

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITÉ ABDERRAHMANE MIRA DE BÉJAÏA

FACULTÉ DES SCIENCES EXACTES

DÉPARTEMENT D'INFORMATIQUE



Mémoire de Master Professionnel

Option : Génie Logiciel

T H E M E

CONCEPTION ET RÉALISATION D'UNE APPLICATION WEB
SERVICE POUR LA GESTION D'UN CABINET MÉDICAL

CAS D'ÉTUDES : "CELLULE D'ACCUEIL ET D'ORIENTATION DES
CANCÉREUX, CHU DE BEJAIA."

Présenter Par :

FERKAL Karim CHAIBI Yassina

Soutenu devant le jury composé de :

Président	Mme ADEL Karima	U. A/Mira Béjaïa.
Examineur	Mme SABRI Salima	U. A/Mira Béjaïa.
Examineur	Mr ALLEM K.	U. A/Mira Béjaïa.
Rapporteur	Mme AIT HACÈNE Souhila	U. A/Mira Béjaïa.

Promotion 2017

REMERCIEMENTS

Nous remercions Dieu tout puissant pour son aide.

Ce mémoire n'aurait pas été possible sans l'intervention d'un grand nombre de personnes, nous souhaitons les en remercier.

Nous tenons à remercier notre promotrice Mme AIT HACÈNE Souhila pour ses précieux conseils, sa confiance, sa patience qui ont constitué un apport considérable pour nous durant tous le déroulement de ce projet.

Nos vifs remerciements vont également aux membres du jury pour l'intérêt qu'ils portent à notre travail en acceptant de l'examiner et de l'enrichir par leurs propositions.

Nous tenons à saisir cette occasion pour adresser nos remerciements à Mme maître de stage qui par sa compréhension , son aide et sa disponibilité nous a permis d'accomplir notre travail.

DÉDICACE

Nous dédions ce travail à :

Nos très chers parents.

Nos frères et sœurs.

Nos grands parents.

Nos tantes, nos oncles et leurs femmes.

Nos cousins et cousines.

Tous nos adorables amis et amies.

Tous nos enseignants.

Ainsi qu'à toute personne qui nous a soutenue.

Table des Matières

Table des Matières	i
Table des Figures	v
Listes des Tableaux	vi
Introduction générale	3
1 Organisme d'accueil et analyse des besoins	5
1.1 Historique de l'hôpital	6
1.2 Objectifs de l'hôpital	6
1.3 Organigramme général de l'hôpital	7
1.3.1 Service technique et d'exploration	7
1.3.2 Services médicaux	7
1.4 Cadre d'étude	9
1.4.1 Présentation de la cellule d'accueil et d'orientation des cancéreux	9
1.4.2 Situation informatique	10
1.5 Problématiques et objectifs	10
1.6 eXtreme Programming (XP) :	11
1.7 Langage de modélisation	12
1.8 Analyse des besoins	12
1.8.1 Identification des acteurs	12
1.8.2 Les besoins fonctionnels :	12
1.8.3 Les besoins non fonctionnels	13
1.8.4 Identification des cas d'utilisation	13
1.9 Diagrammes de cas d'utilisation	14
1.9.1 Cas d'utilisation "Gérer rendez-vous"	14
1.9.2 Cas d'utilisation "Générer bilan d'activités"	15
1.9.3 Cas d'utilisation "Gérer dossier médical"	15

1.9.4	Cas d'utilisation "Gérer plage horaire"	16
1.9.5	Cas d'utilisation "Consulter dossier médical"	16
1.9.6	Cas d'utilisation "Consulter rendez-vous"	17
1.9.7	Diagrammes de cas d'utilisation "Global"	18
1.10	Description textuel des cas d'utilisation	19
1.10.1	Cas d'utilisation "s'authentifier"	19
1.10.2	Cas d'utilisation "Gérer Rendez-vous"	19
1.10.3	Cas d'utilisation "Générer Bilan d'activités"	20
1.10.4	Cas d'utilisation "Gérer dossier médical"	20
1.10.5	Cas d'utilisation "Gérer Plage Horaire"	21
1.10.6	Cas d'utilisation "Consulter Dossier"	21
1.10.7	Cas d'utilisation "Consulter Rendez-vous"	22
1.11	Environnement de développement	22
1.11.1	Netbeans	22
1.11.2	Tomcat	22
1.11.3	PhpMyAdmin	22
1.11.4	GlassFish	23
1.11.5	SoapUI	23
1.11.6	Visual Paradigme	23
1.12	Langages de développement	23
1.12.1	JAVA	23
1.12.2	Bootstrap	24
1.12.3	JavaServer Page (JSP]	24
1.12.4	Mysql	24
1.12.5	PHP	24
1.13	Conclusion	25
2	Les services Web	26
2.1	Définition	27
2.2	Architecture des services web	27
2.2.1	Architecture de référence	28
2.2.2	Architecture etendue	29
2.3	Technologies de bases	30
2.3.1	XML	30
2.3.2	La communication SOAP	31
2.3.3	Le langage WSDL	31
2.3.4	UDDI	31

2.4	Les problèmes liés au service web	32
2.5	Les avantages des services web	32
2.6	Conclusion	32
3	Conception	33
3.1	Diagrammes de séquence des cas d'utilisation	34
3.1.1	Diagramme de séquence du cas d'utilisation "s'authentifier"	34
3.1.2	Diagramme de séquence du cas d'utilisation "Gérer Rendez-vous"	35
3.1.3	Diagramme de séquence du cas d'utilisation "Gérer Dossier médical"	38
3.1.4	Diagramme de séquence du cas d'utilisation "Gérer Plage Horaire"	42
3.1.5	Diagramme de séquence "Générer Bilan d'activités"	46
3.2	Diagramme de classes	47
3.3	Passage au modèle relationnel	48
3.3.1	Règles de passage du diagramme de classes au modèle logique	48
3.3.2	Modèle logique de données	49
3.4	Conclusion	49
4	Implémentation	50
4.1	Interfaces et déroulement de l'application	51
4.1.1	Interface d'authentification	51
4.1.2	Interface secrétaire	51
4.2	Interface Médecin	56
4.2.1	Consulter Dossier médicale	57
4.2.2	Consulter Rendez-vous médecin IRM	58
4.3	Conclusion	59
	Conclusion générale	60
	<i>Bibliographie</i>	61

Table des figures

1.1	Logo de l'hôpital KHELLIL AMRANE	6
1.2	Organigramme de l'hôpital	8
1.3	Diagramme de cas d'utilisation "Gérer rendez-vous"	14
1.4	Diagramme de cas d'utilisation "Générer bilan d'activités"	15
1.5	Diagramme de cas d'utilisation "Gérer dossier médical"	15
1.6	Diagramme de cas d'utilisation "Gérer plage horaire"	16
1.7	Diagramme de cas d'utilisation "Consulter dossier médical"	16
1.8	Diagramme de cas d'utilisation "Consulter rendez-vous"	17
1.9	Diagramme de cas d'utilisation "Global"	18
2.1	Architecture de référence de service web	29
2.2	Architecture en pile	30
3.1	Diagramme de séquence du cas d'utilisation "authentification"	34
3.2	Diagramme de séquence du cas d'utilisation "Affecter un Rendez-vous"	35
3.3	Diagramme de séquence du cas d'utilisation "Rechercher Rendez-vous"	36
3.4	Diagramme de séquence du cas d'utilisation "Annuler Rendez-vous"	37
3.5	Diagramme de séquence du cas d'utilisation "Consulter Rendez-vous"	37
3.6	Diagramme de séquence du cas d'utilisation "Ajouter dossier"	38
3.7	Diagramme de séquence du cas d'utilisation "Rechercher dossier"	39
3.8	Diagramme de séquence du cas d'utilisation "Supprimer dossier"	40
3.9	Diagramme de séquence du cas d'utilisation "Consulter dossier"	40
3.10	Diagramme de séquence du cas d'utilisation "Modifier dossier"	41
3.11	Diagramme de séquence du cas d'utilisation "Ajouter Plage horaire"	42
3.12	Diagramme de séquence du cas d'utilisation "Rechercher une Plage Horaire"	43
3.13	Diagramme de séquence du cas d'utilisation "Annuler une Plage Horaire"	44
3.14	Diagramme de séquence du cas d'utilisation "Consulter une Plage Horaire"	44
3.15	Diagramme de séquence du cas d'utilisation "Modifier une Plage Horaire"	45
3.16	Diagramme de séquence du cas d'utilisation "Générer Bilan d'activités"	46

3.17 Diagramme de classes	47
4.1 Interface d'authentification	51
4.2 Interface Gestion des Rendez-vous	52
4.3 Méthode d'invocation	52
4.4 Interface de l'annuaire	53
4.5 WSDL du fichier	53
4.6 Interface Ajouter un Dossier médical	54
4.7 Méthode d'invocation	54
4.8 Interface de l'annuaire	55
4.9 WSDL du fichier	55
4.10 Interface Générer Bilan d'activités	56
4.11 Interface Impression bilan	56
4.12 Interface Gérer Plage horaire	57
4.13 Méthode d'invocation	57
4.14 interface Consulter Dossier médicale	58
4.15 interface Consulter Dossier médicale	58

Liste des tableaux

1.1	Liste des cas d'utilisation	14
1.2	Cas d'utilisation "s'authentifier"	19
1.3	Cas d'utilisation "Gérer Rendez-vous"	19
1.4	Cas d'utilisation "Générer Bilan d'activités"	20
1.5	Cas d'utilisation "Gérer dossier médical"	20
1.6	Cas d'utilisation "Gérer Plage Horaire"	21
1.7	Cas d'utilisation "Consulter Dossier"	21
1.8	Cas d'utilisation "Consulter Rendez-vous"	22

Liste des Abréviations

CHU Centre Hospitalo-Universitaire

CSS Cascade Style Sheet

EDI Environnement de Développement Intégré

JDK Java Development Kit

JSP JavaServer Page

HTML HyperText Markup Language

HTTP Hypertext Transfert Protocole

PHP Hypertext Preprocessor

SGBDR Système de gestion de base de données relationnel

SOA Architecture Orientée Service

SOAP Simple Object Access Protocol

SQL Structured Query Language

UDDI Universal Description Discovery and Integration

URI Universal Ressource Identifier

WSDL Web Service Description Language

XML Extensible Markup Language

XP eXtrême Programming

Introduction Générale

Les services web permettent aujourd'hui l'utilisation d'applications distante par d'autres applications. L'architecture des services web repose sur trois standards : WSDL, UDDI, et SOAP. Une fois publiées les services web sont potentiellement utilisable par les utilisateurs de profils hétérogènes à partir des différents dispositifs d'accès.

L'utilisation des services web rend possible la réalisation d'environnements technologiques distribués dans lesquels des applications peuvent interagir entre elles de façons hétérogènes et ce de manière indépendante de la plateforme physique, des langages de programmation ou encore des systèmes d'exploitation.

Les services web sont développés pour que les développeurs les utilisent dans tous les domaines de l'informatique, afin de récolter les divers bénéfices de cette technologie, ils représentent aujourd'hui la technologie la plus adaptée pour le développement des systèmes d'information distribuées sur l'internet.

La cellule d'accueil et d'orientation des cancéreux de l'hôpital KHALLIL AMRANE de bejaia ainsi que le service IRM se retrouvent confrontés au problèmes de gestion des opérations effectuées au sein de ces services et qui garantissent leur bon fonctionnement.

sachant que ces services ne disposent pas d'une application leurs permettant la gestion de leurs rendez-vous ou des dossiers de malades du service, les employés se retrouve surcharger par des tâches qui pourraient facilement être simplifiées, rajouter à cela, le manque de collaboration entre le personnel de la cellule d'accueil et d'orientation et celui du service d'IRM sachant que les deux services sont dépendants car les patients doivent avoir des rendez-vous en IRM pour leur radios.

Dans cette optique, est né le besoin d'une application qui permettra à l'hôpital KHALLIL AMRANE et en particulier au service de la cellule d'accueil et d'orientation des cancéreux

et celui de l'IRM de profiter de multiples fonctionnalités qu'offre les services web. Notre travail est de promouvoir et de faciliter la collaboration entre les utilisateurs et de gérer les ressources. En effet, un manque de structuration et de coordination donnerait lieu à une perte d'efficacité de l'hôpital et donc l'informatisation de la gestion du service de la cellule d'accueil et d'orientation des cancéreux et celui de l'IRM permettrait de palier à ce problème.

Le présent document expose les différentes étapes franchies afin de mener à bien ce projet et qui se traduisent à travers quatre chapitres.

Nous commençons dans le premier chapitre par introduire le cadre général du travail en présentant l'organisme d'accueil et la problématique traitée dans ce projet. Par la suite, nous établirons une description complète du comportement du système à développer, où nous allons présenter les différents besoins fonctionnels et non fonctionnels capturés, ainsi qu'une identification des acteurs du système. Ensuite, nous analyserons les besoins à travers l'élaboration des diagrammes de cas d'utilisation, qui seront joints d'une description textuelle détaillée et on termine par la présentation des outils utilisés pour le développement.

Le second chapitre est consacré à la définition de la notion de services web ainsi que les raisons principales de leur apparition et les différents avantages qu'ils offrent.

La phase de conception fait l'objet du troisième chapitre dans lequel nous détaillons notre solution. Nous présentons les diagrammes de séquence et le diagramme de classes ainsi que le modèle logique de données nécessaires à la conception de la solution proposée.

Enfin, nous exposons dans le quatrième et dernier chapitre le travail réalisé en présentant les différentes interfaces.

Nous concluons ce rapport par une conclusion générale résumant les principales fonctionnalités réalisées et proposant quelques perspectives en vue d'élargir et d'améliorer ce travail.

1

ORGANISME D'ACCUEIL ET ANALYSE DES BESOINS

Introduction

Dans ce chapitre, nous présentons les différents services de l'hôpital KHELLIL AMRANE de Bejaia, ainsi que leurs capacités et l'organigramme général de ses services. Ensuite, nous allons exposer la problématique et les objectifs de notre projet et on termine par la représentation des diagrammes de cas d'utilisation.

1.1 Historique de l'hôpital

Le centre hospitalo-universitaire KHELLIL AMRANE de Bejaia est un établissement public à caractère administratif, doté de la personnalité morale et de l'autonomie financière. Il a été créé en 1991 par décret exécutif, sur proposition conjointe du ministre chargé de la santé et du ministre chargé de l'enseignement supérieur et de la recherche scientifique. En 2011, l'hôpital KHELLIL AMRANE est devenu le siège du centre Hospitalo-universitaire (CHU) de Bejaia. La création de ce dernier a été faite suite à l'inauguration de la faculté de médecine à l'université A. Mira de Bejaia au campus d'ABOUDAOU [1]



FIGURE 1.1 – Logo de l'hôpital KHELLIL AMRANE

1.2 Objectifs de l'hôpital

Durant notre stage au sein de l'hôpital KHELLIL Amrane, nous avons pu constater ses objectifs principaux qui sont :

- Soins à haut niveau.
- Formation médicale et soins infirmiers.
- Recherches en science de la santé.
- Développer les systèmes d'information, de communication et audio-visuel.

- Soutenir les opérations répondant aux critères d'efficience.
- Assurer les mises en normes de sécurité.

1.3 Organigramme général de l'hôpital

L'hôpital KHELLIL AMRANE de Bejaia comporte plusieurs services médicaux que nous allons présenter dans la figure 1.1.

1.3.1 Service technique et d'exploration

- **Laboratoire d'analyse médicale** : c'est un lieu où sont prélevés et analysés divers fluides biologique d'origine humaine sous la responsabilité des biologistes médicaux qui interprètent les résultats dans le but de participer au diagnostic et au suivi de certaines maladies.
- **Radiologie et imagerie médicale** : la radiologie dans le domaine médical, désigne l'ensemble des modalités, diagnostiques et thérapeutiques utilisent les rayons X, ou plus généralement qui utilisent des rayonnements.

1.3.2 Services médicaux

- **Orthopédie traumatologie** : il prend en charge la traumatologie quotidienne, orthopédie, prothétique, rachidienne, tumorale infantile, arthroscopie.
- **Urgences médico-chirurgicales** : réception et prise en charge des urgences vitales, médicales, chirurgicales, tri et consultation.
- **Médecine interne** : représentée par l'accueil des patients consultants aux urgences et nécessitant une hospitalisation ; pour des motifs aussi divers qu'un syndrome corona-rien aigu, une phlébite, une ischémie aigue d'un membre, une insuffisance cardiaque, un accident vasculaire cérébral, décompensation aigue du diabète sucré, les pathologies gastro-intestinales, une maladie cancéreuse, etc. La médecine interne prend en charge dans leurs globalité ses patients, notamment quand plusieurs pathologies associées rendent les choses complexes.

