

الجمهورية الجزائرية الديمقراطية الشعبية  
République Algérienne Démocratique et Populaire  
وزارة التعليم العالي والبحث العلمي  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



N° Réf :.....

Centre Universitaire  
Abdel Hafid Boussouf Mila

Institut des sciences et de la technologie

Département de Mathématiques et Informatique

Mémoire préparé En vue de l'obtention du diplôme de  
Licence

En : Filière informatique

CONCEPTION ET RÉALISATION D'UN SITE WEB  
DYNAMIQUE POUR GÉRER AUTOMATIQUEMENT  
LES RENDEZ-VOUS DANS UNE CLINIQUE  
MÉDICALE

Préparé par : *ZOUAGHI Ishaq*

Encadré par : *HETTAB Abdel kamel*

*M.A. A au C.U. Abdel Hafid Boussouf Mila*

Année universitaire : 2014/2015

# Remerciement

---

C'est avec l'aide de Dieu qu'a vu le jour ce présent travail. Ensuite, il n'aurait pas pu être achevé sans le soutien, les conseils et les encouragements de mon promoteur, Mr Hettab Abdel Kamel, Maître assistant A au Centre Universitaire Abdel Hafid Boussouf Mila, à qui j'exprime toute ma gratitude et mes sincères remerciements

---

# Dédicace

A mes chers parents, Mohammed et Akila, pour  
leur soutien et leur patience,  
A mon adorable sœur Asma  
A mes frères Ayoub et Oussama  
A vous tous, je dédie mon travail de fin d'étude.

## ملخص:

تطورت التقنيات وتطورت الانترنت واحتياجات الناس بصورة لم يعد مزود الويب قادر على مجاراتها. فاحتيج لمواقع ذات صفحات آلية أو ديناميكية. في هذا العمل قمنا بتطوير موقع ديناميكي لحجز مواعيد الفحوصات في عيادة طبية عبر الإنترنت نظرا لما يعانيه المرضى في قاعات الانتظار، حيث يوفر عنهم عناء التنقل للعيادة لأجل الحجز. وقد انجزنا مشروعنا هذا باستخدام لغة النمذجة الموحدة UML مع عملية التطوير المبسطة UP خلال جميع مراحل المشروع. كما استعملنا العديد من الأدوات ولغات البرمجة الخاصة بتطوير المواقع.

## الكلمات الرئيسية:

الانترنت، الشبكة، الموقع، عيادة طبية، مواعيد، لغة النمذجة UML، عملية التطوير المبسطة UP.

## Résumé :

La technologie Internet ainsi que les besoins des personnes à celle-ci ont connu une évolution dont le serveur web n'est plus en mesure de suivre. Le développement de sites avec des pages dynamiques s'avère nécessaires pour répondre aux exigences du quotidien de chacun de nous. C'est pourquoi dans ce travail nous avons développé un site web dynamique pour la prise de rendez-vous dans une clinique médicale pour effectuer des consultations générales ou spécialisée ou encore pour faire des analyses médicales. Cela permet aux patients d'éviter les attentes inutiles et d'économiser du temps et de l'argent. Nous avons accompli notre projet dans toutes ses phases en utilisant le langage de modélisation unifié UML et le processus UP simplifiée. Nous avons utilisé également de nombreux outils et langages de programmation.

## Mots clés :

Internet, Les sites WEB, clinique médicale, rendez-vous, UML, UP simplifié.

## Abstract:

The internet Technology as well as the needs of people to it knew an evolution whose Web servers are not able to follow. The development site with dynamic pages proves necessary to reply the daily requirements of each of us. Therefore, in this work we have developed a dynamic Web site for the appointments reservation in a medical clinic to make general or specific consultations or to make medical analyses. That makes it possible to the patients to avoid useless waiting and to save money and time. We achieved our project in all its phases by using the unified modeling language UML and the "UP simplifié" process. We also used many tools and programming languages.

## Key words:

Internet, the WEB sites, medical clinic, appointment, UML, simplified UP.

# SOMMAIRE

Introduction générale .....	1
Chapitre I : L'internet et les applications web	
1. Introduction .....	3
2. Le réseau Internet.....	3
2.1. Définition .....	3
2.2. Historique .....	3
2.4. Les services d'Internet .....	5
2.5. L'architecture de l'Internet .....	5
3. Le World Wide Web .....	5
3.1. Définition:.....	5
3.2. Le principe du Web .....	6
3.3. Les niveaux du Web .....	6
3.4. Les sites web .....	7
3.4.1. Les sites statiques.....	7
3.4.2. Les sites dynamiques .....	7
4. Le modèle client-serveur .....	8
4.1. Description .....	8
4.2. Avantages de l'architecture client/serveur .....	9
4.3. Inconvénients du modèle client/serveur .....	9
4.4. Fonctionnement d'un système client/serveur .....	10
5. Conclusion .....	10
Chapitre II : L'UML et le processus unifié UP	
1. Introduction .....	11
2. L'UML ( <i>Unified Modeling Language</i> ) .....	11
2.1. Définition .....	11
2.2. Les méthodes objets.....	11
2.3. Intérêt d'une méthode objet .....	12
2.4. Les Diagrammes d'UML.....	13
3. Le Processus Unifié (UP).....	14
3.1. Définition .....	14
3.2. Les principes d'UP.....	14
3.2.1. UP itératif et incrémental.....	14
3.2.2. UP centré sur l'architecture.....	14
3.2.3. UP piloté par les cas d'utilisation d'UML.....	16
4. Une méthode simple et générique .....	16

<b>4.1. Identification des besoins et spécification des fonctionnalités</b> .....	17
<b>4.2. Phase d'analyse</b> .....	19
<b>4.3. Phase de conception</b> .....	22
<b>5. Conclusion</b> .....	<b>23</b>

### Chapitre III : Identification des besoins

<b>1. Introduction</b> .....	<b>24</b>
<b>2. Identification des acteurs</b> .....	<b>24</b>
<b>3. Identification des cas d'utilisation</b> .....	<b>24</b>
<b>4. Diagramme de cas d'utilisation</b> .....	<b>25</b>
<b>5. Les fiches descriptives</b> .....	<b>26</b>
5.1- Fiche descriptive du cas d'utilisation « S'identifier» .....	26
5.2- Fiche descriptive du cas d'utilisation « S'inscrire» .....	28
5.3- Fiche descriptive du cas d'utilisation « visiter le site» .....	30
5.4- Fiche descriptive du cas d'utilisation « Tester la disponibilité des RDV» .....	32
5.5- Fiche descriptive du cas d'utilisation «Choisir un RDV» .....	34
5.6- Fiche descriptive du cas d'utilisation «Modifier un RDV» .....	36
5.7- Fiche descriptive du cas d'utilisation «Annuler un RDV» .....	38
5.8- Fiche descriptive du cas d'utilisation «Ajouter un service» .....	40
5.9- Fiche descriptive du cas d'utilisation «Modifier un service» .....	42
5.10- Fiche descriptive du cas d'utilisation «Annuler un service» .....	44
5.11- Fiche descriptive du cas d'utilisation «Rechercher un RDV» .....	46
5.12- Fiche descriptive du cas d'utilisation «Rechercher un Patient» .....	48
<b>6. Conclusion</b> .....	<b>49</b>

### Chapitre IV : Analyse du domaine

<b>1. Introduction</b> .....	<b>50</b>
<b>2. Modèle de domaine</b> .....	<b>50</b>
<b>3. Les diagrammes de classes participantes</b> .....	<b>50</b>
3.1- Le cas d'utilisation « S'identifier » .....	50
3.2- Fiche descriptive du cas d'utilisation « S'inscrire » .....	51
3.3- Le cas d'utilisation « Visiter le site » .....	51
3.4- Le cas d'utilisation « Tester la disponibilité des RDV » .....	51
3.5- Le cas d'utilisation « Choisir un RDV » .....	52
3.6- Le cas d'utilisation « Modifier un RDV » .....	53
3.7- Le cas d'utilisation « Annuler un RDV » .....	53
3.8- Le cas d'utilisation « Ajouter un service » .....	54
3.9- Le cas d'utilisation « Modifier un service » .....	54
3.10- Le cas d'utilisation « Annuler un service » .....	55

3.11- Le cas d'utilisation « Rechercher un RDV ».....	55
3.12- Le cas d'utilisation « Rechercher un Patient ».....	56
<b>7. Conclusion .....</b>	<b>56</b>

## Chapitre V : Phase de conception

<b>1. Introduction .....</b>	<b>57</b>
<b>2. Les diagrammes d'interaction .....</b>	<b>57</b>
2.1- Le cas d'utilisation « S'identifier » .....	57
2.2- Le cas d'utilisation « S'inscrire ».....	58
2.3- Le cas d'utilisation « Visiter le site ».....	58
2.4- Le cas d'utilisation « Tester la disponibilité des RDV ».....	59
2.5- Le cas d'utilisation « Choisir un RDV ».....	59
2.6- Le cas d'utilisation « Modifier un RDV » .....	60
2.7- Le cas d'utilisation « Annuler un RDV ».....	61
2.8- Le cas d'utilisation « Ajouter un service ».....	62
2.9- Le cas d'utilisation « Modifier un service ».....	63
2.10- Le cas d'utilisation « Annuler un service ».....	64
2.11- Le cas d'utilisation « Rechercher un RDV ».....	65
2.12- Le cas d'utilisation « Rechercher un Patient ».....	65
<b>3. Les diagrammes de classe de conception.....</b>	<b>66</b>
3.1- Le cas d'utilisation « S'identifier » .....	66
3.2- Fiche descriptive du cas d'utilisation « S'inscrire » .....	66
3.3- Le cas d'utilisation « Visiter le site ».....	67
3.4- Le cas d'utilisation « Tester la disponibilité des RDV ».....	67
3.5- Le cas d'utilisation « Choisir un RDV » .....	68
3.6- Le cas d'utilisation « Modifier un RDV » .....	69
3.7- Le cas d'utilisation « Annuler un RDV ».....	70
3.8- Le cas d'utilisation « Ajouter un service ».....	70
3.9- Le cas d'utilisation « Modifier un service ».....	71
3.10- Le cas d'utilisation « Annuler un service ».....	71
3.11- Le cas d'utilisation « Rechercher un RDV ».....	72
3.12- Le cas d'utilisation « Rechercher un Patient ».....	72
<b>4. Diagramme de classe .....</b>	<b>73</b>
<b>5. Conclusion .....</b>	<b>73</b>

## Chapitre VI : L'implémentation

<b>1. Introduction .....</b>	<b>74</b>
<b>2. Les Langages de programmation.....</b>	<b>74</b>
2.1. Les langages HTML et CSS.....	74

2.1.1. HTML ( <i>HyperText MarkupLanguage</i> ).....	75
2.1.2. CSS ( <i>Cascading Style Sheets</i> , aussi appelées <i>Feuilles de style</i> ) .....	75
2.2. Le langage Java Script .....	75
2.3. Les langages PHP et MYSQL.....	76
2.3.1. Le langage PHP ( <i>HypertextPreprocessor</i> ).....	76
2.3.2. MYSQL ( <i>StructuredQueryLanguage</i> ) .....	76
2.3.3. Côté-client et côté-serveur .....	76
<b>3. Les outils de développement et de modélisation .....</b>	<b>78</b>
3.1. StarUML .....	78
3.2. WampServer .....	78
3.3. Adobe Photoshop CC .....	79
3.3. Adobe Dreamweaver CC .....	80
<b>4. Le passage du diagramme de classe au modèle relationnel .....</b>	<b>81</b>
<b>5. Implémentation de la base de données .....</b>	<b>82</b>
<b>6. Carte de site.....</b>	<b>83</b>
<b>7. Présentation de quelques interfaces de l'application .....</b>	<b>84</b>
7.1- Page d'accueil .....	84
7.2- Page de service .....	85
7.3- L'interface du cas d'utilisation « Tester la disponibilité des RDV ».....	86
7.4- L'interface du cas d'utilisation « S'inscrire » .....	87
7.5- L'interface du cas d'utilisation « S'identifier » .....	87
7.6- L'interface du cas d'utilisation « M-A-J RDV » ( <i>patient</i> ).....	88
7.7- L'interface du cas d'utilisation « Modifier un RDV » ( <i>patient</i> ) .....	89
7.8- L'interfaces du cas d'utilisation « Annuler un RDV »( <i>patient</i> ).....	90
7.9- L'interface du cas d'utilisation «Rechercher un RDV» ( <i>Agent administratif</i> ) .....	90
7.10- L'interface du cas d'utilisation « Rechercher un patient »( <i>Agent administratif</i> ).....	91
7.11- L'interface du cas d'utilisation «RDV de patient » ( <i>Agent administratif</i> ).....	92
7.12- L'interface du cas d'utilisation «M-A-J service »( <i>Agent administratif</i> ).....	93
7.13- L'interface du cas d'utilisation « Modifier service » ( <i>Agent administratif</i> ) .....	93
7.14- L'interface du cas d'utilisation « Annuler service » ( <i>Agent administratif</i> ).....	94
<b>8. Conclusion .....</b>	<b>94</b>
<b>Conclusion générale.....</b>	<b>95</b>

# Liste des figures

## Chapitre I : L'internet et les applications web

Figure I.1. Internet.....	3
Figure I.2. les trois couches du web.....	6
Figure I.3. Transferts avec un site statique.....	7
Figure I.4. Transferts avec un site dynamique.....	8
Figure I.5. Le modèle client-serveur.....	9
Figure I.6. Système Client / Serveur.....	10

## Chapitre II : L'UML et le processus unifié UP

Figure II.1. Historique d'UML.....	12
Figure II.2. L'itération dans l'UP.....	14
Figure II.3. Les différentes perspectives définissant un modèle d'architecture.....	15
Figure II.4. Schéma d'ensemble d'UP.....	15
Figure II.5. Diagramme de cas d'utilisation.....	16
Figure II.6. Quelle méthode pour passer de l'expression des besoins au code de l'application ? .....	16
Figure II.7. Les besoins modélisés par un diagramme de cas d'utilisation.....	17
Figure II.8. Les diagrammes de séquence système illustrant la description textuelle des cas d'utilisations.....	17
Figure II.9. Une maquette d'IHM facilitant les discussions avec les futurs utilisateurs.....	18
Figure II.10. La phase d'analyse du domaine permettant d'élaborer la première version du diagramme de classes.....	19
Figure II.11. Le diagramme de classes participantes effectuant la jonction entre les cas d'utilisation, le modèle du domaine et les diagrammes de conception logicielle.....	20
Figure II.12. Les diagrammes d'activités de navigation représentant graphiquement l'activité de navigation dans l'IHM.....	21
Figure II.13. Les diagrammes d'interaction permettant d'attribuer précisément les responsabilités de comportement aux classes d'analyse.....	22
Figure II.14. Chaîne complète de la démarche de modélisation du besoin jusqu'au code.....	22

## Chapitre III : Identification des besoins

Figure III.1. Diagramme de cas d'utilisation.....	25
Figure III.2. Diagramme de séquence système «S'identifier».....	26
Figure III.3. Diagramme d'activité de navigation «S'identifier».....	27
Figure III.4. Diagramme de séquence système «S'inscrire».....	28
Figure III.5. Diagramme d'activité de navigation « S'inscrire».....	29
Figure III.6. Diagramme de séquence système «Visiter le site».....	30
Figure III.7. Diagramme d'activité de navigation « Visiter le site».....	31
Figure III.8. Diagramme de séquence système «Tester la disponibilité des RDV».....	32
Figure III.9. Diagramme d'activité de navigation «Tester la disponibilité des RDV».....	33
Figure III.10. Diagramme de séquence système «Choisir un RDV».....	34
Figure III.11. Diagramme d'activité de navigation «Choisir un RDV».....	35
Figure III.12. Diagramme de séquence système «Modifier un RDV».....	36

Figure III.13. Diagramme d'activité de navigation «Modifier un RDV».	37
Figure III.14. Diagramme de séquence système «Annuler un RDV».	38
Figure III.15. Diagramme d'activité de navigation «Annuler un RDV».	39
Figure III.16. Diagramme de séquence système «Ajouter un service».	40
Figure III.17. Diagramme d'activité de navigation «Ajouter un service».	41
Figure III.18. Diagramme de séquence système «Modifier un service».	42
Figure III.29. Diagramme d'activité de navigation «Modifier un service».	43
Figure III.20. Diagramme de séquence système «Annuler un service».	44
Figure III.21. Diagramme d'activité de navigation «Annuler un service».	45
Figure III.22. Diagramme de séquence système «Rechercher un RDV».	46
Figure III.23. Diagramme d'activité de navigation «Rechercher un RDV».	47
Figure III.24. Diagramme de séquence système «Rechercher un patient».	48
Figure III.25. Diagramme d'activité de navigation «Rechercher un patient».	49

## Chapitre IV : Analyse du domaine

Figure IV.1. Modèle de domaine.	50
Figure IV.2. Diagramme de classe participante « S'identifier »	50
Figure IV.3. Diagramme de classe participante « S'inscrire »	51
Figure IV.4. Diagramme de classe participante « Visiter le site »	51
Figure IV.5. Diagramme de classe participante « Tester la disponibilité des RDV »	51
Figure IV.6. Diagramme de classe participante « Choisir un RDV »	52
Figure IV.7. Diagramme de classe participante « Modifier un RDV »	53
Figure IV.8. Diagramme de classe participante « Annuler un RDV »	53
Figure IV.9. Diagramme de classe participante « Ajouter un service »	54
Figure IV.10. Diagramme de classe participante « Modifier un service »	54
Figure IV.11. Diagramme de classe participante « Annuler un service »	55
Figure IV.12. Diagramme de classe participante « Rechercher un RDV »	55
Figure IV.13. Diagramme de classe participante « Rechercher un Patient »	56

## Chapitre V : Phase de conception

Figure V.1. Diagramme d'interaction« S'identifier »	57
Figure V.2. Diagramme d'interaction« S'inscrire »	58
Figure V.3 Diagramme d'interaction« Visiter le site »	58
Figure V.4 Diagramme d'interaction« Tester la disponibilité des RDV »	59
Figure V.5. Diagramme d'interaction« Choisir un RDV »	59
Figure V.6. Diagramme d'interaction« Modifier un RDV »	60
Figure V.7. Diagramme d'interaction« Annuler un RDV »	61
Figure V.8. Diagramme d'interaction« Ajouter un service »	62
Figure V.9. Diagramme d'interaction« Modifier un service »	63
Figure V.10. Diagramme d'interaction« Annuler un service »	64
Figure V.11. Diagramme d'interaction« Rechercher un RDV »	65
Figure V.12. Diagramme d'interaction« Rechercher un Patient »	65
Figure V.13. Diagramme de classe de conception « S'identifier »	66
Figure V.14. Diagramme de classe de conception « S'inscrire »	66
Figure V.15. Diagramme de classe de conception « Visiter le site »	67

Figure V.16. Diagramme de classe de conception « Tester la disponibilité des RDV ».....	67
Figure V.17. Diagramme de classe de conception« Choisir un RDV » .....	68
Figure V.18. Diagramme de classe de conception« Modifier un RDV ».....	69
Figure V.19. Diagramme de classe de conception « Annuler un RDV ».....	70
Figure V.20. Diagramme de classe de conception« Ajouter un service » .....	70
Figure V.21. Diagramme de classe de conception« Modifier un service ».....	71
Figure V.22. Diagramme de classe de conception« Annuler un service » .....	71
Figure V.23. Diagramme de classe de conception« Rechercher un RDV ».....	72
Figure V.24. Diagramme de classe de conception« Rechercher un Patient » .....	72
Figure V.25. Diagramme de classe.....	73

## Chapitre VI : L'implémentation

Figure VI.1. Langages HTML et CSS.....	74
Figure VI.2. Résultat visible à l'écran.....	75
Figure VI.3. Le Côté-client. ....	76
Figure VI.4. Le côté-serveur. ....	77
Figure VI.5. StarUML. ....	78
Figure VI.6. WampServer. ....	79
Figure VI.7. Adobe Photoshop CC.....	80
Figure VI.8. Adobe Dreamweaver CC. ....	80
Figure VI.9. PHP MyAdmin. ....	82
Figure VI.10. Carte de site.....	83
Figure VI.11. L'interface « Accueil » .....	84
Figure VI.12. L'interface « Service ».....	85
Figure VI.13. L'interface « Tester la disponibilité des RDV » .....	86
Figure VI.14. L'interface « S'inscrire » .....	87
Figure VI.15. L'interface « S'identifier ».....	87
Figure VI.16. L'interface « M-A-J RDV ».....	88
Figure VI.17. L'interface « Modifier un RDV » .....	89
Figure VI.18. L'interface « Annuler un RDV » .....	90
Figure VI.19. L'interface « Rechercher un RDV » .....	90
Figure VI.20. L'interface « Rechercher un patient ».....	91
Figure VI.21. L'interface «RDV du patient » .....	92
Figure VI.22. L'interface « M-A-J service ».....	93
Figure VI.23. L'interface « Modifier service » .....	93
Figure VI.24. L'interface « Annuler service » .....	94

# Introduction générale

Ces dernières années ont été marquées par l'essor phénoménal de ce média qu'on appelle Internet, il est indéniable de constater que ce nouvel outil a d'ores et déjà profondément modifié la manière dont beaucoup de gens communiquent entre eux, l'Internet a aussi renforcé la capacité des administrations, par conséquent des entreprises et des individus à échanger des informations très rapidement. En utilisant l'Internet, on peut diffuser rapidement, efficacement et économiquement des informations ou des documents sous une forme électronique. Il permet ainsi une prise de décision rapide, et une actualisation rapide des informations, en particulier en situation de crise. C'est pourquoi les sociétés sont particulièrement prises de la capacité offerte par l'Internet afin de devenir plus compétitives. De plus les nouvelles technologies de la communication permettent d'établir une relation plus étroite entre les individus et les sociétés. Le site web est donc une vitrine pour les différentes sociétés à travers le monde, il permet de promouvoir l'activité de ces sociétés. En étant présente sur le web la société se fait de la publicité en se faisant connaître auprès de sa future clientèle et s'assure ainsi une plus grande facilité de prise de contact en attirant plus de clients, et favorisant le dialogue avec eux.

Les cliniques médicales font partie intégrante des établissements nécessitant une aide informatique, car jusqu'à ce jour, la prise d'un rendez-vous pour une consultation générale ou spécialisée ou encore pour faire des analyses médicales constitue un vrai cauchemar pour le patient. C'est pourquoi, l'informatisation de tels secteurs s'avère une nécessité absolue.

L'objectif de ce projet est la réalisation d'un site web dynamique pour une clinique médicale, permettant la prise et la gestion des rendez-vous pour effectuer des consultations générales ou spécialisées.

Ce mémoire est constitué de 6 chapitres :

**Chapitre I :** Dans ce chapitre nous définissons quelques concepts de l'Internet, le World Wide Web et les différents services d'Internet.

**Chapitre II :** Dans ce chapitre on propose une présentation générale d'UML et d'UP simplifié, ainsi que leurs objectifs et leurs domaines d'application.

**Chapitre III :** Ce chapitre est consacré pour identifier tous les besoins d'utilisateurs.

**Chapitre IV :** Le chapitre de la phase d'analyse du domaine permet d'élaborer le modèle de domaine. Puis, Nous développerons les diagrammes de classes participantes.

**Chapitre V :** Ce chapitre traite la conception du site en suivant le processus de développement UP simplifié.

**Chapitre VI :** Dans ce chapitre, nous allons citer la liste des langages et les outils utilisés pour la réalisation du site tout en expliquant le rôle de chacun.

Nous terminerons ce mémoire par une conclusion générale.

# Chapitre I : L'internet et les applications web



## 1. Introduction

Dans ce chapitre, nous allons présenter d'une façon générale le réseau internet et le web.

Autrement dit, ce chapitre vise à répondre aux questions suivant :

Qu'est-ce qu'un réseau Internet ?

Qu'est-ce qu'un Web ?

Qu'est-ce qu'un modèle client-serveur ?

## 2. Le réseau Internet

### 2.1. Définition

Internet est l'ensemble des nœuds (connexions, câbles, etc.) entre les machines qui nous donnent accès au web. L'Internet est, avant tout, une collection de millions de réseaux. Personne n'est propriétaire de l'Internet dans son collection. Ces réseaux nous permettent de partager des données sur la toile (entre autres, car il n'existe pas que le web sur Internet). Donc, quand une personne vous demande si vous avez Internet, elle veut savoir si votre ordinateur a accès à Internet. Par ailleurs c'est encore un abus de langage que de dire que l'on a Internet : ce réseau gigantesque n'appartenant à personne, on ne peut qu'avoir accès à Internet. [1][2]



**Figure I.1. Internet.**

### 2.2. Historique

Les premières versions mettant en place ces idées apparurent à la fin des années 1950. L'application pratique de ces concepts commença à la fin des années 1960. Dès les années 1980, les techniques que nous reconnaissons maintenant comme les fondements de l'Internet moderne commencèrent à se répandre autour du globe. Dans les années 1990, sa popularisation passa par l'apparition du World Wide Web.

L'infrastructure d'Internet se répandit autour du monde pour créer le large réseau mondial d'ordinateurs que nous connaissons aujourd'hui. Il se répandit au travers des pays occidentaux puis frappa à la porte des pays en voie de développement, créant ainsi un accès mondial à l'information et aux communications sans précédent ainsi qu'une fracture numérique. Internet contribua à modifier fondamentalement l'économie mondiale, y compris avec les retombées de la bulle Internet. [3]

### **2.3. Le rôle de l'Internet**

Aujourd'hui, la science et la technologie se développent rapidement. L'internet est une application des technologies nouvelles. Il joue un rôle très important dans la science comme dans la vie quotidienne.

En effet, actuellement. L'internet est en train d'envahir dans notre vie quotidienne et est devenu un phénomène mondial. Depuis l'apparition de l'internet, la distance semble réduite et les hommes peuvent se communiquer facilement. Il nous aide encore à échanger des lettres. Au lieu d'écrire une lettre à notre ami, nous lui envoyons le courrier électronique ou le message grâce à l'internet. Cela est rapide et moins cher. De plus, à la maison, nous pouvons réserver les billets d'avion, de train, les places de théâtre ... et savoir encore tout ce qui se passe dans le monde.

Quant à l'éducation, l'internet nous aide beaucoup dans l'enseignement et dans l'apprentissage. Les élèves se passent d'aller à l'école parce qu'ils étudient grâce aux programmes éducatifs sur l'internet. Il est considéré comme un professeur dévoué qui répond à toutes nos questions. De plus, il y a aussi des formations nécessaires pour servir leurs études grâce aux pages web.

D'ailleurs, plus l'économie se développe, plus on a besoin d'échanges des informations. Alors, le rôle de l'internet est très important non seulement dans l'éducation mais encore dans l'économie. Il aide les entreprises à mettre en relation permanente tous les services, les employés à classer les dossiers, les cadres peuvent discuter facilement de leur travail ou échanges des documents. De plus, il réalise aussi des opérations d'achat et de vente et annonce encore des informations du marché auxquelles beaucoup d'investisseurs s'intéressent. A part ces intérêts, l'internet est aussi un outil de distraction. On peut regarder les films, faire des jeux vidéo. Donc, les services que l'internet nous rend sont très variés et très utiles.

## 2.4. Les services d'Internet

Nous avons vu qu'Internet s'appuyait sur le protocole TCP/IP. Il existe de nombreux autres protocoles en usage sur ce réseau, en particulier, les principaux services offerts par Internet qui sont : **le web, la messagerie électronique (e-mail) les forums, le chat et le FTP**, utilisent chacun un protocole différent. Ainsi, il existe :

**HTTP (HyperText Transfert Protocol) :** Visualisation de pages Web.

**FTP (File Transfer Protocol) :** Transferts de fichiers.

**SMTP (Simple Mail Transfert Protocol) :** Transferts de courriers électroniques (e-mail).

**POP (Post Office Protocol) :** Récupération des e-mails sur le serveur.

**NNTP (Network News Transfert Protocol) :** Forums de discussions en temps différés.

**IRC (Internet Relay Chat) :** Dialogue en temps réel. [2]

## 2.5. L'architecture de l'Internet

Le réseau Internet – réseau de réseaux – est composé de deux architectures distinctes :

- l'«architecture physique» d'Internet qu'est le réseau lui-même. Il peut s'agir d'un réseau filaire (réseau téléphonique commuté, le câble, désormais la fibre optique) ou hertzien (Internet par satellite, 3G, Wifi, Wimax) ;
- l'«architecture logique» d'Internet, composée des protocoles et standards de communication permettant le transport de données sur les réseaux, tels que l'Internet Protocol (IP), le Transmission Control Protocol (TCP) ou l'HyperText Transfer Protocol (HTTP) sur lequel est basé le Web.

## 3. Le World Wide Web

### 3.1. Définition

Le World Wide Web, ou Web pour faire plus court, est l'ensemble des sites Web (appelés par abus de langage "sites Internet") présents sur le réseau Internet. La toile, comme on dit parfois en français, c'est donc l'ensemble de tous les sites Web que nous pouvons visiter grâce à notre navigateur Web (Firefox, Opéra, ...). [1]

### 3.2. Le principe du Web

Le principe de web repose sur l'utilisation de hyperliens pour naviguer entre des documents (appelés «pages web») grâce à un logiciel appelé navigateur (parfois également appelé fureteur ou butineur ou en anglais browser). Une page web est ainsi un simple fichier texte écrit dans un langage de description (appelé HTML), permettant de décrire la mise en page du document et d'inclure des éléments graphiques ou bien des liens vers d'autres documents à l'aide de balises. [3]

### 3.3. Les niveaux du Web

L'architecture logique du système est divisée en trois niveaux ou couches :

- La présentation des données : correspondant à l'affichage, la restitution sur le poste de travail, le dialogue avec l'utilisateur.
- Le traitement métier des données : correspondant à la mise en œuvre de l'ensemble des règles de gestion et de la logique applicative.
- L'accès aux données persistantes : correspondant aux données qui sont destinées à être conservées sur la durée, voire de manière définitive. [4]

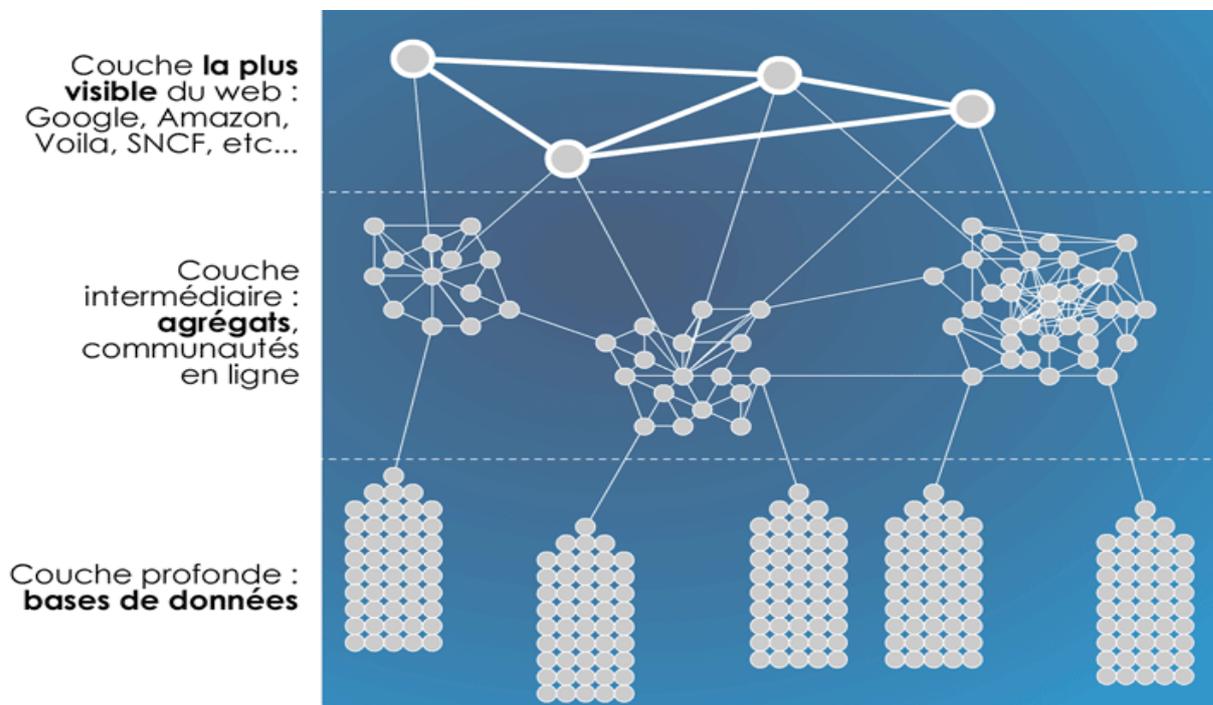


Figure I.2. Les trois couches du web.

### 3.4. Les sites web

Un site web est un ensemble de fichiers HTML stockés sur un ordinateur connecté en permanence à internet et hébergeant les pages web (serveur web).

On considère qu'il existe deux types de sites web : les sites **statiques** et les sites **dynamiques**.

#### 3.4.1. Les sites statiques

Ce sont des sites réalisés uniquement à l'aide des langages (X) HTML et CSS. Ils fonctionnent très bien mais leur contenu ne peut pas être mis à jour automatiquement : il faut que le propriétaire du site (le webmaster) modifie le code source pour y ajouter des nouveautés. Ce n'est pas très pratique quand on doit mettre à jour notre site plusieurs fois dans la même journée, Les sites statiques sont donc bien adaptés pour réaliser des sites "vitrine", pour présenter par exemple notre entreprise, mais sans aller plus loin. Ce type de site se fait de plus en plus rare aujourd'hui, car dès que l'on rajoute un élément d'interaction (comme un formulaire de contact), on ne parle plus de site statique mais de site dynamique. [5]

Un site Web statique est un site où chacune des pages est créée en HTML. L'ordinateur qui se connecte au serveur, demande une page. Celle-ci lui est directement servie (elle est stockée toute prête sur le serveur). [6]

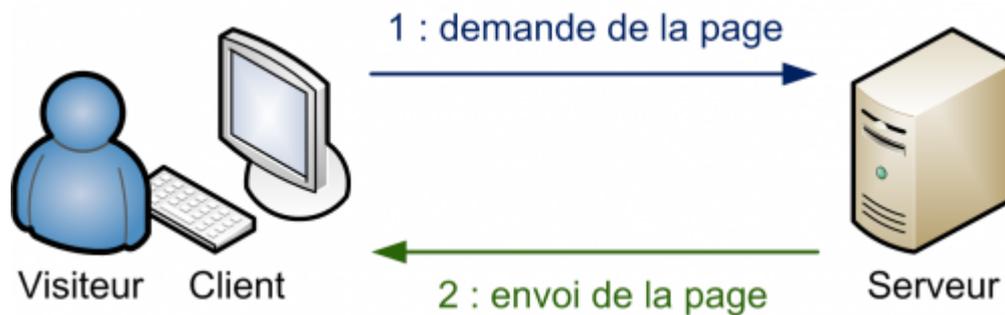


Figure I.3. Transferts avec un site statique.

#### 3.4.2. Les sites dynamiques

Ils utilisent d'autres langages en plus de (X) HTML et CSS, tels que PHP et MySQL. Le contenu de ces sites web est dit "dynamique" parce qu'il peut changer sans l'intervention du webmaster, La plupart des sites web que nous visitons aujourd'hui, sont des sites dynamiques. [5]

Un site Web dynamique est un site Web dont les pages sont générées dynamiquement à la demande. Le contenu est obtenu (par exemple) en combinant l'utilisation d'un langage de scripts ou de programmation et une base de données. Il s'agit souvent de PHP pour le langage et MySQL pour la base de données. [6]



Figure I.4. Transferts avec un site dynamique.

## 4. Le modèle client-serveur

### 4.1. Description

De nombreuses applications fonctionnent selon un environnement client/serveur, cela signifie que des machines clientes (des machines faisant partie du réseau) contactent un serveur, une machine généralement très puissante en termes de capacités d'entrée-sortie, qui leur fournit des services. Ces services sont des programmes fournissant des données telles que l'heure, des fichiers, une connexion, etc.

Les services sont exploités par des programmes, appelés programmes clients, s'exécutant sur les machines clientes. On parle ainsi de client (client FTP, client de messagerie, etc.) lorsque l'on désigne un programme tournant sur une machine cliente, capable de traiter des informations qu'il récupère auprès d'un serveur (dans le cas du client FTP il s'agit de fichiers, tandis que pour le client de messagerie il s'agit de courrier électronique). [7]

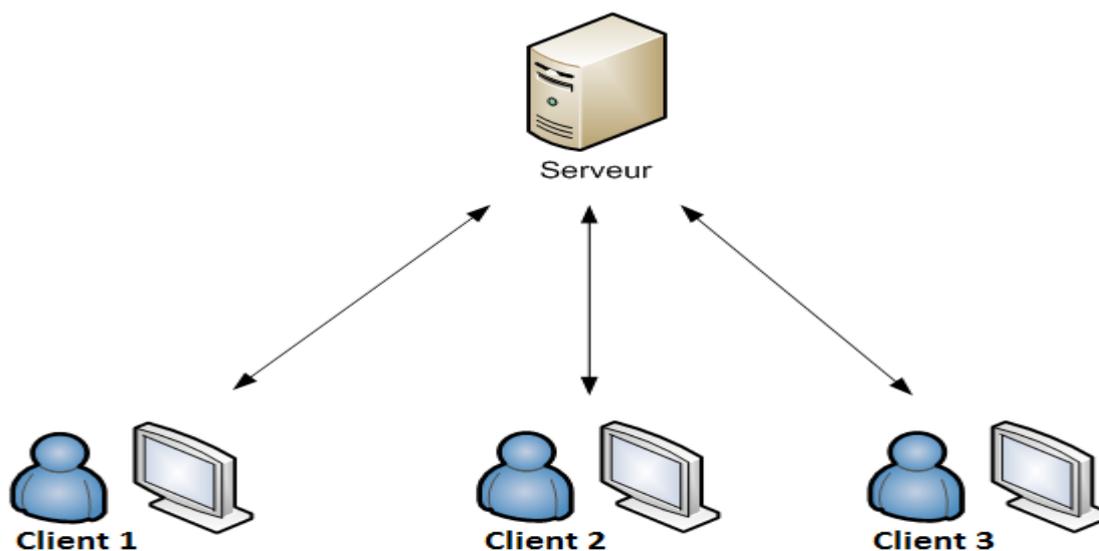


Figure I.5. Le modèle client-serveur.

## 4.2. Avantages de l'architecture client/serveur

Le modèle client/serveur est particulièrement recommandé pour des réseaux nécessitant un grand niveau de fiabilité, ses principaux atouts sont :

- Des ressources centralisées : étant donné que le serveur est au centre du réseau, il peut gérer des ressources communes à tous les utilisateurs, comme par exemple une base de données centralisée, afin d'éviter les problèmes de redondance et de contradiction.
- Une meilleure sécurité : car le nombre de points d'entrée permettant l'accès aux données est moins important.
- Une administration au niveau serveur : les clients ayant peu d'importance dans ce modèle, ils ont moins besoin d'être administrés.
- Un réseau évolutif : grâce à cette architecture il est possible de supprimer ou rajouter des clients sans perturber le fonctionnement du réseau et sans modification majeure.

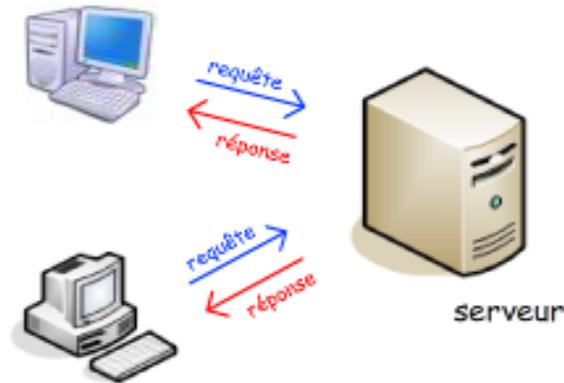
## 4.3. Inconvénients du modèle client/serveur

L'architecture client/serveur a tout de même quelques lacunes parmi lesquelles :

- Un coût élevé dû à la technicité du serveur.
- Un maillon faible : le serveur est le seul maillon faible du réseau client/serveur, étant donné que tout le réseau est architecturé autour de lui ! Heureusement, le serveur a une grande tolérance aux pannes (notamment grâce au système RAID).

#### 4.4.Fonctionnement d'un système client/serveur

Un système client/serveur fonctionne selon le schéma suivant :



**Figure I.6. Système Client / Serveur.**

Le client émet une requête vers le serveur grâce à son adresse IP et le port, qui désigne un service particulier du serveur. Le serveur reçoit la demande et répond à l'aide de l'adresse de la machine cliente et son port. [7]

#### 5. Conclusion

Après avoir détaillé quelques notions concernant l'Internet et le World Wide Web, ainsi que le modèle client-serveur, les sites web. Nous allons présenter dans le chapitre suivant le langage de modélisation que nous utiliserons dans la modélisation de notre système ainsi que la démarche suivie.

# Chapitre II : L'UML et le processus unifié UP



## 1. Introduction

Dans ce chapitre on propose une présentation générale d'UML, d'UP, ses objectifs, ses domaines d'application. On va parler des concepts de base d'UML. Principalement ce chapitre vise à répondre aux questions suivantes :

Qu'est qu'UML ?

Qu'est qu'UP ?

## 2. L'UML (*Unified Modeling Language*)

### 2.1. Définition

UML (*Unified Modeling Language*), que l'on peut traduire par "*langage de modélisation unifié*" est une notation permettant de modéliser un problème de façon standard. Ce langage est né de la fusion de plusieurs méthodes existant auparavant, et est devenu désormais la référence en terme de modélisation objet, à un tel point que sa connaissance est souvent nécessaire pour obtenir un poste de développeur objet. [8]

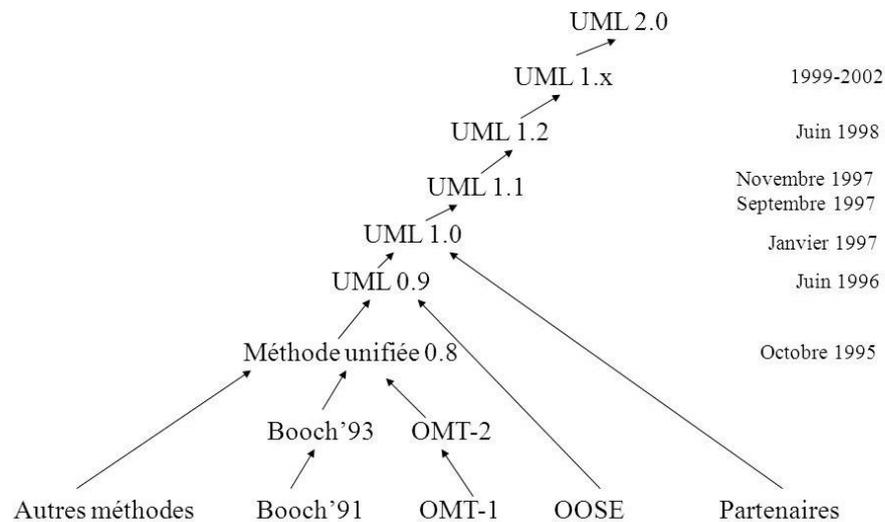
### 2.2. Les méthodes objets

La modélisation objet consiste à créer une représentation informatique des éléments du monde réel auxquels on s'intéresse, sans se préoccuper de l'implémentation, ce qui signifie indépendamment d'un langage de programmation. Il s'agit donc de déterminer les objets présents et d'isoler leurs données et les fonctions qui les utilisent. Pour cela des méthodes ont été mises au point. Entre 1970 et 1990, de nombreux analystes ont mis au point des approches orientées objets, si bien qu'en 1994 il existait plus de 50 méthodes objet. Toutefois seules 3 méthodes ont véritablement émergé :

- La méthode OMT de Rumbaugh.
- La méthode BOOCH'93 de Booch.
- La méthode OOSE de Jacobson (Object Oriented Software Engineering).

