

République Algérienne Démocratique et Populaire
Ministère de L'enseignement Supérieur et de la Recherche Scientifique
Université A/Mira de Béjaïa
Faculté des Sciences Exactes
Département Informatique



Mémoire de Fin de Cycle

En vue de l'obtention du diplôme Master en Informatique
Option : Administration et Sécurité des Réseaux

Thème

Conception et réalisation d'un
système informatique d'assistance à
distance pour les patients et la
vieillesse.
Cas d'étude : Bureau
d'hospitalisation à domicile du CHU
de Béjaïa.

Réalisé par :

M^r KHEBAT Racim.
M^r REZKELLAH Nassim.

Devant le jury composé de :

Président : M^{me} GHIDOUCHE K.
Examineur1 : M^r BEDJOU K.
Examinatrice2 : M^{lle} BOUCHELAGHEM S.
Promoteur : M^r AKILAL A.

PROMOTION 2016-2017

Dédicaces

Je dédie ce modeste travail

À mes chers parents, Pour tous vos sacrifices pour moi, nul mot ne saura exprimer mon amour envers vous. Que Dieu vous protège et vous accorde une longue vie,

À mon frère Lamine et ma sœur Emilia.

À tous ceux qui m'aiment, À tous ceux que j'aime Que cette modeste dédicace puisse vous témoigner ma profonde gratitude pour votre immense amour, votre confiance et vos paroles apaisantes qui m'ont toujours fait garder le sourire.

Racim

Je dédié ce travail, À mes très chers parents pour m'avoir apporté support et soutien, tant psychologique que financier, pendant toute la durée de mes études.

À mes frères et sœurs.

J'exprime mes sentiments les plus profonds à mes amis et je leurs dédié ce modeste travail.

Nassim

Remerciements

Nous exprimons nos vifs remerciements envers tous ceux en qui, par leur présence, leur soutien, leur disponibilité et leurs conseils, nous avons trouvé courage afin d'accomplir ce projet.

Nous tenons remercier vivement M. AKILAL Abdellah qui nous a fait l'honneur d'être notre encadreur. Nous le remercions profondément de son encouragement continu et aussi d'être toujours là pour nous écouter, nous aider et nous guider à retrouver le bon chemin par sa sagesse et ses précieux conseils, ce qui nous a donné la force et le courage d'accomplir ce projet.

Nos sincères remerciements s'adressent aussi au personnel du CHU de Béjaia.

Nous remercions tout particulièrement les membres de jury qui ont accepté de juger notre travail ainsi que tous les enseignants qui ont contribué à notre formation.

Liste des Acronymes

2TUP	2 Track Unified Process.
AJAX	Asynchronous JavaScript and XML.
ASP	Application Service Provider.
ARPANET	Advanced Research Projects Agency Network.
CHU	Centre Hospitalo-Universitaire.
CSS	Cascading Styles Sheet.
FTP	File Transfert Protocol.
HTML	Hyper Text Markup Language.
HTTP	Hyper Text Transfert Protocol.
IDE	Integrate Développement Environnement.
IP	Internet Protocol.
ISO	International Standard Organisation.
JDK	Java Développement Kit.
LAN	Local Area Network.
MAN	Metropolitan Area Network.
MySQL	MyOMG Query Language.
OSI	Open Systems Interconection.
PHP	Prés HyperText Processor.
SDF	Software Developer's Kit.
SGML	Standart Generalized Markup Language.
SGBD	Système Gestion Bases de Données.
SQL	Structured Query Language.
TCP	Transmission Control Protocol.
UML	Unified Modeling Language .
UP	Unified Process.
URL	Uniform Resource Locator.
XML	Xetensible Markup Language.

Table des matières

Liste des Acronymes	i
Table des matières	ii
Liste des figures	iv
Liste des tableaux	v
Introduction générale	1
1 Étude préliminaire et analyse des besoins	3
1.1 Introduction	3
1.2 Étude préliminaire	3
1.2.1 Présentation de l'organisme d'accueil	3
1.2.2 Présentation et mission du bureau d'hospitalisation à domicile	5
1.2.3 Problématiques	6
1.2.4 Les objectifs de l'application	6
1.3 L'analyse du personnel	7
1.3.1 Identifications des acteurs	7
1.3.2 Identifications des messages échangés	8
1.3.3 Diagramme de contexte	9
1.4 Capture des besoins	9
1.4.1 Les besoins fonctionnels	9
1.4.2 Les besoins techniques	16
1.5 conclusion	17
2 L'analyse et conception	18
2.1 Introduction	18
2.2 Diagramme de séquence	18
2.2.1 Diagramme de séquence du cas d'utilisation « Authentification »	19
2.2.2 Diagramme de séquence du cas d'utilisation « Faire des prélèvements »	20
2.2.3 Diagramme de séquence du cas d'utilisation « Gestion des prescriptions »	22
2.2.4 Diagramme de séquence du cas d'utilisation « Ajouter une prescription »	23
2.3 Diagramme d'activités	23
2.4 Diagramme de classes	25
2.4.1 Dictionnaire de données	27
2.4.2 Modèle relationnel	28
2.5 Conclusion	29

3	Réalisation	30
3.1	Introduction	30
3.2	Environnement de développement	30
3.2.1	Le Wamp Server	30
3.2.2	Eclipse	31
3.3	Les outils de développement d'application	31
3.3.1	Le PHP	31
3.3.2	AJAX	31
3.3.3	Le HTML	31
3.3.4	Le CSS	31
3.3.5	JavaScript	31
3.3.6	Bootstrap	32
3.3.7	Java	32
3.3.8	SDK	32
3.3.9	Visual Paradigm	32
3.3.10	Notepad++	32
3.3.11	Photoshop	32
3.4	Présentation des interfaces de l'application	33
3.4.1	Page d'authentification	33
3.4.2	Page de réinitialisation de mot de passe	33
3.4.3	Interface " Administrateur "	34
3.4.4	Interface " Médecin "	37
3.4.5	Interface " Patient "	41
3.4.6	Interface " Ambulancier "	44
3.5	Conclusion	44
	Conclusion générale	45
	Bibliographie	vi
A	Annexe A	i
A.1	Présentation du langage UML	i
A.1.1	Les diagrammes UML	i
A.2	Le processus unifié	ii
A.2.1	Définition	ii
A.2.2	Caractéristiques du processus unifié	ii
A.2.3	Différentes phases du processus unifié	ii
A.2.4	Activités du processus unifié	iii
A.3	Le processus 2TUP	iii
A.3.1	La branche gauche (fonctionnelle)	iv
A.3.2	La branche droite (architecture technique)	iv
A.3.3	La branche du milieu	iv

LISTE DES FIGURES

1.1	<i>Organigramme du CHU de Béjaia [12].</i>	4
1.2	<i>Diagramme de contexte du système.</i>	9
1.3	<i>Diagramme de cas d'utilisation générale.</i>	14
1.4	<i>Diagramme de cas d'utilisation associé à l'administrateur.</i>	14
1.5	<i>Diagramme de cas d'utilisation associé au docteur.</i>	15
1.6	<i>Diagramme de cas d'utilisation associé à l'ambulancier.</i>	15
1.7	<i>Diagramme de cas d'utilisation associé au patient.</i>	16
2.1	<i>Diagramme de séquence du cas d'utilisation " Authentification ".</i>	19
2.2	<i>Diagramme de séquence du cas d'utilisation " Faire des prélèvements ".</i>	21
2.3	<i>Diagramme de séquence du cas d'utilisation " Gestion des prescriptions ".</i>	22
2.4	<i>Diagramme de séquence du cas d'utilisation " Ajouter une prescription ".</i>	23
2.5	<i>Diagramme d'activité " Gestion des prescriptions ".</i>	24
2.6	<i>Diagramme d'activité " Faire des prélèvements ".</i>	25
2.7	<i>Diagramme de classes.</i>	26
3.1	<i>Page d'authentification.</i>	33
3.2	<i>Page de réinitialisation de mot de passe.</i>	34
3.3	<i>Interface " Administrateur ".</i>	34
3.4	<i>Formulaire d'ajout d'un médecin.</i>	35
3.5	<i>Fenêtre de "confirmation de la suppression".</i>	35
3.6	<i>Formulaire de modification d'un médecin.</i>	36
3.7	<i>Interface " Médecin ".</i>	36
3.8	<i>Interface " Médecin ".</i>	37
3.9	<i>Page " Gestion des rendez-vous ".</i>	38
3.10	<i>Page " Gestion des prescriptions ".</i>	39
3.11	<i>Page " Gestion des patients ".</i>	40
3.12	<i>Page " Gestion des messages ".</i>	40
3.13	<i>Interface " Patient ".</i>	41
3.14	<i>Page " Prélèvement ".</i>	41
3.15	<i>Page " rendez-vous ".</i>	42
3.16	<i>Page " Prescription ".</i>	43
3.17	<i>Page " transport ".</i>	43
3.18	<i>Interface " Ambulancier ".</i>	44

LISTE DES TABLEAUX

1.1	Les trois hôpitaux de Béjaia.	4
1.2	Les différents acteurs du système.	7
1.3	Liste des messages échangés.	8
1.4	Les différents cas d'utilisation du système à réaliser.	10
1.5	Description textuelle du cas d'utilisation " Authentification ".	11
1.6	<i>Description textuelle du cas d'utilisation " Faire des prélèvements "</i>	12
1.7	<i>Description textuelle du cas d'utilisation " Gestion des prescriptions "</i>	13

Introduction générale

L'informatique fait aujourd'hui partie intégrante de la majorité des entreprises, à un tel point que le métier d'informaticien s'est diversifié en une multitude de fonctions spécialisées, touchant au réseau, au développement informatique, à la sécurité des systèmes informatiques ou encore à l'infographie ou l'ergonomie.

À certains métiers correspondent des études particulières, tandis que la carence en informaticien rend la porte d'entrée de certaines fonctions techniques ouverte à tous. La bonne connaissance du cœur de métier de chaque fonction peut permettre d'en connaître les spécificités, les limites ainsi que les compétences indispensables.

Parmi ces métiers, on trouve la conception et la réalisation qui consistent à analyser un besoin, à concevoir des solutions, les modéliser informatiquement et les implémenter, c'est-à-dire les transcrire dans un langage informatique. Il s'agit de métiers alliant une capacité d'analyse à un esprit de synthèse et mettant en œuvre technique et créativité.

Notre travail consiste à concevoir et réaliser un système de santé public moderne afin d'améliorer le suivi des patients et favoriser l'assistance à distance en termes de délais et de qualité de service, et ainsi faciliter le quotidien des patients (les diabétiques, les hypertendus et les personnes âgées).

Pour la réalisation de cette tâche, notre choix s'est porté sur le Processus Unifié. En effet, le processus unifié est une solution de développement logiciel adaptée à tout type de projet. Ses traits distinctifs tiennent en trois notions : piloté par les cas d'utilisation, centré sur l'architecture, itératif et incrémental.

Le langage de modélisation qu'on a utilisé est UML (*Unified Modeling Language*), qui est une partie intégrante de la démarche UP. Ses diagrammes sont largement utilisés dans chaque étape et phases de ce processus de développement.

Pour l'implémentation, le choix du langage de programmation a été dicté par le type de l'application qui devrait être réalisée. Ainsi, le choix s'est porté sur le langage de programmation PHP. La base de données est implémentée avec MySQL qui est largement compatible avec PHP.

Ayant présenté les outils et la méthode adoptée, nous allons maintenant exposer le plan du mémoire qui se subdivise en trois principaux chapitres.

Dans le premier chapitre intitulé « **Étude préliminaire et analyse des besoins** » nous commençons à présenter l'organisme d'accueil, approfondir la compréhension du contexte du système par un processus continu de collecte d'informations. ensuite déterminer les inconvénients majeurs, énumérer des suggestions informatiques qui peuvent remédier aux difficultés rencontrées, et enfin nous recenserons les acteurs qui interagissent avec le système à développer et les cas d'utilisation relatifs à chaque acteur.

Au niveau du deuxième chapitre intitulé « **L'analyse et conception** », nous réaliserons les diagrammes de séquence associés à chaque cas d'utilisation et nous présenterons le diagramme de classes associé à notre système et le modèle relationnel obtenu par l'application des règles de passage.

La « **Réalisation** » fera l'objet du troisième chapitre dans lequel nous définirons les outils de développement qui nous ont servi pour le développement de système, et enfin nous illustrerons également, quelques interfaces de l'application développé.

Enfin, nous conclurons ce travail en résumant les connaissances acquises le long de ce projet.

1

Étude préliminaire et analyse des besoins

1.1 Introduction

Dans ce chapitre, nous présentons l'organisme d'accueil « Centre Hospitalo-Universitaire de Béjaia ». Ensuite, nous effectuerons une étude préliminaire qui nous aidera à déterminer nos objectifs et les besoins de notre application. En enfin, nous traitons le rôle que tient UML pour compléter la capture des besoins fonctionnels et techniques ébauchés durant l'étude préliminaire, qui va nous permettre d'effectuer une étude du contexte fonctionnel et non fonctionnel du système à réaliser.

1.2 Étude préliminaire

1.2.1 Présentation de l'organisme d'accueil

Le centre hospitalo-universitaire (CHU) est un établissement public de santé à caractère administratif, doté de la personnalité morale et de l'autonomie financière. Il est créé par décret exécutif n° 07-14 du 19 mai 2007. La tutelle pédagogique est assurée par le ministre chargé de l'enseignement supérieur. Le CHU est chargé, en relation avec l'établissement d'enseignement et/ou de formation supérieure en sciences médicales concernées, des missions : de diagnostic, d'exploration, de soins, de prévention, de formation, d'études et de recherche [12].

2.2.1.1 Organisation et structure du CHU de Béjaia

Le CHU de Béjaia est composé de plusieurs services médicaux spécialisés répartis dans les trois hôpitaux principaux de la ville comme suit [12] :

Dénomination	Siège	Consistances Physique
CHU Béjaia	Hôpital Khelil Amrane	<ul style="list-style-type: none"> - Hôpital Khelil Amrane - Hôpital Frantz Fanon - Hôpital Tergua Ouzemmour

TABLE 1.1 – Les trois hôpitaux de Béjaia.

2.2.1.2 Organigramme du CHU de Béjaia

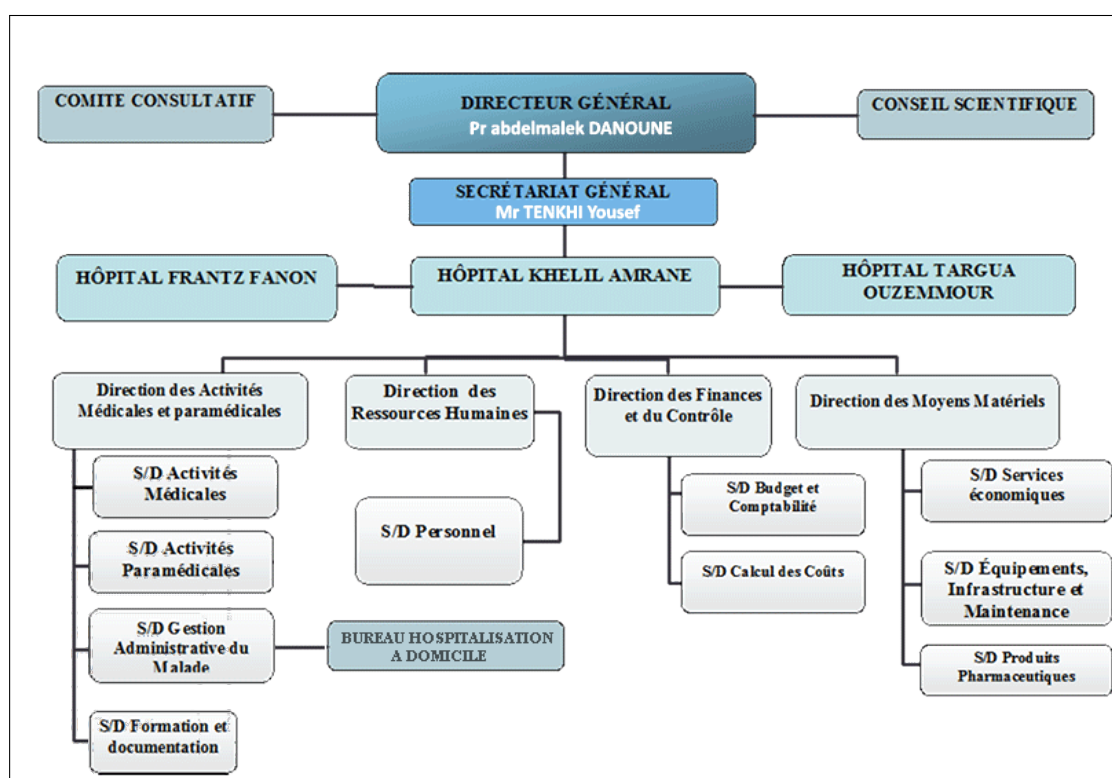


FIGURE 1.1 – Organigramme du CHU de Béjaia [12].

1.2.2 Présentation et mission du bureau d'hospitalisation à domicile

Le dispositif d'hospitalisation à domicile lancé par le secteur de la santé publique apporte une solution adaptée aux malades souffrant d'un handicap et aux personnes âgées ayant des difficultés à se déplacer vers les structures de soins.

L'hospitalisation à domicile est donc une structure de soins alternative à l'hospitalisation. Elle permet d'assurer au domicile du patient des soins médicaux et paramédicaux importants, pour une période limitée mais renouvelable en fonction de l'évolution de son état de santé.

Elle a pour finalité d'éviter ou de raccourcir une hospitalisation en établissement.

