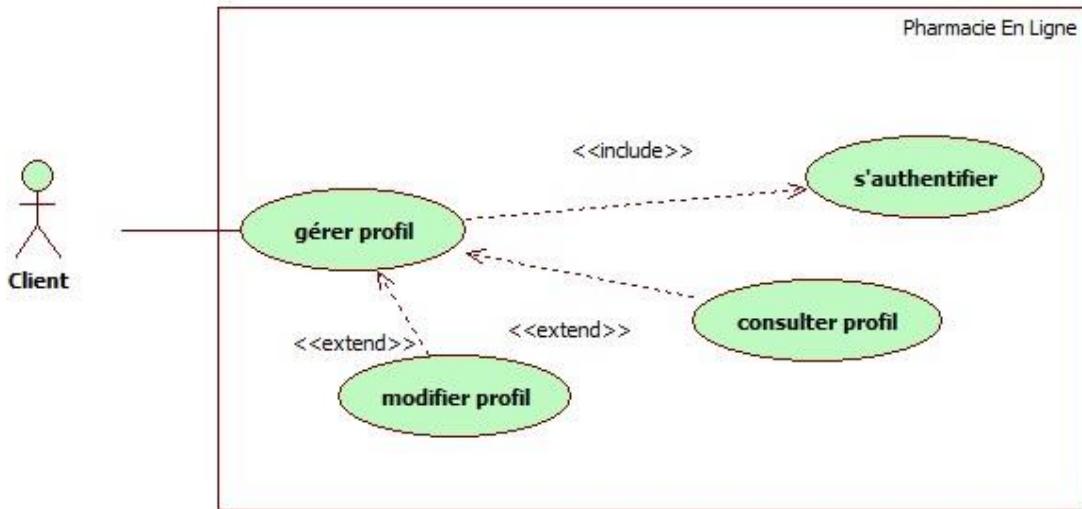


Figure 17: Diagramme de cas d'utilisation << Gérer Profil >>

c. Diagramme du cas d'utilisation << Effectuer des commandes >>

Le service illustré par la figure 18 représente la fonctionnalité principale de système qui sert à effectuer un achat en ligne en envoyant une commande de la partie client.

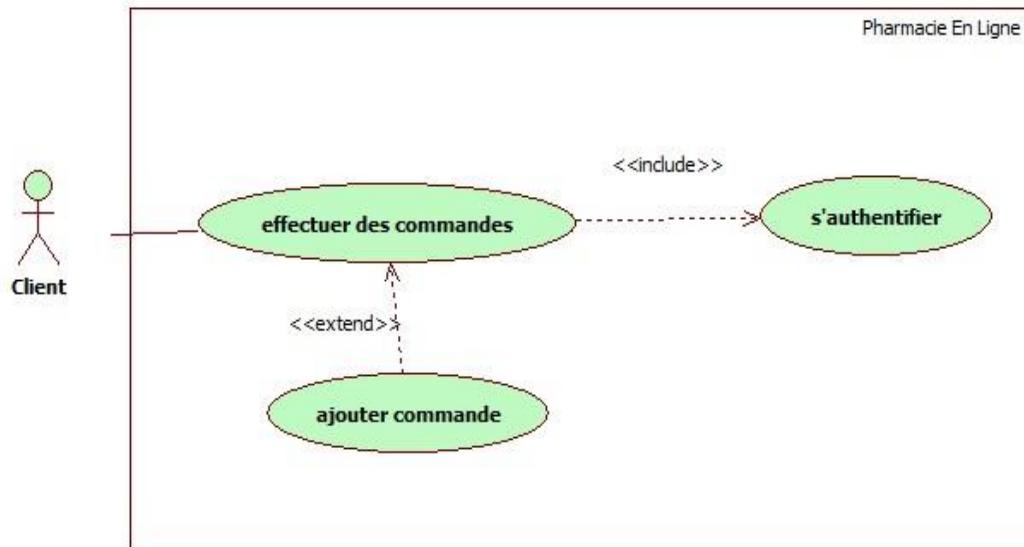


Figure 18: Diagramme de cas d'utilisation << Effectuer des commandes >>

d. Diagramme du cas d'utilisation << Gérer des favoris >>

Au sein de la gestion des favoris représentée par la figure 19, le client aura une permission soit pour ajouter un produit à sa liste des favoris, soit pour le supprimer soit pour consulter la liste totale.

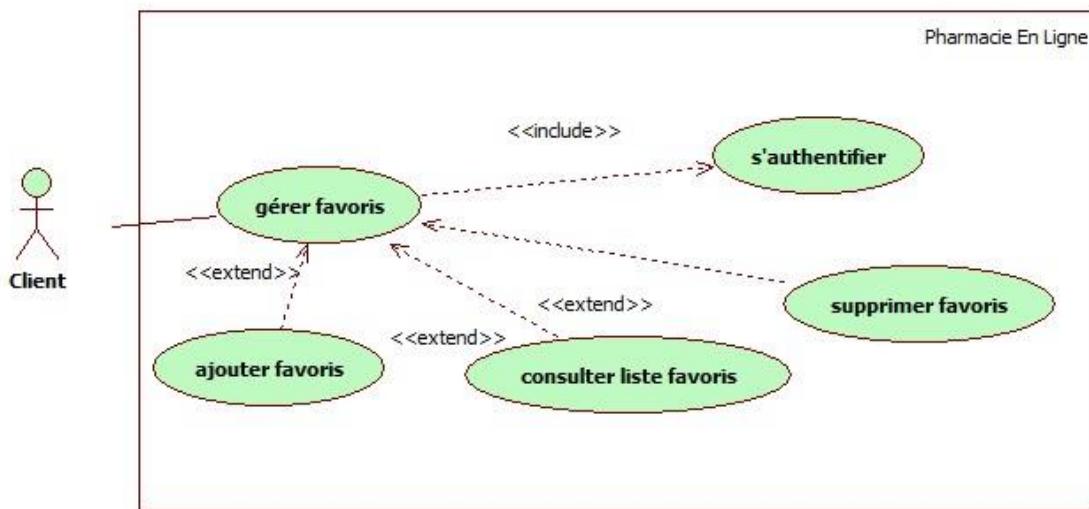


Figure 19: Diagramme de cas d'utilisation << Gérer favoris >>

e. Diagramme du cas d'utilisation << Effectuer de recherche >>

La figure 20 montre que le client peut effectuer des recherches des médicaments après la phase d'authentification.

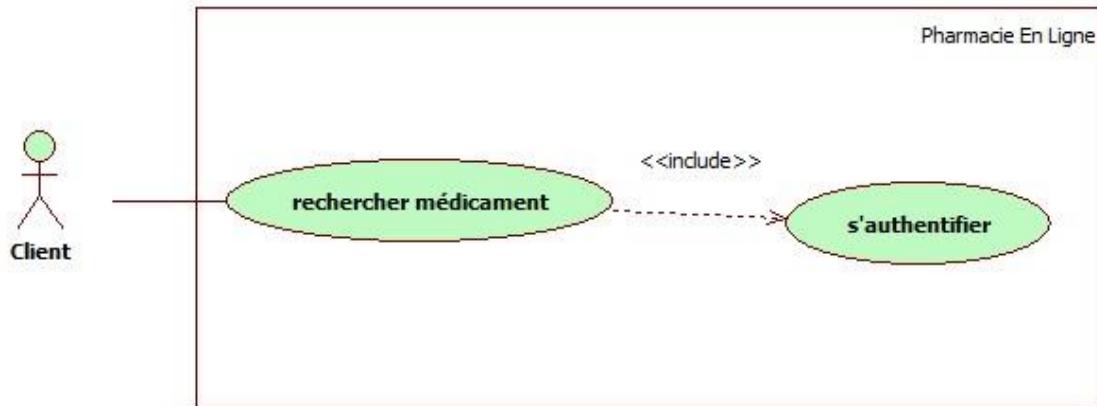


Figure 20: Diagramme de cas d'utilisation << Effectuer recherche >>

f. Diagramme du cas d'utilisation << Consulter des diagnostics >>

Cette fonctionnalité est illustrée par la figure 21. Elle permet au client d'avoir un diagnostic approximatif ainsi que des suggestions des médicaments après avoir entrer ses symptômes.

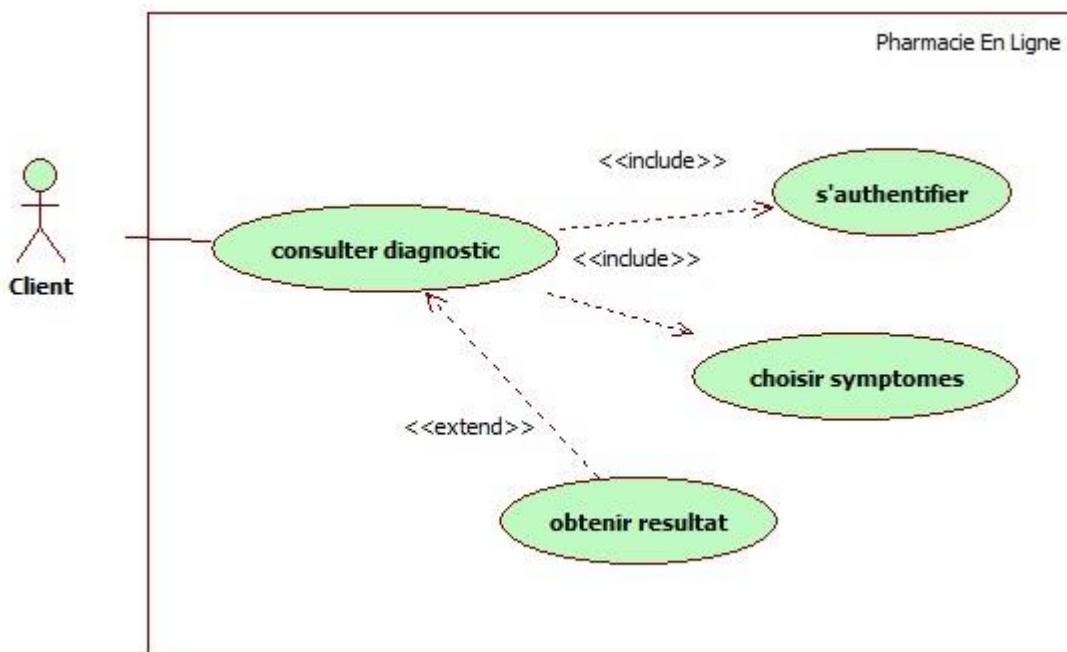


Figure 21:Diagramme de cas d'utilisation << Consulter des diagnostics>>

g. Diagramme du cas d'utilisation <<Envoyer des messages>>

Ce service permet au client d'envoyer son avis concernant l'application Android ou demander au pharmacien la méthode d'effectuer une commande, une recherche, etc.

Ceci est indiqué dans la figure 22.

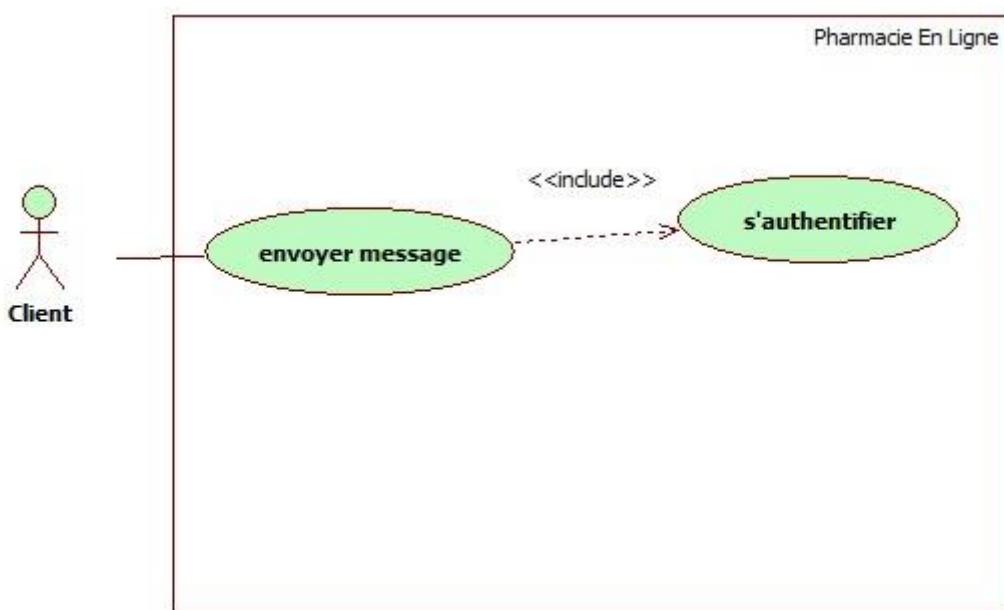


Figure 22: Diagramme de cas d'utilisation << Envoyer des messages >>

h. Diagramme du cas d'utilisation << consulter les alertes>>

Ce service permet au client de recevoir des alertes concernant la date de commande du médicament régulier.

Ceci est indiqué dans la figure 23.

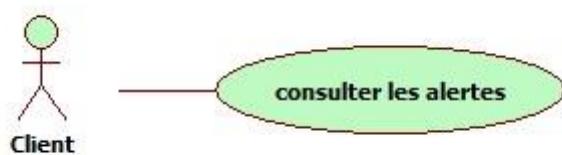


Figure 23: Diagramme de cas d'utilisation << consulter les alertes >>

i. Diagramme du cas d'utilisation << Consulter des statistiques >>

Cette fonctionnalité est décrite par la figure 24, elle permet au client de suivre les statistiques des médicaments qui se trouvent dans le stock.