A partir de 1994, Rumbaugh et Booch (rejoints en 1995 par Jacobson) ont unis leurs efforts pour mettre au point la méthode unifiée (unified method 0.8), incorporant les avantages de chacune des méthodes précédentes.

La méthode unifiée à partir de la version 1.0 devient UML (Unified Modeling Language), une notation universelle pour la modélisation objet. [9]



**Figure II.1. Historique d'UML.**

Fin 1997, UML est devenu une norme OMG (Object Management Group). Il s'agit d'un organisme créé en 1989 à l'initiative de grandes sociétés (HP, Sun, Unisys, American Airlines, Philips...). La dernière version diffusée par l'OMG est UML 2.5 bêta 2 depuis septembre 2013. [10]

En l'espace d'une poignée d'années seulement, l'UML est devenu un standard incontournable à tel point qu'aujourd'hui, utiliser les technologies objet sans l'UML relève désormais de l'hérésie.

### 2.3. Intérêt d'une méthode objet

Les langages orientés objet constituent chacun une manière spécifique d'implémenter le paradigme objet. Ainsi, une méthode objet permet de définir le problème à haut niveau sans rentrer dans les spécificités d'un langage. Il représente ainsi un outil permettant de définir un problème de façon graphique, afin par exemple de le présenter à tous les acteurs d'un projet (n'étant pas forcément des experts en un langage de programmation).

De plus, le fait de programmer à l'aide d'un langage orienté objet ne fait pas d'un programmeur un concepteur objet. En effet il est tout à fait possible de produire un code syntaxiquement juste sans pour autant adopter une approche objet. Ainsi la programmation orientée objet implique

- en premier lieu une conception abstraite d'un modèle objet (c'est le rôle de la méthode objet)
- en second plan l'implémentation à l'aide d'un langage orienté objet (tel que C++/Java/...)

Une méthode objet est donc d'une part une méthode d'analyse du problème (afin de couvrir toutes les facettes du problème), d'autre part un langage permettant une représentation standard stricte des concepts abstraits (la modélisation) afin de constituer un langage commun. [9]

## 2.4. Les Diagrammes d'UML

L'UML représente un ensemble de schéma appelé diagramme, ils nous permettent de voir l'ensemble du programme avant de commencer à le coder, et ainsi il nous permet donc de ne pas faire, ou très peu, d'erreurs ou d'oublis.

Le mode de représentation statique ou structurel s'appuie sur les 7 diagrammes ci-après.

- Le diagramme de cas d'utilisation.
- Le diagramme de classes.
- Le diagramme de packages.
- Le diagramme d'objets.
- Le diagramme de structure.
- Le diagramme de composants.
- Le diagramme de déploiement. [11]

Le mode de représentation dynamique ou comportemental s'appuie sur les 6 diagrammes ci-après, dont 2 nouveaux diagrammes introduits par UML 2.0.

- Le diagramme d'états.
- Le diagramme d'activité.
- Les diagrammes de communication.
- Les diagrammes de séquence.
- Le diagramme global d'interactions.
- Le diagramme de temps. [11]

### 3. Le Processus Unifié (UP)

#### 3.1. Définition

Le processus unifié est un processus de développement logiciel itératif, centré sur l'architecture, piloté par des cas d'utilisation et orienté vers la diminution des risques. C'est un patron de processus pouvant être adapté à une large classe de systèmes logiciels, à différents domaines d'application, à différents types d'entreprises, à différents niveaux de compétences et à différentes tailles de l'entreprise. [9]

#### 3.2. Les principes d'UP

##### 3.2.1. UP itératif et incrémental

L'itération est une répétition d'une séquence d'instructions ou d'une partie de programme un nombre de fois fixé à l'avance ou tant qu'une condition définie n'est pas remplie, dans le but de reprendre un traitement sur des données différentes.

Elle qualifie un traitement ou une procédure qui exécute un groupe d'opérations de façon répétitive jusqu'à ce qu'une condition bien définie soit remplie. [9]

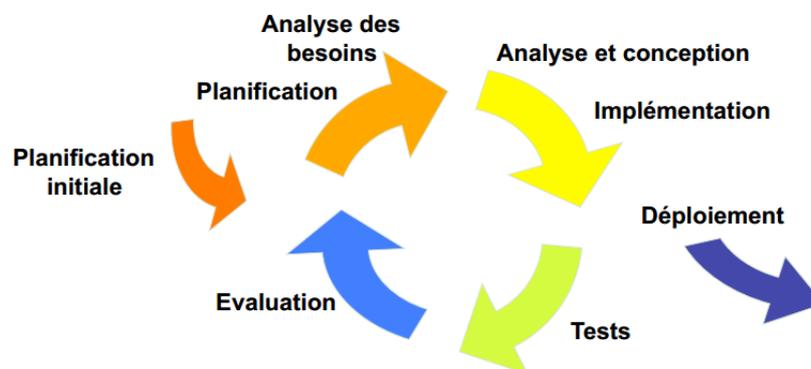
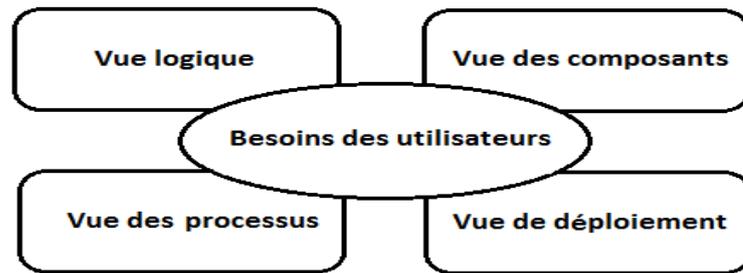


Figure II.2. L'itération dans l'UP.

##### 3.2.2. UP centré sur l'architecture

Une architecture adaptée est la clé de voûte du succès d'un développement. Elle décrit des choix stratégiques qui déterminent en grande partie les qualités du logiciel (adaptabilité, performances, fiabilité...).

Ph.Kruchten propose différentes perspectives, indépendantes et complémentaires, qui permettent de définir un modèle d'architecture. [9]

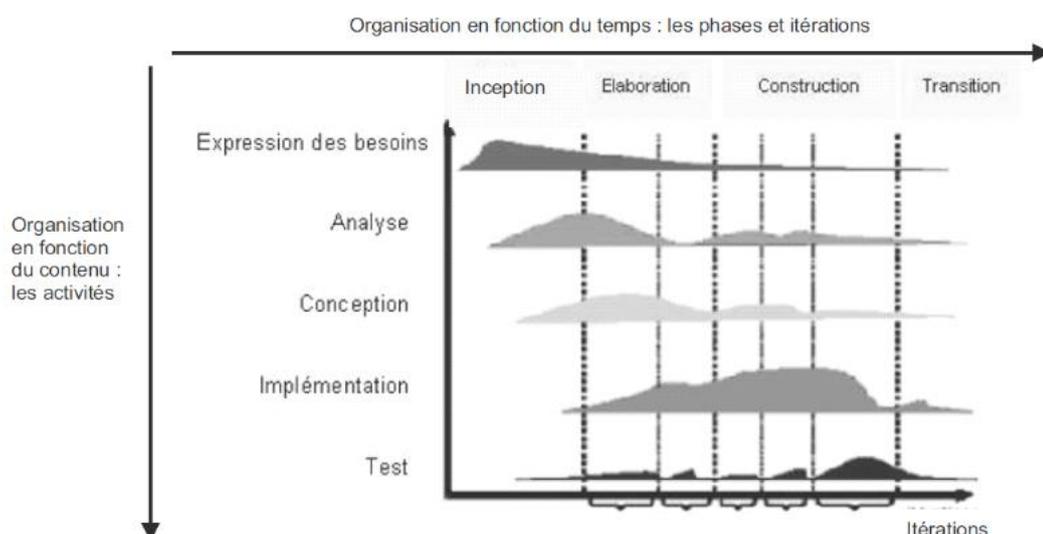


**Figure II.3. Les différentes perspectives qui définissent un modèle d'architecture.**

UP gère le processus de développement par deux axes.

**L'axe vertical** représente les principaux enchaînements d'activités, qui regroupent les activités selon leur nature. Cette dimension rend compte l'aspect statique du processus qui s'exprime en termes de composants, de processus, d'activités, d'enchaînements, d'artefacts et de travailleurs.

**L'axe horizontal** représente le temps et montre le déroulement du cycle de vie du processus, cette dimension rend compte de l'aspect dynamique du processus qui s'exprime en terme de cycles, de phases, d'itérations et de jalons.



**Figure II.4. Schéma d'ensemble d'UP.**

### 3.2.3. UP piloté par les cas d'utilisation d'UML

Le but principal d'un système informatique est de satisfaire les besoins du client. Le processus de développement sera donc accès sur l'utilisateur.

Les cas d'utilisation permettent d'illustrer ces besoins.

Ils détectent puis décrivent les besoins fonctionnels (du point de vue de l'utilisateur), et leur ensemble constitue le modèle de cas d'utilisation qui dicte les fonctionnalités complètes du système. [9]

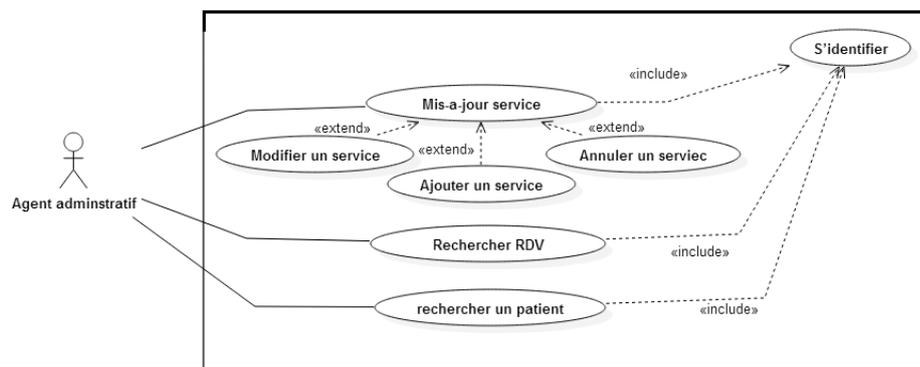


Figure II.5. Diagramme de cas d'utilisation.

## 4. Une méthode simple et générique

C'est une méthode qui se situe à mi-chemin entre UP (Unified Process), qui constitue un cadre général très complet de processus de développement, et XP (eXtreme Programming) qui est une approche minimaliste à la mode centrée sur le code [12].

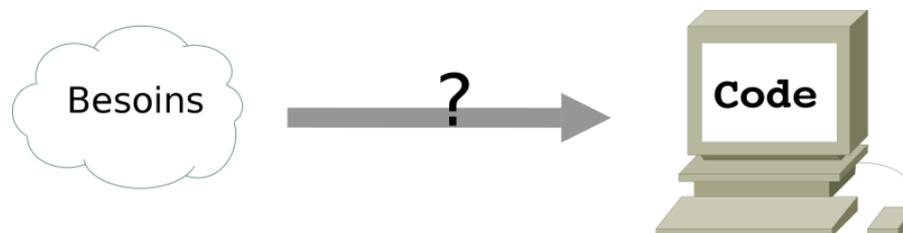


Figure II.6. Quelle méthode pour passer de l'expression des besoins au code de l'application ?

#### 4.1. Identification des besoins et spécification des fonctionnalités

- Identification et représentation des besoins : diagramme de cas d'utilisation.

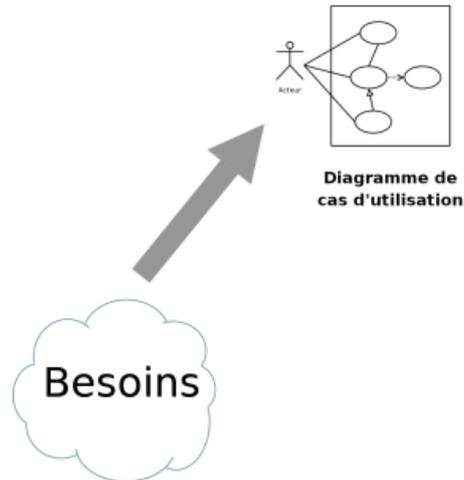


Figure II.7. Les besoins modélisés par un diagramme de cas d'utilisation. [8]

- Spécification détaillée des besoins : diagrammes de séquence système.

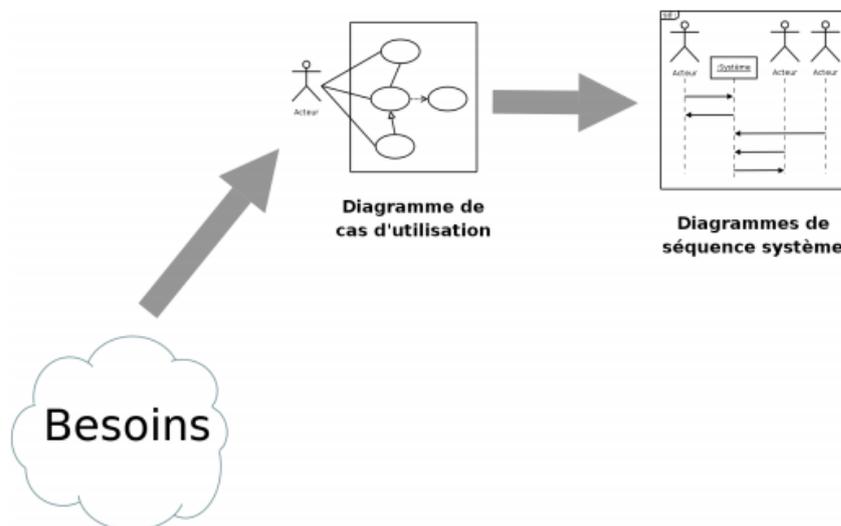
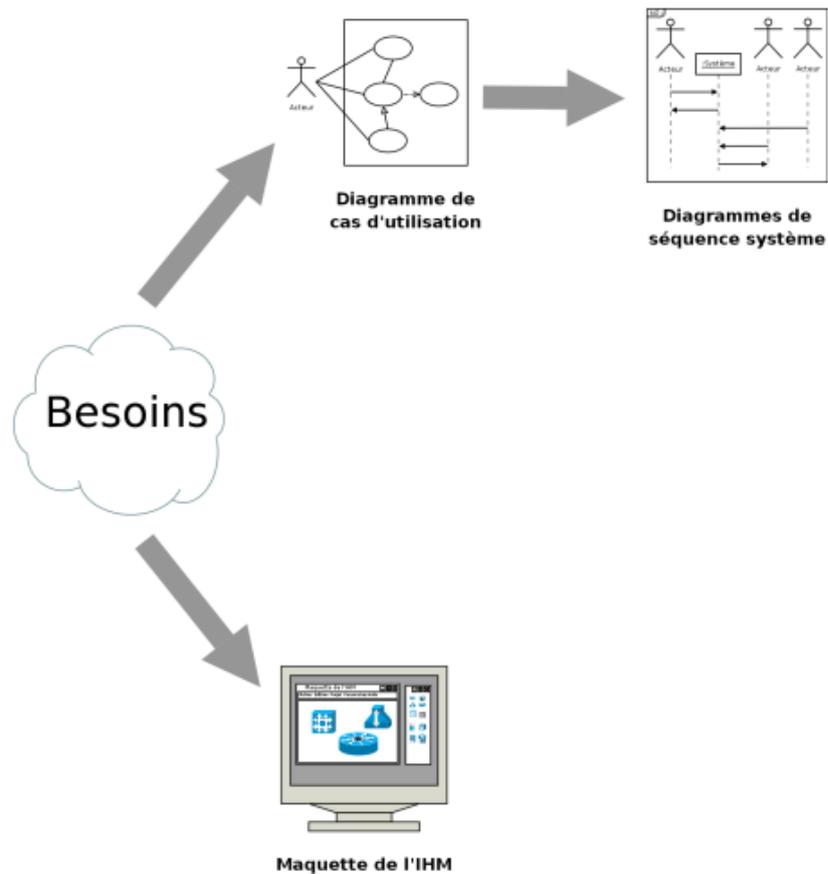


Figure II.8. Les diagrammes de séquence système illustrant la description textuelle des cas d'utilisations. [8]

➤ **Maquette de l'IHM de l'application (non couvert par UML).**



**Figure II.9. Une maquette d'IHM facilitant les discussions avec les futurs utilisateurs.**

[8]

## 4.2. Phase d'analyse

- **Analyse du domaine : modèle du domaine.**

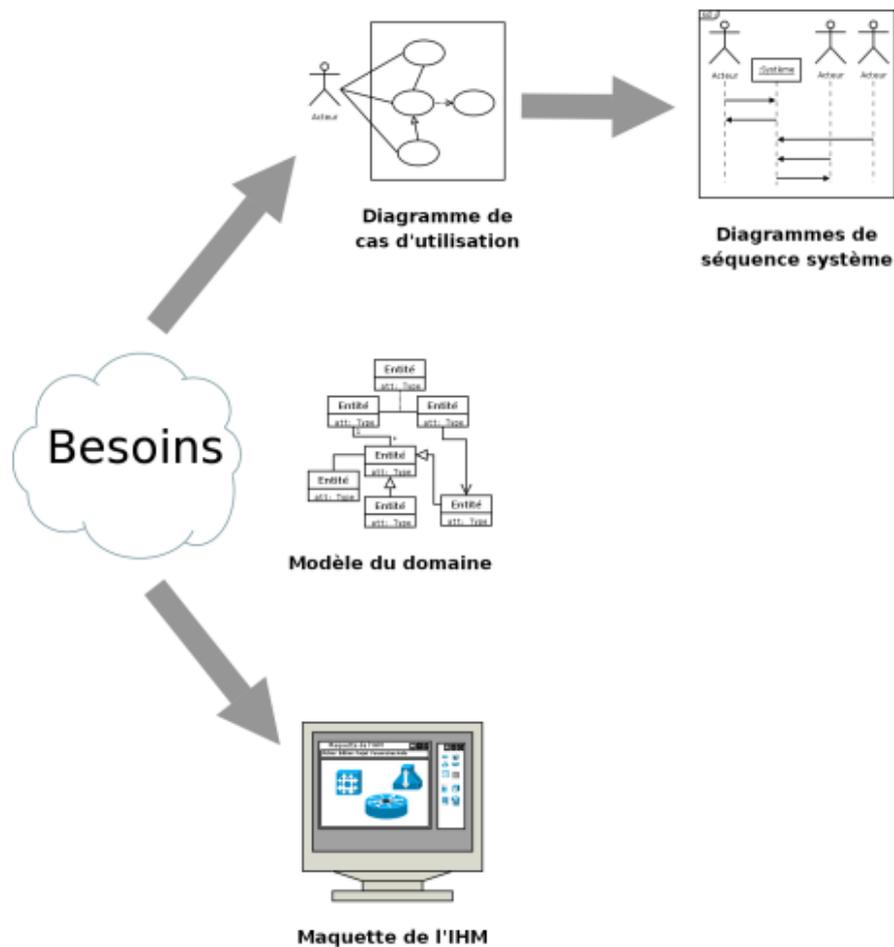
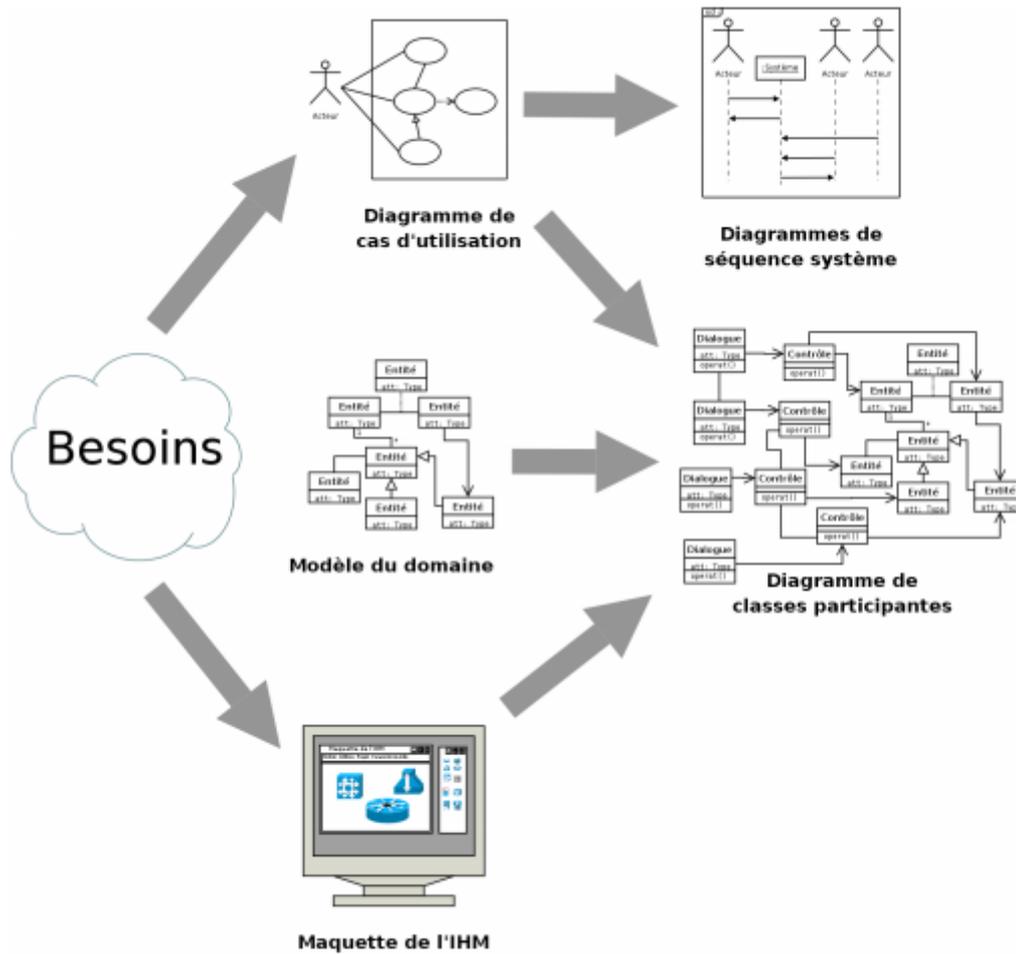


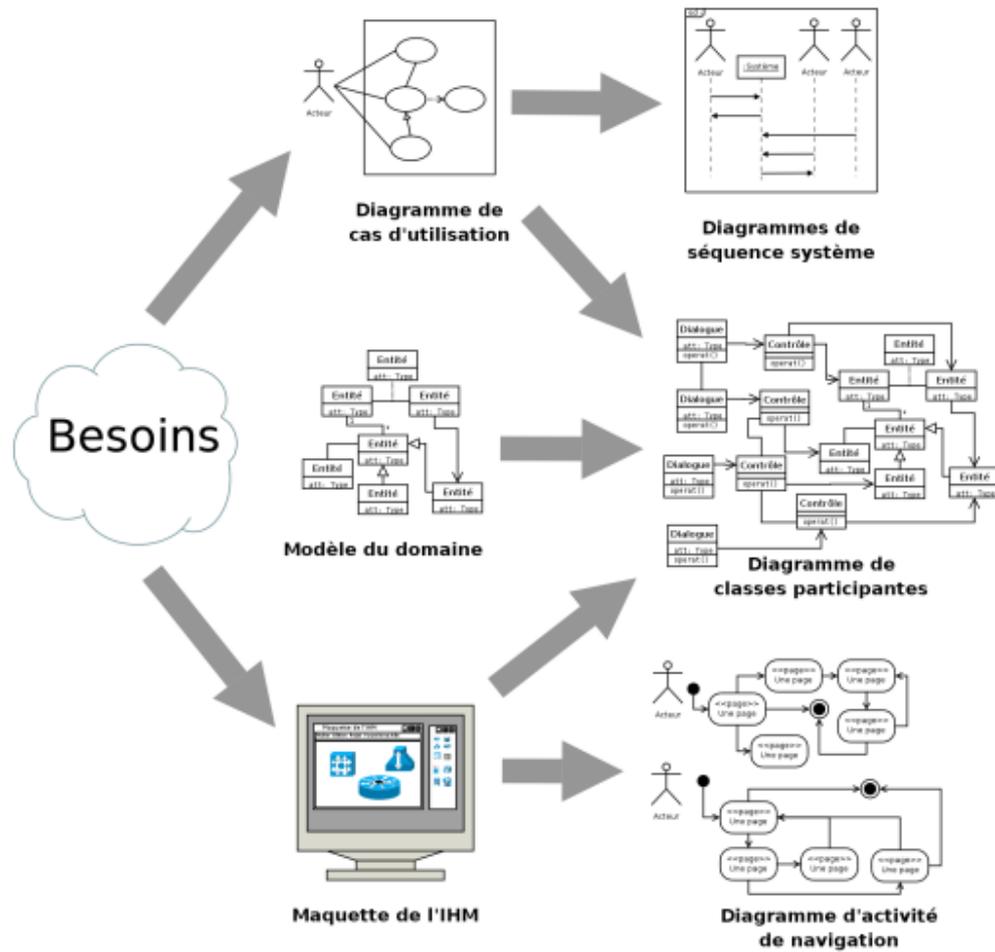
Figure II.10. La phase d'analyse du domaine permettant d'élaborer la première version du diagramme de classes. [8]

➤ **Diagramme de classes participantes.**



**Figure II.11. Le diagramme de classes participantes effectuant la jonction entre les cas d'utilisation, le modèle du domaine et les diagrammes de conception logicielle. [8]**

➤ **Diagramme d'activité de navigation.**



**Figure II.12. Les diagrammes d'activités de navigation représentant graphiquement l'activité de navigation dans l'IHM. [8]**

### 4.3. Phase de conception

➤ Diagramme d'interaction.

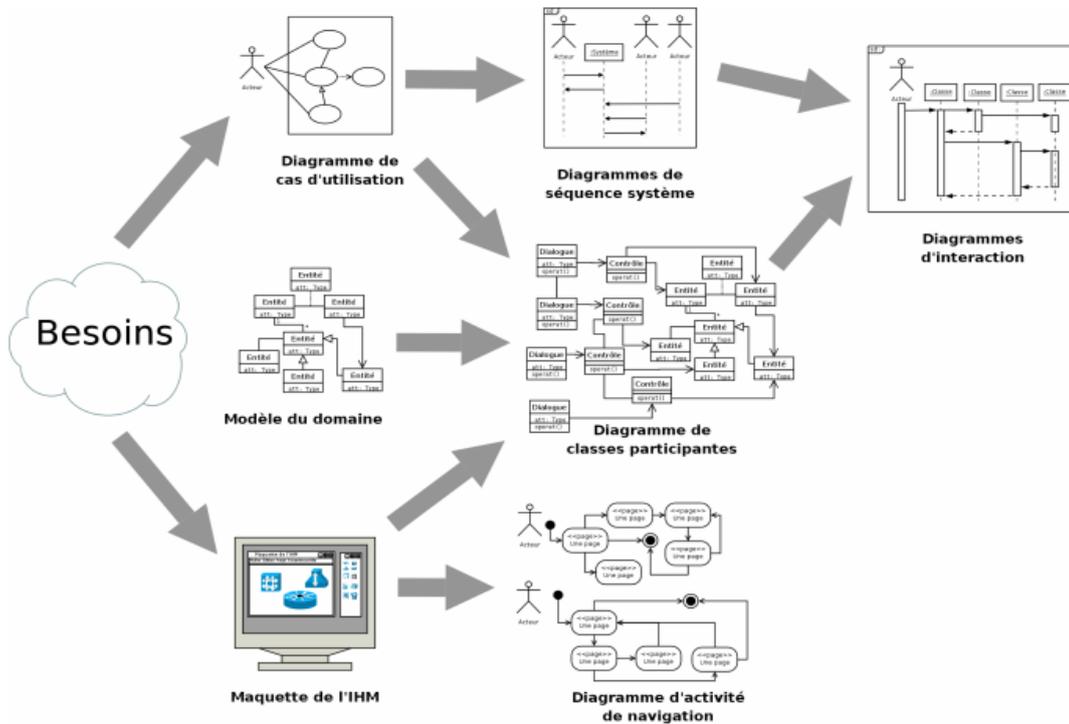


Figure II.13. Les diagrammes d'interaction permettant d'attribuer précisément les responsabilités de comportement aux classes d'analyse. [8]

➤ Diagramme de classe de conception.

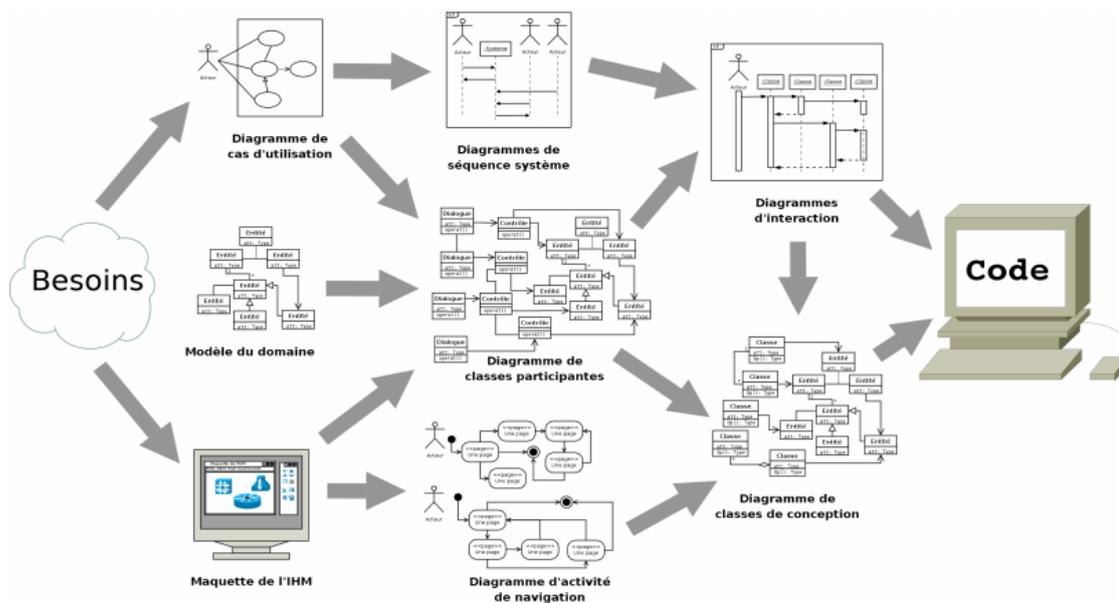


Figure II.14. Chaîne complète de la démarche de modélisation du besoin jusqu'au code. [8]

## **5. Conclusion**

UML est un moyen d'exprimer des modèles en faisant une abstraction de Leur implémentation. Il offre une manière de représenter le système selon différentes vues complémentaires grâce aux diagrammes. Pour une pertinente conception, UML doit suivre la démarche de développement UP simplifiée destinée pour les applications web.

# Chapitre III : Identification des besoins



## 1. Introduction

Après avoir bien étudié notre domaine et choisis le langage de modélisation et le processus de développement, nous allons entamer dans ce chapitre la première étape du Processus UP qui est l'identification des besoins. Nous commençons à identifier les acteurs qui interagiront avec le système, nous identifions et décrivons les cas d'utilisation du système. Nous allons présenter les diagrammes de séquences système, d'activité de navigation basant sur les scénarios de chaque cas d'utilisation.

## 2. Identification des acteurs

Un acteur est l'idéalisation d'un rôle joué par une personne externe, un processus ou une chose qui interagit avec un système.

Dans notre projet il existe trois acteurs qui sont :

**Le Visiteur** : la personne qui consulte le site et peut s'inscrire dans le site.

**Le Patient** : la personne qui consulte et participe dans le site pour prend un rendez-vous.

**Agent administratif** : la personne qui consulte et ajouter, modifier, ou supprimer des informations données.

## 3. Identification des cas d'utilisation

Pour l'acteur visiteur :

- ✚ Visiter le site.
- ✚ S'inscrire.
- ✚ Tester la disponibilité des rendez-vous.

Pour l'acteur patient :

- ✚ Visiter le site.
- ✚ Tester la disponibilité des rendez-vous.
- ✚ S'identifier.
- ✚ Prendre un rendez-vous.
- ✚ Modifier un rendez-vous.
- ✚ Annuler un rendez-vous.

Pour l'acteur agent administratif :

- ✚ S'identifier.
- ✚ Ajouter un service.

- ✚ Annuler un service.
- ✚ Modifier un service.
- ✚ Rechercher des rendez-vous.
- ✚ Rechercher un patient.

#### 4. Diagramme de cas d'utilisation

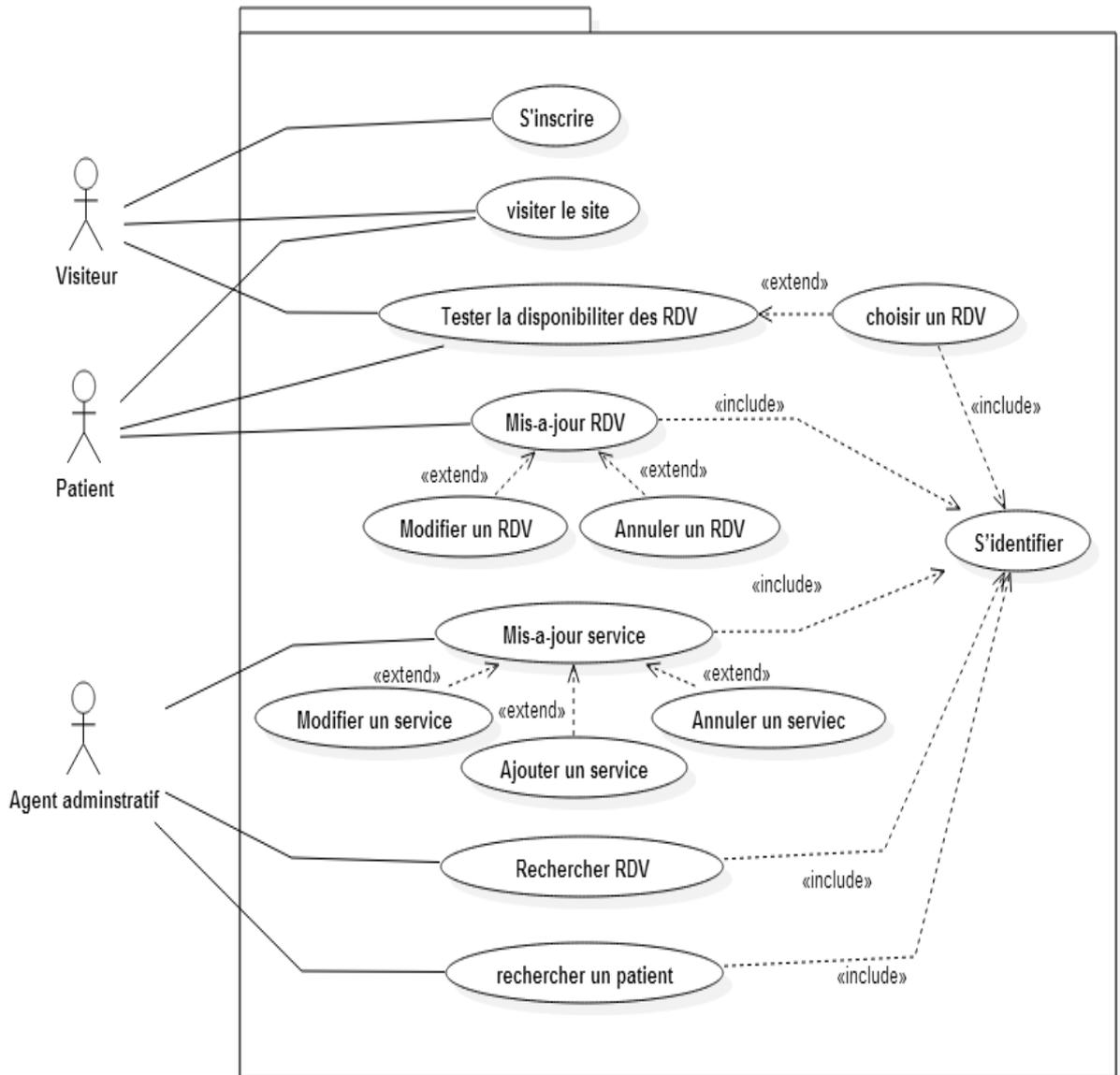


Figure III.1. Diagramme de cas d'utilisation.

## 5. Les fiches descriptives

### 5.1- Fiche descriptive du cas d'utilisation « S'identifier »

S'identifier	
<b>Description</b>	Ce cas permet de vérifier l'identité de l'agent administratif et le patient.
<b>Acteur</b>	Agent administratif, Patient.
<b>Pré condition</b>	Chaque utilisateur possède un compte.
<b>Post condition</b>	L'utilisateur peut accéder à son propre espace.
<b>Nominal</b>	<ol style="list-style-type: none"> <li>1. L'utilisateur entre ses informations et les valide.</li> <li>2. Le système valide l'opération.</li> </ol>
<b>Alternatif</b>	Les informations erronées (retour au scénario nominal 1).
<b>Exception</b>	L'utilisateur quitte le système.

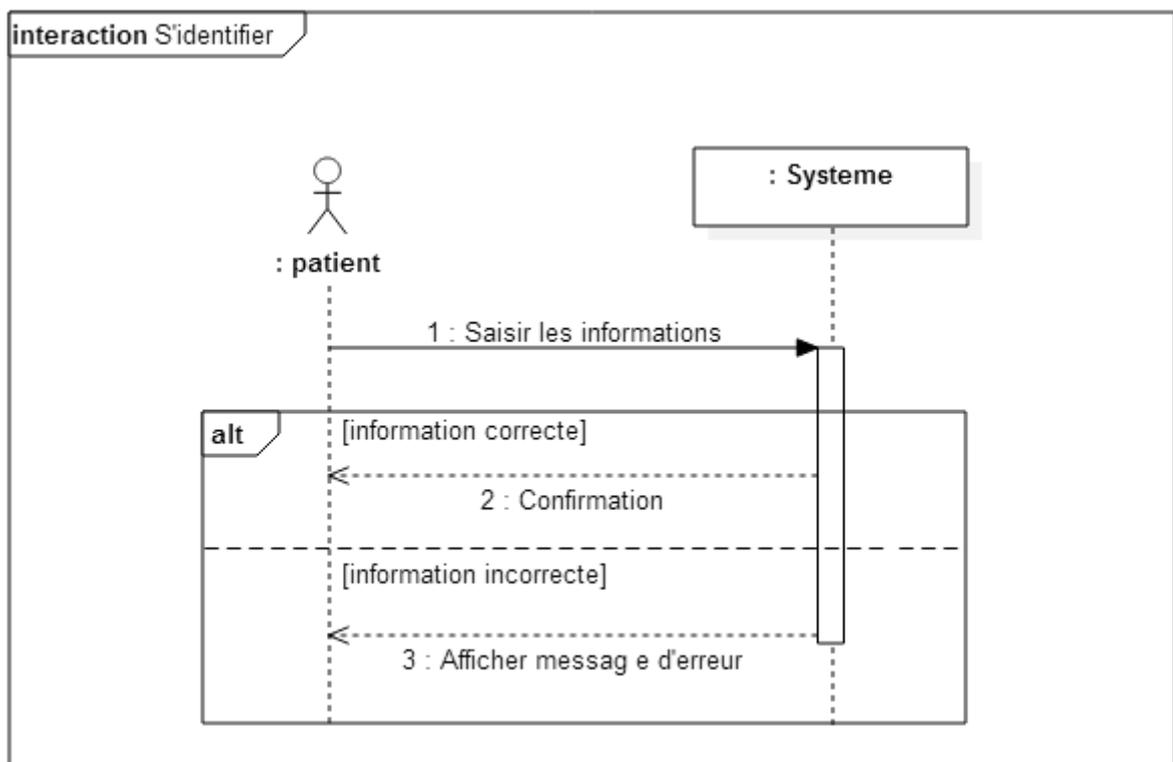


Figure III.2. Diagramme de séquence système «S'identifier ».

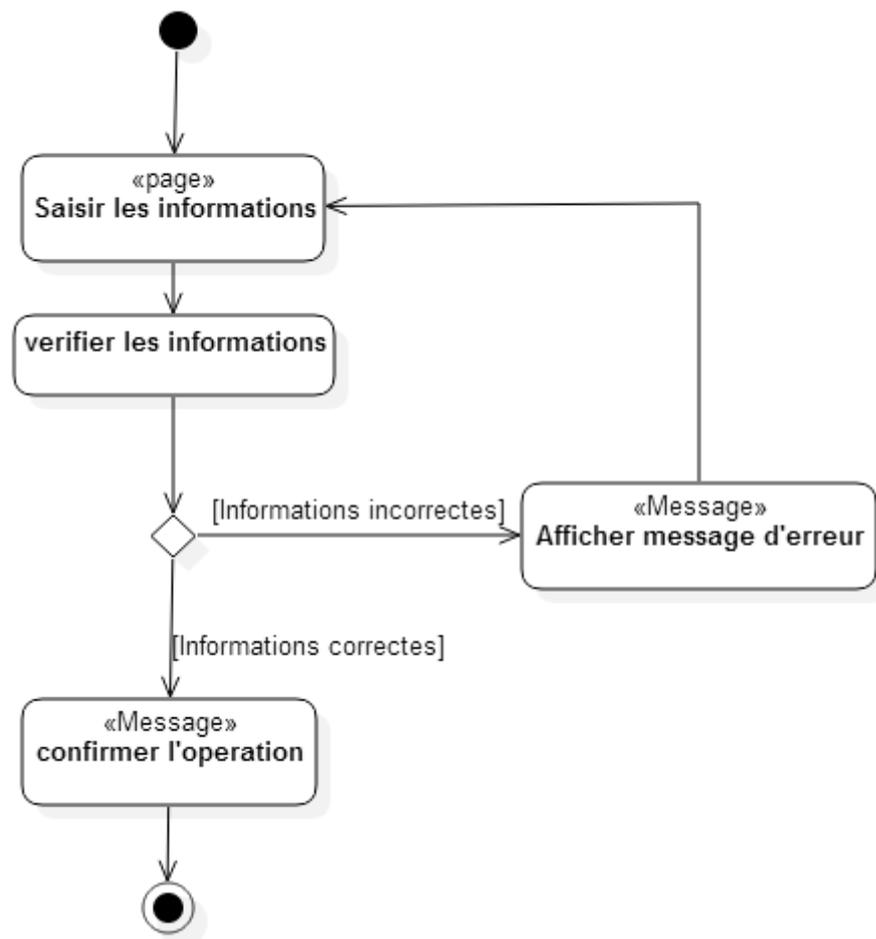


Figure 7. Diagramme d'activité de navigation «S'identifier».