1.2.2.1 Personnel du bureau d'hospitalisation à domicile

Le bureau d'hospitalisation à domicile contient un nombre limité de personnel médical et administratif, L'équipe est constitué de :

- Des docteurs ;
- Des infirmières ;
- Une secrétaire.

1.2.2.2 Situation informatique du bureau

Le bureau d'hospitalisation à domicile du CHU de Béjaia dispose de cinq ordinateurs "DELL" possèdent les caractéristiques et les périphériques suivants :

- Lecteur CD DVD ;
- Quatre imprimantes ;
- la mémoire vive est deux GO ;
- Windows XP comme système d'exploitation ;
- Processeur intel pentium quatre ;
- Windows Office 2007 pour le traitement du texte.

1.2.3 Problématiques

Durant notre stage au sein du centre hospitalo-universitaire de Béjaia, nous avons soulevé les limites suivantes :

- ✓ **Absence d'un moyen de recherche rapide** : La moindre tentative d'évaluation, ou de recherche à partir des dossiers, demande des heures de recherche manuelle fastidieuse, ce qui engendre une perte de temps en plus risqué que les fiches peuvent se mélanger et surtout leurs contenus ;
- ✓ **Perte de documents** : puisqu'un dossier médical englobe un ensemble de documents tels que, fiche médicale, ordonnance et les feuilles qui contiennent les dates des RDV. Il est possible qu'un document qui appartient à un dossier soit rangé par erreur dans un autre dossier lors de l'organisation et le stockage dans les boîtes d'archives ;
- ✓ **Le manque de confidentialité de l'information** : nous avons remarqué que la secrétaire peut accéder aux informations confidentielles du patient, or le respect du secret médical impose que seul le médecin puisse consulter ce dossier ;
- ✓ Encombrement de non-clarté de la fiche médicale qui contient plusieurs informations à cause de sa petite taille, chose qui peut générer l'ajout certaines informations inutiles ou la suppression parfois de certaines informations utiles ;
- ✓ Les suivis sont souvent mal structurés, difficiles à lire et parfois introuvables.

1.2.4 Les objectifs de l'application

Après avoir critiqué et détecté les anomalies, nous proposons une solution qui consiste à concevoir et à développer une application qui palliera au limites énumérées précédemment et visera à atteindre les objectifs suivants :

- ✓ Implanter une base de données complète pour la gestion des dossiers médicaux du patient, assurer une meilleure communication et cohérence de l'information ;
- ✓ Garantir un haut niveau de sécurité notamment concernant la confidentialité des dossiers médicaux des patients ;
- ✓ Optimiser le temps d'accès aux différentes données et éviter les tâches pénibles et ennuyeuses ;
- ✓ Gérer les droits d'accès afin de permettre un accès sélectif à l'information médicale ;
- ✓ Donner beaucoup d'importance à l'interface homme-machine et l'adapter au maximum à l'utilisateur de l'application.

1.3 L'analyse du personnel

Il s'agit de commencer à déterminer les besoins fonctionnels en considérant le système comme une boîte noire. On procède en trois étapes :

- Identifications des acteurs ;
- Identifications des messages échangés ;
- Développement du diagramme de contexte UML.

1.3.1 Identifications des acteurs

Un acteur représente une abstraction d'un rôle joué par une entité externe (utilisateur humain, dispositif matériel ou autre système) qui interagissent directement avec le système étudié [2].

Dans le cadre de notre étude, nous avons distingué quatre acteurs qui interagissent directement avec le système :

Acteur
Patient
Ambulancier
Docteur
Administrateur

TABLE 1.2 – Les différents acteurs du système.

- **La description des rôles de chaque acteur**

- **Patient** : est un utilisateur à qui une inscription aura été faite au préalable par un médecin dans le cas où le patient en question a besoin d'un suivi médical, si un autre médecin veut suivre l'état du patient qui possède déjà un compte, il n'aura qu'à l'ajouter dans la liste de ses suivis, ils pourront ainsi s'échanger des messages destinés à réduire le nombre de visites. Le patient peut aussi réserver un véhicule de transport médical en cas de besoin.

- **Ambulancier** : est un utilisateur qui prend en charge le service de transport des personnes à mobilité réduite afin de faciliter les déplacements des patients.

- **Docteur** : est un utilisateur qui a le droit d'accéder à tout moment à tous les dossiers des patients déjà ajoutés par lui-même dans le but de suivre leur état à distance

et ceux par envois des messages entre les deux acteurs (docteur et patient), il peut aussi planifier un rendez-vous pour son patient.

- **Administrateur** : est un utilisateur qui prend en charge la gestion et l'administration du site, il est en mesure d'apporter des modifications nécessaires à l'application, pour gérer le système, l'administrateur doit suivre et maintenir plusieurs parties :

- ✓ La partie gestion des comptes utilisateurs ;
- ✓ La gestion des unités de service que CHU de Béjaia dispose ;
- ✓ Afficher les statistiques de l'application.

1.3.2 Identifications des messages échangés

Les messages permettent [3] :

- Spécification d'une communication entre objets ;
- Transport de l'information ;
- Intention de déclencher une activité chez le récepteur.

Le tableau suivant résume les messages échangés entre les acteurs et le système :

Acteur	Messages	Services offerts
Patient	Demande d'envoi de message à un médecin.	Sauvegarder le message.
	Demande de consultation d'un message du médecin.	Afficher le message.
	Demande de réserver un véhicule pour une évacuation.	Réservation du véhicule.
Ambulancier	Demande de consultation de la liste des demandes de transport des patients.	Afficher la liste des demandes adéquate.
Docteur	Demande d'envoi de message à un patient.	Sauvegarder le message.
	Demande de consultation des messages.	Chercher le message.
	Demande de gestion des patients.	Fournir la liste des patients.
Administrateur	Demande de gestion des utilisateurs.	Fournir la liste des utilisateurs.
	Demande de gestion des unités de service.	Fournir la liste des unités de service.

TABLE 1.3 – Liste des messages échangés.

1.3.3 Diagramme de contexte

Le diagramme de contexte d'un système est l'ensemble des objets échangés entre un système organisationnel et des acteurs de l'environnement externe du système à l'étude [2].

Ce type de diagramme nous permet de représenter et de visualiser les composants de l'environnement de notre système à cet effet les différents acteurs interagissant avec le système.

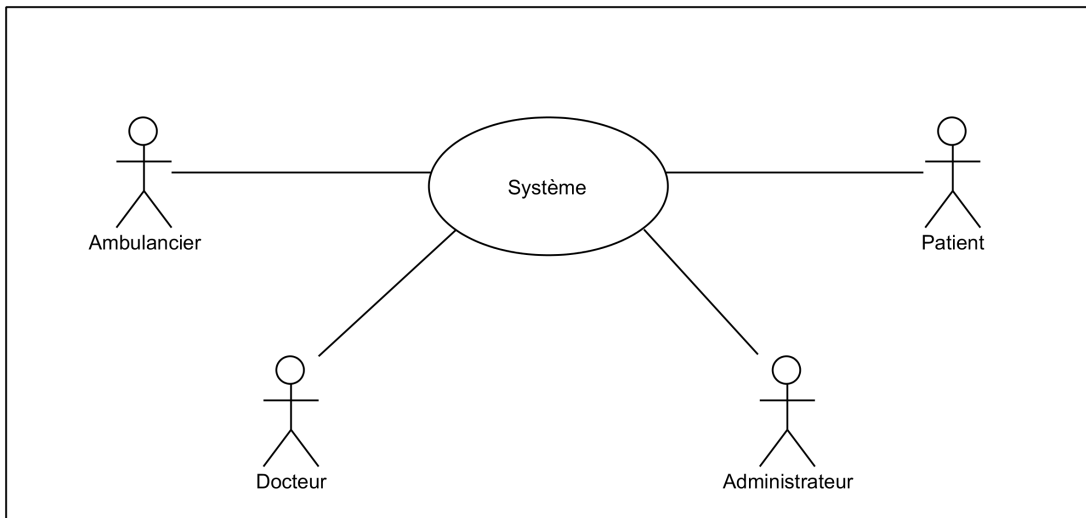


FIGURE 1.2 – *Diagramme de contexte du système.*

1.4 Capture des besoins

1.4.1 Les besoins fonctionnels

Les besoins fonctionnels représentent les actions que le système doit exécuter. Notre application doit couvrir principalement les besoins fonctionnels suivants :

- La gestion des utilisateurs ;
- La gestion des messages ;
- La gestion de service du transport des malades ;
- Impression de la prescription du docteur ;
- La gestion des rendez-vous de consultation ;
- Calcul des statistiques.

1.4.1.1 Identifications des cas d'utilisation

a) Définition d'un cas d'utilisation

Un cas d'utilisation est une narration qui décrit un scénario appliqué à une utilisation particulière, dans lequel les acteurs fournissent des entrées pour lesquelles le système produit une sortie observable. Le cas d'utilisation doit juste exprimer ce que doit faire un acteur, il décrit le comportement du système vu de l'extérieur [3].

b) La liste des cas d'utilisation

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

N°	Cas d'utilisation	Acteur
1	Authentification	Administrateur, docteur, patient, ambulancier.
2	Gestion des utilisateurs	Ajout
		Modification
		Suppression
3	Gestion des unités de service	Ajout
		Modification
		Suppression
4	Gestion des patients	Ajout
		Modification
		Suppression
5	Gestion des RDV	Ajout
		Modification
		Suppression
		Traitement d'une demande de RDV
6	Gestion des prescriptions	Ajout
		Modification
		Suppression
7	Gestion des messages	Ajout
		Suppression
8	Demander un RDV	Patient.
9	Saisir des prélèvement	
10	Consulter les prescriptions	
11	Demande de transport	
12	Traitement d'une demande de transport	Valider
		Refuser

TABLE 1.4 – Les différents cas d'utilisation du système à réaliser.

2.4.1.2 Description textuelle des cas d'utilisation

La fiche de description textuelle d'un cas d'utilisation n'est pas définie par UML. Le formalisme qu'on utilisera est juste une proposition.

a) Description de cas d'utilisation " Authentification "

Les utilisateurs du site à développer possèdent des interfaces différentes. Chaque catégorie d'utilisateurs possède une interface qui lui est associée, à travers laquelle il peut accéder aux informations qui le concernent.

Cas d'utilisation N° 1	Authentification
Description	Vérifier les possibilités d'accès au système.
Acteurs	Administrateur, docteur, ambulancier, patient.
Pré-conditions	Site accessible.
Scénario nominal	[Début] 1. Accès au site ; 2. Saisie du nom d'utilisateur et du mot de passe ; 3. Le système vérifie les données saisies (A1) ; 4. Le système affiche l'interface correspondante. [Fin]
Scénario alternatif	A1 (alternative) est une étiquette qui indique que le Le système affiche échec d'accès à l'interface.

TABLE 1.5 – Description textuelle du cas d'utilisation " Authentification ".

b) Description de cas d'utilisation " Faire des prélèvements "

Le cas d'utilisation " Faire des prélèvements " est caractérisé par les trois scénarios suivants :

- Prise de poids ;
- Prise de glycémie ;
- Prise de tension.

Cas d'utilisation N° 2	Faire des prélèvements
Description	Ce cas d'utilisation permet au patient de faire trois types de prélèvements (prise de poids, prise de glycémie, prise de tension) selon la demande de son docteur.

Acteurs	Patient.
Pré-conditions	Patient doit être authentifié.
Scénario nominal	<p>[Début]</p> <ol style="list-style-type: none"> 1. Accès au site ; 2. Authentification ; 3. Le système affiche l'interface correspondante ; 4. Le patient demande la page prélèvement ; 5. Le système fournit la page correspondante ; 6. Trois alternatives se présente : prise de poids si le patient doit introduire son poids, prise de tension si le patient doit introduire sa tension et prise de glycémie si le patient doit introduire son taux de glycémie ; 7. Le système affiche un message de confirmation (A1). <p>[Fin]</p>
Scénario alternatif	A1 : Si le champ du prélèvement reste vide ou bien les données saisies sont incompatibles avec le type de champs, alors le système affiche un message d'erreur.

TABLE 1.6 – Description textuelle du cas d'utilisation " Faire des prélèvements "

c) Description de cas d'utilisation " Gestion des prescriptions "

Ce cas d'utilisation est caractérisé par les trois scénarios suivants :

- Ajout d'une prescription ;
- Modification d'une prescription ;
- Suppression d'une prescription.

Cas d'utilisation N° 3	Gestion des prescriptions
Description	Ce cas d'utilisation permet au docteur d'ajouter, modifier ou supprimer une prescription ainsi que d'ajouter un rapport à son patient en cas de besoin.
Acteurs	Docteur.
Pré-conditions	<ul style="list-style-type: none"> - Docteur doit être authentifié ; - Le patient doit faire partie de la liste des patients du docteur.

Scénario nominal	<p>[Début]</p> <ol style="list-style-type: none"> 1. Accès au site ; 2. Authentification ; 3. Le système affiche l'interface correspondante ; 4. Le docteur demande la page de prescription ; 5. Le système fournit la page correspondante ; 6. Trois alternatives se présente : dans le cas où le docteur voudrait ajouter une prescription à son patient, le système affiche le formulaire d'ajout. Si le docteur choisit de modifier (ou de supprimer) une prescription, il doit alors sélectionner la prescription en question, puis effectue l'opération ; 7. Le système affiche un message de confirmation. <p>[Fin]</p>
------------------	---

TABLE 1.7 – Description textuelle du cas d'utilisation " Gestion des prescriptions "

2.4.1.3 Organisation des cas d'utilisation

Afin d'optimiser la formalisation des besoins, Trois relations peuvent être décrites entre cas d'utilisation : la relation d'inclusion (notée « include »), la relation d'extension (notée « extend ») et la relation de généralisation [2].

2.4.1.4 Réalisation du diagramme de cas d'utilisation

Le diagramme de cas d'utilisation est un formalisme permettant de modéliser le fonctionnement d'un système par un découpage en fonctionnalités [2].

Le diagramme ci-après représente le diagramme général de cas d'utilisation de notre application.

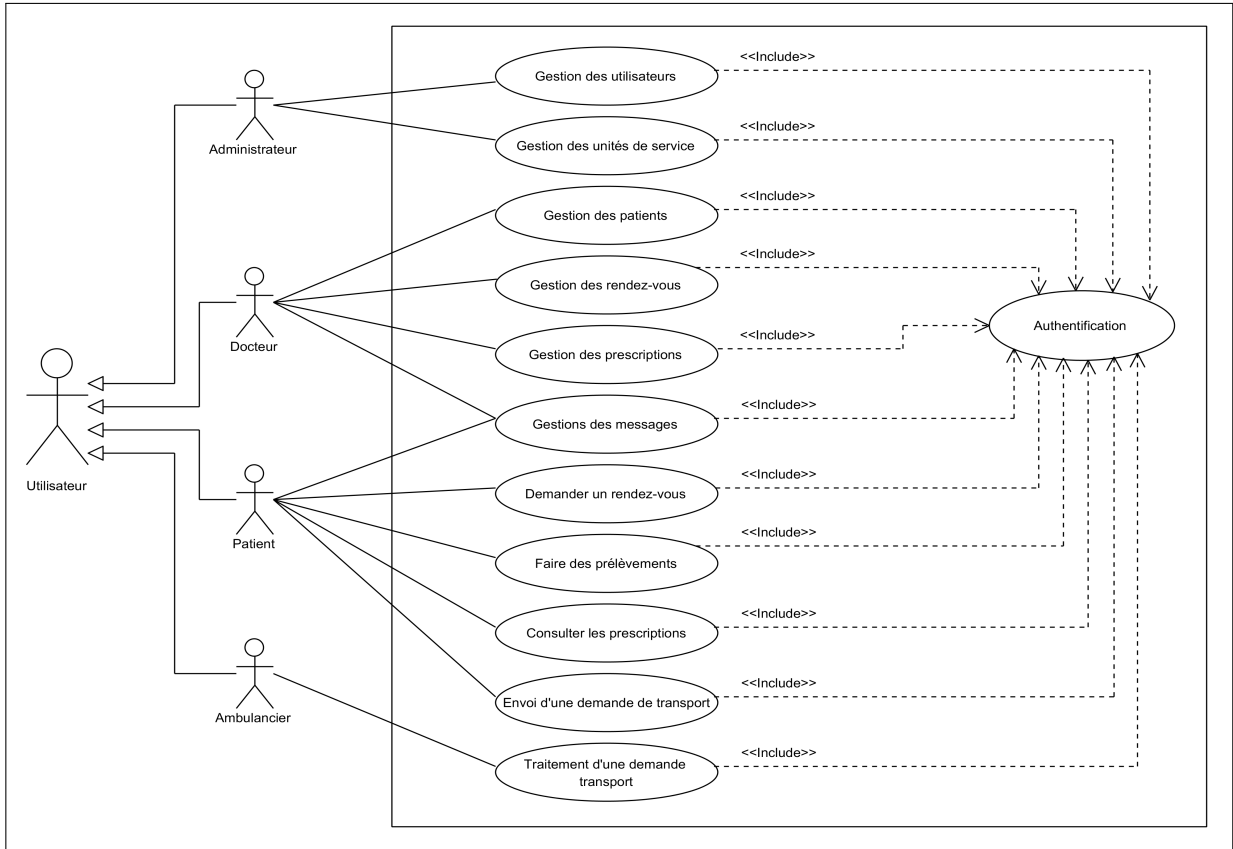


FIGURE 1.3 – Diagramme de cas d'utilisation générale.

Dans le diagramme ci-dessous, nous illustrons les différents cas d'utilisations assignés à l'administrateur.