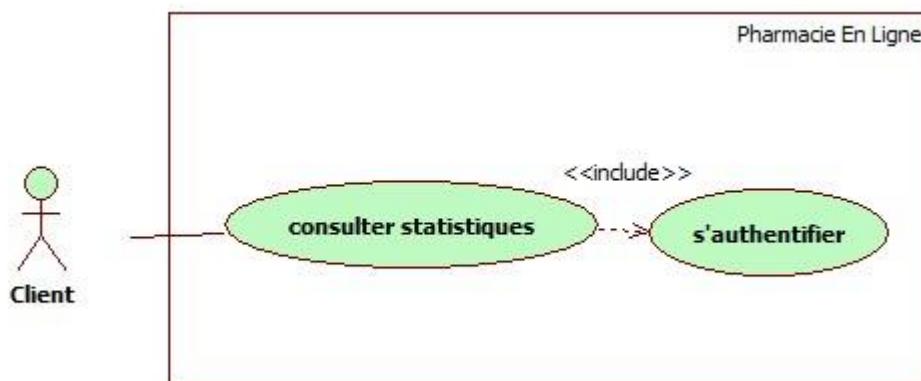


Figure 24: Diagramme de cas d'utilisation << Consulter des statistiques >>

2.2.2. Diagramme de séquence système

a) Partie Back-office

i. Diagramme de séquence système de cas d'utilisation « ajouter médicament »

Le troisième diagramme est illustré par la figure 25 Il représente le scénario <<ajouter médicament>>

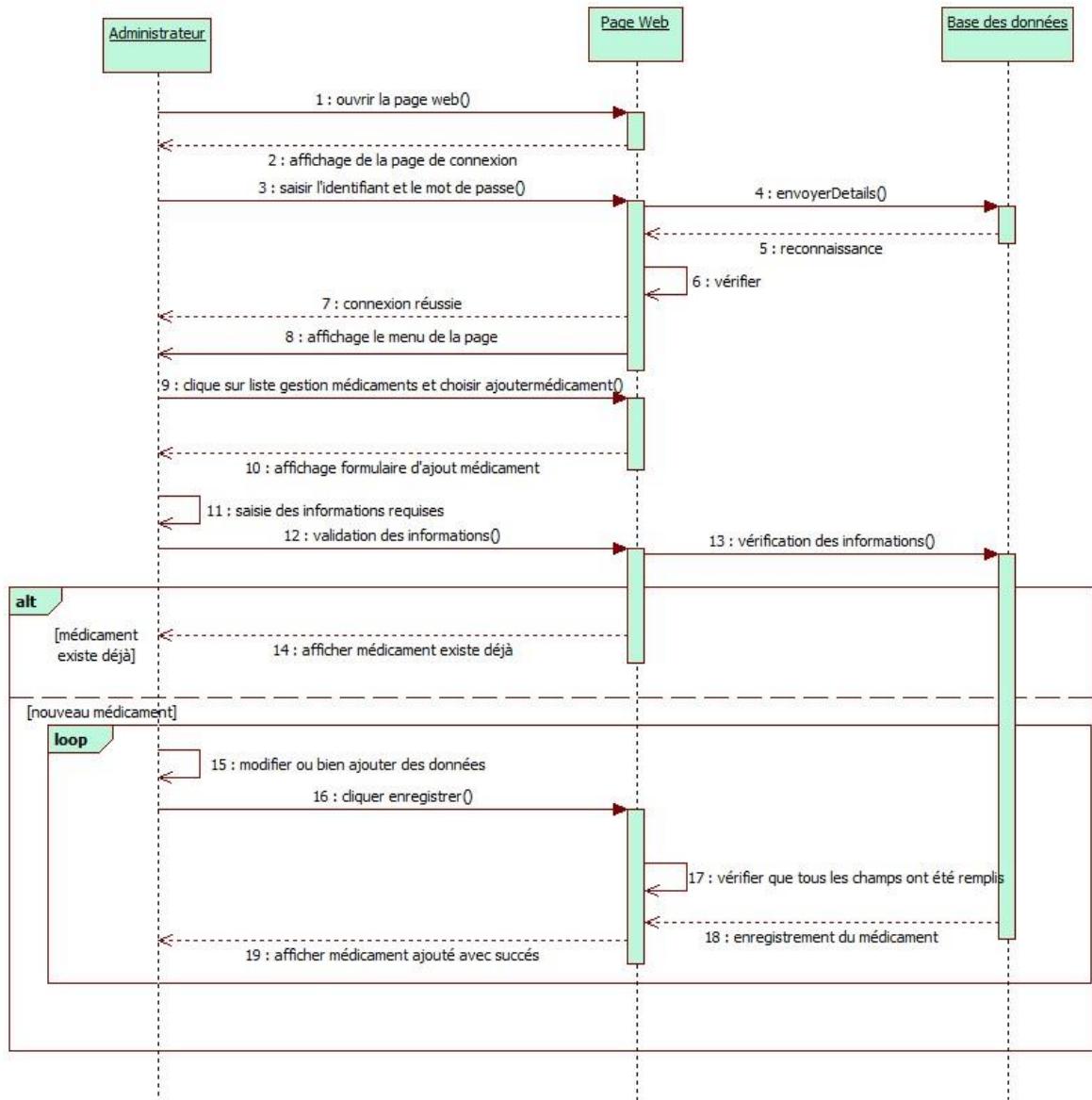


Figure 25: Diagramme de séquence <<ajouter médicament>>

La description textuelle de ce scénario est la suivante : l'ajout d'un nouveau médicament nécessite que la base des données ne possède pas auparavant un médicament de même identifiant. En plus lors de la saisie des informations du médicament, l'administrateur doit

respecter les types de ces informations sinon un message d'erreur s'affiche. Si toutes les conditions sont remplies correctement, le nouveau médicament s'ajoute à la base des données.

Cas d'utilisation	Ajouter médicament
Acteur principal	Pharmacien
Pré-condition	l'application est disponible et l'administrateur est authentifié.
Post-condition	Le médicament est ajouté avec succès.
Scénario nominal	<p>Processus commence lorsque l'administrateur clique sur le bouton ajouter médicament</p> <p>L'administrateur saisi les informations requises.</p> <p>La page web assure la vérification de l'existence du médicament.</p> <p>La suite l'administrateur assure le remplissage ou la modification des données.</p> <p>Lorsque l'administrateur clique sur valider le système enregistre les données d'un médicament.</p>
Exception	La page web affiche les messages informations manquantes.

Tableau 4:Diagramme de séquence <<ajouter médicament>>

b) Partie client

i. Diagramme de séquence de cas d'utilisation « s'authentifier»

Le premier diagramme présentant la partie mobile est illustré dans la figure 26. Il présente le scénario «s'authentifier».

Le client est invité à saisir ses paramètres de connexion. La base des données vérifie la disponibilité du login et du mot de passe, pour donner ensuite accès aux fonctionnalités de l'application.

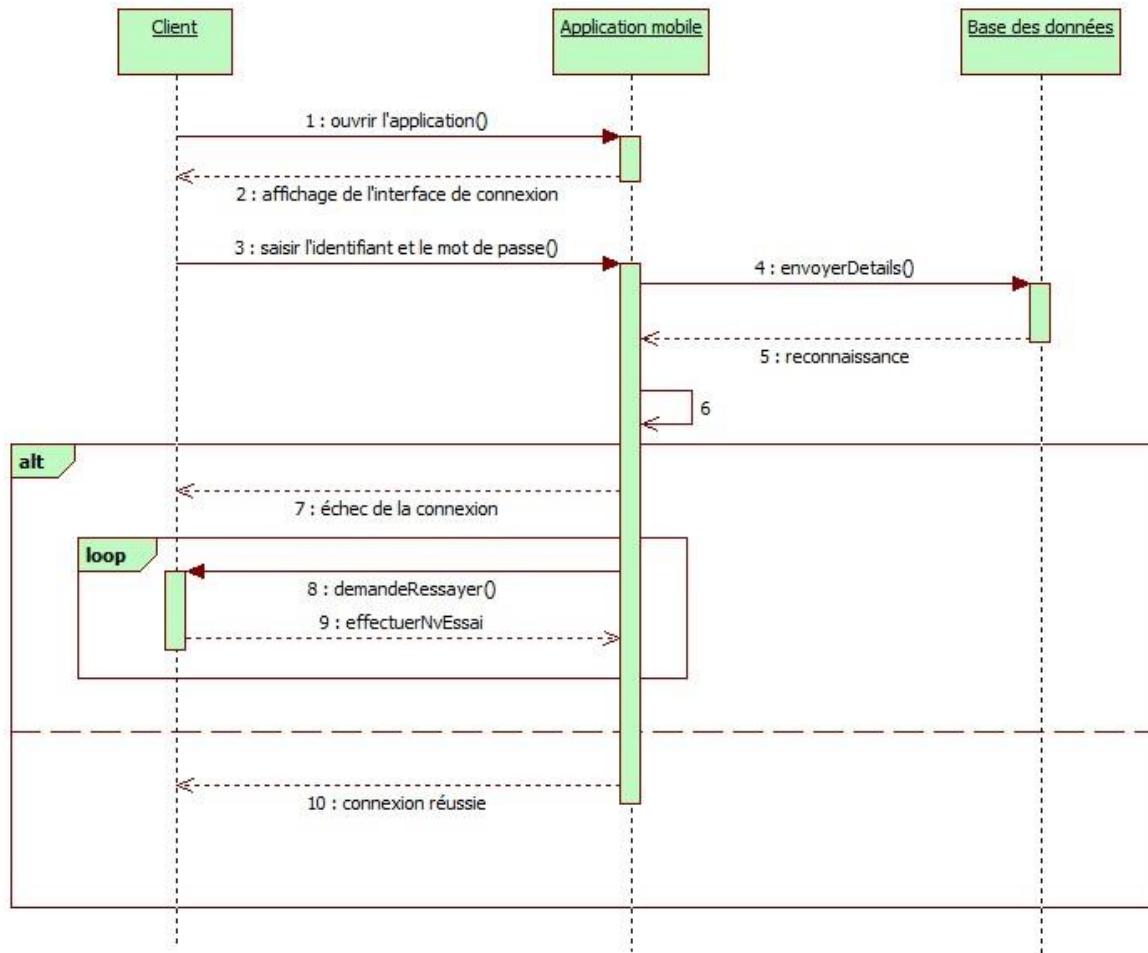


Figure 26: Diagramme de séquence << s'authentifier>>

Cas d'utilisation	S'authentifier
Acteur principal	Client
Pré-condition	l'application est disponible.
Post-condition	l'acteur est authentifié et l'interface adéquate se charge.
Scénario nominal	L'acteur demande l'accès pour se connecter.

	L'application mobile affiche un formulaire à remplir dans lequel il saisit son login et son mot de passe.
	L'acteur saisit le login et le mot de passe.
	L'activation du compte et l'affichage de l'interface.

Tableau 5:Description du Scénario " s'authentifier "

ii. Diagramme de séquence de cas d'utilisation « commander un médicament »

Après avoir s'authentifier, le client pourrait commander un ou plusieurs médicaments.

Le diagramme de la Figure 27 illustre le déroulement de ce cas.

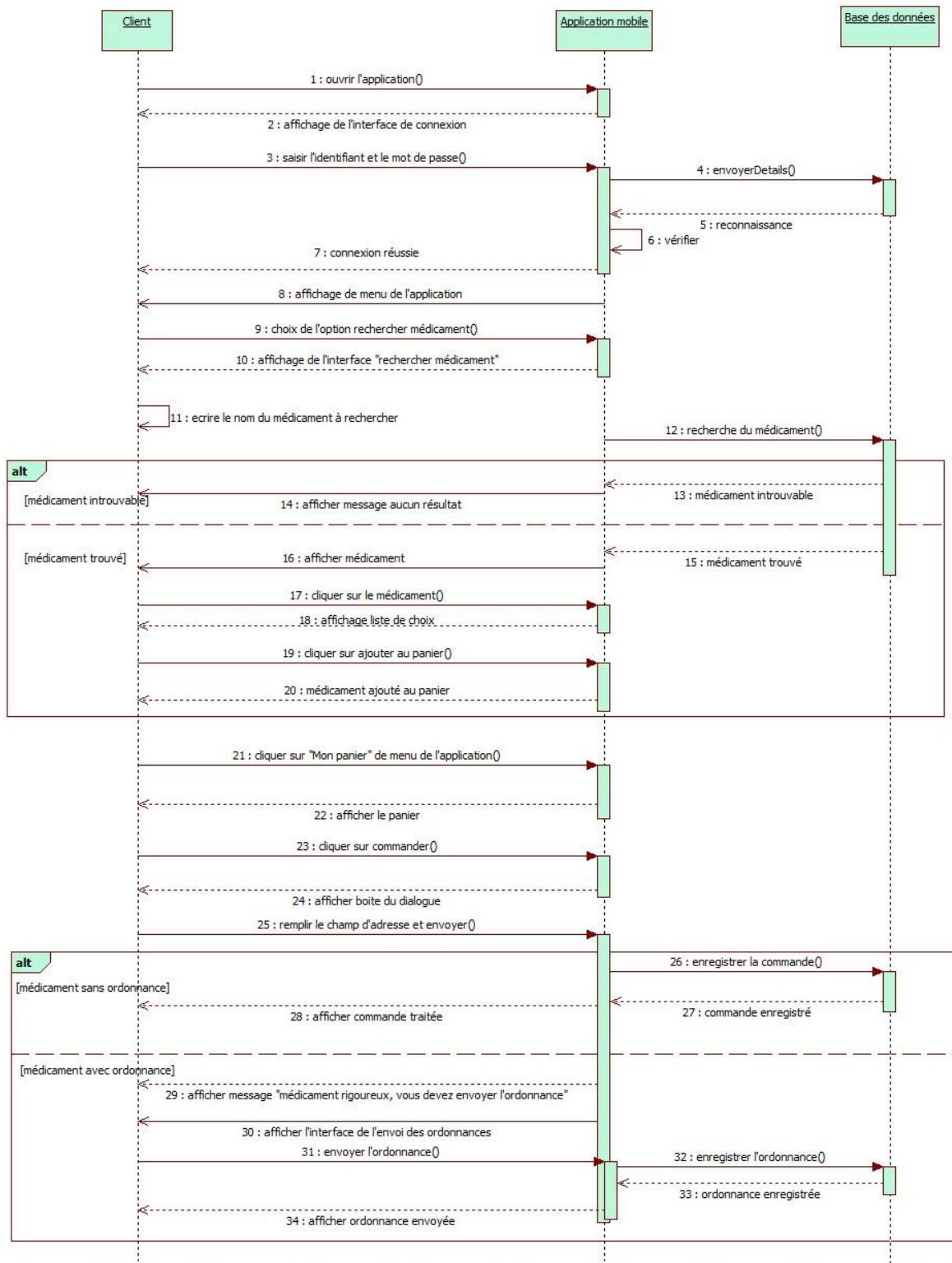


Figure 27: Diagramme de séquence << commander un médicament>>

Cas d'utilisation	Commander un médicament
Acteur principal	Client
Pré-condition	l'application est disponible et le client est authentifié.
Post-condition	La commande est traitée.
Scénario nominal	<p>Processus commence lorsque le client fait la recherche du médicament. Il clique sur ajouter au panier.</p> <p>Ensuite, il clique sur commander et saisi son adresse.</p> <p>Lorsque l'utilisateur clique sur envoyer, l'application vérifie si le médicament est rigoureux ou pas, si oui, le client doit envoyer une ordonnance puis, la commande s'enregistre dans la base des données.</p>

Tableau 6:Description du Scénario " commander un médicament "

iii. Diagramme de séquence de cas d'utilisation « Consulter diagnostics »

Après avoir s'authentifier, le client pourrait consulter la liste des diagnostics disponibles. Ce scénario est décrit dans la Figure 28.