### 5.2- Fiche descriptive du cas d'utilisation « S'inscrire »

S'inscrire	
<b>Description</b>	Ce cas permet à n'importe quel utilisateur d'inscrire dans le site.
<b>Acteur</b>	Visiteur.
<b>Pré condition</b>	
<b>Post condition</b>	L'utilisateur possède un compte.
<b>Nominal</b>	<ol style="list-style-type: none"> <li>1. L'utilisateur demande d'inscrire dans le site.</li> <li>2. Le système affiche à l'utilisateur le formulaire d'inscription.</li> <li>3. L'utilisateur saisit ses informations.</li> <li>4. Le système valide l'opération.</li> </ol>
<b>Alternatif</b>	<ol style="list-style-type: none"> <li>1. L'information entrée existe déjà (retour au scénario nominal 3).</li> <li>2. Erreur d'information (retour au scénario nominal 3).</li> </ol>
<b>Exception</b>	L'utilisateur quitte le système.

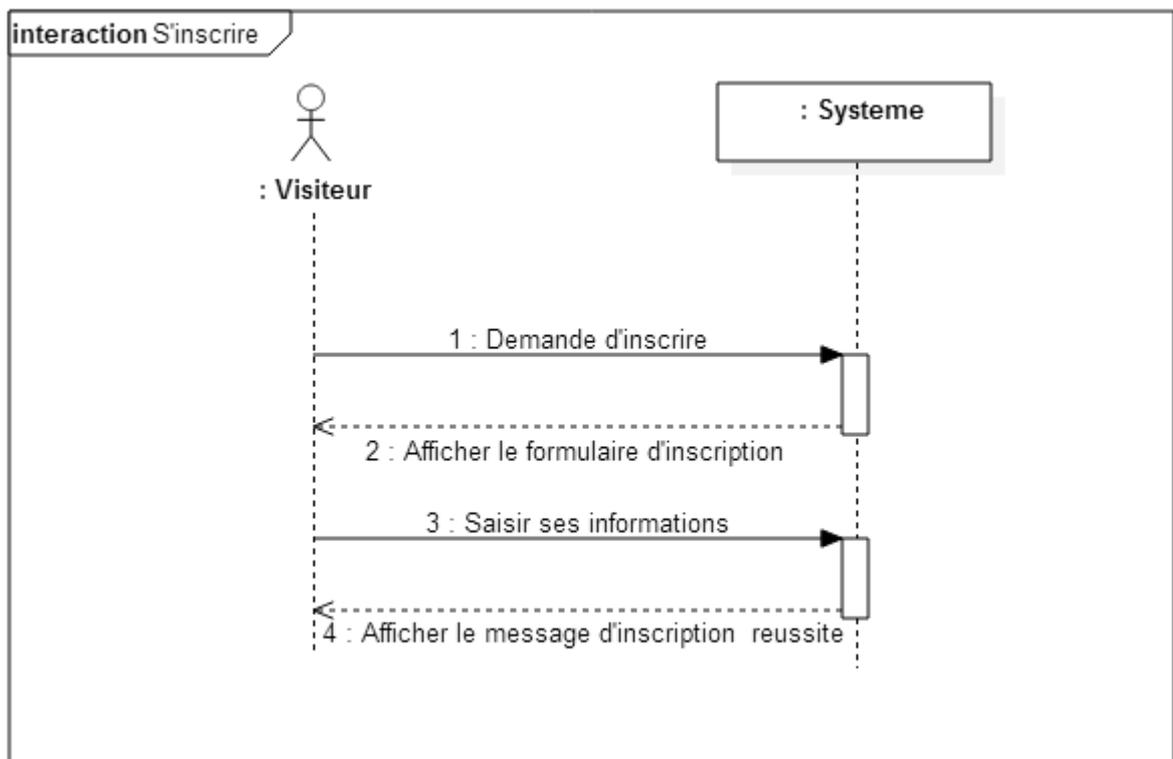


Figure 8. Diagramme de séquence système «S'inscrire».

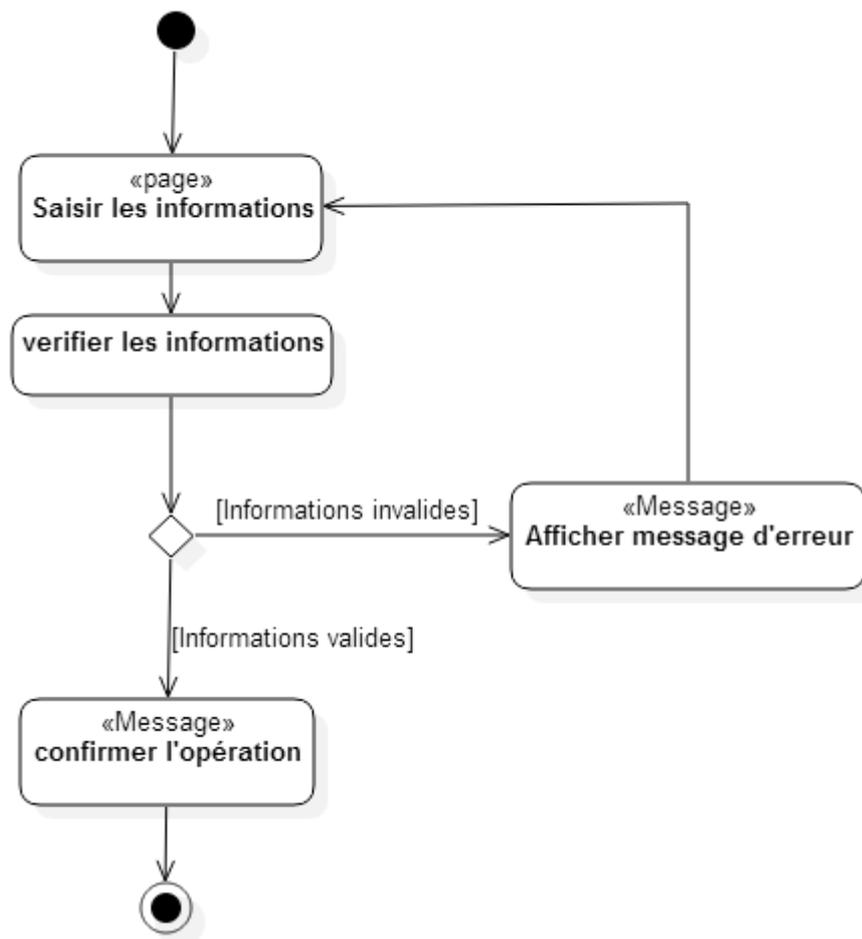


Figure 9. Diagramme d'activité de navigation « S'inscrire ».

### 5.3- Fiche descriptive du cas d'utilisation « visiter le site»

visiter le site	
<b>Description</b>	Ce cas permet à n'importe quel utilisateur d'inscrire dans le site.
<b>Acteur</b>	Visiteur.
<b>Pré condition</b>	La connexion à l'Internet existe.
<b>Post condition</b>	donner une présentation globale sur les services de la clinique médicale.
<b>Nominal</b>	<ol style="list-style-type: none"> <li>3. le visiteur est connecté à l'internet.</li> <li>4. Le visiteur veut consulter le site.</li> <li>5. le système affiche la page d'accueil, à partir de laquelle il peut découvrir les services que cette clinique médicale offre.</li> <li>6. le visiteur peut accéder à n'importe quel service.</li> </ol>
<b>Alternatif</b>	Néant.
<b>Exception</b>	On suppose qu'il n'y a pas de connexion, le cas d'utilisation se termine.

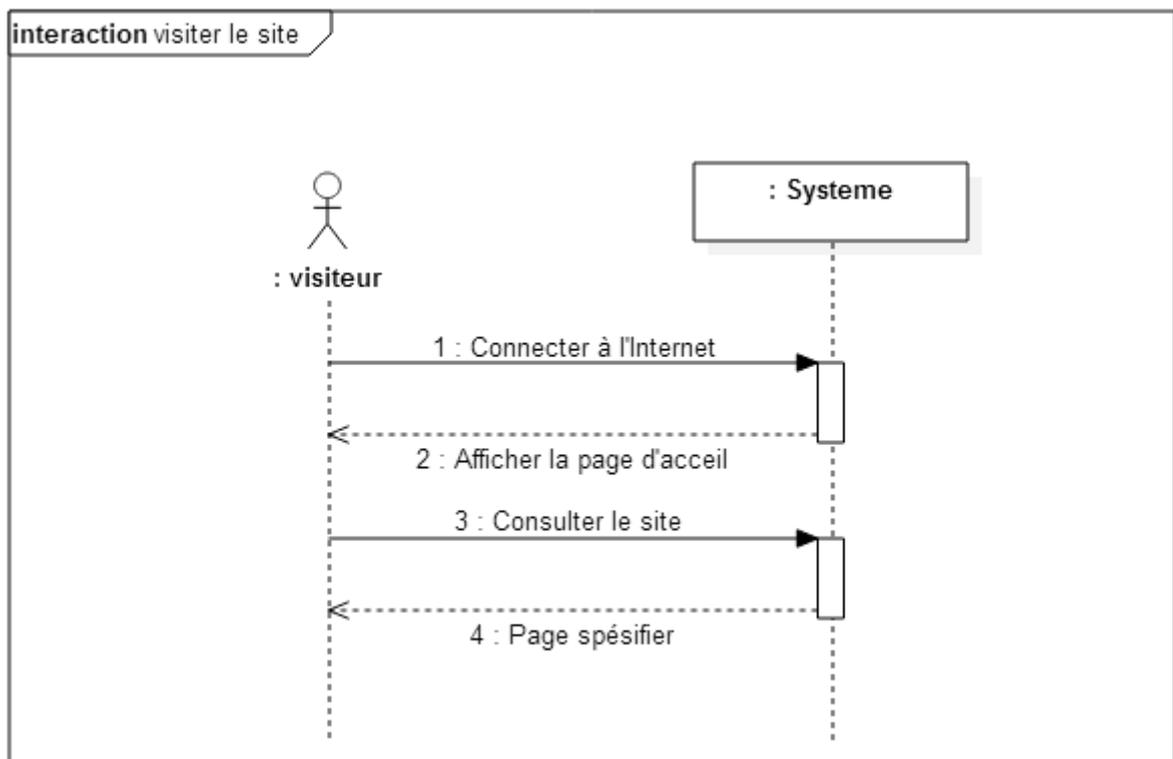


Figure 10. Diagramme de séquence système «Visiter le site».

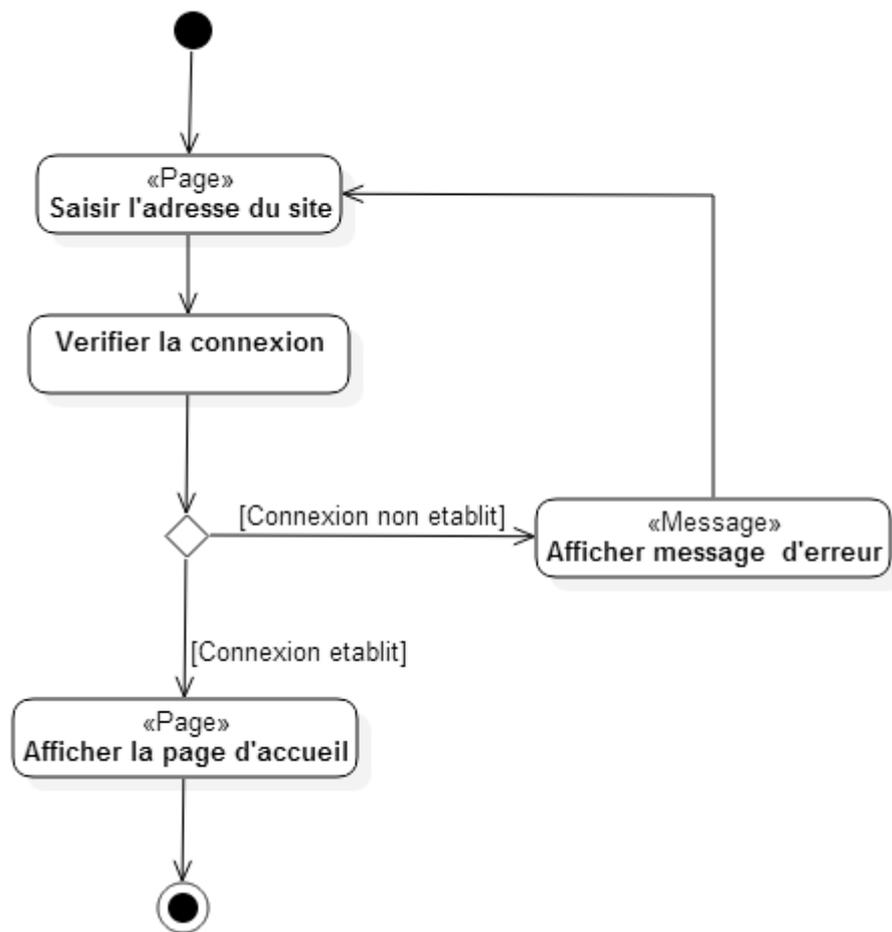


Figure 11. Diagramme d'activité de navigation « Visiter le site ».

### 5.4- Fiche descriptive du cas d'utilisation « Tester la disponibilité des RDV »

Tester la disponibilité des rendez-vous.	
<b>Description</b>	Ce cas permet à n'importe quel utilisateur d'inscrire dans le site.
<b>Acteur</b>	Visiteur.
<b>Pré condition</b>	Le visiteur est identifié.
<b>Post condition</b>	Définir l'état de RDV.
<b>Nominal</b>	<ol style="list-style-type: none"> <li>1. Le visiteur demande la recherche d'un RDV.</li> <li>2. Le système affiche le formulaire de recherche.</li> <li>3. Le visiteur saisit les informations du RDV.</li> <li>4. les Le système affiche les RDV disponibles.</li> </ol>
<b>Exception</b>	Néant.

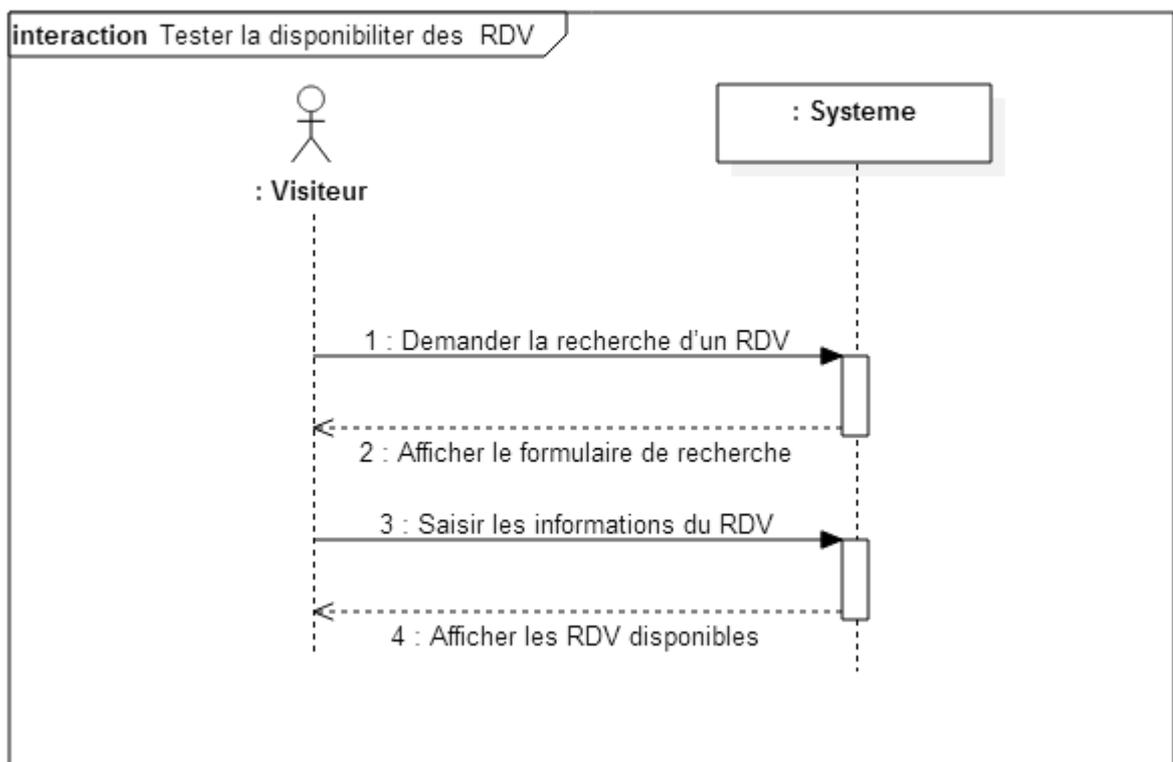


Figure 12. Diagramme de séquence système «Tester la disponibilité des RDV».

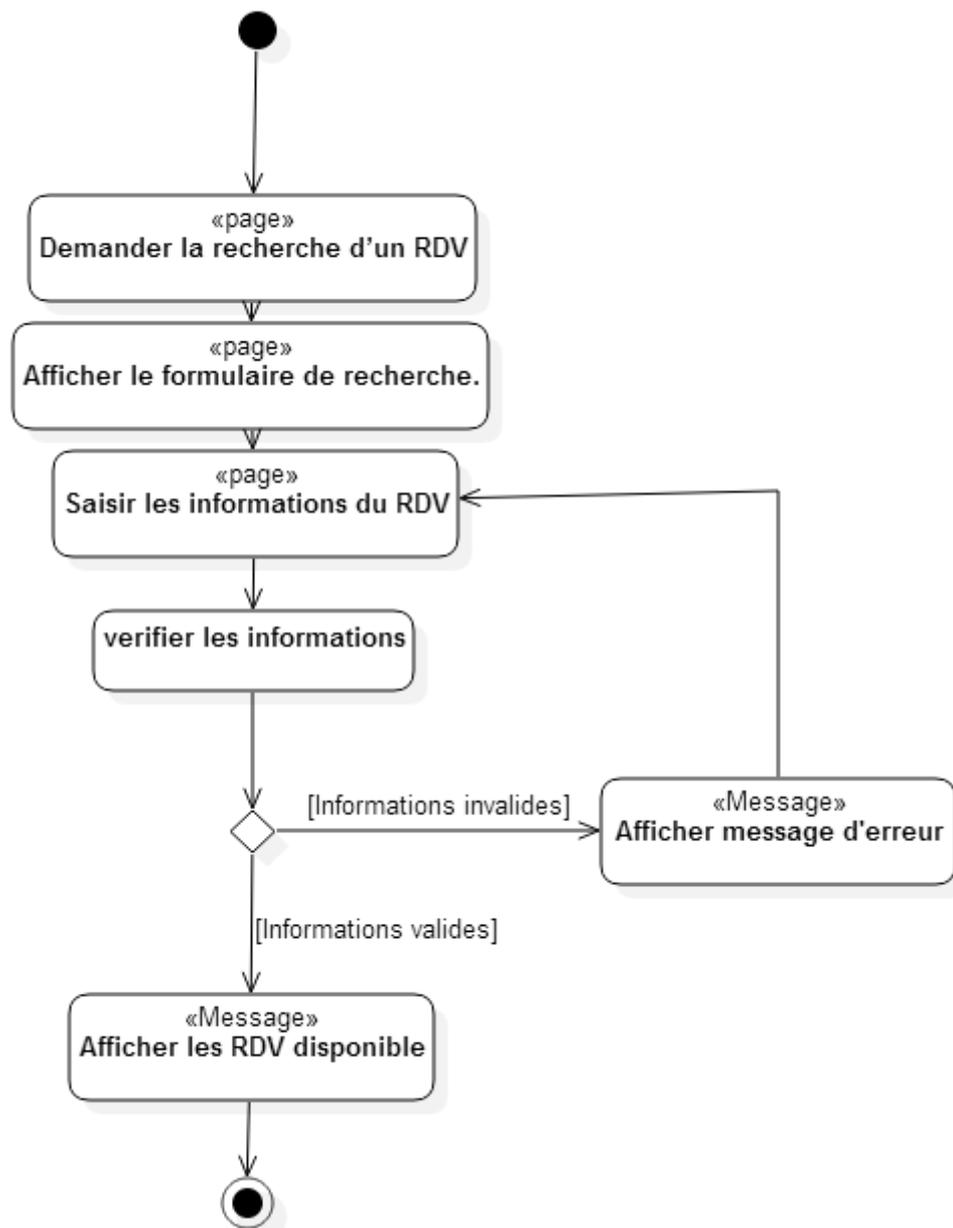


Figure 13. Diagramme d'activité de navigation «Tester la disponibilité des RDV».

### 5.5- Fiche descriptive du cas d'utilisation «Choisir un RDV»

Choisir un RDV	
<b>Description</b>	Permettre au patient de fixer un rendez-vous pour la consultation.
<b>Acteur</b>	Patient.
<b>Pré condition</b>	RDV disponibles
<b>Post condition</b>	Le RDV est fixé.
<b>Nominal</b>	<ol style="list-style-type: none"> <li>1. Le patient demande la fixation d'un RDV.</li> <li>2. Le système affiche les rendez-vous disponible pour chaque date.</li> <li>3. Le patient définit la date du rendez-vous.</li> <li>4. Le système confirme la fixation du rendez-vous.</li> </ol>
<b>Alternatif</b>	<ol style="list-style-type: none"> <li>1. S'il n'y a pas un RDV disponible (retour au scénario nominal 3).</li> <li>2. Erreur d'information (retour au scénario nominal 3).</li> </ol>
<b>Exception</b>	L'utilisateur quitte le système.

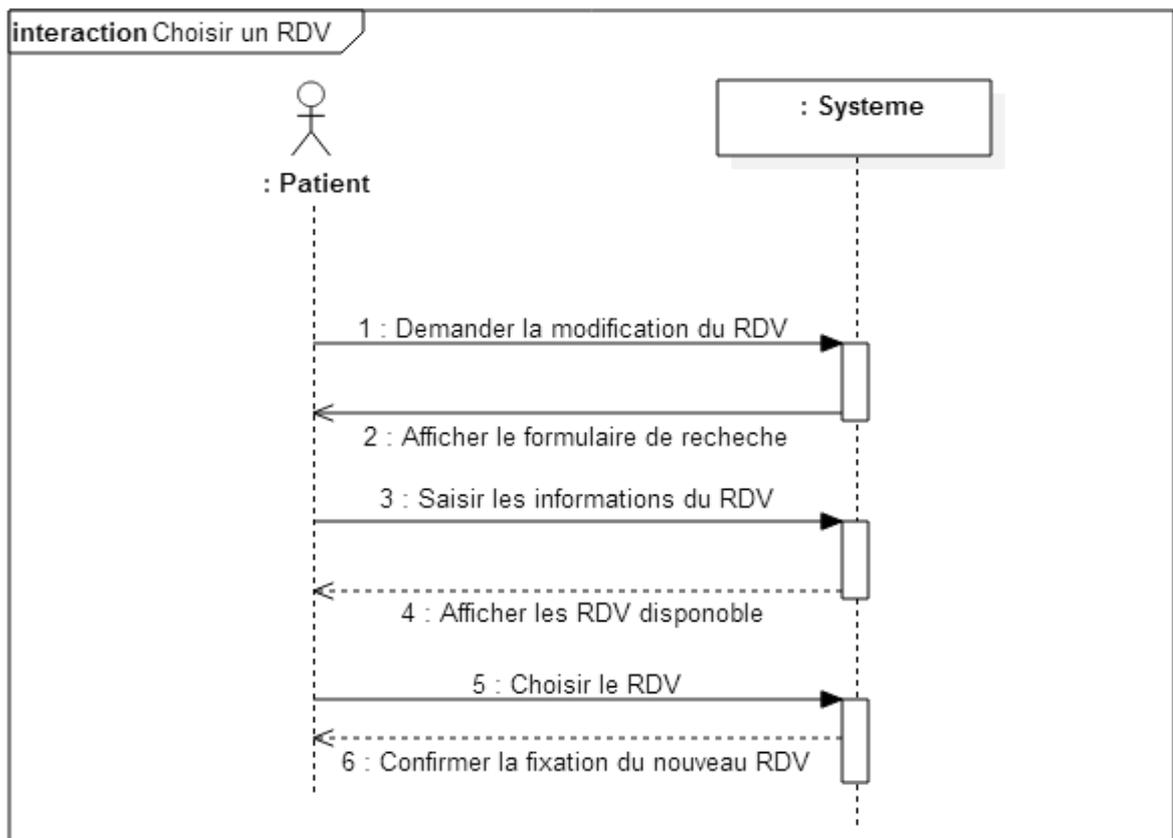


Figure 14. Diagramme de séquence système «Choisir un RDV».

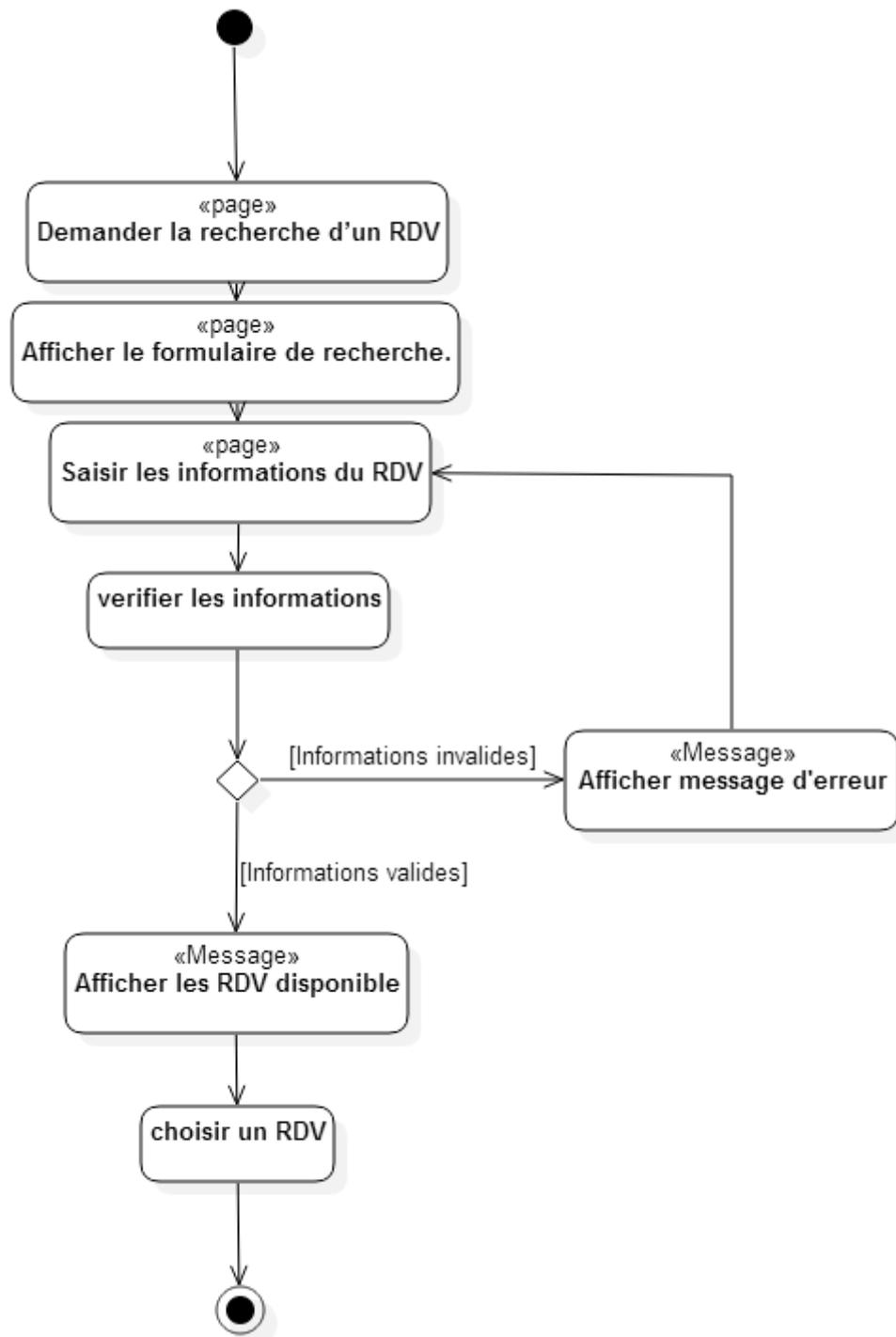


Figure 15. Diagramme d'activité de navigation «Choisir un RDV».

### 5.6- Fiche descriptive du cas d'utilisation «Modifier un RDV»

Modifier un RDV	
<b>Description</b>	Permettre au patient de modifier le rendez-vous.
<b>Acteur</b>	Patient.
<b>Pré condition</b>	Le RDV existe.
<b>Post condition</b>	Le RDV est modifié.
<b>Nominal</b>	<ol style="list-style-type: none"> <li>1. Le patient demande la modification du RDV.</li> <li>2. Le système affiche le RDV indiqué.</li> <li>3. Le patient définit le nouveau rendez-vous.</li> <li>4. Le système confirme la modification du rendez-vous.</li> </ol>
<b>Alternatif</b>	<ol style="list-style-type: none"> <li>1. S'il n'y a pas un RDV disponible (retour au scénario nominal 3).</li> <li>2. Erreur d'information (retour au scénario nominal 3).</li> </ol>
<b>Exception</b>	L'utilisateur quitte le système.

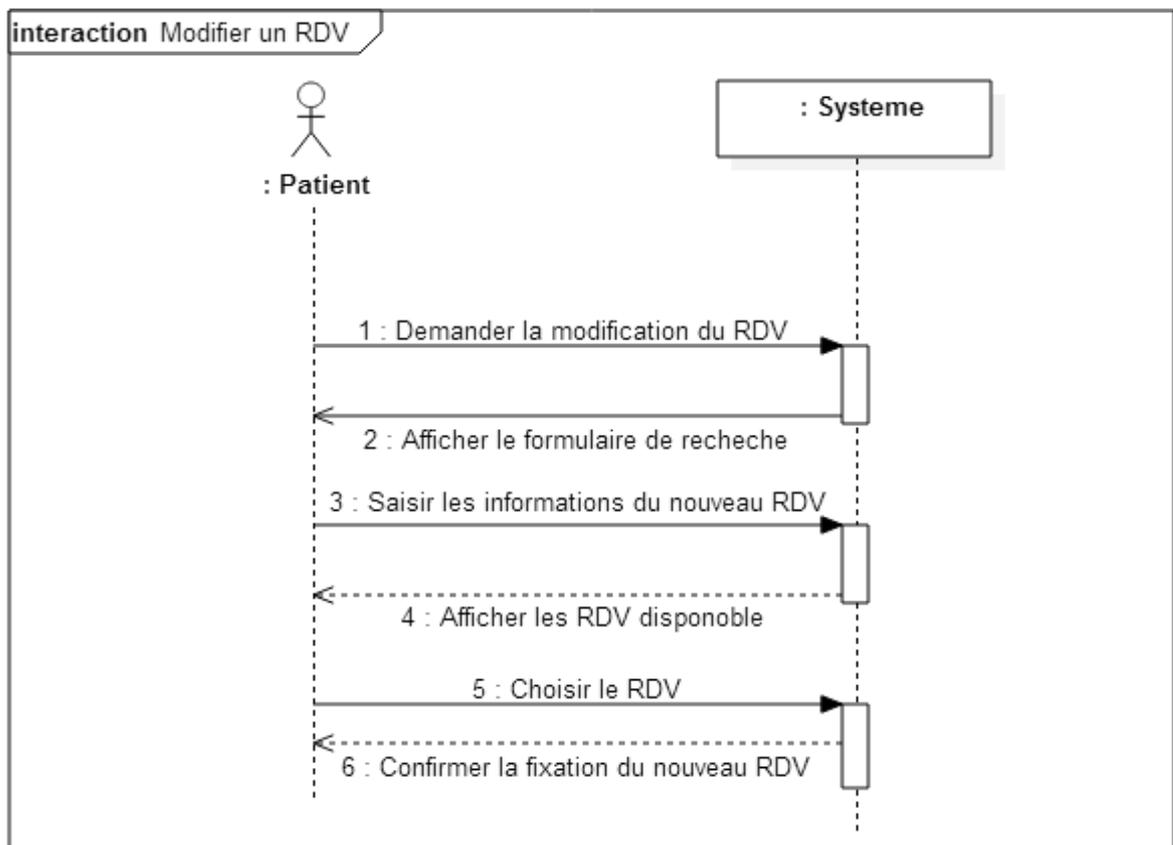


Figure 16. Diagramme de séquence système «Modifier un RDV».

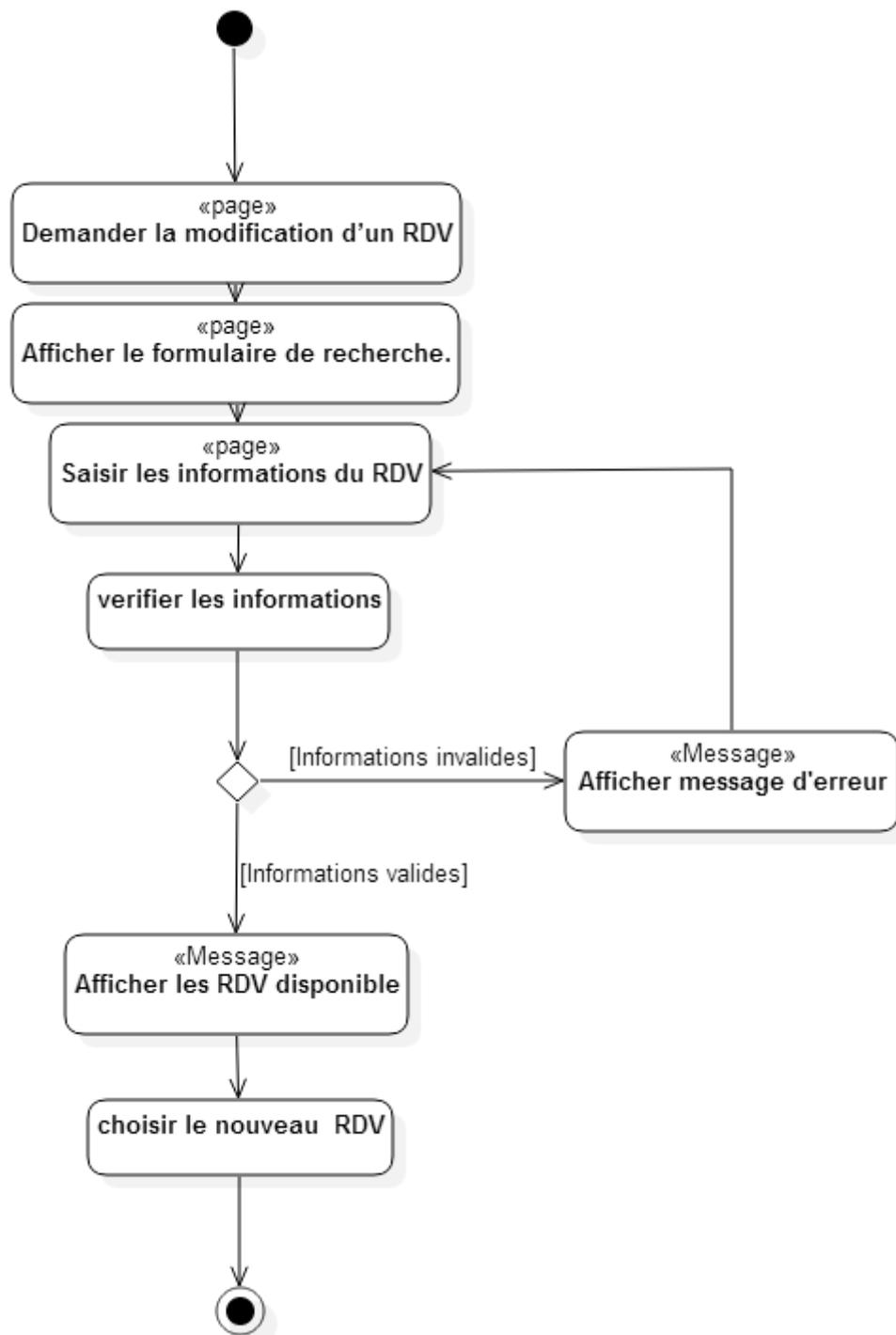


Figure 17. Diagramme d'activité de navigation «Modifier un RDV».

### 5.7- Fiche descriptive du cas d'utilisation «Annuler un RDV»

Annuler un RDV	
<b>Description</b>	Permettre au patient d'annuler le rendez-vous.
<b>Acteur</b>	Patient.
<b>Pré condition</b>	RDV existe.
<b>Post condition</b>	Le RDV annuler.
<b>Nominal</b>	<ol style="list-style-type: none"> <li>1. Le patient demande d'annuler le RDV.</li> <li>2. Le système affiche une boîte de dialogue de suppression.</li> <li>3. Le patient annule le RDV.</li> <li>4. Le système confirme l'annulation du rendez-vous.</li> </ol>
<b>Alternatif</b>	Néant.
<b>Exception</b>	L'utilisateur quitte le système.

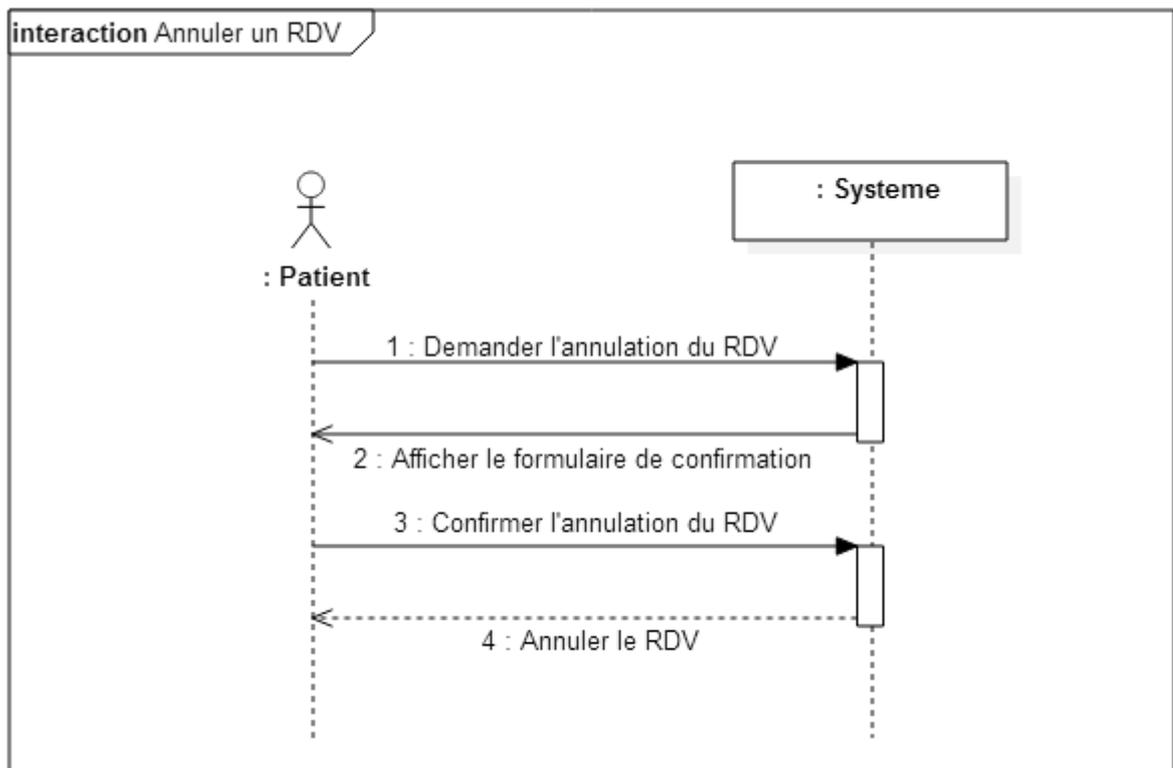
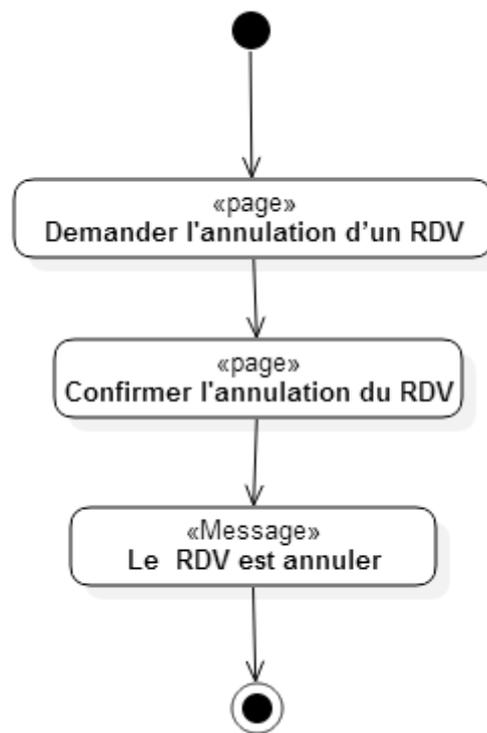


Figure 18. Diagramme de séquence système «Annuler un RDV».



**Figure 19.** Diagramme d'activité de navigation «Annuler un RDV».

### 5.8- Fiche descriptive du cas d'utilisation «Ajouter un service»

Ajouter un service	
<b>Description</b>	Permettre à l'agent administratif d'ajouter un service.
<b>Acteur</b>	Agent administratif.
<b>Pré condition</b>	Pas de service.
<b>Post condition</b>	Le service est ajouté.
<b>Nominal</b>	<ol style="list-style-type: none"> <li>1. L'agent administratif demande d'ajouter un service.</li> <li>2. Le système affiche le formulaire.</li> <li>3. L'agent administratif saisit les informations du service.</li> <li>4. Le système confirme l'ajout du service.</li> </ol>
<b>Alternatif</b>	Le service existe déjà (retour au scénario nominal 3). Erreur d'information (retour au scénario nominal 3).
<b>Exception</b>	L'utilisateur quitte le système.