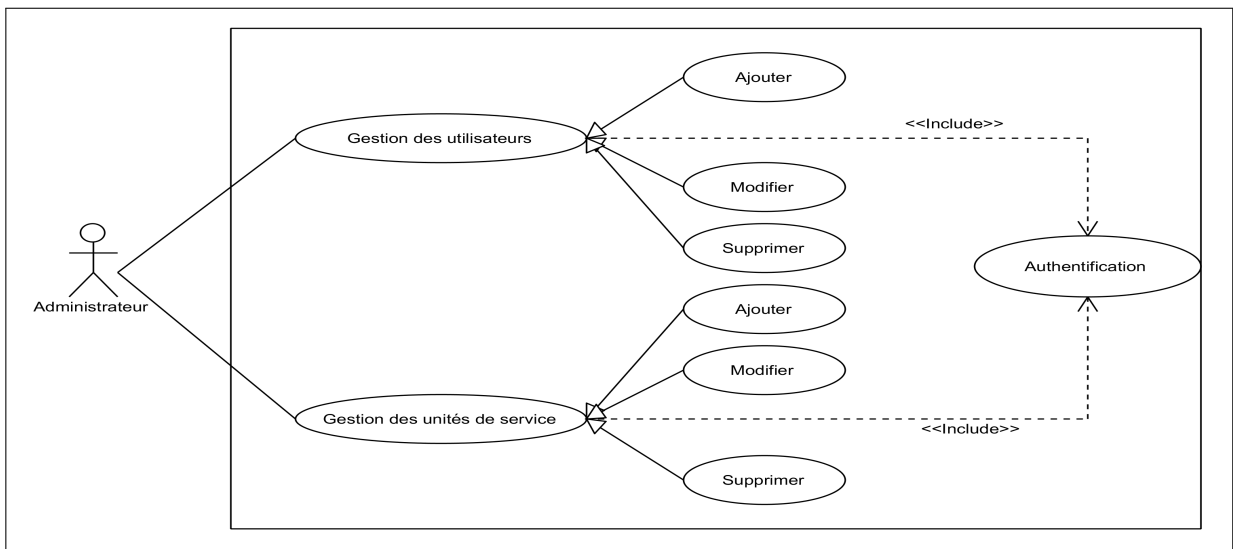


FIGURE 1.4 – Diagramme de cas d'utilisation associé à l'administrateur.

Dans le diagramme ci-après, nous illustrons les différents cas d'utilisations assignés au docteur.

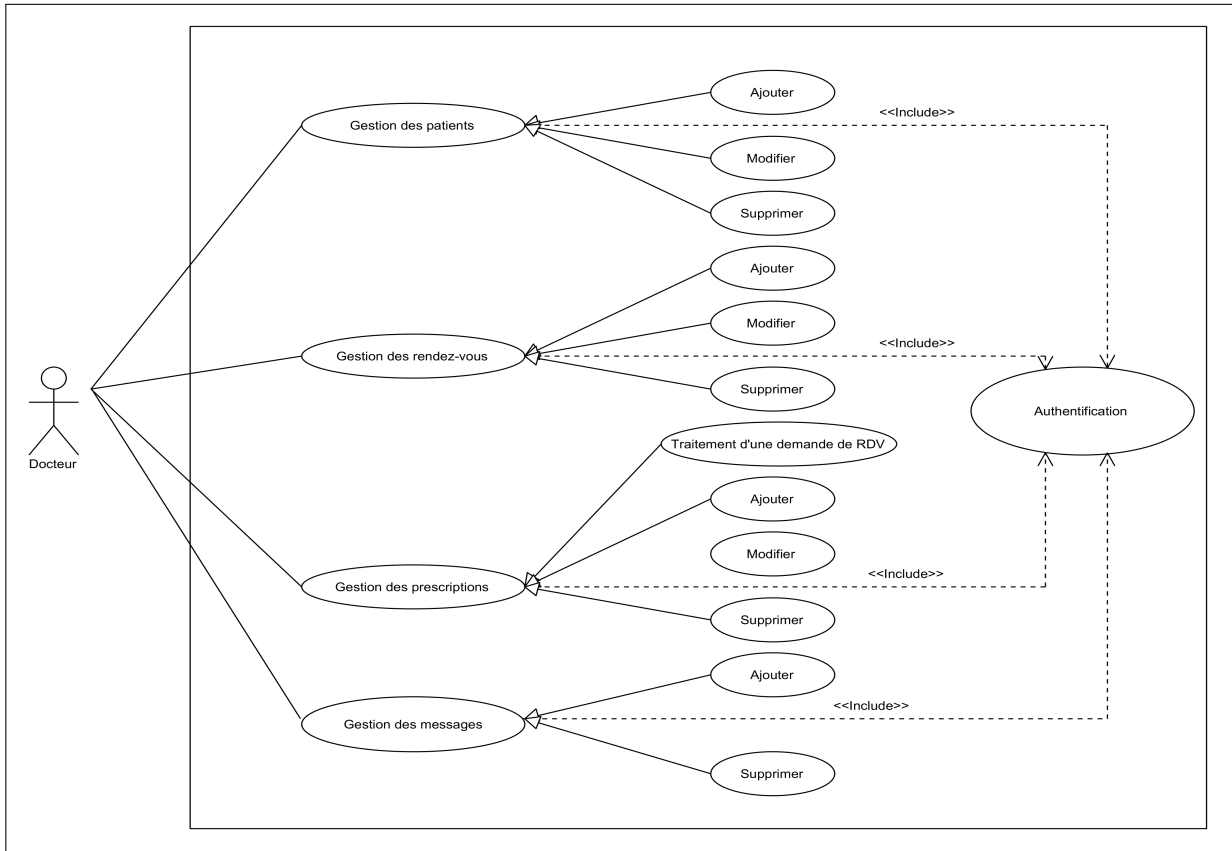


FIGURE 1.5 – Diagramme de cas d'utilisation associé au docteur.

Dans le diagramme suivant, nous illustrons les différents cas d'utilisations assignés à l'ambulancier.

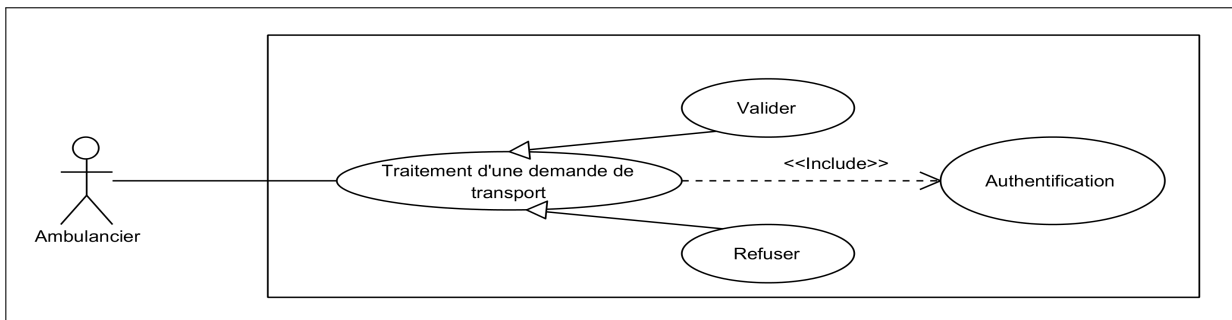


FIGURE 1.6 – Diagramme de cas d'utilisation associé à l'ambulancier.

Dans le diagramme ci-dessous, nous représentons les différents cas d'utilisation relatifs au patient.

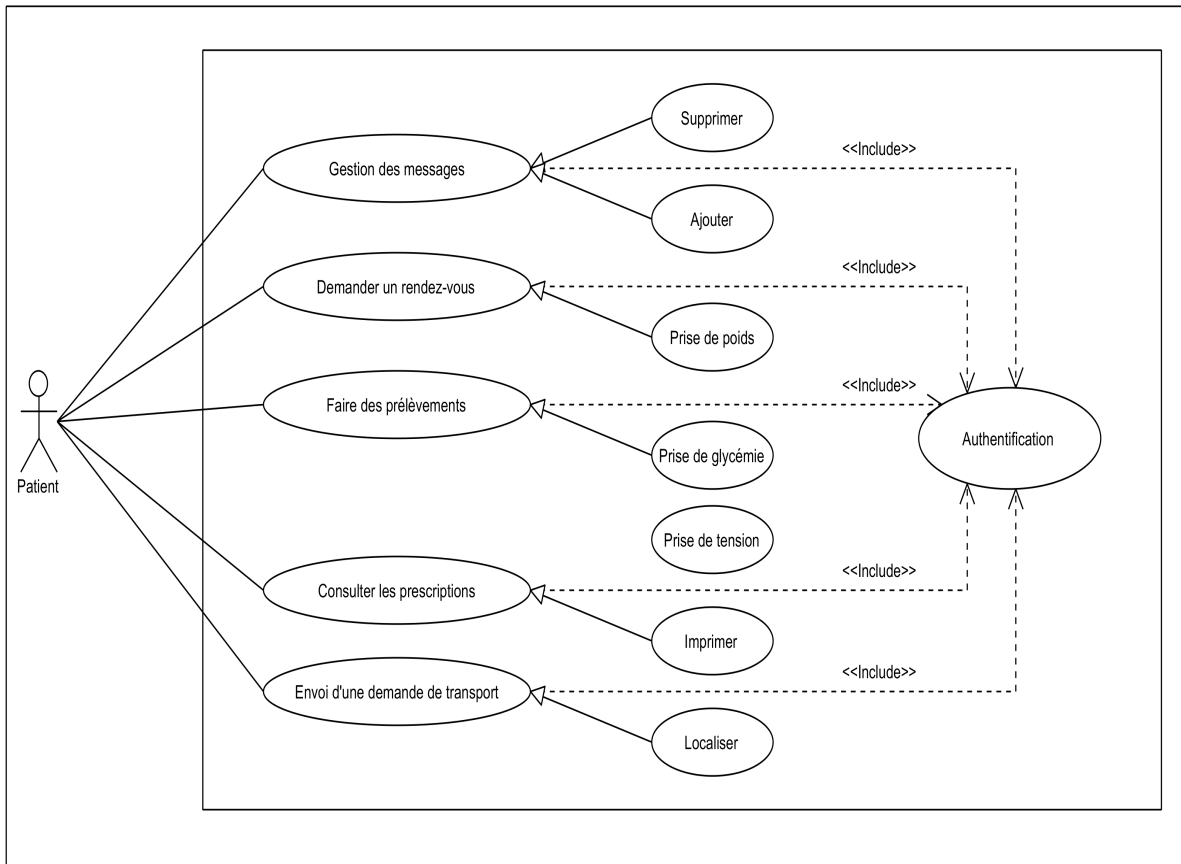


FIGURE 1.7 – Diagramme de cas d'utilisation associé au patient.

1.4.2 Les besoins techniques

La capture des besoins techniques recense toutes les contraintes et les choix dimensionnant la conception du système [2]. Les outils et les matériels sélectionnés ainsi que la prise en des contraintes d'intégration conditionnant généralement des prés requis d'architecture générale.

2.4.2.1 L'architecture d'application

Pour notre application, nous avons choisi de mettre en place le style d'architecture de 3-tiers qui définit le moyen intermédiaire entre les applications client et le serveur de la base de données.

2.4.2.2 Les exigences de qualité

- Le choix de couleur doit avoir une signification par rapport au domaine ;
- les couleurs doivent être harmonieux et liées ;
- La police et la taille des textes doivent être facilement lisibles.

2.4.2.3 La sécurité

Les répertoires privés (les comptes) doivent être protégés par une authentification et une déconnexion après l'utilisation.

2.4.2.4 La simplicité

Le système de navigation doit indiquer à l'utilisateur sa position dans le site, C'est-à-dire dans quelle page on est et vers quelle page on peut aller.

1.5 conclusion

Dans ce chapitre, nous avons fait une analyse de l'existant et une spécification des différents besoins fonctionnels et techniques, ses besoins regroupant l'ensemble des services que doit le système offrir.

Dans le prochain chapitre, nous allons entamer une étape très importante du processus 2TUP, qui est l'analyse et conception.

2

L'analyse et conception

2.1 Introduction

On se référant sur la démarche 2TUP, nous passons à la phase d'analyse ainsi que la phase de conception qui représente deux étapes essentielles de la démarche, ces étapes nous permettent de passer d'une structure fonctionnelle via les cas d'utilisations à une structure objet sous forme de modèle de classe.

Au cours de ce qui suit, nous allons présenter une vue dynamique du système par la modélisation des diagrammes de séquences, d'activités et une vue statique par la modélisation du diagramme de classes.

2.2 Diagramme de séquence

L'objectif du diagramme de séquence est de représenter les interactions entre objets en indiquant la chronologie des échanges. Cette représentation peut se réaliser par cas d'utilisation en considérant les différents scénarios associés [2].

2.2.1 Diagramme de séquence du cas d'utilisation « Authentification »

L'authentification consiste à assurer la confidentialité des données, elle se base sur la vérification des informations associées à un utilisateur (généralement un email et un mot de passe). Ces informations sont préétablies dans une base de données. Lors d'une authentification deux cas se présentent : les informations introduites par l'utilisateur sont incomplètes, dans ce cas un message d'erreur s'affiche, ou les informations saisies sont complètes et le système procède à leur vérification. Cela explique l'utilisation de l'opérateur « alt ». Le même opérateur illustre les deux réactions du système, après la vérification des informations saisies par l'utilisateur, soit par l'affichage d'un message d'erreur, ou de l'interface correspondante.

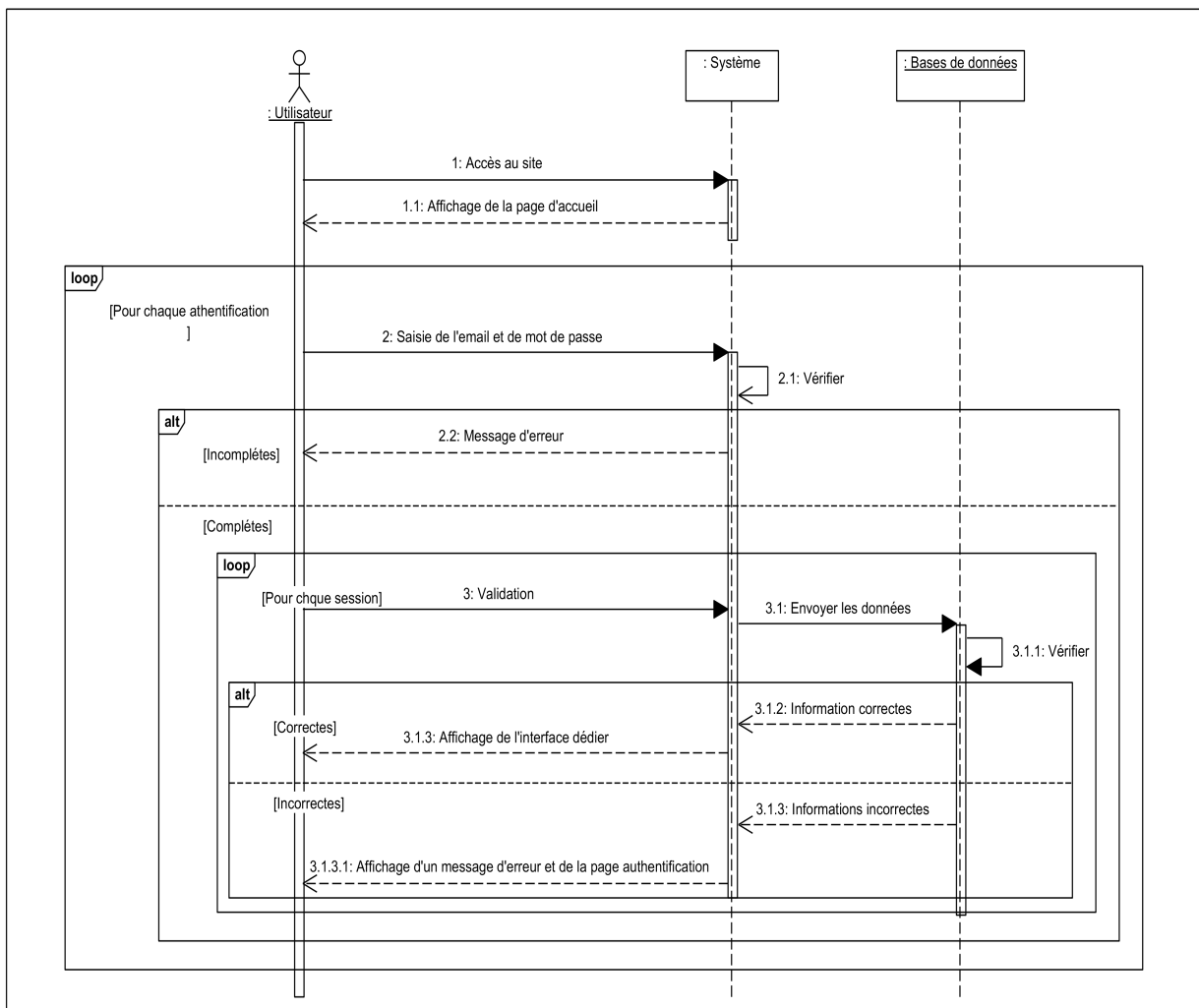


FIGURE 2.1 – Diagramme de séquence du cas d'utilisation " Authentification ".

2.2.2 Diagramme de séquence du cas d'utilisation « Faire des prélèvements »

Ce cas d'utilisation compte trois scénarios d'ajout de prélèvements " Glycémie " , " Tension " ou " Poids/Taille ". **a)** Dans le cas glycémie : Le système répond à la demande de patient concernant l'ajout d'un prélèvement de glycémie par l'affichage d'un formulaire qui sera validé après remplissage.

b) Dans le cas de tension : La réponse du système, pour la requête de patient concernant l'ajout d'un prélèvement de tension est l'affichage d'un formulaire qui sera validé après remplissage.

c) Dans le cas poids/taille : Le système répond à la demande de patient concernant l'ajout d'un prélèvement de poids/taille par l'affichage d'un formulaire qui sera validé après remplissage.