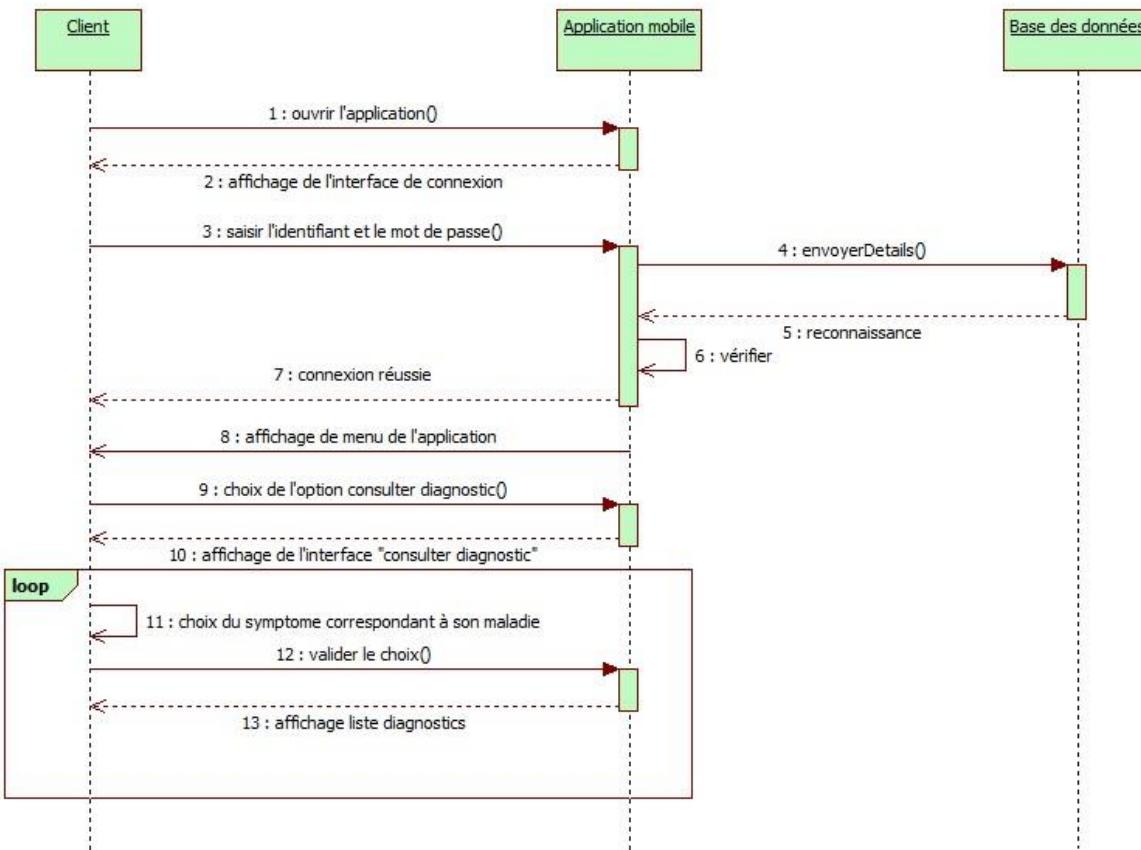


Figure 28: Diagramme de séquence << consulter diagnostics>>

Cas d'utilisation	Consulter diagnostic
Acteur principal	Client
Pré-condition	l'application est disponible et le client est authentifié.
Post-condition	Affichage de la liste des diagnostics.
Scénario nominal	<p>Processus commence lorsque le client clique sur « diagnostic ».</p> <p>Le client choisit le symptôme correspondant à son maladie.</p> <p>L'application affiche la liste des diagnostics.</p>

Tableau 7:Description du Scénario "consulter diagnostics "

3. Les maquettes de la partie Back-office

3.1. Page de connexion

Un administrateur ne pourra accéder au système que s'il s'identifie en indiquant son login et son mot de passe. La figure 29 présente l'interface authentification du site web.



Figure 29:Page de connexion de la partie Back-office

3.2. Page d'accueil

Une fois l'utilisateur s'est authentifié, selon son niveau d'accès une page écran lui sera affichée présentant plusieurs choix

- Gestion des administrateurs
- Gestion des médicaments
- Gestion des diagnostics
- Gestion des clients
- Gestion des commandes

La figure 30 illustre la page d'accueil de notre site web.



Figure 30:Page d'accueil de la partie Back-office

3.3. Page d'ajout d'un administrateur

Le super administrateur peut ajouter des sous administrateurs. Il suffit de remplir les paramètres d'accès et confirme l'ajout.

Cette page web est présentée par la figure 31 représente ce cas.

A screenshot of a 'Create New Admin' form. The left sidebar is identical to Figure 30. The main form has a light gray header with a search bar and a notifications icon. The body of the form is titled 'Ajouter un Administrateur'. It contains two input fields: 'Login' and 'Password', both with placeholder text. Below the password field is a horizontal line. At the bottom of the form is a green 'Submit' button.

Figure 31:Page d'ajout d'un administrateur de la partie Back-office

3.4. Page de la modification d'un administrateur

L'administrateur peut modifier ses paramètres d'accès. La figure 32 illustre cette page web.

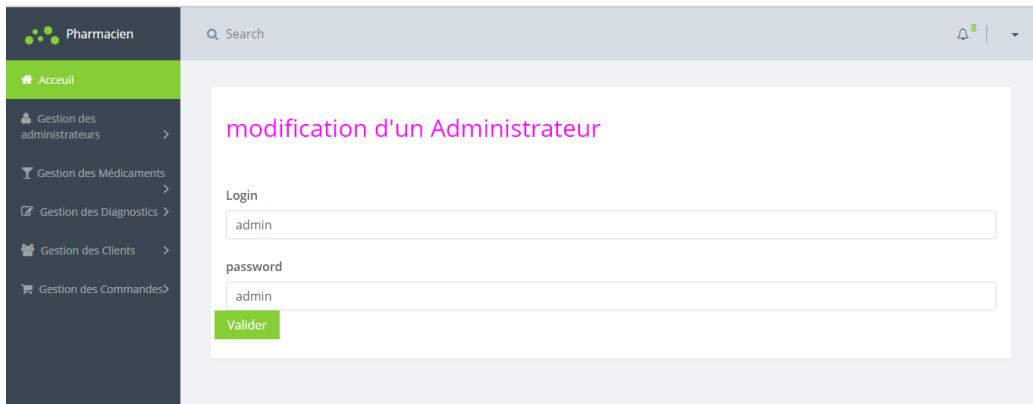


Figure 32:Page de la modification d'un administrateur de la partie Back-office

3.5. Page de la liste des administrateurs

La figure 33 présente la liste des administrateurs. L'administrateur peut modifier ou supprimer un d'eux.

The screenshot shows a table titled 'Liste des administrateurs' with three rows of data. The columns are labeled '#', 'login', 'password', and 'action'. The first row contains '#1', 'admin', 'admin', and two buttons: 'Modifier' (orange) and 'Supprimer' (red). The second row contains '#18', 'zayneb', 'c7796ceb695beaffb4a6abe195ba2cd901202968', and two buttons. The third row contains '#20', 'malek', '7c4a8d09ca3762af61e59520943dc26494f8941b', and two buttons.

#	login	password	action
1	admin	admin	<button>Modifier</button> <button>Supprimer</button>
18	zayneb	c7796ceb695beaffb4a6abe195ba2cd901202968	<button>Modifier</button> <button>Supprimer</button>
20	malek	7c4a8d09ca3762af61e59520943dc26494f8941b	<button>Modifier</button> <button>Supprimer</button>

Figure 33:Page de la liste des administrateurs de la partie Back-office

3.6. Page d'ajout d'un médicament

La figure 34 montre que lorsque l'administrateur choisit d'ajouter un médicament, il doit remplir tous les champs requis.

The screenshot shows a web-based application interface for adding a medicine. On the left, there's a dark sidebar with a green header bar containing the word "Pharmacien". Below it, several menu items are listed: "Accueil", "Gestion des administrateurs", "Gestion des Médicaments", "Gestion des Diagnostics", "Gestion des Clients", and "Gestion des Commandes". The main content area has a light gray header with a search bar and a "Search" button. Below this, the title "Ajouter un Médicament" is displayed. The form consists of several input fields: "Nom de médicament", "Référence", "Libellé", "Forme", "prix", and an "Image" field which contains a small green plus sign icon. At the bottom of the form is a green "Submit" button.

Figure 34:Page d'ajout d'un médicament de la partie Back-office

3.7. Page de la modification d'un médicament

L'administrateur peut modifier les informations d'un médicament en saisissant tous les champs demandés. Cette page web est présentée par la figure 35.

The screenshot shows a web-based application interface for modifying a medicine. The layout is similar to Figure 34, with a sidebar on the left and a main form on the right. The sidebar includes "Accueil", "Gestion des administrateurs", "Gestion des Médicaments", "Gestion des Diagnostics", "Gestion des Clients", and "Gestion des Commandes". The main area has a title "modification d'un Médicament". Below it are four input fields: "Nom de Médicament" (with placeholder "Enter nom_med"), "Référence" (with placeholder "Enter ref"), "Libelle" (with placeholder "Enter lib"), and "Forme" (with placeholder "Enter forme"). There is also a "prix" field below the "Forme" field. The overall design is clean and functional.

Figure 35:Page de la modification d'un médicament de partie Back-office

3.8. Page de la liste des médicaments

La figure 36 illustre le fait que lorsque l'administrateur choisit de consulter la liste des médicaments, il visualise cette liste et il peut modifier ou supprimer un d'eux. .

The screenshot shows a web-based application interface for a pharmacy. The left sidebar has a green header 'Pharmacien' with a logo and a navigation menu: 'Accueil', 'Gestion des administrateurs', 'Gestion des Médicaments', 'Gestion des Diagnostics', 'Gestion des Clients', and 'Gestion des Commandes'. The main content area is titled 'Liste des Médicaments' and displays a table with the following data:

#	Nom medicament	Référence	Prix	Action
3	Fervex	00004	15	<button>Modifier</button> <button>Supprimer</button>
4	Alergica	00005	7	<button>Modifier</button> <button>Supprimer</button>
5	Acide Folique	025455	25	<button>Modifier</button> <button>Supprimer</button>
6	Acide Lactique	588884	5	<button>Modifier</button> <button>Supprimer</button>
7	Amoxicilline	85555	14	<button>Modifier</button> <button>Supprimer</button>

Figure 36:Page de la liste des médicaments de la partie Back-office

3.9. Page d'ajout des diagnostics

Lorsque l'administrateur choisit d'ajouter un médicament diagnostic, il doit d'abord choisir le symptôme approximatif correspondant, il clique sur les médicaments diagnostics et il confirme l'ajout. Cette page web est présentée par la figure 37.

The screenshot shows a web-based application interface for a pharmacy. The left sidebar has a green header 'Pharmacien' with a logo and a navigation menu: 'Accueil', 'Gestion des administrateurs', 'Gestion des Médicaments', 'Gestion des Diagnostics', 'Gestion des Clients', and 'Gestion des Commandes'. The main content area is titled 'Ajouter un Diagnostic approximatif' and displays a form with the following fields:

- Symptômes:** A dropdown menu showing 'Diarrhee'.
- Medicaments:** A list of medications with radio buttons:
 - Fervex
 - Alergica
 - Acide Folique
 - Acide Lactique
 - Amoxicilline
 - Aprotine
 - Adol
 - Calcium
 - Baldo
 - Charbon
 - Codeine
- Sauvegarder:** A green 'Sauvegarder' button at the bottom.

Figure 37:Page d'ajout des diagnostics de la partie Back-office

3.10. Page de la liste des médicaments diagnostics

La figure 38 illustre la liste des médicaments diagnostics d'un seul symptôme. L'administrateur peut modifier ou supprimer un d'eux.

The screenshot shows a web-based application for pharmacists. The left sidebar has a green header 'Pharmacien' and a menu with items like 'Accueil', 'Gestion des administrateurs', 'Gestion des Médicaments', 'Gestion des Diagnostics', 'Gestion des Clients', and 'Gestion des Commandes'. The main content area is titled 'Liste des Médicaments' and displays a table with two rows. The columns are '#', 'nom_med', and 'Action'. Row 1 contains '1', 'Fervex', and buttons for 'Modifier' (orange) and 'Supprimer' (red). Row 2 contains '2', 'Alergica', and buttons for 'Modifier' (orange) and 'Supprimer' (red).

#	nom_med	Action
1	Fervex	Modifier Supprimer
2	Alergica	Modifier Supprimer

Figure 38:Page de la liste des médicaments par diagnostic de la partie Back-office

3.11. Page de la liste des symptômes

La figure 39 présente la liste des symptômes que l'administrateur peut effectuer la modification ou la suppression d'un symptôme.

The screenshot shows the same application interface. The main content area is titled 'Liste des Diagnostics approximatifs' and displays a table with four rows. The columns are '#', 'Symptomes', 'Médicaments', and 'Action'. Row 1 contains '1', 'Difficultes importantes pour respirer', and buttons for 'Modifier' (orange) and 'Supprimer' (red). Row 8 contains '8', 'Diarrhee', and buttons for 'Modifier' (orange) and 'Supprimer' (red). Row 9 contains '9', 'Diarrhee', and buttons for 'Modifier' (orange) and 'Supprimer' (red). Row 10 contains '10', 'Nausee', and buttons for 'Modifier' (orange) and 'Supprimer' (red).

#	Symptomes	Médicaments	Action
1	Difficultes importantes pour respirer	Medicaments	Modifier Supprimer
8	Diarrhee	Medicaments	Modifier Supprimer
9	Diarrhee	Medicaments	Modifier Supprimer
10	Nausee	Medicaments	Modifier Supprimer

Figure 39:Page de la liste des symptômes de la partie Back-office

3.12. Page de la liste des clients

La figure 40 permet d'afficher la liste des clients. L'administrateur peut effectuer la suppression d'un client qui n'est pas actif.

The screenshot shows a web-based application interface for a pharmacien. On the left, there is a vertical navigation menu with the following items: Accueil, Gestion des administrateurs, Gestion des Médicaments, Gestion des Diagnostics, Gestion des Clients, and Gestion des Commandes. The 'Gestion des Clients' item is currently selected. The main content area is titled 'Liste des Clients' and displays a table with five rows of data. The columns are labeled: ID Client, Nom Client, Prenom Client, Adresse, and Action. The data in the table is as follows:

ID Client	Nom Client	Prenom Client	Adresse	Action
2	hmida	zayneb	mahdia	<button>Supprimer</button>
3	hmida	fatma ezzahra	mahdia	<button>Supprimer</button>
4	Mahfoudh	Balkis	Wardenine	<button>Supprimer</button>
5	Karamti	Walid	Sfax	<button>Supprimer</button>

Figure 40:Page de la liste des clients de la partie Back-office

3.13. Page de la liste des commandes

La figure 41 permet d'afficher à l'administrateur la liste des commandes qu'il peut les valider ou supprimer.

The screenshot shows a web-based application interface for a pharmacien. The left navigation menu is identical to Figure 40. The main content area is titled 'Liste des Commandes' and displays a table with one row of data. The columns are labeled: Date, Nom Client, Prenom Client, adresse, liste de medicaments, and Action. The data in the table is as follows:

Date	Nom Client	Prenom Client	adresse	liste de medicaments	Action
2017-03-07	hmida	zayneb	mahdia	medicaments	<button>Valider</button> <button>Supprimer</button>

Figure 41:Page de la liste des commandes de la partie Back-office

3.14. Page d'une ligne de commande

Cette page permet à l'administrateur de visualiser les lignes commandes pour une commande sélectionnée. Cette page web est présentée par la figure 42.

