**Institut Universitaire de l’Ouest (I.U.O)**

**Travail de Fin d'Études**

Sujet: <<Analyse, conception et réalisation d'un système de prévisions des maladies cardiaques. Cas: clinique Claude Bernard.>>

Préparé par:

Azémar Nick Anderson,

Cyprien Jean Maxo,

Cadet Wadson

En accomplissement partiel des exigences académiques pour le grade de licence en Sciences Informatiques

Sous la supervision du professeur:

Jean Dumas Maurice

Port-Au-Prince

Mars 2021

**TABLE DES MATIÈRES**

[**Remerciements**](#_heading=h.gjdgxs) **5**

[**Libellé**](#_heading=h.30j0zll) **6**

[**Introduction**](#_heading=h.1fob9te) **7**

[**Partie I**](#_heading=h.88rqrz6cbyrh) **8**

[**Dossier de Pré-Étude**](#_heading=h.3znysh7) **8**

[**1- Préambule**](#_heading=h.2et92p0) **9**

[1.1 - Objectif](#_heading=h.tyjcwt) 9

[1.2 - Problématique](#_heading=h.3dy6vkm) 9

[1.3 - Contexte du projet](#_heading=h.1t3h5sf) 9

[1.4 - Limites du projet](#_heading=h.4d34og8) 10

[1.5 - Cadre Théorique](#_heading=h.2s8eyo1) 10

[1.5.1 - Cadre conceptuel](#_heading=h.17dp8vu) 10

[1.5.2 - Cadre théorique](#_heading=h.3rdcrjn) 13

[**2- Étude de l'existant**](#_heading=h.26in1rg) **13**

[2.1 - Étude du système existant sur le plan international](#_heading=h.lnxbz9) 14

[2.2 - Étude du système existant sur le plan national](#_heading=h.35nkun2) 16

[2.3 - Positionnement du système à concevoir](#_heading=h.1ksv4uv) 16

[**3- Critique de l'existant**](#_heading=h.44sinio) **17**

[3.1 - Inventaires des problèmes](#_heading=h.2jxsxqh) 17

[3.1.1 - Sur le plan international](#_heading=h.z337ya) 18

[3.1.2 - Sur le plan national](#_heading=h.3j2qqm3) 19

[3.2 - Inventaires des besoins](#_heading=h.1y810tw) 20

[3.3 - Les opportunités](#_heading=h.4i7ojhp) 20

[**4. Étude de Faisabilité**](#_heading=h.2xcytpi) **20**

[4.1 - Les Objectifs](#_heading=h.1ci93xb) 21

[4.2 - Les solutions possibles](#_heading=h.3whwml4) 21

[4.2.1 Solutions sur le Plan Technologique](#_heading=h.2bn6wsx) 22

[4.2.2 les possibilités technologiques sur le marché](#_heading=h.qsh70q) 22

[4.2.3 Solutions sur le Plan du Déploiement](#_heading=h.3as4poj) 23

[4.2.3.1 Notions d'architecture client-serveur](#_heading=h.1pxezwc) 23

[Présentation de l'architecture d'un système client/serveur](#_heading=h.49x2ik5) 23

[Avantages de l'architecture client/serveur](#_heading=h.2p2csry) 23

[Inconvénients du modèle client/serveur](#_heading=h.147n2zr) 24

[Fonctionnement d'un système client/serveur](#_heading=h.3o7alnk) 24

[Présentation de l'architecture à trois niveaux](#_heading=h.23ckvvd) 25

[4.2.3 La première solution](#_heading=h.32hioqz) 27

[4.2.3.1 Architecture logicielle](#_heading=h.1hmsyys) 27

[4.2.3.2 Architecture matérielle](#_heading=h.41mghml) 27

[4.2.4 - La deuxième solution](#_heading=h.2grqrue) 27

[4.2.4.1 Architecture logicielle](#_heading=h.vx1227) 27

[4.2.4.2 Architecture matérielle](#_heading=h.3fwokq0) 28

[4.2.5 - La troisième solution](#_heading=h.1v1yuxt) 28

[4.2.5.1Architecture logicielle](#_heading=h.4f1mdlm) 28

[4.2.5.2 Architecture matérielle](#_heading=h.2u6wntf) 30

[4.3 - Bénéfices et coûts de chaque solution](#_heading=h.19c6y18) 30

[4.3.1 - Bénéfices de chaque solution](#_heading=h.3tbugp1) 30

[4.3.1.1 benefice de la première solution](#_heading=h.28h4qwu) 30

[4.3.1.2 bénéfice de la deuxième solution](#_heading=h.nmf14n) 30

[4.3.1.3 Bénéfice de la troisième solution](#_heading=h.37m2jsg) 31

[4.3.2 - Coûts de chaque solution](#_heading=h.1mrcu09) 31

[4.3.2.1 - Coûts de la première solution](#_heading=h.111kx3o) 32

[4.3.2.2 - Coût de la deuxième solution](#_heading=h.3l18frh) 32

[$999.99](#_heading=h.206ipza) 33

[4.3.2.3 - Coût de la troisième solution](#_heading=h.4k668n3) 33

[$2000.00](#_heading=h.2zbgiuw) 34

[4.4 - Choix de la solution](#_heading=h.1egqt2p) 35

[4.5 - Risque et enjeux](#_heading=h.3ygebqi) 35

[**Partie II**](#_heading=h.sqyw64) **36**

[2.1 - Description général du produit](#_heading=h.1rvwp1q) 37

[2.1.1 - Les parties du système](#_heading=h.4bvk7pj) 37

[2.1.2 - Des utilisateurs](#_heading=h.2r0uhxc) 37

[2.2 - Description des fonctionnalités du système](#_heading=h.1664s55) 40

[2.1.4 - Contraintes d'ordre général](#_heading=h.3q5sasy) 41

[2.1.5 - Répartitions des exigences](#_heading=h.25b2l0r) 42

[2.2 - Description détaillée](#_heading=h.kgcv8k) 42

[2.2.1 - Les cas d'utilisation](#_heading=h.34g0dwd) 42

[2.2.2 - Scénarios des cas d'utilisation](#_heading=h.1jlao46) 43

[2.3 - Exigences logiques de base de données](#_heading=h.43ky6rz) 56

[2.3.1 - Capacité d'accès](#_heading=h.2iq8gzs) 56

[2.3.2 - Persistance](#_heading=h.xvir7l) 56

[2.3.3 - Fréquence d'utilisation](#_heading=h.3hv69ve) 56

[2.4 - Contraintes de conception](#_heading=h.1x0gk37) 56

[2.5 - Exigences non-fonctionnelles](#_heading=h.4h042r0) 57

[**Partie III**](#_heading=h.c4yybr5vcmoh) **59**

[**Dossier d'Analyse**](#_heading=h.1opuj5n) **59**

[3.1 - Méthodologie Utilisée](#_heading=h.48pi1tg) 60

[3.2 - Présentation d'UML](#_heading=h.2nusc19) 65

[3.3 - Présentation du système du point de vue statique et dynamique](#_heading=h.1302m92) 67

[3.3.1- Diagramme de cas d'utilisation (vue statique)](#_heading=h.3mzq4wv) 68

[3.3.2 - Diagramme d'activités (vue dynamique)](#_heading=h.319y80a) 72

[3.3.2 - Diagramme de séquences (vue dynamique)](#_heading=h.1gf8i83) 94

[3.3.4 - Diagramme de classes](#_heading=h.40ew0vw) 111

[**Partie IV**](#_heading=h.ybw2a44bqi4g) **116**

[**Dossier de Conception**](#_heading=h.2fk6b3p) **116**

[4.1 - Interface Homme-Machine](#_heading=h.upglbi) 117

[4.2 - Schémas de base de données](#_heading=h.3ep43zb) 117

[4.2.1- Modèle conceptuel de données](#_heading=h.1tuee74) 117

[4.3 - Dictionnaire de données](#_heading=h.4du1wux) 124

[4.4 - Architecture du système](#_heading=h.184mhaj) 127

[4.4.1 - Architecture matérielle](#_heading=h.3s49zyc) 127

[4.4.2 - Architecture physique](#_heading=h.279ka65) 127

[4.4.3 - Le diagramme de déploiement](#_heading=h.meukdy) 128

[4.4.4 - Architecture logicielle](#_heading=h.36ei31r) 129

[**Partie V**](#_heading=h.dgwnm7jkl181) **130**

[**Développement**](#_heading=h.1ljsd9k) **130**

[5.1 - Méthode de développement](#_heading=h.45jfvxd) 131

[5.2-Planification](#_heading=h.2koq656) 133

[5.3 - Outils de développement](#_heading=h.zu0gcz) 134

[**Partie VI**](#_heading=h.rawy7ihn12no) **136**

[**Conclusion et Perspective**](#_heading=h.3jtnz0s) **136**

[Annexe](#_heading=h.1yyy98l) 137

[A. Acronymes et Abréviations](#_heading=h.4iylrwe) 137

[B. Glossaire](#_heading=h.2y3w247) 138

[D. Bibliographie](#_heading=h.1d96cc0) 140

[E. Webographie](#_heading=h.3x8tuzt) 140

# Remerciements

Ce travail est le fruit d'un travail acharné et de nombreux sacrifices ; nos remerciements vont d'abord au Fondateur de l'univers qui nous a bénis avec sagesse et nous a gardés en sécurité pour mener à bien ces années d'étude.

Nous tenons également à exprimer notre gratitude à nos familles et, plus important encore, à nos frères et sœurs qui nous ont toujours aidés et inspirés à poursuivre nos études.

Nous remercions également tous les experts et amis consultés dans le cadre de cette mission, qui ont aimablement répondu à nos nombreuses questions sur le sujet dans un effort pour améliorer la production de ce travail.

Un merci spécial au Doyen Mackenson Doucet qui nous a présenté et mis sur la voie de ce programme d'études offert par l'Université.

Nos remerciements vont également à notre mentor Jean Dumas Maurice qui, compte tenu de ses moments difficiles, nous a soutenus par ses remarques importantes.

Nous remercions vivement le corps professoral des sciences informatiques de l’IUO d’avoir mis leurs compétences à notre formation pendant ces quatre difficiles années, nous leur devons une fière chandelle.

Enfin, nous souhaitons également remercier l'Administration de l'Université de l'Ouest (I.U.O.), qui nous a fourni du matériel de qualité à travers leur programme, afin que nous puissions simplifier notre apprentissage.

# Libellé

**<<**Analyse, conception et réalisation d'un système de prévisions des maladies cardiaques. Cas : Clinique Claude Bernard.>>

# Introduction

Les progrès scientifiques ont connu une croissance exponentielle fulgurante entre la fin du XXème et le début du XXIème siècle. La médecine moderne telle que nous la connaissons aujourd’hui fait partie des domaines scientifiques qui ont bénéficié, d’une façon incontournable et irréversible, des apports des sciences de l’informatique et des outils technologiques modernes.

Justement parlant, grâce aux progrès de l’informatique dans le domaine de l’intelligence artificielle et de l’apprentissage machine, il devient possible pour la médecine contemporaine de faire des prédictions sur certaines maladies, autrefois connues pour avoir été des maladies difficilement diagnosticables voire prédictibles.

Voulant être en phase au progrès de la médecine, avec les nouvelles technologies l’IA pour prévenir les risques de maladies cardiaques chez ses patients, la clinique Claude Bernard a confié à un groupe d’étudiants finissants de L’IUO le développement d’une application permettant à la clinique d’offrir ce service.

Dans cette optique, l’équipe de développement a baptisé l’application développée pour son client du nom de “CLCB”, dont ce document permet de prendre de plus amples connaissances, et constituant ainsi son guide.

# Partie I

# Dossier de Pré-Étude

# 1- Préambule

## 1.1 - Objectif

L’objectif visé par ce système est d’aider les cardiologues de la clinique Claude Bernard de mieux fournir des prévisions sur les risques de développement de maladies cardiaques chez leurs patients par le biais de technologies innovantes basées sur l’Intelligence Artificielle. Et pour avoir plus d'expériences, afin de mieux adapter le système à d'autres cliniques.

## 1.2 - Problématique

Dans divers domaines de spécialisation, la médecine moderne a fait des progrès fascinants. Cependant, dans le domaine de la cardiologie, en raison du manque d'outils de prévision des médecins praticiens, le pourcentage de patients qui succombent déjà chaque jour à une maladie cardiaque continue d'augmenter. N'y a-t-il pas un ou plusieurs moyens ouverts pour aider ces médecins en Haïti à évaluer en temps réel et de manière spécifique les risques de maladies cardiaques avant qu'elles n'évoluent chez leurs patients, compte tenu des gigantesques développements technologiques et l’évolution de la science médicale en ce XXIe siècle ?

## 1.3 - Contexte du projet

Ce projet fait partie intégrante du contexte d'innovation médicale pour Haïti. Le fonctionnement général et l'organisation de la clinique seraient fortement influencés. À cet égard, nous appliquerons les techniques, les stratégies apprises lors de notre formation à l'université, ainsi que tous les concepts et pratiques qui nous permettent de réaliser ce projet tout en reconnaissant nos limites. Et également, ce projet rentre dans un contexte de travail de fin d'études (TFE) à l'Institut Universitaire de l’Ouest (I.U.O).

## 1.4 - Limites du projet

Dans le cadre de ce projet, il s'agit de créer une application Web dans le cadre de nos recherches faisant partie d'un TFE, dont la réalisation a pris jusqu'à six mois. Dans ce contexte, certaines des fonctions de ce système ne seront que expérimentales ou symboliques à cause du fait qu’il nous faudra des appareils biométriques pour les implémenter lors des différentes consultations des patients de la clinique.

## 1.5 - Cadre Théorique

Les approches proposées sur lesquelles se fonde notre réflexion et la relation qui lie les concepts sur lesquels la recherche est centrée doivent être définies afin d'atteindre, caractériser et établir les limites de ce projet.

### 1.5.1 - Cadre conceptuel

L'intelligence artificielle explique la capacité de la machine à comprendre comment une personne apprend, comme la compréhension des symboles et des modèles dans des situations complexes. L'IA peut faciliter la collecte, l'examen et la production d'informations médicales par les patients, afin de pouvoir fournir des diagnostics précoces concernant des risques potentiels de maladies.

Parmi les différents types d’IA on peut regrouper 4 types principaux[[1]](#footnote-0), notamment :

1. Machines réactives :

Ce type est très ancien, ces machines imitent l’esprit humain d’une manière très limitée. Elles ne peuvent pas utiliser les expériences acquises précédemment pour influencer leurs actions. Elles peuvent être utilisées uniquement pour répondre automatiquement à une combinaison d'entrées programmée à l’avance. Elles n’ont pas la capacité de s’améliorer au fil du temps. Un exemple de machines réactives est le **Deep Blue d'IBM**, une machine qui a battu le grand maître d'échecs **Garry Kasparov** en 1997[[2]](#footnote-1).

1. Intelligence Artificielle à Mémoire limitée :

Ce type d’IA comprend les mêmes caractéristiques du premier, mais la différence c’est qu’elles sont également capables de s’améliorer au fil du temps pour prendre des décisions. Presque toutes les applications existantes que nous connaissons relèvent de cette catégorie d'IA basée sur des algorithmes d’apprentissage[[3]](#footnote-2). Les IA qui se trouvent dans cette catégorie sont entraînés par de grands volumes de données d'entraînement qu'ils stockent dans leur mémoire, pour former un modèle de référence, afin de résoudre les problèmes futurs. Par exemple on peut citer les voiture autonome[[4]](#footnote-3).

1. Intelligence Artificielle avec théorie de l’esprit :

Les deux types d'IA précédentes sont très populaires, ce type d'IA existe, pour l'instant, soit en tant que concept, soit en tant que travail en cours. En théorie cette IA sera capable de mieux comprendre les entités avec lesquelles elle interagit en discernant leurs besoins, leurs émotions, leurs croyances et leurs processus de pensée[[5]](#footnote-4).

1. Intelligence Artificielle consciente de soi :

C’est la dernière étape du développement de l'IA qui existe actuellement de manière incertaine. L'idée est que cette IA consciente de soi est une IA qui a évolué pour être si proche du cerveau humain qu'elle a développé, en quelque sorte, une conscience de soi. La création de ce type d'Ai, est très loin de se matérialiser, est et sera toujours l'objectif ultime de toute recherche sur l'IA[[6]](#footnote-5).