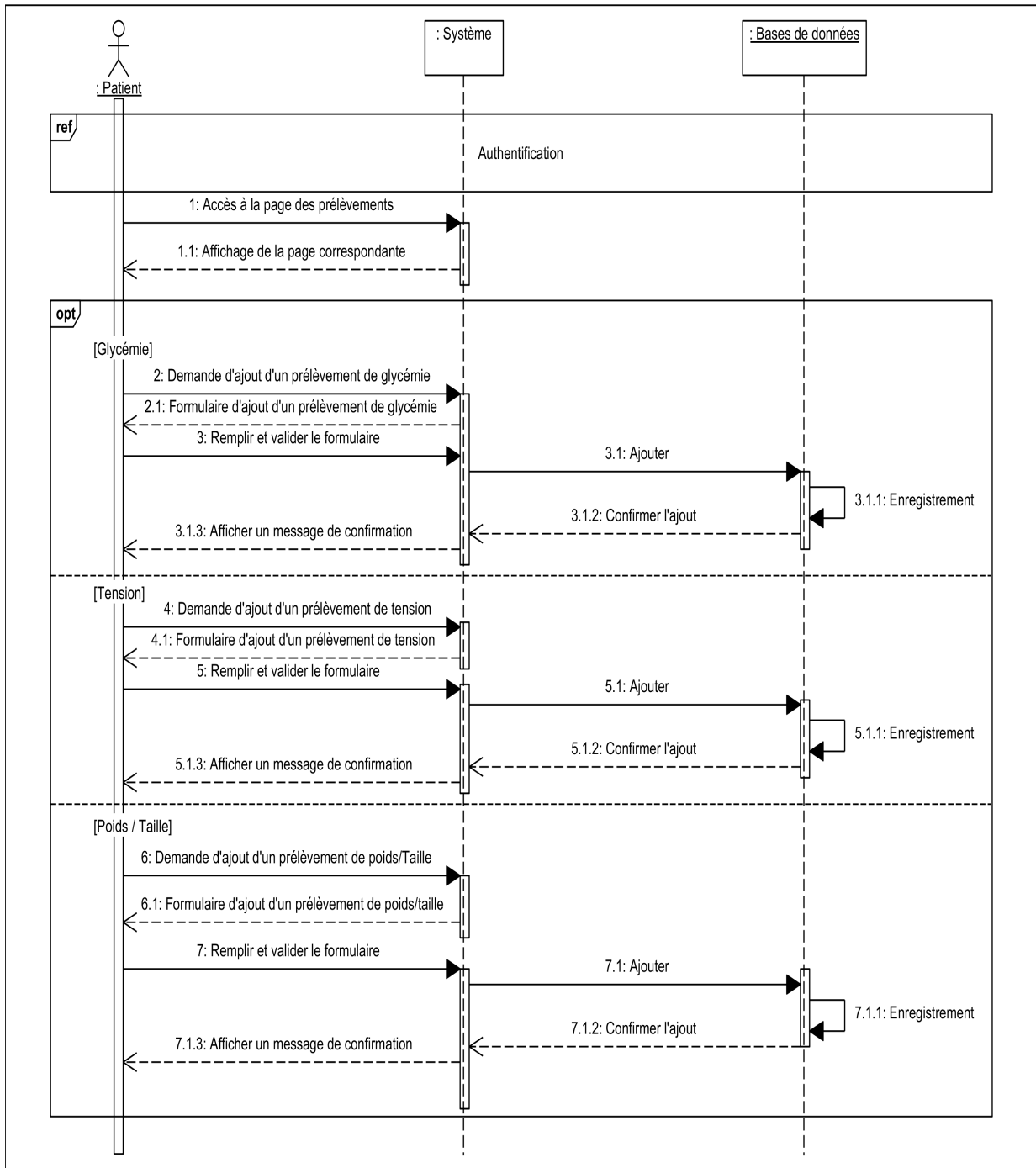


FIGURE 2.2 – Diagramme de séquence du cas d'utilisation " Faire des prélèvements ".

2.2.3 Diagramme de séquence du cas d'utilisation « Gestion des prescriptions »

Le diagramme ci-après représente le diagramme de séquence générale du cas d'utilisation « gestion des prescriptions ».

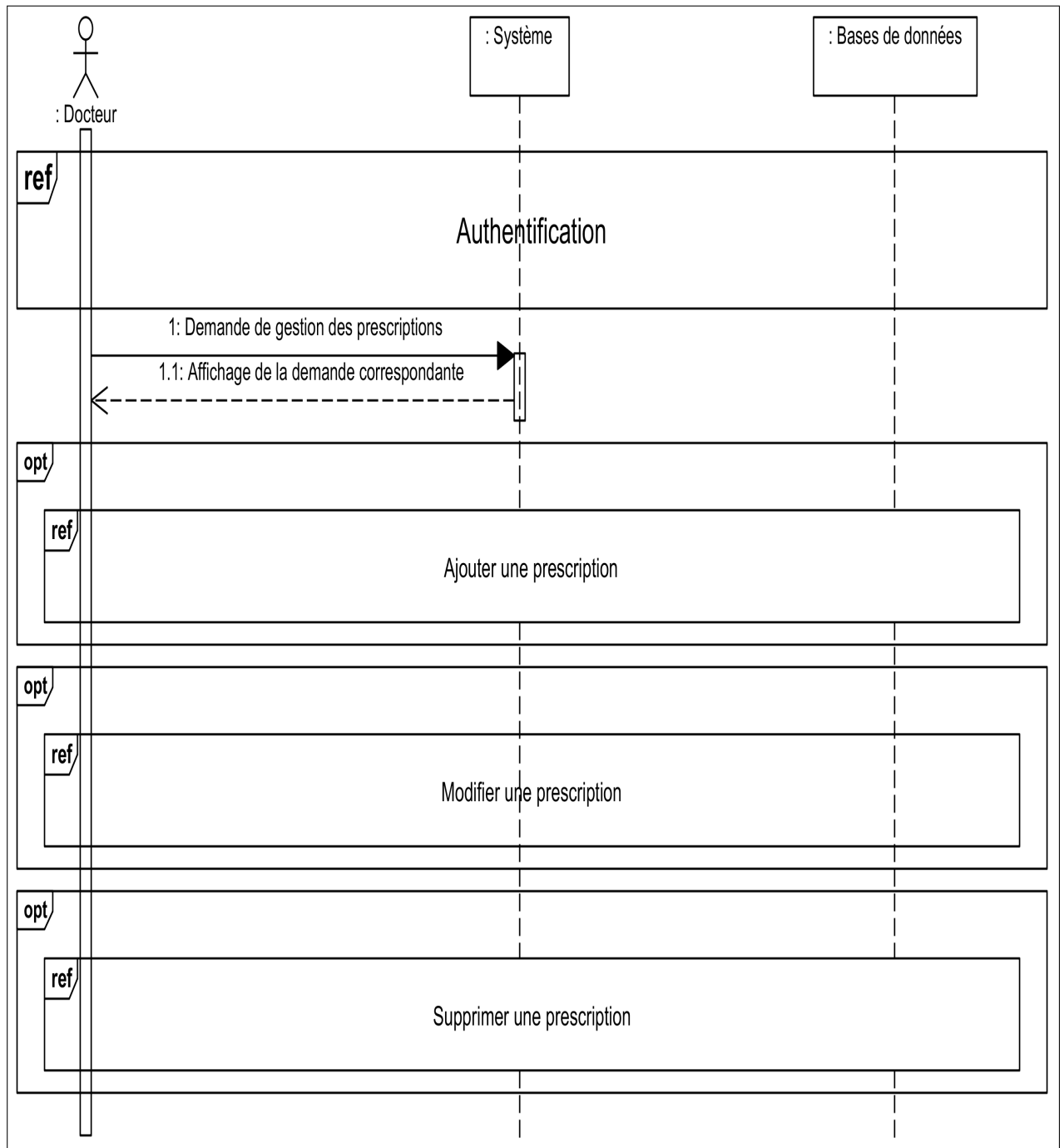


FIGURE 2.3 – Diagramme de séquence du cas d'utilisation " Gestion des prescriptions "

2.2.4 Diagramme de séquence du cas d'utilisation «Ajouter une prescription»

Dans ce diagramme le docteur a la possibilité d'ajouter une prescription à son patient, pour ce faire le docteur doit remplir un formulaire d'ajout avec les informations personnelles du patient puis confirme l'opération.

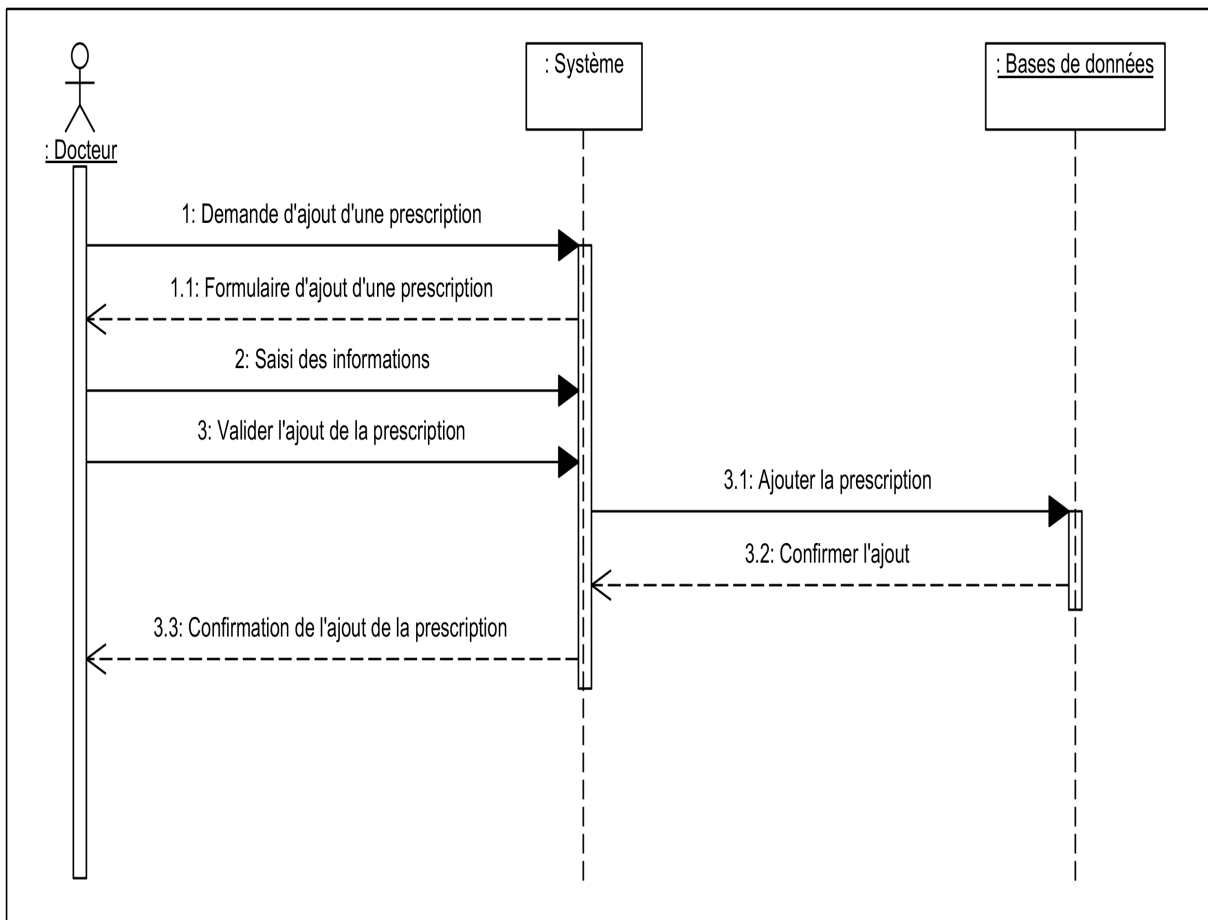


FIGURE 2.4 – Diagramme de séquence du cas d'utilisation " Ajouter une prescription "

2.3 Diagramme d'activités

Un diagramme d'activité permet de modéliser un processus interactif, global ou partiel pour un système donné (logiciel, système d'information). Il est recommandable pour exprimer une dimension temporelle sur une partie du modèle, à partir de diagrammes de classes ou de cas d'utilisation [2].

Les diagrammes d'activités de nos cas d'utilisation sont présentés sur les figures suivantes :

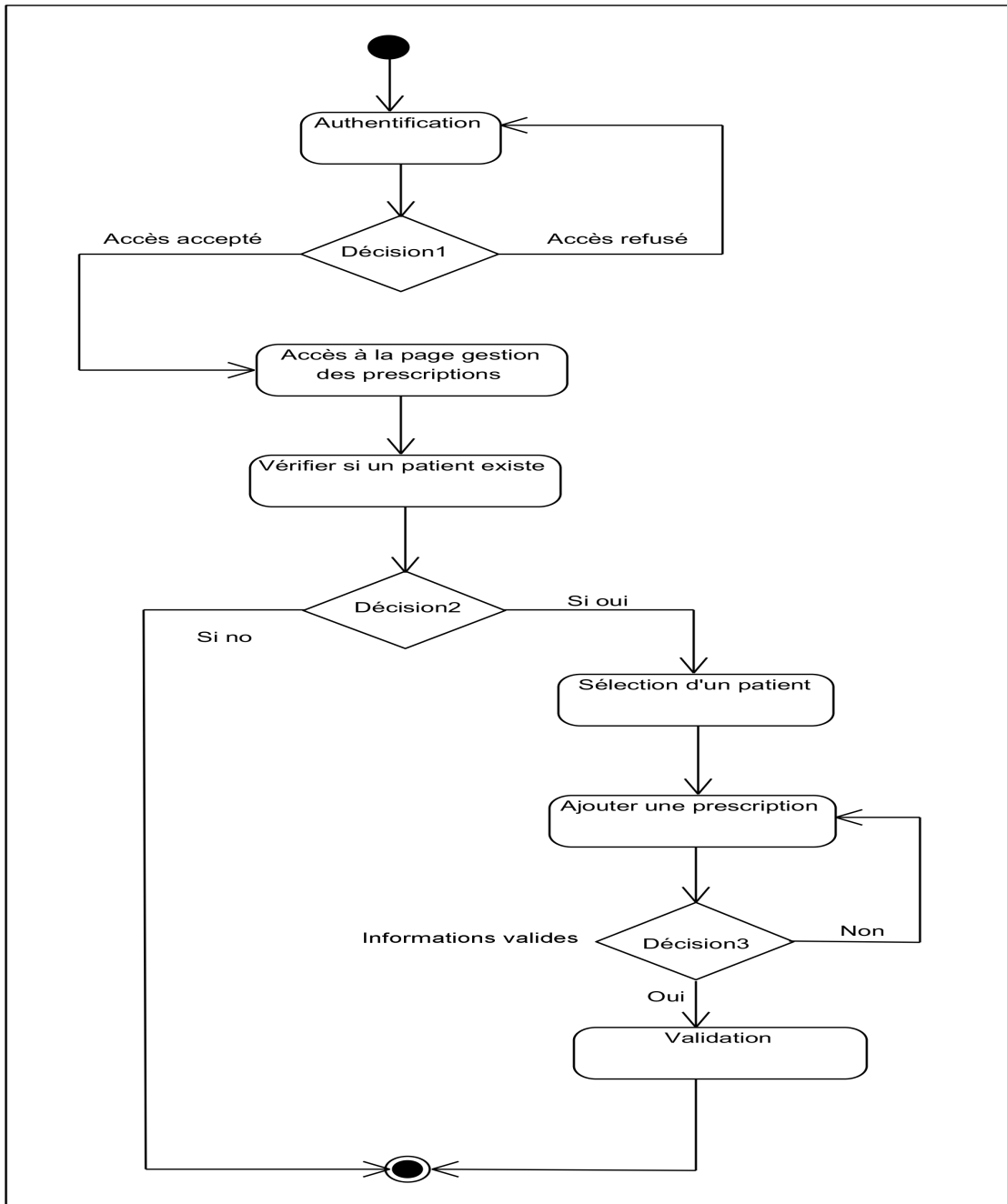


FIGURE 2.5 – Diagramme d'activité " Gestion des prescriptions ".