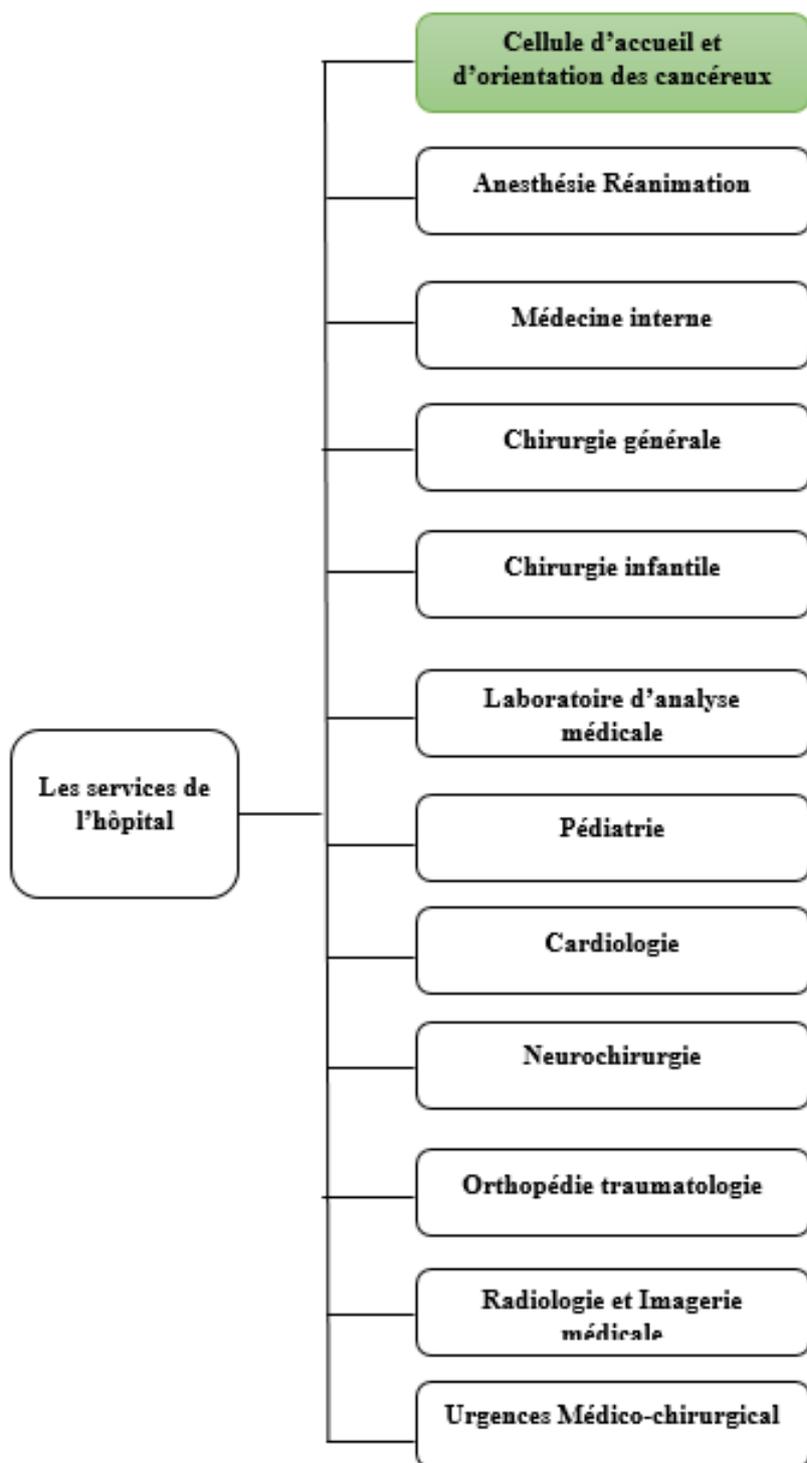


FIGURE 1.2 – Organigramme de l'hôpital

- **Cardiologie** : la cardiologie est la spécialité médicale qui étudie le cœur et ses maladies. Elle regroupe des activités d'hospitalisation, de consultations et d'explorations cardiaques permettent de prendre en charge la prévention, le dépistage, le diagnostic mais aussi le traitement de toutes les pathologies cardiaques.
- **Pédiatrie** : c'est une branche spécialisée de la médecine qui étudie le développement psychomoteur et physiologique normal de l'enfant, ainsi que toute la pathologie qui y a trait, de la naissance à la période post pubertaire ou il devient adulte.
- **Anesthésie et réanimation** : le service accueille des patients ayant plusieurs défaillances d'organes et nécessitant une réanimation, des soins intensifs, et une surveillance continue.
- **Chirurgie générale** : : la discipline mère de toutes les spécialités chirurgicales, c'est une spécialité qui s'intéresse à l'étude du diagnostic et du traitement.
- **Chirurgie infantile** : une spécialité de la chirurgie qui prend en charge les maladies des enfants.
- **Neurochirurgie** : le service neurochirurgie diagnostic et traite par la chirurgie les maladies touchant le cerveau ou la colonne vertébrale.
- **Cellule d'accueil et d'orientation des cancéreux** : : le rôle de ce service est l'accueil, l'accompagnement, prise en charge psychologique et prise de rendez-vous pour les malades.[1]

1.4 Cadre d'étude

Nous avons effectué notre stage au sein de l'hôpital KHELLIL Amrane au niveau de la cellule d'accueil et d'orientation des cancéreux.

1.4.1 Présentation de la cellule d'accueil et d'orientation des cancéreux

Le service a été créé en décembre 2010 afin de prendre en charge les malades atteint du cancer. Il est composé de deux médecins, une secrétaire et un psychologue.

Dans le but d'améliorer l'organisation et la prise en charge des patients, le service d'accueil et d'orientation des cancéreux utilise MS-Excel pour la gestion des informations (bilans des activités, Fiches malades, etc.) et la planification des rendez-vous.

1.4.2 Situation informatique

Les équipements utilisés sont les suivants :

1. Matériel

Le service dispose du matériel suivant :

— Des ordinateurs caractérisés par :

- Processeur Intel pentium 3,5 GHz.
- Capacité mémoire 2 à 4 Go de RAM.
- Système d'exploitation Windows 7.

— Des imprimantes de type :

- CANON MF4430.

— Des onduleurs de type :

- Maxi power.

2. Logiciel

Ce service utilise MS-Excel pour établir la liste des rendez-vous et pour gérer ces derniers.

1.5 Problématiques et objectifs

— **Problématiques**

Les principaux problèmes que nous avons détectés lors de notre séjour au service et qui ont vraiment attiré notre attention sont :

- Difficulté de la recherche et d'accès à l'information.
- Génération manuel des bilans d'activités ce qui peut engendrer des erreurs.

- Le patient doit s'occuper lui-même de la prise de rendez-vous dans un des services relia à la cellule tels que IRM, chimiothérapie. . .

- L'insécurité des données.

— **Objectifs de notre travail**

Pour résoudre les problèmes rencontrés par ce service qui réside essentiellement dans sa gestion, nous proposons de développer une application orientée service web pour la gestion de ce service qui permettra principalement de :

- Garantir la sécurité des données par une authentification.

- Solution web qui sera utilisé par n'importe quel service en utilisant les architectures orienté service basé sur la technologie XML.

- La prise de rendez-vous se fera désormais via une plateforme reliée au serveur de l'hôpital.

- Affectation des rendez-vous selon la disponibilité des médecins.

Pour la réalisation de notre application nous allons suivre la méthodologie de travail eXtrême Programming. Nous avons choisi cette méthodologie car notre travail nécessite l'implication intense du client et que nous serons tous le temps confronter à adapter notre travail selon les besoins de notre organisme d'accueil.

1.6 eXtreme Programming (XP) :

C'est un ensemble de pratique qui couvre toutes les activités de réalisation d'un logiciel. Ces pratiques n'ont rien de révolutionnaire : il s'agit simplement de pratique de bon sens mises en œuvre par des développeurs ou des chefs de projet expérimentés, telles que les livraisons fréquentes, la relecture du code et la mise en place de testes automatiques. . . [2]
Nous allons commencer par la première phase de la méthode XP, qui est l'analyse des besoins.

1.7 Langage de modélisation

UML est un langage de modélisation graphique. 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. En effet, l'UML nous permet une meilleure conception du côté de l'application avec ses notions d'objets et de classes, et nous donne une décomposition claire et simple afin de dégager les entités et les classes nécessaires.[3]

1.8 Analyse des besoins

1.8.1 Identification des acteurs

Un acteur représente un rôle joué par une personne externe ou par un processus qui interagit avec le système.[13]

Les acteurs de notre système sont :

- **Secrétaire** : il s'agit d'un acteur qui s'occupe de la gestion des dossiers médicaux, prise de rendez-vous et génération des bilans d'activités.

- **Médecin responsable de la cellule et médecin responsable de service IRM** : il s'agit de deux acteurs qui peuvent planifier les plages horaires et consulter le dossier médical de leurs patients.

1.8.2 Les besoins fonctionnels :

Ce sont les exigences du client spécifiant un comportement d'entrée et de sortie du système. Les besoins fonctionnels des différents acteurs peuvent être résumés comme suit :

Côté secrétaire :

- Gestion des dossiers des patients.
- Gestion des rendez-vous.

- Réalisation des statistiques de la cellule.
- Le système doit assurer l'impression des fiches malades et les bilans d'activités.

Côté Médecin :

- Consultation du planning des rendez-vous.
- Mise en disponibilité de plages horaires.
- Consultation des dossiers malades.

1.8.3 Les besoins non fonctionnels

Ce sont des besoins en relation avec la performance du système, la facilité d'utilisation, l'ergonomie des interfaces, la sécurité etc. Et parmi ses besoins nous citons :

- Accès à l'application via l'authentification.
- Simplicité et ergonomie de l'interface graphique.
- Performance du système en temps de réponse, stockage mémoire ...

1.8.4 Identification des cas d'utilisation

Chaque service offert par le système est modéliser par un cas d'utilisation qui exprime l'interaction acteurs/système.

Pour chaque acteur identifié précédemment, il convient de rechercher les différentes intentions "métier" selon lesquelles il utilise le système ce qui représente les cas d'utilisation.

Le tableau suivant illustre l'ensemble des cas d'utilisation nécessaires pour le bon fonctionnement du système :

Numéro	Cas d'utilisation	Acteurs
1	s'authentifier	Secrétaire Médecin responsable de la cellule Médecin responsable du service IRM
2	Générer Bilan	Secrétaire
3	Gérer rendez-vous	
4	Gérer dossier médical	
5	Consulter rendez-vous	Médecin responsable du service IRM
6	Consulter dossiers	Médecin responsable de la cellule
7	Gérer plage horaire	Médecin responsable du service IRM

TABLE 1.1 – Liste des cas d'utilisation

1.9 Diagrammes de cas d'utilisation

1.9.1 Cas d'utilisation "Gérer rendez-vous"

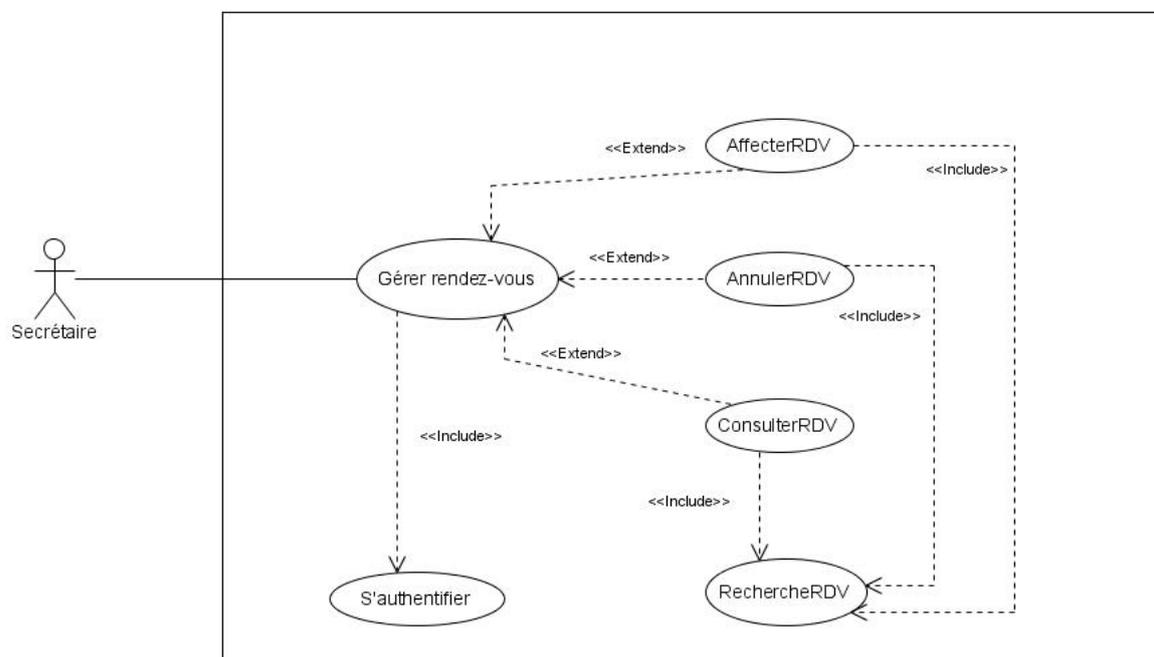


FIGURE 1.3 – Diagramme de cas d'utilisation "Gérer rendez-vous"

Ce diagramme illustre le cas d'utilisation "Gérer rendez-vous". Voir la description textuel dans la table 1.3 dans la page 20 pour plus de détail.

1.9.2 Cas d'utilisation "Générer bilan d'activités"

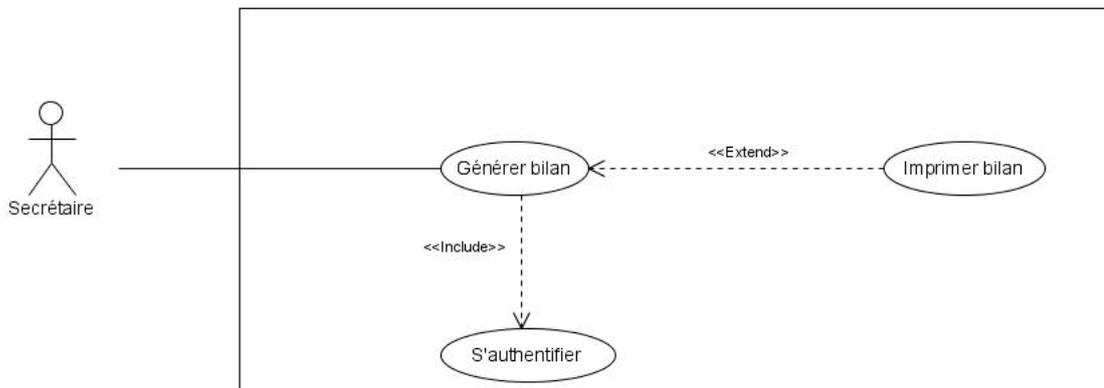


FIGURE 1.4 – Diagramme de cas d'utilisation "Générer bilan d'activités"

Ce diagramme illustre le cas d'utilisation "Générer bilan d'activités". Voir la description textuel dans la table 1.4 dans la page 21 pour plus de détail.

1.9.3 Cas d'utilisation "Gérer dossier médical"

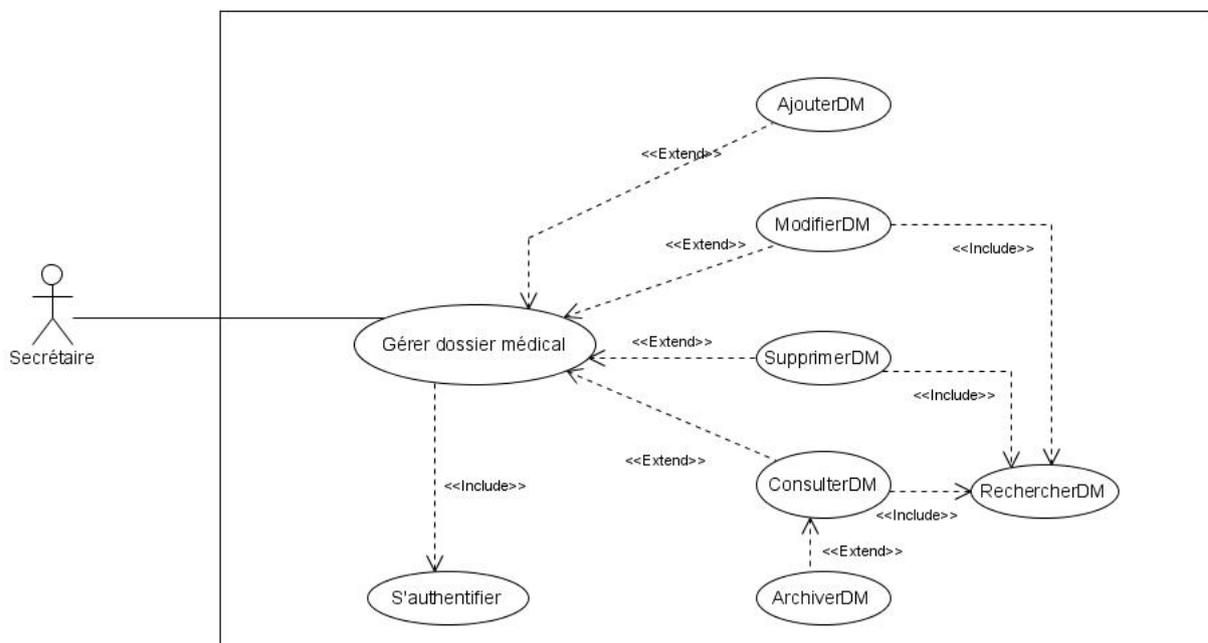


FIGURE 1.5 – Diagramme de cas d'utilisation "Gérer dossier médical"

Ce diagramme illustre le cas d'utilisation "Gérer dossier médical". Voir la description textuel dans la table 1.5 dans la page 21 pour plus de détail.

1.9.4 Cas d'utilisation "Gérer plage horaire"

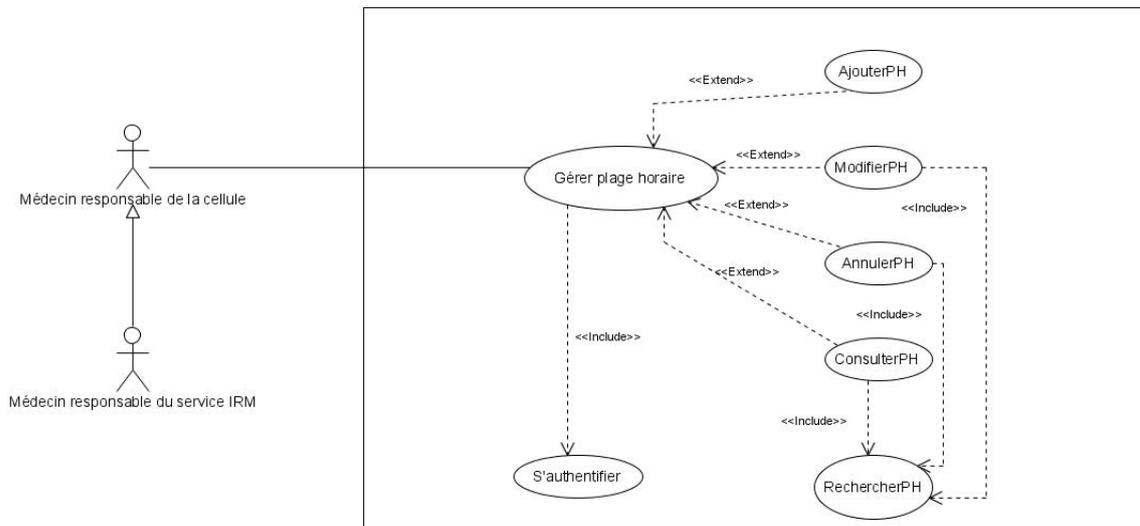


FIGURE 1.6 – Diagramme de cas d'utilisation "Gérer plage horaire"

Ce diagramme illustre le cas d'utilisation "Gérer plage horaire". Voir la description textuel dans la table 1.6 dans la page 22 pour plus de détail.