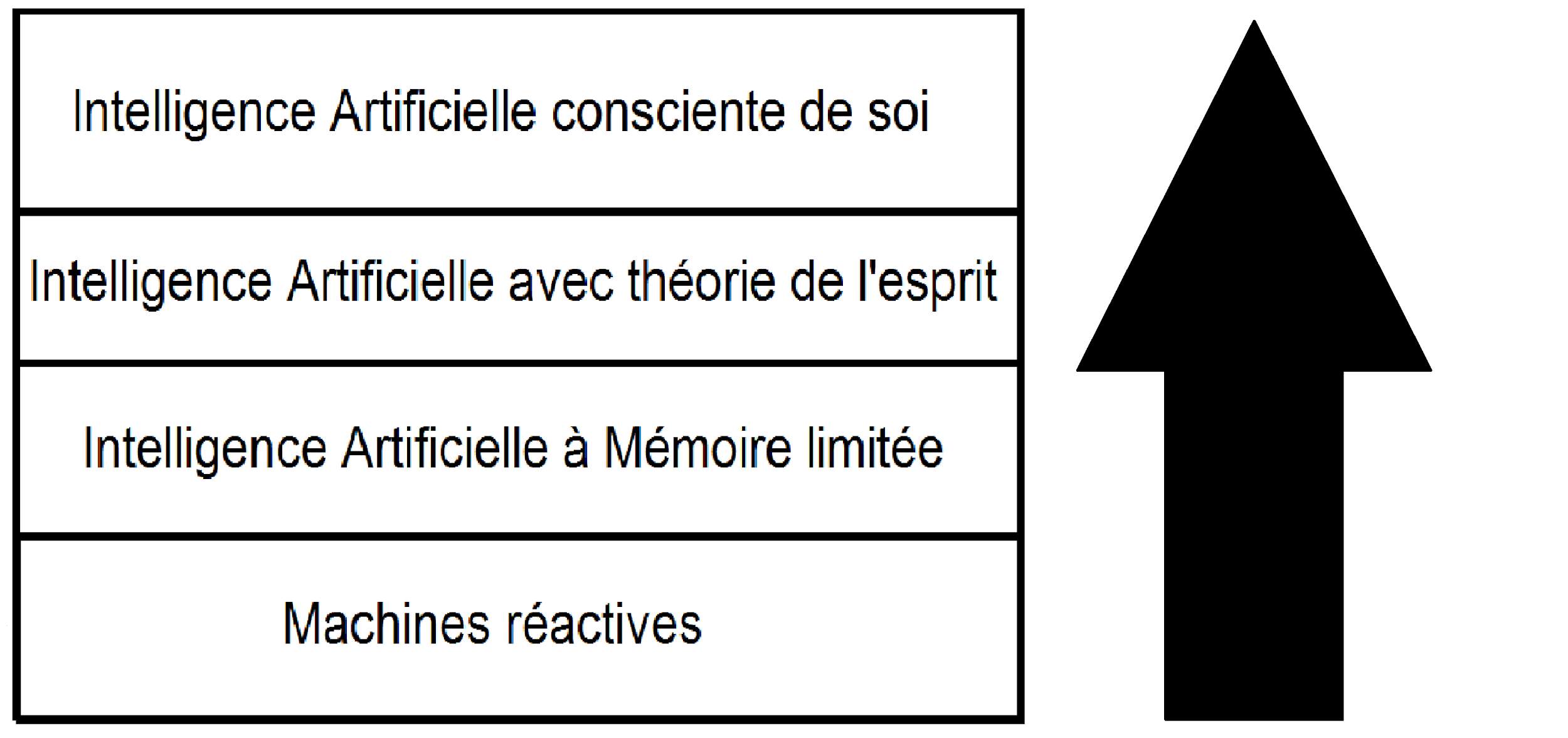


Figure 1 : Illustre les types d'intelligences artificielles avec une flèche indiquant leur niveaux de puissance.

Conformément à l'Organisation mondiale de la santé[[7]](#footnote-6), les maladies cardiaques ou cardiovasculaires englobent une série de troubles qui affectent le cœur et les vaisseaux sanguins, les plus courants d'entre eux étant

1. Les cardiopathies coronariennes :

Elles influencent les vaisseaux sanguins qui renforcent le muscle cardiaque.

1. Les maladies cérébro-vasculaires :

Elles influencent les vaisseaux sanguins qui nourrissent le cerveau.

1. Les artériopathies périphériques :

Elles ont un impact sur les vaisseaux sanguins qui alimentent les bras et les jambes.

1. Les cardiopathies rhumatismales :

Elles endommagent le muscle et les valves cardiaques, émergeant d'une raideur articulaire intense causée par des organismes microscopiques streptococciques.

1. Les malformations cardiaques congénitales :

Ce sont des anomalies du système cardiaque qui sont présentes depuis la naissance.

1. Les thromboses veineuses profondes et les embolies pulmonaires :

Elles entraînent le blocage des veines des jambes par un caillot sanguin, qui est susceptible de se décharger et de se déplacer dans le cœur ou les poumons.

### 1.5.2 - Cadre théorique

L'intelligence artificielle dans les soins de santé utilise des algorithmes avancés conçus pour exécuter des tâches de manière automatisée. Lorsque les chercheurs, les médecins et les scientifiques injectent des données dans des ordinateurs, une série d'algorithmes peut analyser, identifier et même fournir des réponses à des problèmes médicaux complexes.

En tant que chercheurs nous avions conçu un système, qui utilise en forme d’outil, une IA de type secondaire, c’est-à-dire le type d’Intelligence Artificielle à Mémoire limitée. Car elle pourra au fil du temps apprendre à fournir des prévisions qui seront ensuite mises à disposition des spécialistes pour leurs diagnostiques approfondies.

Nous précisons que cette IA ne remplacera pas les spécialistes, elle aura seulement pour but de faire des prévisions médicales basées sur un ensemble de paramètres liés aux patients.

# 2- Étude de l'existant

Chaque travail d'étude comprend l'analyse du scénario existant, ce qui permet non seulement de mieux comprendre le sujet étudié, mais également de fournir des suggestions réalistes le concernant. Pour mieux comprendre l’existant, nous passerons en revue certains logiciels disponibles à l’international et au niveau national pour reconnaître leurs forces et leurs faiblesses.

## 2.1 - Étude du système existant sur le plan international

Il existe de nombreuses applications santé sur le marché international, parmi lesquelles on peut citer : Sense.ly, SkinVision, ADA, WebMD… dans le cadre de notre étude celles qui retiennent notre attention sont ADA[[8]](#footnote-7), WebMD[[9]](#footnote-8).

Après son introduction mondiale en 2016, ADA est devenu un logiciel médical utilisé dans 140 pays. Les utilisateurs peuvent saisir des symptômes à l'aide de la conception de message instantané de l'application. L’intelligence artificielle de ADA peut examiner d’une façon claire les symptômes des utilisateurs.

ADA possède une base de données dans sa bibliothèque qui est automatisée par classification, regroupement et recherche de dossiers médicaux pour aider les patients à mieux comprendre et à contrôler leurs symptômes.

ADA collecte des informations de soins adaptées aux patients pour en saisir les effets de manière globale. Avec son excellente ergonomie, les patients peuvent de manière intuitive utiliser les différentes options qu’offre cette application. Cependant ADA ne fournit pas de prescription de médicaments, c'est-à-dire que si leurs conditions s'aggravent, les consommateurs sont invités à localiser un spécialiste.

WebMD est une organisation américaine offrant des ressources avec des informations sur la santé. WebMD est surtout connu pour le site Web webmd.com, qui contient des ressources sur la santé et le bien-être telles qu’un vérificateur de symptômes, des informations sur les médicaments, des blogs de médecins sur certains sujets et un endroit où vous pouvez stocker des informations confidentielles sur la santé.

WebMD utilise l'apprentissage automatisé pour des résultats fiables. Les utilisateurs peuvent également utiliser leur voix comme commande plutôt que d'appuyer sur un bouton pour obtenir des informations. L'application est utilisée par les bénéficiaires pour trouver des informations sur les effets des médicaments, les maladies, les symptômes, etc.

WebMD dispose d'un index essentiel, et les utilisateurs peuvent facilement utiliser la fonction de recherche ou modifier leur recherche via les catégories spécifiées. La plateforme utilise les fonctionnalités suivantes :

1. Vérificateur de symptômes : Les patient peuvent trouver la partie du corps qui leur trouble, trouvez vos signes et renseignez-vous sur les maladies ou complications possibles.
2. Répertoire des médecins : recherchez un médecin, un hôpital ou une pharmacie local en entrant votre code postal.
3. Conditions : Accès détaillés les causes, les remèdes et les effets des maladies évaluées par des professionnels qui sont importantes pour vous.
4. Médecine : Vérifiez les connaissances pertinentes sur les médicaments. Son instrument d'identification de pilule intégré permet des résultats précis pour s'adapter au type de médicament sur ordonnance.
5. Info : utilise des publications, des photographies et des diaporamas sur les dernières informations sur la santé.
6. Rappels de médicaments : rappels de doses quotidiennes de photos de médicaments et informations sur la stimulation.

## 2.2 - Étude du système existant sur le plan national

LASWENYAY est un outil moderne de bien-être numérique et d'assurance santé numérique, selon l'initiateur Jeeph Sergilles. LASWENYAY permet à des milliers de personnes, au moyen d'appareils intelligents, d'accéder à distance aux soins, comme une traduction créole du mot « santé ». En outre, cette application permet à tout médecin inscrit à ce système, quelle que soit la situation géographique du patient, de se voir également attribuer un bureau de consultation de téléconsultation en direct sans fil.

LASWENYAY prend en charge trois sections : consultation à distance, expertise à distance et surveillance à distance.

Les initiateurs du projet tentent de concilier médecine et technologie et visent l'évolution de la médecine à travers les smartphones et les ordinateurs. Pour un travail qui touche toute une nation, une escouade est mise en place.

Grâce à cet outil de santé numérique, près de dizaines de milliers d'Haïtiens sera en mesure d'utiliser la technologie moderne pour les soins de santé. Les patients pourront être en contact étroit avec les médecins un service de vidéoconférence que fournit l’application.

## 2.3 - Positionnement du système à concevoir

Tous les systèmes examinés présentent des caractéristiques très intéressantes. Cependant, la norme et le contexte particulier de la clinique dans laquelle le système a été construit doivent être pris en compte. Les plateformes explorées nous permettent de mieux nous positionner par rapport aux différents besoins auxquels la Clinique Claude Bernard doit répondre.

La méthode que nous développons se concentre sur la prédiction des maladies cardiaques les plus fréquentes pour faciliter le dépistage et l'ordonnance de traitements pour le bien-être des patients.

Le déploiement de l'intelligence artificielle est l'un des bénéfices du système que nous allons développer, cette Intelligence artificielle sera un assistant virtuel dont la mission sera de faciliter l'interprétation des résultats biométriques récupérés auprès de patients, pour en fournir des prévisions.

# 3- Critique de l'existant

Il est important de faire une enquête systématique sur les avantages et conséquences des systèmes étudiés, afin de comprendre leurs limites et leurs validités.

## 3.1 - Inventaires des problèmes

Suite à la pandémie mondiale du Coronavirus, notre système doit être adapté aux nouvelles circonstances, notamment la distanciation sociale, c'est-à- dire qu'il doit être conçu de façon à limiter les déplacements des patients tout en fournissant les services pour lesquels il a été conçu.

Les avantages de la télémédecine n'est pas très connue par le grand public en Haïti, cependant des initiatives au niveau national, mais aussi international ont vu le jour ces dernières années. De ce fait, notre inventaire des problèmes se fera d'abord autour des systèmes étudiés au niveau international et ensuite au niveau national.

### 3.1.1 - Sur le plan international

Indépendamment des avantages précédemment observés des programmes Ada Health et WebMD ils ont aussi des risques, que nous verrons dans les lignes suivantes.

Les vulnérabilités détectées en utilisant Ada Health peuvent être répertoriés comme suit:

1. Probabilité de résultats de santé mal fournis : Il existe une possibilité importante que l'existence de dossiers médicaux erronés crée une menace. L’application ne connaît pas toutes les causes pertinentes impliquées chez le patient ou peut être mal diagnostiquée.
2. Manque de communication face à face : De nombreuses personnes connaissent mieux les expériences d'humain à humain.L'application n'est pas capable de reconnaître les sentiments des patients. Par conséquent, l'étendue essentielle des besoins d'un patient peut être négligée. D'un autre côté, il peut être difficile pour certaines personnes d'utiliser les technologies modernes lorsque l'interaction humaine est peu présente.
3. Vulnérabilités dans la protection des données : Les utilisateurs peuvent avoir peur de divulguer leurs données lors de l'utilisation de l’application. Il y a une probabilité de piratage lorsqu'une interface Internet se produit. Et la sécurité des clients est une préoccupation essentielle pour tout programme d'IA, et il semble moins digne de confiance de partager les détails sur l'état de santé d'un patient d'IA que de les partager avec une personne avec les mêmes données. Les experts de la santé et les patients profiteront de toute la valeur de cette technologie en prenant conscience de ces dangers potentiels.

Les vulnérabilités détectées en utilisant WebMD peuvent être répertoriés comme suit:

1. Rien ne vaut les conseils personnalisés et le traitement de votre d'un médecin humain : Bien que votre étude des symptômes vous aidera à reconnaître les problèmes médicaux potentiels via le site web, votre stratégie thérapeutique ou votre admissibilité à une aide aux personnes handicapées ne dépendra jamais de votre propre diagnostic.
2. Inductions d’anxiété excessive : Vous trouverez un certain nombre de diagnostics potentiels, allant de symptômes bénins à des conditions potentiellement mortelles si vous recherchez des mots tels que « nausée », « fatigue », « souffrance abdominale » et « douleur articulaire ». Vous pouvez trouver impossible d'exclure ces diagnostics différentiels en utilisant la plateforme, ce qui entraîne un inconfort, une panique et une anxiété élevés.

### 3.1.2 - Sur le plan national

Sur le plan national, LASWENYAY a été conçue de manière à répondre aux besoins de la population haïtienne. Après notre analyse, nous avons constaté que ce système a pour le moment des problèmes indépendamment de ses côtés positifs. Les préoccupations énumérées sont les suivantes.

1. Les services de consultation sont très limités.
2. Les cartes d’assurance santé ne sont pas encore disponibles.
3. Les instruments biométriques ne sont pas encore mis en disponibilités.

## 3.2 - Inventaires des besoins

Cet inventaire des besoins vise à identifier tous les besoins conduisant à l'utilisation de CLCB. Les besoins les plus critiques répertoriés sont:

1. Besoins d’assistance lors des consultations en ce qui a trait aux diagnostics cardiaques des patients de la Clinique Claude Bernard.
2. Acquérir plus d'expériences, pour améliorer le système.

## 3.3 - Les opportunités

L'intégration de CLCB dans la clinique permettra de fournir un traitement et d'augmenter les performances et l'efficacité des diagnostics cardiaques. Ce qui se traduirait par une plus grande quantité de traitement de données biométriques des patients qui se ferait dans un temps record.

L'intégration de CLCB nécessitera une approche stratégique qui augmentera la responsabilité et l'ouverture tout en favorisant la créativité, encourage un accès responsable des informations recueillies afin de mieux améliorer la capacité de diagnostic et par la suite suscitera l'intérêt des patients pour les consultations dans cette clinique.

# 4. Étude de Faisabilité

Une analyse de faisabilité est réalisée afin d'évaluer l'efficacité du projet et les effets économiques et organisationnels. L'objet de l'étude de faisabilité est de vérifier que le projet est « théoriquement » conforme à la stratégie et aux ressources de l'entreprise. Le projet doit donc être abordé sous tous les angles pour valider le concept et déterminer sa viabilité technique, commerciale, économique, juridique et organisationnelle.

Cette étude de faisabilité constitue donc une condition préalable à l'analyse de marché et une condition préalable à la réussite d'un business plan.

## 4.1 - Les Objectifs

4.1.1 Objectifs visés

* Permettre aux médecins de faire la prévision de maladies cardiaques chez un patient
* Développer une plate-forme web permettant aux patients de prendre un rendez-vous
* Offrir des services à valeurs ajoutées se basant sur les nouvelles technologies.

4.1.2 Objectifs potentiels

* Développer une plate-forme web permettant aux patients d’avoir accès aux informations sur l’évolution de leur santé cardiaque.
* Développer une application permettant aux médecins de faire la gestion des consultations des clients de la clinique.
* Offrir des services à valeur ajoutée se basant sur une technologie de l’Intelligence Artificielle, l’apprentissage machine communément appelé en anglais Machine Learning.

## 4.2 - Les solutions possibles

Les objectifs ci-dessus peuvent être atteints en explorant de nombreuses méthodes potentielles concernant les innovations, les techniques et la mise en œuvre dans un environnement informatisé. Nous allons donc présenter et expliquer les solutions potentielles pour préserver une solution idéale, à savoir la solution qui correspond le mieux aux limites et aux exigences du projet.

### 4.2.1 Solutions sur le Plan Technologique

Sur le plan technologique, notre objectif est de valider ou de choisir entre les différentes technologies que nous considérerons. En effet, nous venons de définir les objectifs principaux de notre système. Pour atteindre ces objectifs, nous devons déterminer les technologies disponibles et sélectionner celles capables de répondre efficacement à nos attentes.

Nous commencerons par le fait que, d’une part, le système doit offrir aux patients de la clinique la possibilité d’une consultation en ligne sur l’évolution de leur diagnostic cardiaque, et, d’autre part, donner aux médecins la possibilité à la fois d’enregistrer les informations du patient pour en conserver l’historique de ces consultations, mais aussi de transmettre les informations générées par le diagnostic du patient au système de l’intelligence artificielle (IA) pour la prévision. Nous analyserons les possibilités technologiques qui nous permettront de réaliser ces fonctionnalités

### 4.2.2 les possibilités technologiques sur le marché

Le système que nous avons pour charge de développer consiste en une plateforme web devant tourner sur un serveur web hébergé chez un fournisseur de service internet. Aussi, proposons-nous de faire dans les sections suivantes un étalage des outils potentiels à son implémentation.

| Langage | License | Plateforme | Version | Webserver |
| --- | --- | --- | --- | --- |
| Python | libre | Windows | 3.8 | Apache, Microsoft IIS |
| C# | license | Windows | 8.0 | Apache, Microsoft IIS |
| Javascript | libre | Windows | ES6 | Apache, Microsoft IIS |
| Java | libre | Windows | SE 15.0. 1 | Apache, IIS |

### 4.2.3 Solutions sur le Plan du Déploiement

#### 4.2.3.1 Notions d'architecture client-serveur

### **Présentation de l'architecture d'un système client/serveur**

De nombreuses applications fonctionnent selon un environnement clients/[serveur](https://stph.scenari-community.org/bdd/lap2/co/webUC002archi.html#footnotesN73), 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...

Les services sont exploités par des programmes, appelés programmes clients, s'exécutant sur les machines clientes. On parle ainsi de client FTP, client de messagerie...

Dans un environnement purement client/serveur, les ordinateurs du réseau (les clients) ne peuvent voir que le serveur, c'est un des principaux atouts de ce modèle.