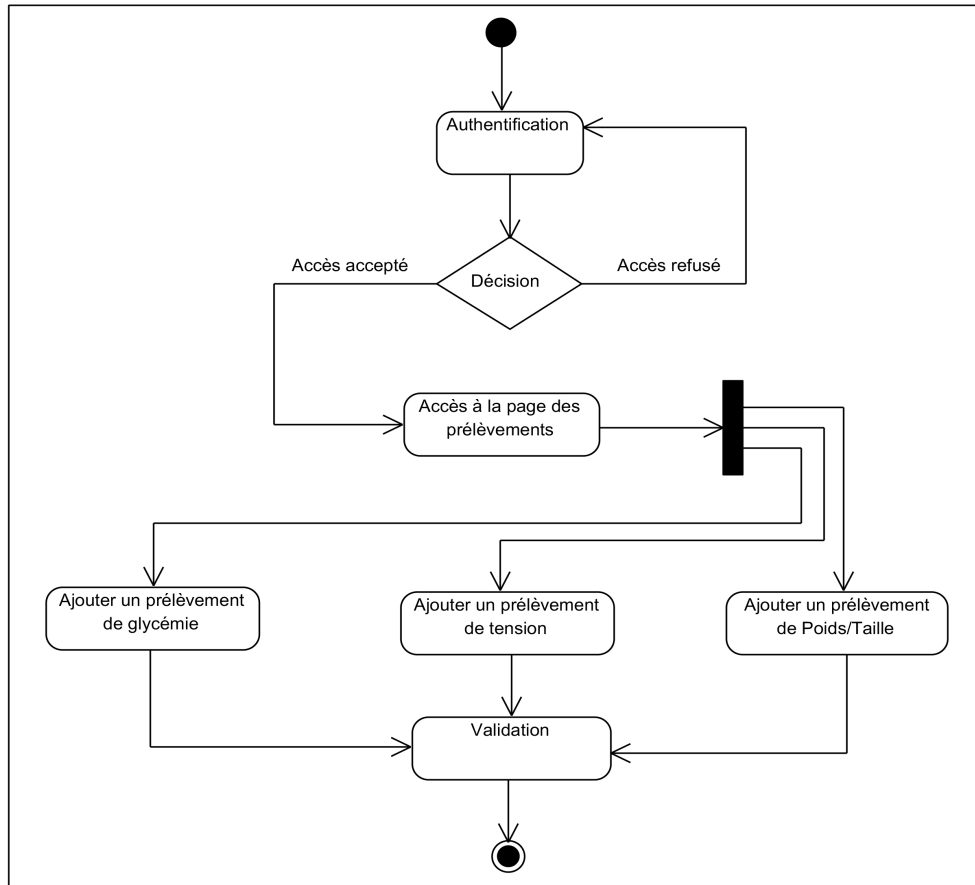


FIGURE 2.6 – Diagramme d'activité " Faire des prélèvements ".

2.4 Diagramme de classes

Le diagramme de classes constitue l'un des pivots essentiels de la modélisation avec UML. En effet, ce diagramme permet de donner la représentation statique du système à développer. Cette représentation est basée sur les concepts de classe et d'association. Chaque classe se décrit par les données et les traitements dont elle est responsable. Les traitements sont matérialisés par des opérations. Le détail des traitements n'est pas représenté directement dans le diagramme de classes, mais dans l'algorithme général et le pseudo-code correspondant [2].

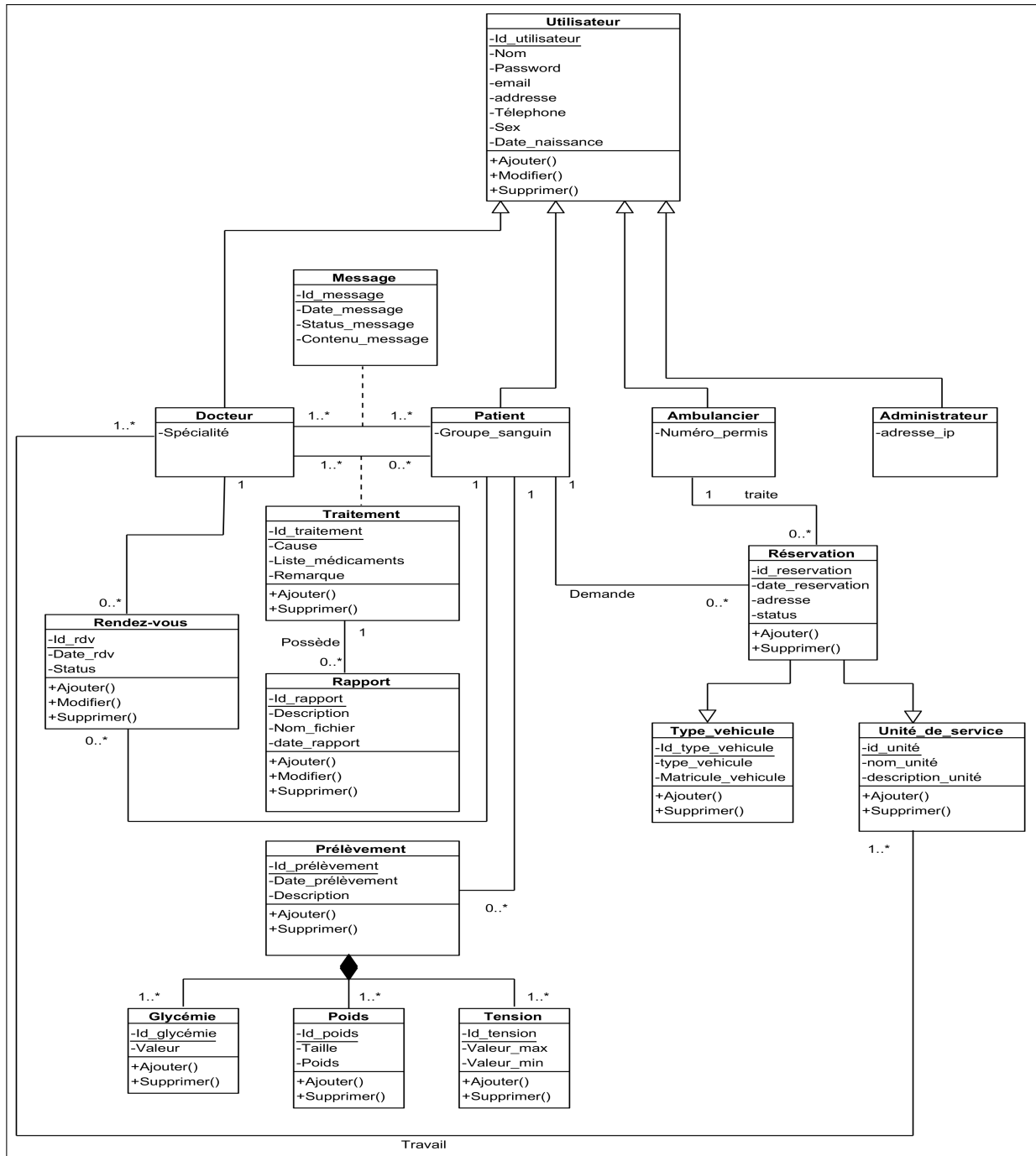


FIGURE 2.7 – Diagramme de classes.

2.4.1 Dictionnaire de données

Le tableau suivant décrit les attributs et les méthodes de chacune des classes et classes d'association du diagramme de notre application.

Classe	Attribut	Définition de l'attribut	Type
Utilisateur	Id_utilisateur	Identificateur de l'utilisateur	INT
	Nom	Nom et prénom de l'utilisateur	VARCHAR
	Date_naissance	Date de naissance de l'utilisateur	DATE
	Sex	Sexe de l'utilisateur	VARCHAR
	Adresse	Adresse de l'utilisateur	VARCHAR
	Téléphone	Numéro de téléphone de l'utilisateur	INT
	Email	Email de l'utilisateur	VARCHAR
	Password	Mot passe d'utilisateur	VARCHAR
Administrateur	Adresse_ip	Adresse IP de connexion de l'administrateur	TEXT
Docteur	Spécialité	Spécialité du docteur	VARCHAR
Patient	Groupe_sanguin	Groupe sanguin du patient	VARCHAR
Ambulancier	Numéro_permis	Numéro de permis du l'ambulancier	INT
Rendez-vous	Id_RDV	Identificateur du rendez-vous	INT
	Date_RDV	Date et heure du rendez-vous	DATE
	Statut	Statut du rendez-vous	INT
Traitement	Id_traitement	Identificateur du traitement	INT
	Cause	Cause du traitement	VARCHAR
	Liste_médicaments	Listes des médicaments du traitement	TEXT
	Remarque	Remarque sur le traitement	VARCHAR
Rapport	Id_rapport	Identificateur du rapport	INT
	Description	Description du rapport	VARCHAR
	Nom_fichier	Fichier du rapport	VARCHAR
	Date_rapport	Date et heure du rapport	DATE
Prélèvement	Id_prélèvement	Identificateur du prélèvement	INT
	Date_prélèvement	Date et heure du prélèvement	DATE
	Description	Description du prélèvement	VARCHAR
Tension	Id_tension	Identificateur du tension	INT
	Valeur_max	valeur maximale d'une tension	VARCHAR
	Valeur_min	valeur minimale d'une tension	VARCHAR
Poids	Id_poids	Identificateur du poids	INT
	Taille	Taille	LONG
	Poids	poids	LONG
Glycémie	Id_glycémie	Identificateur du glycémie	INT
	Valeur	Valeur de la glycémie	VARCHAR

Réservation	Id_réservation	Identificateur du réservation	INT
	Date_réservation	Date de la réservation	DATE
	Adresse	Adresse de la réservation	VARCHAR
	Statut	Statut de la réservation	INT
Unité_de_service	Id_unité	Identificateur de l'unité	INT
	Nom_unité	Nom de l'unité	VARCHAR
	Description_unité	Description de l'unité	VARCHAR
Type_véhicule	Id_type_véhicule	Identificateur du type de véhicule.	INT
	Type_véhicule	Le nom du type de véhicule.	VARCHAR
	Matricule_véhicule	Le numéro d'immatriculation du véhicule.	VARCHAR
Message	Id_message	Identificateur du message.	INT
	Contenu_message	Le contenu du message.	TEXT
	Date_message	La date et heure du message.	DATE
	Status_message	Statut du message.	INT

2.4.2 Modèle relationnel

Le concepteur d'une base de données relationnelle doit élaborer ce qu'il est convenu d'appeler le schéma relationnel de la base de données. Cette activité consiste à définir toutes les relations normalisées de la base de données et les domaines de leurs attributs. Théoriquement cela consisterait à décrire par intention chaque relation et définir les domaines de chaque attribut de la relation [3].

L'application des règles de dérivation du modèle relationnel à partir d'un diagramme de classes, nous permet d'avoir le modèle relationnel de la base de données de l'application à mettre en œuvre.

- Admin (Id_admin, nom, password, email, adresse, téléphone, sex, date_naissance , adresse_ip).
- Docteur (Id_docteur, nom, password, email, adresse, téléphone, sex, date_naissance, spécialité).
- Patient (Id_patient, nom, password, email, adresse, téléphone, sex, date_naissance, groupe_sanguin).
- Ambulancier (Id_ambulancier, nom, password, email, adresse, téléphone, sex, date_naissance , numéro_permis).
- Rendez-vous (Id_rdv, Date_rdv, statut, #Id_docteur, #Id_patient).
- Traitement (Id_traitement, Cause, Liste_médicaments, Remarque, #Id_docteur , #Id_patient).
- Rapport (Id_rapport, Description, Nom_fichier, date_rapport, #id_traitement).

- Prélèvement (id_Prélèvement, date_Prélèvement , description, #Id_patient).
- Glycémie (id_glycémie, valeur, #id_Prélèvement).
- Poids (id_poids, taille, poids, #id_Prélèvement).
- Tension (id_tension, valeur_max, valeur_min, #id_Prélèvement).
- Type_vehicule (id_type_vehicule, type_vehicule , matricule_vehicule).
- Unité_de_service (id_unité, nom_unité, description_unité, #id_docteur).
- Réservation (id_réservation, date_réservation, adresse, statut, #id_type_vehicule , #id_patient , #id_ambulancier , #id_unité).
- Message (id_message, Date_message, status_message, contenu_message, #id_patient , #id_docteur).

2.5 Conclusion

Dans ce chapitre, nous avons présenté l'étape d'analyse, qui nous a permis de réaliser des diagrammes de séquences suivis des diagrammes d'activité associée à chaque cas d'utilisation. Enfin, nous avons terminé par le diagramme de classes et le modèle relationnel de données qui nous permettent d'avoir le schéma de la base de données de l'application à réaliser dans le chapitre suivant.

3

Réalisation

3.1 Introduction

L'étape de réalisation est la dernière de notre projet, elle se présente comme étant l'étape la plus cruciale vu qu'elle traite l'onglet pratique du projet.

Nous commençons d'abord par une brève illustration de l'environnement de travail, ainsi que l'ensemble des logiciels que nous avons utilisé dans la réalisation du système, et l'implémentation de base de données, puis nous passons à un aperçu des interfaces les plus importantes de notre projet.

3.2 Environnement de développement

3.2.1 Le Wamp Server

WampServer (anciennement WAMP5) est une plateforme de développement Web de type WAMP, permettant de faire fonctionner localement des scripts PHP. WampServer n'est pas en soi un logiciel, mais un environnement comptant deux serveurs (Apache et MySQL), un interpréteur de script (PHP), ainsi que phpMyAdmin pour l'administration Web des bases MySQL [4].

3.2.2 Eclipse

Eclipse IDE est un environnement de développement intégré libre, extensible et polyvalent, permettant potentiellement de créer des projets de développement mettant en œuvre n'importe quel langage de programmation. Il est principalement écrit en java [13].

3.3 Les outils de développement d'application

3.3.1 Le PHP

PHP (pour pré *HyperText Processor*), est un langage de programmation libre principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP, mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale [8].

3.3.2 AJAX

AJAX (*Asynchrones JavaScript And XML*), est une méthode de développement web basée sur l'utilisation d'un script JavaScript pour effectuer des requêtes web à l'intérieur d'une page web sans recharger la page. AJAX rend plus interactifs les sites web et offre une meilleure ergonomie ainsi qu'une réactivité améliorée en permettant de modifier interactivement une partie de l'interface web seulement [8].

3.3.3 Le HTML

HTML (*Hyper Texte Markup Language*) est un langage de description d'une page web. Sa syntaxe provient du SGML (*Standard Generalized Markup Language*). Cette description porte l'extension "html" [9].

3.3.4 Le CSS

Les feuilles de styles (en anglais "*Cascading Style Sheets*", abrégé CSS) sont un langage qui permet de gérer la présentation d'une page Web. Le langage CSS est une recommandation du World Wide Web Consortium (W3C), au même titre que HTML [9].

3.3.5 JavaScript

Le Javascript est à ce jour utilisé majoritairement sur Internet, conjointement avec les pages Web (HTML ou XHTML). Le Javascript s'inclut directement dans la page Web et permet de dynamiser une page HTML, en ajoutant des interactions avec l'utilisateur, des animations, de l'aide à la navigation [10].

3.3.6 Bootstrap

Bootstrap est une collection d'outils utile à la création du design (graphisme, animation et interactions avec la page dans le navigateur ... etc) de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option [5].

3.3.7 Java

Java est un langage de programmation dont le squelette principal est constitué du langage C++. Son atout majeur réside dans l'efficacité de son utilisation sur le WEB et la réalisation de jeux fortement basés sur le graphisme [10].

3.3.8 SDK

Un SDK, pour Software Development Kit est un ensemble d'outils logiciels permettant de faciliter le développement d'un logiciel sur une plateforme donnée. Le SDK d'Android est multiplateforme, et se compose d'un plugin s'intégrant à Eclipse et d'un émulateur, permettant de tester son application sur plusieurs versions différentes d'Android, plusieurs tailles d'écran, etc. et ce, même si on n'a pas d'appareil physique [14].

3.3.9 Visual Paradigm

Visual Paradigm dispose de tous les diagrammes UML, des outils essentiellement dans le système et la conception de base de données [11].

3.3.10 Notepad++

Notepad++ est un éditeur de texte et l'éditeur de code source qui supporte plusieurs langages. Il peut aussi fonctionner sur Linux et Mac os x. Parmi les avantages principaux de Notepad++ est de travailler facilement avec plusieurs fichiers ouverts, il colore le code saisi et qui numérote vos lignes en plus, c'est un logiciel en français [6].

3.3.11 Photoshop

Photoshop est un logiciel de retouche, de traitement et de dessin assisté par ordinateur édité par Adobe. Il est principalement utilisé pour le traitement de photographies numériques, mais sert également à la création d'image [7].

3.4 Présentation des interfaces de l'application

Dans ce qui suit, nous allons présenter quelques interfaces.

3.4.1 Page d'authentification

Lors de la connexion à Health Safety, l'écran suivant vous paraîtra, c'est à partir de cet écran que vous pouvez vous authentifier pour vous connecter et d'accéder par la suite aux autres interfaces.

En cas d'erreur d'email et/ou mot de passe, et au bout de trois tentatives, le système bloquera votre adresse IP pour une durée de 15 minutes.