The screenshot shows a web-based application for pharmacists. The header 'Pharmacien' is at the top left, with a green notification bar indicating 8 messages. The sidebar on the left contains links: 'Accueil', 'Gestion des administrateurs', 'Gestion des Médicaments', 'Gestion des Diagnostics', 'Gestion des Clients', and 'Gestion des Commandes'. The main content area is titled 'Liste des Médicaments' and shows a table with two rows of medication data:

nom medicament	qt	prix	Action
Fervex	24	15	<button>Modifier</button> <button>Supprimer</button>
Alergica	11	7	<button>Modifier</button> <button>Supprimer</button>

Figure 42:Page des lignes de commandes de la partie Back-office

Conclusion

L’analyse des besoins procure une vision plus claire du sujet et une compréhension plus profonde des tâches à réaliser. Dans ce chapitre, nous avons exprimé d’une manière semi-formelle les besoins fonctionnels et non fonctionnels de notre système. Nous avons fourni une analyse plus détaillée de ces besoins grâce aux diagrammes des cas d’utilisation et les diagrammes de séquences.

Ces spécifications serviront de base à l’étape de conception de l’application, objet du chapitre suivant.



Chapitre3 : Conception

Introduction

La conception est une étape primordiale dans le cycle de vie d'une application. Elle présente un pont entre l'analyse et l'implémentation.

Elle a pour objectif d'élaborer à partir de l'architecture de système. Elle vise également la réduction de la complexité du système.

Ce chapitre permet de déterminer l'architecture de l'application et d'élaborer le dictionnaire de données, afin de dégager les classes des objets avec leurs relations et leurs attributs.

1. Conception générale de l'application

La conception consiste à scinder les tâches de l'application en différentes petites parties afin de mieux organiser et développer le logiciel.

Pour établir un programme correctement conçu, il faut se pencher sur les patrons de conception qui vont nous guider tout au long du projet. Pour cela nous avons choisi d'utiliser les patrons de conception MVC : Model - View - Contrôleur. La partie web et la partie mobile de notre application respectent le modèle MVC.

1.1. MVC

MVC est un design pattern (un modèle de conception) qui permet d'organiser le code de l'application selon trois parties bien précises : Modèle, Vue et Contrôleur. Son principal intérêt est la **séparation entre** le modèle de données, l'interface utilisateur et la logique de contrôle. [9]

Il est constitué des éléments suivants

- **Le Modèle**

Il représente le comportement de l'application : traitements des données, interactions avec la base de données, etc.

- **La Vue**

Elle constitue les éléments d'interface utilisateurs : pages web, les interfaces de l'application mobile, etc.

- **Le Contrôleur**

Il gère l'interface entre le modèle et le client. Il va interpréter la requête de ce dernier pour lui envoyer la vue correspondante. Il effectue la synchronisation entre le modèle et les vues.

La figure 43 présente les relations entre ces trois éléments.

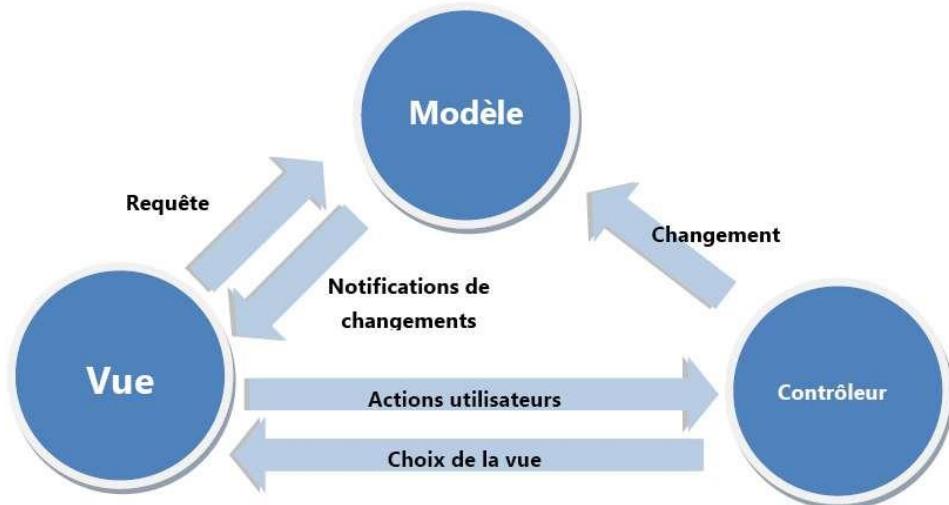


Figure 43:Les interactions entre le modèle, la vue et le contrôleur

1.1.1. MVC et la partie client

Il est primordiale à la conception de tout système informatique de choisir le modèle d'architecture qui lui sera adéquat pouvant assurer un bon fonctionnement, des meilleures performances ainsi que la réutilisation et l'interconnexion fiable de ce système avec d'autres.

C'est à cet effet que nous optons pour le modèle MVC qui sera également très pratique pour gérer l'interaction entre les différents composants de notre application Android.

a) Le modèle

Le modèle correspond aux données, la plupart du temps stockées dans une base de données. Mais celles-ci peuvent également être contenues dans un fichier XML ou dans des fichiers texte.

Les données peuvent être exploitées sous forme de classes, dans un langage de programmation orientée objet.

Dans le cas de notre projet la partie modèle représente les classes de notre application. Ci-dessous, le modèle de l'application mobile.

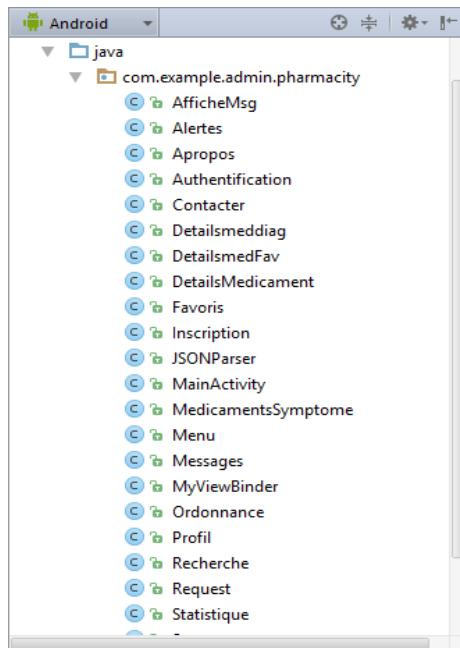


Figure 44: le modèle de l'application mobile

b) Le contrôleur

Le contrôleur est l'élément qui va utiliser les données pour les envoyer à la vue. Son rôle est donc **de récupérer les informations, de les traiter en fonction des paramètres** demandés par la vue (par l'utilisateur), puis de **renvoyer à la vue** les données afin d'être affichées.

Dans le cas de notre projet la partie contrôleur représente les documents PHP qui vont récupérer les données de la base des données.

Ci-dessous, le contrôleur de l'application mobile.

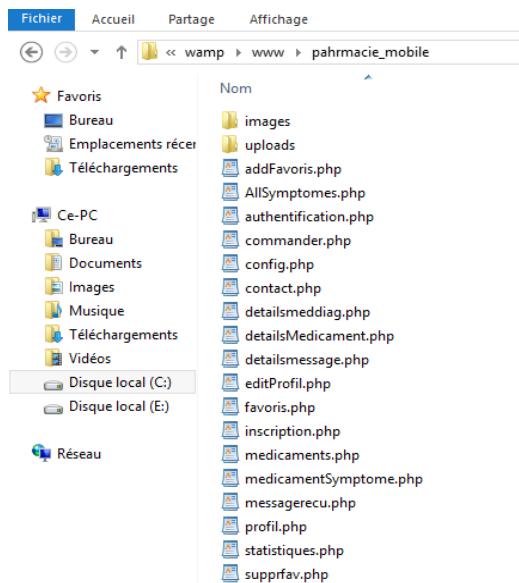


Figure 45: le contrôleur de l'application mobile

c) La vue

Elle se contente d'afficher le contenu qu'elle reçoit sans avoir connaissance des données. Elle ne contient que les informations liées à l'affichage.

Dans le cas de notre projet la partie vue représente les layouts de l'application Android. Ci-dessous, la vue de l'application mobile.

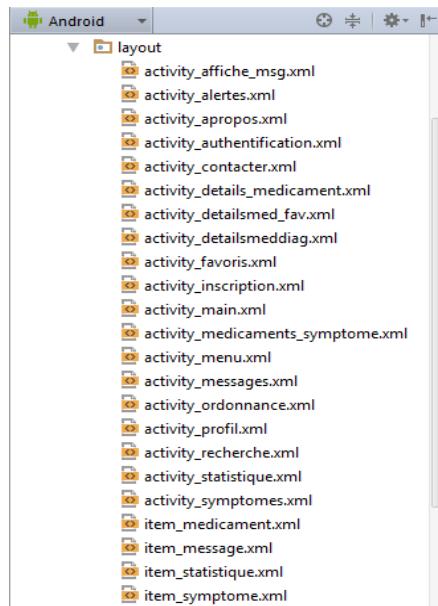


Figure 46: la vue de l'application mobile

1.1.2. MVC et la partie Back office PHP

PHP (*Hypertext Preprocessor*) est un langage de programmation langage impératif orienté objet principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP.

Nous avons recouru à l'utilisation de ce langage puisqu'il permet la rapidité de développement. De plus, l'application du modèle MVC permet aux pages HTML (qui constituent la partie Vue) de contenir le moins possible de code serveur, étant donné que le scripting est regroupé dans les deux autres parties de l'application. []

En suivant les directives du modèle MVC, nous saurons quels fichiers créer et quels rôles leur donner

- Modèle : Cette partie gère les données du site. Elle récupère les données dans la base de données (via des requêtes SQL) et les met en forme pour qu'elles puissent être traitées par la partie Contrôleur.
- Vue : Cette partie est dédiée à l'affichage sur l'écran. Elle est généralement composée d'un mélange de code HTML et PHP.

- Contrôleur : Cette partie contient la portion « intelligente » du code. Elle reçoit les données du visiteur, demande au Modèle de trouver les données correspondantes dans la base de données, analyse les données fournies par le Modèle et décide ce qui doit être affiché par la partie Vue.

2. Conception détaillée

2.1. Les vues de Kruchten

Pour pouvoir passer à la conception nous allons choisir la démarche de Kruchten qui propose différentes perspectives indépendantes et complémentaires pour mieux cerner l'architecture logicielle choisie à savoir MVC. Le langage utilisé sera le langage UML qui propose 13 diagrammes différents qui seront utilisés selon le besoin et répartis selon les vues de Kruchten.

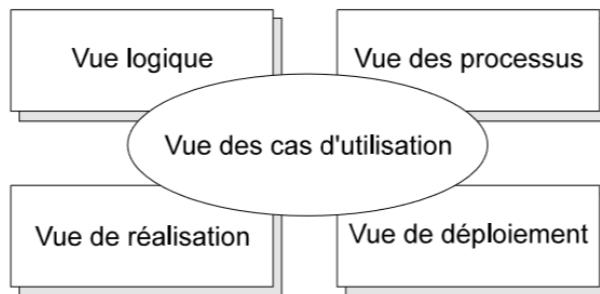


Figure 47: Vue 4+1 (Kruchten) [10]

La vue des cas d'utilisation

Nous avons présenté cette vue dans le chapitre 2.

La vue logique

Elle constitue la principale description architecturale d'un système informatique et beaucoup de petits projets se contentent de cette seule vue. Cette vue décrit, de façon statique et dynamique, le système en termes d'objets et de classes. La vue logique permet d'identifier les différents éléments et mécanismes du système à réaliser.

Cette vue présente

- ❖ Diagramme de classe.
- ❖ Diagramme d'objet.

La vue des processus

Elle décrit les interactions entre les différents processus, threads (fils d'exécution) ou tâches, elle permet également d'exprimer la synchronisation et l'allocation des objets. Cette vue permet avant tout de vérifier le respect des contraintes de fiabilité, d'efficacité et de performances des systèmes multitâches.

Les diagrammes utilisés dans la vue des processus sont exclusivement dynamiques

- ❖ Diagrammes d'activités
- ❖ Diagramme de séquence
- ❖ Diagrammes de communication ou d'états-transitions.

La vue des composants (ou réalisation)

Elle permet de visualiser l'organisation des composants (bibliothèque dynamique et statique, code source...) dans l'environnement de développement. Elle permet aux développeurs de se retrouver dans le capharnaüm que peut être un projet de développement informatique. Cette vue permet également de gérer la configuration (auteurs, versions...).

Les seuls diagrammes de cette vue sont les diagrammes de composants.

La vue de déploiement

Elle présente le système dans son environnement d'exécution. Elle traite des contraintes géographiques (distribution des processeurs dans l'espace), des contraintes de bandes passantes, du temps de réponse et des performances du système ainsi que de la tolérance aux fautes et aux pannes. Cette vue est fort utile pour l'installation et la maintenance régulière du système. []

Les diagrammes de cette vue sont

- ❖ Les diagrammes de composants
- ❖ Les diagrammes de déploiement.

2.2. Conception détaillée de la Partie client

2.2.1. Conception de cas d'utilisation « commander un médicament »

Vue logique

La figure 48 présente une portion de classes participantes au développement de cas d'utilisation « commander un médicament ».