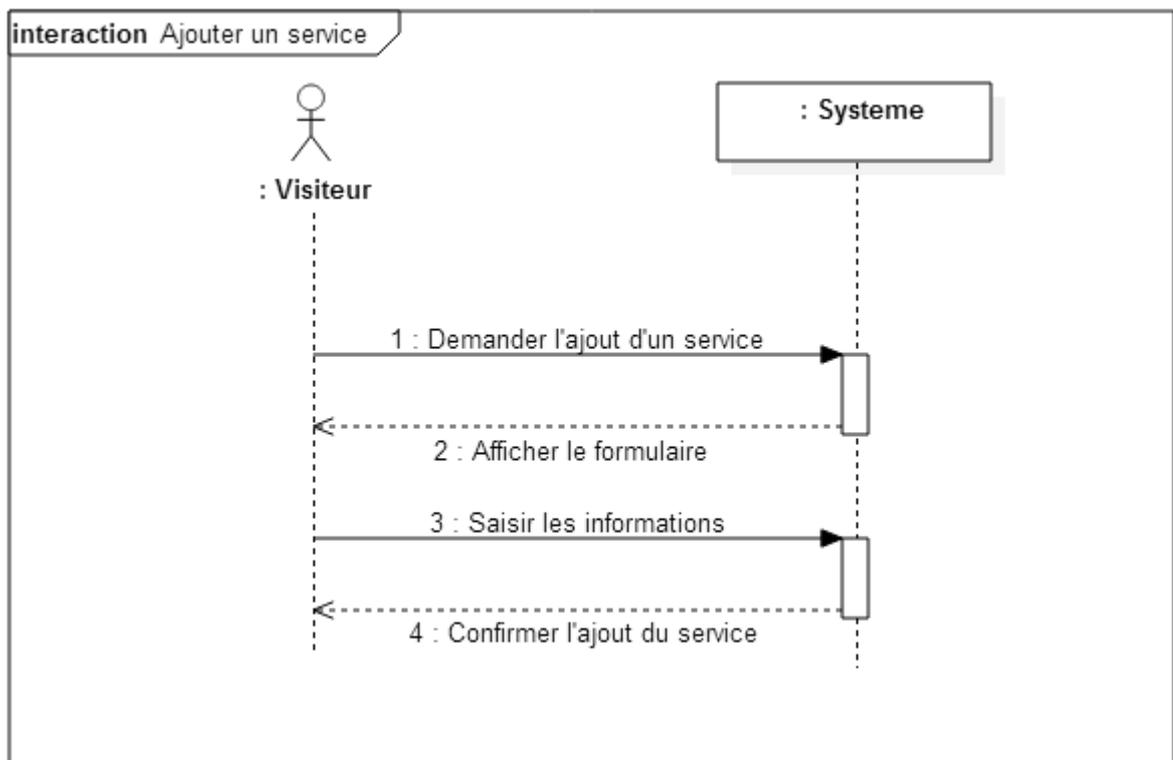


Figure 20. Diagramme de séquence système «Ajouter un service».

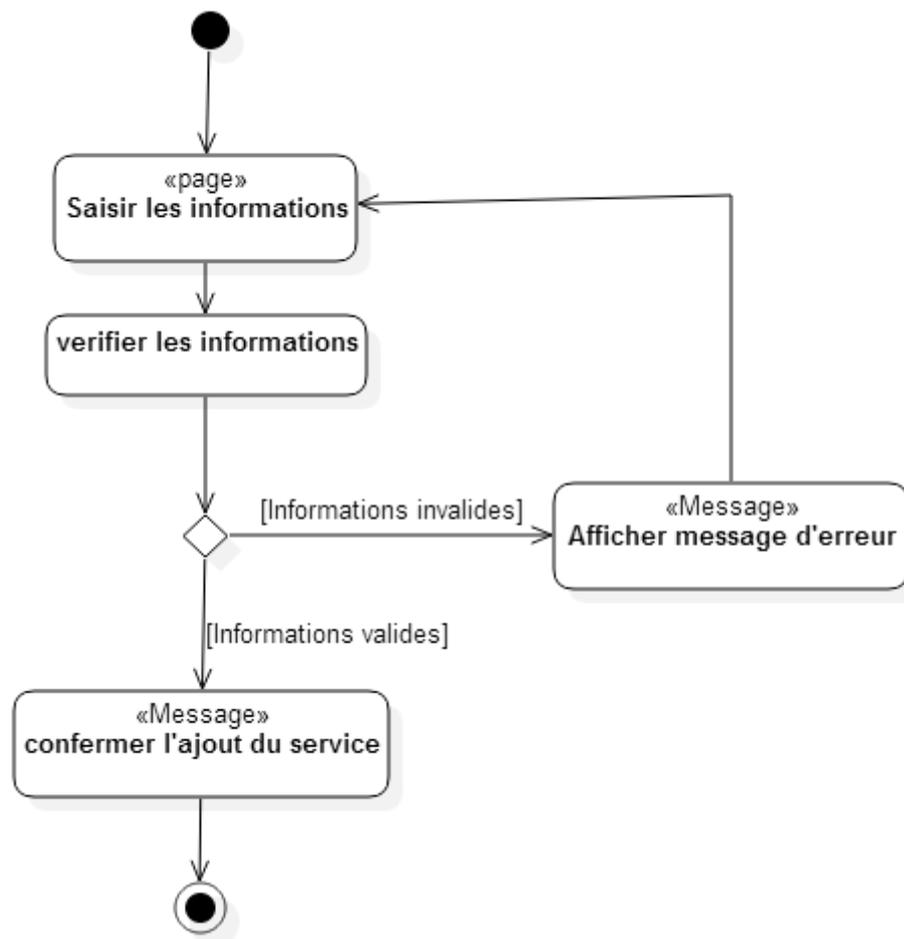


Figure 21. Diagramme d'activité de navigation «Ajouter un service».

### 5.9- Fiche descriptive du cas d'utilisation «Modifier un service»

Modifier un service	
<b>Description</b>	Permettre à l'agent administratif de modifier un service.
<b>Acteur</b>	Agent administratif.
<b>Pré condition</b>	Le service existe.
<b>Post condition</b>	Le service est modifié.
<b>Nominal</b>	<ol style="list-style-type: none"> <li>1. L'agent administratif demande de modifier un service.</li> <li>2. Le système affiche le formulaire.</li> <li>3. L'agent administratif saisit les informations du nouveau service.</li> <li>4. Le système confirme la modification du service.</li> </ol>
<b>Alternatif</b>	Le nouveau service existe déjà.
<b>Exception</b>	L'utilisateur quitte le système.

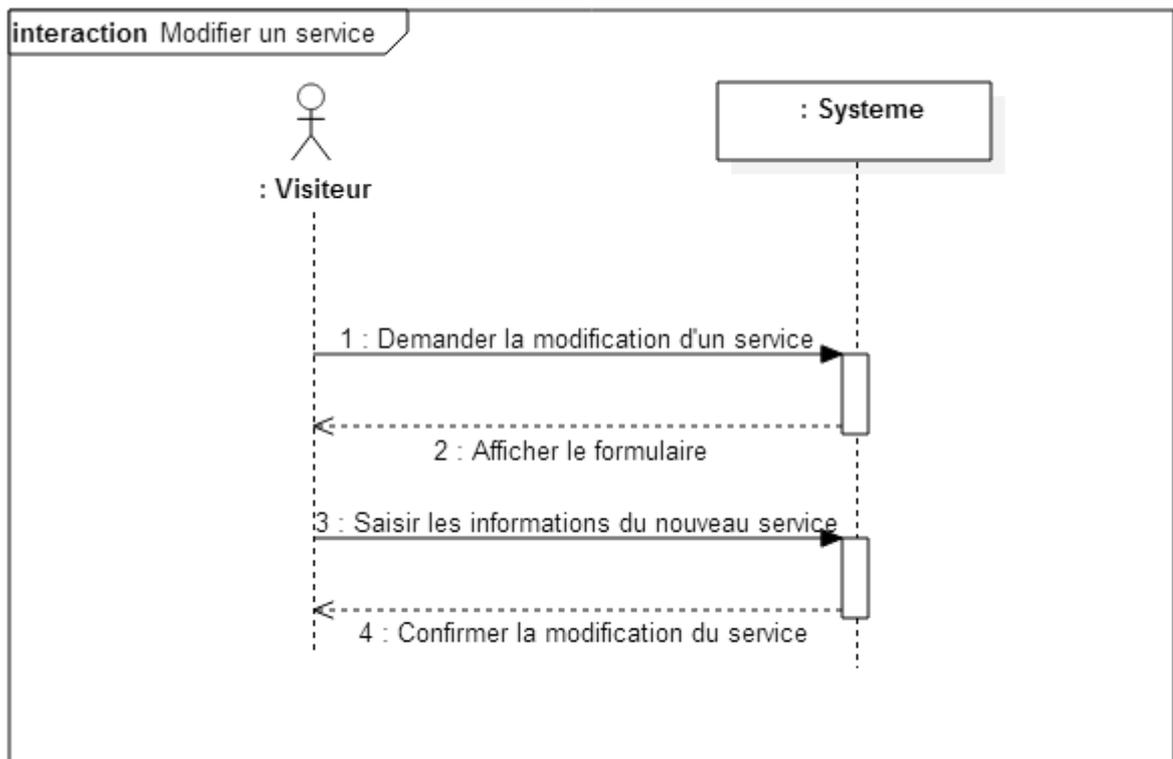


Figure 22. Diagramme de séquence système «Modifier un service».

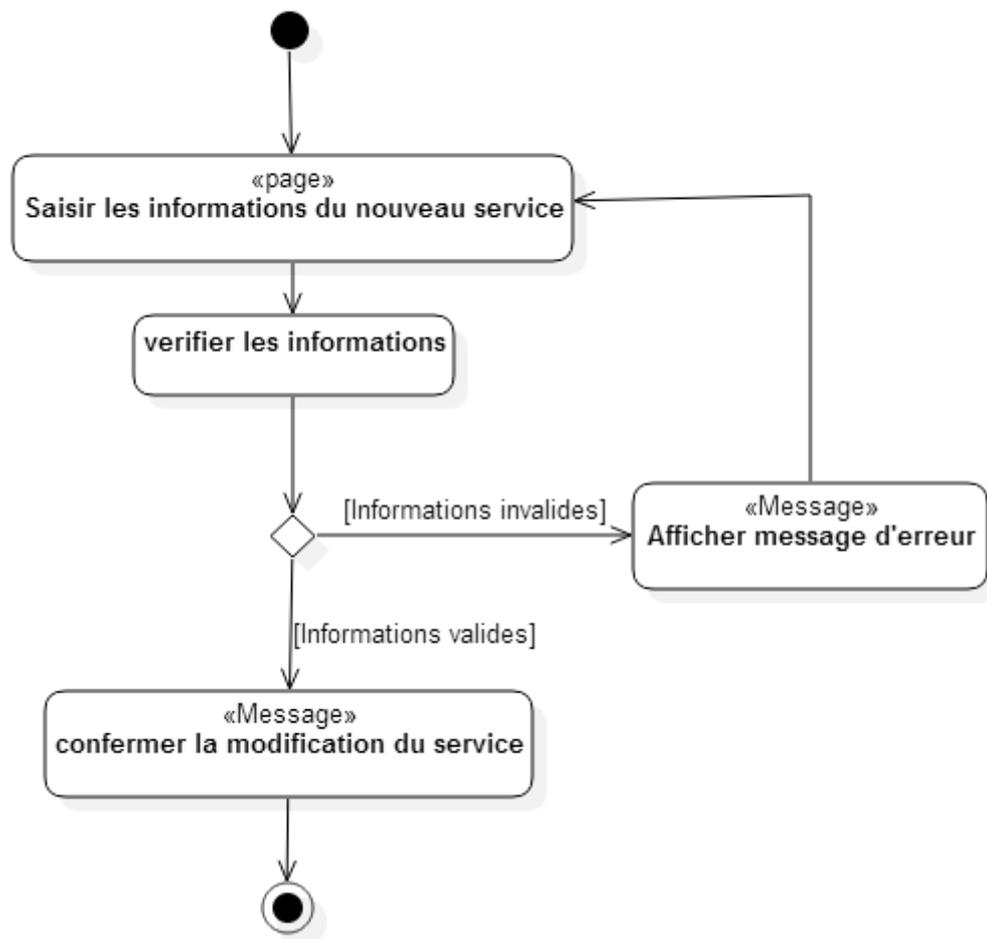


Figure 23. Diagramme d'activité de navigation «Modifier un service».

### 5.10- Fiche descriptive du cas d'utilisation «Annuler un service»

Annuler un service	
<b>Description</b>	Permettre à l'agent administratif d'annuler un service.
<b>Acteur</b>	Agent administratif.
<b>Pré condition</b>	Le service existe.
<b>Post condition</b>	Le service est annulé.
<b>Nominal</b>	<ol style="list-style-type: none"> <li>1. L'agent administratif demande d'annuler un service.</li> <li>5. Le système affiche une boîte de dialogue de suppression.</li> <li>6. L'agent administratif annule le service.</li> <li>2. Le système confirme l'annulation du service.</li> </ol>
<b>Alternatif</b>	Néant.
<b>Exception</b>	L'utilisateur quitte le système.

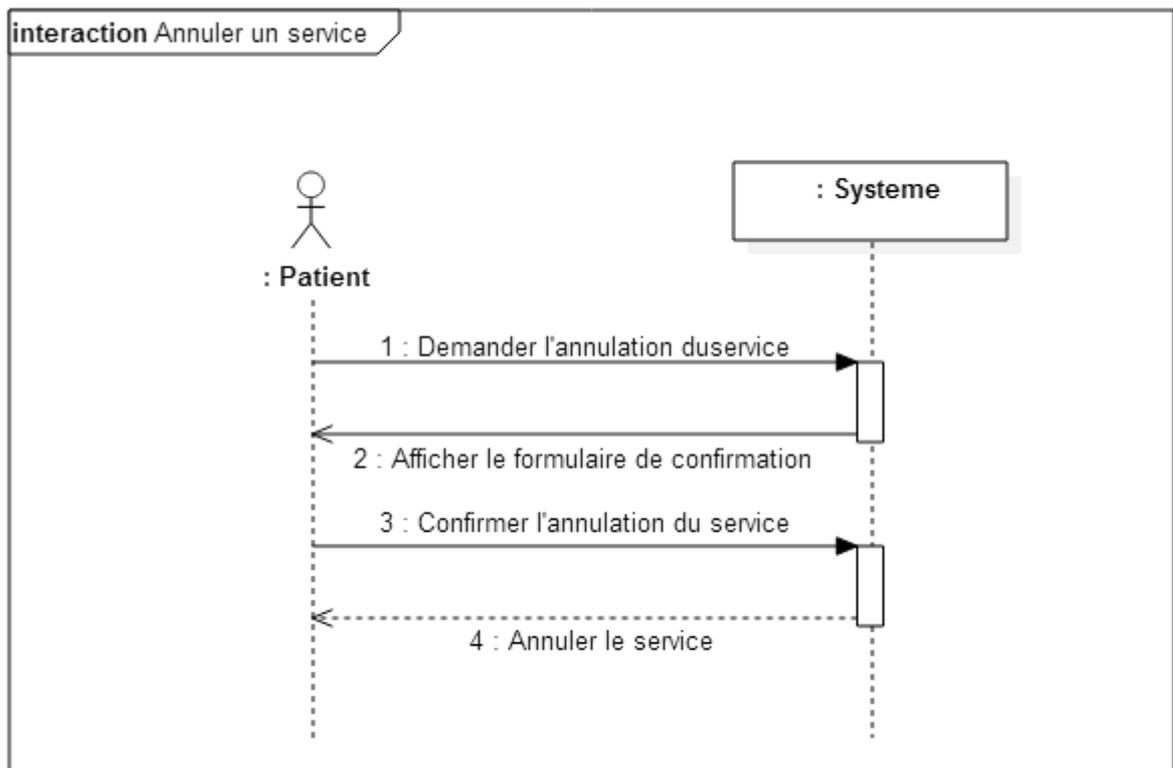
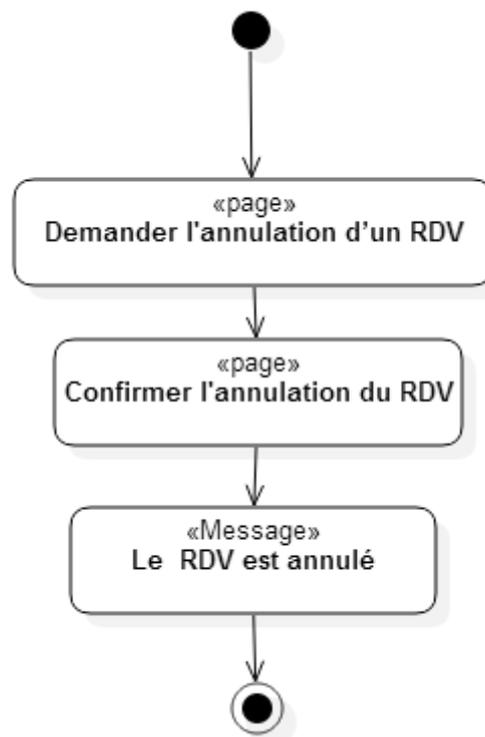


Figure24. Diagramme de séquence système «Annuler un service».



**Figure 25. Diagramme d'activité de navigation «Annuler un service».**

### 5.11- Fiche descriptive du cas d'utilisation «Rechercher un RDV»

Rechercher des RDV	
<b>Description</b>	Permettre à l'agent administratif de rechercher des RDV.
<b>Acteur</b>	agent administratif.
<b>Pré condition</b>	L'agent administratif s'identifier.
<b>Post condition</b>	Les RDV est affiché.
<b>Nominal</b>	<ol style="list-style-type: none"> <li>1. L'agent administratif demande de rechercher d'un RDV.</li> <li>2. Le système affiche le formulaire de recherche.</li> <li>3. L'agent administratif saisit les informations du RDV.</li> <li>4. Le système affiche les RDV.</li> </ol>
<b>Alternatif</b>	<ol style="list-style-type: none"> <li>1. S'il n'y a pas un RDV (retour au scénario nominal 3).</li> <li>2. Erreur d'information (retour au scénario nominal 3).</li> </ol>
<b>Exception</b>	L'utilisateur quitte le système.

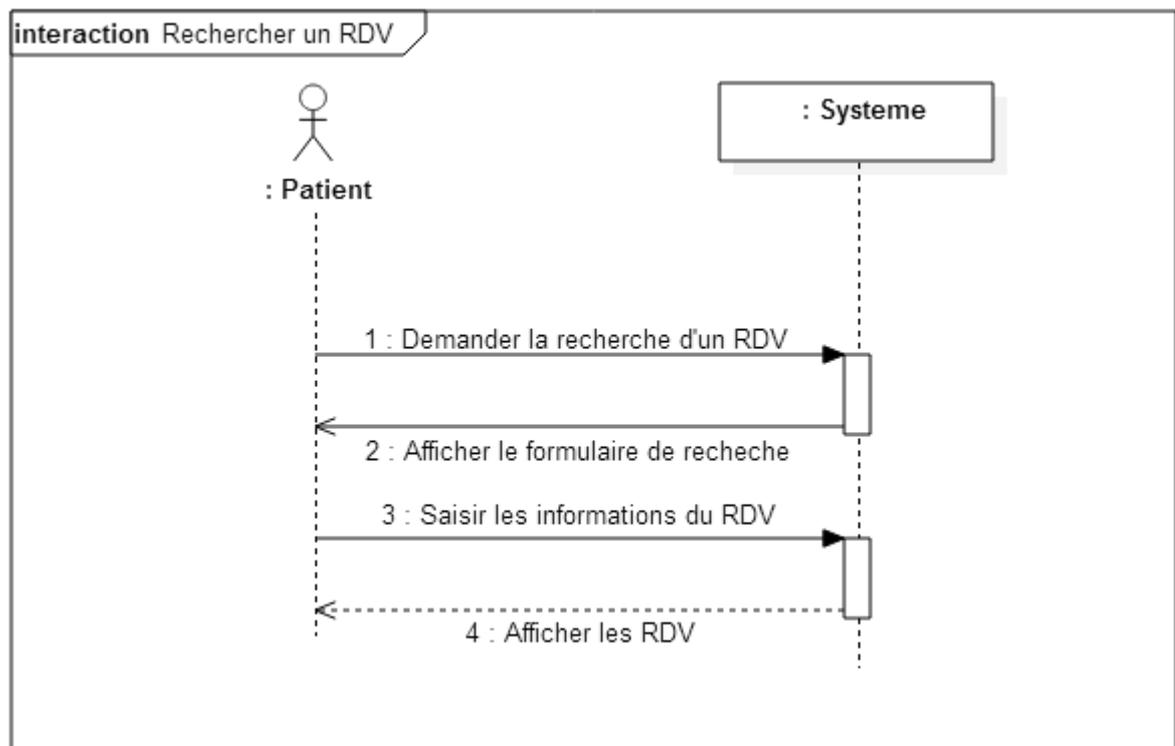


Figure 26. Diagramme de séquence système «Rechercher un RDV».

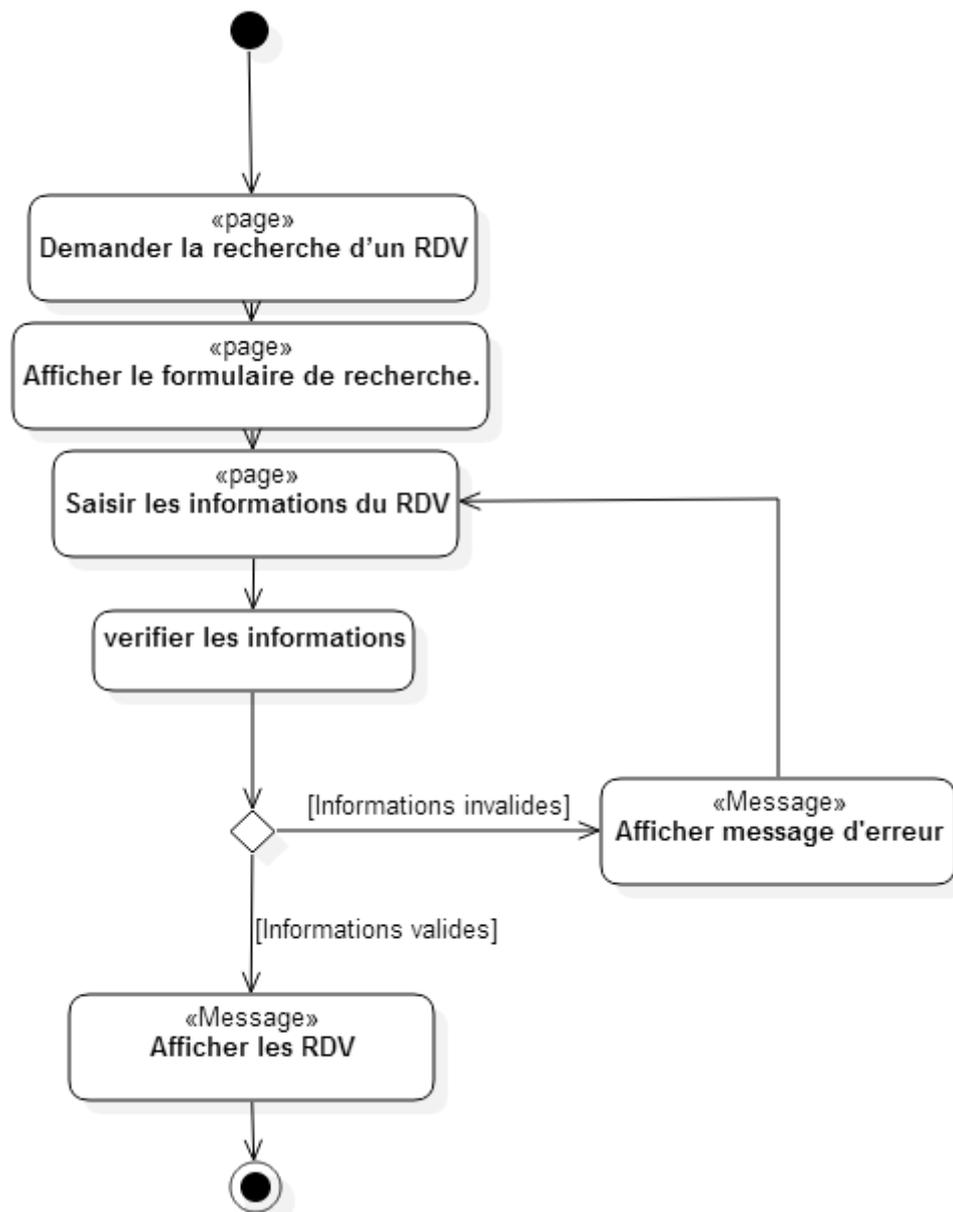


Figure27. Diagramme d'activité de navigation «Rechercher un RDV».

### 5.12- Fiche descriptive du cas d'utilisation «Rechercher un Patient»

Rechercher un patient	
<b>Description</b>	Permettre à l'agent administratif de rechercher un Patient.
<b>Acteur</b>	agent administratif.
<b>Pré condition</b>	L'agent administratif s'identifier.
<b>Post condition</b>	Le patient est affiché.
<b>Nominal</b>	5. L'agent administratif demande de rechercher un patient. 6. Le système affiche le formulaire de recherche. 7. L'agent administratif saisit les informations du patient. 1. Le système affiche l'information du patient.
<b>Alternatif</b>	1. S'il n'y a pas un patient (retour au scénario nominal 3). 2. Erreur d'information (retour au scénario nominal 3)...
<b>Exception</b>	L'utilisateur quitte le système.

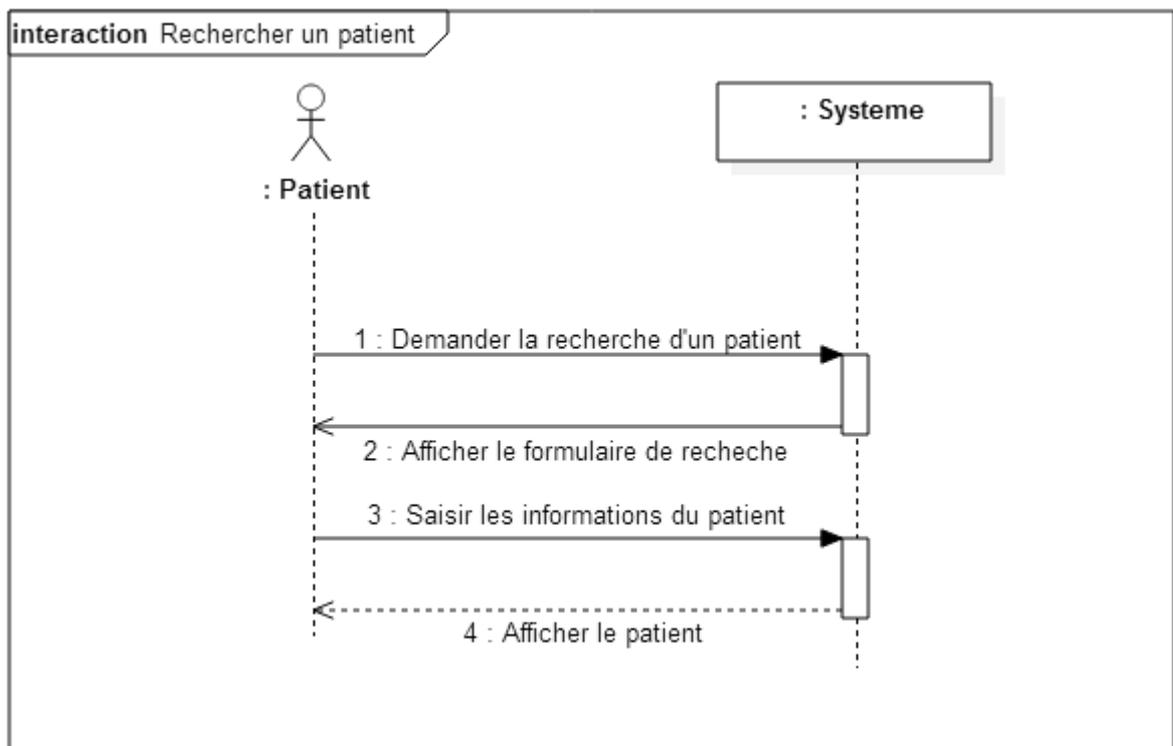


Figure 28. Diagramme de séquence système «Rechercher un patient».

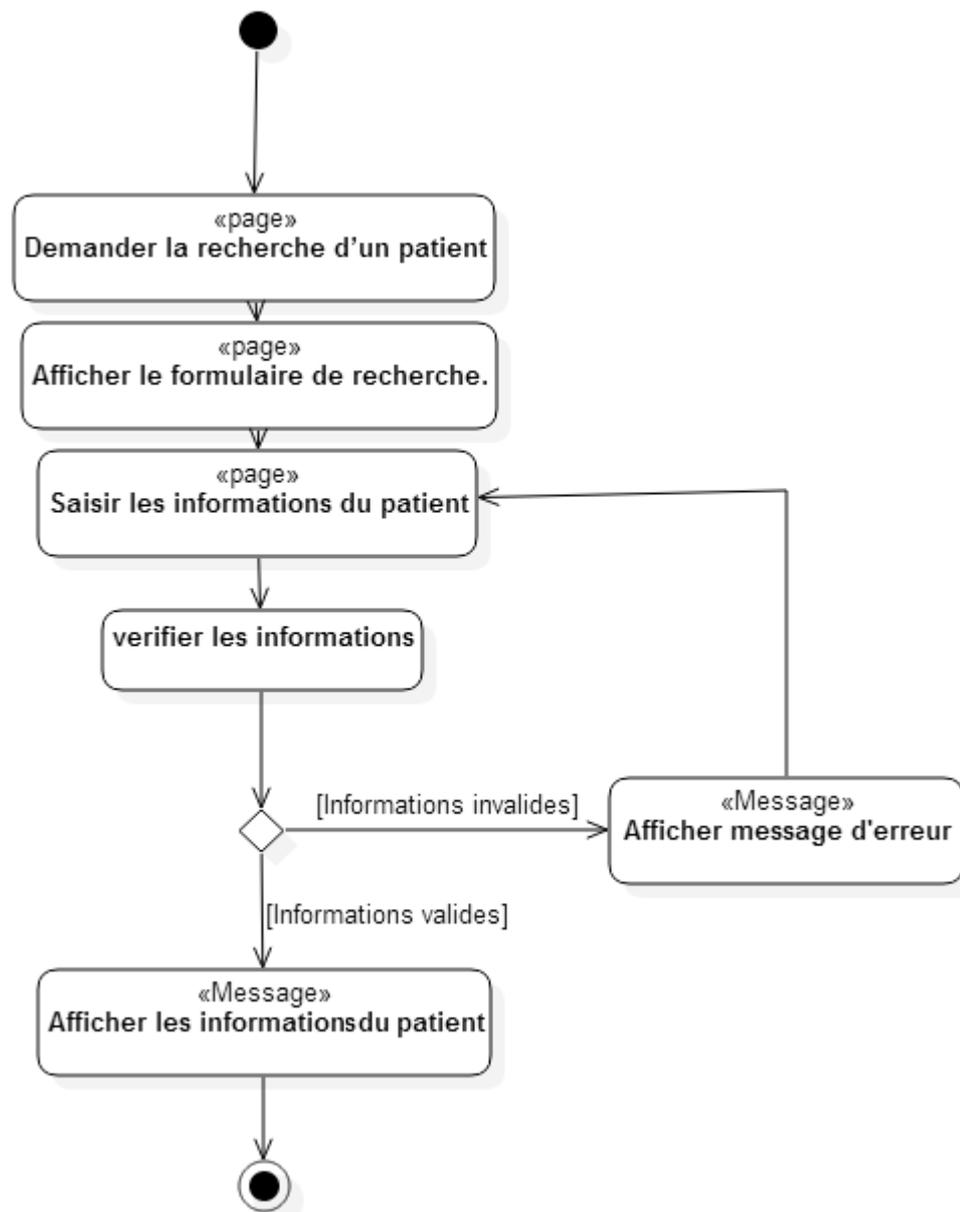


Figure 29. Diagramme d'activité de navigation «Rechercher un patient».

## 6. Conclusion

La phase d'identification des besoins nous a permis de modéliser les cas d'utilisation, les diagrammes de séquence, et les diagrammes d'activité en prenant en considération que le système a toujours vu comme une boîte noire. Ensuite cette phase prépare la phase d'analyse qui est l'objet de chapitre suivant.

# Chapitre IV : Analyse du domaine



## 1. Introduction

Dans ce chapitre nous allons élaborer le modèle de domaine. Puis, Nous développerons les diagrammes de classes participantes qui effectuent la jonction entre, d'une part, les cas d'utilisation, le modèle du domaine et la maquette, et d'autre part, les diagrammes de conception logicielle à savoir ceux d'interaction et celui de classes de conception.

## 2. Modèle de domaine

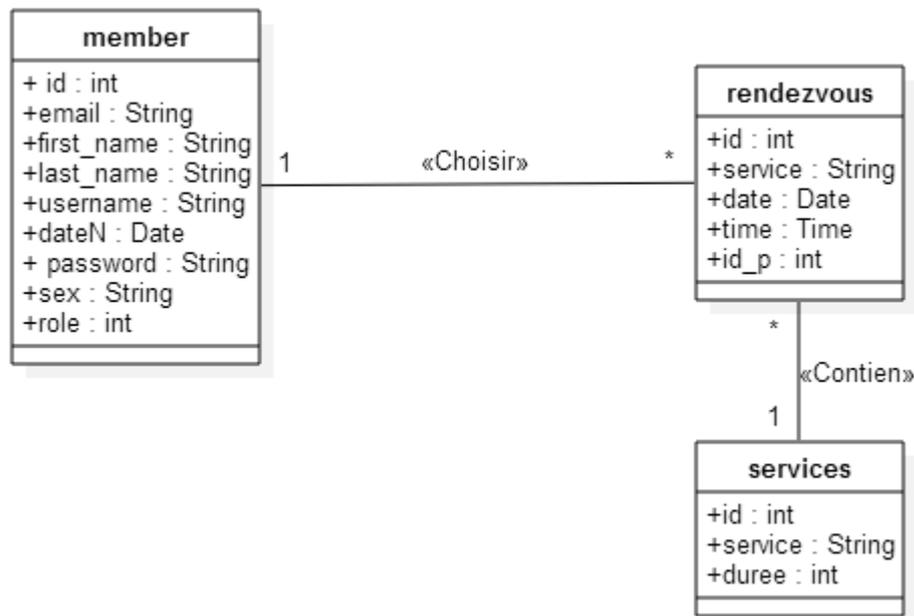


Figure IV.1. Modèle de domaine.

## 3. Les diagrammes de classes participantes

### 3.1- Le cas d'utilisation « S'identifier »

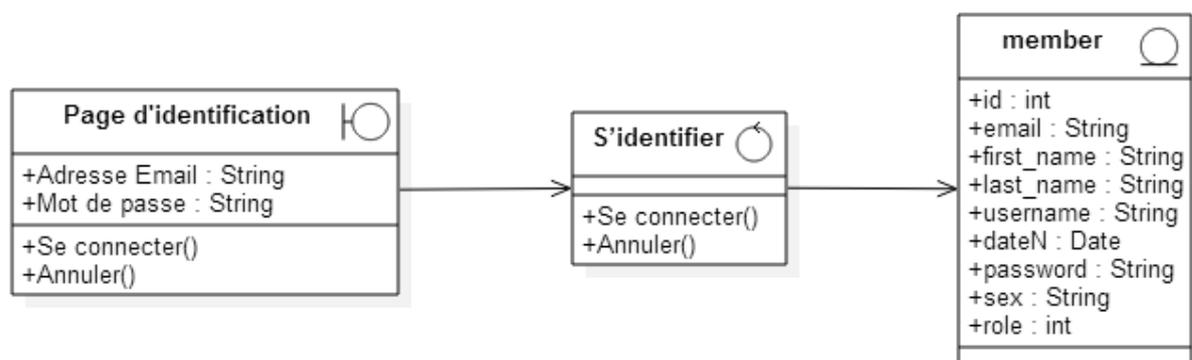


Figure IV.2. Diagramme de classe participante « S'identifier »

### 3.2- Fiche descriptive du cas d'utilisation « S'inscrire »

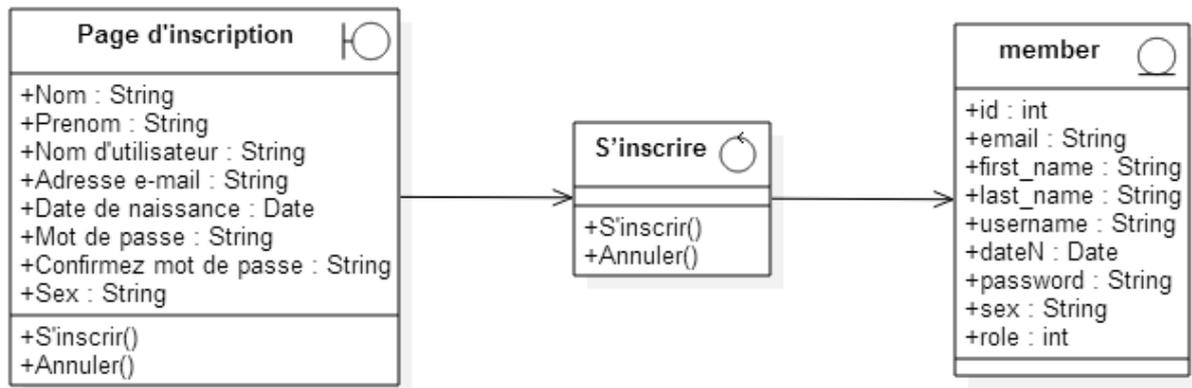


Figure IV.3. Diagramme de classe participante « S'inscrire »

### 3.3- Le cas d'utilisation « Visiter le site »

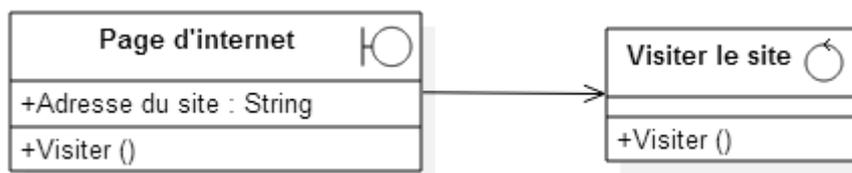


Figure IV.4. Diagramme de classe participante « Visiter le site »

### 3.4- Le cas d'utilisation « Tester la disponibilité des RDV »

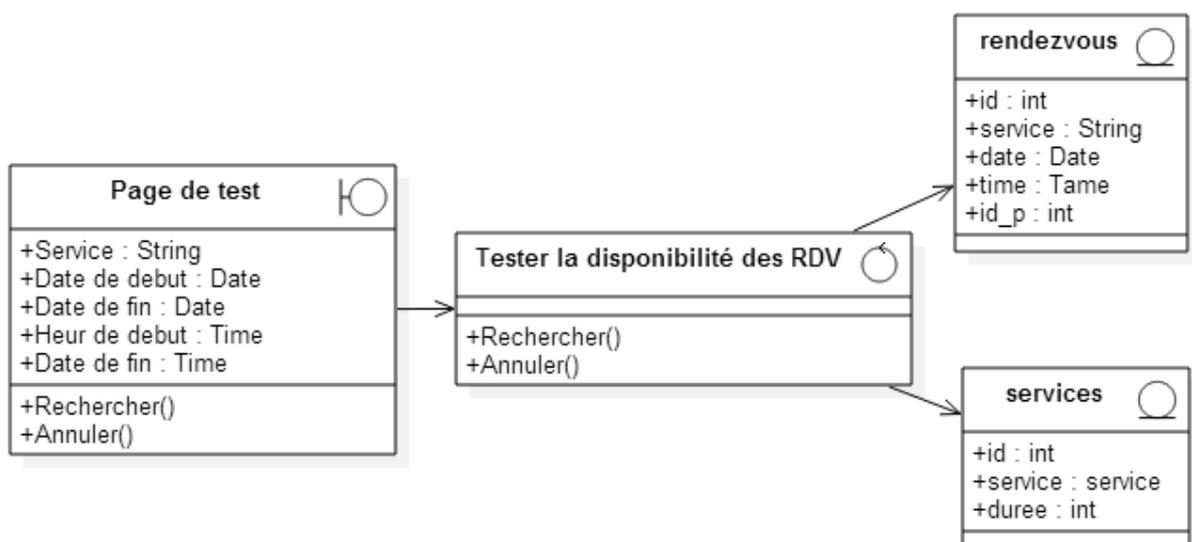


Figure IV.5. Diagramme de classe participante « Tester la disponibilité des RDV »