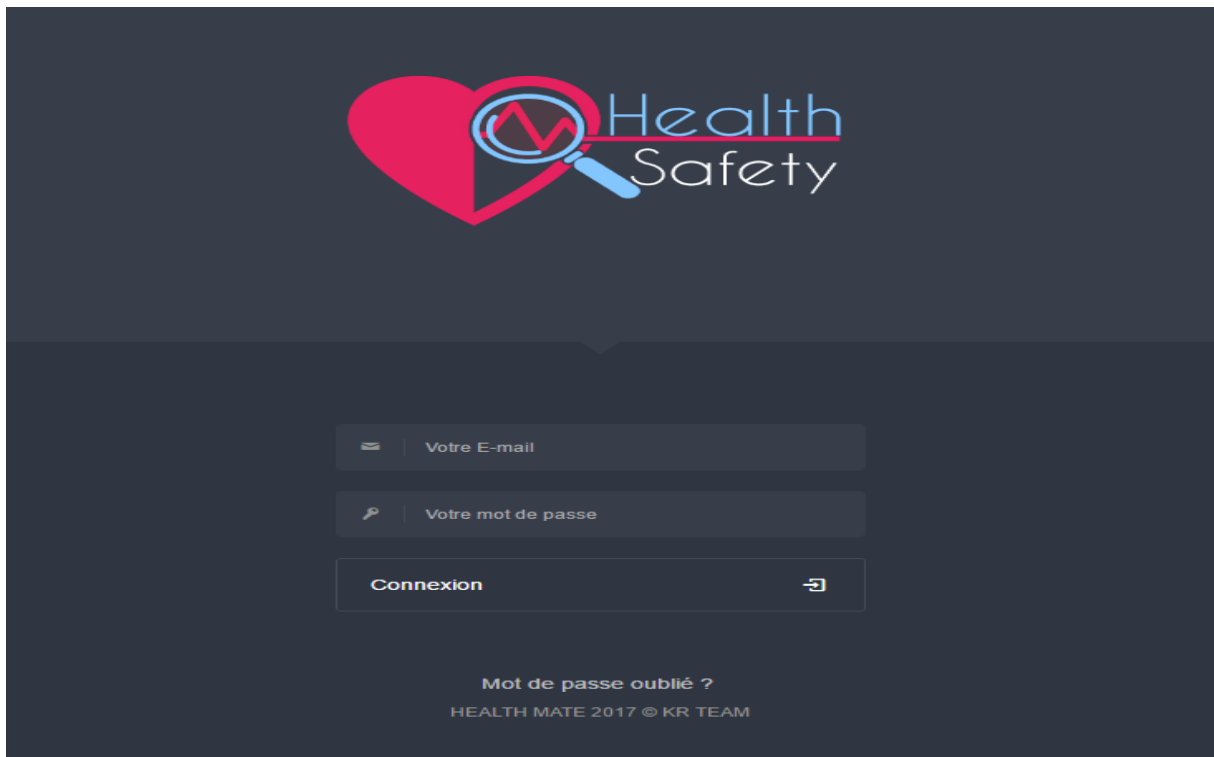


FIGURE 3.1 – Page d'authentification.

3.4.2 Page de réinitialisation de mot de passe

En cas de perte de mot de passe, Health Safety dispose d'une fonction qui permet aux utilisateurs le renvoi par mail à condition qu'il ait préalablement enregistré une adresse électronique correcte.

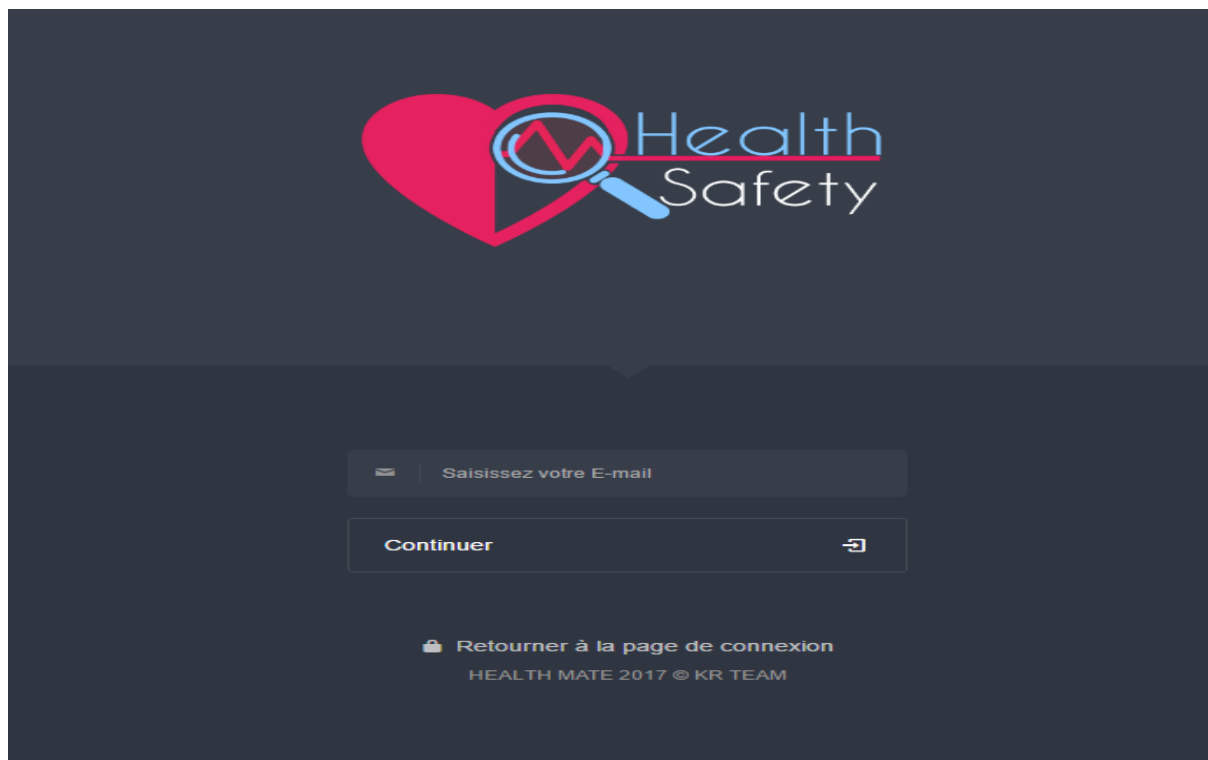


FIGURE 3.2 – Page de réinitialisation de mot de passe.

3.4.3 Interface " Administrateur "

Cette interface permet au superviseur d'avoir un contrôle total sur le système, lui permettant ainsi de gérer les différents utilisateurs (médecin, patient et ambulancier) et les unités de service, c'est-à-dire de faire des mises à jour sur ces derniers (l'ajout, la modification ou la suppression) ainsi que la possibilité de porter des modifications sur les paramètres du système (titre et la langue).

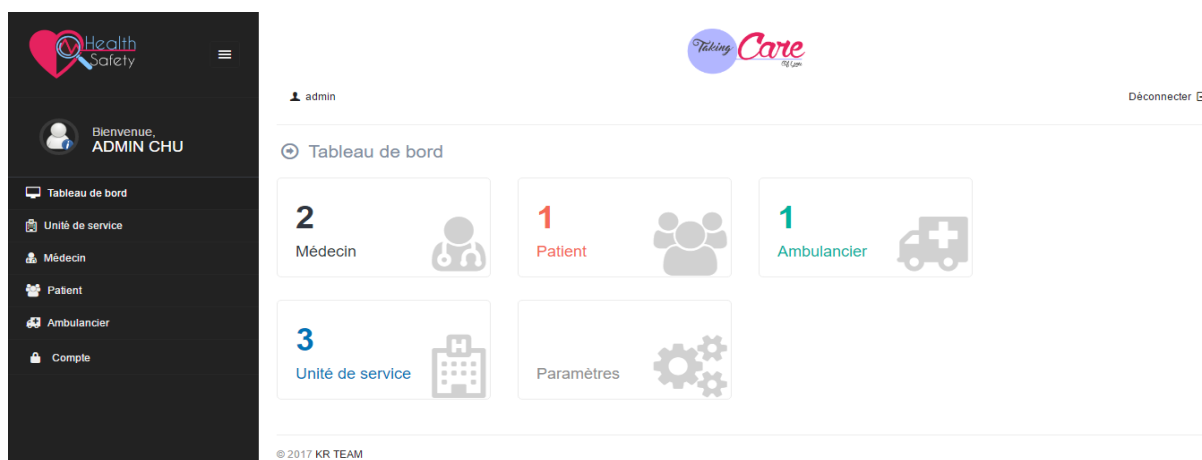


FIGURE 3.3 – Interface " Administrateur ".

● **Ajout d'un nouveau médecin**

Cliquez sur le bouton « Ajouter un médecin » de la page Médecin du panneau admin, remplissez le formulaire puis cliquez sur le bouton envoyer.

Lors de la création du nouveau compte, un courrier électronique sera automatiquement envoyé vers adresse mail du médecin en lui envoyant son mot de passe.

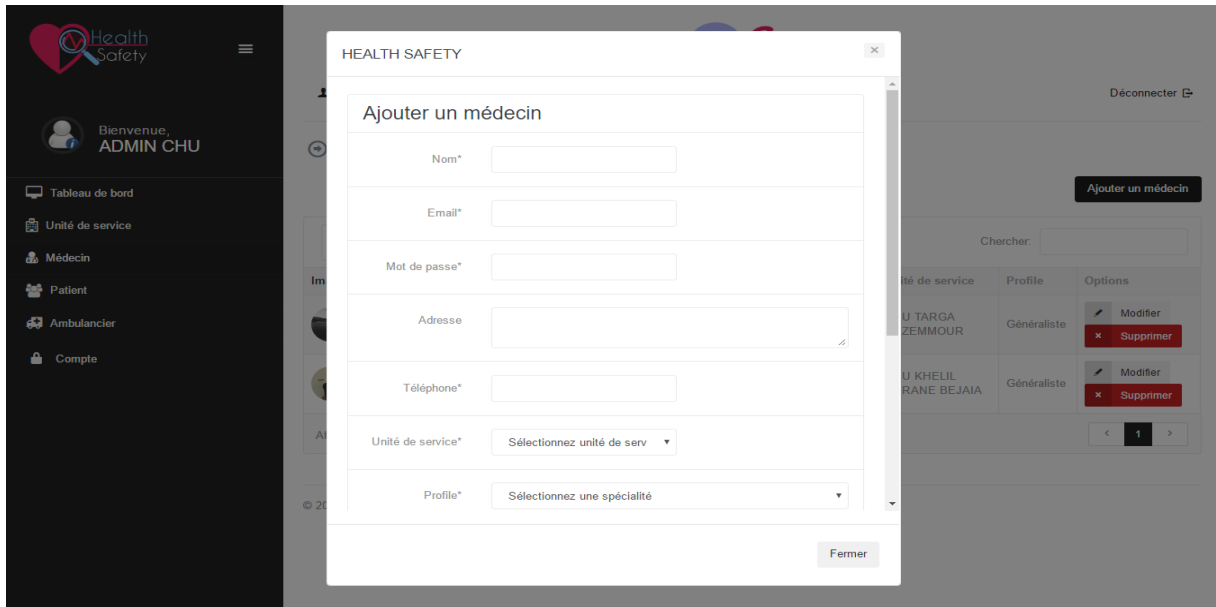


FIGURE 3.4 – *Formulaire d'ajout d'un médecin.*

● **Supprimer un médecin**

Choisissez le médecin voulu puis cliquez sur « Supprimer ».

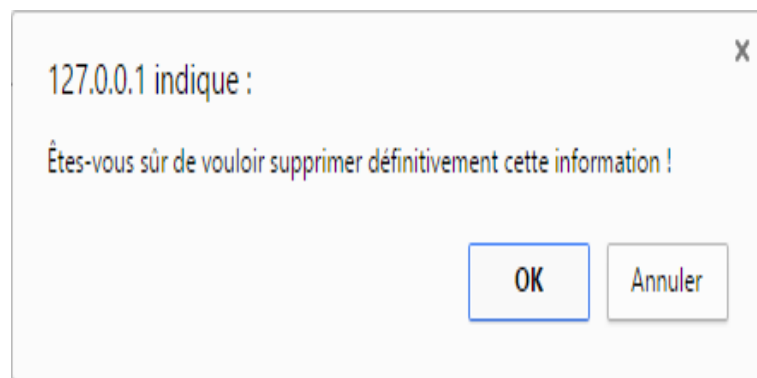


FIGURE 3.5 – *Fenêtre de "confirmation de la suppression".*

● **Modifier un médecin**

Choisissez le médecin parmi l'ensemble des médecins précédemment enregistré puis cliquez sur « Modifier ».

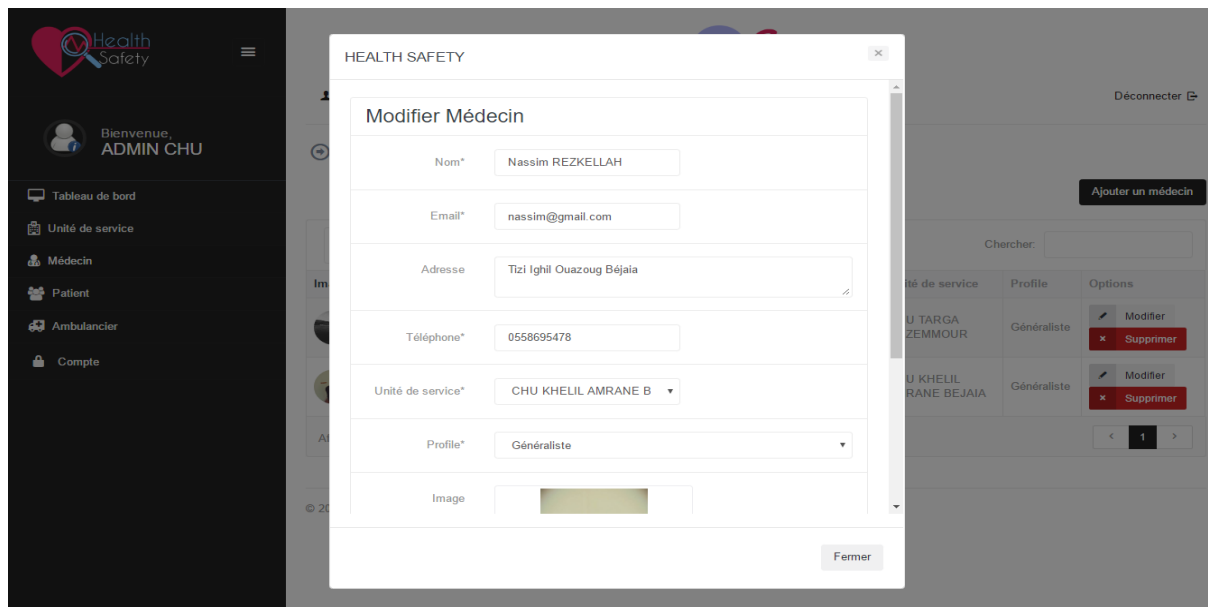


FIGURE 3.6 – Formulaire de modification d'un médecin.

● **Modifier les paramètres du système**

La page ci-après offre à l'administrateur la possibilité de modifier le titre et le nom du système, ainsi que la possibilité de choisir une des trois langues disponibles (anglais, français et l'arabe).

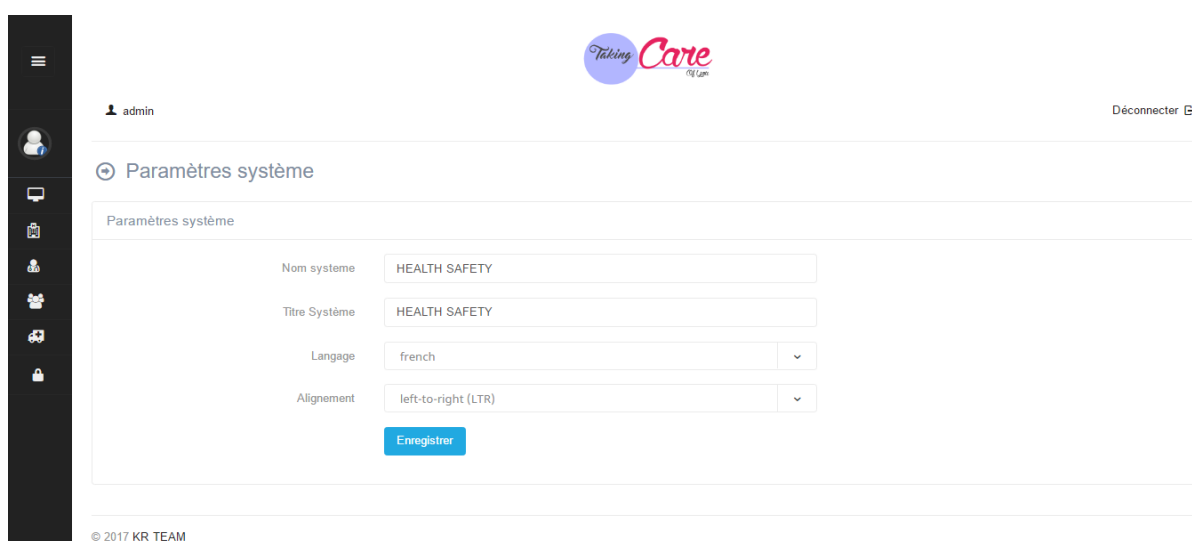


FIGURE 3.7 – Interface " Médecin ".

3.4.4 Interface " Médecin "

Cette interface offre au médecin une vue du calendrier de tous ses rendez-vous et aussi des notifications des messages envoyés par ses patients, ainsi que de faire des prescriptions pour ces derniers. De plus, le médecin pourra gérer ses patients, c'est-à-dire de leur faire des mises à jour (l'ajout, la modification ou la suppression) et d'avoir la possibilité de visualiser leur liste de prélèvements.

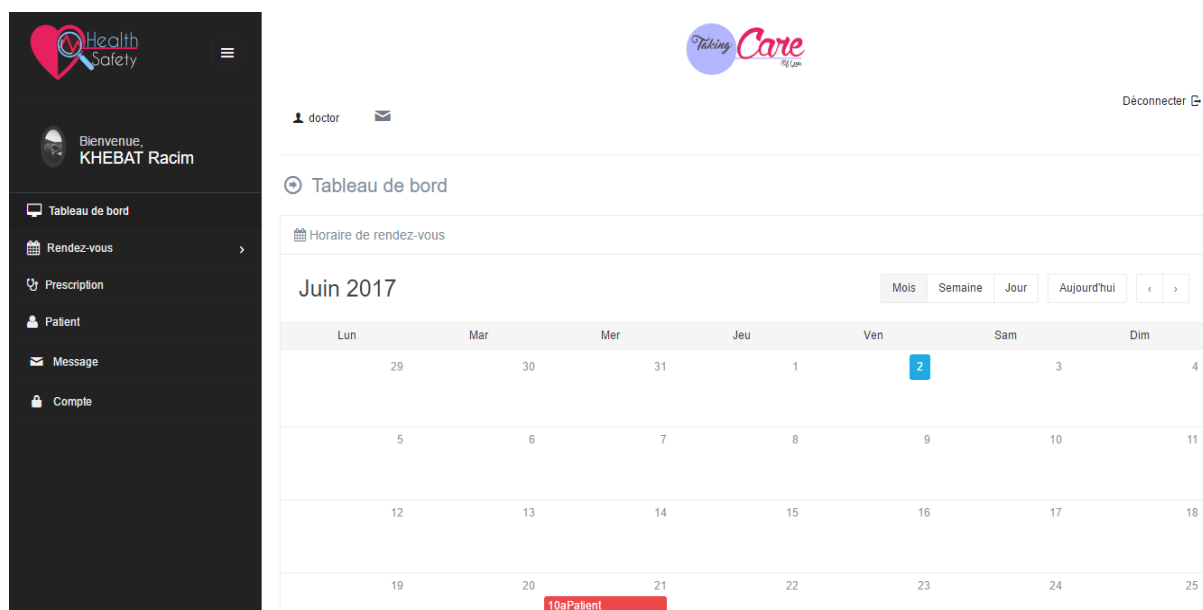


FIGURE 3.8 – Interface " Médecin ".