1.9.5 Cas d'utilisation "Consulter dossier médical"

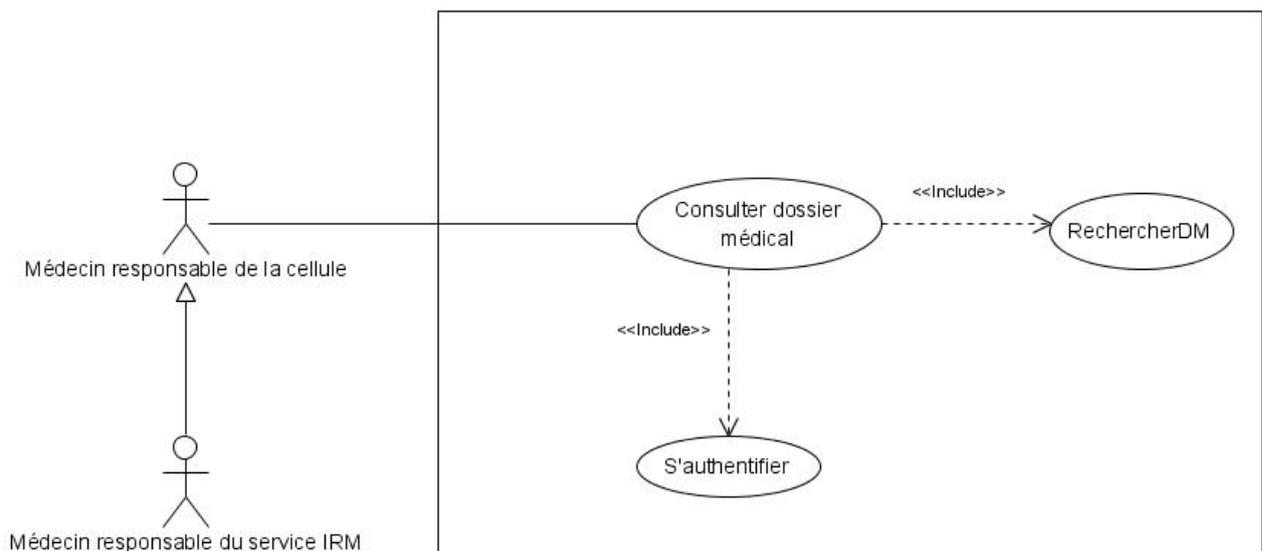


FIGURE 1.7 – Diagramme de cas d'utilisation "Consulter dossier médical"

Ce diagramme illustre le cas d'utilisation "Consulter dossiers médicaux". Voir la description textuel dans la 1.7 dans la page 22 pour plus de détail.

1.9.6 Cas d'utilisation "Consulter rendez-vous"

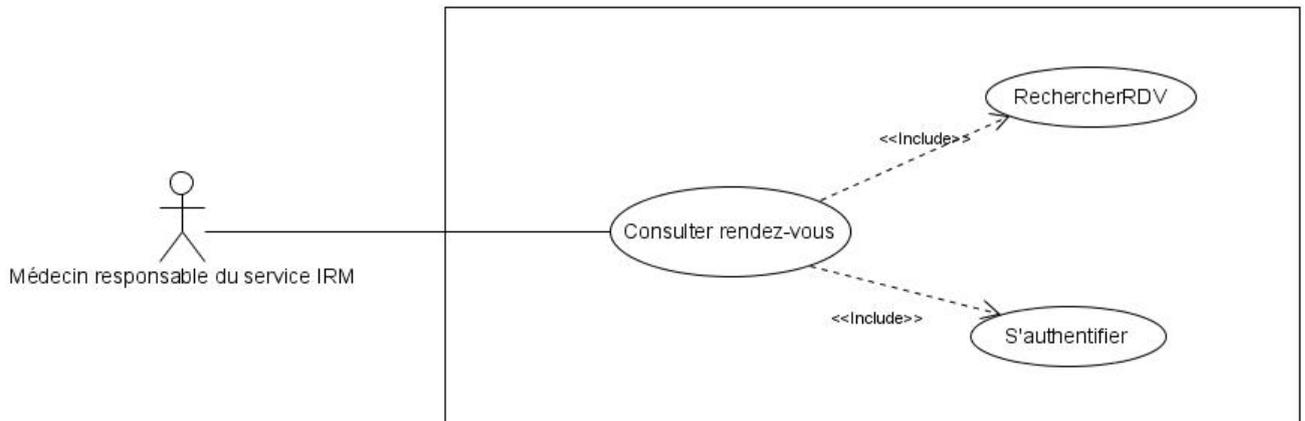


FIGURE 1.8 – Diagramme de cas d'utilisation "Consulter rendez-vous"

Ce diagramme illustre le cas d'utilisation "Consulter rendez-vous". Voir la description textuel dans la table 1.8 dans la page 23 pour plus de détail.

1.9.7 Diagrammes de cas d'utilisation "Global"

Ce diagramme résume tous les cas d'utilisation associés à tous les acteurs.

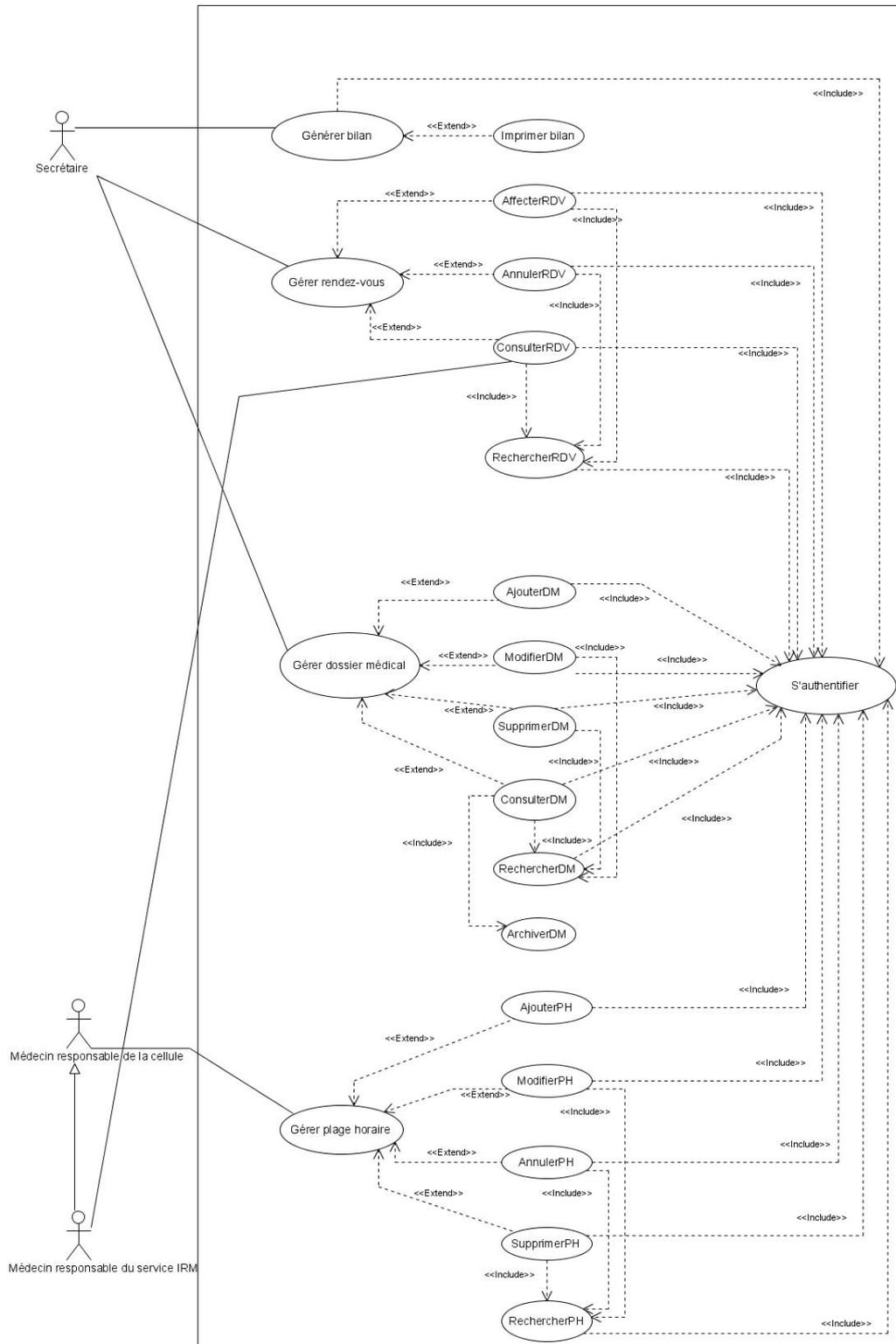


FIGURE 1.9 – Diagramme de cas d'utilisation "Global"

1.10 Description textuel des cas d'utilisation

1.10.1 Cas d'utilisation "s'authentifier"

Cas d'utilisation	s'authentifier
Acteur	Secrétaire Médecin responsable de la cellule Médecin responsable du service IRM
Objectif	Accès de l'utilisateur aux fonctions qui lui sont réservés.
Scénario nominal	1) L'utilisateur accède à l'application. 2) Le système affiche l'interface d'authentification. 3) L'utilisateur saisit son identifiant et son mot de passe. 4) Le système vérifie l'existence du compte. [A] 5) Le système donne l'accès à l'interface correspondante.
Alternatif[A]	Si les informations saisies sont fausses, le système affiche un message d'erreur et il reprend du point 2 du scénario nominal de cas d'utilisation s'authentifier.

TABLE 1.2 – Cas d'utilisation "s'authentifier"

1.10.2 Cas d'utilisation "Gérer Rendez-vous"

Cas d'utilisation	Gérer rendez-vous.
Acteur	Secrétaire
Objectif	Mise à jour de la liste des rendez-vous.
Pré-condition	S'authentifier.
Scénario nominal	1) L'utilisateur demande l'interface de gestion de rendez-vous. 2) Le système affiche l'interface. 3) L'utilisateur effectue la mise à jour et valide. [A] 4) Le système enregistre les données.
Alternatif[A]	Si un champ d'information est incomplet ou le rendez-vous est déjà pris, le système affiche un message d'erreur et il reprend au point 2 du scénario nominal.

TABLE 1.3 – Cas d'utilisation "Gérer Rendez-vous"

1.10.3 Cas d'utilisation "Générer Bilan d'activités"

Cas d'utilisation	Générer Bilan d'activités.
Acteur	Secrétaire
Objectif	Générer automatiquement les bilans d'activités.
Pré-condition	S'authentifier.
Scénario nominal	1) L'utilisateur demande l'interface des bilans d'activités. 2) Le système affiche l'interface. 3) L'utilisateur remplit les champs puis valide. [A] 4) Le système génère le bilan.
Alternatif[A]	Si un champ d'information est incomplet, le système affiche un message d'erreur et reprend au point 2 du scénario nominal

TABLE 1.4 – Cas d'utilisation "Générer Bilan d'activités"

1.10.4 Cas d'utilisation "Gérer dossier médical"

Cas d'utilisation	Gérer dossier médical.
Acteur	Secrétaire
Objectif	Mise à jour de dossier médical.
Pré-condition	S'authentifier.
Scénario nominal	1) L'utilisateur demande l'interface de gestion de dossiers médicaux. 2) Le système affiche l'interface. 3) L'utilisateur effectue la mise à jour et valide. [A] 4) Le système enregistre les données. 5) L'utilisateur peut archiver le dossier.
Alternatif[A]	Si un champ d'information est incomplet ou le dossier existe déjà, le système affiche un message d'erreur et il reprend au point 2 du scénario nominal.

TABLE 1.5 – Cas d'utilisation "Gérer dossier médical"

1.10.5 Cas d'utilisation "Gérer Plage Horaire"

Cas d'utilisation	Plage Horaire.
Acteur	Médecin responsable de la cellule d'accueil, Médecin responsable du service IRM
Objectif	Mise à jour des Plage Horaire.
Pré-condition	S'authentifier.
Scénario nominal	1)L'utilisateur demande l'interface de gestion de Plage Horaire. 2)Le système affiche l'interface. 3)L'utilisateur effectue la mise à jour et valide. [A] 4)Le système enregistre les données.
Alternatif[A]	Si un champ d'information est incomplet ou la Plage Horaire existe déjà, le système affiche un message d'erreur et il reprend au point 2 du scénario nominal.

TABLE 1.6 – Cas d'utilisation "Gérer Plage Horaire"

1.10.6 Cas d'utilisation "Consulter Dossier"

Cas d'utilisation	Consulter Dossier.
Acteur	Médecin responsable de la cellule d'accueil, Médecin responsable du service IRM
Objectif	Consulter les dossier des patients.
Pré-condition	S'authentifier.
Scénario nominal	1)L'utilisateur demande l'interface des dossiers. 2)Le système affiche l'interface. 3)L'utilisateur effectue une recherche et valide. [A] 4)Le système affiche le dossier.
Alternatif[A]	Si un champ d'information est incomplet ou le dossier n'existe pas, le système affiche un message d'erreur et il reprend au point 2 du scénario nominal.

TABLE 1.7 – Cas d'utilisation "Consulter Dossier"

1.10.7 Cas d'utilisation "Consulter Rendez-vous"

Cas d'utilisation	Consulter Rendez-vous.
Acteur	Médecin responsable du service IRM
Objectif	Consulter les Rendez-vous.
Pré-condition	S'authentifier.
Scénario nominal	1) L'utilisateur demande l'interface des rendez-vous. 2) Le système affiche l'interface. 3) L'utilisateur effectue une recherche et valide. [A] 4) Le système affiche le rendez-vous.
Alternatif[A]	Si un champ d'information est incomplet ou le rendez-vous n'existe pas, le système affiche un message et il reprend au point 2 du scénario nominal.

TABLE 1.8 – Cas d'utilisation "Consulter Rendez-vous"

1.11 Environnement de développement

1.11.1 Netbeans



C'est un environnement de développement intégré (EDI), il permet de supporter différents autres langages, comme C, C++, JavaScript, XML, etc. Un environnement Java Development Kit JDK est requis pour les développements en Java. NetBeans constitue par ailleurs une plateforme qui permet le développement d'applications spécifiques (bibliothèque Swing (Java)). L'IDE NetBeans s'appuie sur cette plateforme.[12]

1.11.2 Tomcat



C'est un serveur d'applications web conçu pour faire fonctionner et gérer des applications client-serveur.[10]

1.11.3 PhpMyAdmin



PhpMyAdmin est un logiciel libre écrit en PHP, destiné à gérer l'administration de MySQL sur le Web, il supporte une large gamme d'opérations sur MySQL, tel que la gestion des bases de données, des tables, des colonnes, des relations, des index, des utilisateurs, des permissions, etc. Ses opérations peuvent être effectués via l'interface utilisateur fréquemment utilisé. Il offre aussi la possibilité d'exécuter directement toute instruction SQL.[11]

1.11.4 GlassFish



GlassFish est le nom du serveur d'applications Open Source Java EE 5 et désormais Java EE 7 avec la version 4.1 qui sert de base au produit Oracle GlassFish Server(anciennement Sun Java System Application Server de Sun Microsystems). Sa partie Toplink persistence provient d'Oracle. C'est la réponse aux développeurs Java désireux d'accéder aux sources et de contribuer au développement des serveurs d'applications de nouvelle génération.[10]

1.11.5 SoapUI



SoapUI

SoapUI est une application open source permettant le test de web service dans une architecture orientée services (SOA). Ses fonctionnalités incluent l'inspection des web service, l'invocation, le développement, la simulation, les tests fonctionnels, les tests de charge et de conformité.[12]

1.11.6 Visual Paradigme



C'est un éditeur qui propose une suite logicielle, constituée de plusieurs outils dont on a utilisé :

Visual Paradigme For UML : permet la création des diagrammes UML et des modèles qui en sont à l'origine. Ceux-ci peuvent alors générer du code dans un langage de programmation déterminé. Il propose également la création d'autres types de diagrammes, comme celui qui permet la modélisation des bases de données pouvant, lui aussi, générer des canevas d'applications basés sur des Framework et Pattern mais en plus, générer du code SQL qu'il peut ensuite déployer automatiquement dans différents environnements.[13]

1.12 Langages de développement

1.12.1 JAVA



Java est un langage de programmation orienté objets, qui permet d'exécuter des programmes à travers une machine virtuelle. Il fournit un ensemble d'outils le JDK, un ensemble de packages et un ensemble de classes. Ses différentes classes couvrent beaucoup de domaines, les entrées /sorties, les interfaces graphiques et les réseaux.[9]

1.12.2 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. C'est l'un des projets les plus populaires sur la plate-forme de gestion de développement GitHub.[10]

1.12.3 JavaServer Page (JSP)

 JavaServer Page est une technique basée sur Java qui permet aux développeurs de créer dynamiquement du code HTML, XML ou tout autre type de page web. Cette technique permet au code Java et à certaines actions prédéfinies d'être ajoutés dans un contenu statique. Depuis la version 2.0 les spécifications et la syntaxe JSP sont complètement conforme au standard XML.