### 3.5- Le cas d'utilisation « Choisir un RDV »

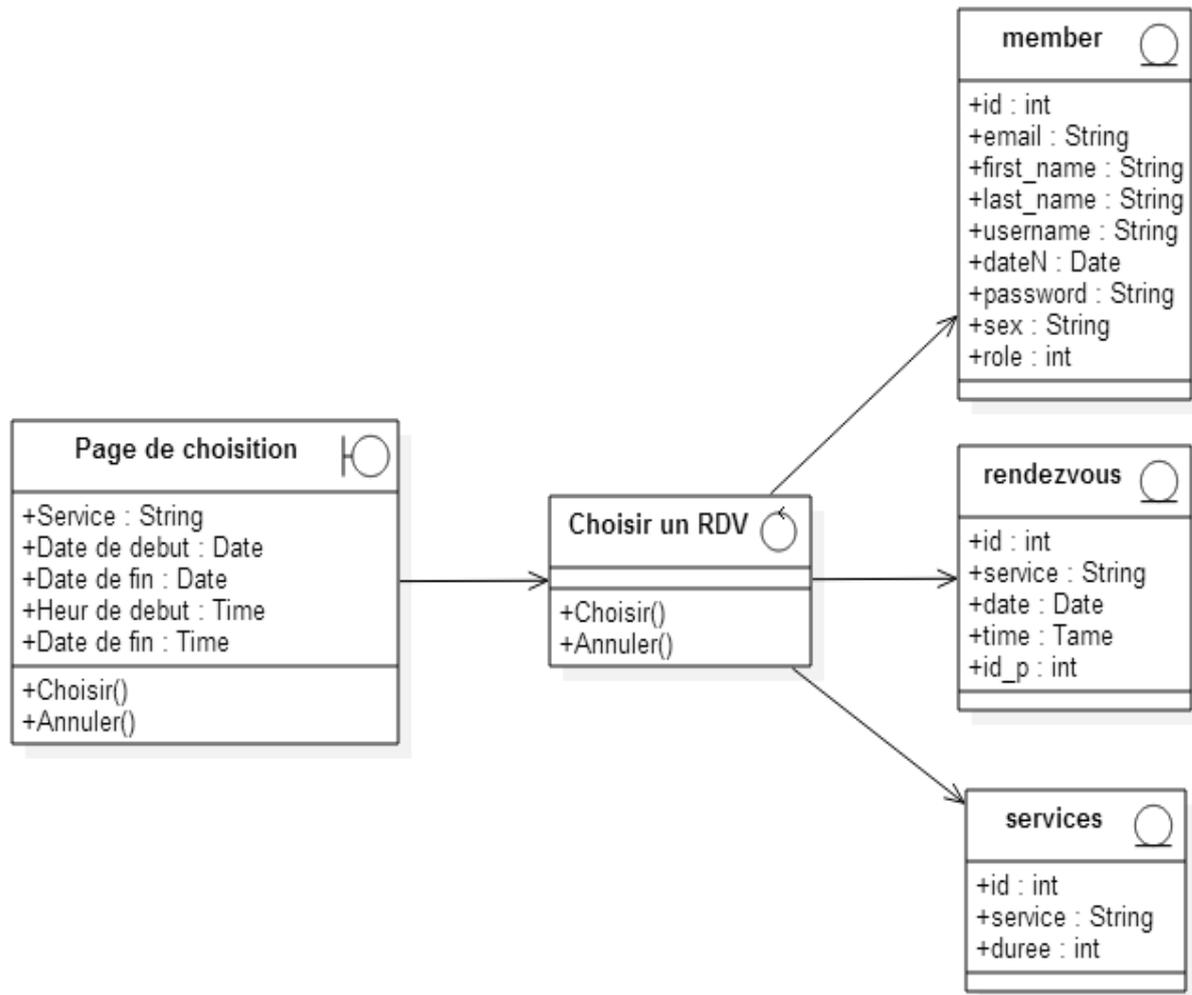


Figure IV.6. Diagramme de classe participante « Choisir un RDV »

### 3.6- Le cas d'utilisation « Modifier un RDV »

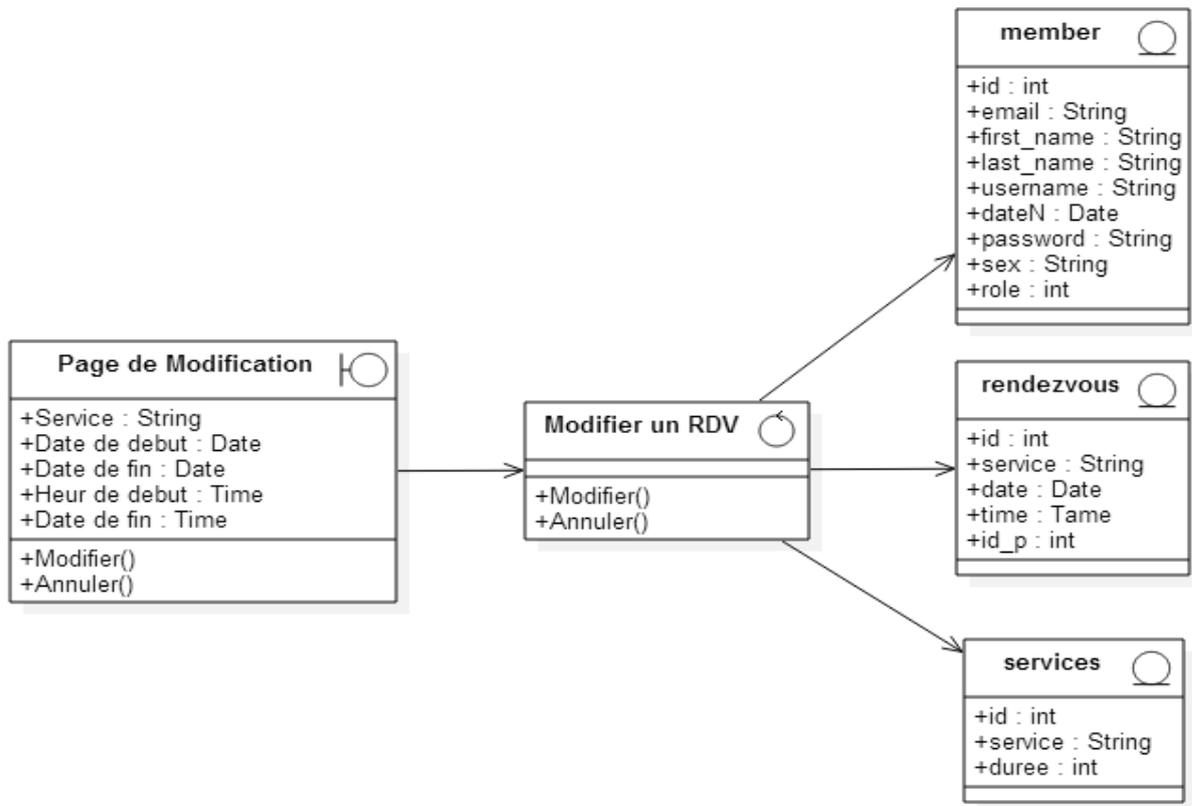


Figure IV.7. Diagramme de classe participante « Modifier un RDV »

### 3.7- Le cas d'utilisation « Annuler un RDV »

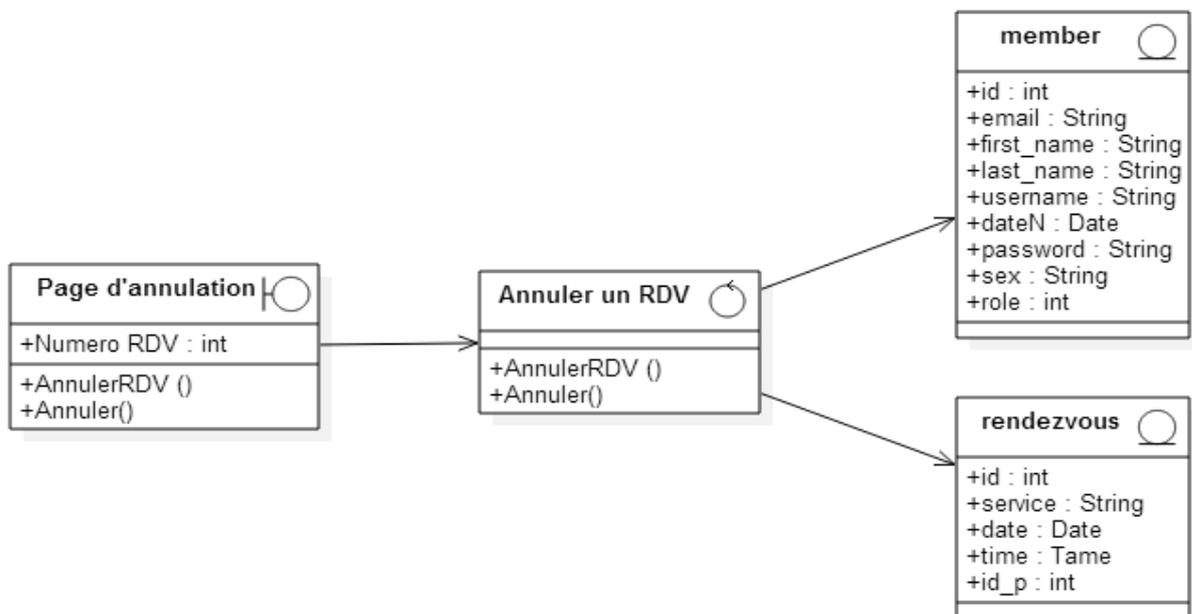


Figure IV.8. Diagramme de classe participante « Annuler un RDV »

### 3.8- Le cas d'utilisation « Ajouter un service »

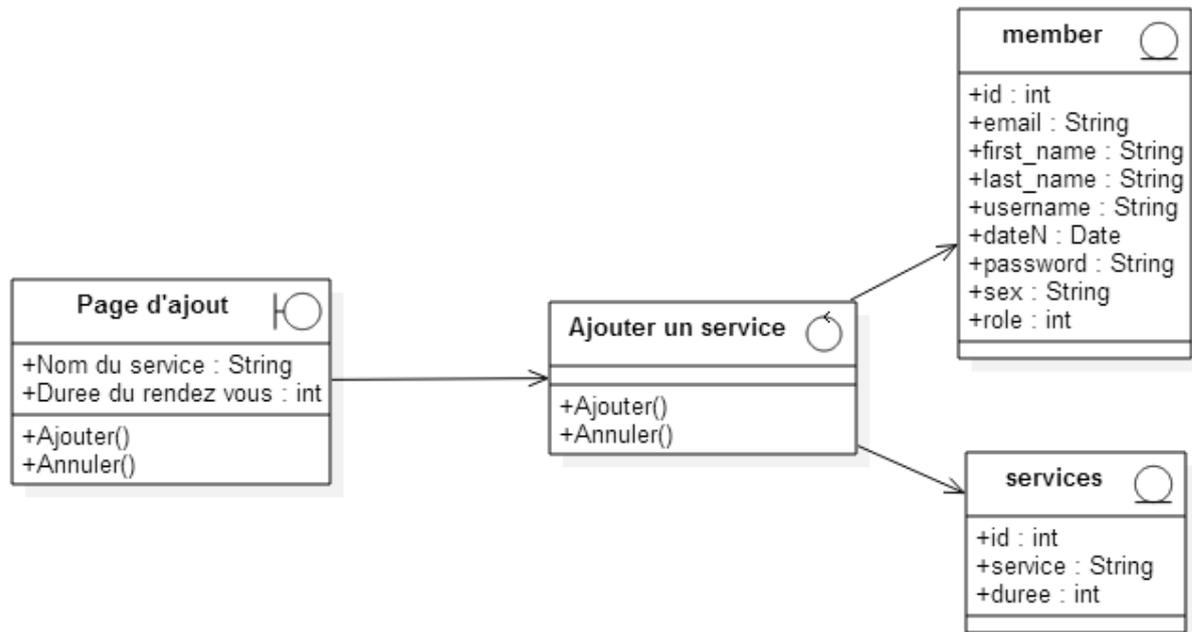


Figure IV.9. Diagramme de classe participante « Ajouter un service »

### 3.9- Le cas d'utilisation « Modifier un service »

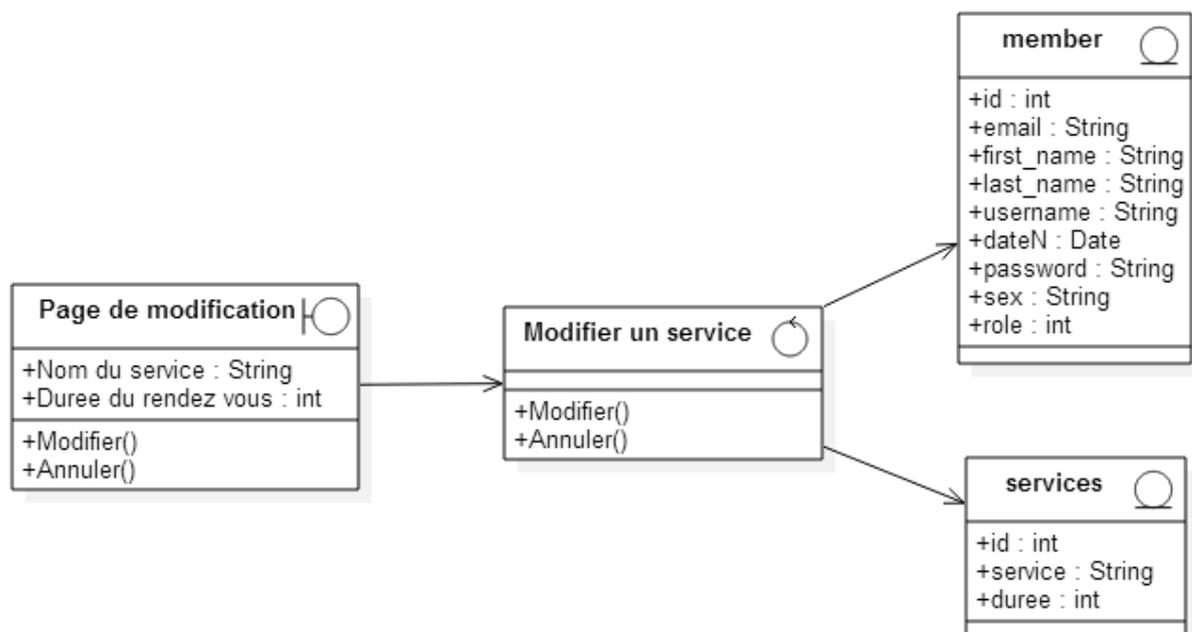
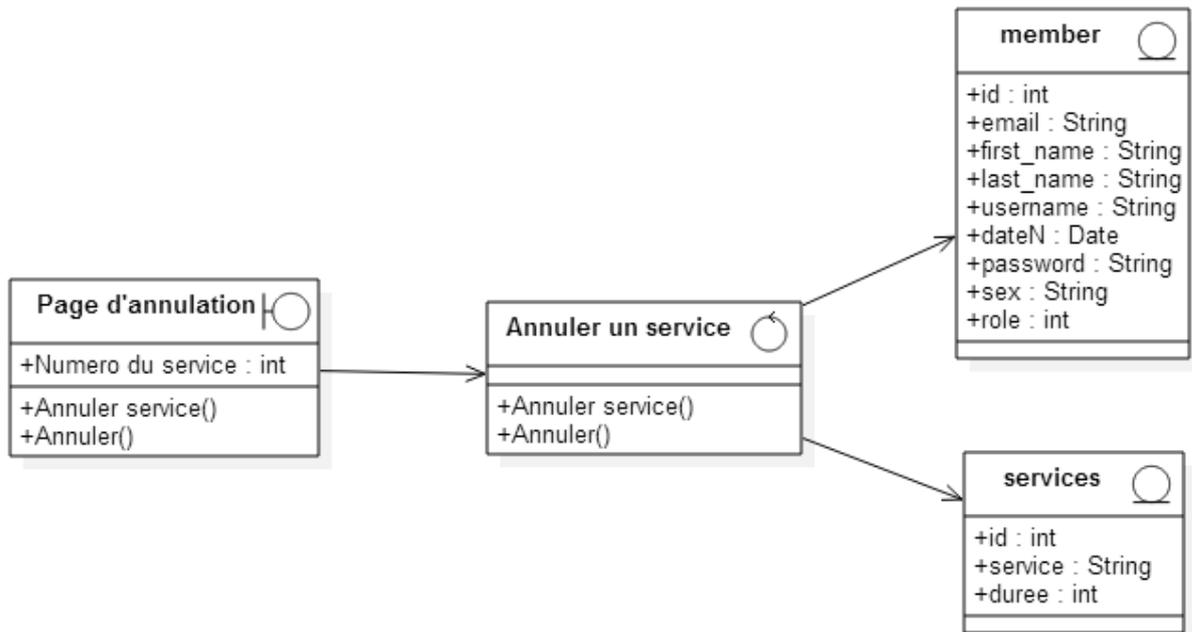


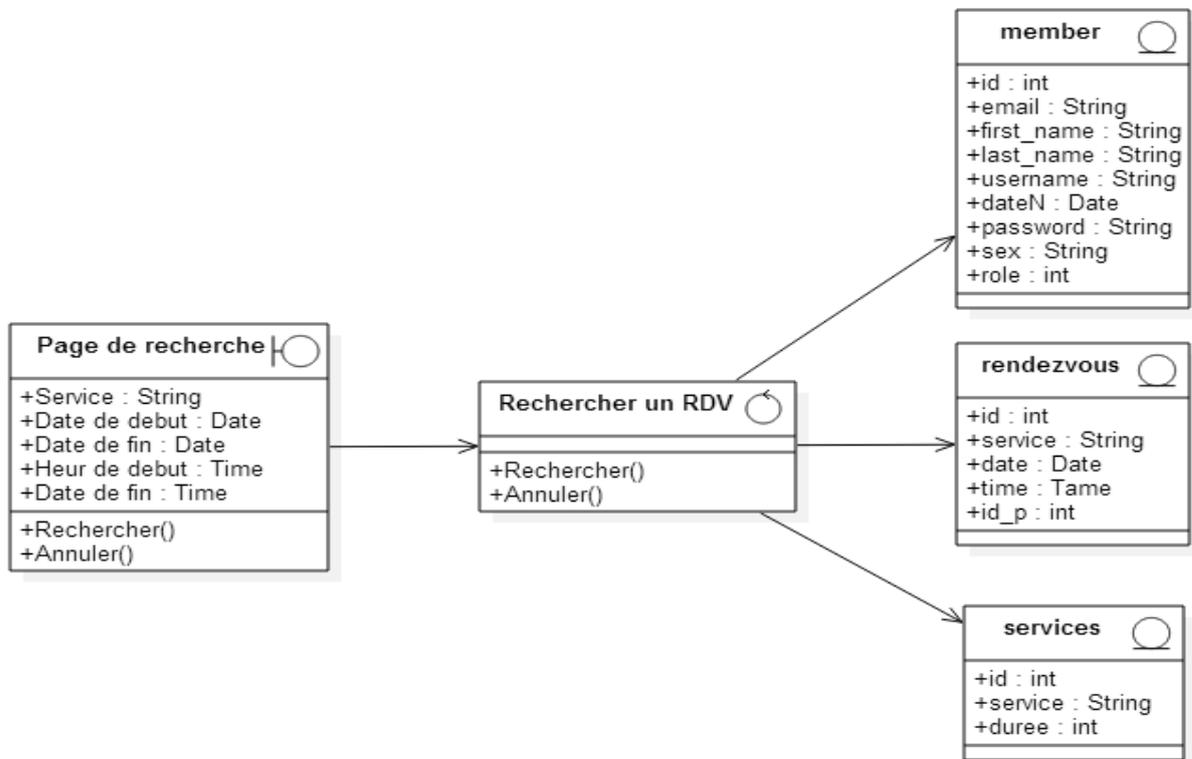
Figure IV.10. Diagramme de classe participante « Modifier un service »

**3.10- Le cas d'utilisation « Annuler un service »**



**Figure IV.11. Diagramme de classe participante « Annuler un service »**

**3.11- Le cas d'utilisation « Rechercher un RDV »**



**Figure IV.12. Diagramme de classe participante « Rechercher un RDV »**

### 3.12- Le cas d'utilisation « Rechercher un Patient »

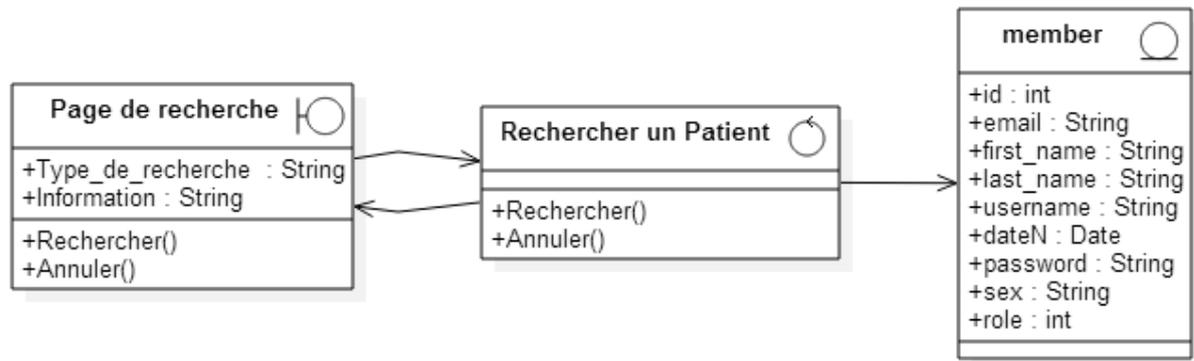


Figure IV.13. Diagramme de classe participante « Rechercher un Patient »

## 7. Conclusion

Dans cette étape nous avons exprimé clairement les objectifs attendus du futur système à concevoir, ainsi que l'analyse associée à chaque cas d'utilisation. Dans le chapitre suivant nous allons détailler la prochaine étape « la conception ».

# Chapitre V : Phase de conception



## 1. Introduction

Dans ce chapitre nous allons élaborer les diagrammes d'interactions qui nous permettent d'attribuer précisément les responsabilités de comportement aux classes d'analyse du diagramme de classes participantes. Parallèlement, une première ébauche de la vue statique de conception (c'est-à-dire du diagramme de classes de conception) sera construite et complétée par le diagramme de classe.

## 2. Les diagrammes d'interaction

### 2.1- Le cas d'utilisation « S'identifier »

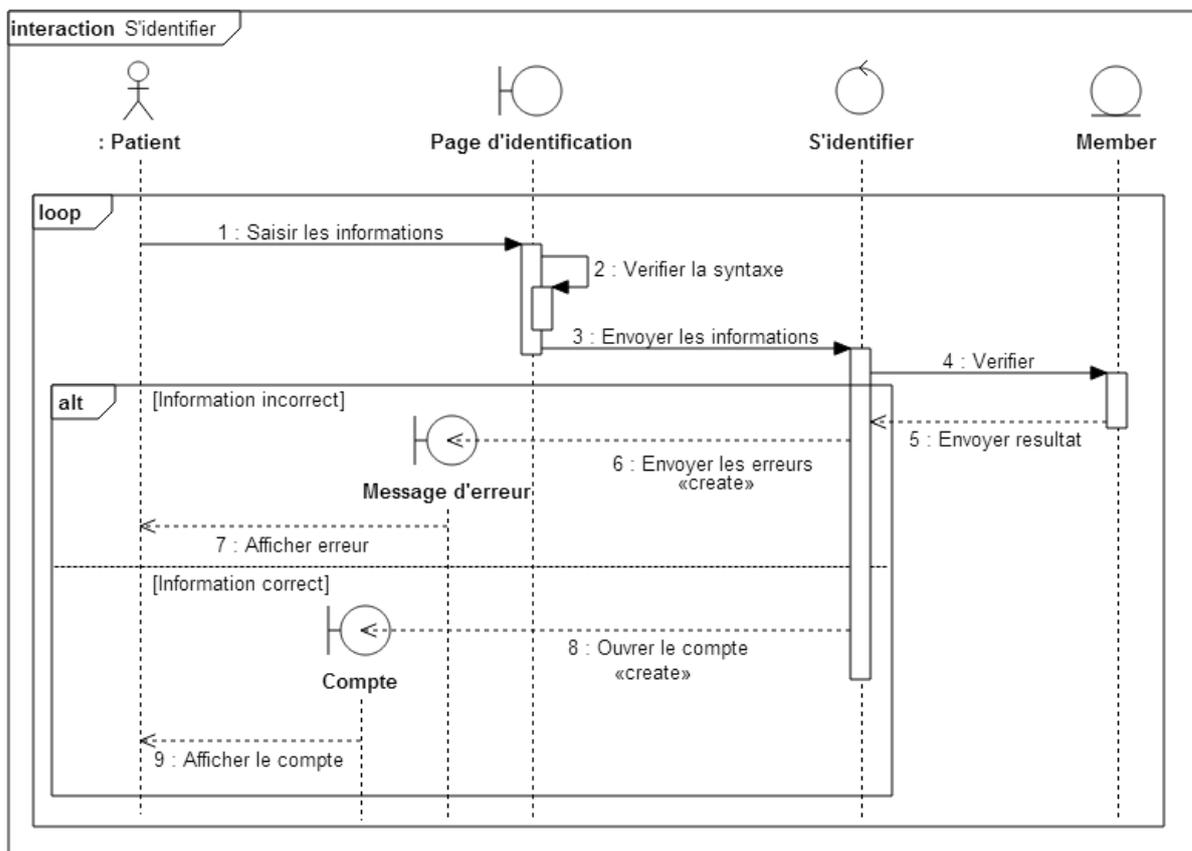


Figure V.1. Diagramme d'interaction« S'identifier »

### 2.2- Le cas d'utilisation « S'inscrire »

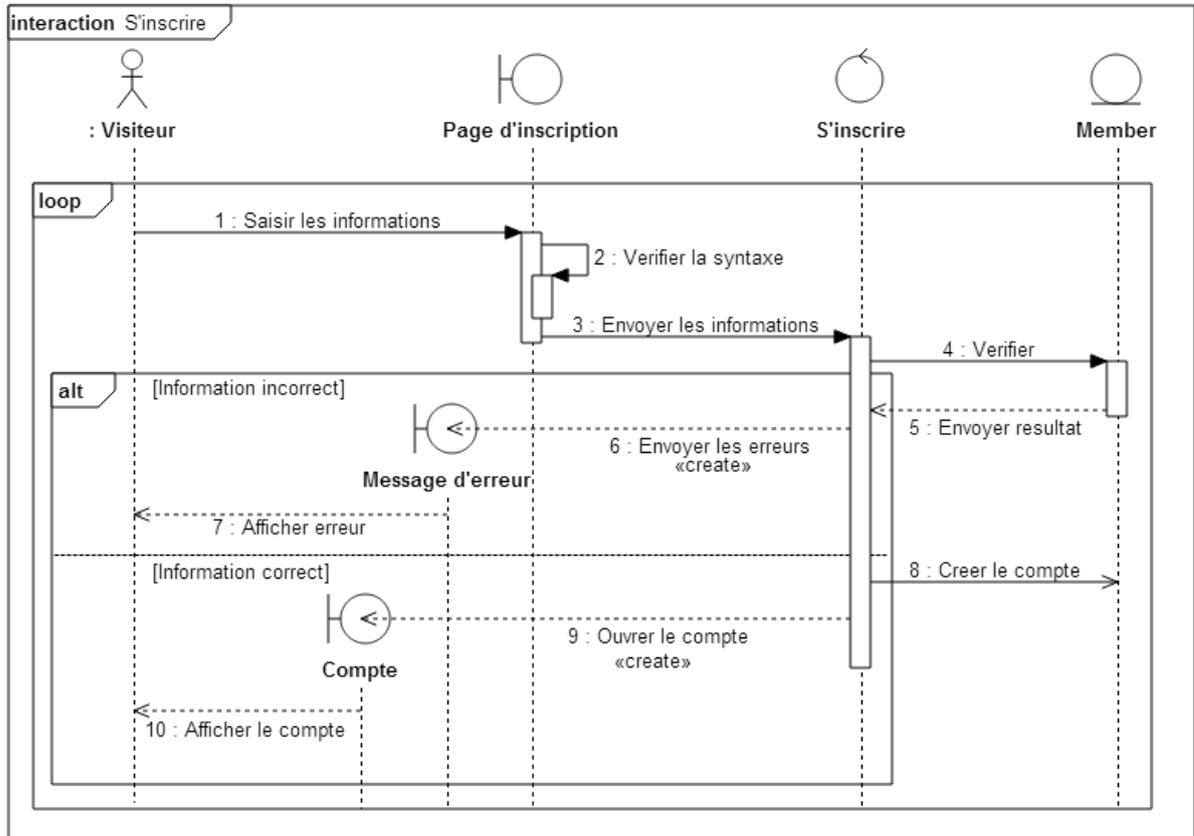
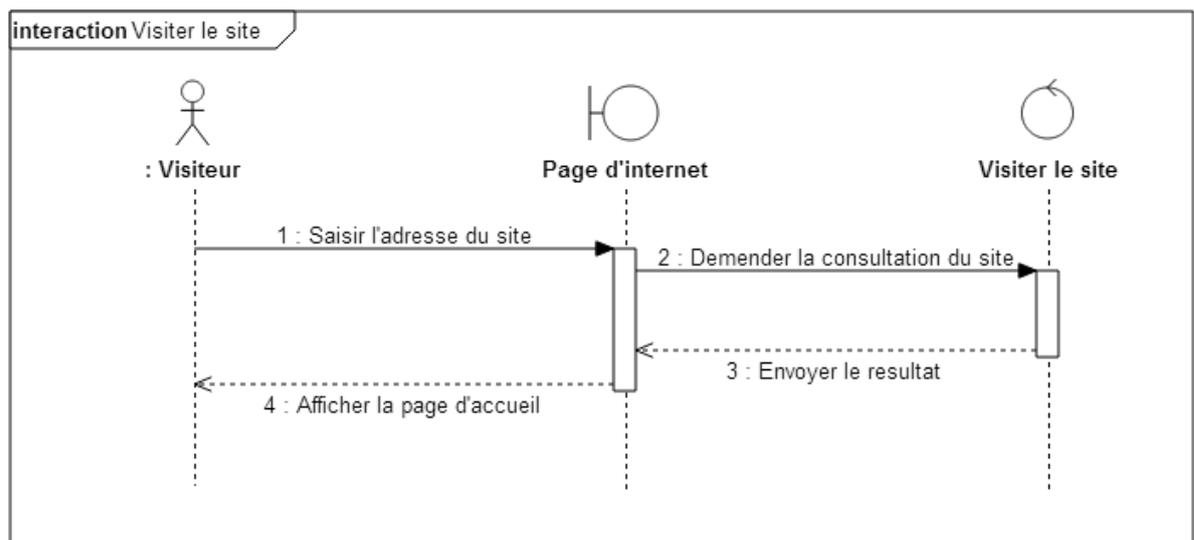


Figure V.2. Diagramme d'interaction« S'inscrire »

### 2.3- Le cas d'utilisation « Visiter le site »



FigureV.3. Diagramme d'interaction« Visiter le site »

### 2.4- Le cas d'utilisation « Tester la disponibilité des RDV »

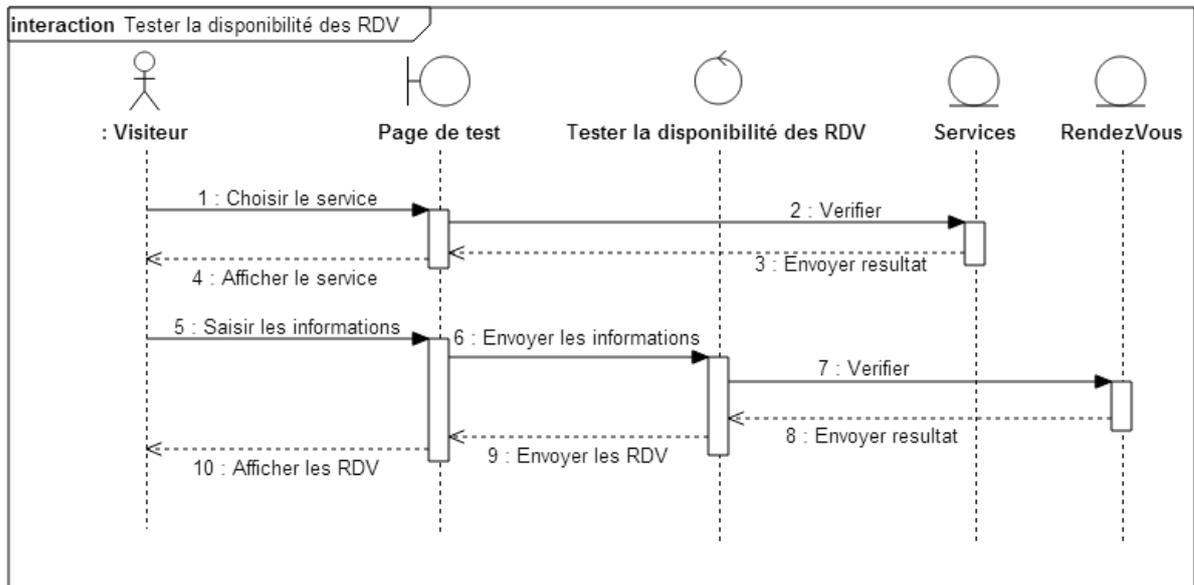


Figure V.4. Diagramme d'interaction« Tester la disponibilité des RDV »

### 2.5- Le cas d'utilisation « Choisir un RDV »

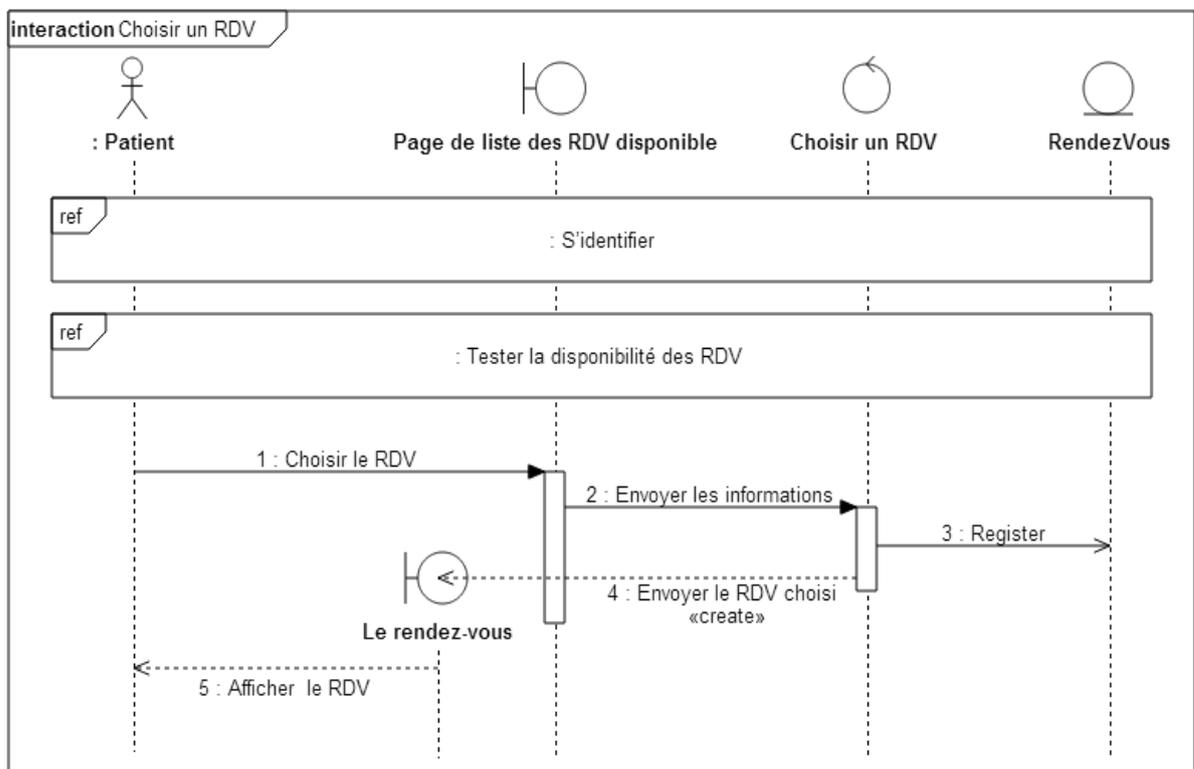
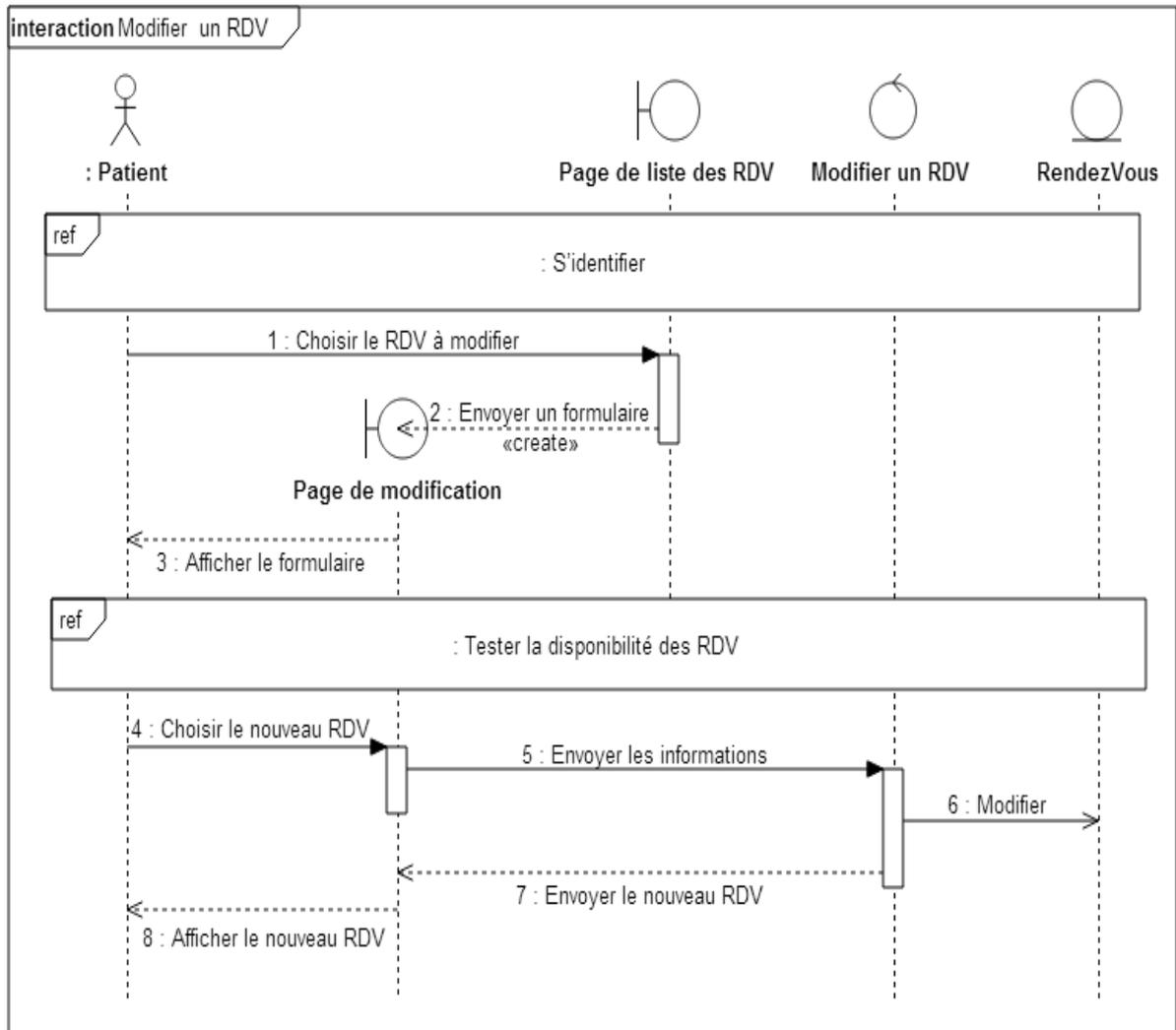


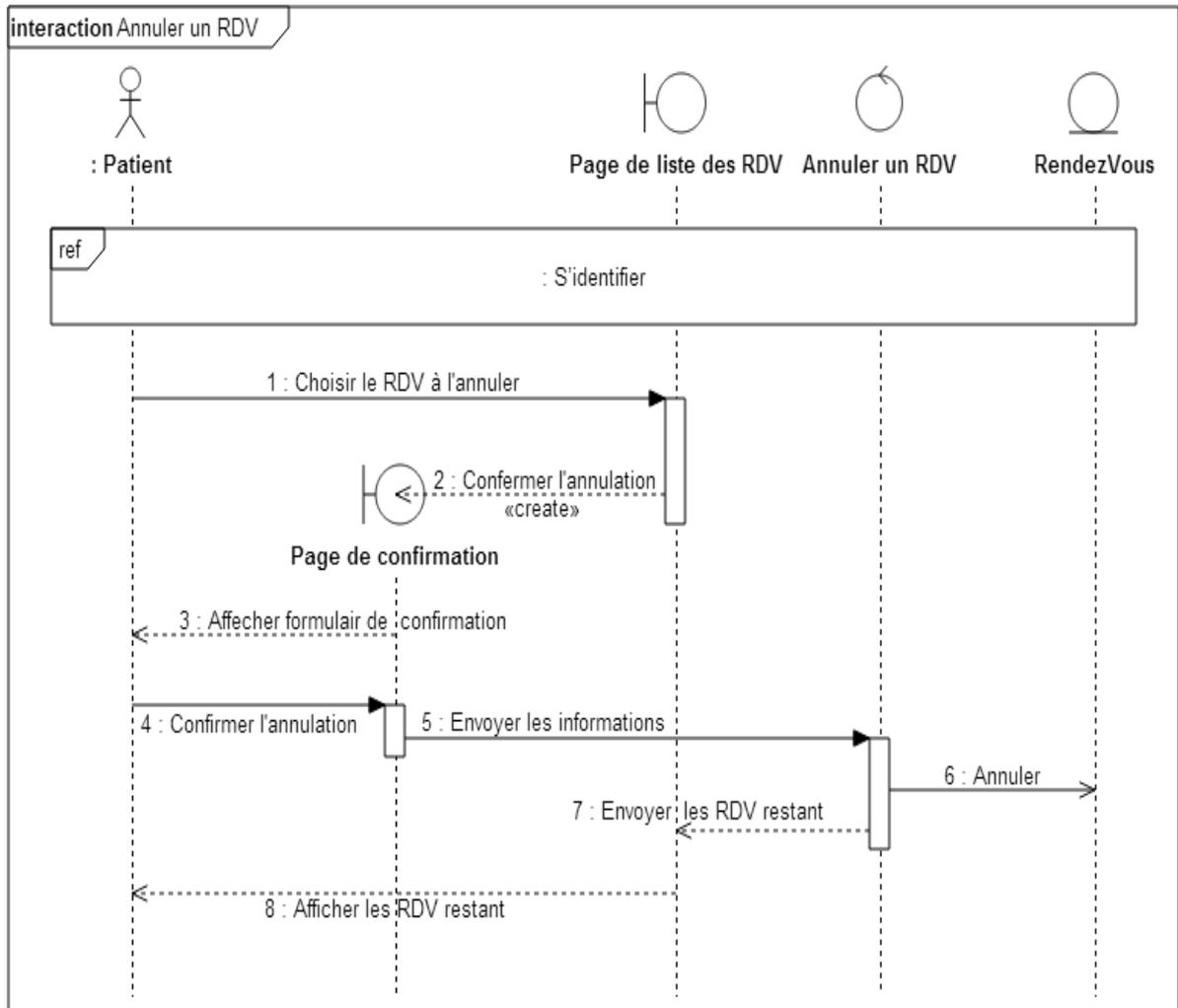
Figure V.5. Diagramme d'interaction« Choisir un RDV »