3.4.4.1 Page gestion des rendez-vous

Cette page contient les actions suivantes :

- **Ajout d'un nouveau rendez-vous** : Cliquez sur le bouton «Ajouter rendez-vous» de la page Horaire de rendez-vous, remplissez le formulaire puis cliquez sur « ajouter». Un courrier électronique sera automatiquement envoyé vers adresse mail du patient en lui envoyant les informations nécessaires au sujet de son rendez-vous lors de la création de ce dernier.
- **Modification d'un rendez-vous** : Dans la page Horaire de rendez-vous du panneau médecin, choisissez le rendez-vous que vous voulez modifier puis cliquez sur « Modifier».
- **Suppression d'un rendez-vous** : Dans la page Horaire de rendez-vous du panneau médecin, choisissez le rendez-vous que vous voulez supprimer puis cliquez sur « Supprimer », une petite fenêtre de confirmation vous apparaîtra choisissez « OK ».

- **Approuver un rendez-vous** : Dans la page Rendez-vous demandées du panneau médecin, choisissez le rendez-vous à approuver cliquez sur le bouton « Approuver ».

The screenshot displays the 'Gestion des rendez-vous' (Appointment Management) page. On the left is a dark sidebar with navigation icons for 'Tableau de bord', 'Rendez-vous', 'Prescription', 'Patient', 'Message', and 'Compte'. The main area features a header with the 'Taking Care' logo and user details for 'doctor'. Below this is a 'Rendez-vous' section with a search bar and a table of appointments. The table has columns for 'Date', 'Patient', 'Médecin', and 'Options'. One appointment is listed for '21/06/2017 - 10:00' with patient 'Patient' and doctor 'KHEBAT Racim'. Below the table is a calendar view for 'Juin 2017' with a grid of dates from 1 to 11.

FIGURE 3.9 – Page " Gestion des rendez-vous ".

3.4.4.2 Page gestion des Prescriptions

Cette page contient les actions suivantes :

- **Ajout d'une prescription** : Cliquez sur le bouton « Ajouter Prescription », remplissez le formulaire puis cliquez sur « soumettre ». Un courrier électronique sera automatiquement envoyé vers adresse mail du patient, lui indiquant l'arrivée d'une nouvelle prescription dans son compte Health Safety.
- **Modifier une prescription** : Faites une recherche de la prescription souhaitée puis cliquez sur le bouton « Modifier ».
- **Supprimer une prescription** : Faites une recherche de la prescription souhaitée puis cliquez sur le bouton « supprimer », une petite fenêtre de confirmation vous apparaîtra choisissez « OK ».

- **Ajouter un rapport de diagnostique** : Faites une recherche de la prescription souhaitée puis cliquez sur le bouton « Ajouter un rapport de diagnostique », remplissez le formulaire puis cliquez sur « Soumettre ».

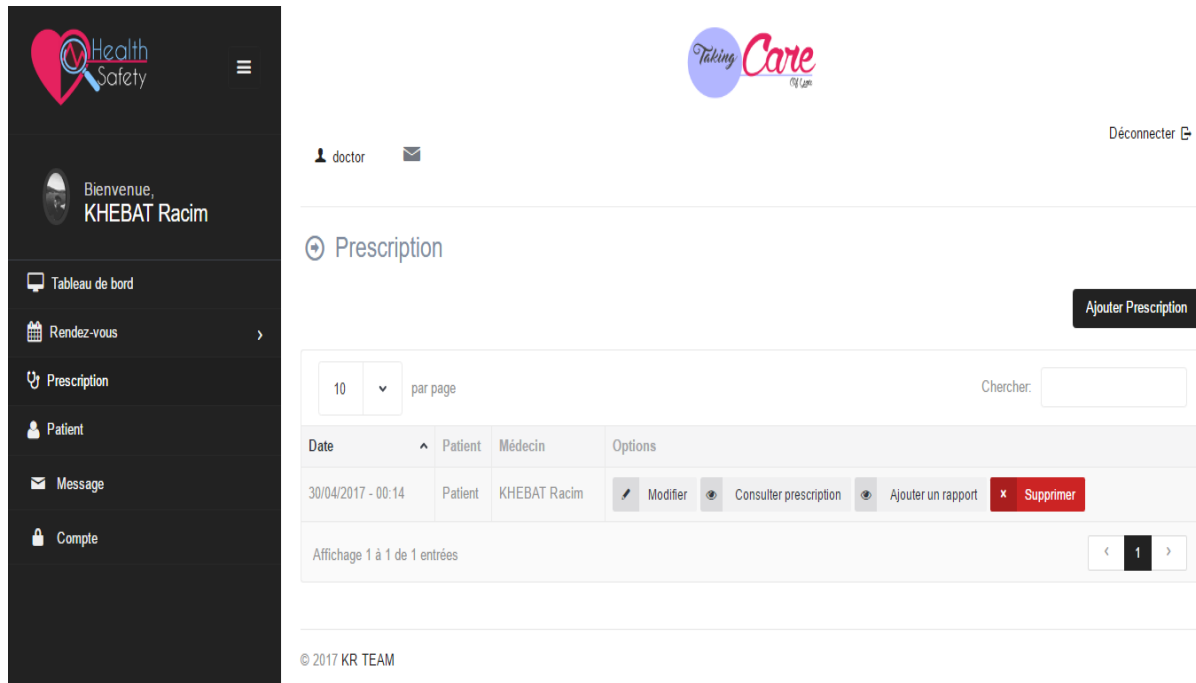


FIGURE 3.10 – Page " Gestion des prescriptions ".

3.4.4.3 Page gestion des patients

Cette page contient les actions suivantes :

- **Ajouter un nouveau patient** : Cliquez sur le bouton « Ajouter patient », remplissez le formulaire puis cliquez sur le bouton « Soumettre ».
- **Afficher le dossier médicale d'un patient** : Recherchez le patient souhaité, cliquez sur le bouton « Action » puis choisissez Profil.
- **Affichez le suivi médicale d'un patient** : Recherchez le patient souhaité, cliquez sur le bouton « Action » puis choisissez Suivi Médicale.
- **Affichez les prélèvements d'un patient** : Recherchez le patient souhaité, cliquez sur le bouton « Action » puis choisissez Afficher les prélèvements.
- **Modifier un patient** : Recherchez le patient souhaité puis cliquez sur le bouton « Modifier ».

- **Supprimer un patient** : Recherchez le patient souhaité, cliquez sur le bouton « Action » puis choisissez Effacer, une fenêtre de confirmation vous apparaîtra choisissez « OK ».

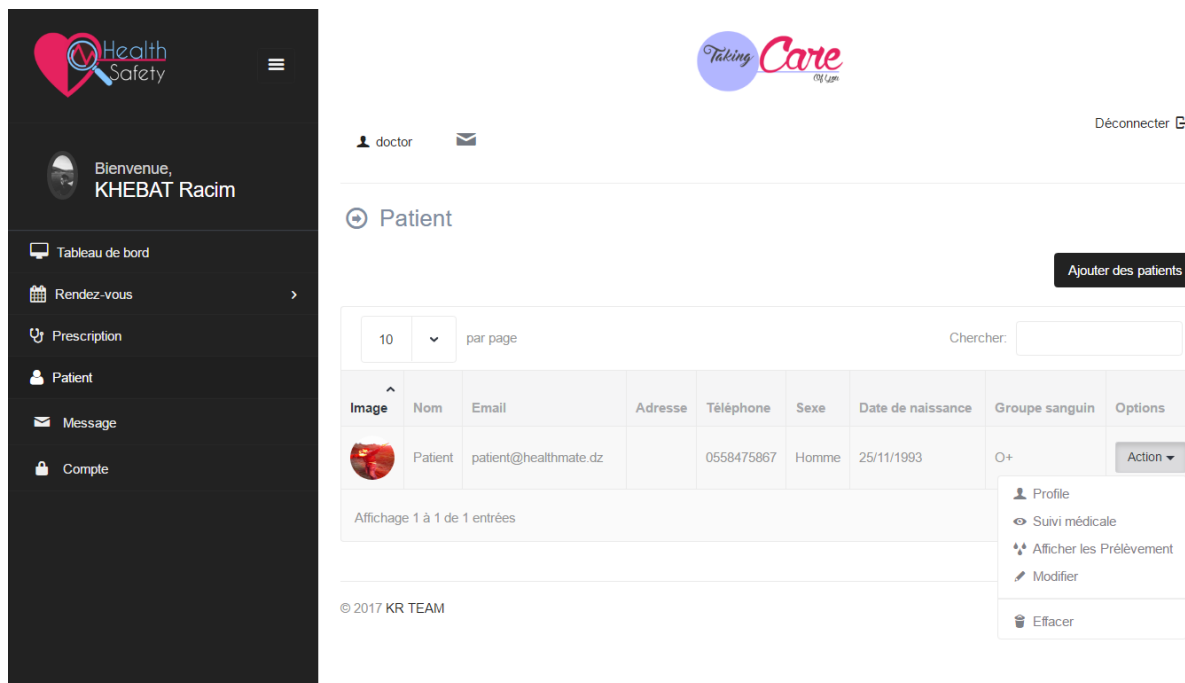


FIGURE 3.11 – Page " Gestion des patients ".

3.4.4.4 Page gestion des messages

Cliquez sur le bouton « Nouveau message », choisissez le destinataire, remplissez le message puis cliquez sur le bouton « Envoyer ».

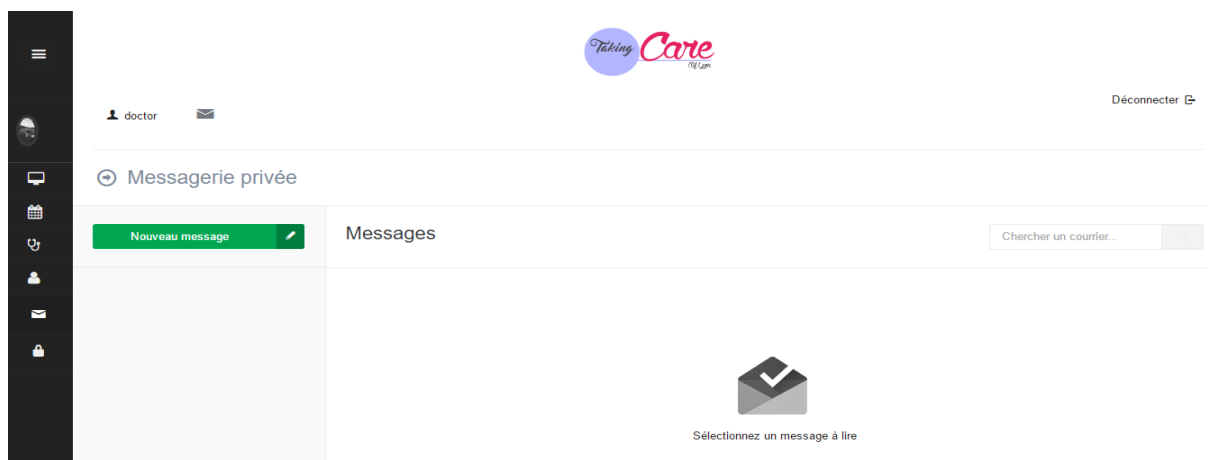


FIGURE 3.12 – Page " Gestion des messages ".

3.4.5 Interface " Patient "

Cette interface offre au patient une vue du calendrier de tous ses rendez-vous et aussi des notifications des messages envoyés par ses médecins traitants, elle permet aussi d'effectuer des prélèvements (glycémie, tension, poids et taille) ainsi que de visualiser des prescriptions. De plus, le patient peut faire une réservation de véhicule (ambulance, normal, handicap) en cas de besoin.



FIGURE 3.13 – Interface " Patient ".

3.4.5.1 Page prélèvement

Choisissez la page Prélèvement, ensuite choisissez le type de prélèvement voulu et cliquez déçu, remplissez le formulaire puis cliquez sur le bouton « Envoyer Prélèvement ».

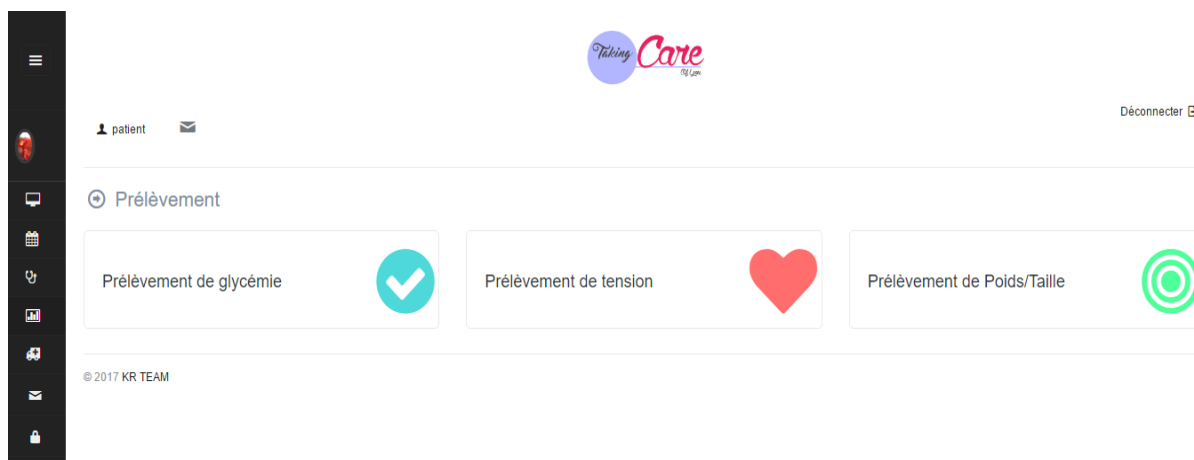


FIGURE 3.14 – Page " Prélèvement ".

3.4.5.2 Page rendez-vous

Cette page contient les actions suivantes :

- **Demander un rendez-vous à son médecin** : Cliquez sur le bouton « Demander un rendez-vous » de la page Horaire de rendez-vous, remplissez le formulaire puis cliquez sur « soumettre ».
- **Annuler un rendez-vous** : Cliquez sur le bouton « Annuler » de la page Demandes RDV en attente, une fenêtre de confirmation vous apparaîtra choisissez « OK ».

The screenshot displays the 'Prescription' page of a patient portal. On the left is a dark sidebar with navigation icons and text: 'Health Safety', 'Bienvenue, Patient', 'Tableau de bord', 'Rendez-vous' (with a dropdown arrow), 'Horaires de rendez-vous', 'Demandes RDV en attente' (with a blue notification bubble containing '1'), 'Prescription', 'Prélèvement', 'Transport', 'Message', and 'Compte'. The main content area has a top bar with the 'Taking Care' logo and a 'Déconnecter' button. Below this is a 'Prescription' section with a search bar and a table. The table has columns: 'Date', 'Patient', 'Médecin', and 'Options'. A single row is visible with the date '30 Apr, 2017 - 00:14', patient 'Patient', and doctor 'KHEBAT Racim'. The 'Options' column for this row contains two buttons: 'Consulter prescription' and 'Consulter rapport'. Below the table, it says 'Affichage 1 à 1 de 1 entrées' and a pagination control showing '1'. At the bottom left of the main area, it says '© 2017 KR TEAM'.

FIGURE 3.15 – Page " rendez-vous ".

3.4.5.3 Page prescription

Cette page contient les actions suivantes :

- **Consulter une prescription** : Dans la page Prescription, recherchez la prescription souhaitée, puis cliquez sur le bouton « Consulter Prescription ».
- **Afficher la liste des rapports de diagnostique associé à une prescription** : recherchez la prescription souhaitée, puis cliquez sur le bouton « consulter rapport ».