Les JSP sont compilées par un compilateur JSP pour devenir des servlets Java. Un compilateur JSP peut créer une servlet Java en code source Java qui peut à son tour être compilé par le compilateur Java, ou peut créer le pseudo-code Java interprétable directement.[10]

1.12.4 Mysql

 C'est un système de gestion de bases de données relationnelles (SGBDR). Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle, Informix et Microsoft SQL Server.[10]

1.12.5 PHP

 C'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 quelles langage interprétées de façon locale. PHP est un langage impératif orienté objet. [10]

1.13 Conclusion

Dans ce chapitre nous avons présenté l'historique et l'organigramme de l'hôpital KHEL-LIL Amrane. Nous sommes passés, ensuite, à la présentation de notre organisme d'accueil à savoir « la cellule d'accueil et d'orientation des cancéreux », nous avons étudié le système qui est utilisé actuellement pour la gestion de ce service et nous avons détecté et défini plusieurs problèmes auxquels nous proposons une solution commune. Notre solution consiste à développer une application qui fait appel à des services web, qui utilise une base de données pour stocker les dossiers des malades et gérer les rendez-vous des patients, ainsi nous avons pu ressortir les différents besoins fonctionnels et non fonctionnels, ainsi que les outils utilisés pour la conception et la réalisation de cette solution.

Pour pallier aux problèmes rencontrés et répondre aux besoins cités au préalable, nous allons aborder la notion des services web dans le chapitre qui suit.

2

Les services Web

Introduction

Les services web fournissent une nouvelle manière de développer des applications conformes aux besoins de l'internet, c'est la solution la plus adaptée pour assurer l'interopérabilité permettant de transmettre les données entre les différentes applications d'une organisation (entreprise, société ou individu). Pour cela, Nous proposons de développer une application qui fait appel à des services web pour la gestion de la cellule. Cette solution permettra ainsi de réaliser le traitement de ces données, de gérer les liaisons entre les différentes applications.

Ce chapitre est dédié à la notion de services web. Nous allons donner une présentation générale des services en passant par la définition, l'architecture de ces services et les technologies standards permettant leurs implémentations et enfin les avantages de l'utilisation des services web.

2.1 Définition

- Les services web sont des applications auto descriptives, modulaire et faiblement couplées. Ils fournissent un modèle simple de programmation et de déploiement d'applications, s'exécutant au travers de l'infrastructure web. Les services web réalisent des fonctions allant de simples requêtes aux processus métier sophistiqués. [4]

- Un service web est un système logiciel identifié par un URI (Universal Resource Identifier) dont les interfaces publiques et les incarnations sont définies et décrites en XML. Sa définition peut être découverte dynamiquement par d'autres systèmes logiciels. Ces autres systèmes peuvent interagir avec le service web d'une façon décrite par sa définition, en utilisant des messages XML transporter par des Protocoles Internet. [5]

2.2 Architecture des services web

Les services web reprennent la plupart des idées et des principes du web (HTTP, XML), et les appliquent à des interactions entre machines. Ils communiquent via un ensemble de technologies fondamentales qui partagent une architecture commune. Ils ont été conçus pour être réalisés sur de nombreux systèmes développés et déployés de façon indépendante.

L'interopérabilité est l'objectif premier des services web. Pour permettre cet échange d'information entre des applications distantes, les services web sont composés des couches

standards. Deux types d'architecture existent pour les services web : une de référence et l'autre étendue.[7]

2.2.1 Architecture de référence

L'architecture de référence (voir Figure 2.1) des services web se base sur l'interaction entre les trois rôles suivants : le fournisseur de service, l'annuaire de service et le client.

- **Le fournisseur de service** : correspond au propriétaire du service. D'un point de vue technique, il est constitué par la plate-forme d'accueil du service.

- **Le client** : correspond au demandeur de service. D'un point de vue technique, il est constitué par l'application qui va rechercher et invoquer un service. L'application cliente peut être elle-même un service web.

- **L'annuaire de service** : correspond à un registre de descriptions de services offrant des facilités de publication de services à l'intention des fournisseurs ainsi que des facilités de recherche de service à l'intention des clients. [6]

L'architecture de référence comprend les opérations suivantes :

- **La publication** : le fournisseur de service crée le service et définit sa description puis la publie dans un annuaire de service en vue d'être localisé par des clients.
- **La recherche** : le client effectue des recherches dans l'annuaire pour retrouver et sélectionner un service qui répond à ses besoins.
- **Découverte et invocation** : le client utilise la description du service sélectionné pour récupérer les informations nécessaires lui permettant de se connecter au fournisseur du service et d'interagir avec lui.

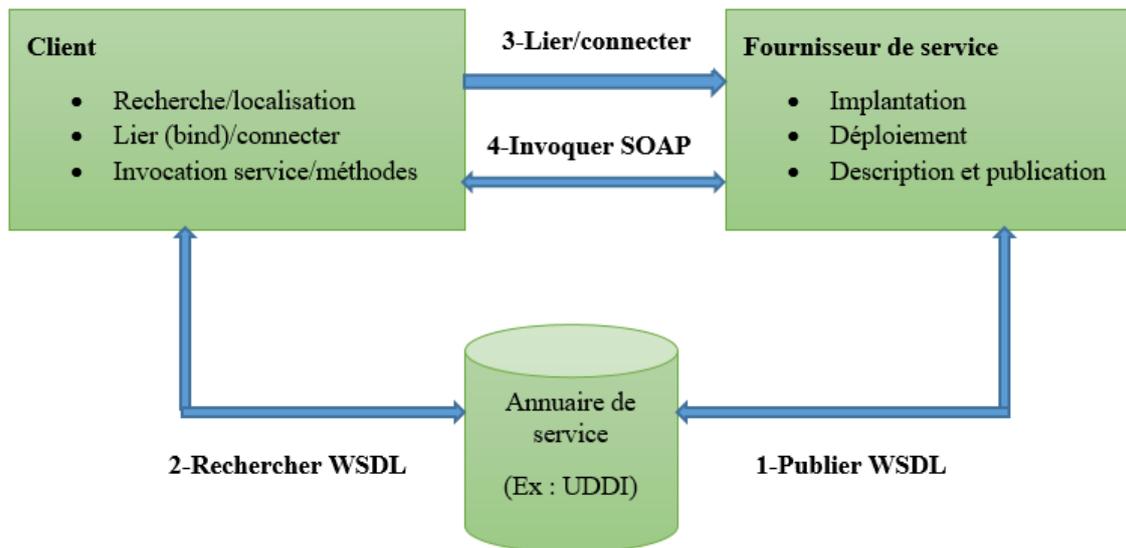


FIGURE 2.1 – Architecture de référence de service web

2.2.2 Architecture étendue

Plusieurs technologies ont été proposées par le groupe W3C pour l'extension de l'architecture de référence et l'élaboration d'une architecture étendue standard.

L'architecture étendue (Voir Figure 2.2) est constituée de plusieurs couches se superposant les unes sur les autres, d'où le nom de la pile service web. [7]

L'architecture étendue peut être décomposée en trois types de couches :

1. **L'infrastructure de base** : elle est constituée de trois couches, ces couches s'appuient sur les standards émergents : SOAP ou Rest pour l'échange de messages, WSDL pour la description de services web, et UDDI pour la publication. Elle définit donc le fondement technique de l'architecture de référence.
2. **La couche Business Processus** : elle permet l'intégration et l'utilisation effective des services web dans le domaine de e-business.
3. **La couche transversale** : (sécurité, transaction, Administration et QoS¹) ces couches

1. 1Quality of service, Ensemble de paramètre échangé pendant une communication avec connexion pour que les informations passent correctement.

rendent viable l'utilisation effective dans le domaine industriel.

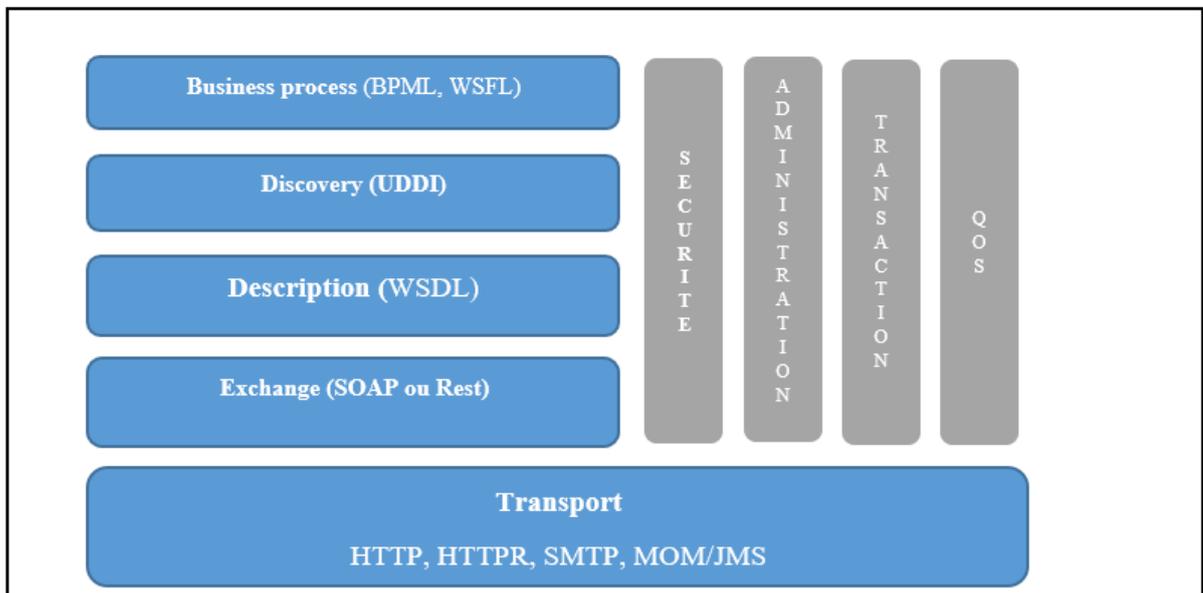


FIGURE 2.2 – Architecture en pile

2.3 Technologies de bases

HTTP est le protocole de base pour le transport des messages entre services web. Il existe une multitude de standards qui sont organisés en quatre catégories : les spécifications XML de bases, les spécifications orientées messages, les spécifications orientées documents et les vocabulaires de communauté.

2.3.1 XML

Le langage XML est un langage balisé, issu d'une recommandation W3C, ayant pour but d'encoder tout type de données, indépendamment du code machine de celle-ci. Il a été développé dans le but de partager facilement des données entre différents systèmes et en particulier à travers un réseau type internet. [8] Aujourd'hui, XML est un standard, il constitue la technologie de base des architectures services web. De plus, grâce à la structuration, XML permet la distinction entre les données des applications et les données des protocoles permettant ainsi une correspondance facile entre les différents protocoles.

2.3.2 La communication SOAP

SOAP (Simple Access Protocol) est un standard du consortium W3C définissant un protocole qui assure des appels de procédures à distance (RPC) s'appuyant principalement sur le protocole HTTP et sur XML, mais aussi sur SMTP et POP. Il assure l'interaction entre services web en transportant les paquets de données encapsulés sous forme de texte structuré au format XML au sein d'architecture orientées objet. [4] Le protocole SOAP est composé de trois parties :

- Une enveloppe : définit un canevas pour décrire le contenu du message et comment le traiter.
- L'entête SOAP : elle contient des données supplémentaires transmises aux services extérieurs, ces données peuvent être des informations d'authentications, de gestion de transaction, de paiement, etc.
- Le corps du message SOAP : permet de transmettre les requêtes et les réponses entre les systèmes.

2.3.3 Le langage WSDL

Un document WSDL, est constitué d'une suite d'éléments décrivant un service web sous forme d'un ensemble d'opérations exploitable depuis l'extérieur.

Il est composé de deux niveaux : un niveau abstrait réutilisable constitué d'éléments qui définissent l'interface des services web tel que les types de données, les messages, les types de ports et un autre niveau non réutilisable appelé concret constitué des éléments qui décrivent des informations liées à un contexte du service web.[5]

2.3.4 UDDI

UDDI (Universal Description Discovery and Integration) est un standard, né de l'initiative d'un certain nombre d'entreprise, dont notamment Microsoft, IBM, Intel et bien d'autres qui se sont réunies pour développer une spécification basée sur des technologies standard afin de faciliter la collaboration entre partenaires dans le cadre des échanges commerciaux.

L'annuaire UDDI permet de publier et d'invoquer des informations sur une entreprise et ses services web. Il contient des informations sur les entreprises et les services qu'elle publie.[4]

2.4 Les problèmes liés au service web

- Les normes de services web dans certains domaines sont actuellement récentes.
- Par l'utilisation du protocole HTTP, les services web peuvent contourner les mesures de sécurité mises en place au travers de par-feu.
- Les services web ne sont pas encore très répandus; il est donc difficile de trouver l'objet de sa recherche.

2.5 Les avantages des services web

- Les services web fournissent l'interopérabilité entre divers logiciels fonctionnant sur diverses plates-formes.
- Les services web utilisent des standards et protocoles ouverts.
- Les protocoles et les formats de données sont au format texte dans la mesure du possible, facilitant ainsi la compréhension du fonctionnement global des échanges.
- Les outils de développement, s'appuyant sur ces standards, permettent la création automatique de programmes utilisant les services web existants.
- Les coûts sont réduits par l'automatisation interne et externe des processus commerciaux.[15]

2.6 Conclusion

Dans ce chapitre, nous avons présenté les différentes architectures des services web, ainsi les technologies de base et en final nous avons cités, les problèmes liés aux services web et les avantages de ce dernier.

3

Conception

Introduction

Dans ce chapitre, nous étendrons la représentation des diagrammes de cas d'utilisation effectués au niveau de l'analyse des besoins. Nous commencerons par établir les diagrammes de séquences des cas d'utilisation puis, l'élaboration du diagrammes de classes et on termine par le modèle logique de données.

3.1 Diagrammes de séquence des cas d'utilisation

3.1.1 Diagramme de séquence du cas d'utilisation "s'authentifier"

Un utilisateur doit s'authentifier en saisissant ses propres coordonnées (identifiant et mot de passe), puis le système procède à la vérification des informations introduites pour les comparer avec les données stockées dans la base de données, si l'une des coordonnées est incomplète, login ou mot de passe est incorrect le système affiche un message d'erreur sinon l'accès est autorisé.

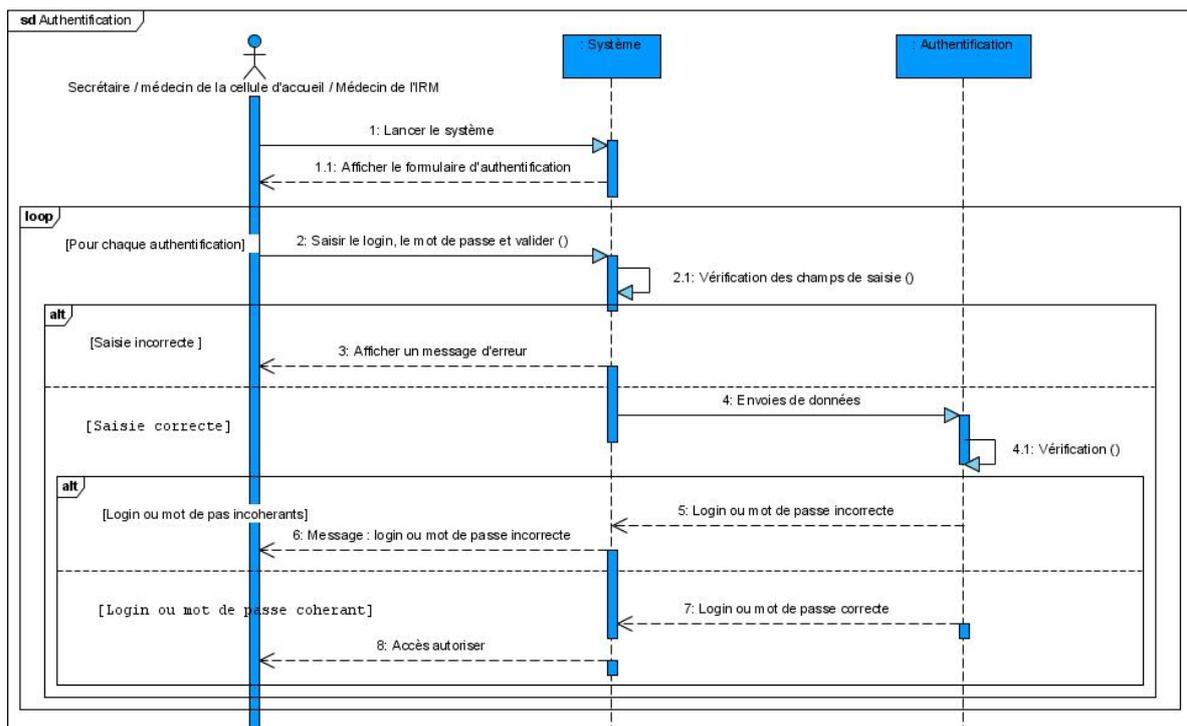


FIGURE 3.1 – Diagramme de séquence du cas d'utilisation "authentification"

3.1.2 Diagramme de séquence du cas d'utilisation "Gérer Rendez-vous"

— Affecter un Rendez-vous

Pour affecter un rendez-vous la secrétaire doit d'abord s'authentifier, choisir le service de rendez-vous puis elle remplit le formulaire. En affectant un rendez-vous, le système doit vérifier la saisie ainsi que la disponibilité du créneau.