**2.6- Le cas d'utilisation « Modifier un RDV »**



**Figure V.6. Diagramme d'interaction« Modifier un RDV »**

**2.7- Le cas d'utilisation « Annuler un RDV »**



**Figure V.7. Diagramme d'interaction« Annuler un RDV »**

### 2.8- Le cas d'utilisation « Ajouter un service »

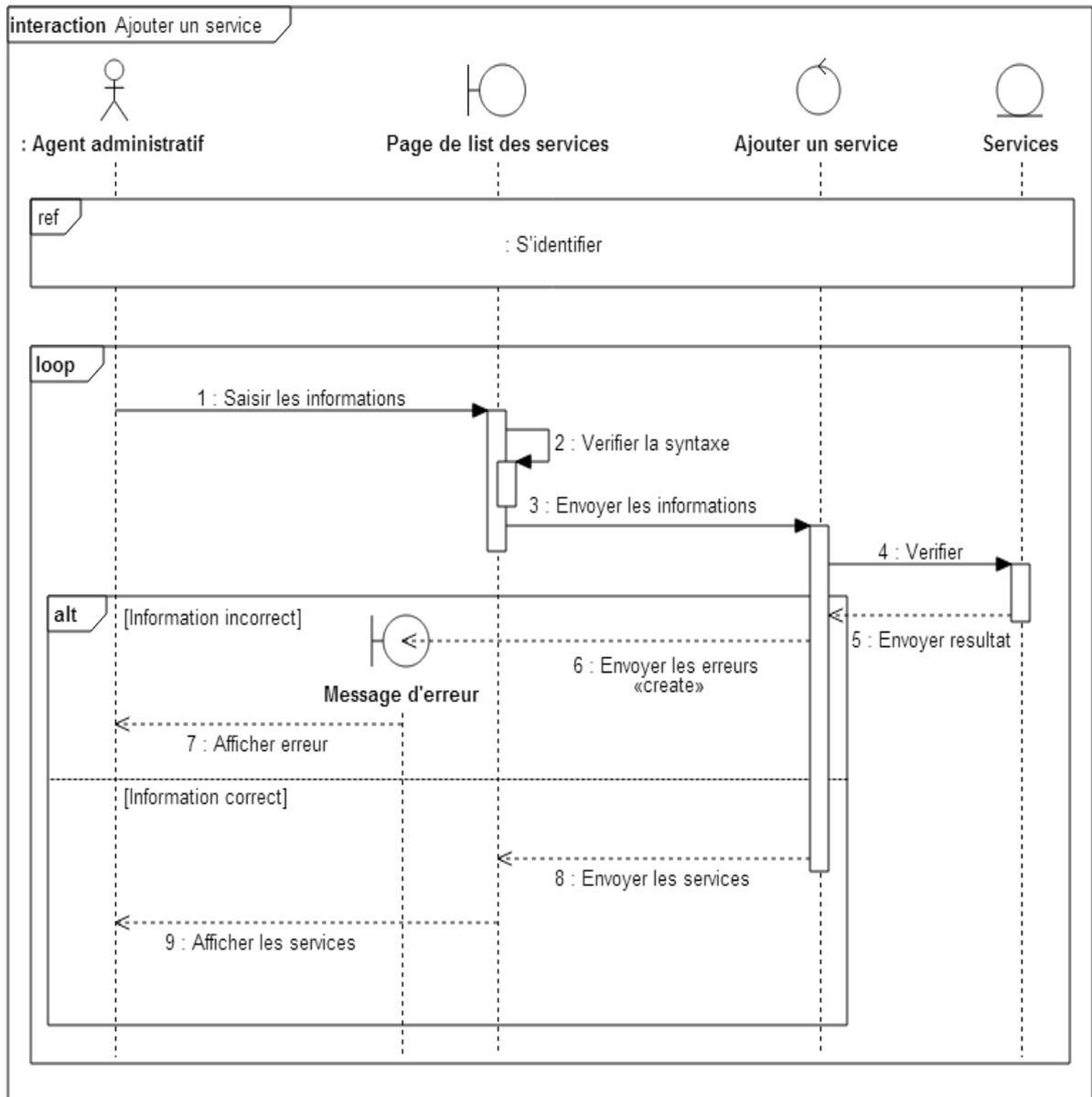
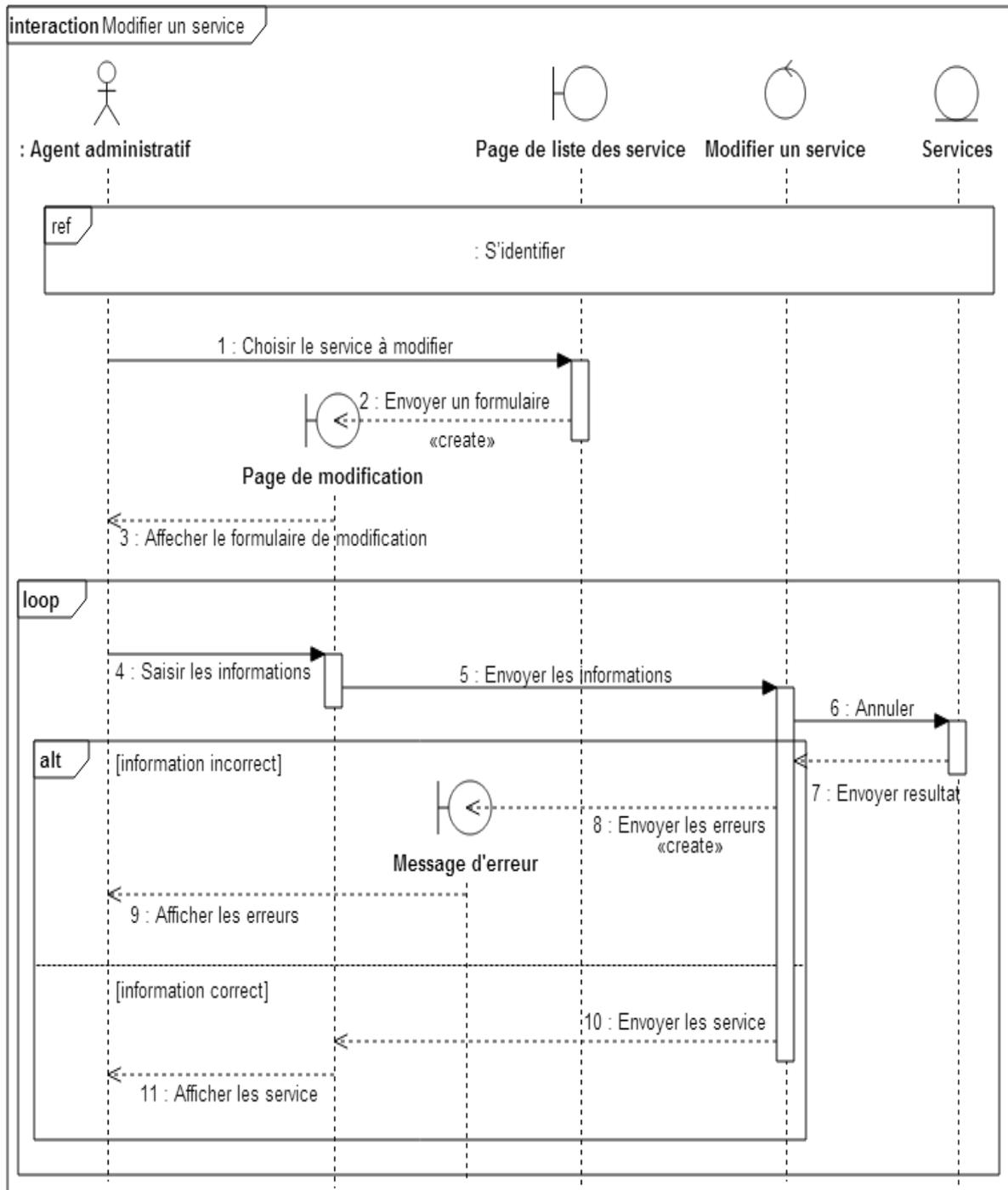


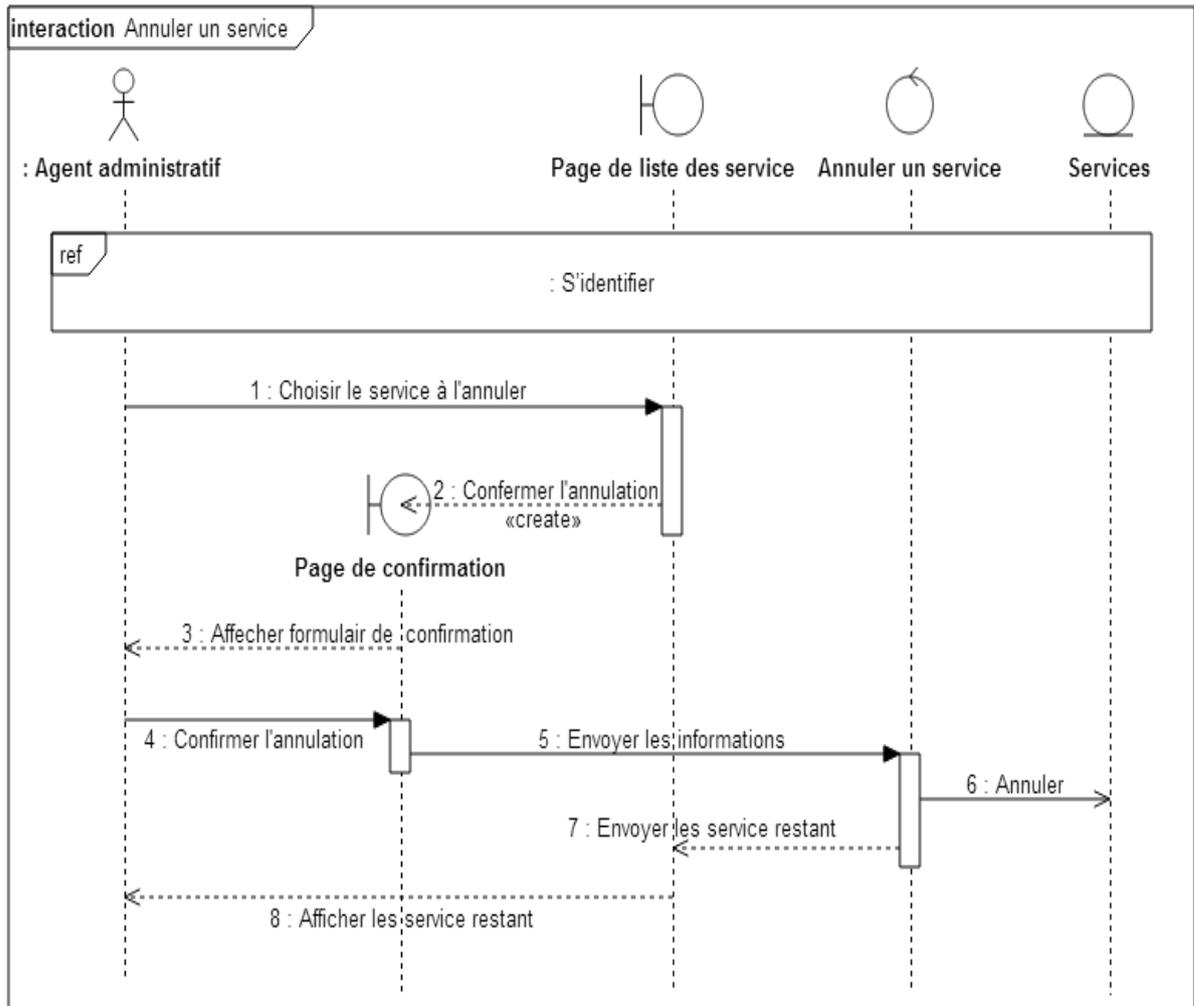
Figure V.8. Diagramme d'interaction « Ajouter un service »

**2.9- Le cas d'utilisation « Modifier un service »**



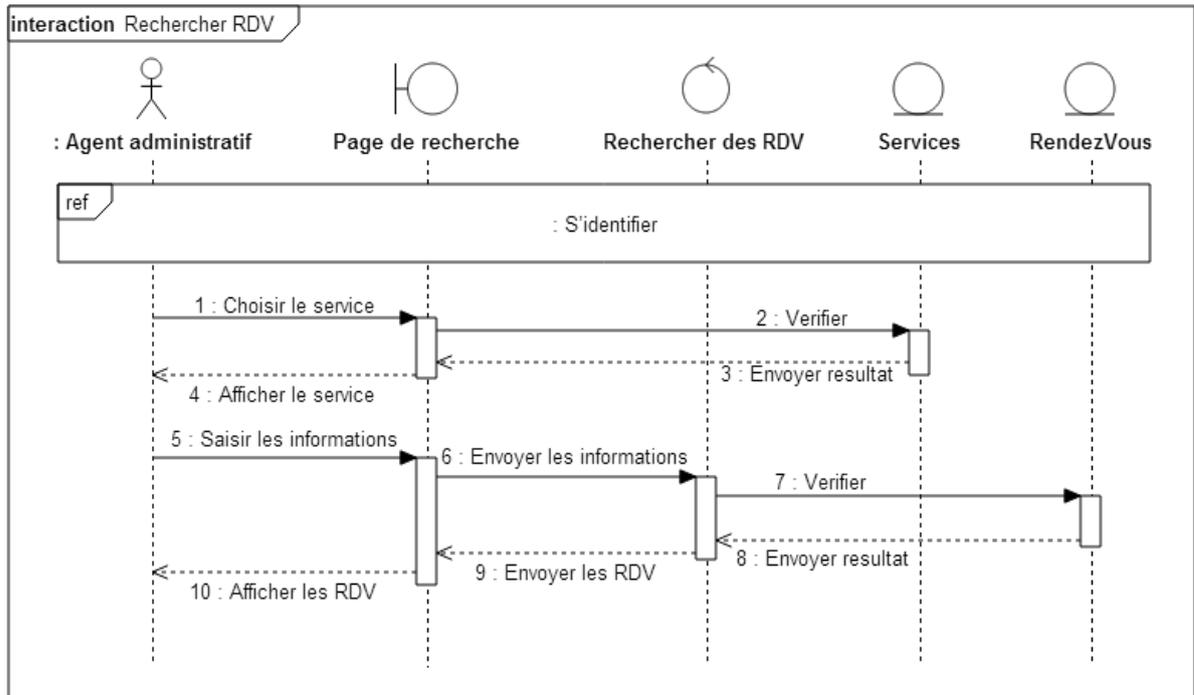
**Figure V.9. Diagramme d'interaction« Modifier un service »**

**2.10- Le cas d'utilisation « Annuler un service »**



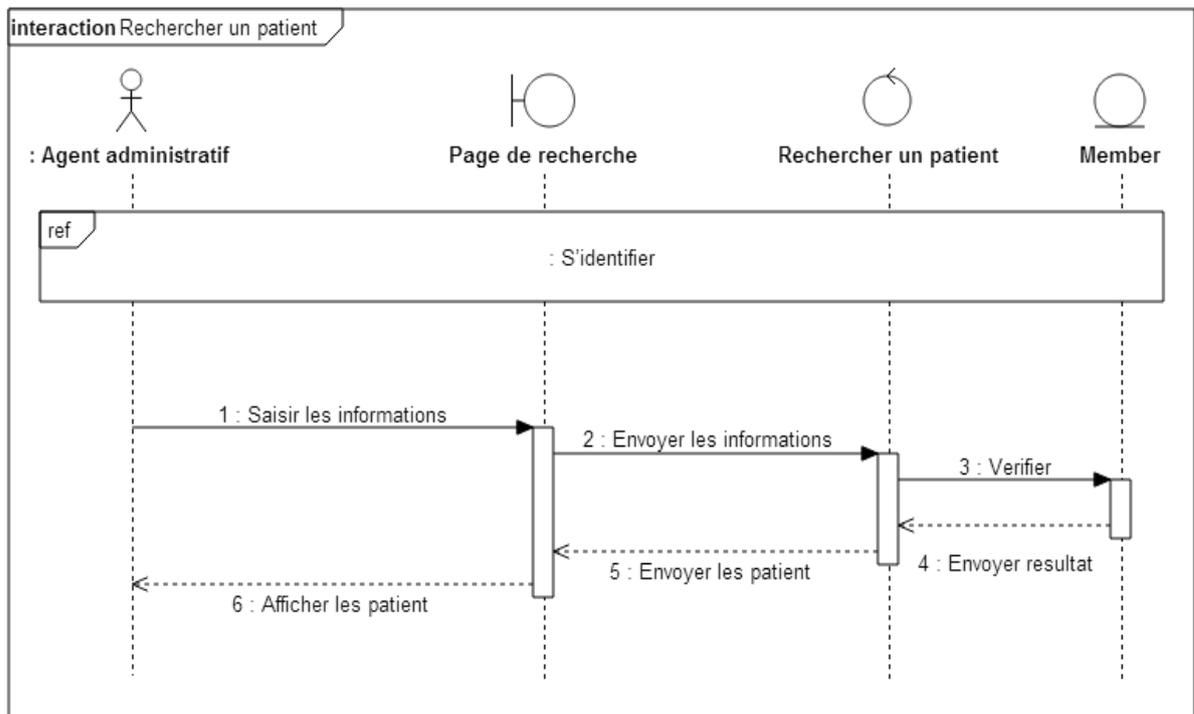
**Figure V.10. Diagramme d'interaction« Annuler un service »**

**2.11- Le cas d'utilisation « Rechercher un RDV »**



**Figure V.11. Diagramme d'interaction« Rechercher un RDV »**

**2.12- Le cas d'utilisation « Rechercher un Patient »**



**Figure V.12. Diagramme d'interaction« Rechercher un Patient »**

### 3. Les diagrammes de classe de conception

#### 3.1- Le cas d'utilisation « S'identifier »

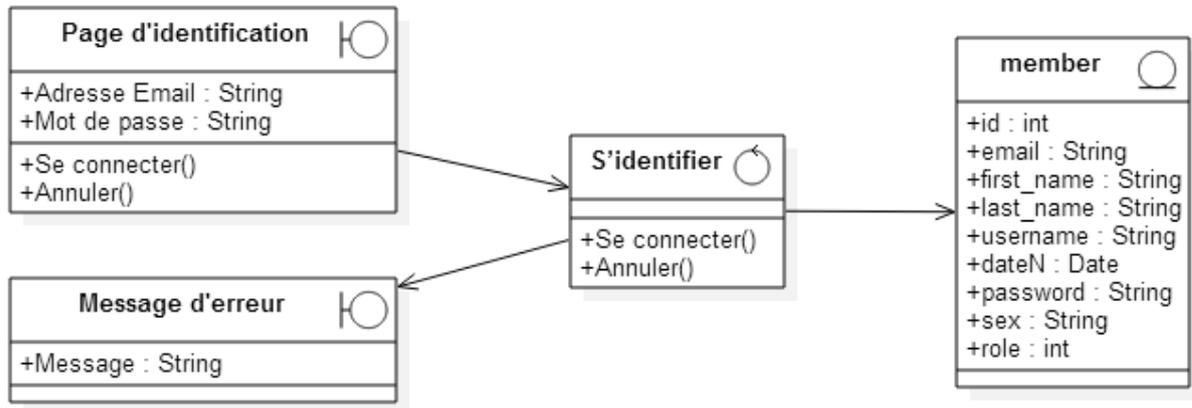


Figure V.13. Diagramme de classe de conception « S'identifier »

#### 3.2- Fiche descriptive du cas d'utilisation « S'inscrire »

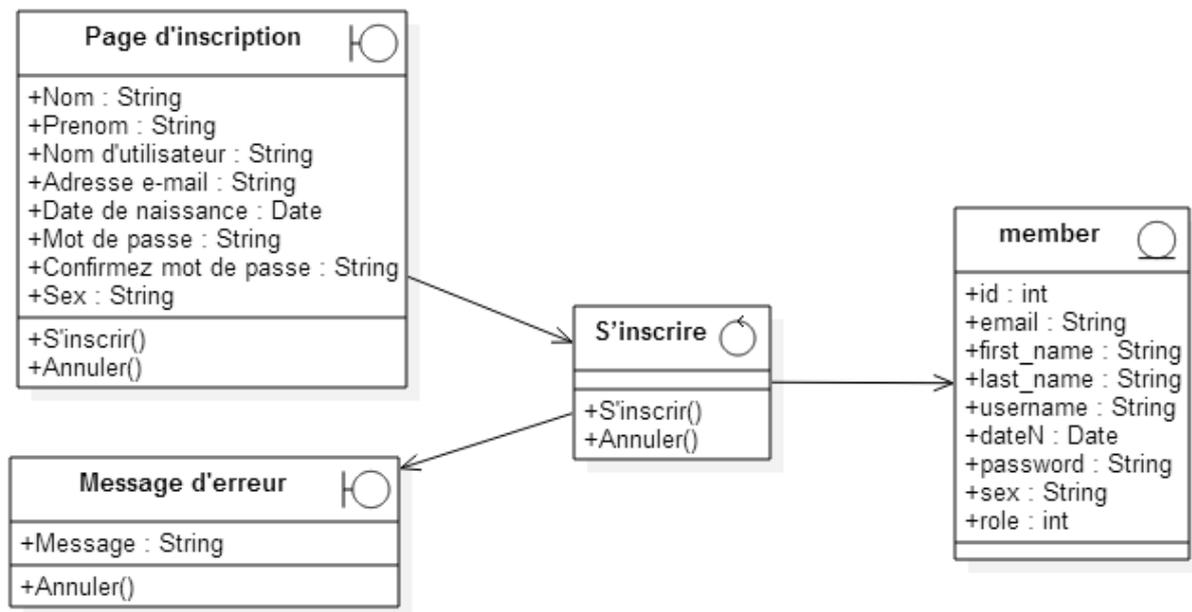


Figure V.14. Diagramme de classe de conception « S'inscrire »

### 3.3- Le cas d'utilisation « Visiter le site »

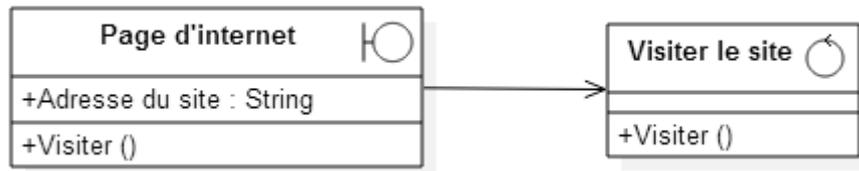


Figure V.15. Diagramme de classe de conception « Visiter le site »

### 3.4- Le cas d'utilisation « Tester la disponibilité des RDV »

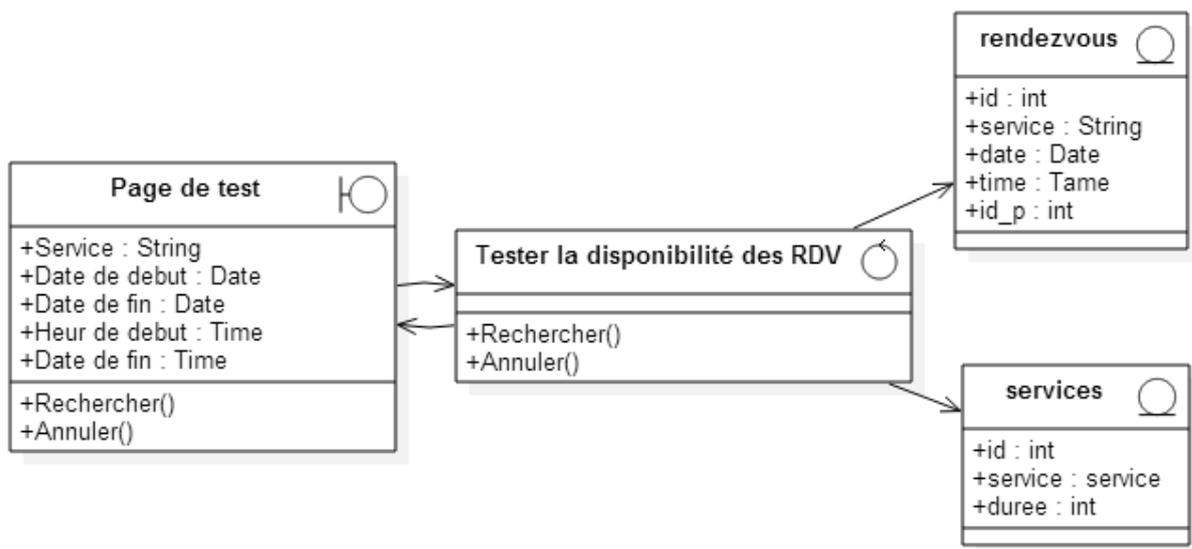


Figure V.16. Diagramme de classe de conception « Tester la disponibilité des RDV »

### 3.5- Le cas d'utilisation « Choisir un RDV »

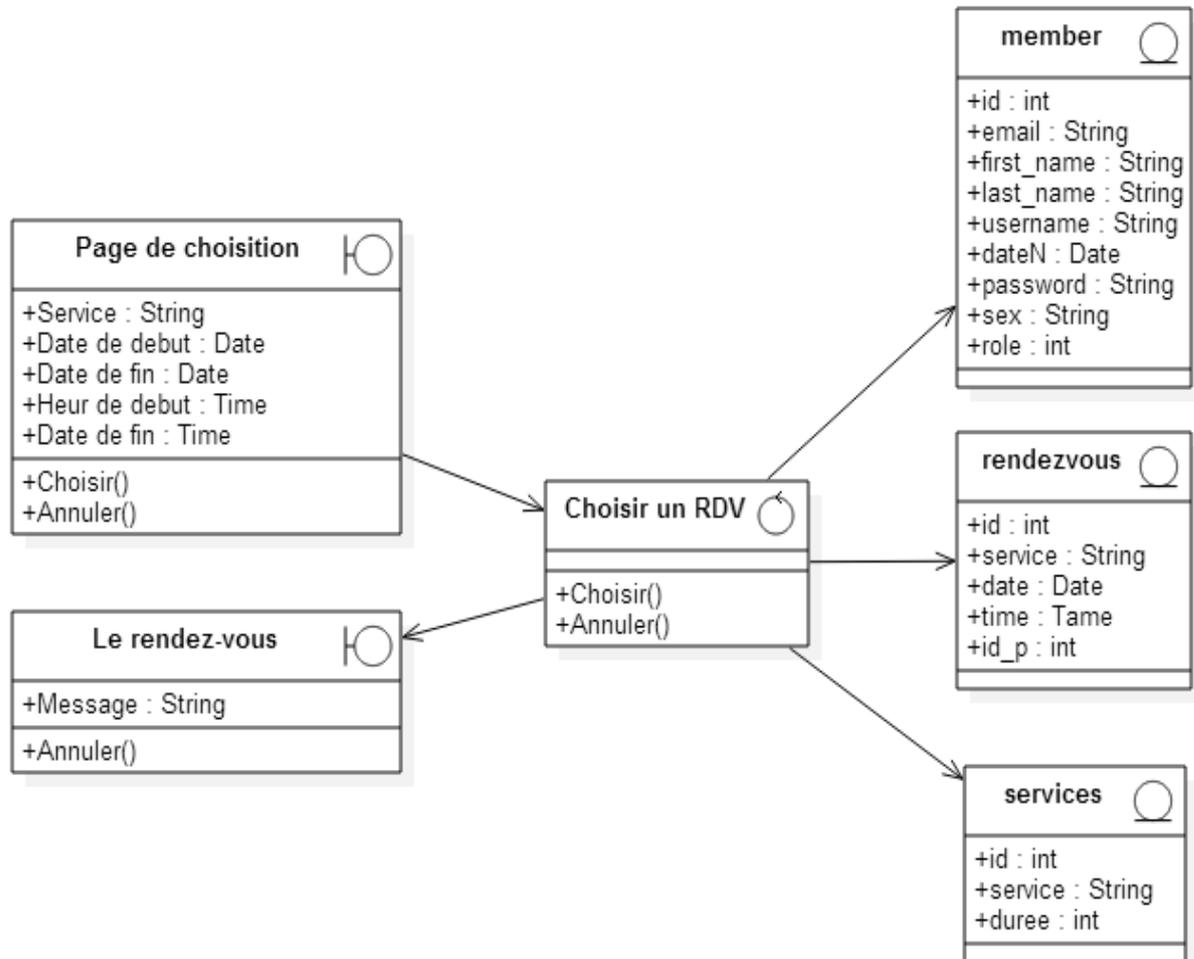


Figure V.17. Diagramme de classe de conception« Choisir un RDV »

### 3.6- Le cas d'utilisation « Modifier un RDV »

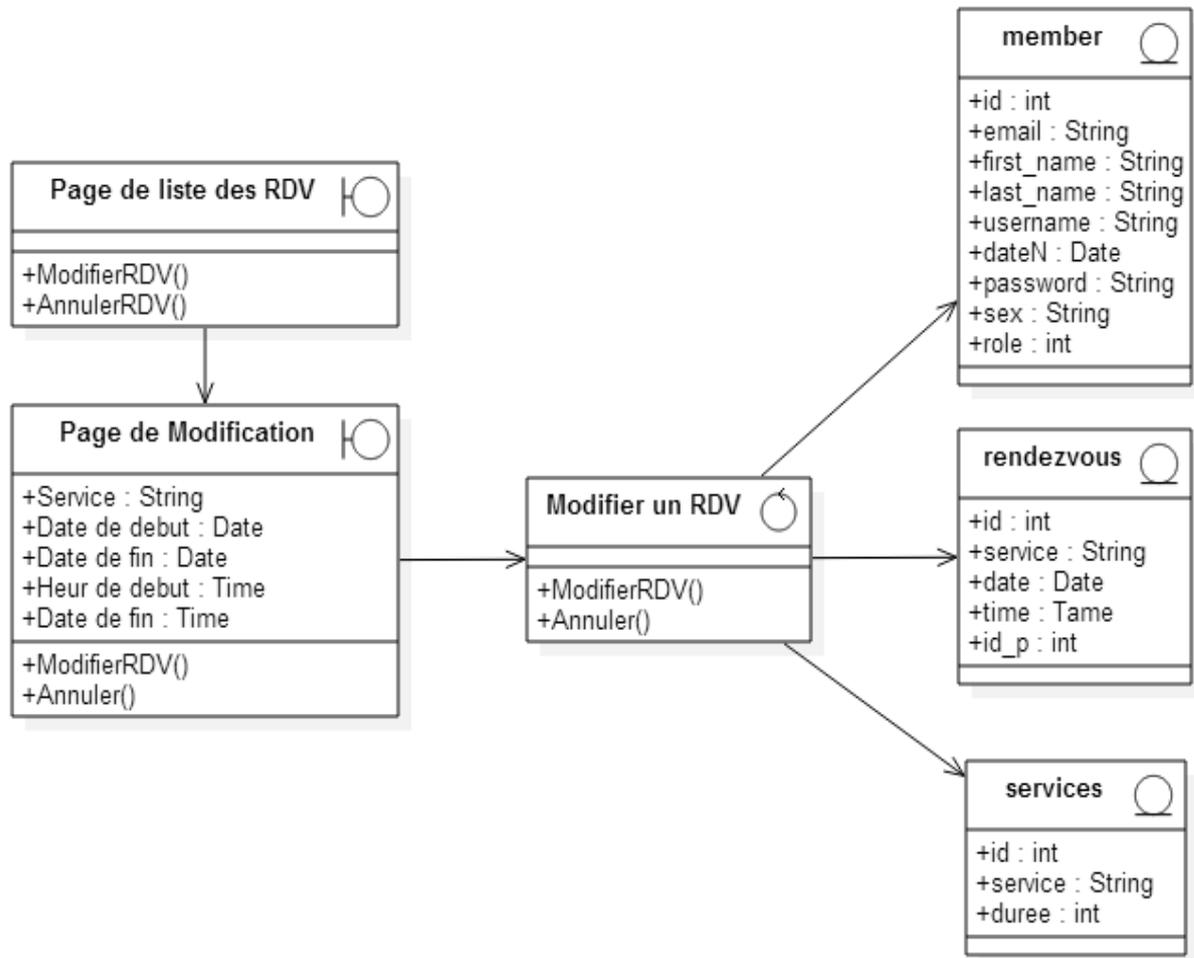


Figure V.18. Diagramme de classe de conception « Modifier un RDV »

### 3.7- Le cas d'utilisation « Annuler un RDV »

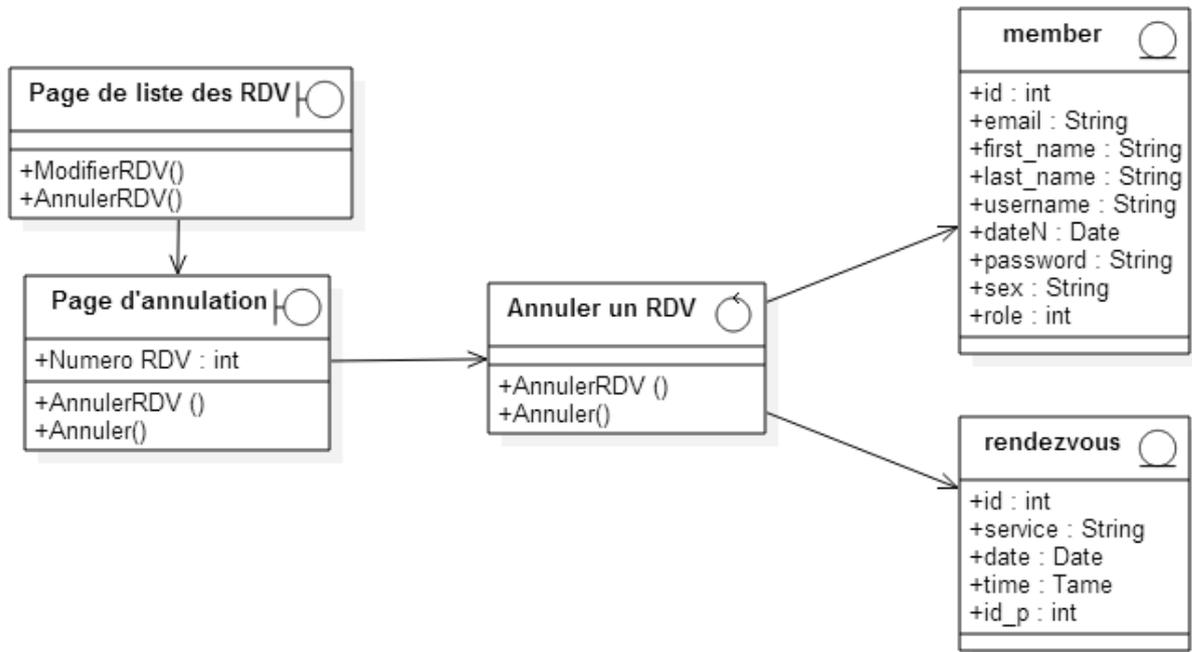


Figure V.19. Diagramme de classe de conception « Annuler un RDV »

### 3.8- Le cas d'utilisation « Ajouter un service »

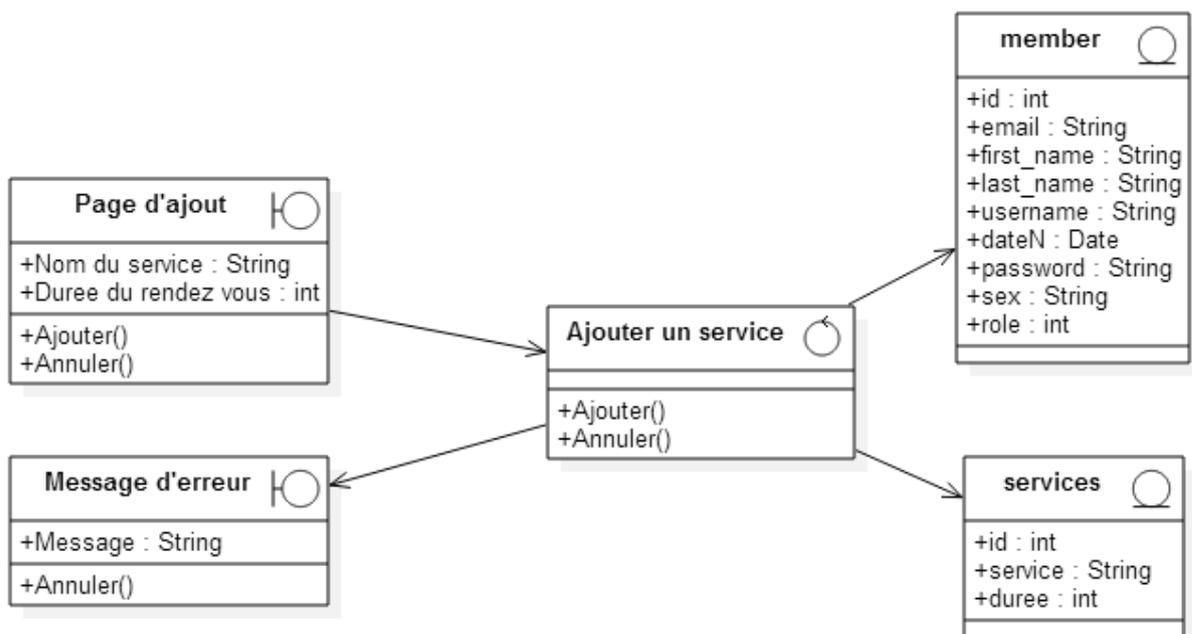


Figure V.20. Diagramme de classe de conception« Ajouter un service »

### 3.9- Le cas d'utilisation « Modifier un service »

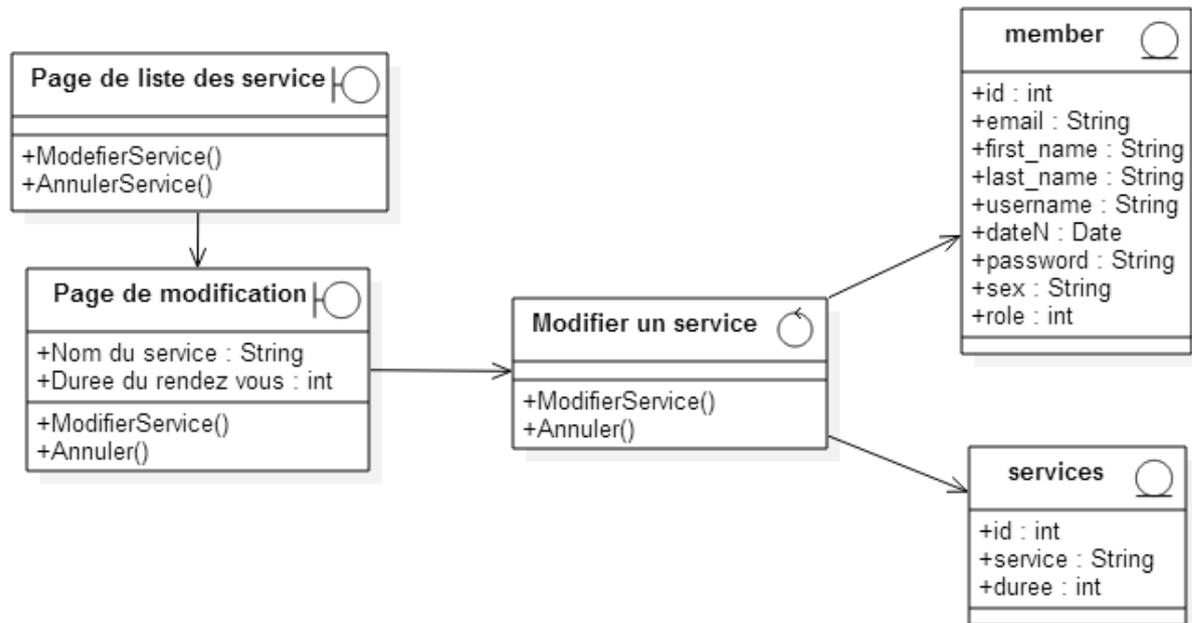


Figure V.21. Diagramme de classe de conception« Modifier un service »

### 3.10- Le cas d'utilisation « Annuler un service »

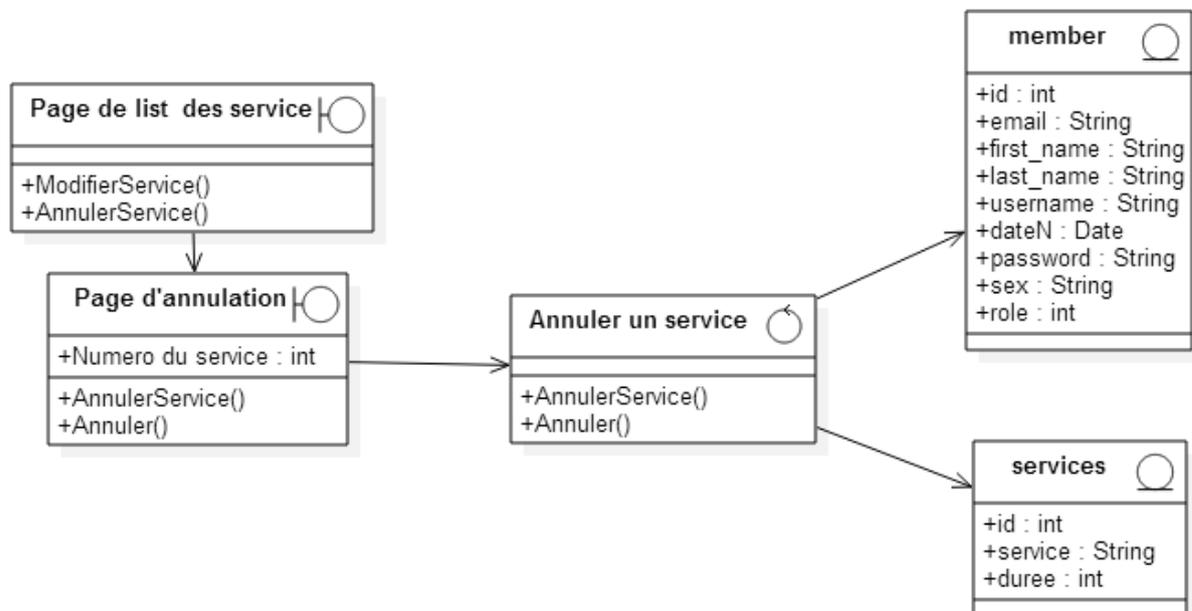
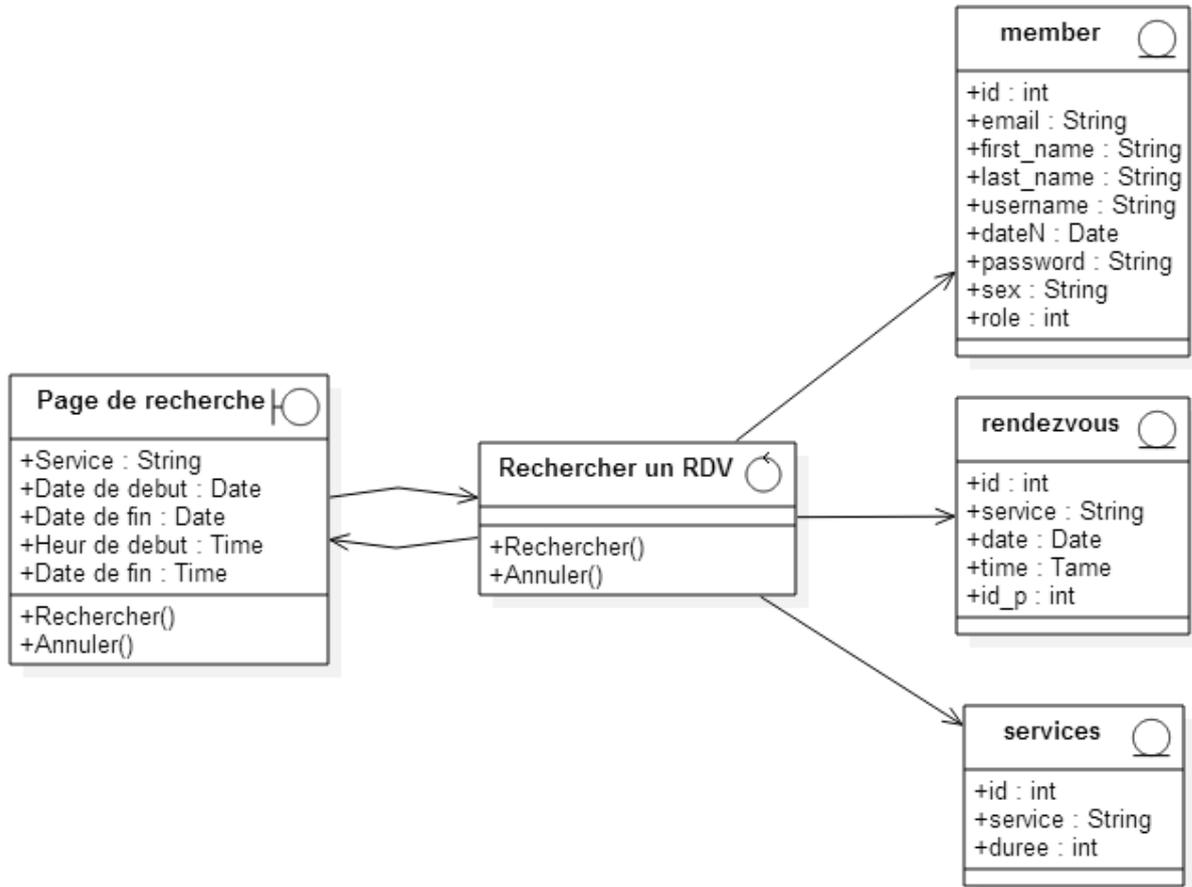