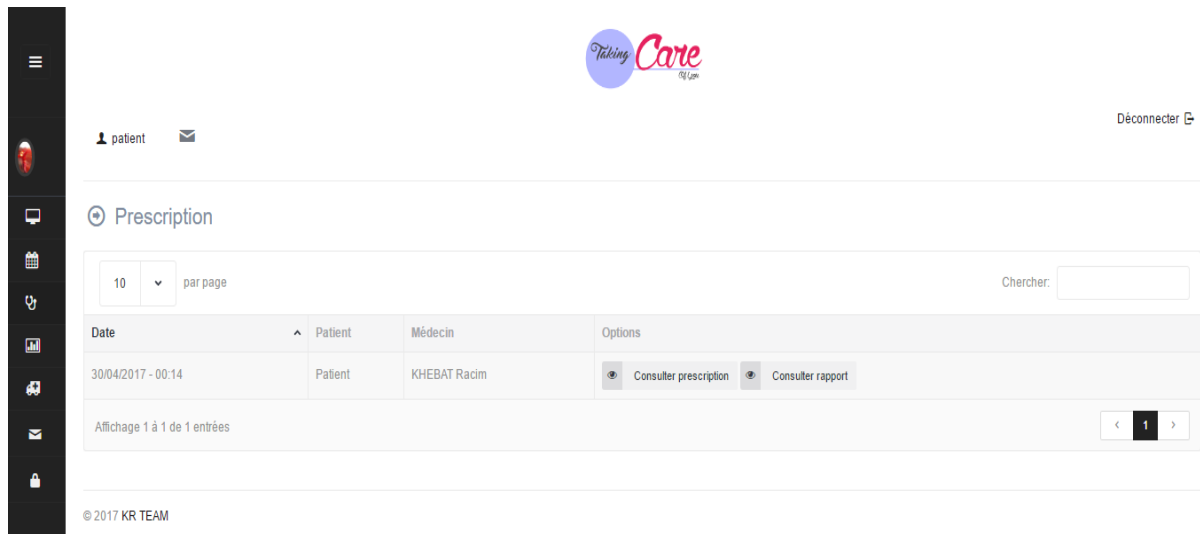


FIGURE 3.16 – Page " Prescription ".

3.4.5.4 Page transport

Cette page contient les actions suivantes :

- **Faire une réservation de véhicule** : Cliquez sur le bouton « Réserver un véhicule » dans la page Transport, remplissez le formulaire puis cliquez sur « Envoyer la demande ».
- **Consulter la localisation** : Pour bénéficier du service de géolocalisation, accédez à la page Transport, une carte géographique sera affichée par default.

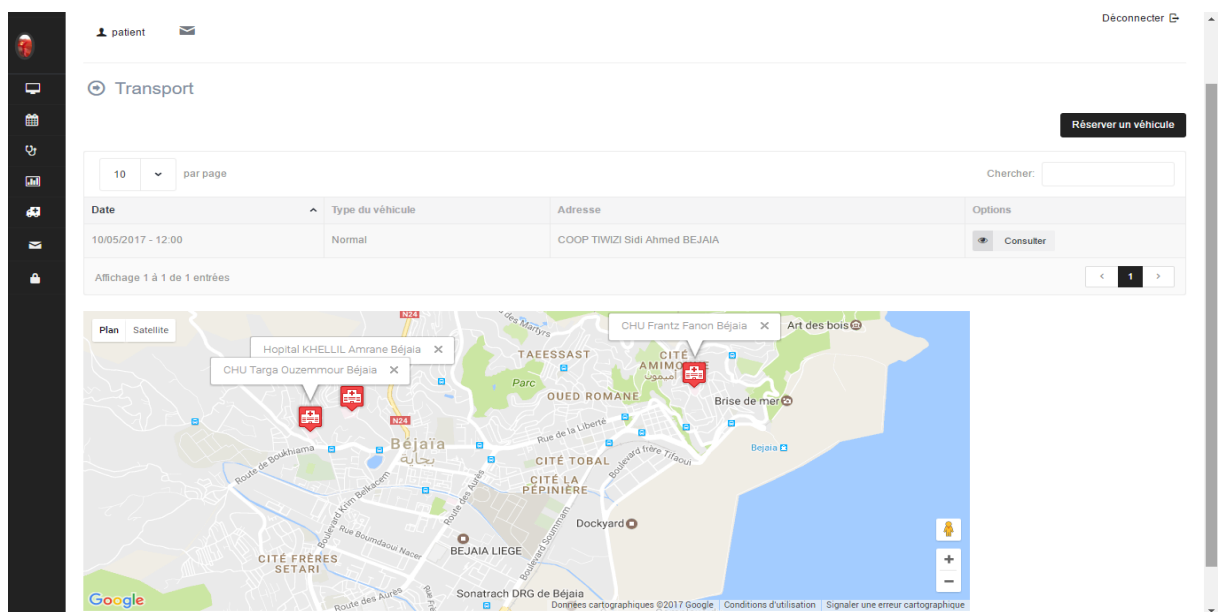


FIGURE 3.17 – Page " transport ".

3.4.6 Interface " Ambulancier "

Cette interface est restreinte aux demandes de réservations de véhicule faites par les patients.



The screenshot displays the 'Ambulancier' interface. On the left is a dark sidebar with a 'Health Safety' logo and navigation options: 'Tableau de bord', 'Demande en attente' (with a notification badge), and 'Compte'. The main content area features the 'Taking Care' logo, the user name 'ambulancier', and a 'Déconnecter' link. Below is the 'Tableau de bord des ambulancier' section, titled 'Horaires de transport approuvé'. It includes a '10 par page' dropdown, a search box, and a table with the following data:

Date	Patient	Type du véhicule	Adresse	Établissement de santé	Options
10/05/2017 - 12:00	Patient	Normal	COOP TIWIZI Sidi Ahmed BEJAIA	CHU KHELIL AMRANE BEJAIA	Consulter

At the bottom of the table, it shows 'Affichage 1 à 1 de 1 entrées' and a pagination control with '1' selected. The footer contains '© 2017 KR TEAM'.

FIGURE 3.18 – Interface " Ambulancier "

3.5 Conclusion

Dans ce chapitre, nous avons présenté les outils de développement que nous avons utilisés dans la réalisation de notre application, tout en justifiant nos choix technologiques.

À la fin de ce chapitre, nous avons présenté quelques interfaces, constituant notre application, et que nous avons jugés les plus importantes.

Conclusion générale

Notre étude menée au niveau du CHU de Béjaia avait pour but d'implémenter un système d'assistance à distance pour les patients et la vieillesse du bureau d'hospitalisation à domicile.

La première partie de notre travail a porté sur la modélisation fonctionnelle dans laquelle nous avons présenté l'organisme d'accueil et le cahier des charges du système. Puis, nous avons fait une spécification et analyse des fonctionnalités du site à travers les diagrammes de cas d'utilisation.

Après cela, nous avons fait une spécification et analyse des fonctionnalités de l'application à travers les diagrammes de séquence et d'activité. Ensuite, nous avons entamé la conception statique dans laquelle nous avons décrit le diagramme de classes associé au projet, suivi du modèle relationnel de données obtenus par l'application des règles de passage.

Enfin, nous avons réalisé notre application en utilisant plusieurs outils de développement dédiés à la programmation.

Cette expérience nous a permis d'acquérir une masse importante de connaissances aussi bien sur le plan théorique que sur le plan pratique. Cela nous a permis de mettre en œuvre un système assurant les fonctionnalités attendues, donc nous pouvons affirmer l'atteinte des objectifs visés par ce travail.

En guise de perspective, « HEALTH SAFETY » peut être enrichi par :

- L'inclusion de la fonctionnalité d'envoi des SMS au lieu des courriels électroniques ;
- Développer cette application en intégrant tous les services existant au niveau de CHU, afin de servir cette dernière.

Bibliographie

- [1] Pascal Roques, Les cahiers du programmeur UML2 modélisé une application web, 4ème Edition, EYROLLES, 2007.
- [2] Joseph Gabay et David Gabay, UML2 Analyse et conception, 1ère Édition, DUNOD, 2008.
- [3] Pascal Roque et Franck Vallée, UML 2 en action, de l'analyse des besoins à la conception, 4ème Édition, EYROLLES, 2007.
- [4] Plate-forme de développement Web sous Windows , <http://www.wampserver.com/>, accédé le 20/05/2017.
- [5] Bootstrap (framework), [https://fr.wikipedia.org/wiki/Bootstrap_\(framework\)](https://fr.wikipedia.org/wiki/Bootstrap_(framework)), accédé le 25/05/2017.
- [6] Notepad++ Home, <https://notepad-plus-plus.org/fr/> , accédé le 21/05/2017.
- [7] Adobe Photoshop CC, <http://www.adobe.com/fr/products/photoshop.html> , accédé le 21/05/2017.
- [8] Jean Marie. PHP/MYSQL avec Dreamweaver8, EYROLLES, 2006.
- [9] Rodolphe Remilé, HTML5 une référence pour le développeur web référence.
- [10] Johann Pardanaud et Sébastien de la Marck, Dynamisez vos sites web avec Javascript !, <http://www.delamarck.com/livre-dynamisez-vos-sites-web-avec-javascript.html> , accédé le 25/05/2017.
- [11] Affordable Price and Incredible Features : UML TOGAF, <https://visual-paradigm.com/features> , accédé 19/05/2017.
- [12] Centre hospital-université de Béjaia, <http://CHUBEJAIA.dz/> , accédé 18/05/2017.
- [13] Actualité scientifique et technologique, <http://www.techno-science.net/?onglet=glossaire&definition=517> , accédé 29/05/2017.
- [14] Difference between, <http://www.differencebetween.net/technology/difference-between-sdk-and-jdk/> , accédé le 31/05/2017.



Annexe A

A.1 Présentation du langage UML

UML (*Unified Modeling Language*) se définit comme un langage de modélisation graphique et textuel destiné à comprendre et à décrire des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue [1].

A.1.1 Les diagrammes UML

a) Les diagrammes structurels

Ces diagrammes structurels, ont vocation à représenter l'aspect statique d'un système (classes, objets, composants, déploiements, paquetages, structures composites) [2].

b) Les diagrammes comportement

Ces diagrammes représentent la partie dynamique d'un système réagissant aux événements et permettant de produire les résultats attendus par les utilisateurs (cas d'utilisations, d'état-transitions, d'activités, séquences, communications, globale d'interactions, temps) [2].

A.2 Le processus unifié

A.2.1 Définition

Le Processus Unifié (UP pour *Unified Process*) est un processus de développement. Il doit donc être compris comme une trame commune des meilleures pratiques de développement, et non comme l'ultime tentative d'élaborer un processus universel. La définition d'un processus UP est donc constituée de plusieurs disciplines d'activité de production et de contrôle de cette production [3].

A.2.2 Caractéristiques du processus unifié

Tout processus UP répond aux caractéristiques ci-après [1, 3] :

- **Itératif et incrémental** : C'est la meilleure pratique de gestion des risques d'ordre à la fois technique et fonctionnel. On peut estimer qu'un projet qui ne produit rien d'exécutable dans les 9 mois court un risque majeur d'échec. Chaque itération garantit que les équipes sont capables d'intégrer l'environnement technique pour développer un produit final et fournir aux utilisateurs un résultat tangible de leurs spécifications. Le suivi des itérations constitue par ailleurs un excellent contrôle des coûts et des délais.
- **Centré sur l'architecture** : Tout système complexe doit être décomposé en parties modulaires afin de garantir une maintenance et une évolution facilitées. Cette architecture (fonctionnelle, logique, matérielle, etc.) doit être modélisée en UML et pas seulement documentée en texte.
- **Piloté par les risques** : Les risques majeurs du projet doivent être identifiés au plus tôt, mais surtout levés le plus rapidement possibles. Les mesures à prendre dans ce cadre déterminent l'ordre des itérations.
- **Conduit par les cas d'utilisation** : Le projet est mené en tenant compte des besoins et des exigences des utilisateurs. Les cas d'utilisation du futur système sont identifiés, décrits avec précision et priorité.

A.2.3 Différentes phases du processus unifié

La gestion d'un tel processus est organisée suivant les quatre phases : initialisation, élaboration, construction et transition [1].

- **La phase d'initialisation** : Elle conduit à définir la "vision" du projet, sa portée et sa faisabilité afin de pouvoir décider au mieux de sa poursuite ou de son arrêt.
- **La phase d'élaboration** : Elle poursuit trois objectifs principaux en parallèle :

- Identifier et décrire la majeure partie des besoins des utilisateurs ;
 - Construire l'architecture de base du système ;
 - Lever les risques majeurs du projet.
- **La phase de construction** : Elle consiste surtout à concevoir et à implémenter l'ensemble des éléments opérationnels (autres que ceux de l'architecture de base). C'est la phase la plus consommatrice en ressources et en effort.
- **La phase de transition** : Permet de faire passer le système informatique des mains des développeurs à celles des utilisateurs finaux. Les mots-clés sont : conversion des données, formation des utilisateurs, déploiement et bêta-tests.

A.2.4 Activités du processus unifié

Ces activités de développement sont définies par cinq disciplines fondamentales qui décrivent la capture des exigences, l'analyse et la conception, l'implémentation, le test et le déploiement. La modélisation métier est une discipline amont optionnelle et transverse aux projets. Enfin, trois disciplines appelées de support complètent le tableau : gestion de projet, gestion du changement et de la configuration, ainsi que la mise à disposition d'un environnement complet de développement incluant aussi bien des outils informatiques que des documents et des guides méthodologiques. UP doit donc être compris comme une trame commune des meilleures pratiques de développement et non comme l'ultime tentative d'élaborer un processus universel [2].

A.3 Le processus 2TUP

2TUP signifie "*2 Track Unified Process*". C'est un processus UP qui répond aux caractéristiques que nous venons de citer. Le processus 2TUP apporte une réponse aux contraintes de changement continu imposées aux systèmes d'information de l'entreprise. En ce sens, il renforce le contrôle sur les capacités d'évolution et de correction de tels systèmes. "2 Track" signifie littéralement que le processus suit deux chemins. Il s'agit des chemins "fonctionnels" et "d'architecture technique", correspond aux deux axes de changement imposés au système informatique. À l'issue des évolutions du modèle fonctionnel et de l'architecture technique, la réalisation du système consiste à fusionner les résultats des deux branches. Cette fusion conduit à l'obtention d'un processus de développement en forme de Y [2].

Dans ce qui suit nous décrivons les branches composantes du processus de développement en " Y " [2] :

A.3.1 La branche gauche (fonctionnelle)

Elle comporte la capture des besoins fonctionnels, qui produit un modèle des besoins focalisé sur le métier des utilisateurs. Elle qualifie au plus tôt le risque de produire un système inadapté aux utilisateurs. De son côté, la maîtrise d'œuvre consolide les spécifications et en vérifie la cohérence et l'exhaustivité l'analyse, qui consiste à étudier précisément la spécification fonctionnelle de manière à obtenir une idée de ce que va réaliser le système en termes de métier. Les résultats de l'analyse ne dépendent d'aucune technologie particulière.

A.3.2 La branche droite (architecture technique)

Elle comporte la capture des besoins techniques qui recense toutes les contraintes et les choix dimensionnant la conception du système. Les outils et les matériels sélectionnés ainsi que la prise en compte de contraintes d'intégration avec l'existant conditionnent généralement des prés requis d'architecture technique. La conception générique, qui définit ensuite les composants nécessaires à la construction de l'architecture technique. Cette conception est la moins dépendante possible des aspects fonctionnels. Elle a pour objectif d'uniformiser et de réutiliser les mêmes mécanismes pour tout un système. L'architecture technique construit le squelette du système informatique et écarte la plupart des risques de niveau technique. L'importance de sa réussite est telle qu'il est conseillé de réaliser un prototype pour assurer sa validité.

A.3.3 La branche du milieu

Cette branche comporte la conception préliminaire qui représente une étape délicate, car elle intègre le modèle d'analyse dans l'architecture technique de manière à tracer la cartographie des composants du système à développer notamment :

- la conception détaillée qui étudie ensuite comment réaliser chaque composant ;
- l'étape de codage qui produit ces composants et teste au fur et à mesure les unités de code réalisées ;
- l'étape de recette qui consiste enfin à valider les fonctions du système développé.

Résumé

Notre projet consiste dans l'étude, conception et réalisation d'un système d'assistance distance pour les patients et la vieillesse pour le compte de CHU de Béjaia, qui permet l'amélioration de la communication et l'échange de l'information entre l'environnement d'accueil et ses patients. Pour ce faire, nous avons choisi comme langage de modélisation le formalisme UML. En effet, notre choix est justifié par rapport sa simplicité, richesse et performance en matière de conception.

Concernant l'implémentation de notre base de données, nous avons utilisé MySQL comme serveur de gestion de base de données.

L'application a été développée en utilisant différents logiciels informatiques tels que Notepad++, WampServer, Adobe Photoshop, etc. Le langage de programmation utilisé est le PHP.

Mots clés : Patient, Médecin, Suivi des patients, CHU, PHP, HTML, CSS.

Abstract

Our project consists in the design and implementation of a system of remote assistance for patients and old age on behalf of Béjaia University Hospital, which allows the improvement of communication and the exchange of information Between the host environment and its patients. To do this, we have chosen the UML formalism as a modeling language. Indeed, our choice is justied in relation to its simplicity, richness and performance in Design matter. Concerning the implementation of our database, we used MySQL as a database management server.

The application has been developed using different computer software such as Dreamweaver, WampServer, JavaScript, Adobe Photoshop, etc. The programming language used is PHP.

Keywords : Patient, Doctor, Patient follow-up, CHU, PHP, HTML, CSS.

ملخص

يتألف مشروعنا في دراسة وتصميم وتنفيذ نظام الدعم عن بعد للمرضى وكبار السن نيابة عن مستشفى جامعة بجاية، مما يسمح تحسين الاتصال وتبادل المعلومات بين بيئة المضيف ومرضاه. للقيام بذلك، لقد اخترنا UML كلغة نمذجة الشكلية. في الواقع، ما يبرر خيارنا له من حيث بساطته وثراء والأداء في التصميم. وفيما يتعلق بقاعدة البيانات الخاصة بنا، لقد قمنا باستخدام MySQL لإدارة قواعد البيانات. تم تطوير التطبيق باستخدام مختلف برامج الكمبيوتر مثل Notepad++، WampServer، فوتوشوب، الخ. لغة البرمجة المستخدمة هي PHP.

الكلمات الرئيسية: المريض، الطبيب، متابعة، CHU، PHP، HTML، CSS.