### **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 on peut supprimer ou rajouter des clients sans perturber le fonctionnement du réseau et sans modifications majeures

### **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**

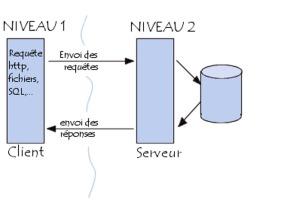
Le serveur est le seul maillon faible du réseau client/serveur, étant donné que tout le réseau est architecturé autour de lui! Heureusement, le serveur a une grande tolérance aux pannes (notamment grâce au système [RAID](https://stph.scenari-community.org/bdd/lap2/co/webUC002archi.html#footnotesNca))

### **Fonctionnement d'un système client/serveur**

Un système client/serveur fonctionne selon le schéma suivant:

· Le client émet une requête vers le serveur grâce à son **adresse** et à son **port**, qui désigne un service particulier du serveur

· Le serveur reçoit la demande et répond à l'aide de l'adresse de la machine client (et de son port)



### **Présentation de l'architecture à trois niveaux**

Dans l'architecture à 3 niveaux (appelée architecture 3-tier), il existe un niveau intermédiaire, c'est-à-dire que l'on a généralement une architecture partagée entre:

1. **Le client**

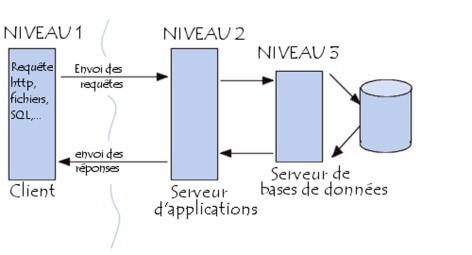
le demandeur de ressources

**2. Le serveur d'application**

(Appelé aussi *middleware*) le serveur chargé de fournir la ressource mais faisant appel à un autre serveur

**3.** **Le serveur secondaire**

(Généralement un serveur de base de données), fournissant un service au premier serveur



Étant donné l'emploi massif du terme d'architecture à 3 niveaux, celui-ci peut parfois désigner aussi les architectures suivantes :

* Partage d'application entre client, serveur intermédiaire, et serveur d'entreprise
* Partage d'application entre client, base de données intermédiaire, et base de données d'entreprise

Sur la base de nos recherches sur la prévision des maladies dans le domaine de la santé, nous nous présentons dans les sections suivantes trois solutions distinctes et non exhaustives envisagées qui permettraient d'implémenter chacune d’elles, le système de prévision de maladie cardiaque que notre client CCB a besoin pour fournir un service de qualité à leurs patients et dont la fiabilité demeure un facteur incontestable. Ces solutions tournent chacune autour de choix technologiques dont les détails sont mentionnés dans les lignes suivantes

### 4.2.3 La première solution

#### 4.2.3.1 Architecture logicielle

Les outils logiciels qui seront utilisés pour le développement de la plateforme web constituant l’architecture logicielle de notre applicatif de prévision sont présentés dans les lignes suivantes :

* Environnement de développement : Pycharm
* Framework : Django Framework
* Langage de programmation : Python
* Base de données : MySQL ou PostgreSQL
* Hébergeur :
* Protocole : HTTPS

#### 4.2.3.2 Architecture matérielle

Les outils matériels qui seront utilisés pour la mise en exploitation de la plateforme web constituant l’architecture matérielle du système sont présentés dans les lignes suivantes :

* Laptops (2) : HP Pavilion
* Switch : Linksys
* Fournisseur d'Accès Internet : DIGICEL
* Imprimante : HP LASERJET
* Câblage : Cat 6e, Connecteurs RJ-45, Moulures, 3 Prises Murales

### 4.2.4 - La deuxième solution

#### 4.2.4.1 Architecture logicielle

Les outils logiciels qui seront utilisés pour le développement de la plateforme web constituant l’architecture logicielle de notre applicatif de prévision sont présentés dans les lignes suivantes:

* Environnement de développement : visual studio code
* Framework : .Net Framework
* Langage de programmation : C#
* Base de données : SQL SERVER
* Hébergeur :

#### 4.2.4.2 Architecture matérielle

Les outils matériels qui seront utilisés pour la mise en exploitation de la plateforme web constituant l’architecture matérielle du système sont présentés dans les lignes suivantes :

* Ordinateurs (laptops) : HP Pavilion
* Switch : Linksys
* Fournisseur d'Accès Internet : DIGICEL
* Imprimante : HP LASERJET
* Câblage: Cat 6e, Connecteurs RJ-45, Moulures, 3 Prises Murales

### 4.2.5 - La troisième solution

#### 4.2.5.1Architecture logicielle

Les outils logiciels qui seront utilisés pour le développement de la plateforme web constituant l’architecture logicielle de notre applicatif de prévision sont présentés dans les lignes suivantes:

* Environnement de développement : Netbeans

La plate-forme NetBeans est un vaste cadre Java sur lequel vous pouvez baser de grandes applications de bureau. L'EDI NetBeans lui-même est l'une des centaines d'applications basées sur la plate-forme NetBeans. La NetBeans Platform contient des API qui simplifient la manipulation des fenêtres, des actions, des fichiers et de bien d'autres choses typiques des applications.

* Framework : Spring

Spring Framework est un Framework de développement d'applications Web puissant et léger, classé premier en raison de son excellente capacité à développer des applications Web complexes nécessitant des performances élevées. Les fonctionnalités principales de Spring Framework permettent aux développeurs Java de créer facilement des applications de niveau entreprise.

* Langage de programmation : Java
* Base de données : ORACLE
* Machine Learning Librairie : JavaML

Il s'agit d'une API Java avec une collection d'algorithmes d'apprentissage automatique et d'exploration de données implémentés en Java. Il est destiné à être facilement utilisé par les développeurs de logiciels et les chercheurs. Les interfaces pour chacun des algorithmes restent simples et faciles à utiliser. Il n'y a pas d'interface graphique mais des interfaces claires pour chaque type d'algorithme. Comparé à d'autres algorithmes de clustering, il est simple et permet une facilité de mise en œuvre d'un nouvel algorithme. Dans la plupart des cas, la mise en œuvre des algorithmes est clairement écrite et correctement documentée, et peut donc être utilisée comme référence. La bibliothèque est écrite en Java.

#### 4.2.5.2 Architecture matérielle

Les outils matériels qui seront utilisés pour la mise en exploitation de la plateforme web constituant l’architecture matérielle du système sont présentés dans les lignes suivantes :

* Ordinateurs (2 laptops) : HP Pavilion
* Switch : Linksys
* Wireless Router : Linksys
* Fournisseur d'Accès Internet : DIGICEL
* Imprimante : HP DESKJET
* Câblage : Cat 6e, Connecteurs RJ-45, Moulures, 3 Prises Murales double

## 4.3 - Bénéfices et coûts de chaque solution

### 4.3.1 - Bénéfices de chaque solution

#### 4.3.1.1 benefice de la première solution

Nous présentons dans les lignes suivantes les bénéfices de cette solution :

* Tous les outils de développement sont libres, le client n’aura pas payer pour des licences
* Les outils utilisés sont les meilleurs dans le domaine de l’IA et du ML sur le marché
* Le langage comprend les meilleures bibliothèques pour l’IA et le ML
* Le système développé sera multi-plateforme
* Le système de gestion de base de données est stable et performant

#### 4.3.1.2 bénéfice de la deuxième solution

Nous présenterons dans les lignes suivantes quelques bénéfices de cette solution:

* Les outils de développement sont propriétaires
* Le système de gestion de base de données est très performant et sécuritaire

#### 4.3.1.3 Bénéfice de la troisième solution

Nous présenterons dans les lignes suivantes quelques bénéfices de cette solution:

* Les outils de développement sont propriétaires
* Support garantie de la part du propriétaire du l’outil
* Le système de gestion de base de données est très performant, sécuritaire et fiable

### 4.3.2 - Coûts de chaque solution

En vue de synthétiser les informations sur les coûts de mise en œuvre de chaque solution, nous présenterons celles-ci sous forme de tableau dans les sections ci-dessous. Ces coûts seront s'échelonnent sur aspects :

* Coût des logiciels
* Coût des matériels
* Coût de développement
* Coût d’installation matérielle

#### 

#### 

#### 4.3.2.1 - Coûts de la première solution

| **Rubrique** | **Description** | **Coûts** |
| --- | --- | --- |
| **Coût des Logiciels** | Langage: Python | $ 0.00 |
| Framework : Django | $ 0.00 |
| SGBD: MySQL | $ 2000.00 |
| IDE : Pycharm | $ 0.00 |
|  |  | $ 0.00 |
| **Coût des Matériels** | 2 Laptops: New XPS 13 DELL Laptop | $2000.00 |
| 1 imprimante: [HP Neverstop Laser MFP 1202w](https://store.hp.com/us/en/pdp/hp-neverstop-laser-mfp-1202w) | $330.00 |
| 1 Linksys WiFi Router Dual-Band AC1000 (WiFi 5) | $39.00 |
| 1Linksys SE3008 8-Port Gigabit Ethernet Switch | $50.00 |
| FAI: Access Haiti | $99.00 |
| CABLE Cat 6e, Moulures, Connecteurs RJ-45, 3 Prises Murales double | $400.00 |
| Main d’oeuvre câblage | $750.00 |
|  | Hébergement | $6 x12/m |
|  |  | $3740.00 |
| **Coût de Développement de la solution** | | $4000.00 |
| **Coût Total pour la mise en oeuvre de la solution** | | $9740.00 |

#### 4.3.2.2 - Coût de la deuxième solution

| **Rubrique** | **Description** | **Coûts** |
| --- | --- | --- |
| **Coût des Logiciels** | Langage: C# | $ 0.00 |
| Framework : .Net | $ 0.00 |
| SGBD: SQL SERVER ENTERPRISE | $ 13748.00 |
| IDE : Visual Studio Professional | $1,199.00 |
| **Coût Total des Logiciels** | | $14947.00 |
| **Coût des Matériels** | 2 Laptops: New XPS 13 DELL Laptop | $999.99 |
| 1 imprimante: [HP Neverstop Laser MFP 1202w](https://store.hp.com/us/en/pdp/hp-neverstop-laser-mfp-1202w) | $329.99 |
| 1 Linksys WiFi Router Dual-Band AC1000 (WiFi 5) | $39.00 |
| 1Linksys SE3008 8-Port Gigabit Ethernet Switch | $50.00 |
| 1 Fournisseur d'Accès Internet |  |
| CABLE Cat 6e, Moulures, Connecteurs RJ-45, 3 Prises Murales double | $400 |
| Main d’oeuvre câblage | $1000 |
|  | Hébergeur: https://www.webhostinghub.com/ | $5.99x12/m |
| **Coût Total des Matériels** | | $2890.98 |
| **Coût de Développement de la solution** | | $4000.00 |
| **Coût Total pour la mise en oeuvre de la solution** | | $21837.98 |
|  |  |  |

#### 4.3.2.3 - Coût de la troisième solution

| **Rubrique** | **Description** | **Coûts** |
| --- | --- | --- |
| **Logiciels** | Langage: Java | $ 0.00 |
| Framework : Spring | $ 0.00 |
| SGBD: ORACLE | $ 47500.00 |
| IDE : NetBeans | $ 0.00 |
| **Coût Total des Logiciels** | | **$ 47500.00** |
| **Matériels** | 2 Laptops: New XPS 13 DELL LAPTOP | $2000.00 |
| 1 imprimante: [HP Neverstop Laser MFP 1202w](https://store.hp.com/us/en/pdp/hp-neverstop-laser-mfp-1202w) | $330.00 |
| 1 Linksys WiFi Router Dual-Band AC1000 (WiFi 5) | $39.00 |
| 1Linksys SE3008 8-Port Gigabit Ethernet Switch | $50.00 |
| 1 FAI : Access Haiti | $99.00 x12/m |
| CABLE Cat 6e, Moulures,  24 Connecteurs RJ-45,  3 Prises Murales double | $400 |
| Main d’oeuvre câblage | $750 |
|  | Hébergeur: https://www.webhostinghub.com/ | $5.99x 12/ m |
| **Coût Total des Matériels et Installation** | | **$52257.00** |
| **Coût de Développement de la solution** | | **$4000.00** |
| **Coût Total pour la mise en oeuvre de la solution** | | **$56257.00** |
|  |  |  |

## 4.4 - Choix de la solution

En comparaison des bénéfices précités sur l’ensemble des trois solutions envisagées, notre choix s'arrête sur la première solution pour l'implémentation du système de prévision de maladies cardiaques en vue de répondre aux besoins de notre client.

## 4.5 - Risque et enjeux

L’utilisation de l’intelligence artificielle dans le domaine de la santé est en pleine expansion et apporte un support importe très intéressant dans bien des champs tels que:

* Aide à la décision
* Chirurgie assistée par ordinateur
* Prévention épidémiologique
* Médecine prédictive

Toutefois, cette utilisation soulève des certains risques, tels que:

* la disparition de certains métiers qui seront remplacés par une intelligence artificielle
* la notion de confiance entre patient et son médecin dans les diagnostics risquent de s’amoindrir considérablement en se reposant davantage sur la capacité de la machine que sur la capacité du médecin
* la confidentialité des données du patient n’est pas garantie
* Le risque de l’erreur de la machine n’est pas négligeable, dans ce cas qui sera responsable de celui-ci.

## 

# **Partie II**

Spécifications des exigences du produit

Cette section du document identifie les besoins du client par des fonctionnalités décrivant les services fournis et les restrictions imposées au système.

## 2.1 - Description général du produit

Le CLCB (Clinique Claude Bernard) **est** un site qui fournit des services de diagnostics de maladies cardio-vasculaires via apprentissage machine. Le CLCBest conçu pour optimiser les travaux des agents médicaux (Cardiologues, Infirmières …) qui parfois ont du mal à travailler et s’adapter efficacement selon les nouvelles technologies. LeCLCB est capable de donner des résultats en temps record, les informations proviennent du cardiologue après avoir diagnostiqué les patients.

### 2.1.1 - Les parties du système

Le système est constitué de plusieurs parties, qui sont les utilisateurs, les composants matériels, les composants logiciels.

### 2.1.2 - Des utilisateurs

Un utilisateur est une personne ou un robot faisant appel aux ressources de l'informatique pour mener à bien ses activités qui n’est pas forcément un informaticien. Pourvu que notre travail soit une plateforme Web, ce dernier exige aux utilisateurs d’avoir de la connaissance autour de l’internet pour un meilleur usage.

✔ **Patients** : Toutes personnes possédant un compte lui permettant d’accéder aux ressources qui lui sont allouées ou prédisposées par le système. Ces derniers pourront accéder à certaines fonctionnalités qui seront décrites plus bas.

✔ **Administrateur** : Il a accès à toutes les fonctionnalités disponibles du système mais il peut aussi restreindre ou donner des droits aux personnes qui lui sont importantes.

✔ **Secrétaire** : Cette catégorie d’utilisateur pourrait accéder à toutes les fonctionnalités suivant le cas, cependant ses droits sont limités a priori.

2.1.3 - Composantes matérielles du système

Un VPS (virtual private server) qui est serveur privé virtuel dédié chez un Fournisseur sera utilisé comme support aux pages Web du site du Clinique Claude Bernard afin de maintenir la robustesse du système.

2.1.4 - Composantes du Système

L’application Web Clinique Claude Bernard s’exécutera sur les divers systèmes d’exploitation (SE) comme les familles de Windows et Unix. Cette dernière sera munie d’un SGBD pour maintenir la pertinence et la sécurité des données.

2.1.5 - Portabilité du Système

Le protocole utilisé sera TCP/IP en raison que la Clinique garantit aux patients un service en ligne de très grande qualité pour les navigateurs tels que Google Chrome, Mozilla, Firefox et Internet Explorer.

2.1.6 - Contrainte de mémoire

La contrainte de mémoire est en relation avec la quantité des informations reçues et stockées sur le serveur. Ces informations sont stockées sur les disques durs du serveur de données et seront continuellement mises à jour. La taille de la mémoire sera calculée en fonction de la quantité des gens inscrits sur le système par jour.

2.1.7 - Adaptation du site d’installation

Afin d’assurer la réception des doléances des Patients en temps réel et une meilleure performance du système, La Clinique doit avoir une connexion Internet fiable et de haut débit. Vu les multiples contraintes que nous avons en Haïti pour un service Internet de bonne qualité. Il est nécessaire d'avoir plusieurs connexions de fournisseurs différents pour assurer l'opérationnalité du système.

2.1.8 - Fonctionnalité du Système

La fonctionnalité de notre système se tiendra en compte sur une application distribuée.

2.1.8.1 - Application distribuée

Les fonctionnalités énumérées pour l’application distribuée sont les suivantes :

* Authentifier
* Créer Un diagnostic
* Modifier Mot-de-Passe
* Afficher un rendez-vous
* Afficher les résultats d’analyses
* Imprimer les résultats d’analyses
* Creer un Patient
* Modifier un Patient
* Recherche un Patient
* Supprimer un Patient
* Créer Un Rendez-vous
* Modifier un Rendez-vous
* Recherche un Rendez-vous
* Supprimer un rendez-vous
* Modifier un Diagnostic
* Recherche un Diagnostic
* Supprimer un Diagnostic
* Fermer session

### 2.2 - Description des fonctionnalités du système

| Fonctionnalités | Objectifs | Descriptions | Priorité |
| --- | --- | --- | --- |
| Authentifier | Permet de contrôler l’accès au contenu du site | L’authentification est nécessaire pour l'utilisation du produit. Pour s’authentifier, un usager doit entrer son email et son mot de passe. | Vital |
| Creer Un diagnostic | Permet au patients d’avoir une idée concernant leur santé | c’est une relève biométrique du patient. | Vital |
| Modifier Mot-de-Passe | Gérer la sécurité | Permet aux utilisateurs de garder la sécurité de leur compte. | Vital |
| Afficher un rendez-vous | Permet de visualiser la rendez-vous | Permet de visualiser la date et l’heure auxquelles les Patients ont un rendez-vous avec le Médecin. | Vital |
| Afficher les résultats d’analyses | Visualiser les résultats | Permet aux patients de voir les rendez-vous avec le Médecin. | Vital |
| Imprimer les résultats d’analyses | Imprimer les résultats | Avoir les résultats d’analyses en dur | Vital |
| Créer un Patient | Création de patient | Les patients seront creer par le staff ayant le droit | Important |
| Modifier un Patient | Modification des patients | Permet de modifier les patients | Important |
| Recherche un Patient | Recherche des patients | Permet de rechercher des patients | Important |
| Supprimer un Patient | Suppression des patient | Permet de supprimer des patient | Important |
| Créer Un Rendez-vous | Création de rendez-vous | Permet de création de rendez-vous | Important |
| Modifier un Rendez-vous | Modification de rendez-vous | Permet de modifier les rendez-vous | Important |
| Recherche un Rendez-vous | Recherche de rendez-vous | Permet la recherche des rendez-vous | Important |
| Supprimer un rendez-vous | Suppression de rendez-vous | Permet de supprimer des rendez-vous | Important |
| Modifier un Diagnostic | Modification de diagnostic | Permet la modification des diagnostics | Important |
| Recherche un Diagnostic | Recherche de diagnostic | Permet la recherche de diagnostics | Important |
| Supprimer un Diagnostic | Suppression de diagnostic | Permet la suppression de diagnostic | Important |
| Fermer session | Fermeture de sessions | Permet la fermeture de sessions | Vital |

### 2.1.4 - Contraintes d'ordre général

**Limitation du matériel :** Toutes les transactions et demandes potentielles sur l'application doivent être acceptables pour le serveur pendant les heures de pointe.

c'est-à-dire qu'elle peut supporter jusqu’à 500 000 connexions au maximum.