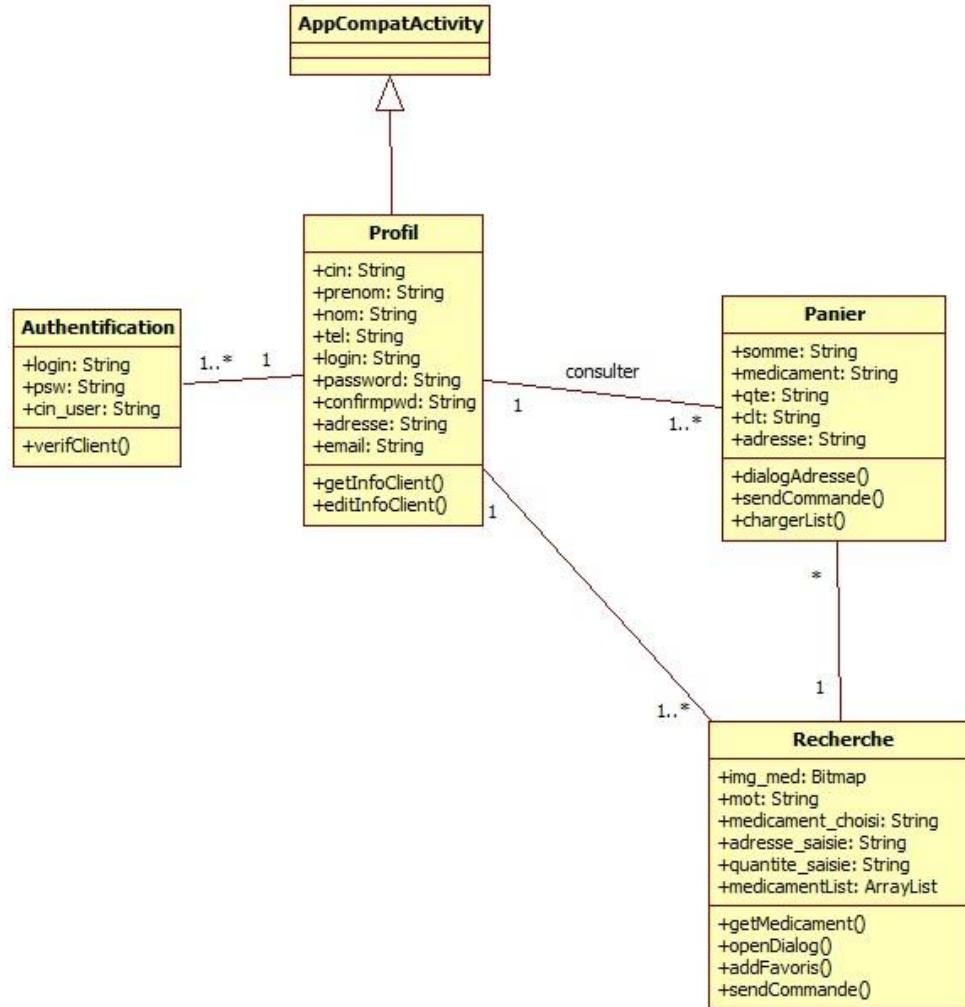


Figure 48: diagramme de classe de cas d'utilisation <<commander médicament>>

Le tableau 1 présente les fonctionnalités de chaque classe

Classe	Fonction
Authentification	C'est la classe qui contient l'identifiant et le mot de passe du client.
Profil	C'est la classe qui contient toutes les informations concernant le client.
Recherche	Cette classe rassemble la liste des médicaments.
Panier	Cette classe contient la liste des médicaments ajoutés par le client pour commander.

Tableau 8:Tableau des fonctions de chaque classe

Vue de composants

La figure 49 présente la vue de composants de cas d'utilisation « commander médicament ».

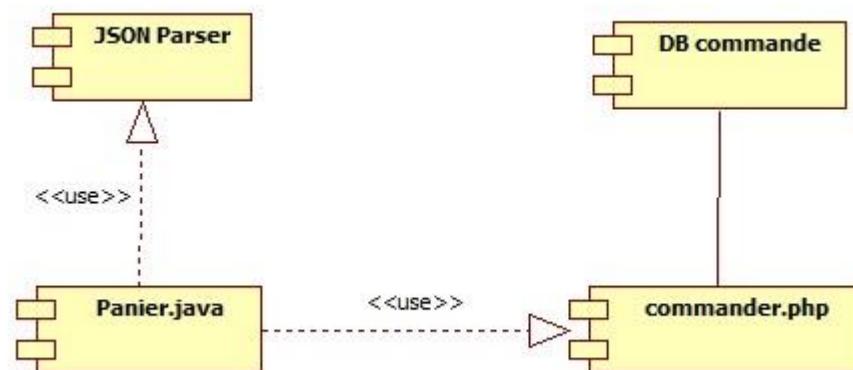


Figure 49:Vue de composants de cas d'utilisation <<commander médicament>>

Vue de processus

Chapitre 3 : Conception

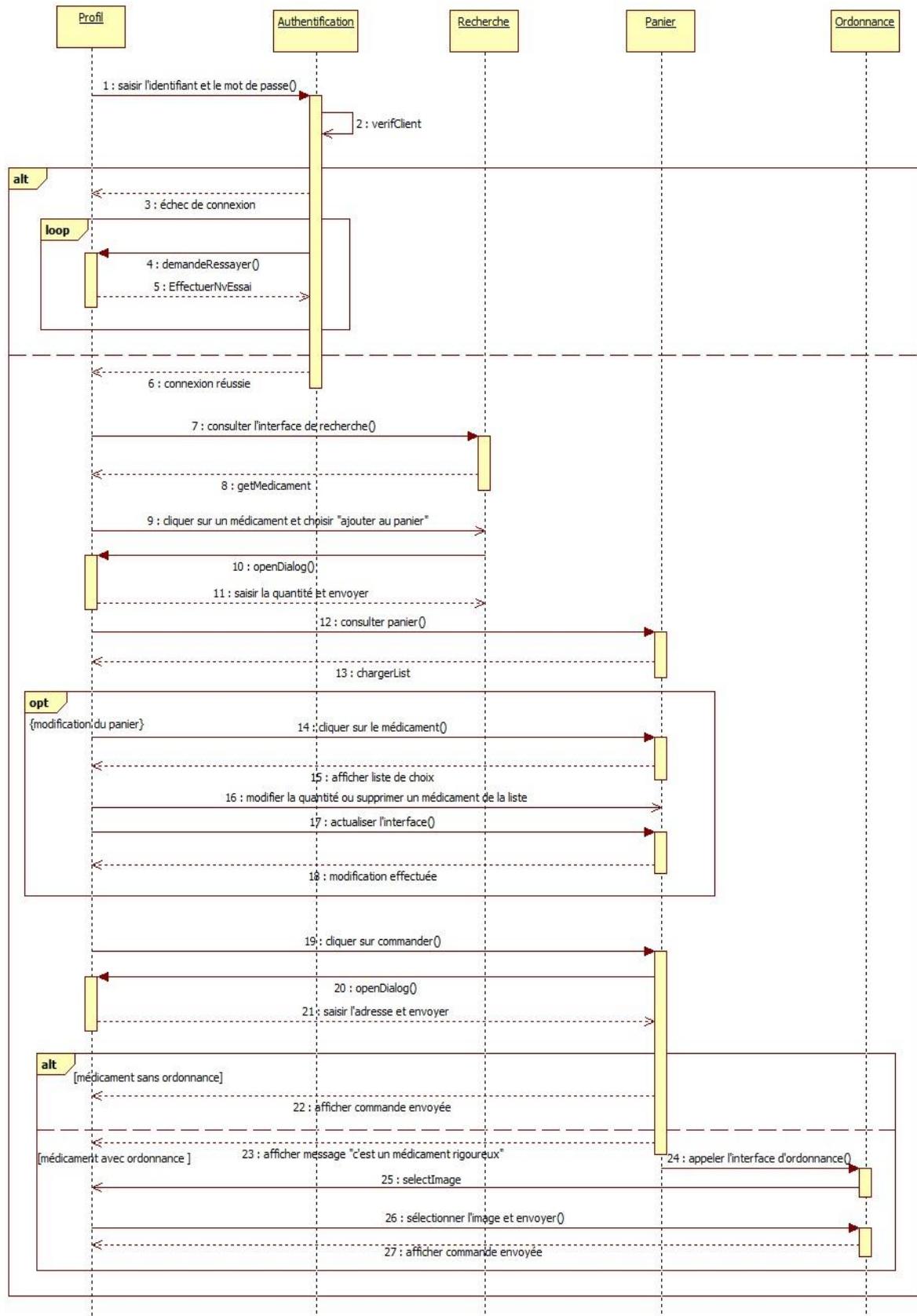


Figure 50: Diagramme de séquence << commander un médicament>>

La figure 50 présente le processus de la commande « commander un médicament », comme le montre la figure, le client passe par l’authentification à l’interface « panier » à travers l’interface « Recherche », si le médicament est rigoureux, le client doit passer à l’interface « ordonnance », ainsi la commande est envoyée.

Vue de déploiement

La figure 51 présente la vue de déploiement de cas d’utilisation « commander un médicament ».

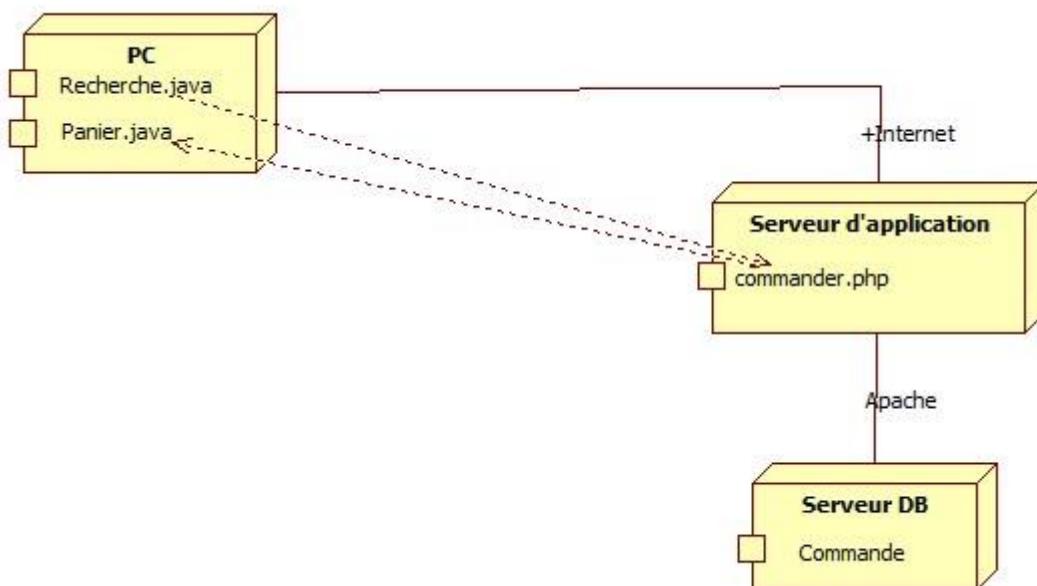


Figure 51: vue de déploiement de cas d'utilisation <<commander un médicament>>

2.2.2. Conception de cas d’utilisation « Consulter diagnostics »

Vue logique

La figure 52 présente une portion de classes participantes au développement de cas d’utilisation « consulter diagnostics ».

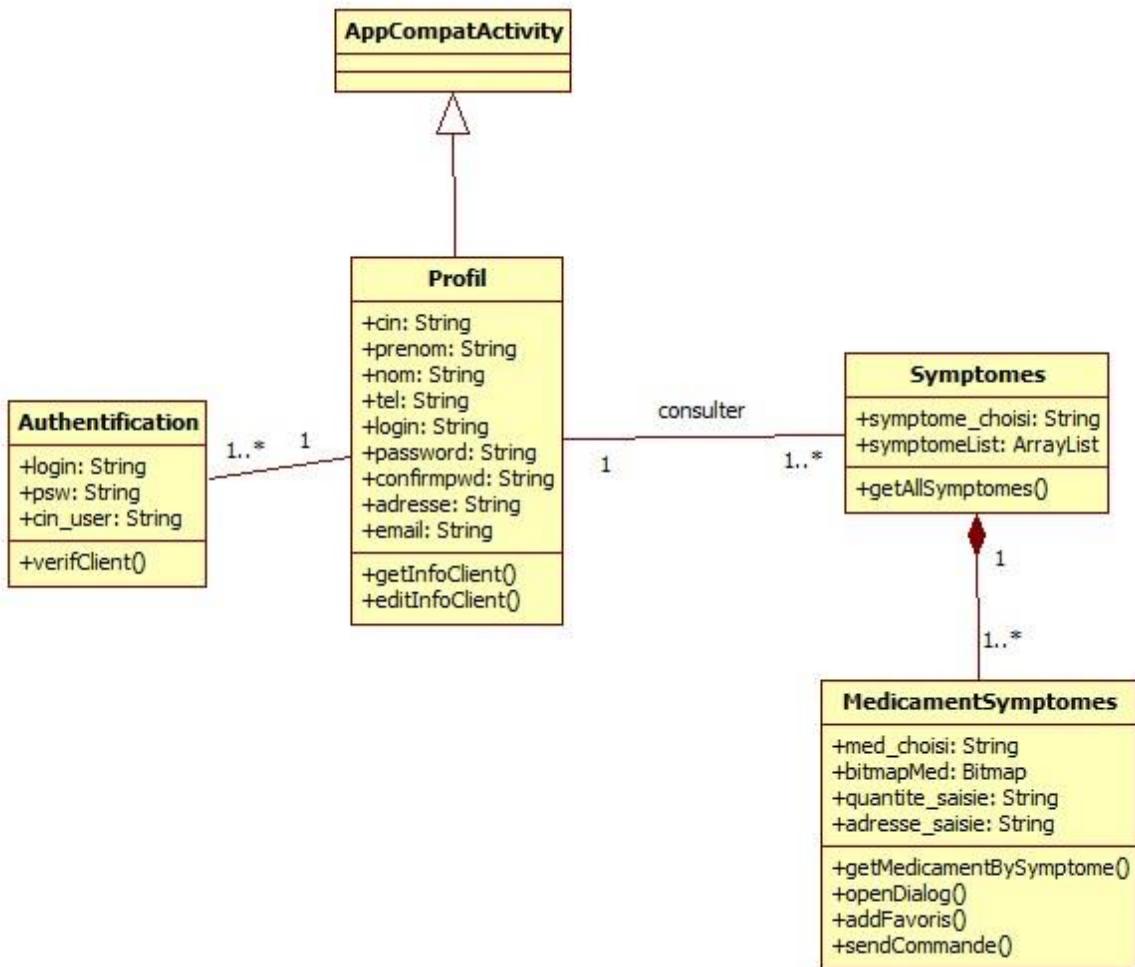


Figure 52: Diagramme de classe de cas d'utilisation << consulter diagnostics>>

Le tableau 2 présente les fonctionnalités de chaque classe

Classe	Fonction
Symptomes	Cette classe rassemble la liste des symptômes.
MedicamentsSymptome	Cette classe contient les diagnostics d'un tel symptôme.

Tableau 9:les fonctions de chaque classe

Vue des composants

La figure 54 présente la vue de composants de cas d'utilisation « consulter diagnostics ».