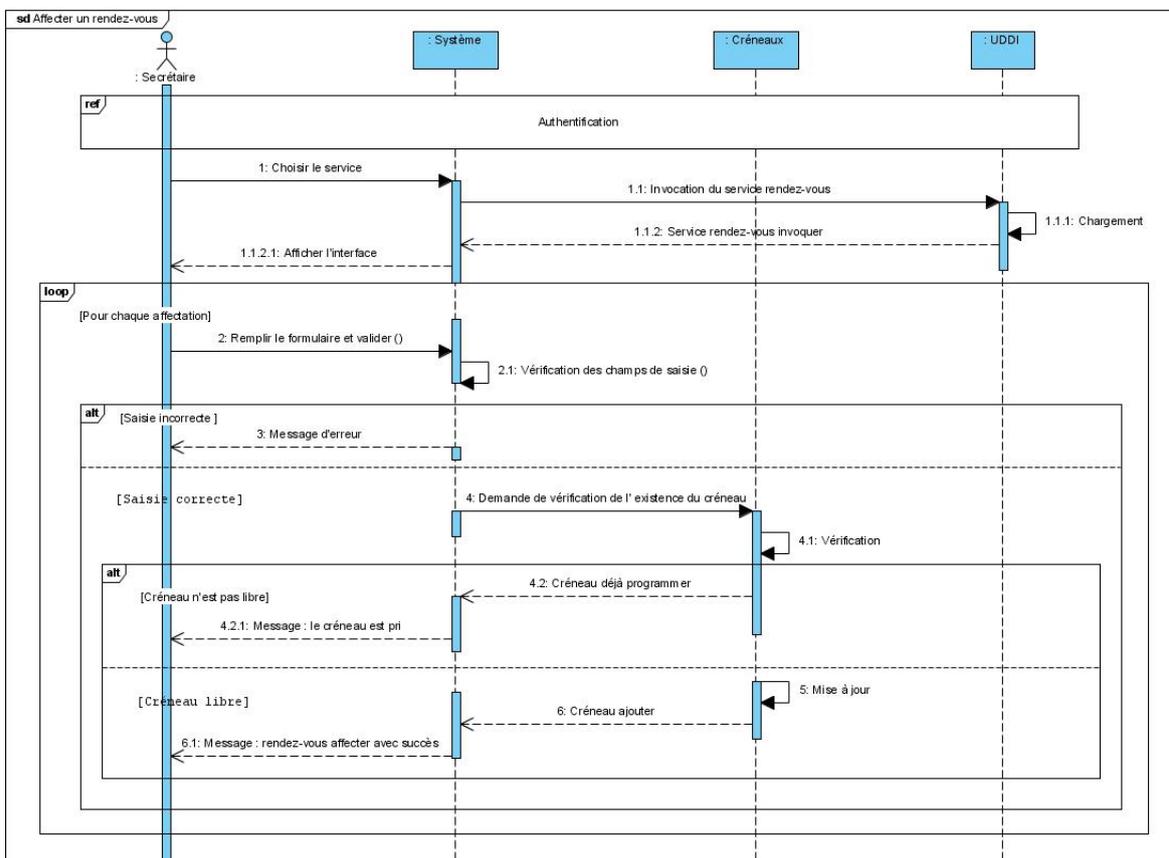


FIGURE 3.2 – Diagramme de séquence du cas d'utilisation "Affecter un Rendez-vous"

— Rechercher Rendez-vous

Pour rechercher un rendez-vous la secrétaire doit d’abord s’authentifier, choisir le service de rendez-vous puis le système affiche l’espace de recherche, la secrétaire introduit la date et elle valide. Le système doit vérifier la saisie ainsi que l’existence du rendez-vous.

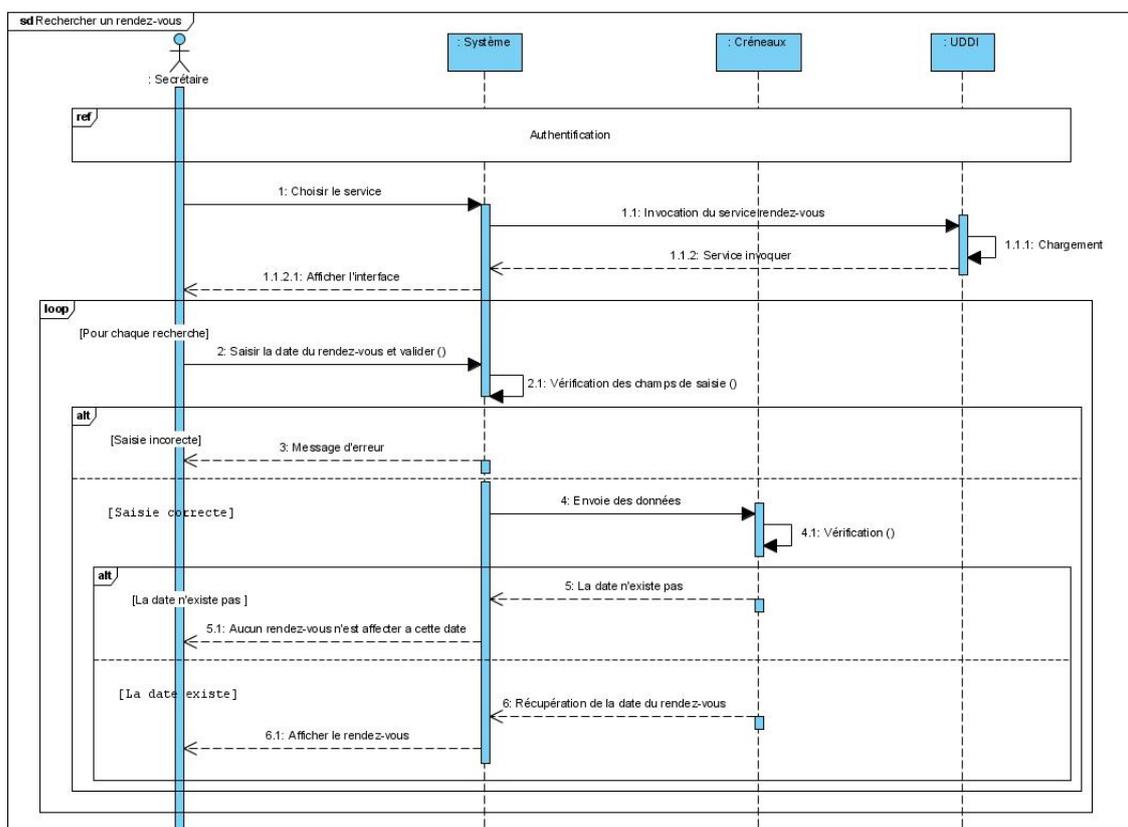


FIGURE 3.3 – Diagramme de séquence du cas d’utilisation ”Rechercher Rendez-vous”

— Annuler Rendez-vous

Après l'authentification la secrétaire effectue une recherche puis elle sélectionne le rendez-vous a annuler et elle valide. Une boite de dialogue sera afficher et la secrétaire a le choix entre confirmer ou ne pas confirmer l'annulation, après la confirmation le système effectue une mise à jour.

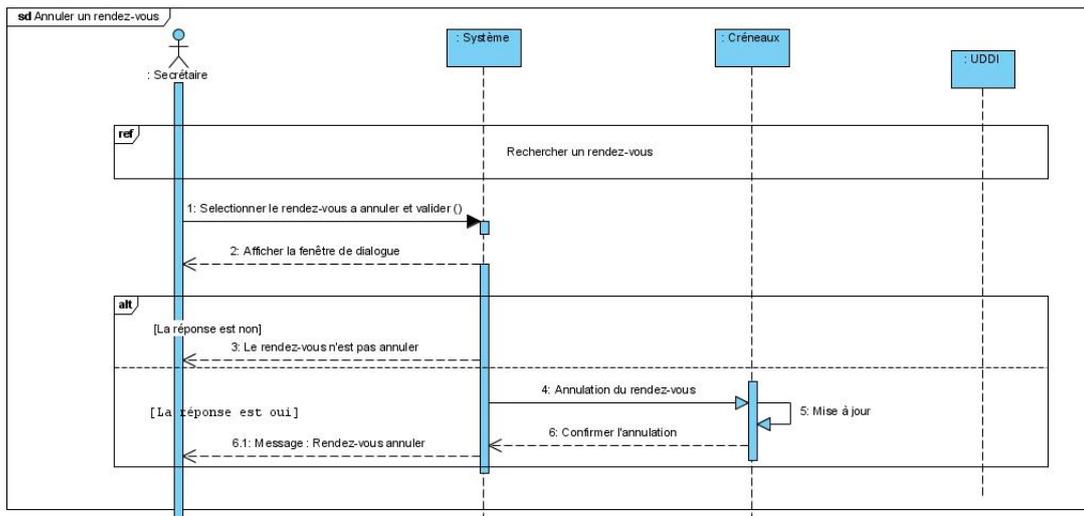


FIGURE 3.4 – Diagramme de séquence du cas d'utilisation "Annuler Rendez-vous"

— Consulter Rendez-vous

Le médecin responsable du service IRM et la secrétaire doivent d'abord s'authentifier et effectuer une recherche pour consulter la liste des rendez-vous.

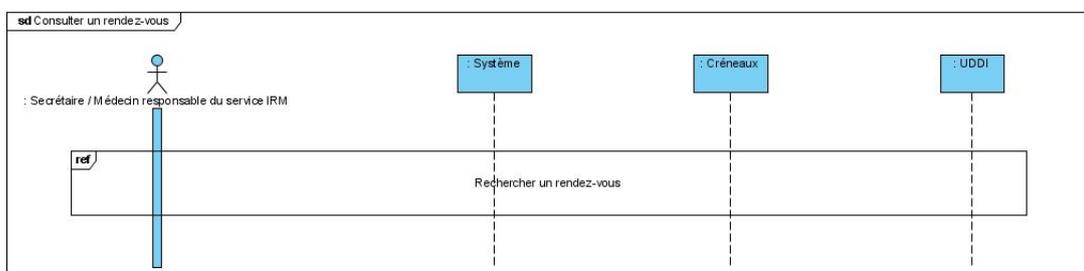


FIGURE 3.5 – Diagramme de séquence du cas d'utilisation "Consulter Rendez-vous"

3.1.3 Diagramme de séquence du cas d'utilisation "Gérer Dossier médical"

— Ajouter Dossier

Après l'authentification la secrétaire choisi le service, puis elle remplit le formulaire. En ajoutant un dossier le système doit vérifier la saisie ainsi que l'existence du dossier.

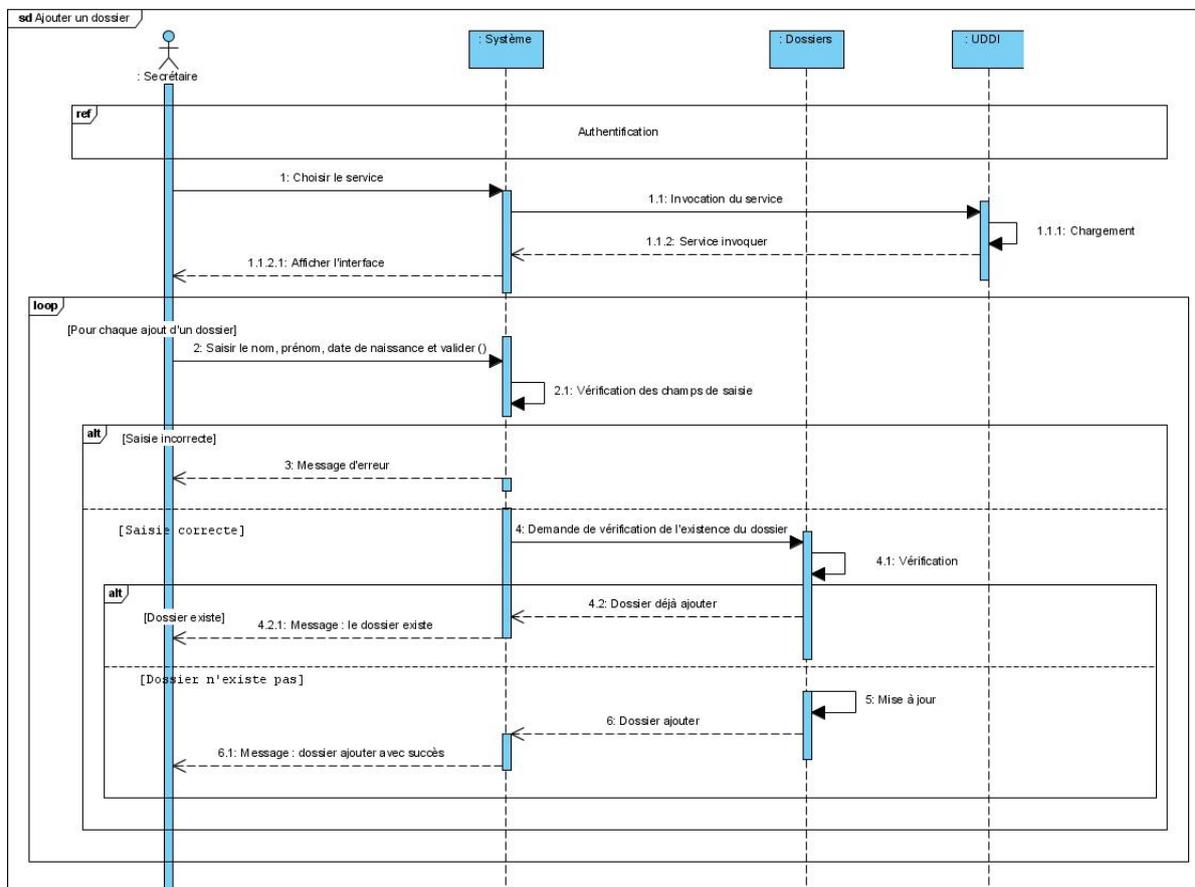


FIGURE 3.6 – Diagramme de séquence du cas d'utilisation "Ajouter dossier"

— Rechercher Dossier

Afin d'accéder aux dossiers la secrétaire doit effectuer une recherche et pour cela elle doit d'abord s'authentifier. elle choisit le service des dossiers puis le système affiche l'espace de recherche, la secrétaire introduit le nom et prénom du patient et elle valide. Le système doit vérifier la saisie ainsi que l'existence du dossier.

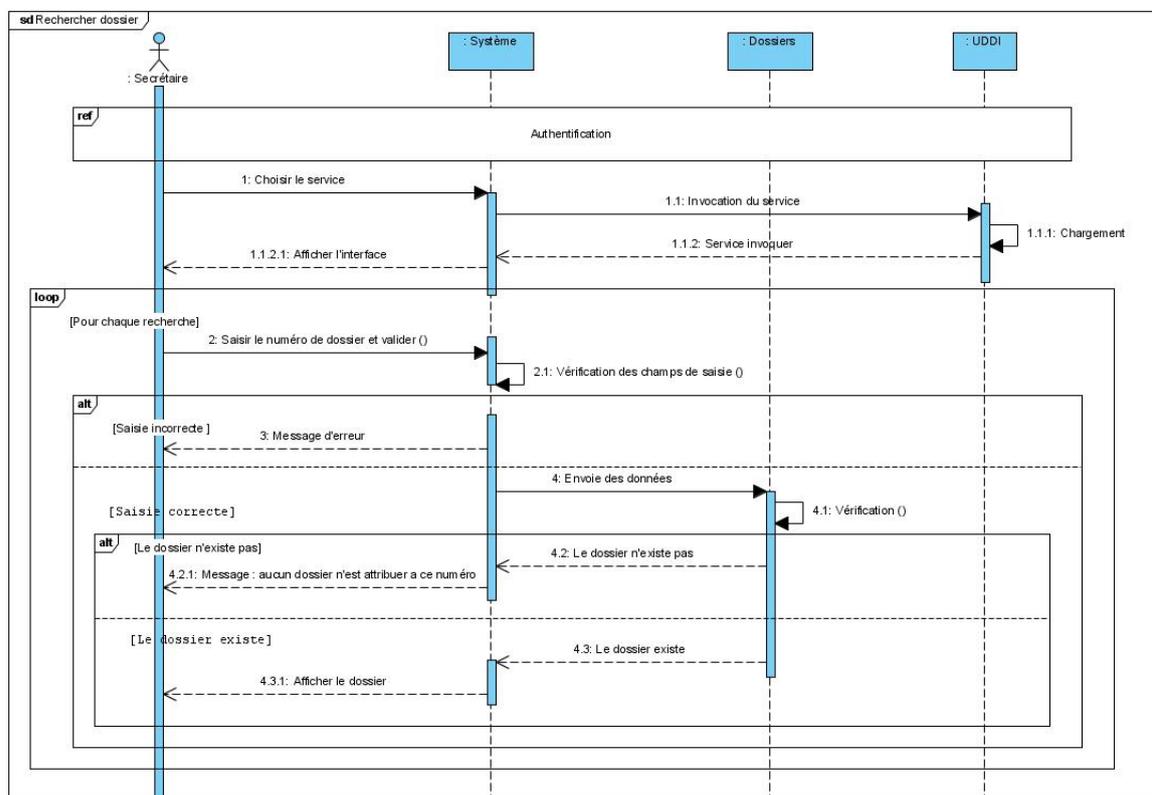


FIGURE 3.7 – Diagramme de séquence du cas d'utilisation "Rechercher dossier"

— **Supprimer Dossier**

pour supprimer un dossier la secrétaire doit d'abord s'authentifier ensuite elle effectue une recherche puis elle sélectionne le dossier à supprimer et elle valide. Une boîte de dialogue sera afficher et la secrétaire a le choix entre confirmer ou ne pas confirmer la suppression, après la confirmation le système effectue une mise à jour.

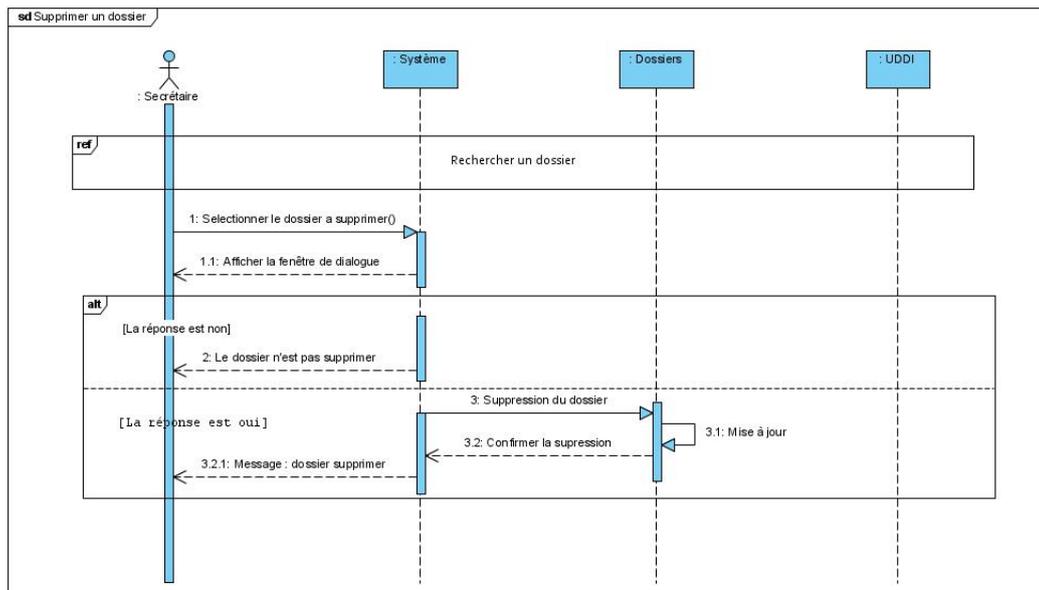


FIGURE 3.8 – Diagramme de séquence du cas d'utilisation "Supprimer dossier"

— **Consulter Dossier**

Le médecin responsable de la cellule d'accueil, Le médecin responsable du service IRM et la secrétaire doivent d'abord s'authentifier et effectuer une recherche pour consulter un dossier.