**Protocoles de communication :** Le protocole TCP/IP sera nécessaire pour accéder au système grâce à une connexion Internet via un fournisseur Internet.

**Aspect critique de l'application :** La gestion de l’application nécessite la supervision

d'un médecin en tant qu'administrateur et le secrétaire en tant qu'assistant.

**Considération de sûreté et de sécurité :** La confidentialité et l’encryption des données est assurée par l’utilisation du protocole https et ssl.

### 2.1.5 - Répartitions des exigences

Ce qui doit être fait en priorité est la génération des diagnostics via apprentissage automatique. Le développement de l’application web se fera en parallèle avec la création de l’IA.

## 2.2 - Description détaillée

Dans cette partie, la description détaillée du système sera abordée et plus spécifiquement les cas d’utilisations, les interfaces, le modèle objet, le modèle relationnel et l’architecture de notre système.

### 2.2.1 - Les cas d'utilisation

Les différents cas d’utilisations identifiés sont les suivants :

* Cas d’utilisation S'authentifier
* Cas d’utilisation Fermer session
* Cas d’utilisation Modifier Mot-de-Passe
* Cas d’utilisation Créer Un Rendez-vous
* Cas d’utilisation Modifier un Rendez-vous
* Cas d’utilisation Recherche un Rendez-vous
* Cas d’utilisation Supprimer un rendez-vous
* Cas d’utilisation Afficher un rendez-vous
* Cas d’utilisation Afficher les résultats d’analyses
* Cas d’utilisation Imprimer les résultats d’analyses
* Cas d’utilisation Créer un Patient
* Cas d’utilisation Modifier un Patient
* Cas d’utilisation Recherche un Patient
* Cas d’utilisation Supprimer un Patient
* Cas d’utilisation Créer Un diagnostic
* Cas d’utilisation Modifier un Diagnostic
* Cas d’utilisation Recherche un Diagnostic
* Cas d’utilisation Supprimer un Diagnostic

### 2.2.2 - Scénarios des cas d'utilisation

1. **Titre**: S’authentifier

**But** : S’authentifier au système

**Acteurs** : Patient, médecin, secrétaire

**Déroulement nominal** :

| **Utilisateurs** | **Système** |
| --- | --- |
| Saisir son nom d'utilisateur et son mot de passe |  |
|  | Valider du nom et du mot de passe |
|  | Vérifier si c'est un médecin |
|  | Vérifier si c'est un secrétaire |
|  | Vérifier si c'est un patient |
|  | Afficher les fonctionnalités disponibles pour cet utilisateur |

1. **Titre**: Modifier Mot-de-Passe

**But** : Modification du mot-de-passe

**Acteurs** : Patient, médecin, secrétaire

**Déroulement nominal** :

| **Utilisateurs** | **Système** |
| --- | --- |
| Demande de modifier son mot de passe |  |
|  | Affiche le champ de saisie pour l'ancien et nouveau mot de passe |
| Rempli les champs nécessaires |  |
|  | Demande de confirmer le nouveau mot de passe |
| Entre à nouveau le nouveau mot de passe |  |
|  | Valide les informations reçues |
|  | Confirme la modification de mot de passe |

1. **Titre**: Créer Un Rendez-vous

**But** : Création de Rendez-vous

**Acteurs** : médecin, secrétaire

**Déroulement nominal** :

| **Utilisateurs** | **Système** |
| --- | --- |
| Demande de créer un rendez-vous |  |
|  | Affiche les champs de saisies |
| Fournit les informations |  |
|  | Demande la confirmation de la création |
| Confirme les informations fournies |  |
|  | Valide les informations reçues |
|  | Confirme la validation |

1. **Titre** : Modifier un Rendez-vous

**But** : Modification de Rendez-vous

**Acteurs** : médecin, secrétaire

**Déroulement nominal** :

| **Utilisateurs** | **Système** |
| --- | --- |
| Demande de faire une recherche de rendez-vous |  |
|  | Affiche la page de recherche de rendez-vous |
| Sélectionne les critères de recherche |  |
| Lance la recherche |  |
|  | Effectue la recherche |
|  | Affiche le résultat de la recherche |
| Demande de modification du rendez-vous |  |
| Saisie des nouvelles informations relatives au rendez-vous |  |
|  | Demande de confirmer les informations relatives au rendez-vous |
| Confirme la demande de modification de rendez-vous |  |
|  | Valide les informations reçues |
|  | Modifie le rendez-vous |
|  | Confirme la modification du rendez-vous |

1. **Titre** : Rechercher un Rendez-vous

**But** : Recherche de Rendez-vous

**Acteurs** : médecin, secrétaire

**Déroulement nominal** :

| **Utilisateur** | **Système** |
| --- | --- |
| Demande de faire une recherche de rendez-vous |  |
|  | Affiche la page de recherche de rendez-vous |
| Sélectionne les critères de recherche |  |
| Lance la recherche |  |
|  | Effectue la recherche |
|  | Affiche le résultat de la recherche |

1. **Titre** : Modifier un Rendez-vous

**But** : Modification de Rendez-vous

**Acteurs** : médecin, secrétaire

**Déroulement nominal** :

| **Utilisateurs** | **Système** |
| --- | --- |
| Demande de faire une recherche de rendez-vous |  |
|  | Affiche la page de recherche de rendez-vous |
| Sélectionne les critères de recherche |  |
| Lance la recherche |  |
|  | Effectue la recherche |
|  | Affiche le résultat de la recherche |
| Demande de modification du rendez-vous |  |
| Saisie des nouvelles informations relatives au rendez-vous |  |
|  | Demande de confirmer les informations relatives au rendez-vous |
| Confirme la demande de modification de rendez-vous |  |
|  | Valide les informations reçues |
|  | Modifie le rendez-vous |
|  | Confirme la modification du rendez-vous |

1. **Titre** : Rechercher un Rendez-vous

**But** : Recherche de Rendez-vous

**Acteurs** : médecin, secrétaire

**Déroulement nominal** :

| **Utilisateur** | **Système** |
| --- | --- |
| Demande de faire une recherche de rendez-vous |  |
|  | Affiche la page de recherche de rendez-vous |
| Sélectionne les critères de recherche |  |
| Lance la recherche |  |
|  | Effectue la recherche |
|  | Affiche le résultat de la recherche |

1. **Titre** : Supprimer un rendez-vous

**But** : Suppression de rendez-vous

**Acteurs** : médecin, secrétaire

**Déroulement nominal** :

| **Utilisateurs** | **Système** |
| --- | --- |
| Demande de faire une recherche de rendez-vous |  |
|  | Affiche la page de recherche de rendez-vous |
| Sélectionne les critères de recherche |  |
| Lance la recherche |  |
|  | Effectue la recherche |
|  | Affiche le résultat de la recherche |
| Demande de faire la suppression du rendez-vous |  |
|  | Demande la confirmation de suppression du rendez-vous |
| Confirme la demande |  |
|  | Effectue la suppression |
|  | Confirme la suppression du rendez-vous |

1. **Titre** : Afficher un rendez-vous

**But** : Affichage de rendez-vous

**Acteurs** : patients, médecin, secrétaire

**Déroulement nominal** :

| **Utilisateur** | **Système** |
| --- | --- |
| Fournir les critères d'affichage de rendez-vous |  |
|  | Afficher les rendez-vous |

1. **Titre** : Afficher les résultats d'analyses

**But** : Affichage des résultats d'analyses

**Acteurs** : patients, médecin, secrétaire

**Déroulement nominal** :

| **Utilisateur** | **Système** |
| --- | --- |
| Fournir les critères d'affichage de résultats d'analyses |  |
|  | Afficher les résultats d'analyses |

1. **Titre** : Créer un Patient

**But** : Création un Patient

**Acteurs** : médecin

**Déroulement nominal** :

| **Utilisateur** | **Système** |
| --- | --- |
| Demande de créer un patient |  |
|  | Affiche les champs de saisies |
| Fournit les informations |  |
|  | Demande la confirmation de la création |
| Confirme les informations fournies |  |
|  | Valide les informations reçues |
|  | Confirme la validation |

1. **Titre** : Modifier un Patient

**But** : Modification de Patients

**Acteurs** : médecin

**Déroulement nominal** :

| **Utilisateur** | **Système** |
| --- | --- |
| Demande de faire une recherche de patient |  |
|  | Affiche la page de recherche de patients |
| Sélectionne les critères de recherche |  |
| Lance la recherche |  |
|  | Effectue la recherche |
|  | Affiche le résultat de la recherche |
| Demande de modification de patient |  |
| Saisie des nouvelles informations relatives au patient |  |
|  | Demande de confirmer les informations relatives au patient |
| Confirme la demande de modification de patient |  |
|  | Valide les informations reçues |
|  | Modifie le patient |
|  | Confirme la modification du patient |

1. **Titre** : Recherche un Patient

**But** : Recherche de Patients

**Acteurs** : médecin

**Déroulement nominal** :

| **Utilisateur** | **Système** |
| --- | --- |
| Demande de faire une recherche de patient |  |
|  | Affiche la page de recherche de patients |
| Sélectionne les critères de recherche |  |
| Lance la recherche |  |
|  | Effectue la recherche |
|  | Affiche le résultat de la recherche |

1. **Titre** : Supprimer un Patient

**But** : Suppression de Patients

**Acteurs** : médecin

**Déroulement nominal** :

| **Utilisateur** | **Système** |
| --- | --- |
| Demande de faire une recherche de patient |  |
|  | Affiche la page de recherche de patients |
| Sélectionne les critères de recherche |  |
| Lance la recherche |  |
|  | Effectue la recherche |
|  | Affiche le résultat de la recherche |
| Demande de faire la suppression du patient |  |
|  | Demande la confirmation de suppression du patient |
| Confirme la demande |  |
|  | Effectue la suppression |
|  | Effectue la suppression |

1. **Titre** : Créer Un diagnostic

**But** : Création de diagnostics

**Acteurs** : médecin

**Déroulement nominal** :

| **Utilisateurs** | **Systèmes** |
| --- | --- |
| Demande de créer un diagnostic |  |
|  | Affiche les champs de saisies |
| Fournit les informations |  |
|  | Demande la confirmation de la création |
| Confirme les informations fournies |  |
|  | Valide les informations reçues |
|  | Confirme la validation |

1. **Titre** : Modifier un Diagnostic

**But** : Modification de Diagnostics

**Acteurs** : médecin

**Déroulement nominal** :

| **Utilisateurs** | **Systèmes** |
| --- | --- |
| Demande de faire une recherche de diagnostics |  |
|  | Affiche la page de recherche de diagnostics |
| Sélectionne les critères de recherche |  |
| Lance la recherche |  |
|  | Effectue la recherche |
|  | Affiche le résultat de la recherche |
| Demande de modification du diagnostic |  |
| Saisie des nouvelles informations relatives au diagnostic |  |
|  | Demande de confirmer les informations relatives au diagnostic |
| Confirme la demande de modification de diagnostic |  |
|  | Valide les informations reçues |
|  | Modifie le diagnostic |
|  | Confirme la modification du diagnostic |

1. **Titre** : Recherche un Diagnostic

**But** : Recherche de Diagnostic

**Acteurs** : médecin

**Déroulement nominal** :

| **Utilisateurs** | **Systèmes** |
| --- | --- |
| Demande de faire une recherche de diagnostics |  |
|  | Affiche la page de recherche de diagnostics |
| Sélectionne les critères de recherche |  |
| Lance la recherche |  |
|  | Effectue la recherche |
|  | Affiche le résultat de la recherche |

1. **Titre** : Supprimer un Diagnostic

**But** : Suppression de Diagnostic

**Acteurs** : médecin

**Déroulement nominal** :

| **Utilisateurs** | **Systèmes** |
| --- | --- |
| Demande de faire une recherche de diagnostics |  |
|  | Affiche la page de recherche de diagnostics |
| Sélectionne les critères de recherche |  |
| Lance la recherche |  |
|  | Effectue la recherche |
|  | Affiche le résultat de la recherche |
| Demande de faire la suppression du diagnostic |  |
|  | Demande la confirmation de suppression du diagnostic |
| Confirme la demande |  |
|  | Effectue la suppression |
|  | Confirme la suppression du diagnostic |

## 2.3 - Exigences logiques de base de données

### 2.3.1 - Capacité d'accès

L'accès au système doit être très rapide, en particulier lors des diagnostics et de la visualisation des données. Le système doit pouvoir fonctionner 24 heures sur 24, 7 jours sur 7, c'est-à-dire en permanence.

### 2.3.2 - Persistance

La persistance du système est relative à la structure des données permanentes. Voici la liste de quelques classes principales auxquelles le système se base :

1. Utilisateur
2. Groupe
3. Dossier
4. Diagnostic
5. Prescription
6. Rendez-vous

### 2.3.3 - Fréquence d'utilisation

Les données concernant les patients sont utilisées par le médecin de la clinique. Ce sont des données critiques, de ce fait elles ont été sauvegardées et encryptées dans une base de données. Avant toute accès ou modification de ces données disponibles, le système effectue une authentification.

## 2.4 - Contraintes de conception

Le système est pris en charge par un serveur virtuel connecté au Web. Le système possède une IA capable de prédiction concernant les maladies cardiaques. La confidentialité des données personnelles est obligatoire. Lors de l'authentification, le cryptage des données n'est utilisé que pour empêcher des personnes non autorisées de les lire afin d'assurer plus de sécurité. Sans l'approbation du client concerné, ou sans l'autorisation de la loi, nous ne divulguerons aucune information.

## 2.5 - Exigences non-fonctionnelles

Les exigences du système sont réparties en plusieurs points :

1. **La fiabilité** :

La gestion et le stockage du système doit s’effectuer en toute confiance par le système, car la fiabilité est l'une des propriétés clés à considérer lors de la création de systèmes informatiques.

1. **L’ergonomie** :

Le un système doit être facilement utilisable par n’importe quels utilisateurs de manière intuitive.

1. **Le rendement** :

Le système doit effectuer, correctement et en temps réel, les tâches auxquelles il a été conçu sans problème majeur.

1. **La portabilité** :

Le système doit avoir la capacité de s'exécuter sur une variété de plateformes (Tels Que ordinateurs et mobiles) correctement avec peu ou pas de modifications.

1. **La maintenabilité** :

La maintenabilité est la capacité de mener à bien une action de réparation dans un délai donné. En outre, la capacité de maintenance mesure la simplicité et la vitesse avec lesquelles le système doit être remis dans un état opérationnel après une défaillance.

1. **La sécurité** :

La sécurité des systèmes englobe tous les aspects de l'accès aux ressources informatiques. De l'authentification aux mises à jour du système, en passant par la protection antivirus et les modifications, la sécurité est un élément clé du fonctionnement optimal du système. Ces bonnes pratiques visent à prévenir les problèmes de sécurité.

1. **La disponibilité** :

La disponibilité du système signifie que les informations sont accessibles aux utilisateurs autorisés. Elle donne l'assurance que les utilisateurs authentifiés peuvent accéder à leurs données chaque fois qu'ils en ont besoin.

# Partie III

# Dossier d'Analyse

## 3.1 - Méthodologie Utilisée

Sans méthodologie appropriée, il n'y a pas de travail scientifique. Puisque la cohérence des résultats attendus est développée par la méthode utilisée pour la conception du projet. Le processus suivi pour atteindre les objectifs fixés consiste en des analyses de tous les documents consultés ainsi que des observations de faits réels.

Il existe une grande quantité de méthodes conçues pour exécuter des projets informatiques, cependant on peut prendre comme exemples des méthodes largement utilisées de nos jours. Par exemple, il y a RUP, OOD, OMT, OOSE et Merise qui sont des candidats robustes pour l'élaboration du projet. En ce qui concerne notre projet nous avions choisi RUP.

L'objectif fondamental du Rational Unified Process (RUP) est de fournir un modèle pour la mise en œuvre efficace d'approches de développement éprouvées sur le plan commercial, à utiliser tout au long du cycle de vie du développement logiciel. Prenant des éléments d'autres modèles de développement de logiciels itératifs, RUP été initialement créé par Rational Software Corporation, qui a été racheté par IBM en 2003[[10]](#footnote-9).

RUP est basé sur quelques idées fondamentales, telles que les phases de développement et les éléments constitutifs, qui définissent qui, quoi, quand et comment le développement aura lieu. RUP se compose de six bonnes pratiques principales:

1. Développement itératif de logiciels: Cette pratique permet d'encourager le développement itératif en localisant et en travaillant sur les éléments à haut risque à chaque phase du cycle de vie du développement logiciel.
2. Gestion des exigences : Cette pratique permet de décrire comment organiser et suivre les exigences de fonctionnalité, la documentation, les compromis et les décisions, ainsi que les exigences commerciales.
3. Utilisation des architectures basées sur des composants : Cette pratique permet de mettre l'accent sur le développement qui se concentre sur les composants logiciels réutilisables dans ce projet et, surtout, dans les projets futurs.
4. Modélisation visuelle : Cette pratique permet la visualisation sur le langage de modélisation unifié (UML). RUP fournit les moyens de modéliser visuellement le logiciel, y compris les composants et leurs relations les uns avec les autres.
5. Vérification de la qualité du logiciel : Cette pratique permet d’aider à la conception, à la mise en œuvre et à l'évaluation de toutes sortes de tests tout au long du cycle de vie du développement logiciel.
6. Contrôles les modifications du logiciel : Cette pratique permet d'écrire comment suivre et de gérer toutes les formes de changement qui se produiront inévitablement tout au long du développement, afin de produire des itérations réussies d'une génération à l'autre.