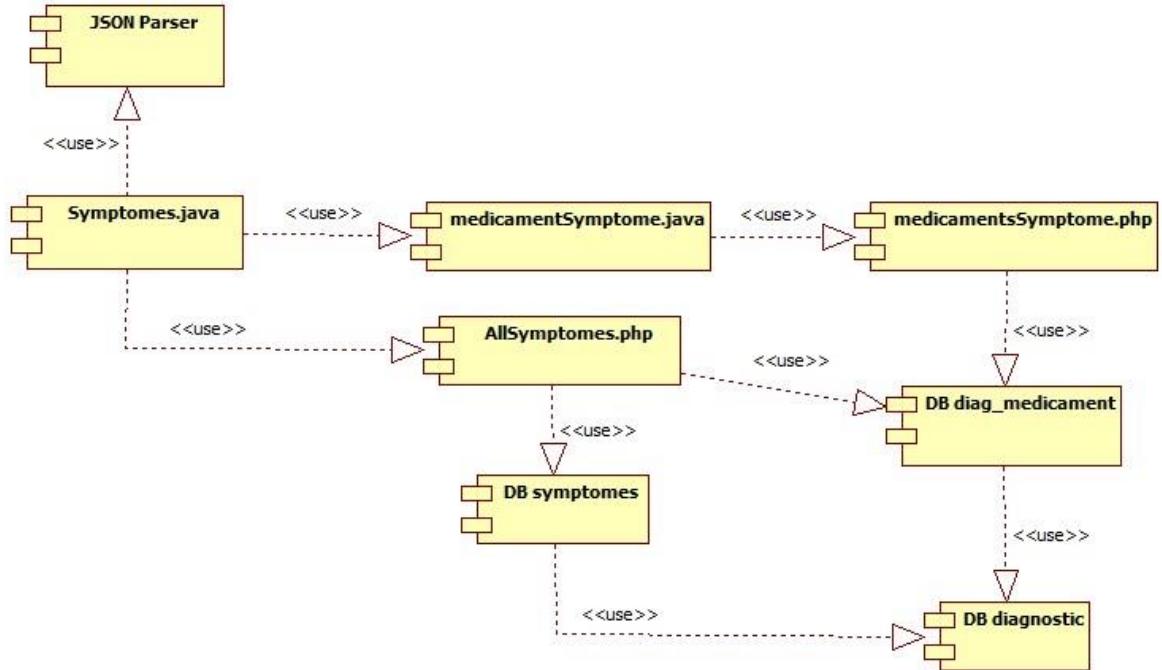


Figure 53: Vue de composants de cas d'utilisation << consulter diagnostics>>

Vue de processus

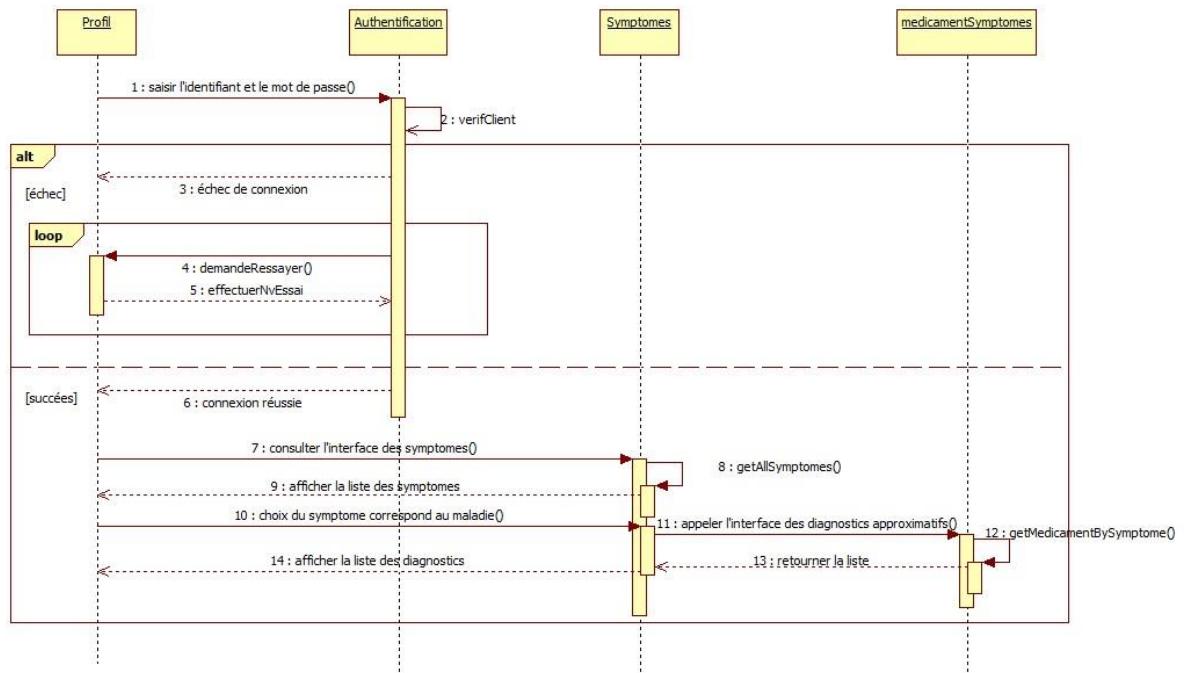


Figure 54: Diagramme de séquence << consulter diagnostics>>

La figure 55 présente le diagramme de séquence « consulter diagnostics ». Après avoir s'authentifier, le client consulte l'interface des symptômes qui appelle la méthode getAllSymptomes() de la classe Symptomes, puis, le client choisit le symptôme correspondant à son maladie ce qui provoque l'appel de l'interface des diagnostics qui appelle de son tour la méthode getMedicamentBySymptome(). La réponse est retournée au client.

Vue de déploiement

La figure 56 présente la vue de déploiement de cas d'utilisation « consulter diagnostics ».

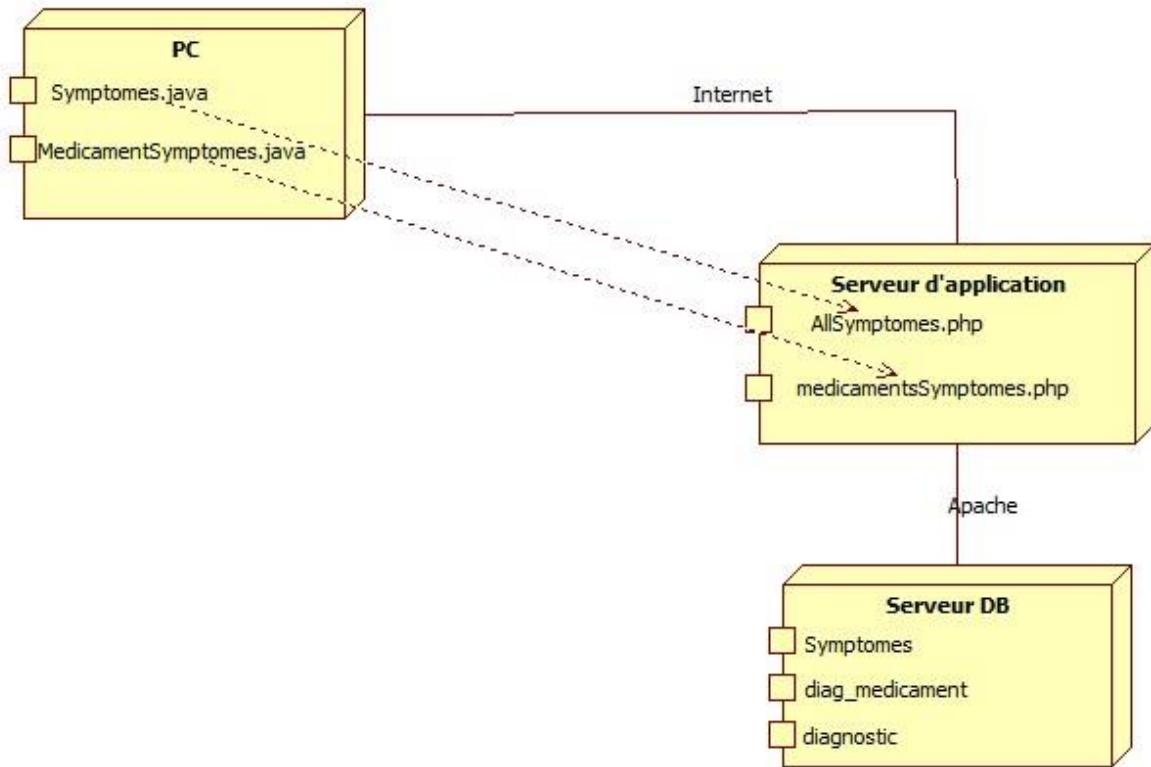


Figure 55: Vue de déploiement de cas d'utilisation << consulter diagnostics>>

2.2.3. Vue logique

a. Diagramme de classe générale de la partie mobile en MVC

Le diagramme de classes est considéré comme le plus important de la modélisation orientée objet.

Il permet de fournir une représentation abstraite des objets du système qui vont interagir ensemble pour réaliser les cas d'utilisation. Il s'agit d'une vue statique car on ne tient pas compte du facteur temporel dans le comportement du système.

La figure 57 illustre le diagramme de classes de la partie mobile.

Chapitre 3 : Conception

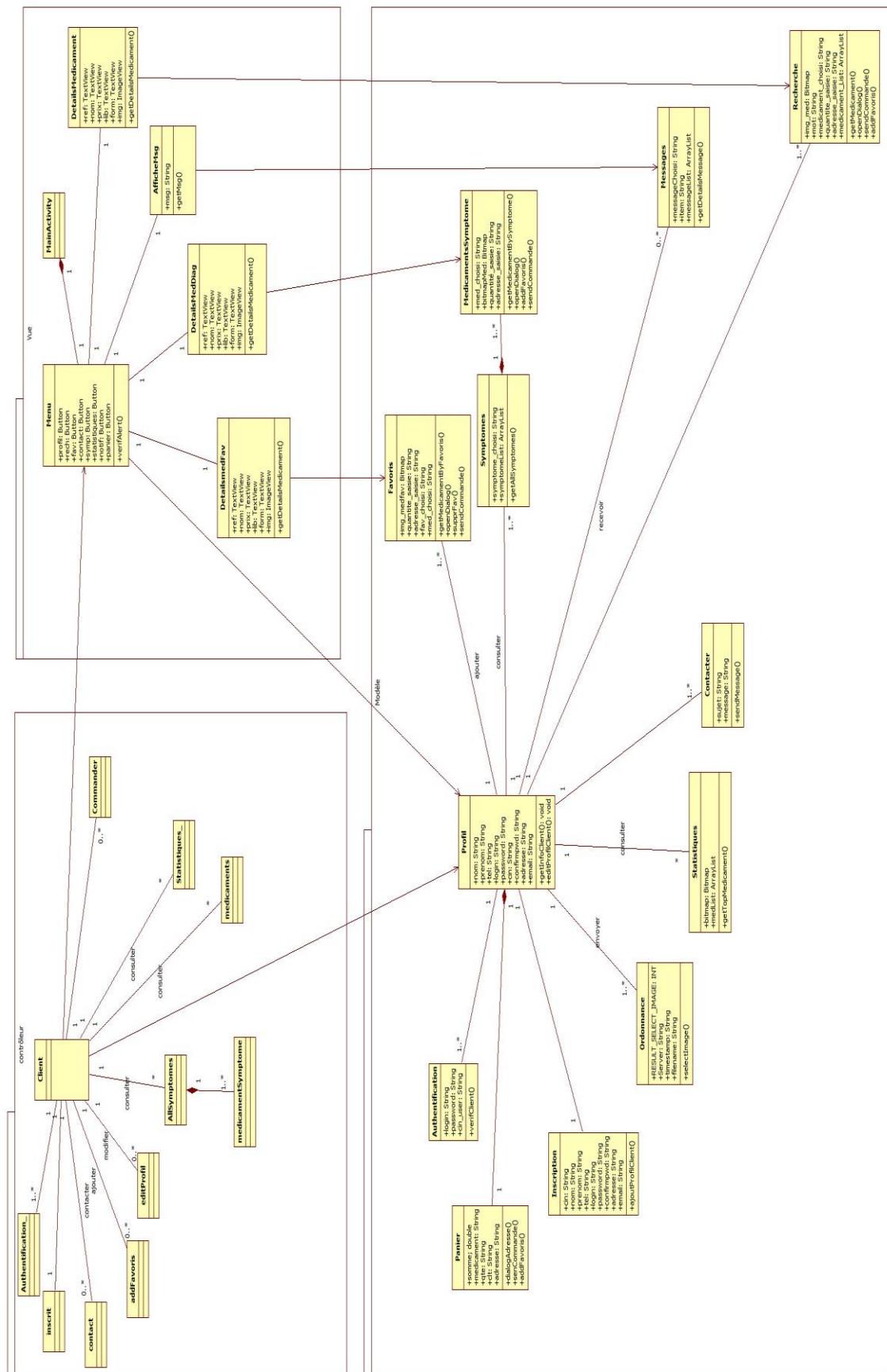


Figure 56: Diagramme de classes de la partie mobile en MVC

i. Explication des classes générées par défaut dans l'application pour la partie modèle

- **AppCompatActivity**

Classe de base pour les activités qui utilisent les fonctionnalités de la barre d'action de la bibliothèque de support.

- **JSON Parser**

JSON signifie JavaScript Object Notation. Il est un format d'échange de données indépendant et est la meilleure alternative pour XML.

Android fournit quatre classes différentes pour manipuler les données JSON. Ces classes sont JSONArray, JSONObject, JSONStringer et JSONTokenizer.

- **ViewBinder**

Cette classe est utilisée pour lier les valeurs du Cursor aux vues qui ne sont pas directement prises en charge par SimpleCursorAdapter ou pour modifier la manière dont la liaison se produit pour les vues prises en charge par SimpleCursorAdapter.

- **Request**

Cette classe est utilisée pour faire la demande Http sur le Web.

ii. Explication des classes et méthodes de l'application de la partie modèle

Le tableau 3 présente les fonctionnalités des classes restantes de la partie modèle.

Classe	Fonction
Inscription	C'est la classe qui contient les informations d'un nouveau client.
Ordonnance	C'est la classe qui contient l'image de l'ordonnance à envoyer.
Favoris	Cette classe rassemble les médicaments qui sont ajoutés aux favoris.
Messages	Cette classe contient la liste des messages reçus.
Contacter	C'est la classe qui contient le sujet et le message à envoyer.
Statistiques	Cette classe rassemble la liste des médicaments statistiques.

Panier	Cette classe contient la liste des médicaments ajoutés par le client pour commander.
--------	--

Tableau 10:Tableau des fonctions de chaque classe

Les tableaux suivants présentent les fonctions des différentes méthodes des classes.

Classe	Méthode	Fonction
Profil	getInfoClient()	Obtenir les informations du client
	editProfilClient()	Cette méthode offre au client la possibilité de modifier ses informations personnelles.
Authentification	verifClient()	Obtenir l'identifiant et le mot de passe du client.
Inscription	ajoutProfilClient()	Saisir les informations d'un nouveau client.
Ordonnance	selectImage()	Sélectionner une image depuis l'appareil mobile.
Favoris	getMedicamentByFavoris()	Obtenir les médicaments qui sont ajoutés aux favoris.
	openDialog()	Afficher une boite du dialogue.

Chapitre 3 : Conception

	Supprfav()	Supprimer un médicament de la liste des favoris.
	sendCommande()	Saisir les détails de la commande.
Recherche	getMedicament()	Cette méthode offre la possibilité au client de consulter les médicaments existants.
	openDialog()	Afficher une boite du dialogue.
	sendCommande()	Saisir les détails de la commande.
	addFavoris()	Ajouter un médicament aux favoris.
Messages	getDetailsMessage()	Obtenir les détails du message.
Contacter	sendMessage()	Saisir le message.
Statistiques	getTopMedicament()	Obtenir les médicaments les plus commandés par les clients.
Symptome	getAllsymptome()	Obtenir les symptômes.
MedicamentsSymptome	getMedicamentBySymptome()	Obtenir les médicaments d'un symptôme sélectionné.
	openDialog()	Afficher une boite du dialogue.

	sendCommande()	Saisir les détails de la commande.
	addFavoris()	Ajouter un médicament aux favoris.
Panier	dialogAdresse()	Afficher une boite du dialogue.
	sendCommande()	Cette méthode offre la possibilité au client de commander un ou plusieurs médicaments.
	ChargerList()	Cette méthode permet de charger la liste des médicaments du panier avec ses détails.