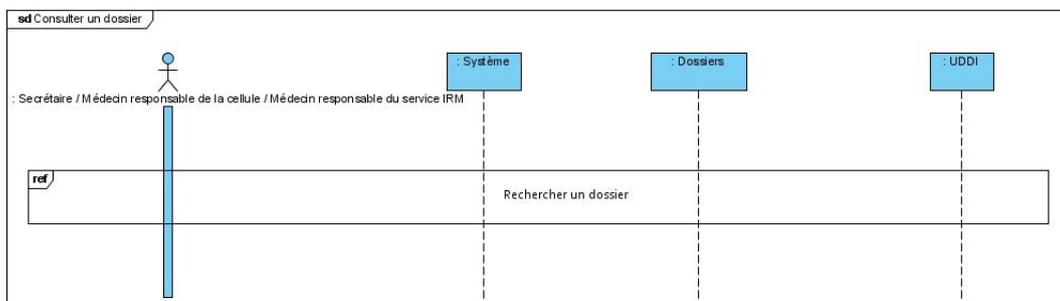


FIGURE 3.9 – Diagramme de séquence du cas d'utilisation "Consulter dossier"

— Modifier Dossier

La modification d'un dossier médical se fait par la secrétaire qui après l'authentification recherche le dossier à modifier puis elle modifie le dossier. Le système vérifie les champs de saisie et effectue une mise à jour.

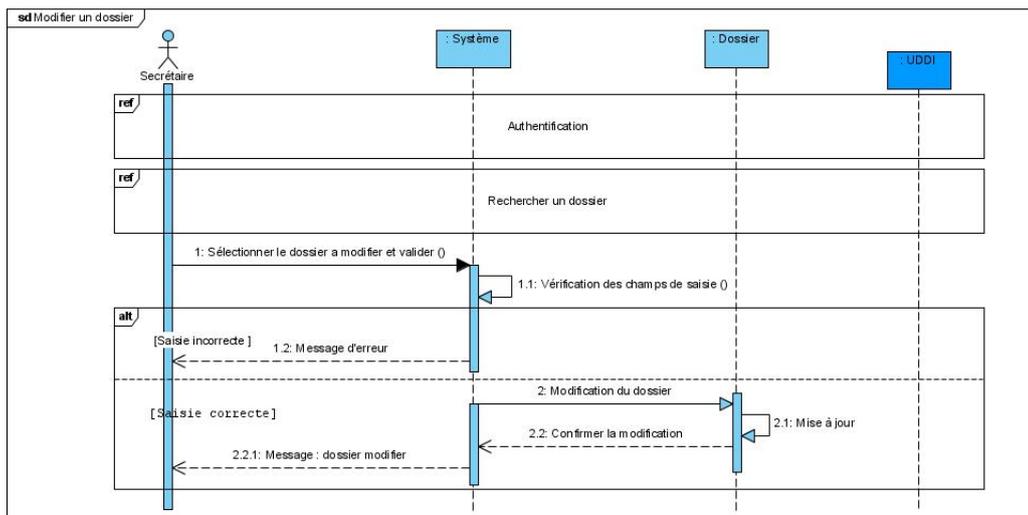


FIGURE 3.10 – Diagramme de séquence du cas d'utilisation "Modifier dossier"

3.1.4 Diagramme de séquence du cas d'utilisation "Gérer Plage Horaire"

— Ajouter Plage Horaire

Après l'authentification le médecin responsable de la cellule et le médecin responsable du service IRM choisissent le service plage horaire puis ils remplissent le formulaire. En ajoutant une plage horaire le système doit vérifier la saisie ainsi que l'existence de la plage horaire.

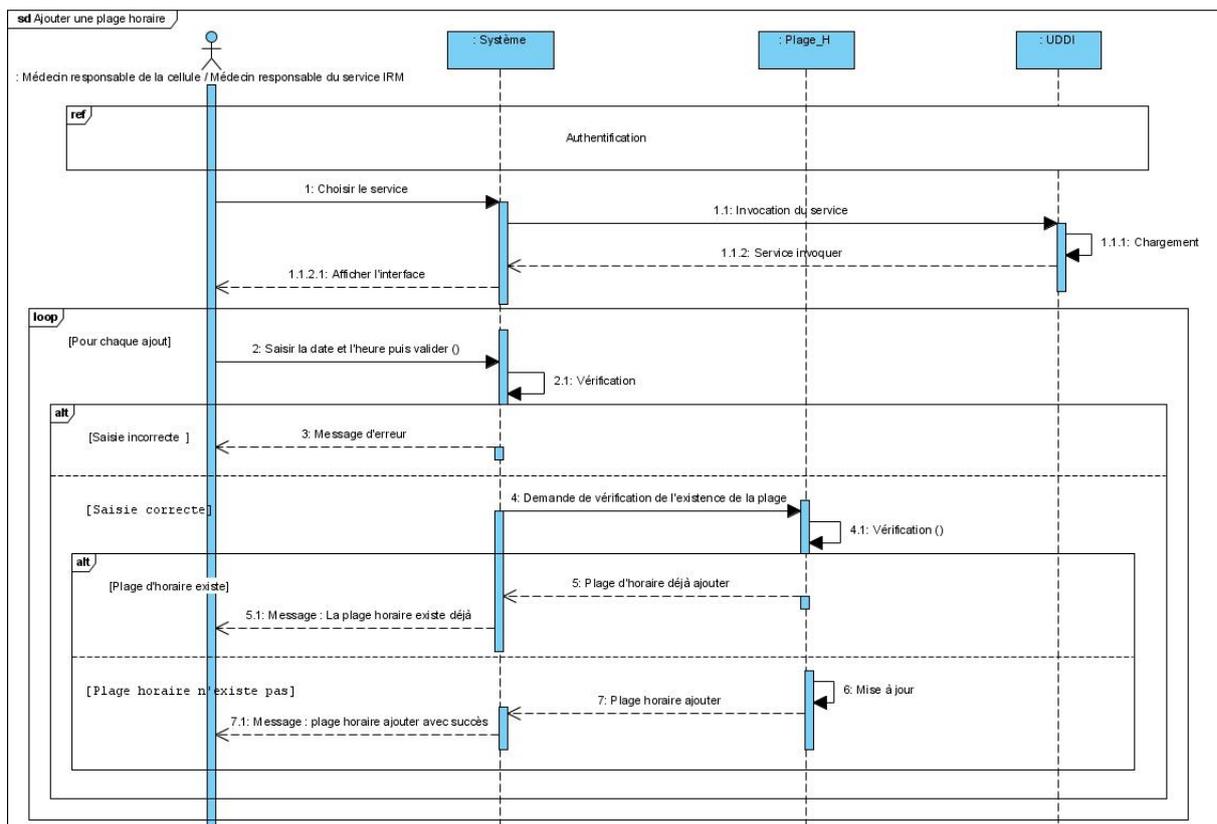


FIGURE 3.11 – Diagramme de séquence du cas d'utilisation "Ajouter Plage horaire"

— Rechercher une Plage Horaire

Pour rechercher une plage horaire le médecin responsable de la cellule ou le médecin responsable du service IRM doivent d’abord s’authentifier, choisir le service plage horaire puis le système affiche l’espace de recherche. Ses dernier introduisent la date et ils valide, le système doit vérifier la saisie ainsi que l’existence de la plage horaire.

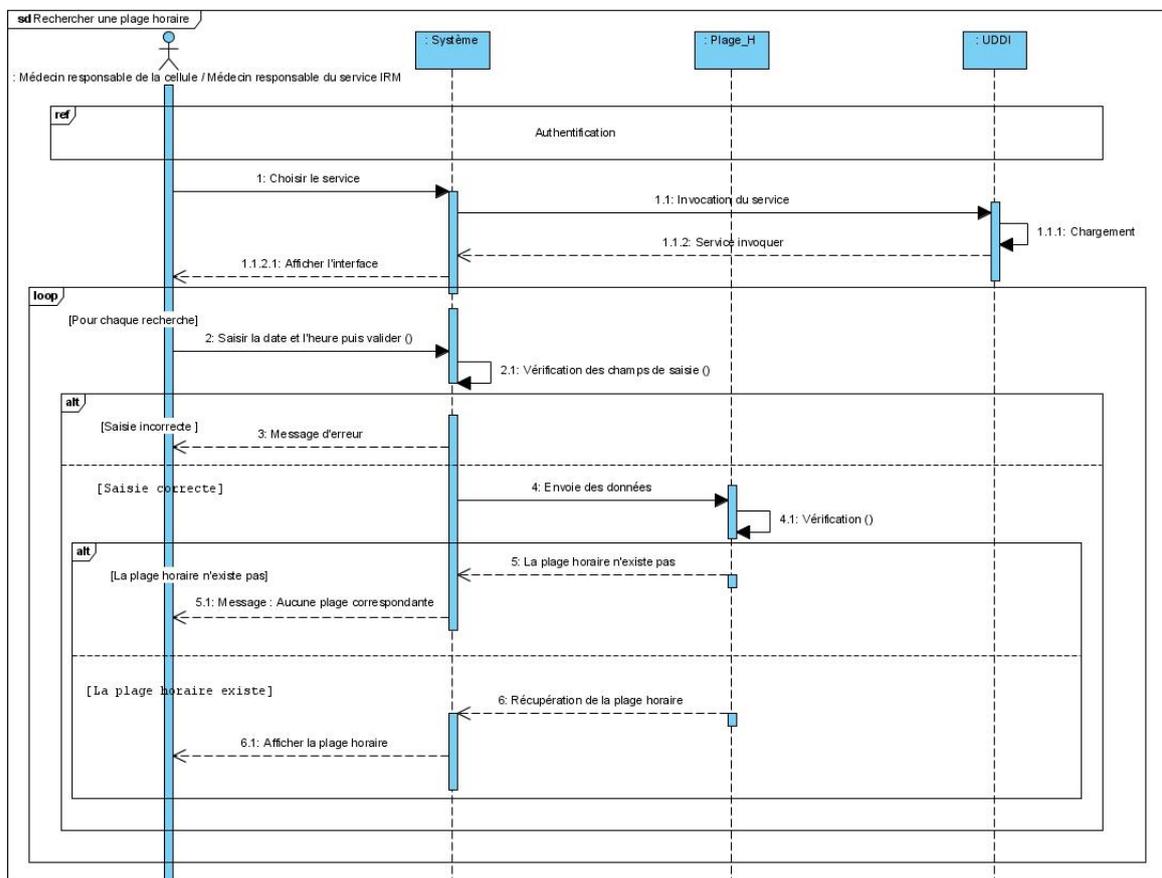


FIGURE 3.12 – Diagramme de séquence du cas d’utilisation ”Rechercher une Plage Horaire”

— Annuler une Plage Horaire

Après l'authentification le médecin responsable de la cellule ou le médecin responsable du service IRM effectuent une recherche puis il sélectionne la plage horaire a annuler et il valide. Une boite de dialogue sera afficher et les deux médecin ont le choix entre confirmer ou ne pas confirmer l'annulation, après la confirmation le système effectue une mise à jour.

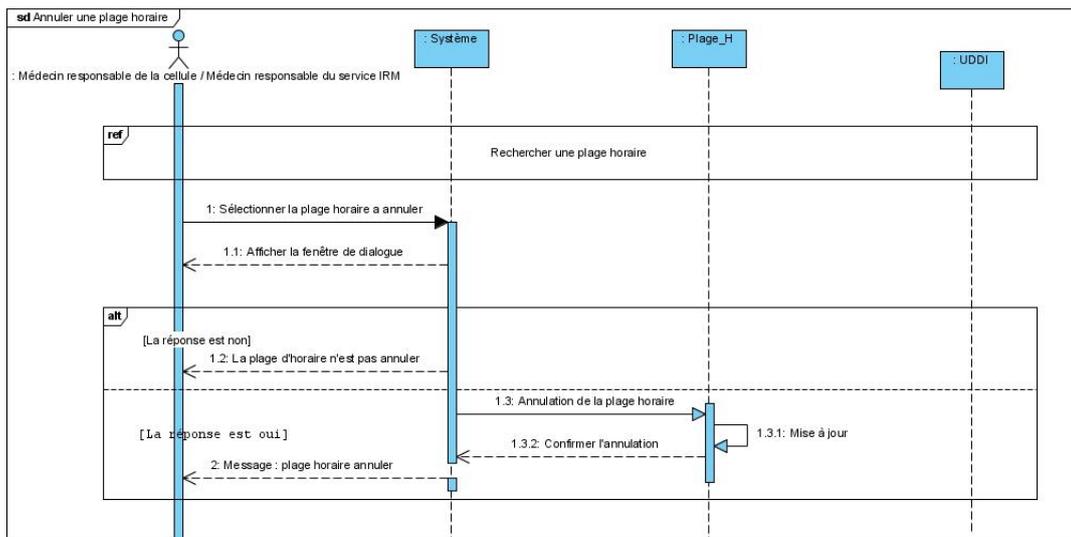


FIGURE 3.13 – Diagramme de séquence du cas d'utilisation "Annuler une Plage Horaire"

— Consulter une Plage Horaire

Le médecin responsable de la cellule d'accueil ou Le médecin responsable du service IRM doivent d'abord s'authentifier et effectuer une recherche pour consulter la liste des plages horaires.

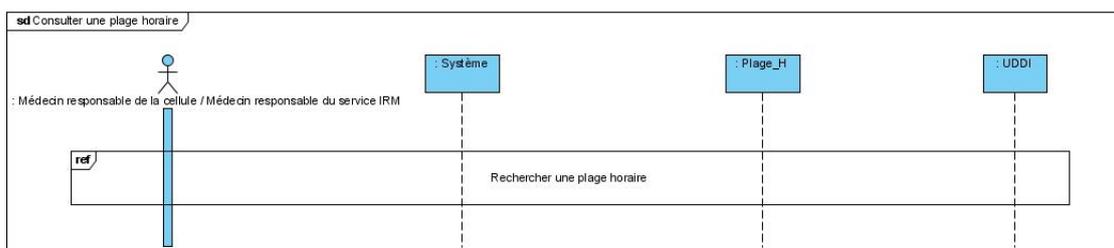


FIGURE 3.14 – Diagramme de séquence du cas d'utilisation "Consulter une Plage Horaire"

— Modifier une Plage Horaire

La modification d'une plage horaire se fait par le médecin responsable de la cellule d'accueil et Le médecin responsable du service IRM qui après l'authentification recherchent la plage horaire à modifier puis annule cette plage horaire et enfin ils ajoutent une plage horaire.

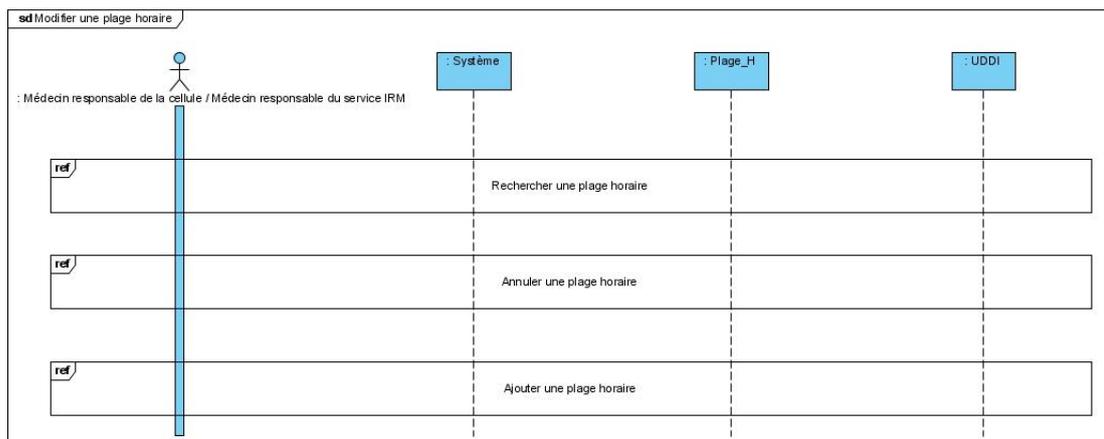


FIGURE 3.15 – Diagramme de séquence du cas d'utilisation "Modifier une Plage Horaire"

3.1.5 Diagramme de séquence "Générer Bilan d'activités"

La secrétaire après authentification saisie l'intervalle des dates puis valide. Le système doit vérifier la saisie et enfin le bilan sera affiché.