Tous les aspects du RUP sont basés sur un ensemble de blocs de construction, qui sont utilisés pour décrire ce qui doit être produit, qui est chargé de le produire, comment la production aura lieu et quand la production sera terminée. Ces quatre éléments de base sont les :

1. Travailleurs : Le comportement et les responsabilités d’un individu, ou d’un groupe d’individus réunis en équipe, travaillant sur toute activité afin de produire des artefacts.
2. Activités : unité de travail qu’un travailleur doit effectuer. Les activités doivent avoir un objectif clair, généralement en créant ou en mettant à jour des artefacts.
3. Artefacts : un artefact représente tout résultat tangible qui émerge du processus ; des nouvelles fonctions de code aux documents en passant par des modèles de cycle de vie supplémentaires.
4. Workflows : Représente une séquence d'activités schématisée, afin de produire une valeur et des artefacts observables.

Le RUP a quatre phases de cycle de vie, tel que :

1. Phase de lancement : Pendant la phase de démarrage, l'idée de base et la structure du projet sont déterminées. Durant cette phase l'équipe s'assiéra et déterminera si le projet vaut la peine d'être poursuivi, en fonction de l'objectif proposé du projet, des coûts estimés (monétaires et en temps) et des ressources nécessaires pour achever le projet une fois le feu vert donné.
2. Phase d’élaboration : Le but de la phase d'élaboration est d'analyser les exigences et l'architecture nécessaire du système. Le succès de cette phase est particulièrement critique, car la dernière étape de cette phase signifie la transition du projet de faible risque à haut risque puisque le développement et le codage réels auront lieu dans la phase suivante.
3. Phase de construction : La phase de construction correspond au moment où le codage et la mise en œuvre de toutes les fonctionnalités de l'application auront lieu. Cette période est également celle où les intégrations avec d'autres services ou logiciels existants doivent avoir lieu.
4. Phase de transition : La phase de transition correspond au moment où le produit fini est enfin lancé et livré aux clients. Cependant, la phase de transition est plus qu'un simple processus de déploiement ; il doit également gérer tout le support post-version, les corrections de bogues, les correctifs, etc.

La méthodologie RUP recommande également que chacune des quatre phases ci-dessus soit subdivisée en itérations. Une itération représente simplement un cycle complet des phases principales mentionnées, jusqu'à ce qu'un produit soit publié sous une forme ou une autre, en interne ou en externe. À partir de cette référence, la prochaine itération peut être modifiée si nécessaire jusqu'à ce qu'un produit complet et complet soit mis à la disposition des clients.

RUP comprend des avantage, tels que :

1. RUP permet la capacité d'adaptation pour faire face à l'évolution des exigences tout au long du cycle de vie du développement, qu'elles proviennent des clients ou du projet lui-même.
2. RUP souligne la nécessité (et la bonne mise en œuvre) d'une documentation précise.
3. RUP diffuse les maux de tête d'intégration potentiels en forçant l'intégration à se produire tout au long du développement, en particulier dans la phase de construction où tous les autres codages et développements ont lieu.

Et aussi des désavantages, tels que :

1. RUP dépend fortement de membres de l'équipe compétents et experts, car l'attribution d'activités à des travailleurs individuels devrait produire des résultats tangibles et préplanifiés sous la forme d'artefacts.
2. Étant donné l'accent mis sur l'intégration tout au long du processus de développement, cela peut également être préjudiciable pendant les tests ou d'autres phases, où les intégrations sont en conflit et gênent d'autres activités plus fondamentales.
3. On peut soutenir que RUP est un modèle assez compliqué. Étant donné l'assortiment de composants impliqués, y compris les meilleures pratiques, les phases, les éléments constitutifs, les critères d'étape, les itérations et les flux de travail, la mise en œuvre et l'utilisation appropriées du processus unifié rationnel peuvent souvent être difficiles pour de nombreuses organisations, en particulier pour les petites équipes ou projets.

## 3.2 - Présentation d'UML

UML, abréviation de Unified Modeling Language, est un langage de modélisation standardisé composé d'un ensemble intégré de diagrammes, développé pour aider les développeurs de systèmes et de logiciels à spécifier, visualiser, construire et documenter les artefacts des systèmes logiciels, ainsi que pour la modélisation d'entreprise et autres systèmes non logiciels.

L'UML représente un ensemble de meilleures pratiques d'ingénierie qui ont fait leurs preuves dans la modélisation de systèmes vastes et complexes. L'UML est une partie très importante du développement de logiciels orientés objet et du processus de développement de logiciels. L'UML utilise principalement des notations graphiques pour exprimer la conception de projets logiciels. L'utilisation de l'UML aide les équipes de projet à communiquer, à explorer les conceptions potentielles et à valider la conception architecturale du logiciel.

Le but d'UML est de fournir une notation standard utilisable par toutes les méthodes orientées objet et de sélectionner et intégrer les meilleurs éléments des notations précurseurs. UML a été conçu pour une large gamme d'applications. Par conséquent, il fournit des constructions pour un large éventail de systèmes et d'activités.

L'UML tente de fournir un langage si expressif que toutes les parties prenantes peuvent bénéficier d'au moins un diagramme UML. Voici un aperçu rapide de chacun de ces 13 diagrammes comme indiqué dans la structure de diagramme UML 2 ci-dessous :

Les diagrammes de structure montrent la structure statique du système et de ses parties à différents niveaux d'abstraction et d'implémentation et comment ils sont liés les uns aux autres. Les éléments d'un diagramme de structure représentent les concepts significatifs d'un système et peuvent inclure des concepts abstraits, réels et de mise en œuvre.Il existe sept types de diagramme de structure comme suit:

1. Diagramme de classe
2. Diagramme des composants
3. Diagramme de déploiement
4. Diagramme d'objets
5. Diagramme de paquet
6. Diagramme de structure composite
7. Diagramme de profil

Les diagrammes de comportement montrent le comportement dynamique des objets dans un système, qui peut être décrit comme une série de modifications du système au fil du temps, il existe sept types de diagrammes de comportement comme suit :

1. Diagramme de cas d'utilisation
2. Diagramme d'activité
3. Diagramme de machine d'état
4. Diagramme de séquençage
5. Diagramme de communication
6. Diagramme de présentation des interactions
7. Diagramme de temps

L’utilisation d’UML comporte des avantages, tels que :

1. UML est hautement reconnue et comprise pour la conception de logiciels. C'est une notation standard parmi les développeurs de logiciels. Vous pouvez supposer en toute sécurité que la plupart des professionnels du logiciel connaîtront au moins, sinon bien, les diagrammes UML, ce qui en fait l'alternative incontournable pour expliquer les modèles de conception de logiciels.
2. UML est un langage riche et étendu qui peut être utilisé pour modéliser non seulement l'ingénierie logicielle orientée objet, mais aussi la structure et le comportement des applications, ainsi que les processus métier.

UML comporte aussi des désavantages, tels que :

1. Beaucoup de développeurs ne préfèrent pas la complexité et la formalité au niveau architectural, ce qui décourage l’utilisation d’UML.
2. Depuis son lancement jusqu'à maintenant, UML a gagné en complexité et en taille. La taille même de l'UML rend beaucoup de gens nerveux dès le début, et ils ont l'impression de ne pas pouvoir l'apprendre et sont mieux sans cela.

## 3.3 - Présentation du système du point de vue statique et dynamique

Dans cette section nous allons présenter le système statiquement et dynamiquement via des diagrammes. Du point de vue statique un diagramme de cas d’utilisation du système sera présenté et du point de vue dynamique, des diagrammes de cas d’utilisation, et d'activités seront présentés.

### 3.3.1- Diagramme de cas d'utilisation (vue statique)

Dans le langage de modélisation unifié (UML), un diagramme de cas d'utilisation résume les détails des utilisateurs également appelés acteurs, et leurs interactions avec le système. Le but d'un diagramme de cas d'utilisation UML est de démontrer les différentes façons dont les utilisateurs pourront interagir avec le système. Ce type de diagramme utilise un ensemble de symboles et de connecteurs spécialisés. Un diagramme de cas d'utilisation efficace permet de discuter et de représenter :

1. Les scénarios dans lesquels le système interagit avec des personnes, des organisations ou des systèmes externes.
2. Les objectifs du système à atteindre.
3. La portée du système.

Lorsque nous prévoyons de dessiner un diagramme de cas d'utilisation, nous devrions avoir les éléments suivants identifiés.

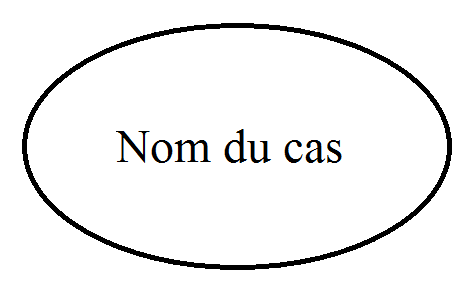
1. Les fonctionnalités à représenter comme cas d'utilisation.
2. Les acteurs ou utilisateurs.
3. Les relations entre les cas d'utilisation et les acteurs.

Après avoir identifié les éléments ci-dessus, les directives suivantes doivent être utilisées pour dessiner un diagramme de cas d'utilisation efficace :

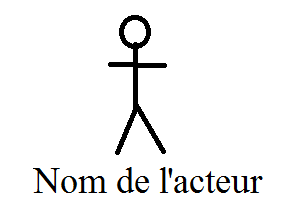
1. Le nom d'un cas d'utilisation est très important. Le nom doit être choisi de manière à pouvoir identifier les fonctionnalités réalisées.
2. Un nom approprié doit être donné aux acteurs.
3. Les relations et les dépendances doivent être clairement indiquées dans le diagramme.

La notation d'un diagramme de cas d'utilisation est assez simple et n'implique pas autant de types de symboles que les autres diagrammes UML. Voici les formes que vous pourrez y trouver :

1. Cas d’utilisation : Ovales de forme horizontale qui représentent les différentes utilisations qu'un utilisateur peut avoir.



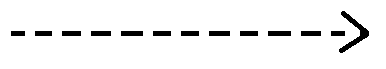
1. Acteurs : Figures qui représentent les personnes employant réellement les cas d'utilisation.



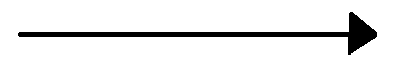
1. Associations : Ligne entre les acteurs et les cas d'utilisation. Dans les diagrammes complexes, il est important de savoir quels acteurs sont associés à quels cas d'utilisation.



1. Relation d’inclusion et d’extension : La relation d'inclusion « include » permet de montrer la dépendance qui existe entre le cas d'utilisation de base et le cas d'utilisation qui en est inclus. Alors que la relation de l'extension «extends» est une chose qui est facultative dans le schéma. Ils sont représentés par une ligne fractionnée.

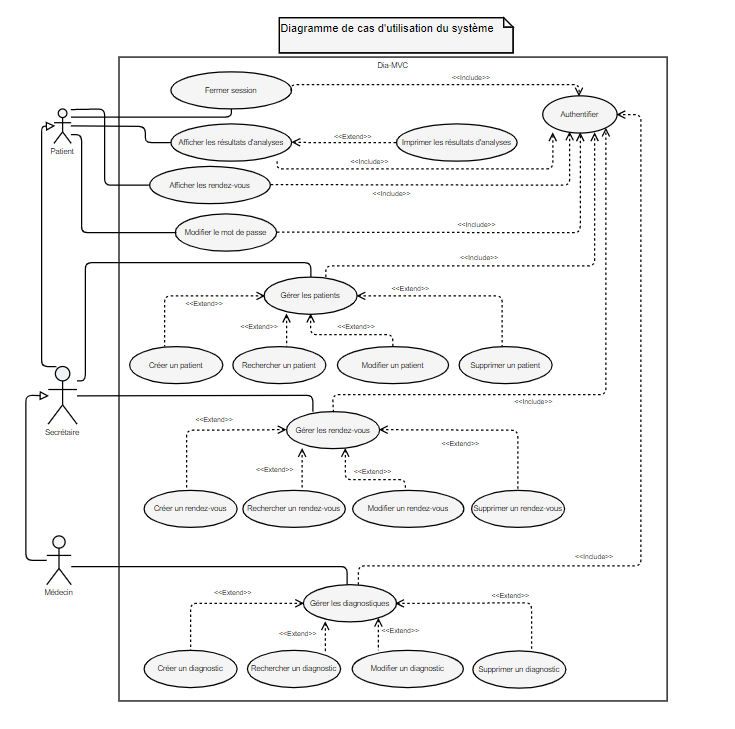


1. Relation de généralisation : La généralisation est le seul lien possible entre les deux acteurs. Le symbole utilisé pour généraliser les acteurs est une flèche avec une ligne continue qui a un triangle fermé qui identifie l'acteur le plus important.



1. Boîtes de délimitation du système : une zone qui définit une portée système pour les cas d'utilisation. Tous les cas d'utilisation en dehors de la boîte seraient considérés comme hors du champ d'application de ce système.





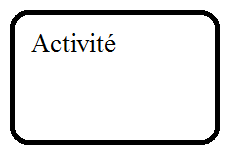
### 

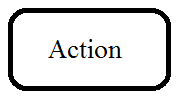
### 

### 3.3.2 - Diagramme d'activités (vue dynamique)

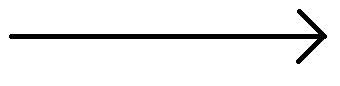
Un diagramme d'activité est un diagramme de comportement important pour décrire les aspects dynamiques du système. Les diagrammes d'activités décrivent comment les activités sont coordonnées pour fournir un service. En règle générale, un événement doit être atteint par certaines opérations, en particulier lorsque l'opération est destinée à réaliser un certain nombre de choses différentes qui nécessitent une coordination, ou comment les événements dans un cas d'utilisation unique sont liés les uns aux autres, en particulier, les cas d'utilisation où les activités peuvent se chevaucher et exiger une coordination.

Voici la liste de notations qu’un diagramme de d'activités peut avoir :

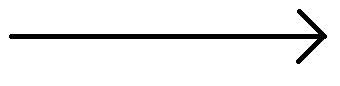
1. Activité : Est utilisé pour représenter un ensemble d'actions.
2. Action : Une tâche à effectuer.



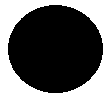
1. Flux de contrôle : Affiche la séquence d'exécution.



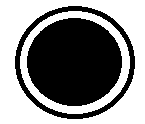
1. Flux d’objets : Affiche le flux d'un objet d'une activité (ou action) à une autre activité (ou action).



1. Nœud initial : Représente le début d'un ensemble d'actions ou d'activités.



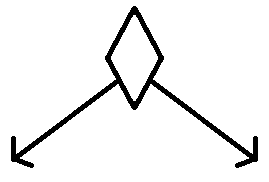
1. Nœud final d’activité : Arrête tous les flux de contrôle et les flux d'objets dans une activité (ou action).



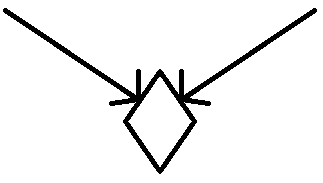
1. Nœud d’objet : Représente un objet connecté à un ensemble de flux d'objets.



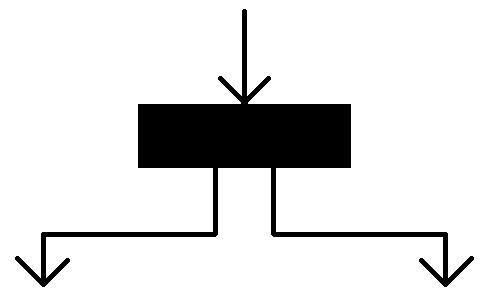
1. Nœud de décision : Représente une condition de test pour garantir que le flux de contrôle ou le flux d'objets ne suit qu'un seul chemin.



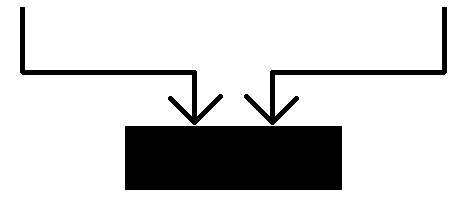
1. Nœud de fusion : Rassemble les différents chemins de décision créés à l'aide d'un nœud de décision.



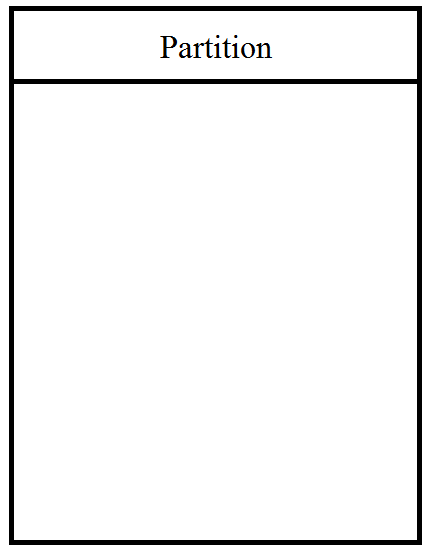
1. Nœud de fourche : Divise le comportement en un ensemble de flux parallèles ou simultanés d'activités (ou d'actions).