Tableau 11:Tableau de définition des méthodes de chaque classe

iii. Explication des classes pour la partie vue

Classe	Fonction
Menu	Cette classe rassemble les différentes options dont le client peut choisir entre eux.
MainActivity	C'est la classe qui contient la 1 ^{ère} interface de l'application.
DetailsMedicament	C'est la classe qui rassemble les détails d'un médicament recherché.
AfficheMsg	C'est la classe qui affiche un message reçu.

DetailsmedDiag	Cette classe contient les détails d'un médicament des diagnostics.
DetailsmedFav	Cette classe contient les détails d'un médicament de la liste des favoris.

Tableau 12:Tableau de définition des classes de la partie vue

iv. Explication des classes pour la partie contrôleur

Classe	Fonction
Client	C'est la classe qui récupère les informations du client de la base des données.
Authentification	C'est la classe qui permet de récupérer le login et le mot de passe du client de la base des données.
Commander	C'est la classe qui enregistre la commande dans la base des données.
Statistiques	C'est la classe qui sélectionne les médicaments les plus commandés dans la base des données.
Medicaments	C'est la classe qui récupère les informations du médicament de la base des données.
Allsymptomes	C'est la classe qui récupère tous les symptômes enregistrés dans la base des données.
Medicament_symptome	C'est la classe qui récupère tous les médicaments d'un symptôme.
editProfil	C'est la classe qui modifie les informations du client dans la base des données.
addFavoris	C'est la classe qui permet d'ajouter un médicament dans la table « favoris » dans la base des données.
Contact	C'est la classe qui permet d'enregistrer un nouveau message du client dans la base des données.
Inscrit	C'est la classe qui permet d'ajouter les informations d'un nouveau client dans la base des données.

Tableau 13:Tableau de définition des classes de la partie contrôleur

2.3. Conception de la base des données

2.3.1. Dictionnaire des données

Un dictionnaire des données est une collection de métadonnées ou de données de référence nécessaire à la conception d'une base de données relationnelle. Il revêt une importance stratégique particulière, car il est le vocabulaire commun de l'organisation.

Nom	Désignation	Type	Longueur	Identifiant
IdCl	Identifiant	Numérique	25	Oui
CIN	Numéro CIN	Numérique	25	Non
nomCl	Nom du client	chaine	255	Non
prenomCl	Prénom du client	Chaine	255	Non
tel	Numéro du téléphone	Numérique	25	Non
mail	Adresse email	Chaine	255	Non
ad	Adresse du client	Chaine	255	Non
login	Mot pour se connecter	Chaine	255	Non
pw	Mot de passe	chaine	255	Non
image	Image du client	chaine	100	Non

Tableau 14:table Client

Nom	Désignation	Type	Longueur	Identifiant
idCm	Identifiant de la commande	Numérique	25	Oui
adresse_client	Adresse du client	Chaine	255	Non
dateC	La date de la commande	date	-	Non

Chapitre 3 : Conception

quantite	La quantité du médicament	Numérique	11	Non
cin_client	L'identifiant du client	Numérique	25	Oui
Id_med	L'identifiant du médicament	Numérique	11	Oui
etat	Etat de la commande	Numérique	1	Non
ordonnance	Ordonnance du médicament	Chaine	255	Non

Tableau 15:table Commande

Nom	Désignation	Type	Longueur	Identifiant
Id_msg	L'identifiant du message	Numérique	25	Oui
Date	Date du l'envoi	Date	-	Non
Sujet	Sujet du message	Chaine	255	Non
Message	Contenu du message	Chaine	255	Non
CIN	L'identifiant du client	Numérique	25	Oui
Er	L'envoyeur du message	Chaine	255	Non
lu	L'état du message	Numérique	1	Non

Tableau 16:table message

Nom	Désignation	Type	Longueur	Identifiant
IdD	L'identifiant du diagnostic	Numérique	25	Oui

Chapitre 3 : Conception

Id_sym	L'identifiant du symptôme	Chaine	255	Non
--------	---------------------------	--------	-----	-----

Tableau 17:table diagnostic

Nom	Désignation	Type	Longueur	Identifiant
Id_favoris	L'identifiant du favori	Numérique	11	Oui
Id_med	L'identifiant du médicament	Numérique	11	Oui
Cin_client	L'identifiant du client	Numérique	11	Oui

Tableau 18:table favoris

Nom	Désignation	Type	Longueur	Identifiant
id_med	L'identifiant du médicament	Numérique	11	Oui
nom_med	Nom du médicament	Chaine	50	Non
Ref	La référence du médicament	Chaine	12	Non
Lib	Libellé du médicament	Chaine	255	Non
Forme	La forme du médicament	Chaine	255	Non
Prix	Le prix du médicament	Chaine	5	Non
Image	L'image du médicament	Chaine	40	Non
Qt	La quantité du médicament	Numérique	11	Non
Type	Type du médicament	Numérique	11	Non
Need_ordo	Type d'ordonnance	Numérique	1	Non

Chapitre 3 : Conception

Tableau 19:table médicament

Nom	Désignation	Type	Longueur	Identifiant
IdDM	L'identifiant du médicament diagnostic	Numérique	25	Oui
IdD	L'identifiant du diagnostic	Numérique	25	Oui
idM	L'identifiant du médicament	Numérique	25	Oui

Tableau 20:table diag_medicament

Nom	Désignation	Type	Longueur	Identifiant
Num_ordonnance	L'identifiant du l'ordonnance	Numérique	11	Oui
Cin_client	L'identifiant du client	Numérique	11	Oui
Date_envoie	La date du l'envoi	Date	-	Non
fichier	Le fichier contenant l'ordonnance	Chaine	150	Non

Tableau 21:table ordonnance

Nom	Désignation	Type	Longueur	Identifiant
idP	L'identifiant du pharmacien	Numérique	25	Oui
Login	Mot pour se connecter	Chaine	255	Non
pw	Mot de passe	Chaine	255	Non
role	Role du pharmacien	Chaine	255	Non

Tableau 22:table pharmacien

Nom	Désignation	Type	Longueur	Identifiant
Id_sym	L'identifiant du symptôme	Numérique	25	Oui

Nom	Désignation du symptôme	Chaine	255	Non
-----	-------------------------	--------	-----	-----

Tableau 23:table symptome

2.3.2. Modèle conceptuel des données

Le modèle conceptuel des données (MCD) a pour but d'écrire de façon formelle les données qui seront utilisées par le système d'information. Il s'agit donc d'une représentation des données, facilement compréhensible, permettant de décrire le système d'information à l'aide d'entités.

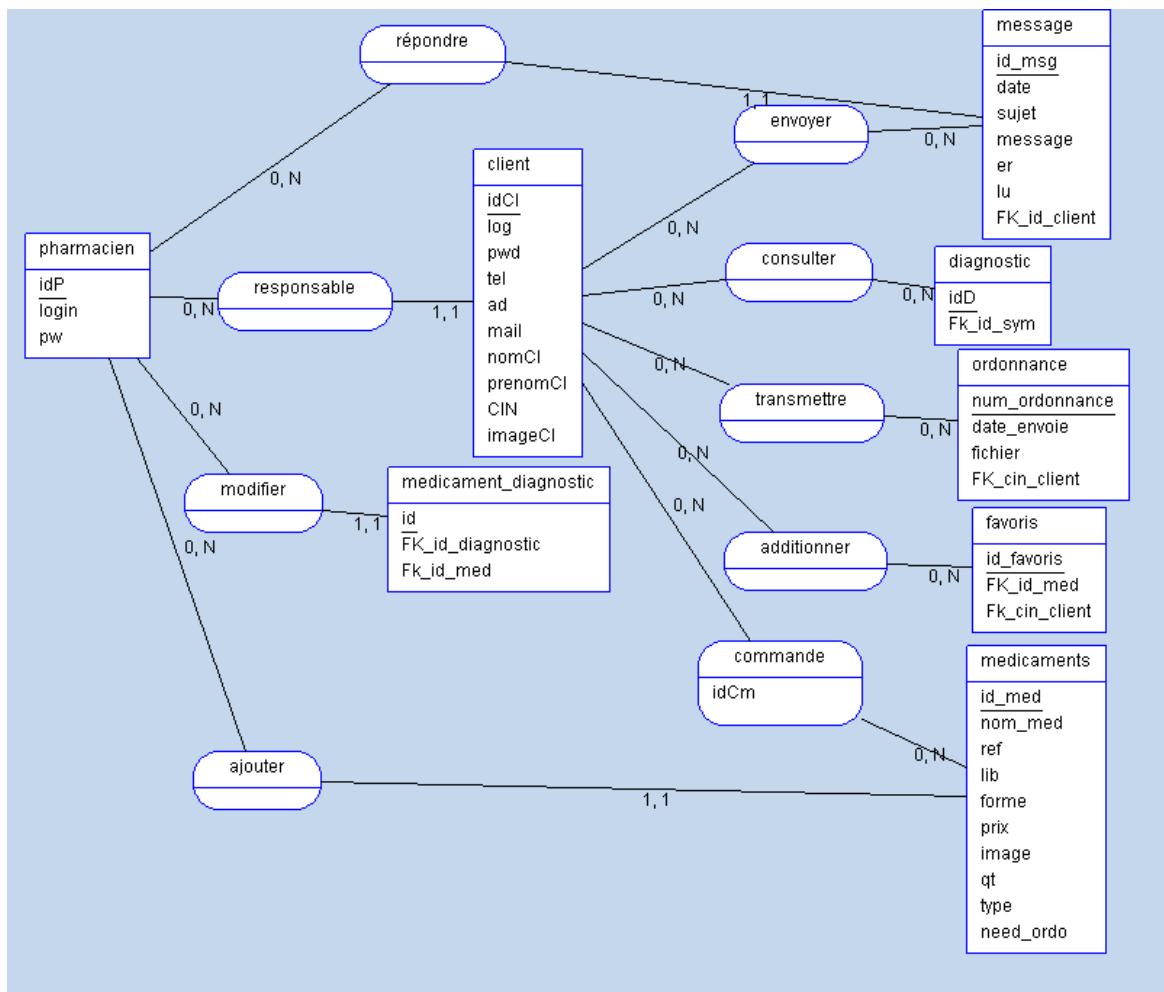


Figure 57: modèle conceptuel des données

La figure 58 présente le MCD du PharmaCity, qui contient tous les relations entre les tables de notre base des données.

Conclusion

Dans ce chapitre, nous avons présenté notre étude conceptuelle du système. , la vue statique, réalisée par le diagramme de classes nous a permis de définir la structure du système et de dégager les différentes entités des composants.



Chapitre4 : Réalisation

Introduction

Après avoir élaboré la conception de notre application, nous abordons dans ce chapitre le dernier volet de ce rapport, qui a pour objectif d'exposer la phase de réalisation.

La phase de réalisation est considérée comme étant la concrétisation finale de toute la méthode de conception.

Dans ce chapitre nous détaillerons d'abord les outils de développement adoptés pour la réalisation de notre système. Ensuite nous présenterons quelques interfaces et fenêtres de l'application.

1. Environnement de développement

Il s'agit dans cette partie d'identifier les différentes caractéristiques de l'environnement logiciel qui nous ont servi à l'implémentation de notre application.

1.1. Wamp Server

WampServer est une plate-forme de développement Web sous Windows pour des applications Web dynamiques à l'aide du serveur Apache2, du langage de scripts PHP et d'une base de données MySQL. Il possède également PHPMyAdmin pour gérer plus facilement la base de données. [11]

1.2. NotePad++

Editeur simple pour l'écriture du code PHP.

1.3. Android Studio

Android Studio est un environnement de développement pour développer des applications Android. Il est basé sur IntelliJ IDEA. [12]

1.4. Genymotion

Genymotion est un émulateur Android basé sur VirtualBox. Genymotion a l'avantage d'être plus rapide que l'émulateur du Kit de développement Android et fournit de meilleures performances générales. [13]

2. Architecture du site web

A travers l'espace d'administration, l'administrateur du site peut :

- Gérer les administrateurs
- Gérer les médicaments
- Gérer les diagnostics
- Gérer les clients
- Gérer les commandes
- Gérer les messages

La figure 59 illustre l'architecture du site du l'administrateur.

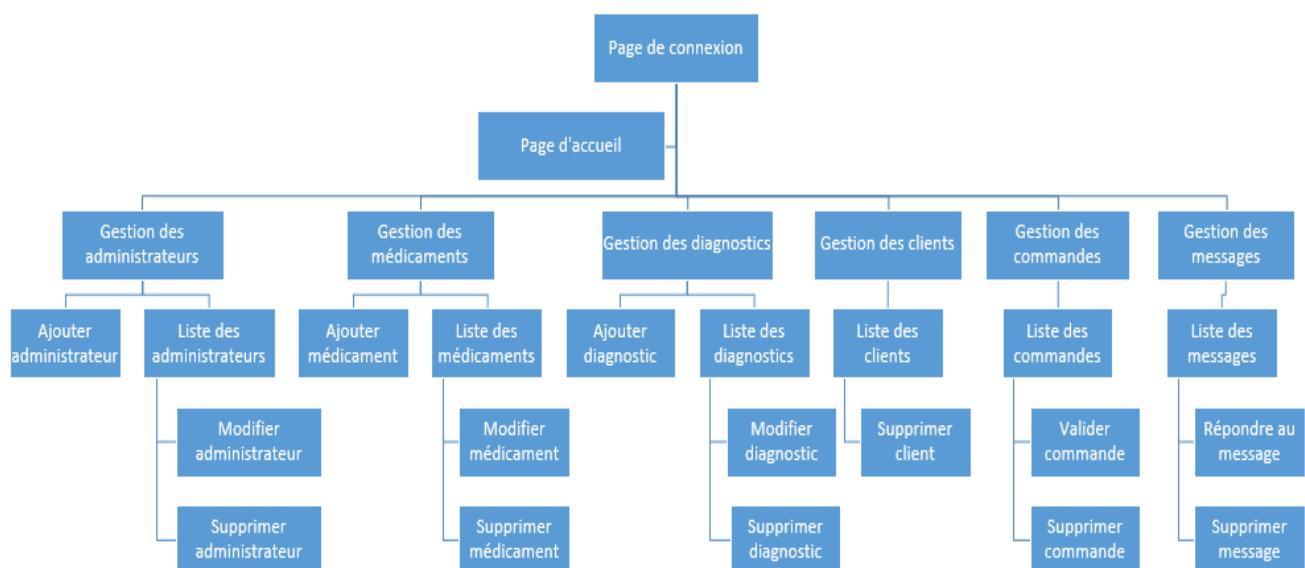


Figure 58: Architecture du site

3. Composition des modules

Au début de développement de site web et de l'application mobile, nous avons décomposé notre système en douze grands modules.