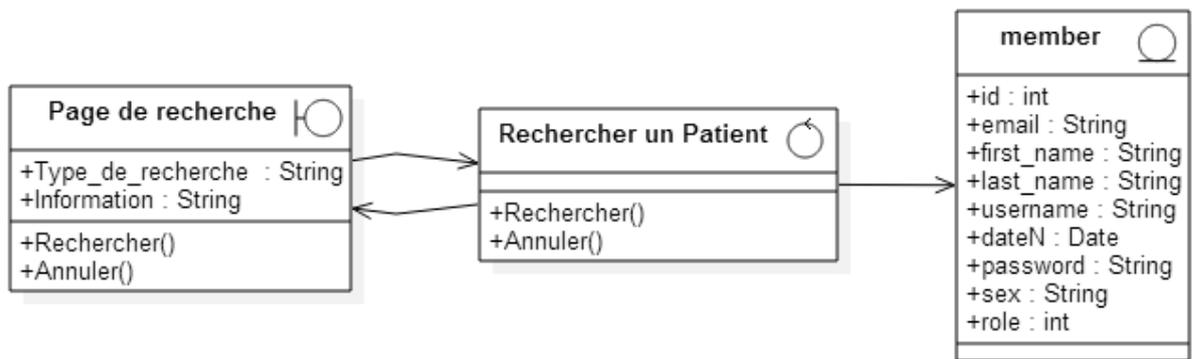
Figure V.22. Diagramme de classe de conception« Annuler un service »

**3.11- Le cas d'utilisation « Rechercher un RDV »**



**Figure V.23. Diagramme de classe de conception« Rechercher un RDV »**

**3.12- Le cas d'utilisation « Rechercher un Patient »**



**Figure V.24. Diagramme de classe de conception« Rechercher un Patient »**

#### 4. Diagramme de classe

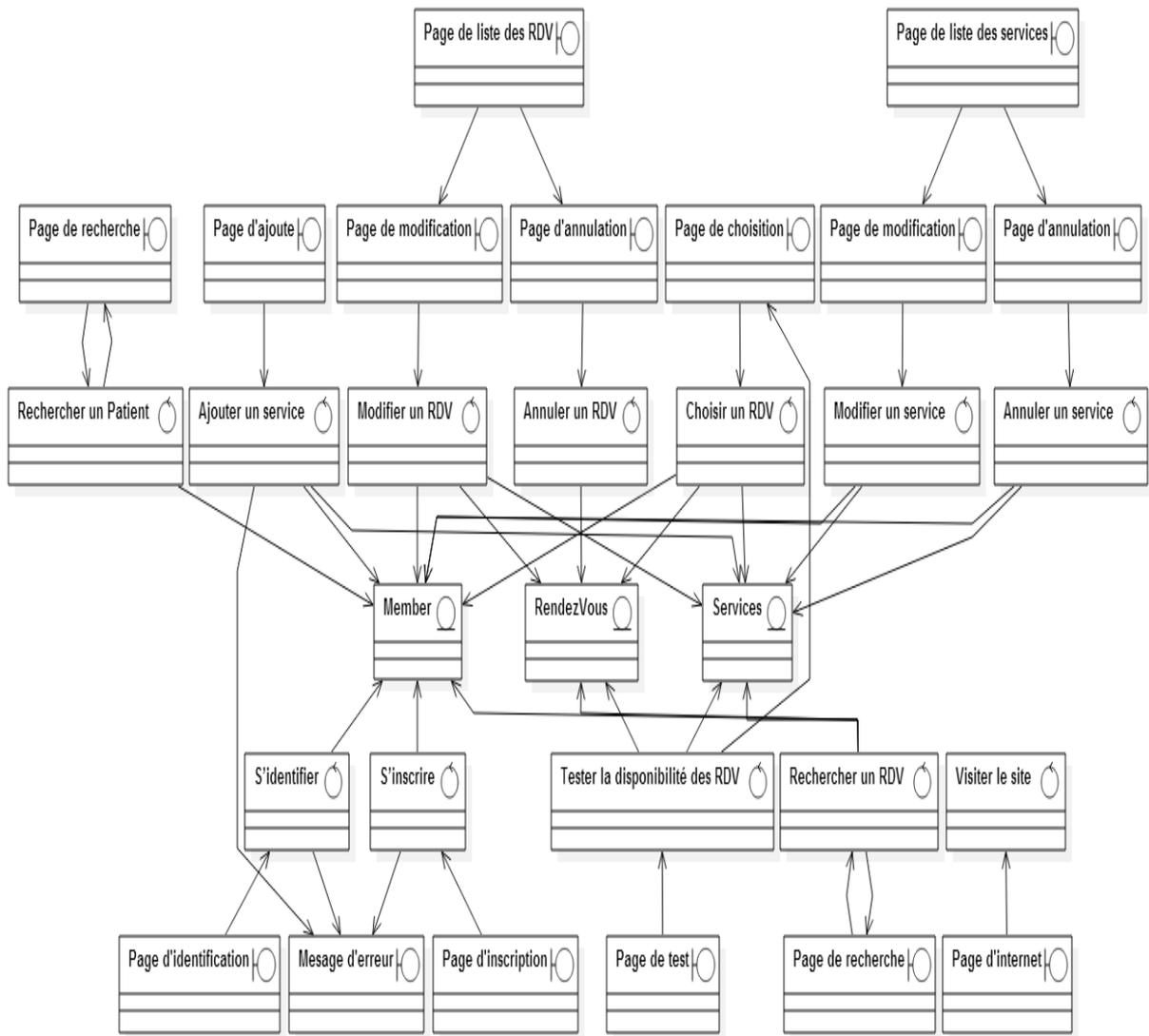


Figure V.25. Diagramme de classe.

#### 5. Conclusion

Dans ce chapitre, nous avons utilisé les diagrammes d'interaction et les diagrammes de classe de conception qu'ils sont appartenent à une démarche simplifiée de développement logiciel issue de l'UP. La phase de conception a intégré des nouveaux classes et méthodes aux modèles d'analyse et la manière de naviguer entre ces classes de conception.

# Chapitre VI : L'implémentation



## 1. Introduction

La réalisation d'un site web dynamique a besoin de plusieurs logiciels et outils. Pour cela, nous avons choisi quelques outils et langages parmi ceux existants et qui répondent à nos besoins.

Dans ce chapitre, nous allons citer la liste des langages et les outils utilisés pour la réalisation du site tout en expliquant le rôle de chacun.

## 2. Les Langages de programmation

### 2.1. Les langages HTML et CSS

Les langages HTML et CSS sont à la base du fonctionnement de tous les sites web.

```

1 <!DOCTYPE html>
2 <html lang="fr" dir="ltr" class="client-nojs">
3 <head>
4 <meta charset="UTF-8" />
5 <title>Wikipédia, l'encyclopédie libre</title>
6 <meta name="generator" content="MediaWiki 1.25wmf21" />
7 <link rel="alternate" type="application/atom+xml" title="Images du jour de Wikipédia" href="/w/api.php?action=featuredfeed&feed=potd&feedformat=atom" />
8 <link rel="alternate" type="application/atom+xml" title="Liste d'articles labellisés de Wikipédia" href="/w/api.php?action=featured&feed=featured&feedformat=atom" />
9 <link rel="apple-touch-icon" href="//bits.wikimedia.org/apple-touch/wikipedia.png" />
10 <link rel="shortcut icon" href="//bits.wikimedia.org/favicon/wikipedia.ico" />
11 <link rel="search" type="application/opensearchdescription+xml" href="/w/opensearch_desc.php" title="Wikipédia (fr)" />
12 <link rel="EditURI" type="application/rsd+xml" href="//fr.wikipedia.org/w/api.php?action=rsd" />
13 <link rel="alternate" hreflang="x-default" href="/wiki/Wikip%C3%A9dia:Accueil_principal" />
14 <link rel="copyright" href="//creativecommons.org/licenses/by-sa/3.0/" />
15 <link rel="alternate" type="application/atom+xml" title="Flux Atom de Wikipédia" href="/w/index.php?title=Sp%C3%A9cial:Modifications_r%C3%A9centes&feed=atom" />
16 <link rel="canonical" href="http://fr.wikipedia.org/wiki/Wikip%C3%A9dia:Accueil_principal" />
17 <link rel="stylesheet" href="//bits.wikimedia.org/fr.wikipedia.org/load.php?debug=false&lang=fr&modules=ext.gadget.HomobookToolbarsStandard%2CuiSearch%2CWikiMiniAtlas%2CWikiOpenStreetMap%2CnewCollapsible%2Cext.uls.nojs%2Cext.visualEditor.viewPageTarget_noscript%2Cext.wikihiero%2CwikimediaBadges%2Cmediawiki.legacy.commonPrint%2Cshared%2Cmediawiki.sectionAnchor%2Cmediawiki.skinning.interface%2Cmediawiki.ui.button%2Cskins.vector.style_s&only=styles&skin=vector&*" />
18 <meta name="ResourceLoaderDynamicStyles" content="*" />
19 <link rel="stylesheet" href="//bits.wikimedia.org/fr.wikipedia.org/load.php?debug=false&lang=fr&modules=site&only=styles&skin=vector&*" />
20 <style>a:lang(ar),a:lang(kk-arab),a:lang(mzn),a:lang(ps),a:lang(ur){text-decoration:none}
21 /* cache key: frwiki:resourceloader:filter:minify-css:7:06be4fff4e895aa83d9f3697a665ff56 */</style>
22 <script src="//bits.wikimedia.org/fr.wikipedia.org/load.php?debug=false&lang=fr&modules=startup&only=scripts&skin=vector&*"></script>
23 <script>if(window.mw){
24 mw.config.set({"wgCanonicalNamespace":"Project","wgCanonicalSpecialPageName":false,"wgNamespacesNumber":4,"wgPageName":"Wikipédia:Accueil_principal","wgRevisionId":110551673,"wgRevisionId":110551673,"wgArticleId":3084756,"wgIsArticle":true,"wgIsRedirect":false,"wgAction":"view","wgUserName":null,"wgUserGroups":[""],"wgCategories":[""],"wgBreakFrames":false,"wgPageContentLanguage":"fr","wgPageContentModel":"wikitext","wgSeparatorTransformTable":["",""],"wgDefaultDateFormat":"dmy","wgMonthNames":["","janvier","février","mars","avril","mai","juin","juillet","août","septembre","octobre","novembre","décembre"],"wgMonthNamesShort":["","janv","fév","mars","avr","mai","juin","juil","août","sept","oct","nov","déc"],"wgRelevantPageName":"Wikipédia:Accueil_principal","wgIsProbablyEditable":false,"wgRestrictionEdit":["sysop"],"wgRestrictionMove":["sysop"],"wgIsMainPage":true,"wgWikiEditorEnabledModules":{"toolbar":true,"dialogs":true,"hideseig":true,"preview":false,"publish":false},"wgBetaFeaturesFeatures":{"wgMediaViewerOnClick":true,"wgMediaViewerEnabledByDefault":true,"wgVisualEditor":{"pageLanguageCode":"fr","pageLanguageDir":"ltr"},"wgPoweredByHHVM":true,"wgULSAcceptLanguageList":["francais"],"wgCategoryTreePageCategoryOptions":{"mode":"0","hideprefix":20,"showcount":true,"namespaces":["wikipedia"],"wgNoticeProject":"wikipedia","wgWikibaseItemId":"Q5296"}}}</script><script>if(window.mw){
25 mw.loader.implement("user.options",function($,jQuery){mw.user.options.set({"variant":"fr"});},{},{,});mw.loader.implement("user.tokens",function($,jQuery){mw.user.tokens.set({"editToken":"+\\","patrolToken":"+\\","watchToken":"+\\"});},{},{,});
26 /* cache key: frwiki:resourceloader:filter:minify-js:7:2e2e8d01a1b86d426e8ffe0861b948f0 */
27 </script>
28 <script>
29 <script>if(window.mw){
30 mw.loader.load(["mediawiki.page.startup","mediawiki.legacy.wikibits","mediawiki.legacy.ajax","ext.centralauth.centralautologin","mmv.head","ext.imageMetrics.head","ext.visualEditor.vi

```

Figure VI.1. Langages HTML et CSS.

La traduction de l'ordinateur



Figure VI.2. Résultat visible à l'écran.

### 2.1.1. HTML (*Hyper Text Markup Language*)

Le HTML a été présenté en version 1 en 1993. Au moment de l'apparition de MOSAIC premier navigateur WWW. Depuis lors, HTML a été largement amélioré et étendu.

Le HTML définit le contenu, c.-à-d. c'est un langage de description qui permet de décrire l'aspect d'un document, d'y inclure des informations variées (textes, images, sons, animations etc.) et d'établir des liens entre ces informations grâce aux liens hypertextes. [13][14]

### 2.1.2. CSS (*Cascading Style Sheets, aussi appelées Feuilles de style*)

Le rôle du CSS est de gérer l'apparence de la page web (agencement, positionnement, décoration, couleur, taille du texte...). Ce langage est venu compléter le HTML en 1996. [14]

## 2.2. Le langage Java Script

JavaScript est un langage de programmation de scripts orienté objet, il est à ce jour utilisé majoritairement sur Internet, conjointement aux pages Web (HTML ou XHTML). Le JavaScript s'inclut directement dans la page Web (ou dans un fichier externe) et permet de dynamiser une page HTML, en ajoutant des interactions avec l'utilisateur, des animations, de l'aide à la navigation, comme par exemple :

- Afficher/masquer du texte.
- Faire défiler des images.
- Créer un diaporama avec un aperçu "en grand" des images.
- Créer des infobulles.

JavaScript est un langage dit client-side, c'est-à-dire que les scripts sont exécutés par le navigateur chez l'internaute (le client).

Cela diffère des langages de scripts dits server-side qui est exécutés par le serveur Web. C'est le cas des langages tel que PHP. [15]

### 2.3. Les langages PHP et MYSQL

#### 2.3.1. Le langage PHP (*Hyper text Preprocessor*)

Le langage PHP a été mis au point au début d'automne 1994 par Rasmus Lerdorf, il a justement été conçu pour créer des sites "vivants" (on parle de sites dynamiques), c'est un langage de script exécuté par le serveur Web qui héberge le site (comme les scripts CGI, ASP, ...) et non par la navigateur du visiteur (comme une page Html, un script écrit en JavaScript ou une applet Java qui s'exécutent directement sur votre ordinateur...). La syntaxe du langage PHP est fortement inspirée de celles du langage C et du Perl. [16]

#### 2.3.2. MYSQL (*Structured Query Language*)

MySQL est un système de gestion de bases de données relationnelles (SGBDR), il fait partie des logiciels de gestion de base de données les plus utilisés au monde.

Le serveur de base de données MySQL est très souvent utilisé avec le langage de création de pages web dynamiques : PHP. [17]

#### 2.3.3. Côté-client et côté-serveur

##### Le Côté-client

Dans l'apprentissage des langages de publication sur le Web (Html, JavaScript ou le VBscript), des applications dites côté-client sont utilisé par le navigateur (le client) de l'utilisateur final.

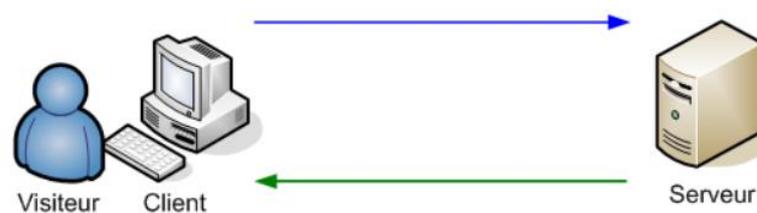


Figure VI.3. Le Côté-client.

- Le navigateur envoie l'adresse URL (Uniform Ressource Locator).
- Le serveur Web (l'ordinateur) qui héberge la page que vous demandez, va chercher le fichier demandé dans son disque dur et envoie la page Html telle qu'elle a le navigateur.
- Le navigateur interprète les différents langages se trouvant dans ce fichier (Html, JavaScript, CSS, etc.) et affiche la page. [17]

### Le côté-serveur

Pour des raisons de sécurité, Les langages utilisés dans le côté-client sont, assez limitatifs. Les applications plus complexes seront traitées dans un espace plus sécurisé qu'est le serveur qui héberge le site Web. Les traitements sont alors exécutés de côté-serveur et seulement les résultats seront envoyés au navigateur de l'utilisateur.

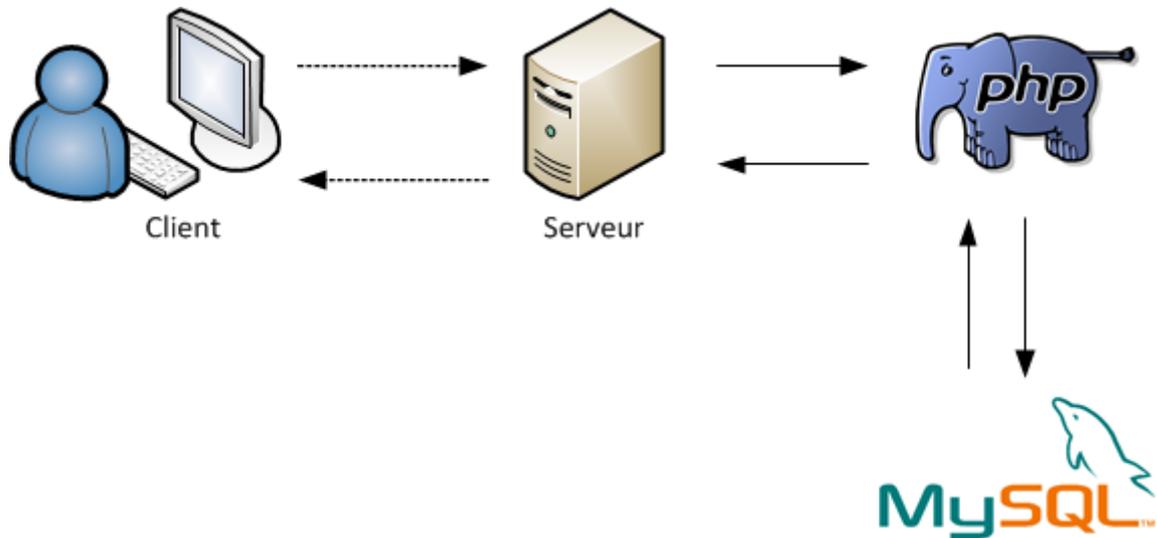


Figure VI.4. Le côté-serveur.

- Le navigateur envoie l'adresse URL (Uniform Resource Locator).
- Le serveur Web cherche dans son arborescence si le fichier existe et si celui-ci porte une extension reconnue comme une application PHP. Si c'est le cas, le serveur Web transmet ce fichier à PHP.
- PHP interprète le fichier, c'est-à-dire qu'il va analyser et exécuter le code PHP. Si ce code contient des requêtes vers une base de données MySQL, PHP envoie la requête SQL. La base de données renvoie alors les informations voulues au script qui peut les exploiter (pour les afficher par exemple).
- PHP continue d'interpréter la page, puis retourne le fichier dépourvu du code PHP (puisque'il est exécuté) au serveur Web.
- Le serveur Web renvoie finalement le fichier au navigateur de l'utilisateur. Ce fichier ne contient plus que du Html. [17]

### 3. Les outils de développement et de modélisation

#### 3.1. StarUML

StarUML est un logiciel de modélisation UML, il est compatible avec la norme UML 2.x et soutient totalement 11 types de diagrammes UML: classe, un objet, cas d'utilisation, de composants, de déploiement, de structures composites, Séquence, Communication, états-transitions, l'activité et le profil de diagramme, l'export peut se faire dans les formats JPEG, SVG et PNG.[18]

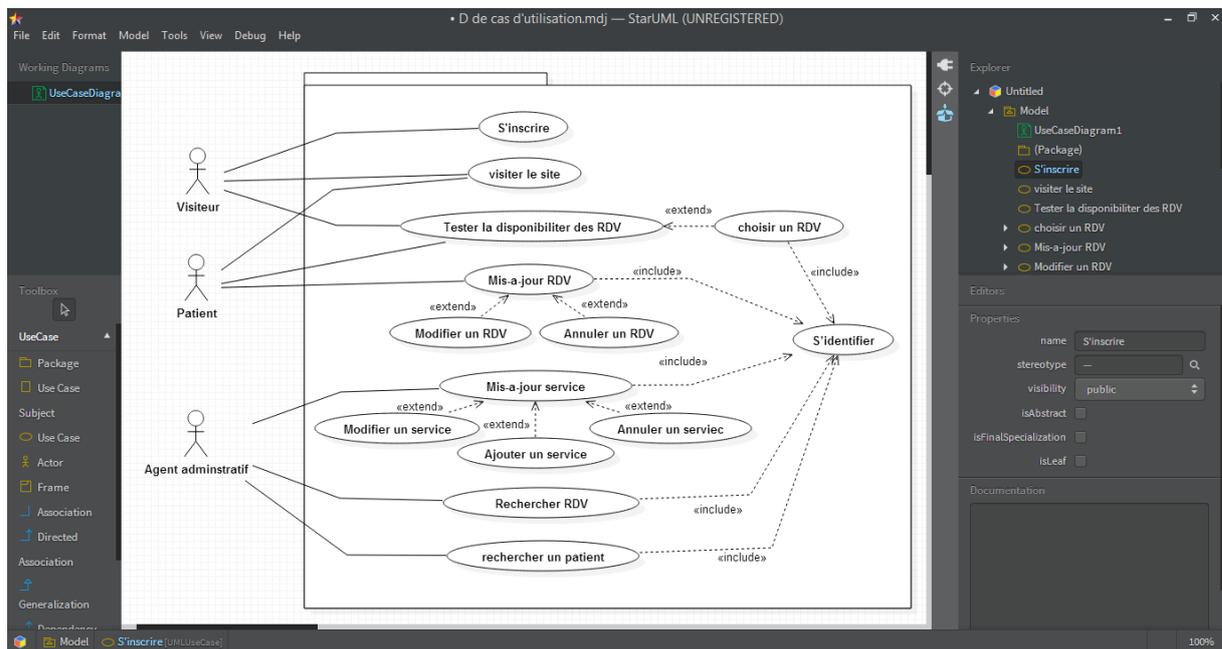


Figure VI.5. StarUML.

#### 3.2. WampServer

WampServer est une plate-forme de développement Web sous Windows pour des applications Web dynamiques à l'aide du serveur Apache2, du langage de scripts PHP et d'une base de données MySQL. Il possède également PHPMyAdmin pour gérer plus facilement vos bases de données, sans nécessiter d'une connaissance avancée des requêtes SQL, elle utilise conjointement une base MySQL et un moteur de scripts PHP, cette interface peut donc fonctionner directement sur le serveur Web et être accessible par le biais d'un simple navigateur. Elle permet la : [19]

- Gestion des bases de données (Créer, copier, supprimer et modifier des tables).

- Gestion des utilisateurs et leurs droits d'accès.
- Exécution des requêtes SQL.

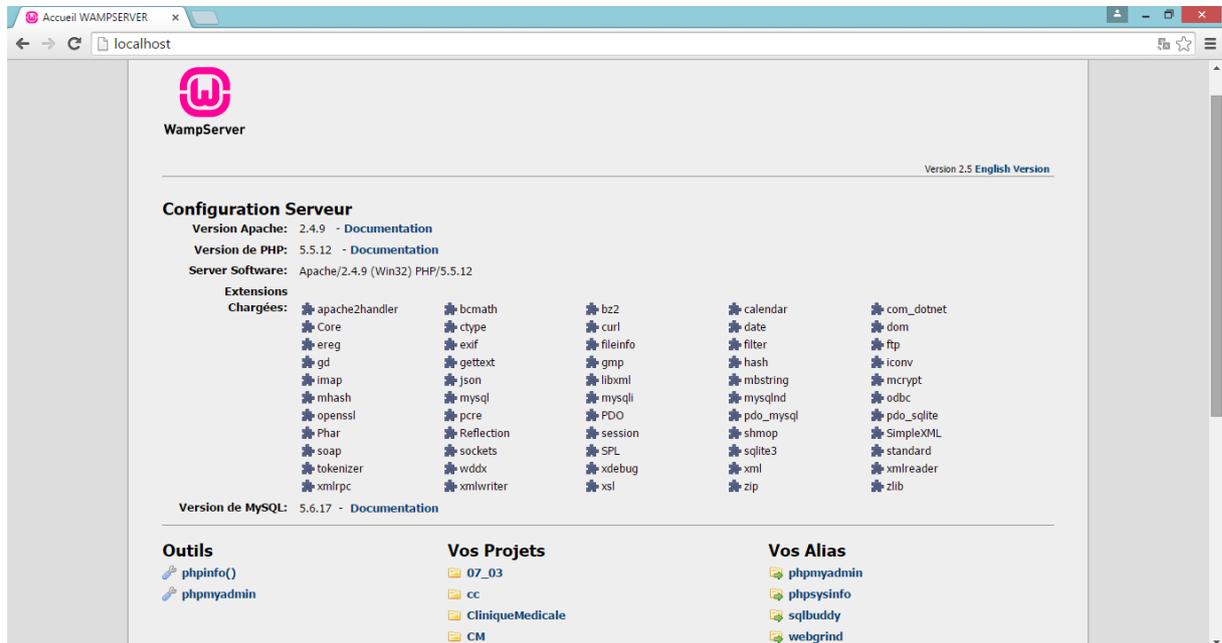


Figure VI.6. WampServer.

### 3.3. Adobe Photoshop CC

Adobe 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'images ex nihilo. Il travaille essentiellement sur images matricielles car les images sont constituées d'une grille de points appelés pixels. L'intérêt de ces images est de reproduire des graduations subtiles de couleurs.

Il possède son propre format de fichier (extension PSD). Celui-ci permet de conserver distincts les différents calques formant l'image afin de les manipuler séparément. Le programme accepte également d'importer et d'exporter des fichiers d'image dans les formats les plus courants (extensions : GIF, JPG, tif, PNG, etc.). [20]

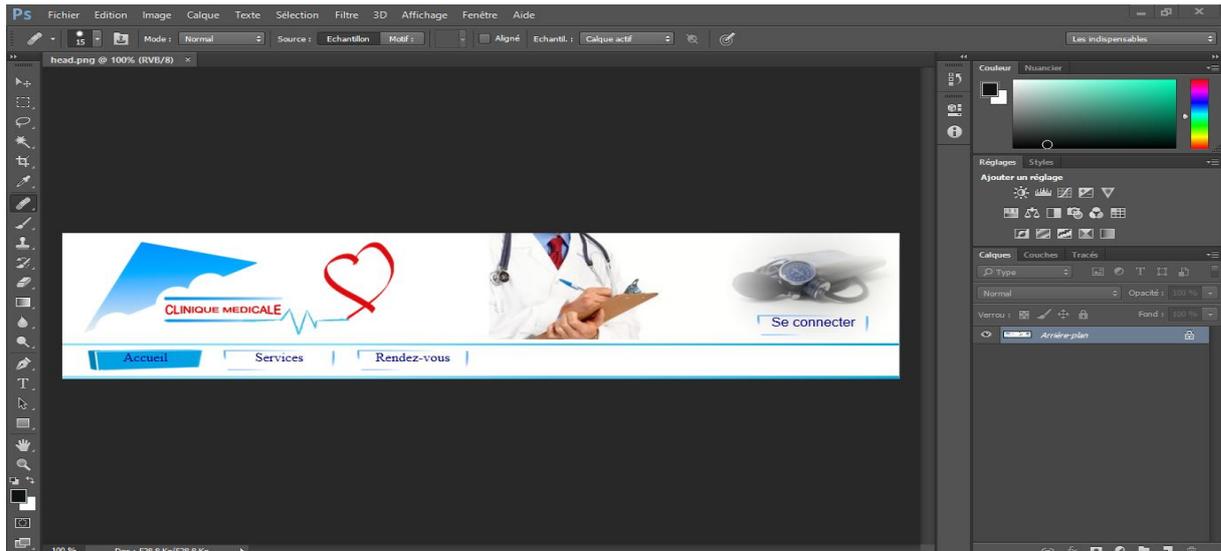


Figure VI.7. Adobe Photoshop CC.

### 3.3. Adobe Dreamweaver CC

Adobe Dreamweaver est une application de conception et de développement web qui fournit un éditeur visuel WYSIWYG (familièrement appelé le mode Création) et un éditeur de code avec fonctions standard telles que la coloration syntaxique, complétion de code, et le code se effondrer ainsi que des fonctionnalités plus sophistiquées telles que en temps réel vérification de la syntaxe et le code introspection pour générer du code conseils pour aider l'utilisateur à écrire de code. [21]

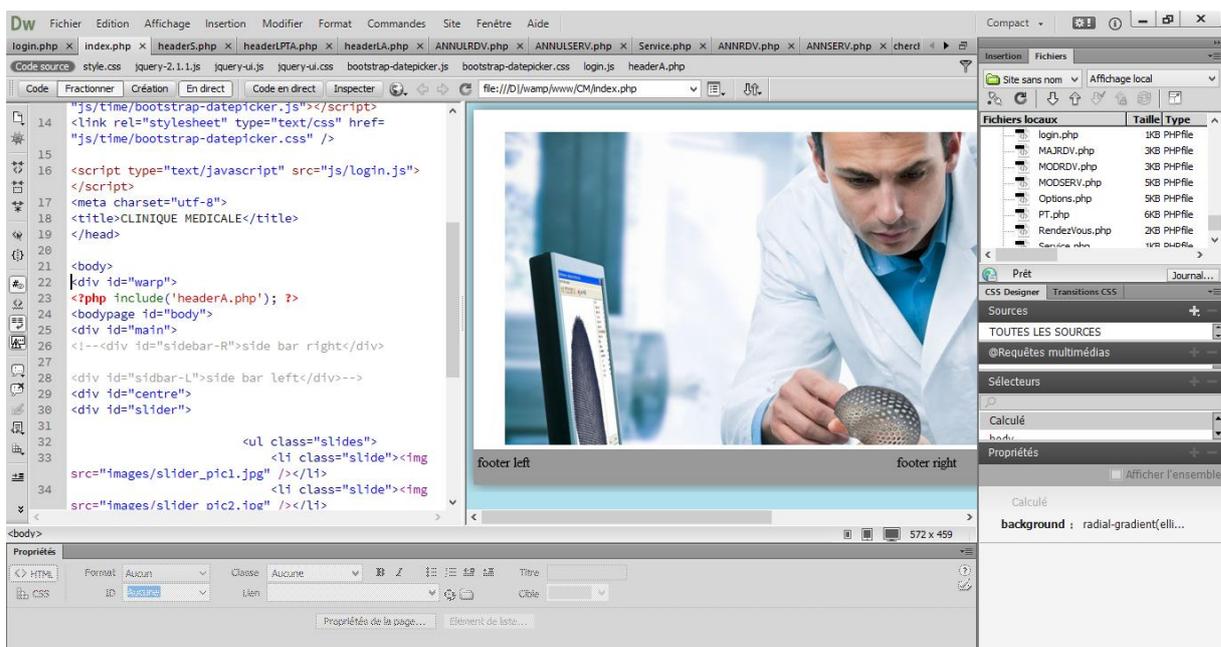


Figure VI.8. Adobe Dreamweaver CC.

#### 4. Le passage du diagramme de classe au modèle relationnel

Le diagramme de classe est un schéma qui représente les classes du système et les différentes relations entre celles-ci, et le modèle relationnel est basé sur une organisation des données sous forme de tables, la manipulation des données se fait selon le concept mathématique de relation de la théorie des ensembles (l'algèbre relationnelle), et l'algèbre relationnelle est constituée d'un ensemble d'opérations formelles sur les relations (l'union, la différence, la sélection, la projection, le produit).

Nous donnons ici- après quatre règles (de R1 à R4) pour traduire un schéma conceptuel entité association ou UML en un schéma relationnel équivalent. Il existe d'autres solutions de transformation mais ces règles sont les plus simples et les plus opérationnelles :

1) Transformation des entités /classes :

La règle R1 :

- Chaque entité devient une relation, l'identifiant de l'entité devient clé primaire pour la relation.
- Chaque classe du diagramme UML devient une relation. Il faut choisir un attribut de la classe pouvant jouer le rôle identifiant.

2) Transformation des associations :

Les règles de transformation que nous allons voir dépendent des cardinalités / multiplicités maximales des associations. Nous distinguons trois familles d'association :

**a.** Association 1 vers plusieurs (1...\*) : La règle est la suivante :

R2 : Il faut ajouter un attribut de type clé étrangère dans la relation fils de l'association. L'attribut porte le nom de la clé primaire de la relation père de l'association.

**b.** Association plusieurs vers plusieurs (\*...\*) : La règle est la suivante :

R3 : association/classe-association devient une relation. La clé primaire de cette relation est la concaténation des identifiants des entités connectées à l'association. Chaque attribut devient clé étrangère si l'entité/classe connectée dont il devient une relation en vertu de la règle R1. Les attributs d'association/classe-association doivent être ajoutés à la nouvelle relation. Ces attributs ne sont ni clé primaire, ni clé étrangère.

**c.** Association 1 vers 1 (1...1) : La règle est la suivante :

R4 : il faut ajouter un attribut de type clé étrangère dans la relation dérivée de l'entité ayant la cardinalité minimale égale à zéro. Dans le cas de diagramme UML il faut ajouter un attribut de

type clé étrangère dans la relation dérivée de la classe ayant la multiplicité minimale égale à un. L'attribut porte le nom de la clé primaire de la relation dérivée d'entité classe connectée à l'association. Si les deux cardinalités minimales égales à zéro, le choix est donné entre les deux relations dérivées de la R1. Si les deux cardinalités minimales égales à un, il est préférable de fusionner les deux entités/classe en une seule. [12]

Après application des quatre règles de passage précédentes, nous avons obtenus les tables suivantes :

Member (id, email, first\_name, last\_name, username, dateN, password, sex, #role)

Rendezvous (id, service, date, time, id\_p)

Services (id, service, duree)

Roles (id, name, slog, level)

## 5. Implémentation de la base de données

Pour implémenter notre base des données baptisée «CliniqueMedicale», nous avons utilisé PHP MyAdmin et le système de gestion de base des Donnée MySQL.

La figure ci-dessous présente notre base de données.

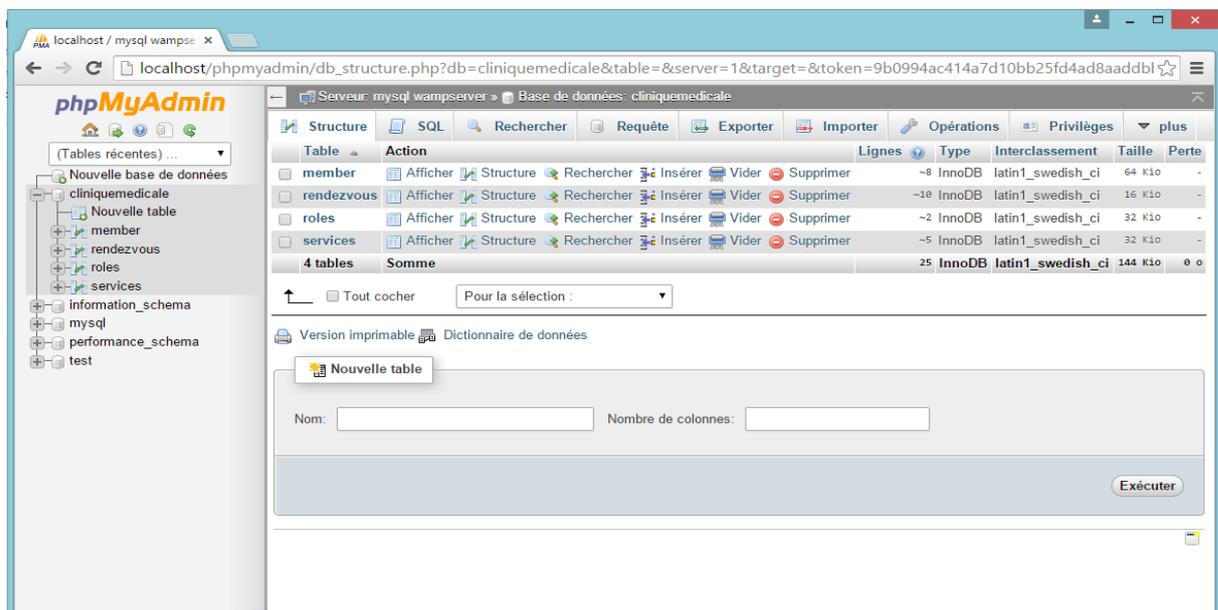


Figure VI.9. PHP MyAdmin.

6. Carte de site

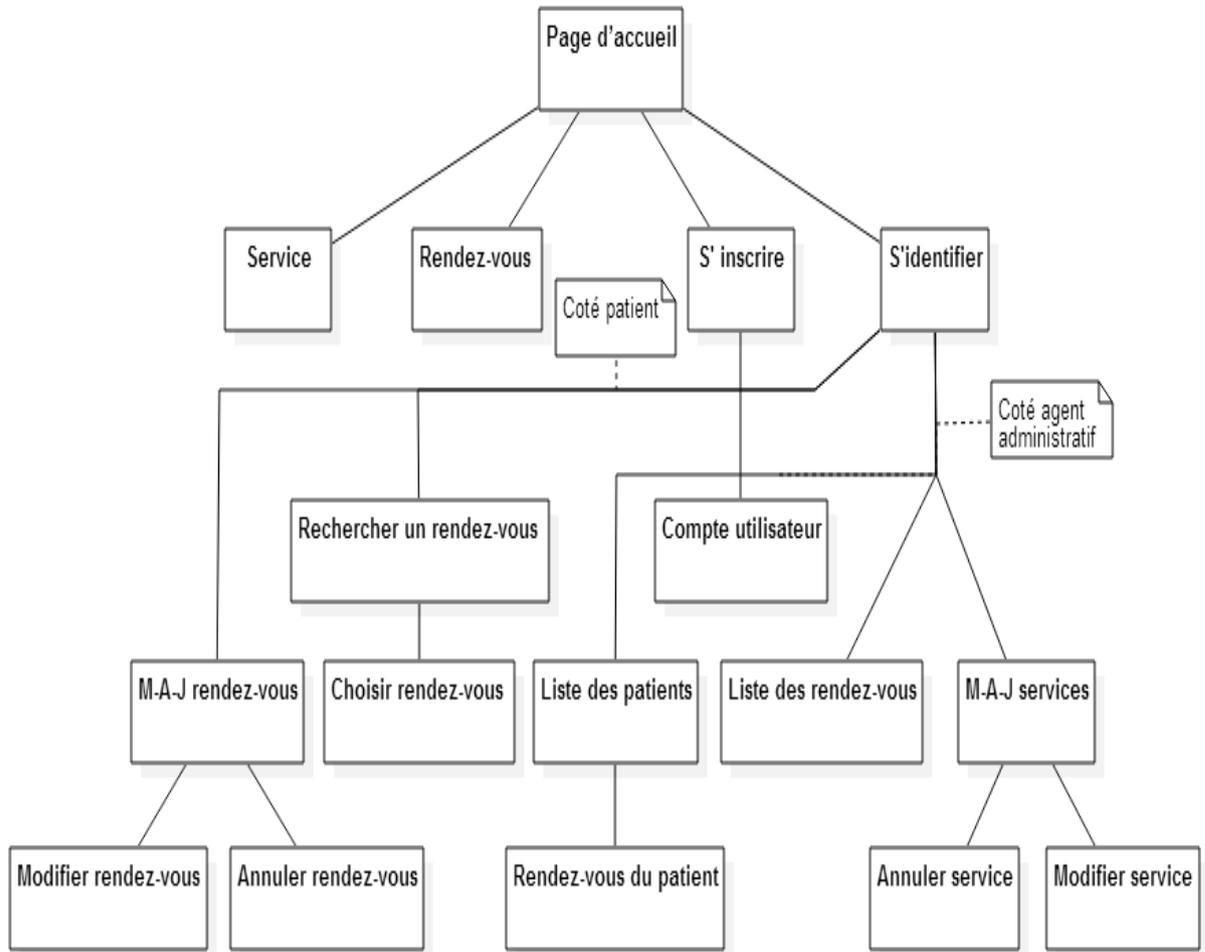


Figure VI.10. Carte de site.