1. Nœud de jointure : Rassemble un ensemble de flux parallèles ou simultanés d'activités (ou d'actions).

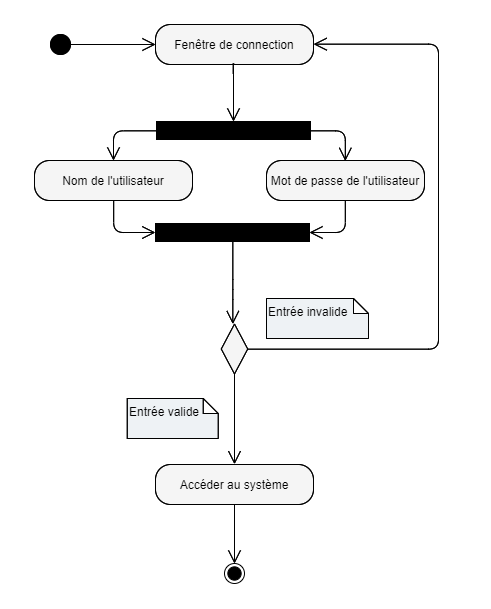


1. Couloir et partition : Moyen de regrouper les activités exécutées par le même acteur sur un diagramme d'activités ou de regrouper les activités dans un seul fil.

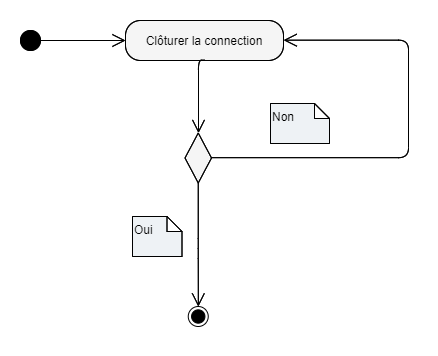


Voici la liste des diagrammes d'activités des cas d’utilisations du système:

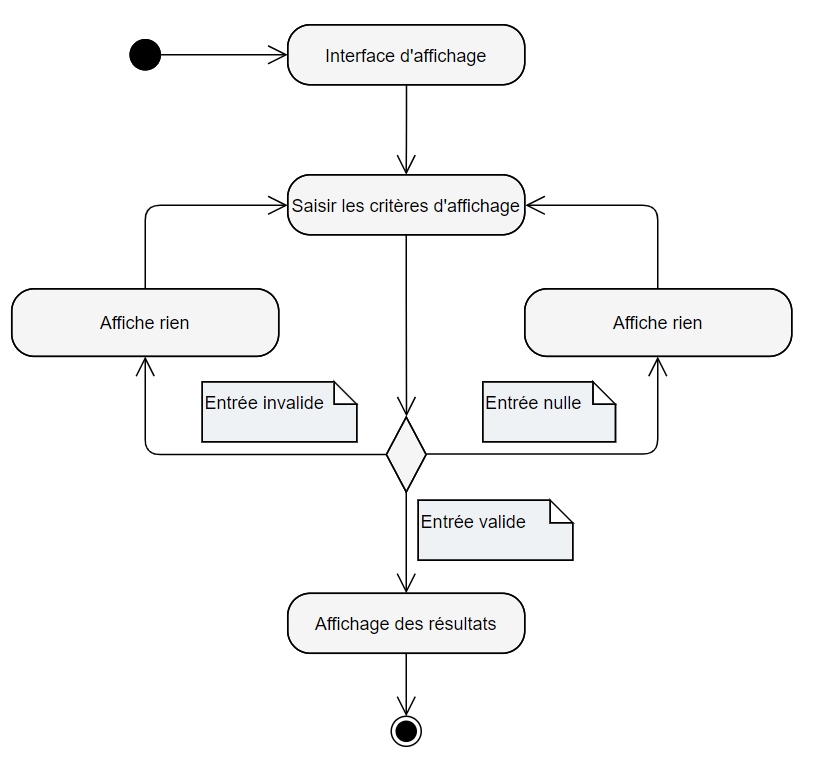
1. Diagramme d’activité pour le cas d’utilisation : Authentifier



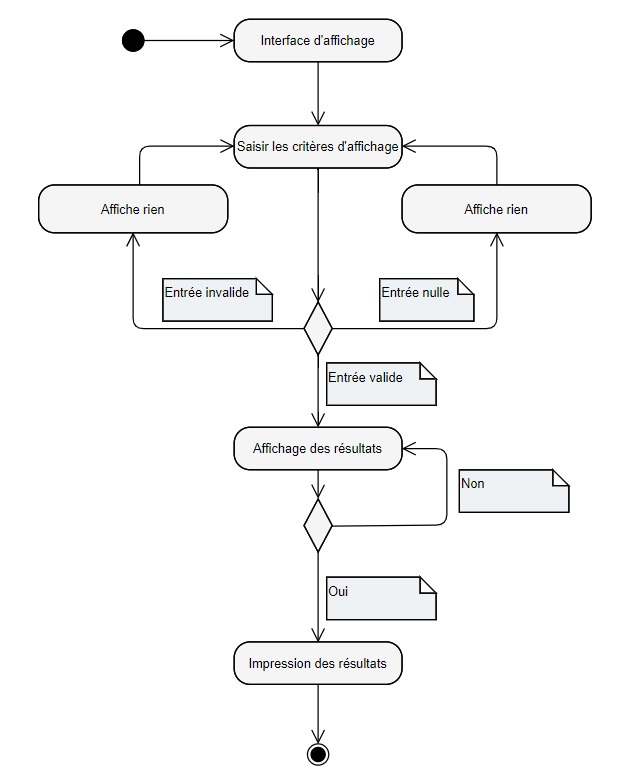
1. Diagramme d’activité pour le cas d’utilisation : Fermer session



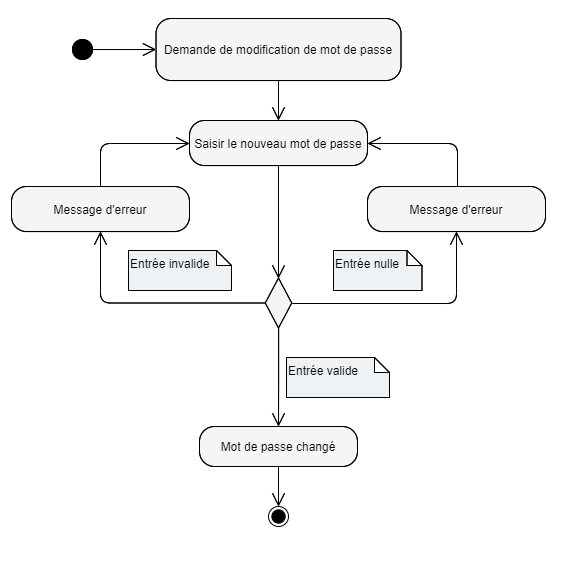
1. Diagramme d’activité pour le cas d’utilisation : Afficher les résultats d'analyses



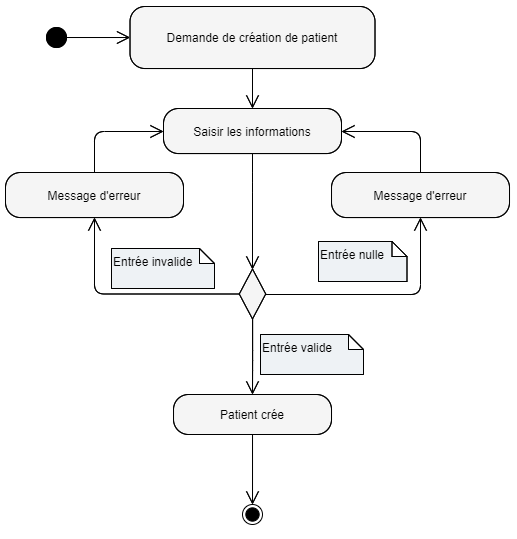
1. Diagramme d’activité pour le cas d’utilisation : Imprimer les résultats d'analyses



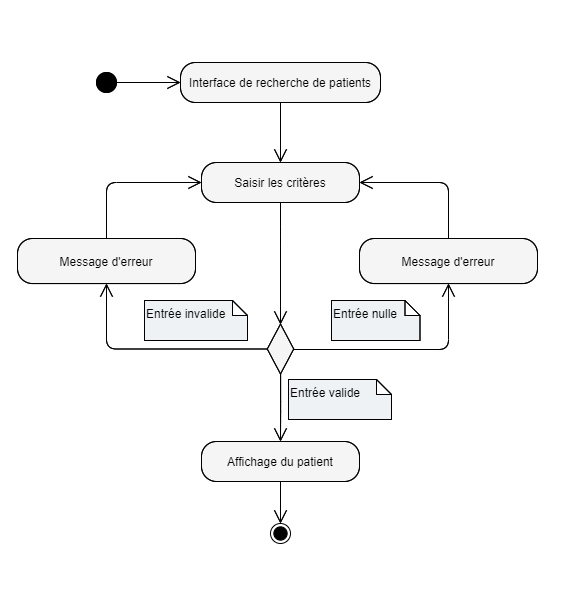
1. Diagramme d’activité pour le cas d’utilisation : Modifier le mot de passe



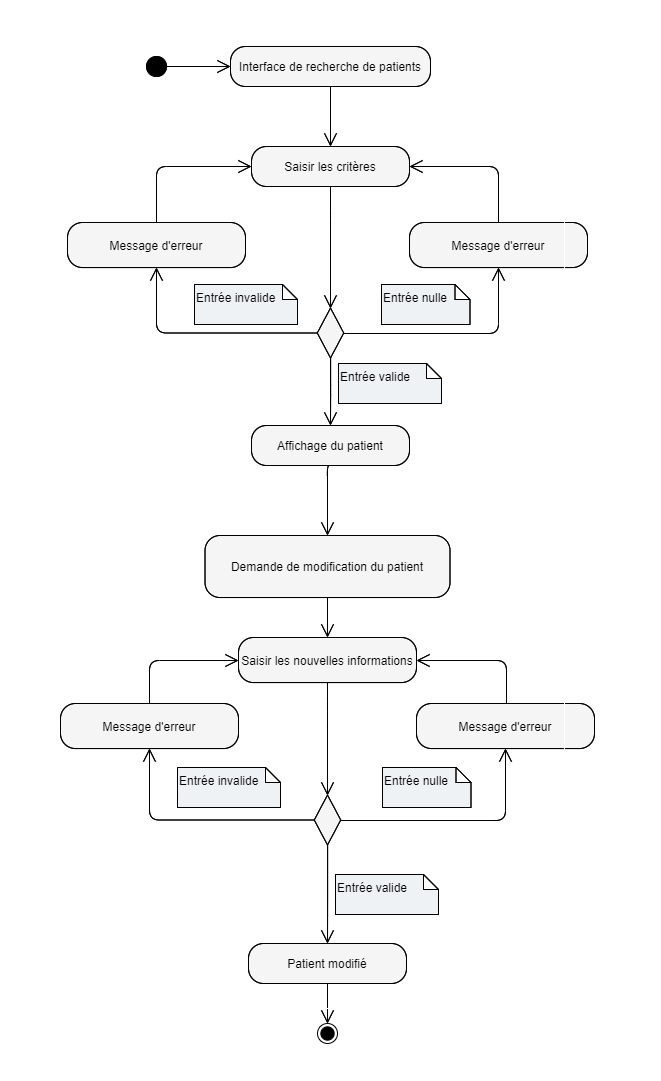
1. Diagramme d’activité pour le cas d’utilisation : Créer un patient



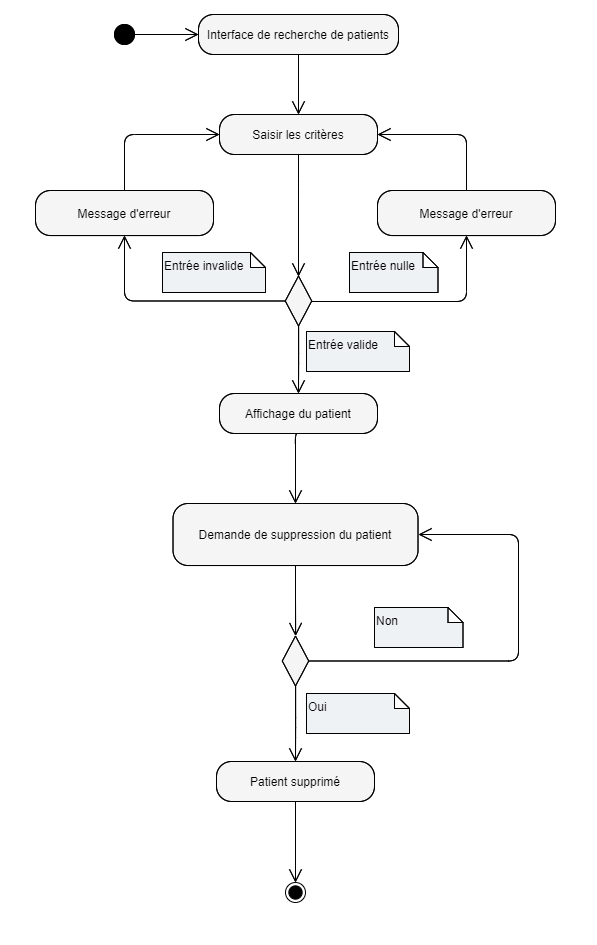
1. Diagramme d’activité pour le cas d’utilisation : Rechercher un patient



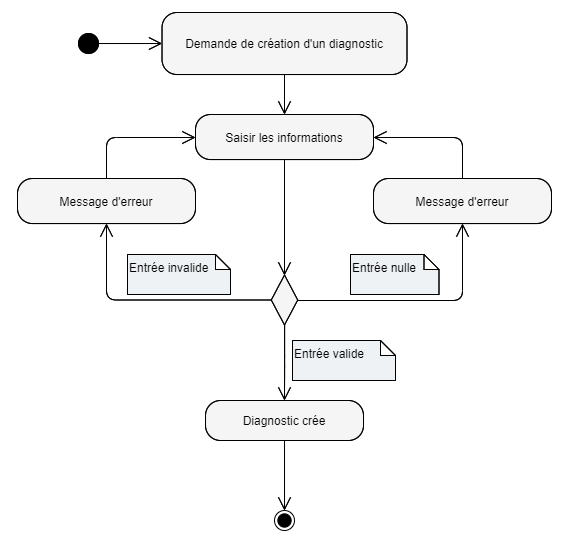
1. Diagramme d’activité pour le cas d’utilisation : Modifier un patient



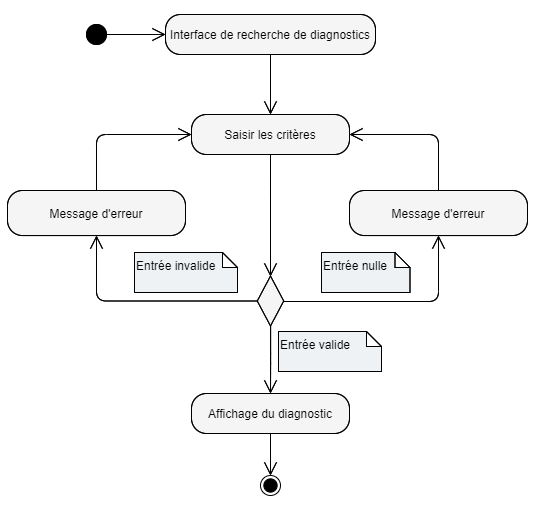
1. Diagramme d’activité pour le cas d’utilisation : Supprimer un patient



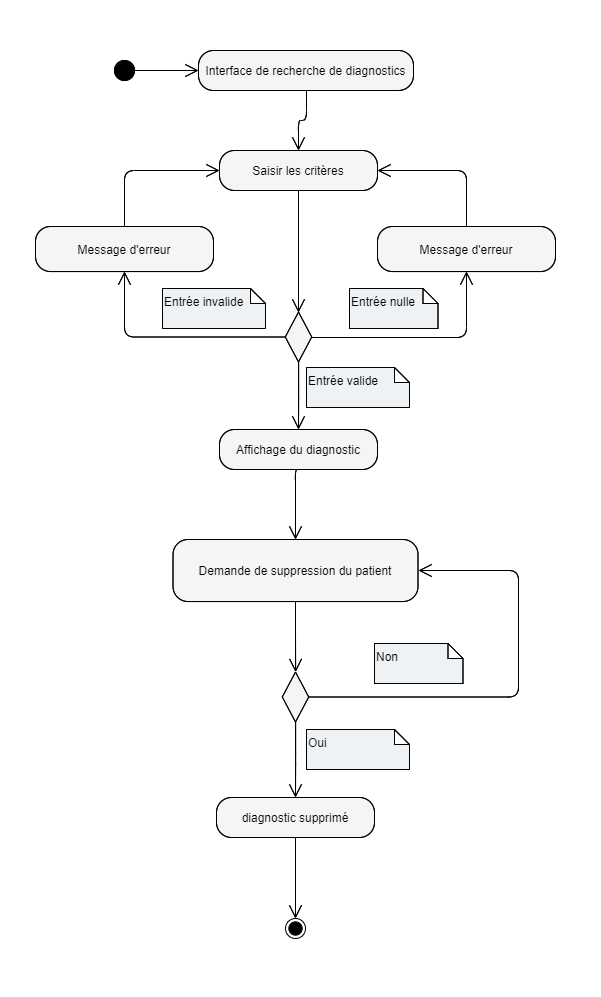
1. Diagramme d’activité pour le cas d’utilisation : Créer un diagnostic



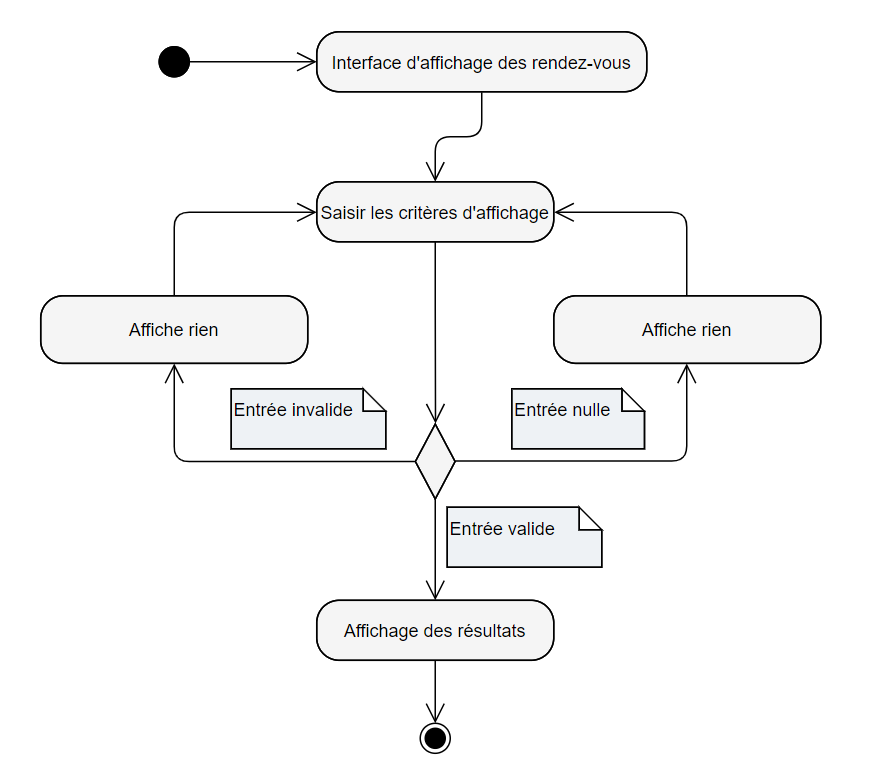
1. Diagramme d’activité pour le cas d’utilisation : Rechercher un diagnostic