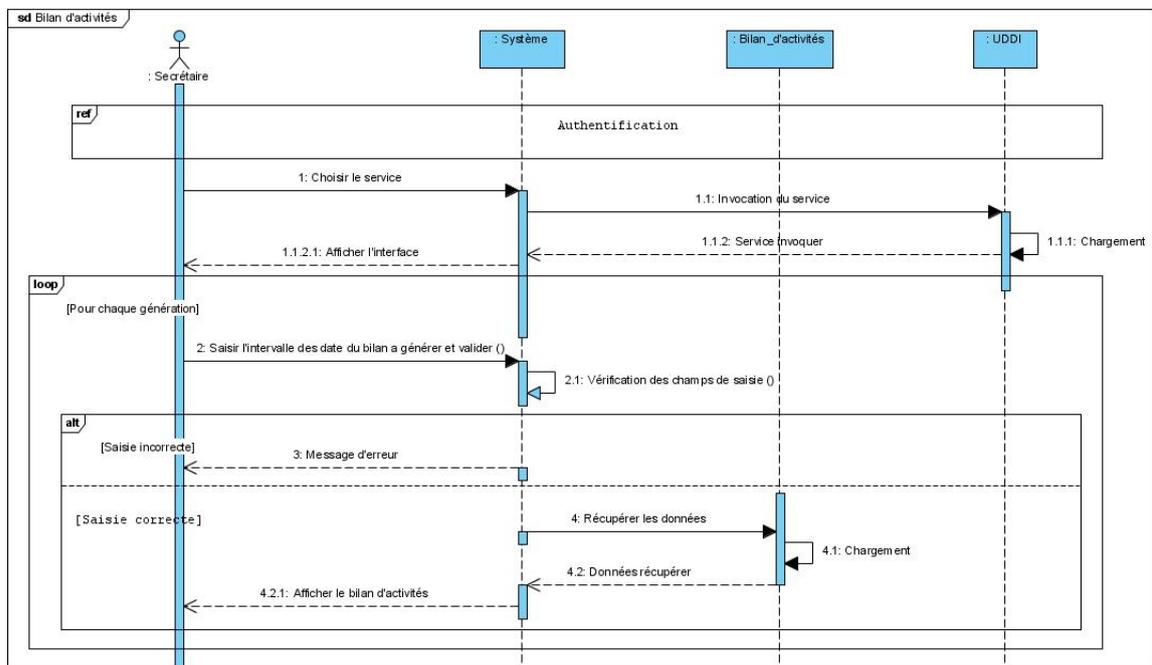


FIGURE 3.16 – Diagramme de séquence du cas d'utilisation "Générer Bilan d'activités"

3.2 Diagramme de classes

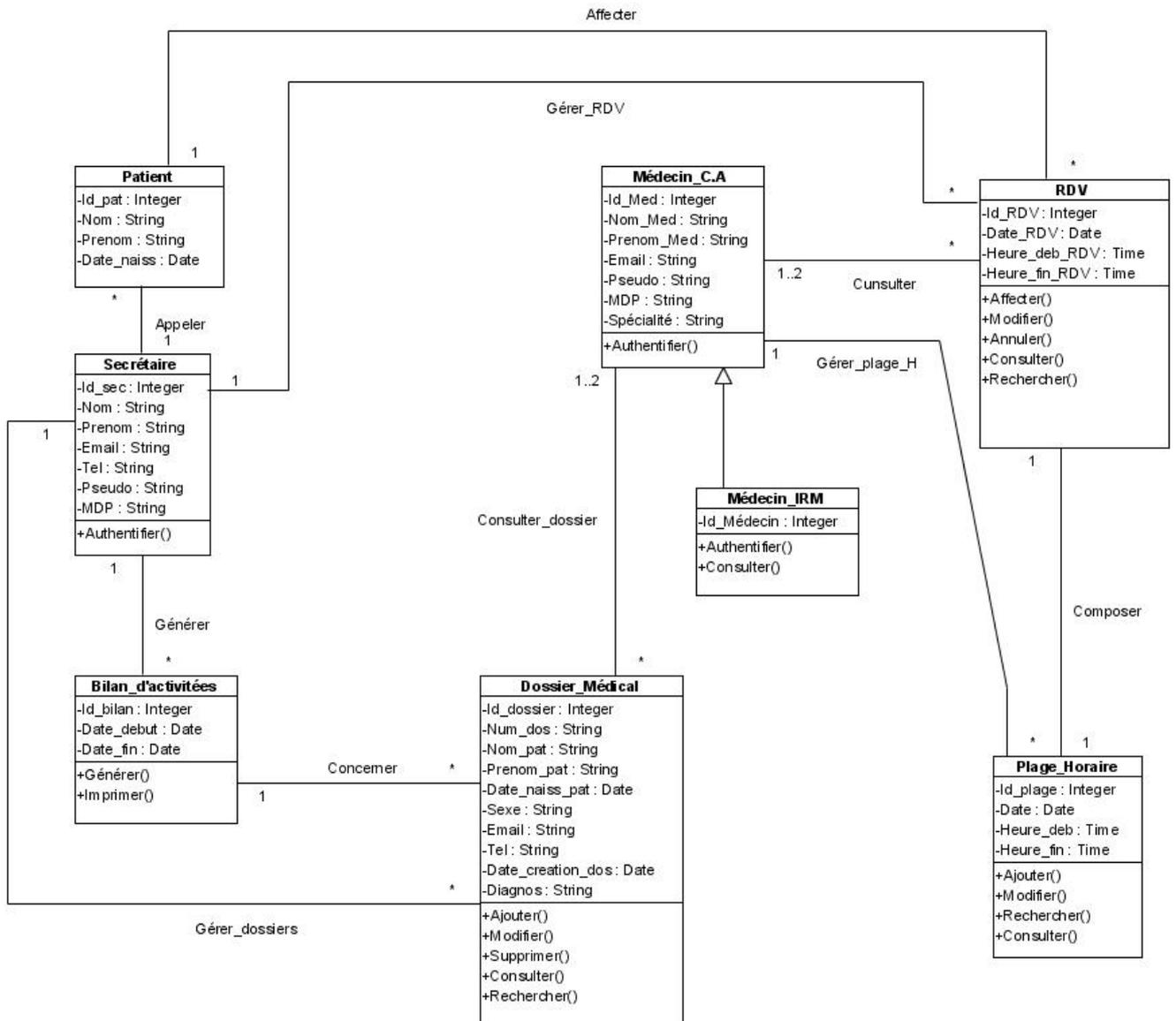


FIGURE 3.17 – Diagramme de classes

3.3 Passage au modèle relationnel

Le modèle relationnel est le modèle logique de données qui correspond à l'organisation des données dans les bases de données relationnelles. Un modèle relationnel est composé de relations, appelée table. Ces tables sont décrites par des attributs ou champs. Pour décrire une relation, on indique tout simplement son nom, suivi du nom de ses attributs entre parenthèses. L'identifiant d'une relation est composé d'un ou plusieurs attributs qui forment la clé primaire. Une relation peut faire référence à une autre en utilisant une clé étrangère, qui correspond à la clé primaire de la relation référencée

3.3.1 Règles de passage du diagramme de classes au modèle logique

Les règles de passage sont :

1. Relation (1..*) : il faut ajouter un attribut de type clé étrangère dans la relation fils de l'association. L'attribut aura le nom de la clé primaire de la relation père de l'association.
2. Relation (1..1) : il faut ajouter une relation qui prend les deux clé primaire des classes mère comme clé étrangère.
3. Relation d'héritage : la clé primaire de la classe mère est utilisée pour identifier chacune de ses classes filles : cette clé étant pour chaque classe fille, à la fois clé primaire et une clé étrangère vers la classe mère.[14]

3.3.2 Modèle logique de données

En appliquant les règles de passages pour notre diagramme de classes, nous aboutissons au schéma relationnel suivant :

Médecin_CA(Id_Med, Nom_Med, Prenom_Med, Email, Pseudo, MDP, Spécialité).

Médecin_IRM(id_médecin, id_Med#, Nom_Med, Prenom_Med, Email, Pseudo, MDP, Spécialité).

Secrétaire(Id_Sec, Nom, Prenom, Email, Tel, Psuedo, MDP).

Patient(Id_pat, Nom, Prenom, Date_naiss, Id_Sec#).

RDV(Id_RDV, Date_RDV, Heure_deb_RDV, Heure_fin_RDV, Id_pat#, Id_Med#, Id_Sec#, Id_plage#).

Plage_Horaire(Id_plage, Date, Heure_deb, Heure_fin, Id_Med#).

composer(Id_plage#, Id_RDV#).

Dossier_Médical(Id_dos, Num_dos, Nom_dos, Nom_pat, prenom_pat, Date_naiss_pat, sexe, Email, Tel, Date_creation_dos, Diagnos, Id_Sec#, Id_Med#).

Bilan_d'activités(Id_bilan, Date_debut, Date_fin, Id_Sec#, Id_dossier).

3.4 Conclusion

Nous avons présenté dans ce chapitre la phase de conception de notre solution via les diagrammes de séquence, qui nous ont permis de décrire de manière globale et détaillée, le fonctionnement du système afin d'en faciliter la réalisation et la maintenance. Ces derniers ont mis en avant les interactions entre les différents objets constituant notre application. en suite, nous avons conçu le diagramme de classes donnant ainsi une vue plus structurée des éléments qui formeront la base de données liée à notre application. Dans le chapitre suivant, nous entamons la phase d'implémentation.

4

Implémentation

Introduction

Ce chapitre représente le dernier volet de ce rapport. Nous entamons la réalisation après l'étape de conception défini au préalable . Nous présentons les différentes interfaces de notre application réalisées à l'aide des outils cités et défini dans le premier chapitre.

4.1 Interfaces et déroulement de l'application

4.1.1 Interface d'authentification

Voici L'interface d'authentification permettant aux différents utilisateurs d'avoir accès au contenu de l'application et ce selon la fonction occupée dans le service.

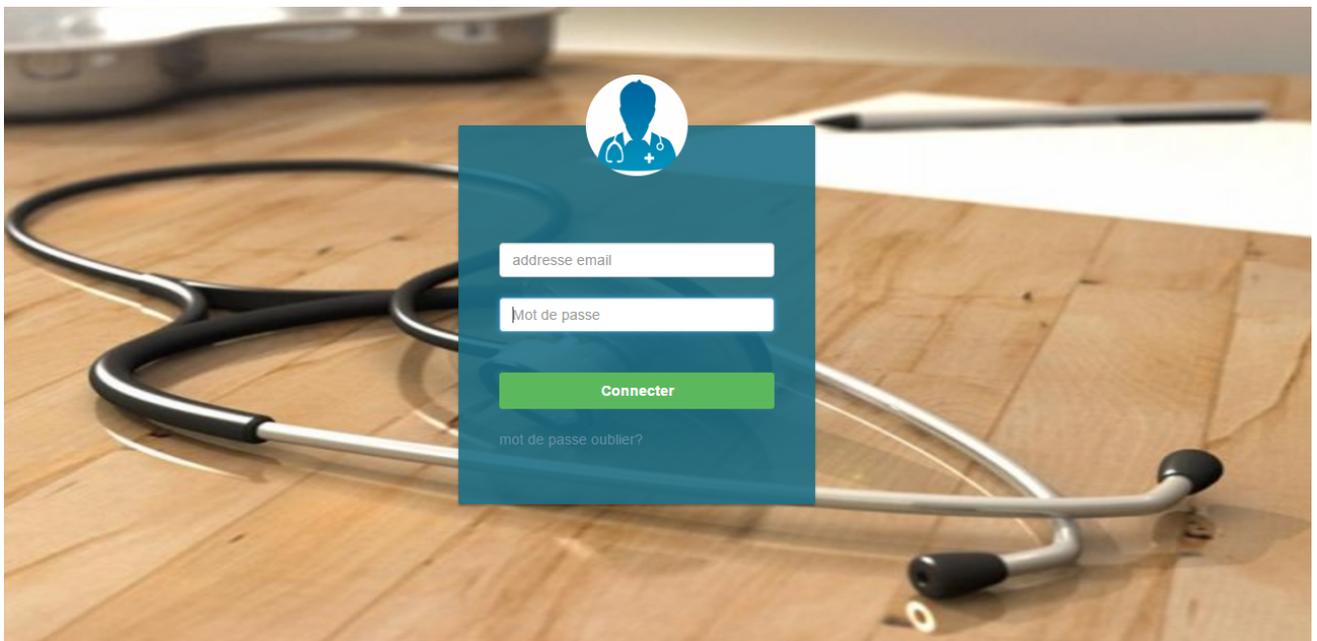


FIGURE 4.1 – Interface d'authentification

4.1.2 Interface secrétaire

Gestion de Rendez-vous

Cette interface offre la possibilité à la secrétaire d'affecter des rendez-vous aux patients, quand ce dernier appel pour prendre rendez-vous, la secrétaire choisit le service rendez-vous puis elle remplit les champs et valide. En cliquant sur le bouton "valider" le service rendez-vous sera invoqué par la méthode décrite dans la Figure 4.3, ce qui fait appel au service

stocker dans l'annuaire JUDDI (Voir, la Figure 4.4).

La Figure 4.5 décrit une partie du fichier WSDL du service Rendez-vous qui assure la liaison entre l'interface client et le fournisseur du service.

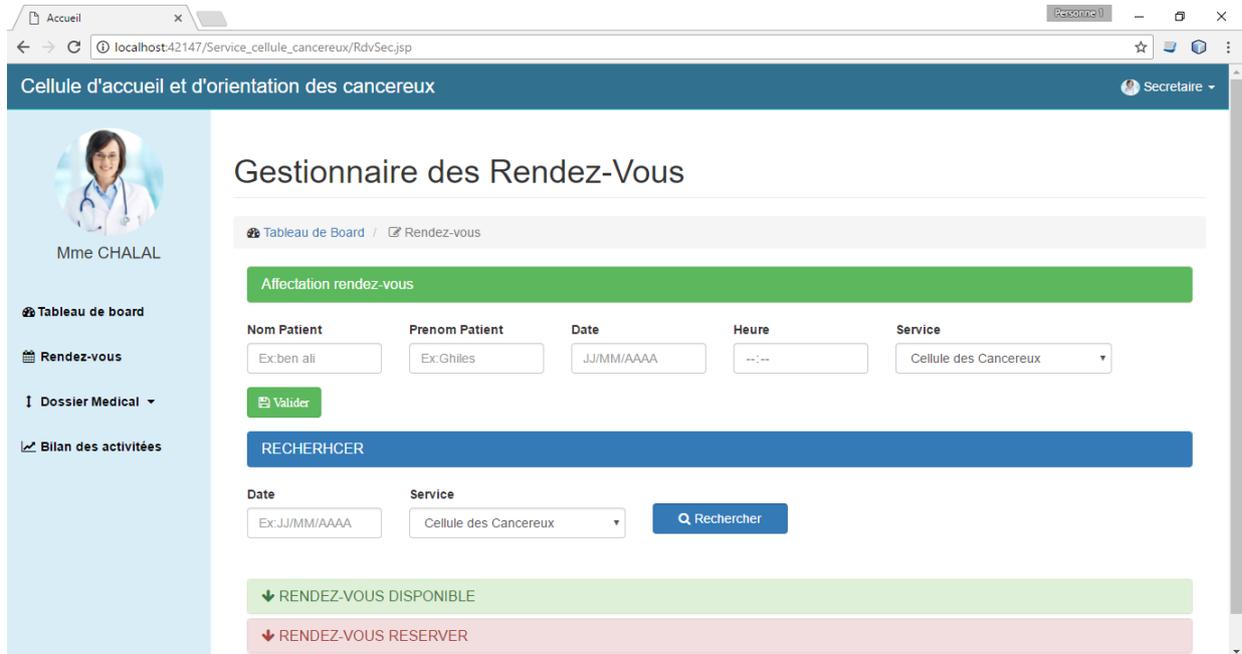


FIGURE 4.2 – Interface Gestion des Rendez-vous

```

org.my.ns.RdvServiceService service = new org.my.ns.RdvServiceService();
org.my.ns.ServiceRDV port = service.getServiceRDVPort();
// TODO initialize WS operation arguments here
java.lang.String nomPatient = a;
java.lang.String prenomPatient = b;
java.lang.String date = c;
java.lang.String heureDebt = d;
java.lang.String specialite = e;
// TODO process result here
java.lang.String result = port.ajouterRendezVous(nomPatient, prenomPatient, date, heureDebt, specialite);

```

FIGURE 4.3 – Méthode d'invocation

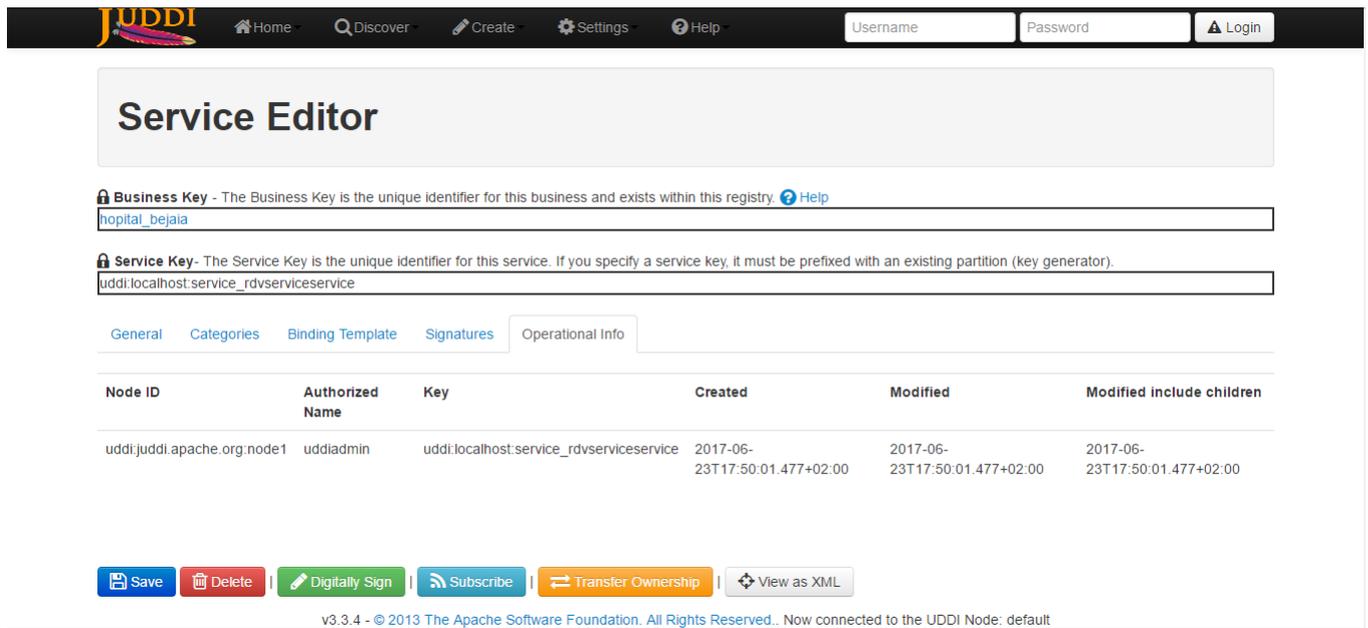


FIGURE 4.4 – Interface de l’annuaire



FIGURE 4.5 – WSDL du fichier

Gestion des Dossiers médicaux

Cette interface est dédiée seulement à la secrétaire afin de gérer le dossier, elle peut ajouter, modifier ou supprimer un dossier. En cliquant sur le bouton "Enregistrer" le service Dossier médical sera invoqué par la méthode (Voir, Figure 4.7) qui est stockée dans l'annuaire (Voir, Figure 4.8).