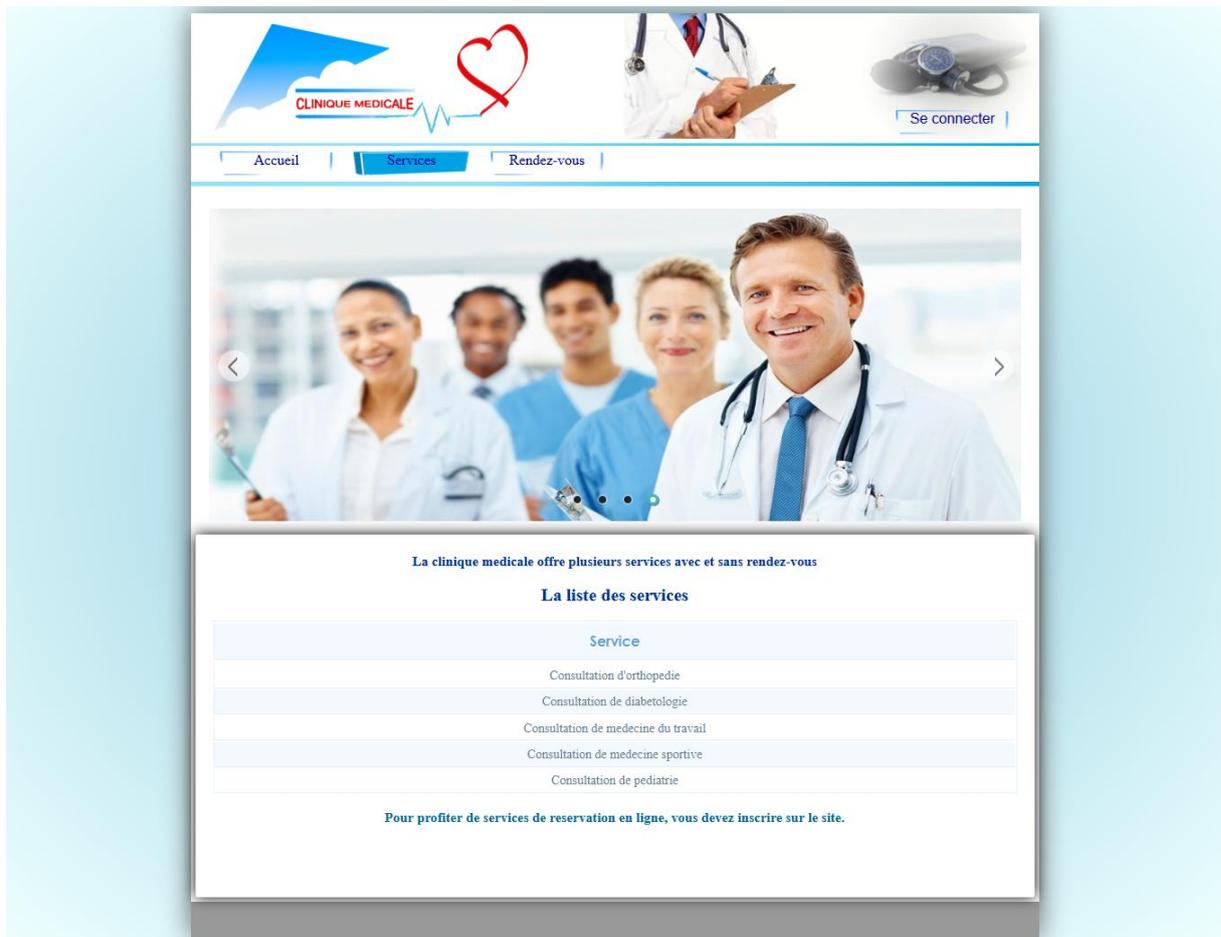
## 7. Présentation de quelques interfaces de l'application

### 7.1- Page d'accueil



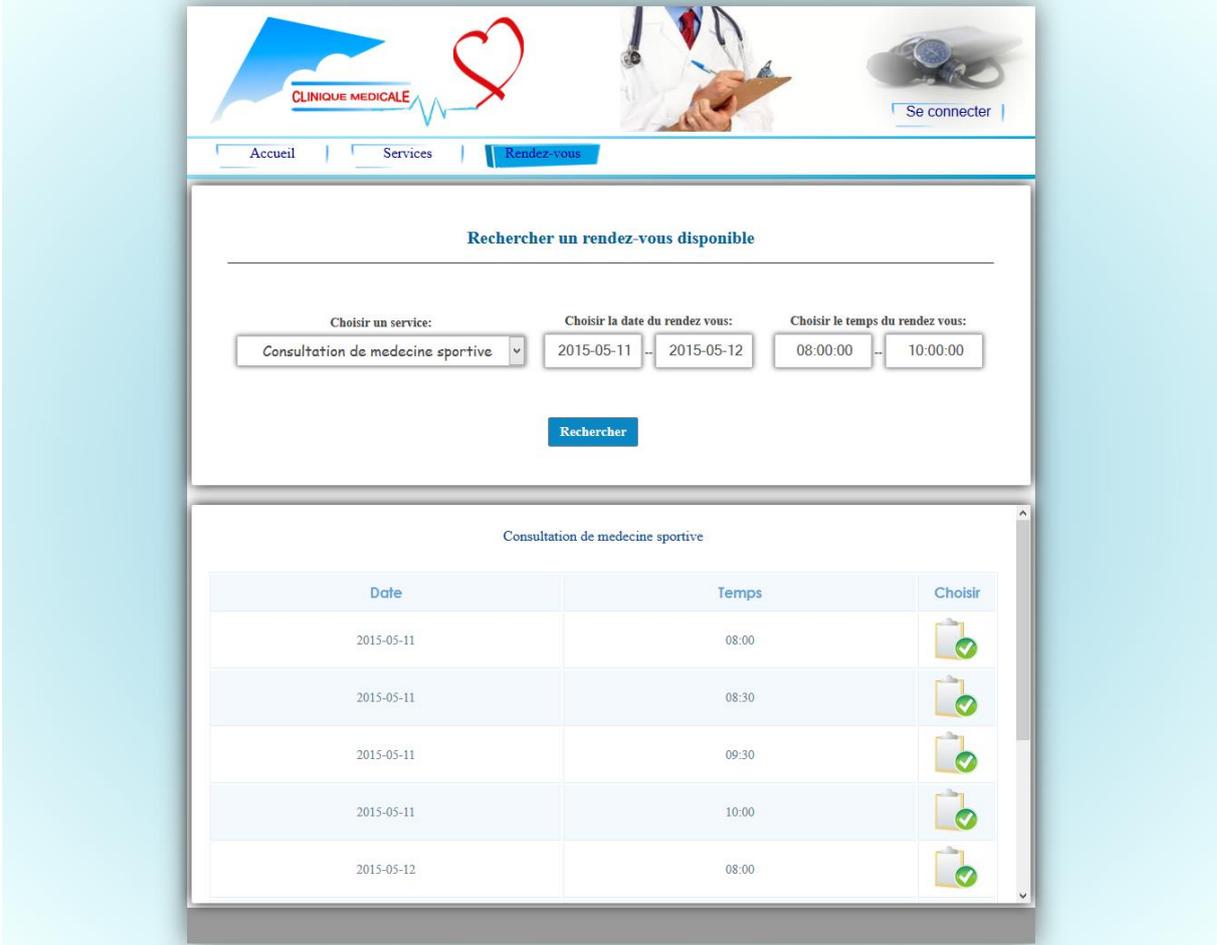
Figure VI.11. L'interface « Accueil »

## 7.2- Page de service



**Figure VI.12. L'interface « Service »**

### 7.3- L'interface du cas d'utilisation « Tester la disponibilité des RDV »

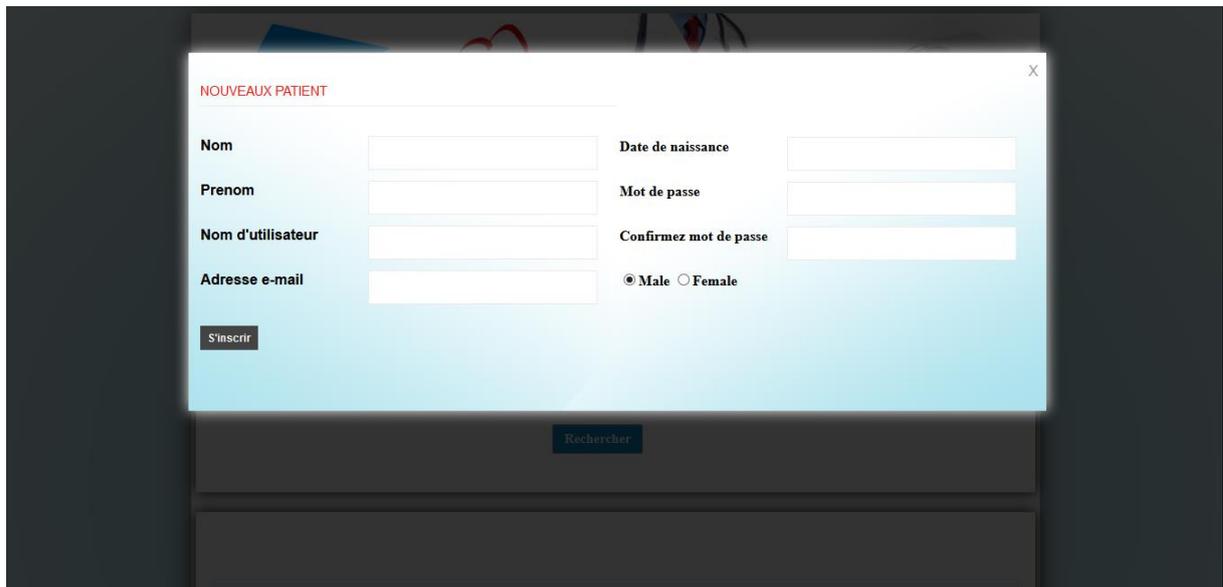


The screenshot shows a web interface for a medical clinic. At the top, there is a header with the logo 'CLINIQUE MEDICALE' and a navigation menu with 'Accueil', 'Services', and 'Rendez-vous'. Below the header, there is a search form titled 'Rechercher un rendez-vous disponible'. The form has three sections: 'Choisir un service:' with a dropdown menu showing 'Consultation de medecine sportive'; 'Choisir la date du rendez vous:' with two date pickers showing '2015-05-11' and '2015-05-12'; and 'Choisir le temps du rendez vous:' with two time pickers showing '08:00:00' and '10:00:00'. A 'Rechercher' button is located below the form. Below the search form, there is a table titled 'Consultation de medecine sportive' with three columns: 'Date', 'Temps', and 'Choisir'. The table contains five rows of data, each with a date, a time, and a green checkmark icon.

Date	Temps	Choisir
2015-05-11	08:00	
2015-05-11	08:30	
2015-05-11	09:30	
2015-05-11	10:00	
2015-05-12	08:00	

Figure VI.13. L'interface « Tester la disponibilité des RDV »

#### 7.4- L'interface du cas d'utilisation « S'inscrire »



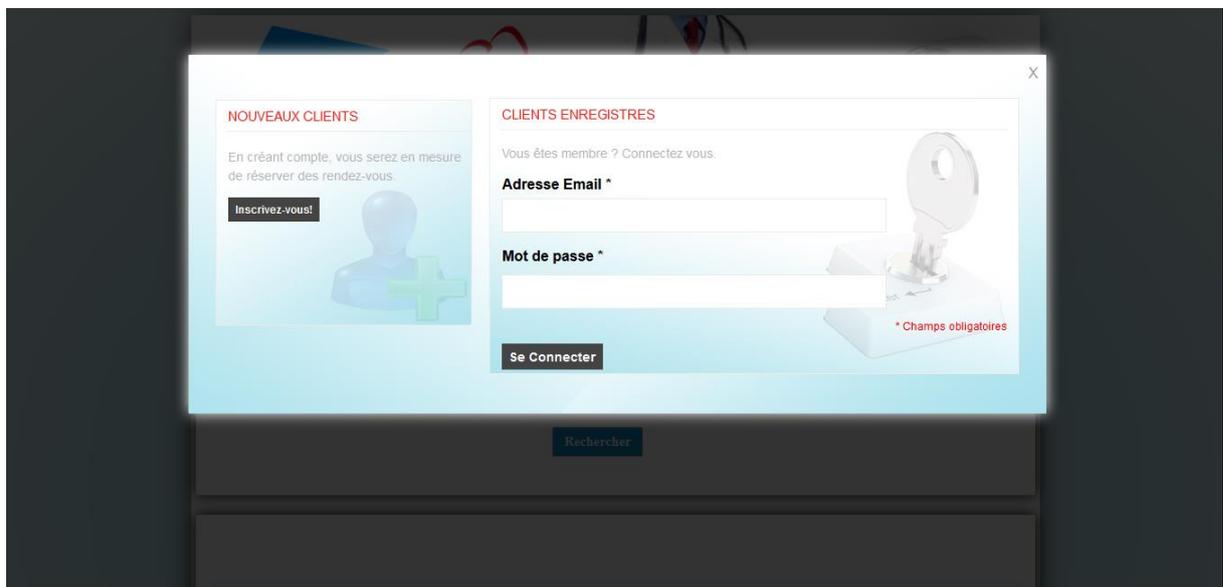
The screenshot shows a web form titled "NOUVEAUX PATIENT" with a close button (X) in the top right corner. The form contains the following fields and controls:

- Nom**: Text input field
- Date de naissance**: Text input field
- Prenom**: Text input field
- Mot de passe**: Text input field
- Nom d'utilisateur**: Text input field
- Confirmez mot de passe**: Text input field
- Adresse e-mail**: Text input field
- Sexe**: Radio buttons for "Male" (selected) and "Female"
- S'inscrire**: Submit button

Below the form, a "Rechercher" button is visible on a dark background.

Figure VI.14. L'interface « S'inscrire »

#### 7.5- L'interface du cas d'utilisation « S'identifier »



The screenshot shows a web interface with two main sections: "NOUVEAUX CLIENTS" and "CLIENTS ENREGISTRES".

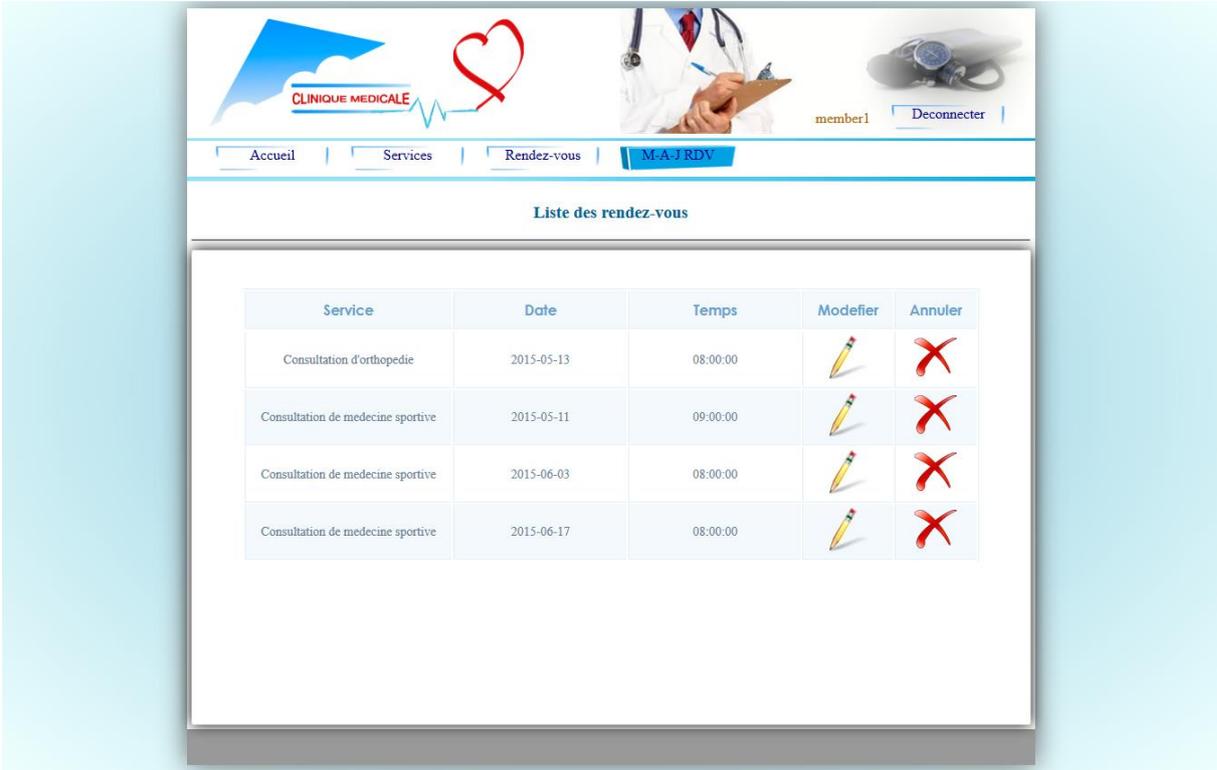
**NOUVEAUX CLIENTS:** Includes the text "En créant compte, vous serez en mesure de réserver des rendez-vous." and an "Inscrivez-vous!" button. An illustration of a person with a green cross is shown below.

**CLIENTS ENREGISTRES:** Includes the text "Vous êtes membre ? Connectez vous." and a "Se Connecter" button. It features two required fields: "Adresse Email \*" and "Mot de passe \*". A key icon is positioned to the right of these fields. A red asterisk at the bottom right indicates "\* Champs obligatoires".

Below the form, a "Rechercher" button is visible on a dark background.

Figure VI.15. L'interface « S'identifier »

## 7.6- L'interface du cas d'utilisation « M-A-J RDV » (patient)



The screenshot displays the 'M-A-J RDV' (My Appointments) interface for a patient. The header includes the clinic logo 'CLINIQUE MEDICALE', a navigation menu with 'Accueil', 'Services', 'Rendez-vous', and 'M-A-J RDV', and a user profile section with 'member1' and a 'Deconnecter' button. The main content area is titled 'Liste des rendez-vous' and contains a table with the following data:

Service	Date	Temps	Modifier	Annuler
Consultation d'orthopedie	2015-05-13	08:00:00		
Consultation de medecine sportive	2015-05-11	09:00:00		
Consultation de medecine sportive	2015-06-03	08:00:00		
Consultation de medecine sportive	2015-06-17	08:00:00		

Figure VI.16. L'interface « M-A-J RDV »

### 7.7- L'interface du cas d'utilisation « Modifier un RDV » (patient)

The screenshot displays the 'Modifier le rendez-vous' (Modify Appointment) interface. At the top, there is a header with the clinic logo 'CLINIQUE MEDICALE', a navigation menu with 'Accueil', 'Services', 'Rendez-vous', and 'M-A-J RDV', and user information 'member1' with a 'Deconnecter' button. Below the header, the title 'Modifier le rendez-vous' is centered. The main content area is divided into two sections. On the left, there is a 'Rechercher d'un RDV' (Search for an Appointment) form with three input fields: 'Choisir un service:' (dropdown menu showing 'Consultation de medecine sportive'), 'Choisir la date du rendez vous:' (date range from 2015-05-12 to 2015-05-12), and 'Choisir le temps du rendez vous:' (time range from 08:00:00 to 10:00:00). A 'Rechercher' button is located below the form. On the right, there is a table titled 'Consultation de medecine sportive' showing available appointment slots. The table has three columns: 'Date', 'Temps', and 'Action'. Each row represents a slot with a date, time, and an 'Action' icon (a folder with a green checkmark).

Service	Date	Temps
Consultation de medecine sportive	2015-05-11	09:00:00

Date	Temps	Action
2015-05-12	08:00	
2015-05-12	08:30	
2015-05-12	09:00	
2015-05-12	09:30	
2015-05-12	10:00	

Figure VI.17. L'interface « Modifier un RDV »

### 7.8- L'interfaces du cas d'utilisation « Annuler un RDV » (patient)

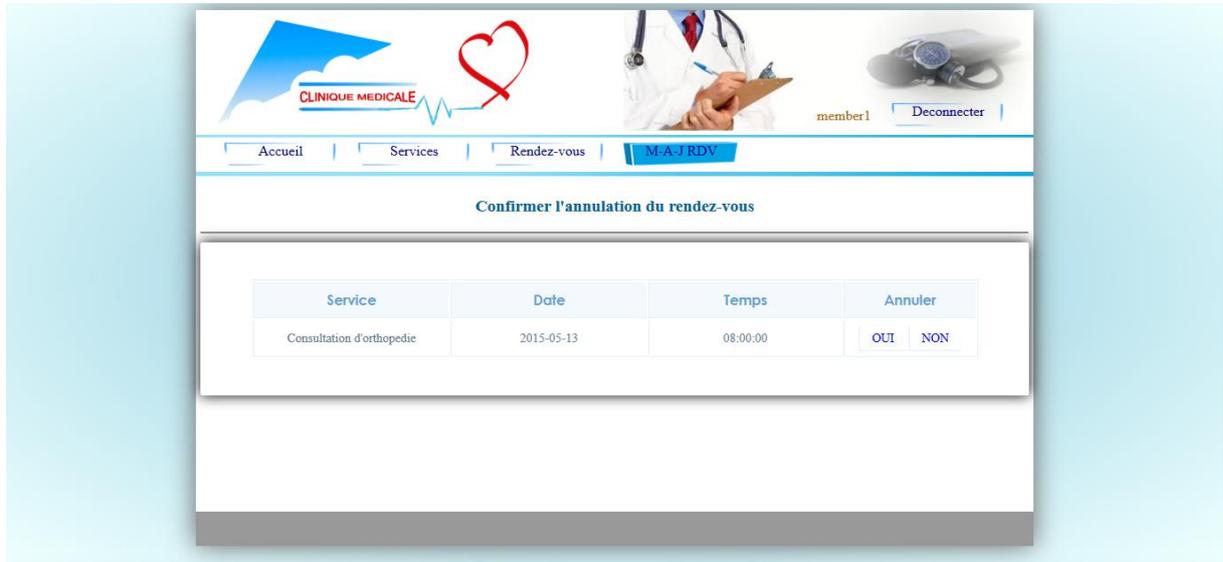


Figure VI.18. L'interface « Annuler un RDV »

### 7.9- L'interface du cas d'utilisation «Rechercher un RDV» (Agent administratif)

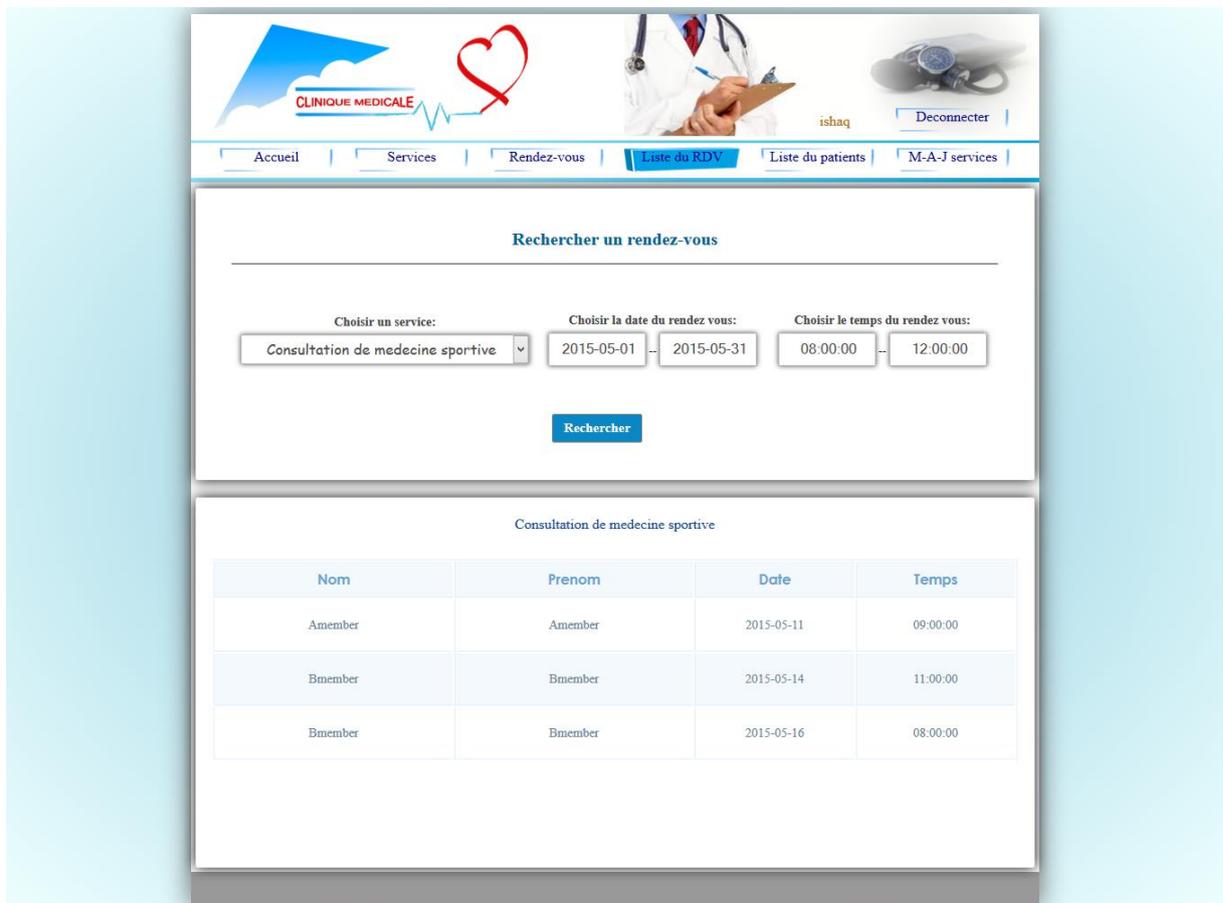


Figure VI.19. L'interface « Rechercher un RDV »

**7.10- L'interface du cas d'utilisation « Rechercher un patient » (Agent administratif)**

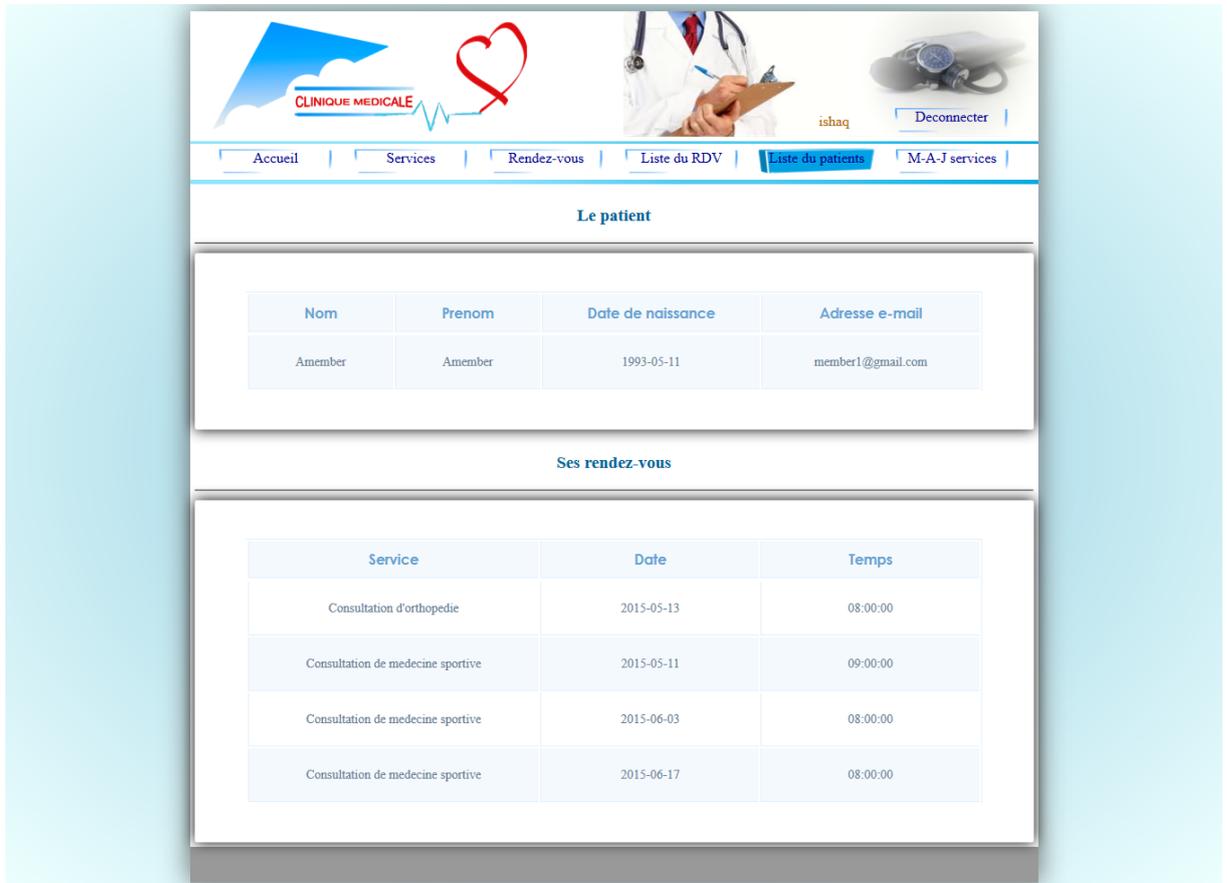
Rechercher un patient

Choisir le type de recherche:  Saisir l'information:

**Rechercher**

Nom	Prenom	Date de naissance	Adresse e-mail	RDV
Amember	Amember	1993-05-11	member1@gmail.com	
Bmember	Bmember	1993-05-11	member2@gmail.com	
Cmember	Cmember	1993-05-11	member3@gmail.com	
Dmember	Dmember	1951-02-06	membe4@gmail.com	
Fmember	Fmember	1960-02-03	member5@gmail.com	

**Figure VI.20. L'interface « Rechercher un patient »**

**7.11- L'interface du cas d'utilisation «RDV de patient » (Agent administratif)**

The screenshot shows a web interface for a medical clinic. At the top, there is a header with the logo 'CLINIQUE MEDICALE' and a heart icon. Below the header is a navigation menu with links: Accueil, Services, Rendez-vous, Liste du RDV, Liste du patients (highlighted), and M-A-J services. The user is logged in as 'ishaq' and there is a 'Deconnecter' button. The main content area is titled 'Le patient' and contains a table with patient information. Below this is a section titled 'Ses rendez-vous' containing a table of appointments.

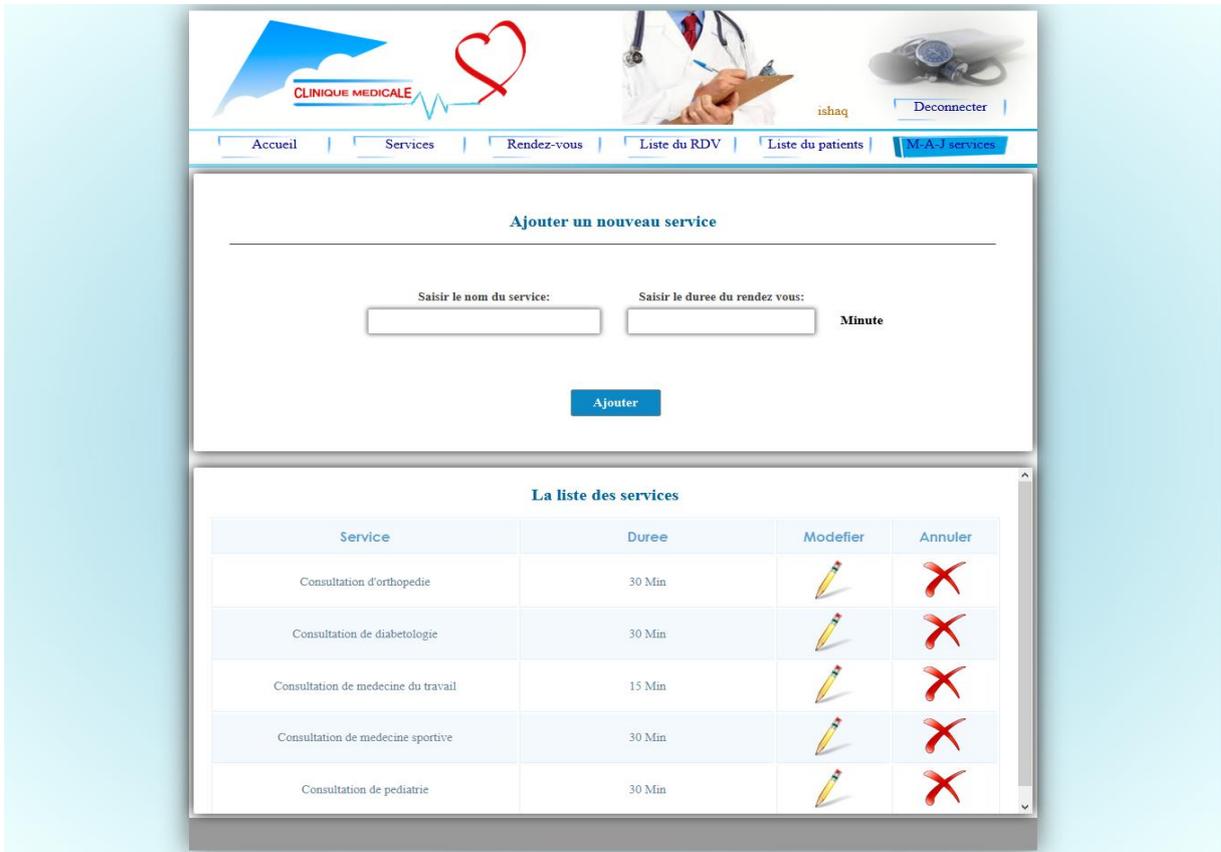
Nom	Prenom	Date de naissance	Adresse e-mail
Amember	Amember	1993-05-11	member1@gmail.com

Service	Date	Temps
Consultation d'orthopedie	2015-05-13	08:00:00
Consultation de medecine sportive	2015-05-11	09:00:00
Consultation de medecine sportive	2015-06-03	08:00:00
Consultation de medecine sportive	2015-06-17	08:00:00

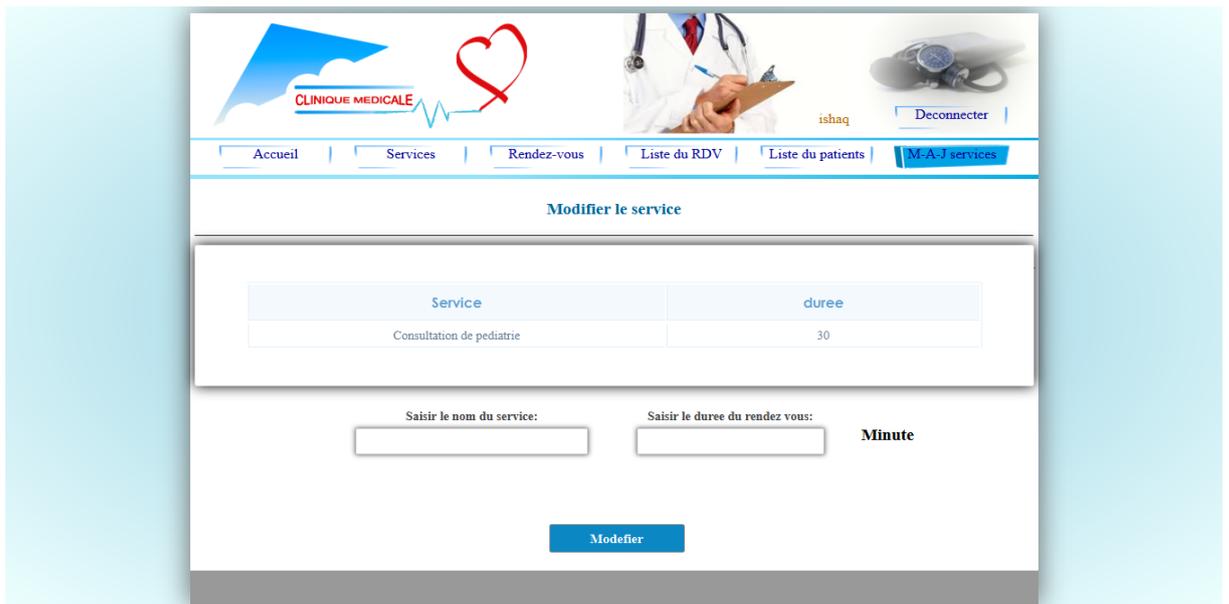
**Figure VI.21. L'interface «RDV de patient »**

**7.12- L'interface du cas d'utilisation «M-A-J service » (Agent administratif)**



**Figure VI.22. L'interface « M-A-J service »**

**7.13- L'interface du cas d'utilisation « Modifier service » (Agent administratif)**



**Figure VI.23. L'interface « Modifier service »**

### 7.14- L'interface du cas d'utilisation « Annuler service » (*Agent administratif*)

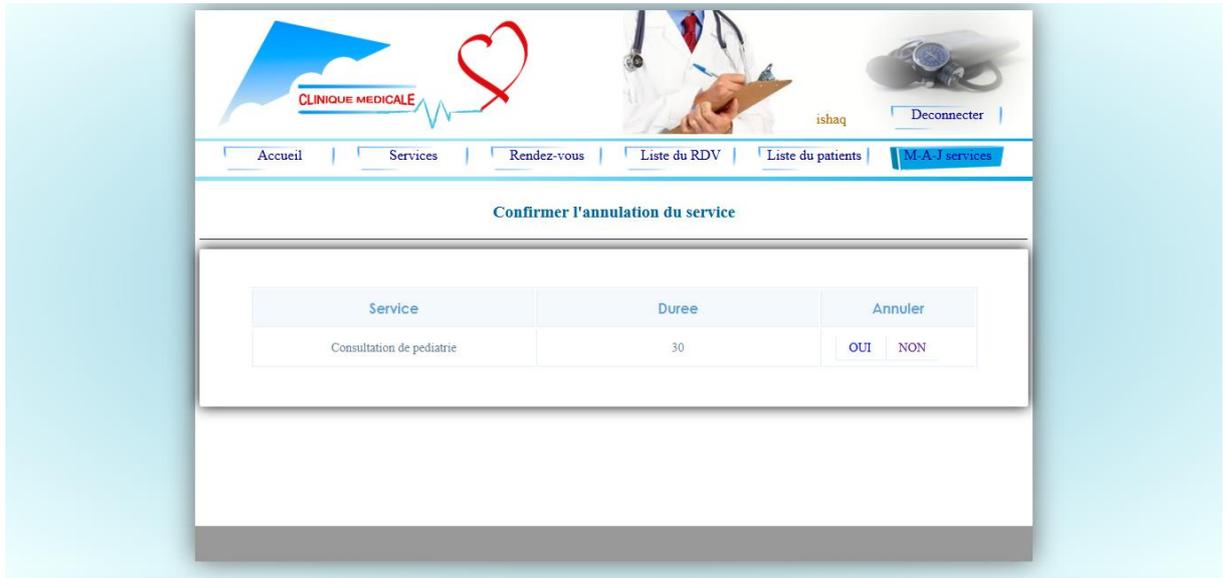


Figure VI.24. L'interface « Annuler service »

## 8. Conclusion

Ce chapitre est la phase finale de notre travail, il comprend une description générale des langages et des outils utilisés pour le développement du site. Par ailleurs, il comporte les interfaces qui déterminent de façon plus claire les activités réalisées dans ce site.

Finalement, on peut dire que notre application répond à l'objectif souhaité dès le départ.

# Conclusion générale

Dans ce mémoire, nous avons réalisé un site web dynamique pour une clinique médicale, permettant la prise et la gestion des rendez-vous pour effectuer des consultations générales ou spécialisées. Pour sa réalisation nous avons suivis une méthode conduite par les cas d'utilisations.

Dans le but de réaliser un système fiable et efficace, nous avons utilisé le langage UML avec l'approche UP simplifié pour la modélisation et les langages de programmation : JavaScript, HTML, PHP et MySQL pour l'implémentation. Sachant que le PHP est un langage important dans le monde d'implémentation des sites Web dynamiques.

Le site créé permet aux patients de prendre facilement leurs rendez-vous et de les programmer depuis leurs domiciles. Il évite ainsi le désagrément de transport et de déplacement à la clinique médicale surtout lorsque cette dernière est très loin.

Nous avons pu réussir à réaliser une application dynamique basée sur une conception adaptée à nos besoins. Ce projet nous a permis aussi d'enrichir nos connaissances quant à l'utilisation des différents langages de programmations et outils de développement.

Nous espérons, que notre travail a été bien détaillé et servira comme support pour les prochains étudiants. Néanmoins il est certain que cette application web est extensible de nature et des améliorations peuvent être apportées à celle-ci.

## Références bibliographiques

- [1] PASCAL Nicolas, cours de réseaux maîtrise d'informatique, [www.info.univ-angers.fr](http://www.info.univ-angers.fr).
- [2] Jean-Luc Montagnier, construire son réseau d'entreprise, ÉDITIONS EYROLLES 61.
- [3] CommentCaMarche ([www.commentcamarche.net](http://www.commentcamarche.net)), Web - Introduction au World Wide Web, Juin 2014.
- [4] [http://fr.wikipedia.org/wiki/Architecture\\_trois\\_tiers](http://fr.wikipedia.org/wiki/Architecture_trois_tiers).
- [5] André Aoun, Jacques Chabert, Michel Jacob : architecture client/serveur : université Paul Sabatier (Toulouse III) ; [www.htrr.ups-tlse.fr/pedagogie/cours/internet/services](http://www.htrr.ups-tlse.fr/pedagogie/cours/internet/services) ; 2001.
- [6] Abdel basset, Qu'est-ce que c'est PHP ?
- [7] CommentCaMarche ([www.commentcamarche.net](http://www.commentcamarche.net)), Environnement Client/Serveur, Juin 2014.
- [8] Pascal Roques, UML2 modéliser une application web, 3<sup>ème</sup> édition.
- [9] Joseph Gabay et David Gabay, UML2 ANALYSE CONCEPTION. Dunod, Paris, 2008.
- [10] [http://fr.wikipedia.org/wiki/UML\\_%28informatique%29](http://fr.wikipedia.org/wiki/UML_%28informatique%29)
- [11] Pascal Roques et Franck Vallée, UML 2 en action De l'analyse des besoins à la conception, 4<sup>e</sup> édition
- [12] Laurent AUDIBERT, UML 2, Édition 2007-2008.
- [13]. Grand livre HTML Daniel KOCH, Oliver KURTEN, Florian HARMS 1<sup>ère</sup> édition 2000.
- [14] Mathieu Nebra, Apprenez à créer votre site web avec HTML5 et CSS3.
- [15] Johann Pardanaud et Sébastien de la Marck, Dynamisez vos sites web avec Javascript !
- [16] Philippe Rigaux, Pratique de MySQL et PHP.
- [17] Mathieu Nebra, Concevez votre site web avec PHP et MySQL.
- [18] <http://staruml.io>
- [19] <http://www.wampserver.com>.
- [20] [http://fr.wikipedia.org/wiki/Adobe\\_Photoshop](http://fr.wikipedia.org/wiki/Adobe_Photoshop).
- [21] [http://en.wikipedia.org/wiki/Adobe\\_Dreamweaver](http://en.wikipedia.org/wiki/Adobe_Dreamweaver).