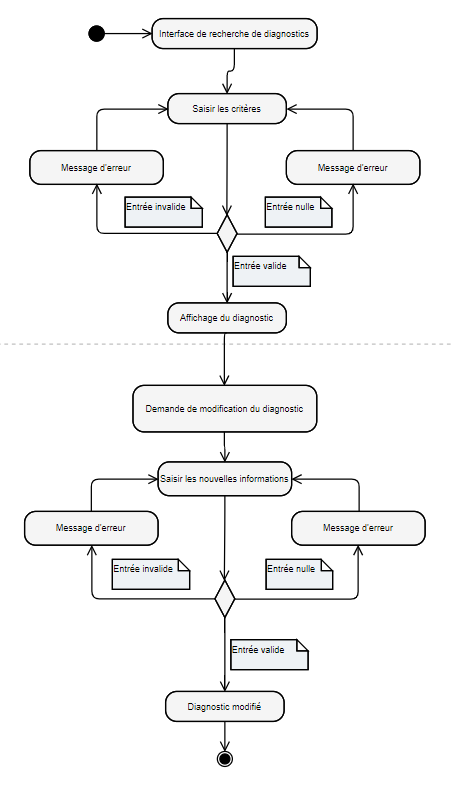
1. Diagramme d’activité pour le cas d’utilisation: Supprimer un diagnostic



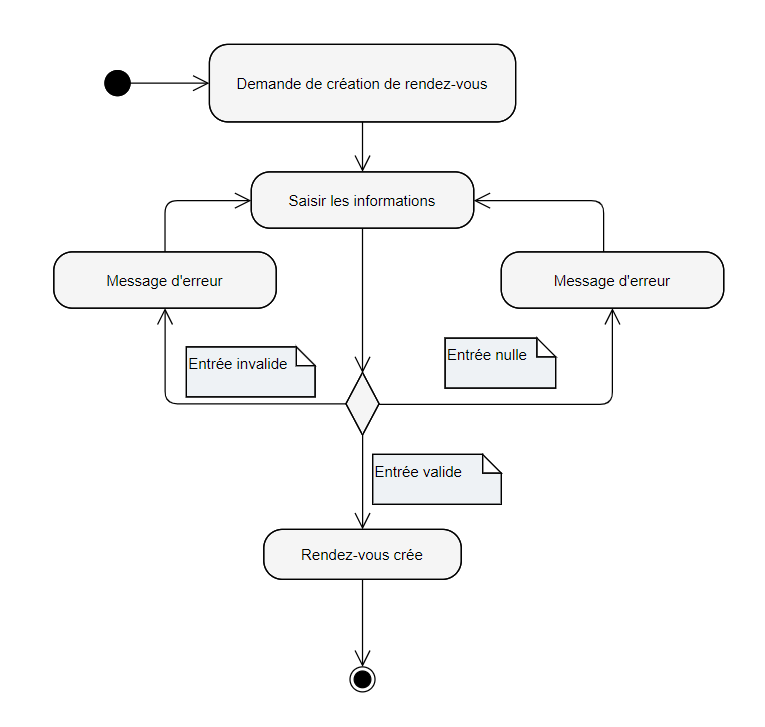
1. Diagramme d’activité pour le cas d’utilisation: Afficher les rendez-vous



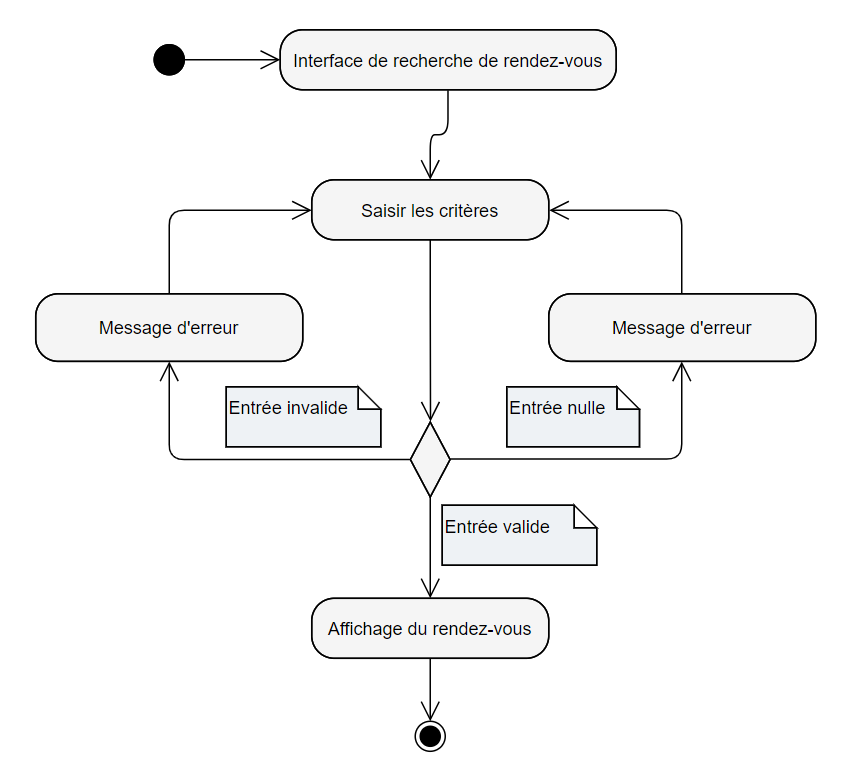
1. Diagramme d’activité pour le cas d’utilisation : Modifier un diagnostic



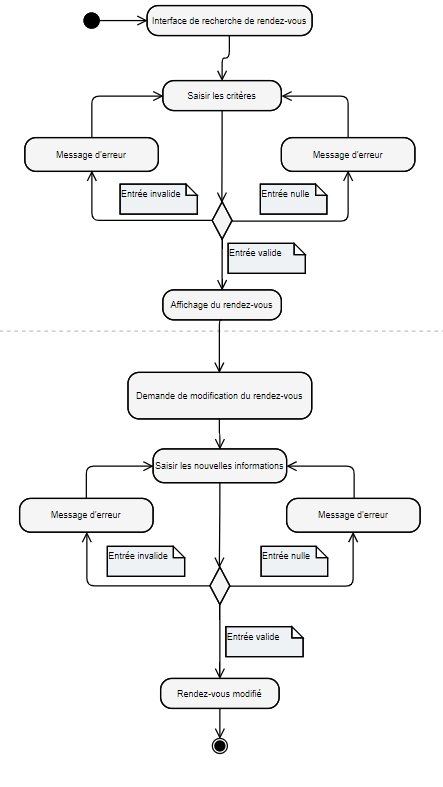
1. Diagramme d’activité pour le cas d’utilisation : Créer un rendez-vous



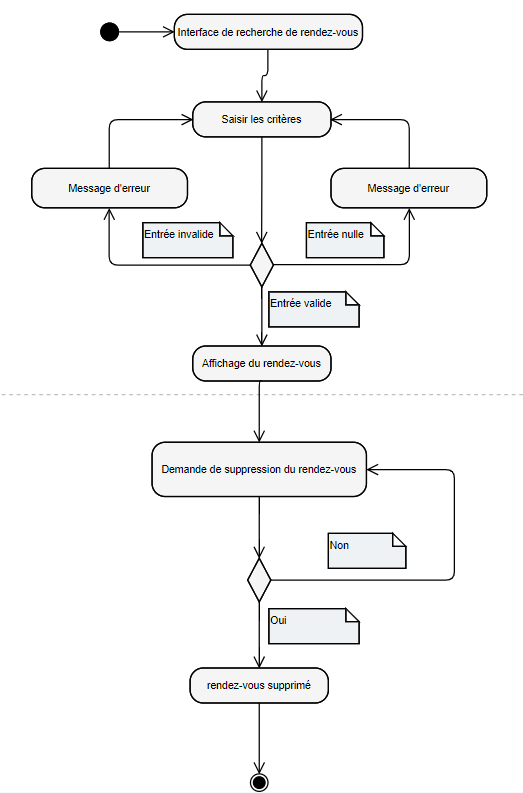
1. Diagramme d’activité pour le cas d’utilisation: Rechercher un rendez-vous



1. Diagramme d’activité pour le cas d’utilisation: Modifier un rendez-vous



1. Diagramme d’activité pour le cas d’utilisation: Supprimer un rendez-vous



### **3.3.2 - Diagramme de séquences (vue dynamique)**

Les diagrammes de séquence sont des diagrammes d'interaction qui détaillent la manière dont les opérations sont effectuées, c'est-à-dire que grâce aux diagrammes de séquences on peut savoir quels messages sont envoyés et quand ils ont été envoyés.

Les diagrammes de séquence sont organisés en fonction du temps. Le temps progresse au fur et à mesure que vous descendez la page. Les objets impliqués dans l'opération sont répertoriés de gauche à droite en fonction du moment où ils participent à la séquence de messages.

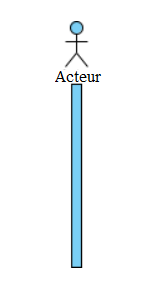
Ces diagrammes capturent l'interaction entre les objets dans le contexte d'une collaboration. Ils sont focalisés sur le temps et ils montrent l'ordre de l'interaction visuellement en utilisant l'axe vertical du diagramme pour représenter le temps, quels messages sont envoyés et quand.

Les diagrammes de séquence capturent :

1. L'interaction qui a lieu dans une collaboration qui réalise soit un cas d'utilisation, soit une opération (diagrammes d'instances ou diagrammes génériques)
2. Les interactions de haut niveau entre l'utilisateur du système et le système, entre le système et d'autres systèmes, ou entre les sous-systèmes (parfois appelés diagrammes de séquence système)

Voici une liste de notation que peut comporter un diagramme de séquence :

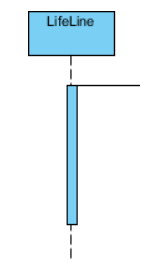
1. Acteur : représentent les rôles joués par des utilisateurs humains, du matériel externe ou d'autres sujets.



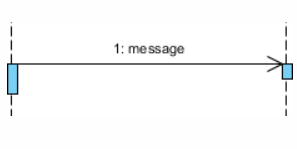
1. Lifeline (Ligne de vie): Une Ligne de vie représente un participant individuel à l'interaction.



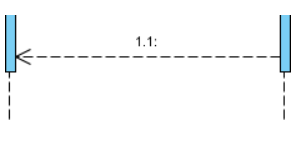
1. Activations: représente la période pendant laquelle un élément effectue une opération.



1. Message d'appel: le message d'appel est un type de message qui représente une invocation d'une opération de la ligne de vie cible.



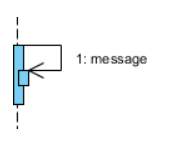
1. Message de retour: le message de retour est une sorte de message qui représente la transmission d'informations à l'appelant d'un ancien message correspondant.



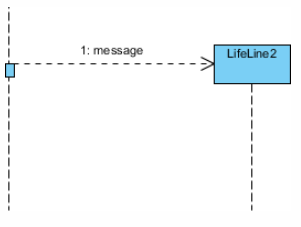
1. Self Message (message personnel): Self message est une sorte de message qui représente l'invocation du message de la même Lifeline.



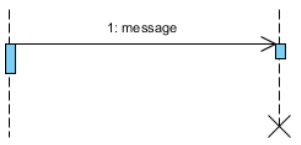
1. Message récursif: le message récursif est une sorte de message qui représente l'invocation du message de la même ligne de vie. Sa cible pointe vers l'activation en plus de l'activation d'où le message a été appelé.



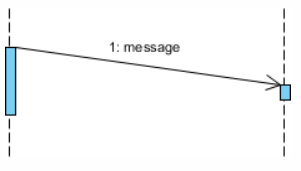
1. Create Message (Créer un message): Create message est une sorte de message qui représente l'instanciation de la ligne de vie (cible).



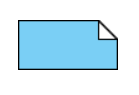
1. Message de destruction: le message de destruction est un type de message qui représente la demande de destruction du cycle de vie de la ligne de vie cible.



1. Message de durée: le message de durée indique la distance entre deux instants pour un appel de message

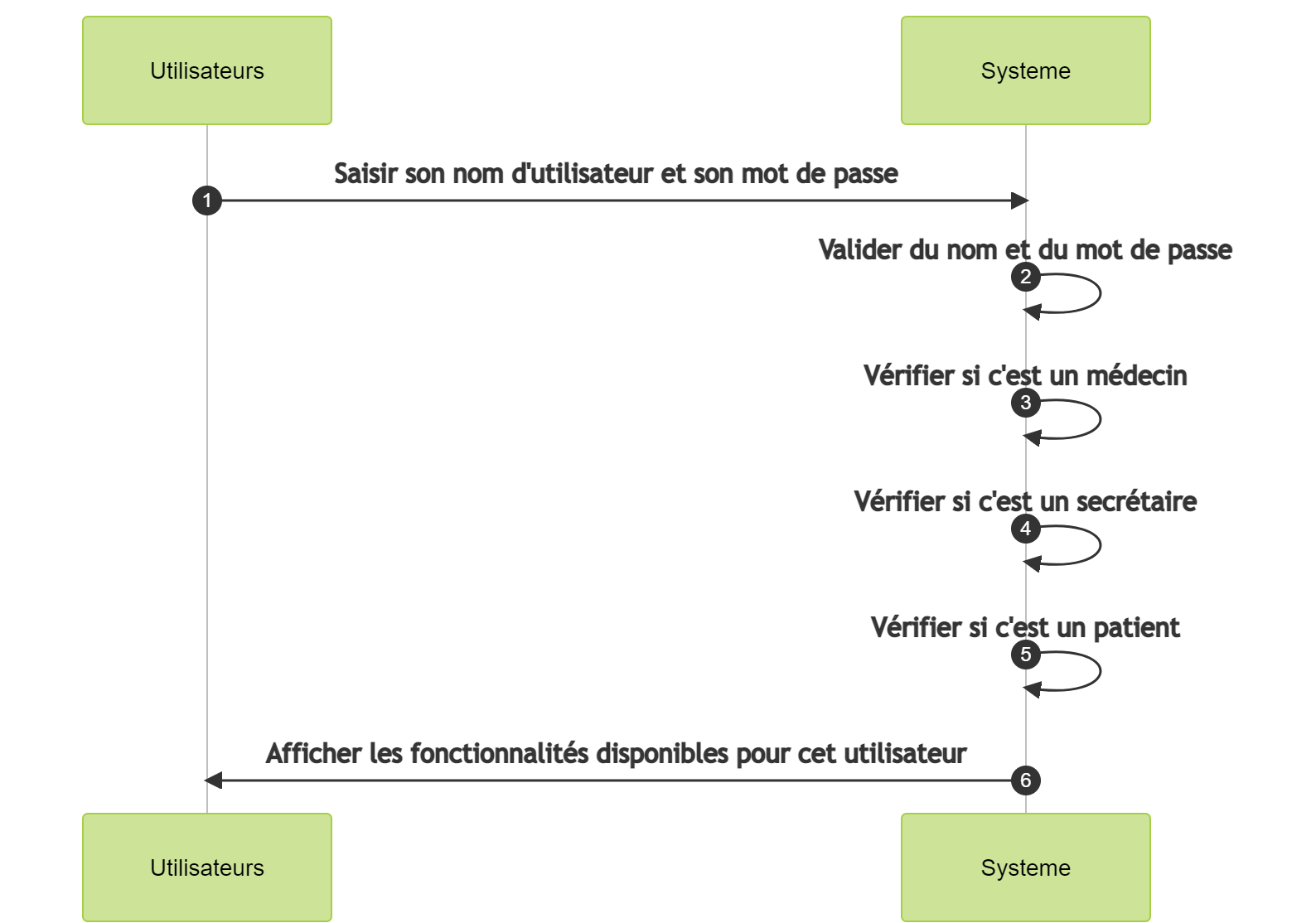


1. Note: Une note permet d'attacher diverses remarques aux éléments. Un commentaire n'a aucune force sémantique mais peut contenir des informations utiles à un modélisateur.