La Figure 4.9 décrit le fichier de WSDL du service Dossier médical.

FIGURE 4.6 – Interface Ajouter un Dossier médical

```

org.my.ns.ServiceDossierService service = new org.my.ns.ServiceDossierService();
org.my.ns.DossierMedical port = service.getDossierMedicalPort();
// TODO initialize WS operation arguments here
java.lang.String dateDoss = request.getParameter("date_doss");
java.lang.String numDossier = request.getParameter("num_doss");
java.lang.String nom = request.getParameter("nom_patient");
java.lang.String prenom = request.getParameter("prenom_patient");
java.lang.String dateNaissance = request.getParameter("dt_patient");
java.lang.String sexe = request.getParameter("sexe").toString();
java.lang.String adresse = request.getParameter("address");
java.lang.String telephone = request.getParameter("telephone");
java.lang.String diagnostique = request.getParameter("diag");

// TODO process result here
java.lang.String result = port.ajouterDossier(dateDoss, numDossier, nom, prenom, dateNaissance,
    sexe, adresse, telephone, diagnostique);

```

FIGURE 4.7 – Méthode d’invocation

Service Editor

Business Key - The Business Key is the unique identifier for this business and exists within this registry. [Help](#)
 hopital_bejaia

Service Key - The Service Key is the unique identifier for this service. If you specify a service key, it must be prefixed with an existing partition (key generator).
 uddi:localhost:service_servicedossierservice

General Categories Binding Template Signatures Operational Info

Node ID	Authorized Name	Key	Created	Modified	Modified include children
uddi:juddi.apache.org:node1	uddiadmin	uddi:localhost:service_servicedossierservice	2017-06-23T15:37:07.674+02:00	2017-06-23T17:50:01.414+02:00	2017-06-23T17:50:01.414+02:00

Save Delete Digitally Sign Subscribe Transfer Ownership View as XML

v3.3.4 - © 2013 The Apache Software Foundation. All Rights Reserved. Now connected to the UDDI Node: default

FIGURE 4.8 – Interface de l’annuaire

```
<!--
Published by JAX-WS RI (http://jax-ws.java.net). RI's version is JAX-WS RI 2.2.9-b138926.1835 svn-revision#5f6196f2898e94688654c2f4e38e85b245e51e.
-->
<!--
Generated by JAX-WS RI (http://jax-ws.java.net). RI's version is JAX-WS RI 2.2.9-b138926.1835 svn-revision#5f6196f2898e94688654c2f4e38e85b245e51e.
-->
<?xml-stylesheet href="http://docs.oasis-open.org/ws/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" xmlns:wsp1="http://www.w3.org/ns/ws-policy" xmlns:wsp2="http://schemas.xmlsoap.org/ws/2004/05/policy" xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://my.org/ns/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://schemas.xmlsoap.org/wsdl/" targetNamespace="http://my.org/ns/" name="ServiceDossierService">
<types>
<xsd:schema
xmlns:import namespace="http://my.org/ns/" schemaLocation="http://localhost:4568/?xsd-1"/>
</xsd:schema>
</types>
<message name="Supprimer_dossier">
<part name="parameters" element="tns:Supprimer_dossier"/>
</message>
<message name="Supprimer_dossierResponse">
<part name="parameters" element="tns:Supprimer_dossierResponse"/>
</message>
<message name="rechercher_dossier">
<part name="parameters" element="tns:rechercher_dossier"/>
</message>
<message name="rechercher_dossierResponse">
<part name="parameters" element="tns:rechercher_dossierResponse"/>
</message>
<message name="recherche_id">
<part name="parameters" element="tns:recherche_id"/>
</message>
<message name="recherche_idResponse">
<part name="parameters" element="tns:recherche_idResponse"/>
</message>
<message name="bilan">
<part name="parameters" element="tns:bilan"/>
</message>
<message name="bilanResponse">
<part name="parameters" element="tns:bilanResponse"/>
</message>
<message name="ajouter_dossier">
<part name="parameters" element="tns:ajouter_dossier"/>
</message>
<message name="ajouter_dossierResponse">
<part name="parameters" element="tns:ajouter_dossierResponse"/>
</message>
<message name="verifier_attribut_existe">
<part name="parameters" element="tns:verifier_attribut_existe"/>
</message>
<message name="verifier_attribut_existeResponse">
<part name="parameters" element="tns:verifier_attribut_existeResponse"/>
</message>
<message name="modifier_dossier">
<part name="parameters" element="tns:modifier_dossier"/>
</message>
<message name="modifier_dossierResponse">
<part name="parameters" element="tns:modifier_dossierResponse"/>
</message>
<message name="afficher_dossier">
<part name="parameters" element="tns:afficher_dossier"/>
</message>
<message name="afficher_dossierResponse">
</message>
</types>
```

FIGURE 4.9 – WSDL du fichier

Générer Bilan d’activités

L’interface ci-dessous est dédiée seulement à la secrétaire ou elle peut générer un bilan d’activité selon l’intervalle du temps souhaiter. Après la génération du bilan la secrétaire peut faire une impression (Voir Figure 4.8).

DE : 01/01/2017 A : 30/12/2017 Générer le Bilan Lancer l'impression

ID DOSSIER	N° DOSSIER	SEXE	ADRESSE	DIAGNOSTIQUE
28	D001	FEMME	BEJAIA	dosage de marqueurs tumoraux.
29	D002	HOMME	sidi aich	IRM (Imagerie par Résonance Magnétique)
30	D003	ENFANT	TAZMALT	IRM (Imagerie par Résonance Magnétique)

FIGURE 4.10 – Interface Générer Bilan d'activités

17/09/2017 Accueil



C.H.U BEJAIA خلية استقبال و توجيه مرضى السرطان

Cellule d'Accueil et d'Orientation des
Cancereux

ID DOSSIER	N° DOSSIER	SEXE	ADRESSE	DIAGNOSTIQUE
28	D001	FEMME	BEJAIA	dosage de marqueurs tumoraux.
29	D002	HOMME	sidi aich	IRM (Imagerie par Résonance Magnétique)
30	D003	ENFANT	TAZMALT	IRM (Imagerie par Résonance Magnétique)
31	D005	HOMME	TOUJA	IRM

http://localhost:42147/Service_cellule_cancereux/BilanActiviteAn.jsp 1/1

FIGURE 4.11 – Interface Impression bilan

4.1.3 Interface Médecin

Gérer Les Plages horaire

L'interface gérer plage horaire etant ergonomique et très simple à utiliser, offre aux médecins la possibilité d'ajouter des plages horaire et de consulter la liste des rendez-vous après la recherche. Cela se fait par l'invocation du service Rendez-vous via la méthode (Voir, Figure 4.13). Le service se trouve dans l'annuaire (Voir, Figure 4.4).

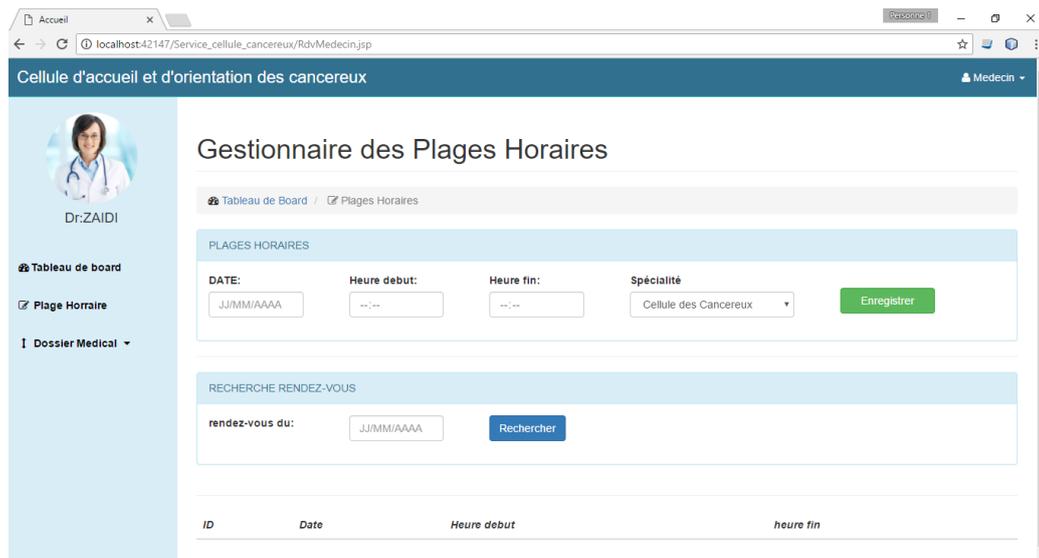


FIGURE 4.12 – Interface Gérer Plage horaire

```

org.my.ns.RdvServiceService service = new org.my.ns.RdvServiceService();
org.my.ns.ServiceRDV port = service.getServiceRDVPort();
// TODO initialize WS operation arguments here
java.lang.String date = request.getParameter("date_plage");
java.lang.String heureDeb = request.getParameter("heured");
java.lang.String heureFin = request.getParameter("heuref");
java.lang.String statut = "libre";
java.lang.String specialite = request.getParameter("spec");
// TODO process result here
java.lang.String result = port.ajoutRdv(date, heureDeb, heureFin, statut, specialite);
    
```

FIGURE 4.13 – Méthode d’invocation

Consulter Dossier médicale

Le médecin introduit le nom et prénom du patient et en cliquant sur le bouton rechercher, la fiche médical de ce dernier sera afficher.

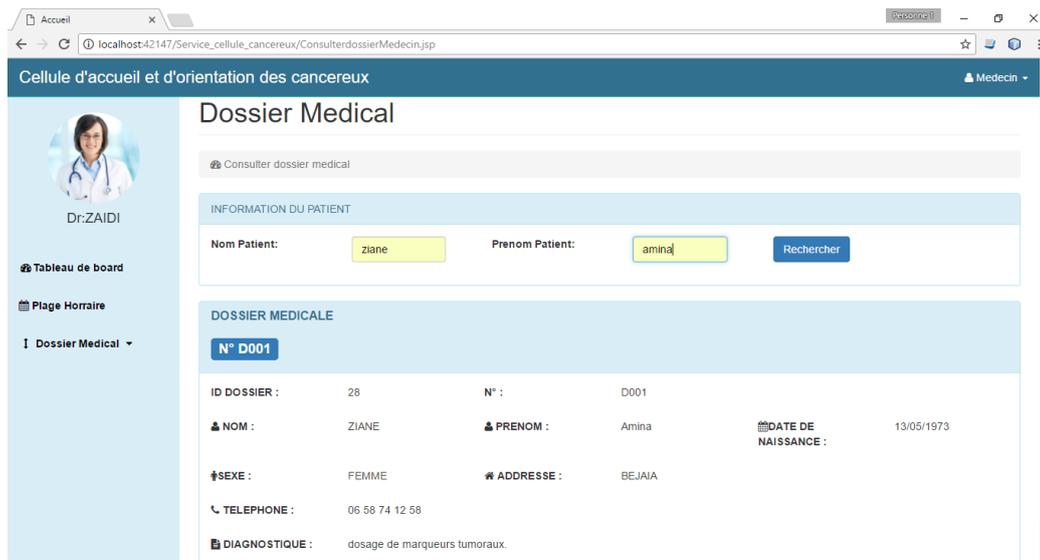


FIGURE 4.14 – interface Consulter Dossier médicale

Consulter Rendez-vous médecin IRM

Le médecin introduit la date des rendez-vous qu'il souhaite consulter et la liste des rendez-vous correspondant à cette dernière sera affichés.

RENDEZ-VOUS					
ID rendez-vous	DATE	HEURE DEBUT	HEURE FIN	NOM	PRENOM
38	08/08/2017	11:00	11:30	AKILAL	leila

FIGURE 4.15 – interface Consulter Dossier médicale

4.2 Conclusion

Au cour de ce chapitre nous avons présenté les différents aspects et fonctionnalité de notre application ainsi que quelques scénario et interface des cas d'utilisation cités au préalable.

Conclusion générale

Au cours de ce mémoire, nous avons présenté les différentes étapes de la conception et la réalisation de notre application pour le services de la cellule d'accueil et d'orientation des cancéreux et service d'IRM pour l'hôpital KHALLIL AMRANE de Bejaia. Pour la réalisation de notre projet, nous nous sommes rapprochés de l'hôpital KHALLIL AMRANE à travers ses différentes unités et nous avons déterminé les points importants que doit traiter notre projet.

L'objectif majeur de notre projet consiste à faciliter le travail des employés, leur faire éviter d'avoir à saisir les documents ainsi que les bilans d'activités des différentes opérations effectuées, mais aussi leur faire gagner du temps, et surtout le point le plus important qui est de garder la trace de chaque opération effectuée ou document saisi dans la base de données pour d'éventuelles mises à jour, consultation ou impression.

Afin de satisfaire les besoins des utilisateurs nous avons commencé la conception en utilisant le formalisme UML et la mise en œuvre des bases de données avec le gestionnaire de bases de données MySQL et enfin la concrétisation de l'application sous le langage de programmation JSP.

Ce projet a fait l'objet d'une expérience très intéressante, car elle nous a permis de nous familiariser avec de nouvelles notions d'une part, et d'améliorer nos connaissances et nos compétences dans le domaine de la programmation, et d'autre part de renforcer notre sens de responsabilité dans la gestion des projets qui permet l'insertion dans le domaine professionnel.

Enfin, nous espérons que notre projet puisse répondre aux besoins fixés et satisfaire toutes les personnes qui ont contribué à sa réalisation, ainsi que les utilisateurs pour l'exercice de leur profession, sachant que ce projet ouvre d'autres possibilité et perspectives d'amélioration comme les prises de rendez-vous en ligne.

Bibliographie

- [1] www.chubejaia.dz *Organigramme général de l'hôpital*
- [2] Jean-louis Bénard, Laurent Bossavit, Régis Medina, Dominic Williams, *Gestion de projet eXtreme Programming*, groupe Eyrolles, 2005.
- [3] Pascal Roques, *les cahiers du programmeur UML2 modélisé une application web*, Eyrolles ,2007,4eme édition.
- [4] Hubert kadima, valérie Monfort , *les web services, technique, démarche, et outils XML, WSDL, SOAP, UDDI, RosettaNet, UML*.
- [5] Eric NEWCOMER. *Understanding Web services- XML, WSDL, SOAP and UDDI*. Addison wesley édition, May 2002.
- [6] Farouk TOUMANI Patrick KELLERT. *Les web services sémantique*. Vuibert informatique édition 2004.
- [7] BOUZERA Toufik, HEROUGE Imade. *Conception et réalisation d'un parseur de services web*. Mémoire de fin d'étude université de Bejaia 2011
- [8] Tarek MELITI. *Interopérabilité des services web complexes. Application système multi-agents*. Mémoire de magister, Université de paris IX Dauphine, Decembre 2004.
- [9] A. H. S. T. A. BOUBEKRI Abdelmalek, CHIKOU Youghorta, *Conception et réalisation d'une application de gestion de stock des matières premières*. Université de Béjaia, 2014.
- [10] [Http ://www.siteduzero.com/informatique/tutoriels/concevez-votre-site-web-avecphp-et-mysql/presentation-des-bases-de-donnees-2](http://www.siteduzero.com/informatique/tutoriels/concevez-votre-site-web-avecphp-et-mysql/presentation-des-bases-de-donnees-2). O. Classrooms, "présentation des bases de données." [http ://www.siteduzero.com/informatique/tutoriels/concevez-votre-site-web-avecphp-et-mysql/presentation-des-bases-de-donnees-2](http://www.siteduzero.com/informatique/tutoriels/concevez-votre-site-web-avecphp-et-mysql/presentation-des-bases-de-donnees-2).
- [11] [Https ://www.phpmyadmin.net/](https://www.phpmyadmin.net/).Phpmyadmin, *Apporter mysql sur le web*.
- [12] ANNICK FRON, *Architectures réparties en JAVA, Internet des objets avec SOAP, RMI, CORBA, JMS, sockets et services web*, 2ème édition
- [13] Lauren Debrouwer et Fien Ven Der Heyde, *UML 2 modélisation des objets*

- [14] R. Pascal and V. Franck, *UML2 en action, de l'analyse des besoins en action*, 4^{ème} édition
- [15] Jonathan FONTANEL, Philippe LACOMME, Libo REN, *Les Services Web, concevoir et utiliser des applications 2.0, C#, JAVA, PHP, API JavaScript, Andoroid SDK, IOS SDK*

RÉSUMÉ

La solution proposée dans ce document consiste à réaliser une solution web, afin d'informatiser et moderniser la gestion des rendez-vous, des dossiers médicaux et proposer et gérer les plages horaires au niveau des deux services Cellule d'accueil des cancéreux et IRM, au sein de l'hôpital KHELLIL AMRANE de Bejaia. Pour la concevoir, nous avons utilisé différents outils tel que le langage de modélisation UML, qui propose un formalisme simple et performant. Sa mise en œuvre, quant à elle, s'est faite sous l'environnement de développement intégré, Netbeans, à l'aide de Java et un serveur d'application. Le tout est combiné au registre UDDI.

Mots-clés : Services web, eXtreme Programming, JAX-WS, Java, Php, UDDI.

ABSTRACT

The solution proposed in this document is to create a web-based solution in order to computerize and modernize the management of appointments, medical files and propose and manage the time slots at the level of both services. Cancer and IRM, at the KHELLIL AMRANE hospital in Bejaia. To design it, we used different tools such the modeling language UML, which offers a simple and efficient formalism. As for here implementation was done under the integrated development environment, Netbeans, using Java and an application server. The all is combined with the UDDI register.

Keywords : Web services, eXtreme Programming, JAX-WS, Java, PHP, UDDI.