- **Connexion et Inscription**

Le client peut se connecter ou créer un compte

- **Gestion des administrateurs**

Le super administrateur peut ajouter un sous administrateur.

- **Gestion des clients**

L'administrateur peut supprimer un client ou consulter la liste des clients.

- **Profil utilisateur**

Le client peut consulter son profil et le gérer à partir de l'application mobile.

- **Gestion des médicaments**

Le client peut faire une recherche d'un médicament.

L'administrateur peut ajouter, consulter, modifier ou supprimer un médicament.

- **Gestion des commandes**

Le client peut commander un médicament.

L'administrateur valide les commandes reçus.

- **Gestion des diagnostics**

Le client peut consulter les diagnostics.

L'administrateur peut ajouter, modifier, consulter ou supprimer les diagnostics.

- **Gestion des favoris**

Le client peut ajouter, consulter ou supprimer un favori à partir de l'application mobile.

- **Gestion des messages**

Le client peut envoyer un message ou consulter la liste des messages.

L'administrateur peut répondre aux messages.

- **Gestion des notifications**

Le client peut recevoir une notification pour lui rappeler de commander un médicament.

- **Gestion des ordonnances**

Le client peut envoyer une ordonnance.

L'administrateur peut consulter les ordonnances.

- **Statistiques**

Le client peut consulter les statistiques des médicaments les plus commandés.

La composition de ces modules est illustrée dans la figure 60.

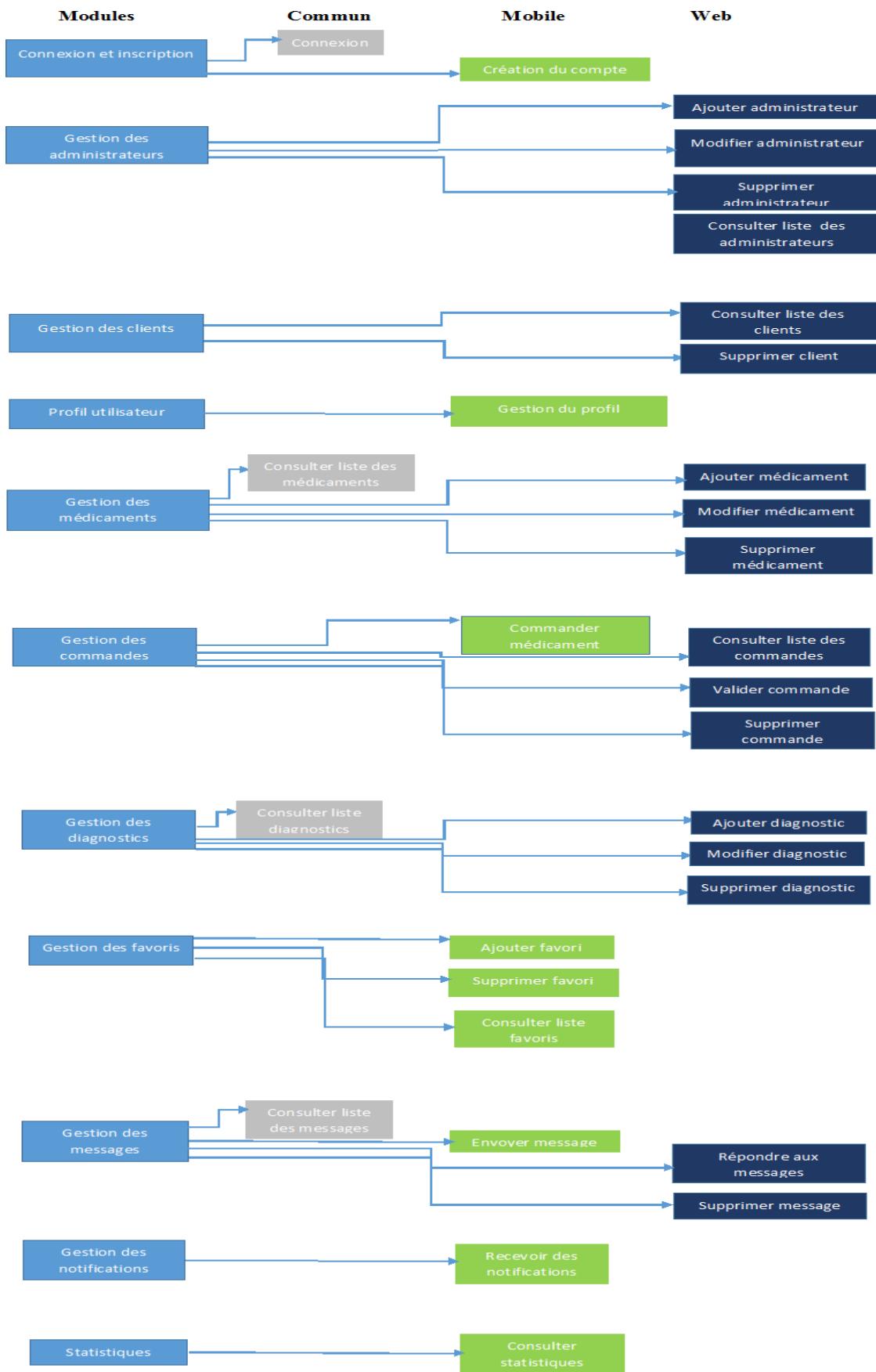


Figure 59: Composition des modules

4. Présentation de la solution mobile

Dans cette section, nous allons présenter quelques interfaces de notre application, en essayant à chaque fois de décrire les différents objets interactifs mis à la disposition de l'utilisateur. Dans la figure 61, nous présentons d'abord le logo de l'application.



Figure 60:Logo de la solution mobile <<PharmaCity>>

4.1. Interface de lancement de l'application et de l'authentification

Lors du lancement du l'application, une interface de bienvenue présentée par la figure 62 apparait mentionnant le nom du l'application. Le client est invité à s'authentifier en saisissant ses paramètres d'accès. La figure 63 illustre l'interface authentification.



Figure 61:Interface de lancement de l'application



Figure 62:Interface de l'authentification

4.2. Interface d'ajout d'un compte utilisateur

Lorsque le client n'admet pas de compte, il pourrait accéder aux fonctionnalités d'ajout d'un compte tout en saisissant quelques informations personnelles et les paramètres d'accès. La figure 64 illustre l'interface d'ajout d'un client



Figure 63:Interface d'inscription

4.3. Interface de menu de l'application

La figure 65 présente une interface de transition où le client doit choisir entre:

- Gérer son profil
- Faire une recherche d'un médicament
- Consulter les diagnostics
- Contacter le pharmacien
- Consulter ses messages
- Consulter ses notifications
- Envoyer une ordonnance au pharmacien
- Consulter son panier
- Consulter les statistiques
- Consulter les favoris
- Consulter les informations à propos du l'application



Figure 64:Interface de menu de l'application

4.4. Interface de profil du client

Lorsque le client choisit de gérer son profil, il visualise d'abord une liste contenant toutes ses informations personnelles dont il peut les modifier en respectant les droits de chaque champ. En cliquant sur modifier, le profil sera mis à jour.

La figure 66 illustre cette interface.



Figure 65:Interface de profil du client

4.5. Interface de la recherche des médicaments

La figure 67 illustre une interface de recherche d'un médicament. Lorsque le client écrit une lettre, une liste de médicament commençant par cette lettre s'affiche. En cliquant sur un médicament, le client peut choisir entre soit « ajouter au panier », soit « ajouter aux favoris », soit « plus détails ». La figure 68 présente l'interface des détails du médicament.



Figure 66:Interface de recherche des médicaments

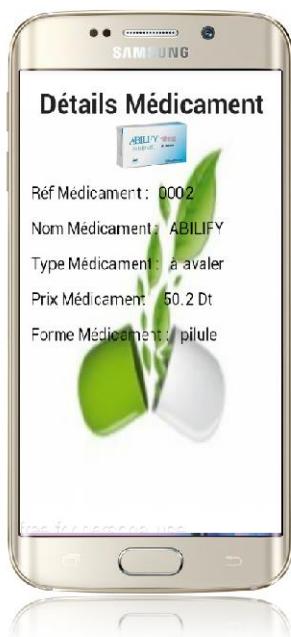


Figure 67:interface de détails d'un médicament

4.6. Interface des symptômes

Lorsque le client choisit de consulter les diagnostics, il visualise d'abord une liste de symptômes dont il peut choisir entre eux un symptôme correspond à son maladie. Ce cas est illustré par la figure 69.

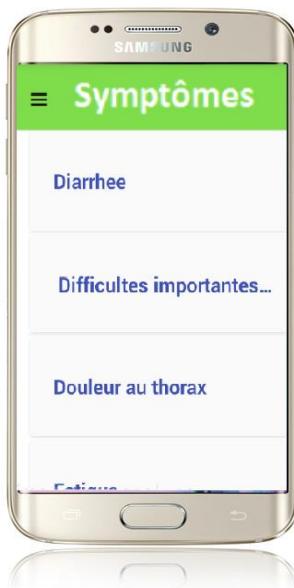


Figure 68:Interface des symptômes

4.7. Interface des diagnostics

Lorsque le client choisit un symptôme, il visualise une liste de médicaments corresponds au symptôme qu'il choisit. La figure 70 illustre cette interface.



Figure 69:Interface des diagnostics

4.8. Interface de l'envoi d'un message

La figure 71 présente l'interface de l'envoi d'un message. Le client choisit un sujet, il écrit son message ensuite il clique sur « envoyer ».

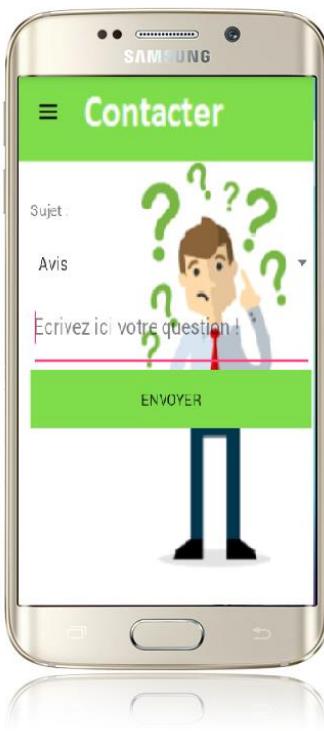


Figure 70:Interface de l'envoi d'un message

4.9. Interfaces de la boîte de réception des messages

Les figures 72 et 73 montrent que le client peut consulter ses messages avec les détails. En cliquant sur un message, il visualise son message complet.

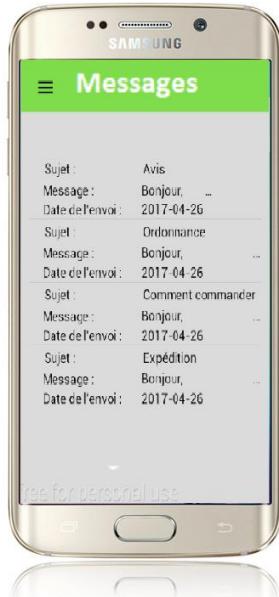


Figure 71:Interface des messages



Figure 72:Interface de l'affichage d'un message

4.10. Interface du panier des commandes

Le client peut ajouter des médicaments à son panier en saisissant la quantité d'abord. Il peut ensuite supprimer ou commander les médicaments. En cliquant sur commander, il doit saisir son adresse. Cette interface est présentée dans la figure 74. L'application doit aller à l'interface de l'envoi des ordonnances si un médicament commandé est rigoureux. Cette interface est illustrée dans la figure 75.



Figure 73:Interface de panier des commandes



Figure 74:Interface de l'envoi des ordonnances

4.11. Interfaces des statistiques des commandes des clients

La figure 76 illustre les statistiques des médicaments les plus commandés.



Figure 75:Interface des statistiques

4.12. Interface des médicaments favoris pour le client

La figure 77 présente la liste des médicaments favoris pour le client.

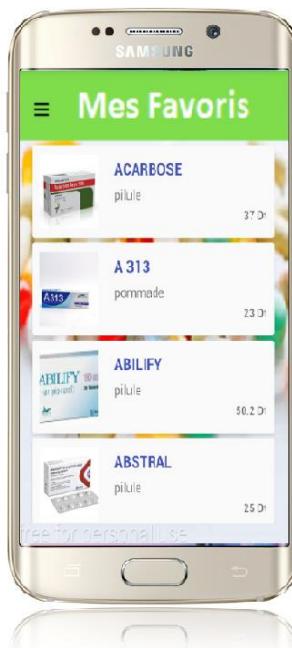


Figure 76:Interface des favoris

4.13. Interface à propos de l'application

La figure 78 présente le logo, le nom et quelques informations sur l'application.



Figure 77:Interface à propos de l'application

Conclusion

Ce chapitre nous a permis de choisir tout d'abord les environnements de travail logiciels les plus adaptés pour la création de notre application. Nous avons également décrit les différentes interfaces de notre application qui résument les fonctionnalités offertes. A présent nous passerons dans la partie suivante à la conclusion globale de notre projet.

Conclusion Générale

Ce projet de fin d'étude, effectué au sein de la société HelpSoft, consistait à développer la partie client et la partie Back-office de l'application mobile PharmaCity qui offre les services d'une pharmacie à l'utilisateur.

Ce rapport englobe toutes les étapes nécessaires pour la conception et le développement de cette application, voir l'étude de l'existant, l'analyse et la spécification des besoins, la conception et la réalisation dans chaque phase de ce rapport. .

L'élaboration de notre application nous a permis, en premier lieu, de nous confronter à une expérience professionnelle dans le domaine de développement des applications en utilisant le PHP et l'Android .

En deuxième lieu, de nous offrir l'opportunité de concrétiser nos connaissances théoriques acquises durant les années de nos études à l'ISIMM.

Ce projet nous a apporté le plus surtout pour la coté professionnelle puisque nous avons découvert plusieurs langages de programmation et une gamme très importante des nouvelles technologies. En plus la nature du produit permet de nous concentrer sur des caractéristiques différentes, de les améliorer et de créer de nouvelles fonctionnalités compatibles avec les anciens. Nous avons d'abord mis l'accent sur la généralisation, puis nous allons nous concentrer sur la distribution de notre application en déclenchant un site web qui offre les memes fonctionnalités que l'application mobile. Enfin, ce projet et comme tout projet de développement reste incomplet et peut avoir plusieurs amélioration.

Bibliographie

Bibliographie

Bibliographie

1. L'art du développement Android de Grant Allen
2. L'art du développement Android de Mark Murphy
3. UML 2 de l'apprentissage à la pratique de Laurent Audibert
4. UML 2 pour les bases de données de Christian Soutou
5. UML pour les décideurs

Auteur(s) : Franck Vallée avec la contribution de Gaël Thomas

6. PHP et MySQL de Luke Welling
7. PHP en action Solutions et exemples pour les développeurs PHP de David Sklar & Adam Trachtenberg

Netographie

www.wikipédia.org (site officiel)

www.primefaces.com (site officiel)