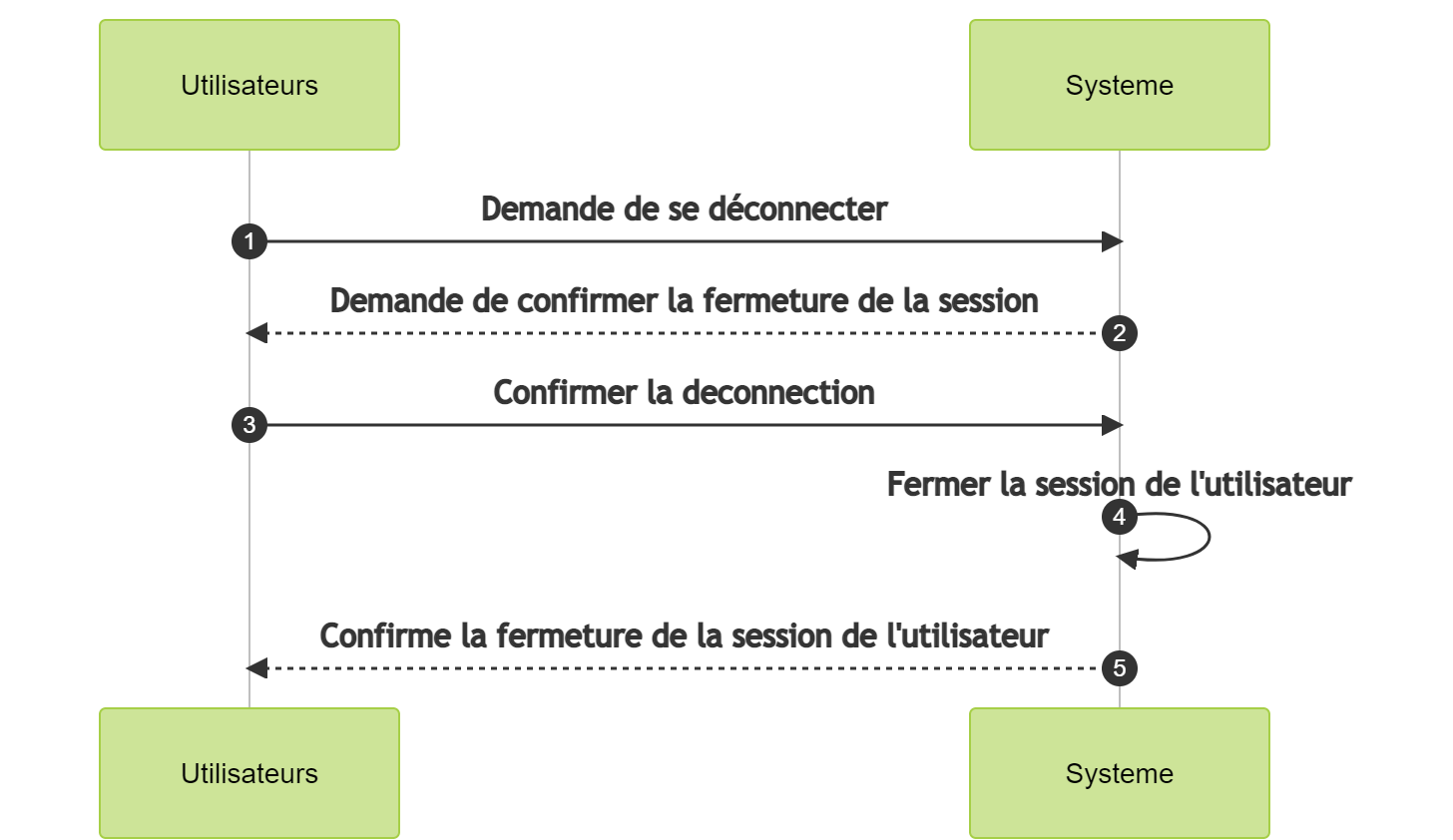


Voici la liste des diagramme de séquence des cas d’utilisations du système:

1. Diagramme de séquence pour le cas d’utilisation : S'authentifier



1. Diagramme de séquence pour le cas d’utilisation : Fermer session



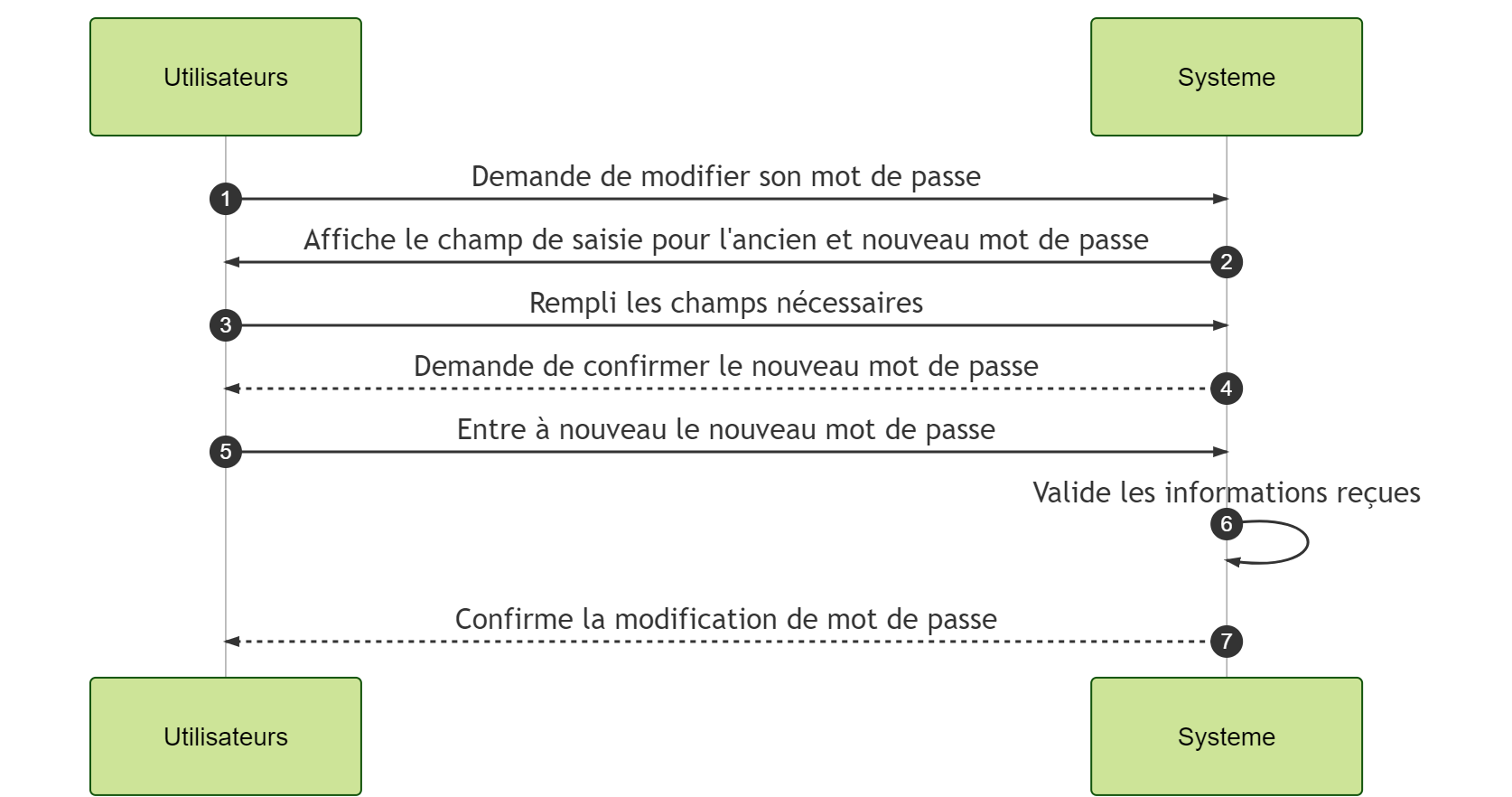
1. Diagramme de séquence pour le cas d’utilisation: Afficher les résultats d'analyses



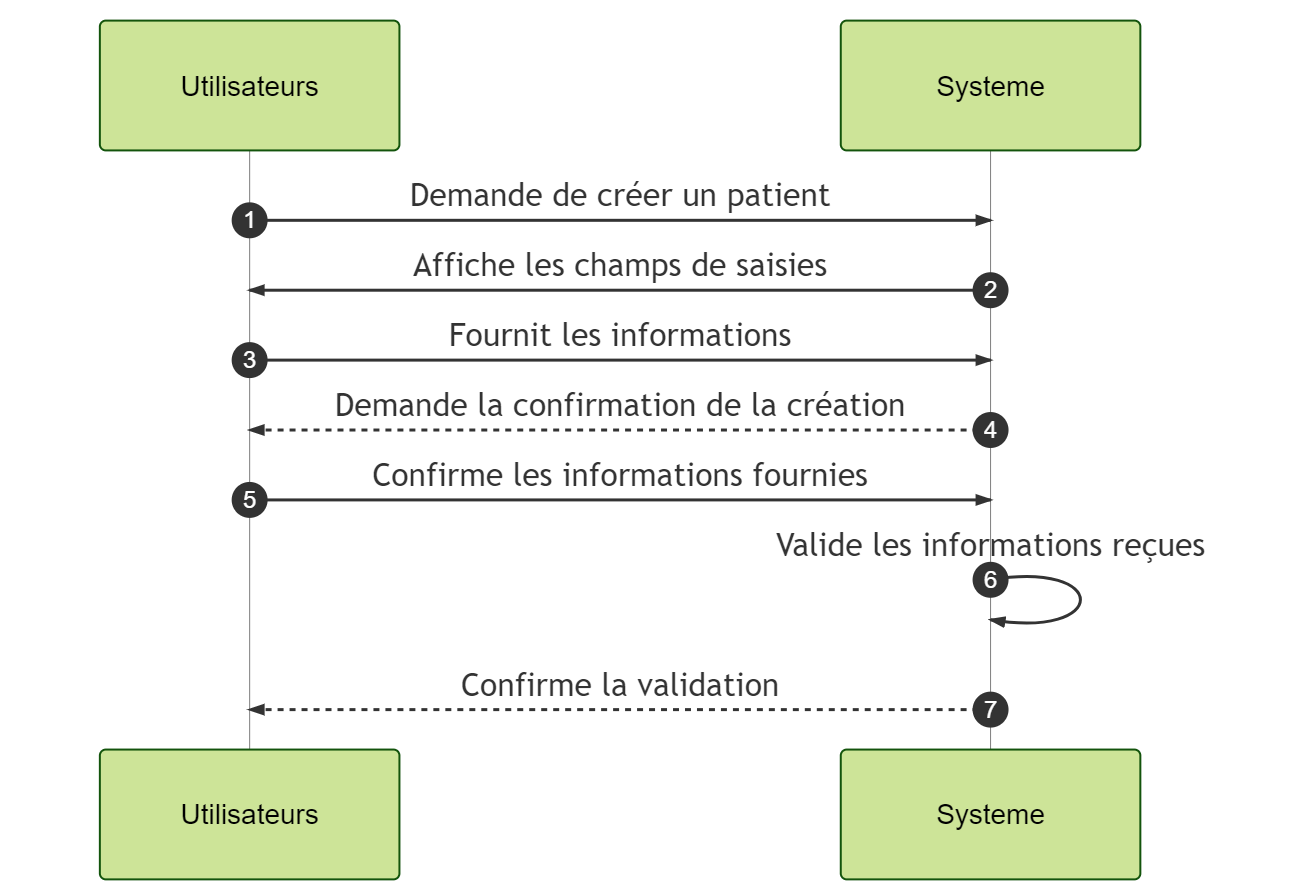
1. Diagramme de séquence pour le cas d’utilisation : Imprimer les résultats d'analyses



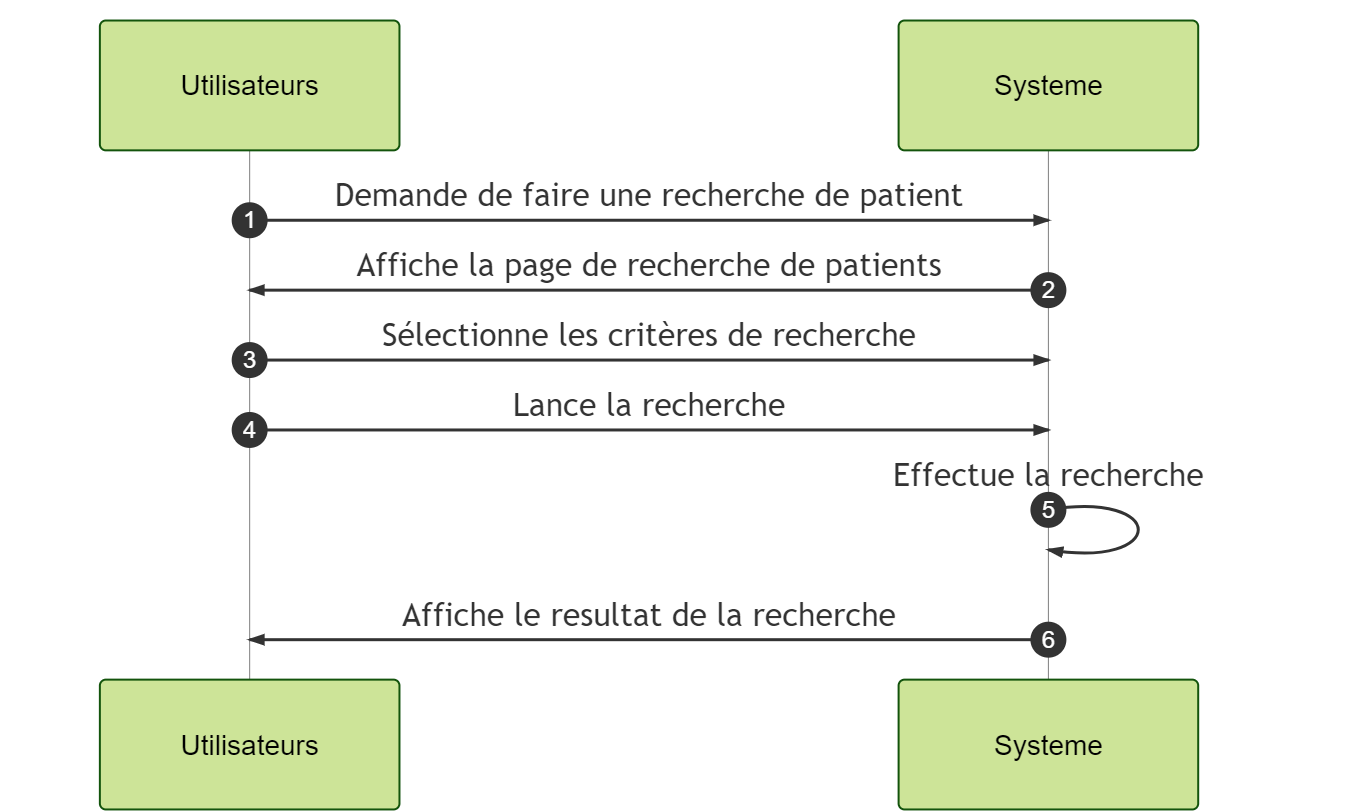
1. Diagramme de séquence pour le cas d’utilisation : Modifier le mot de passe



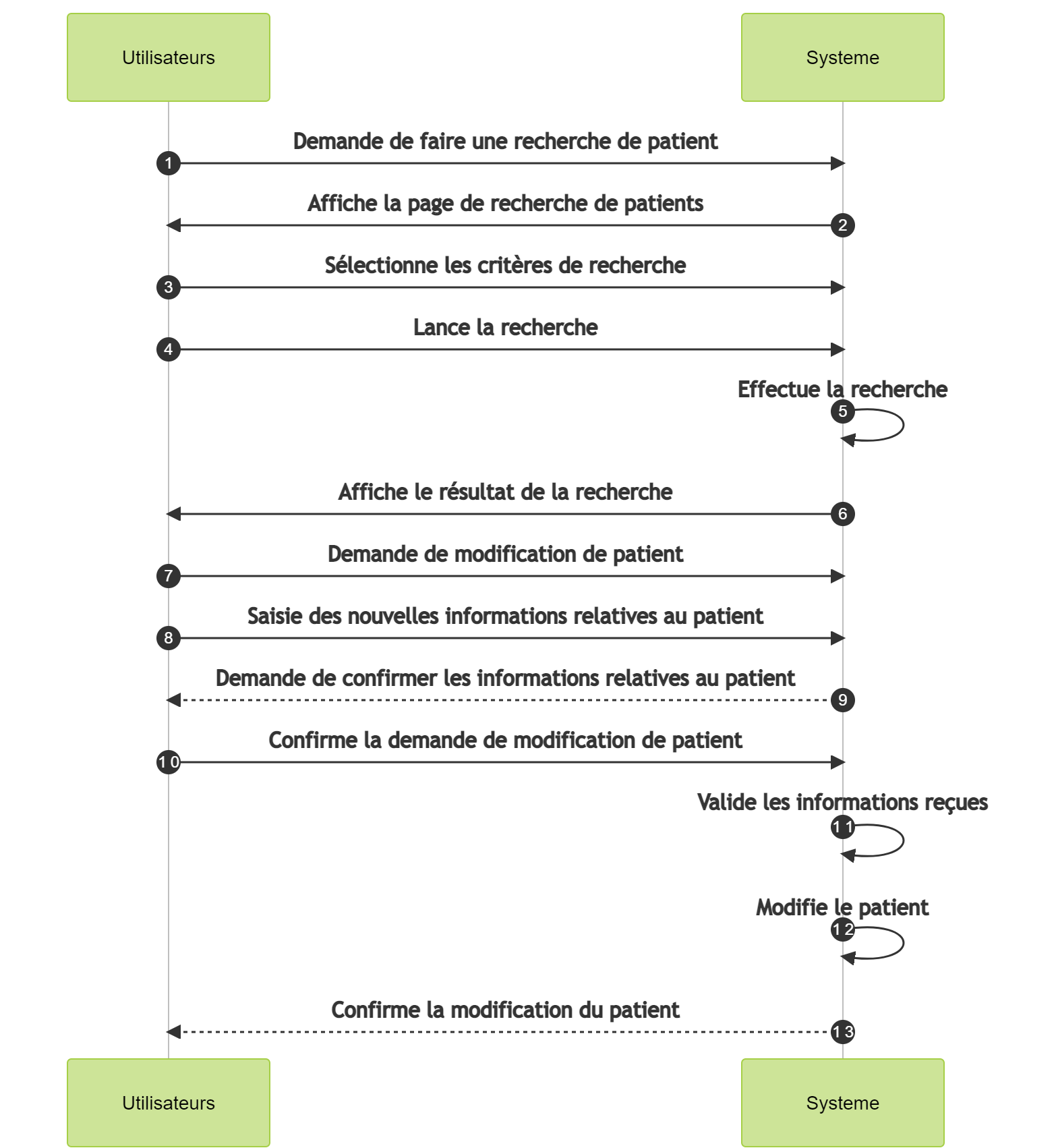
1. Diagramme de séquence pour le cas d’utilisation : Créer un patient



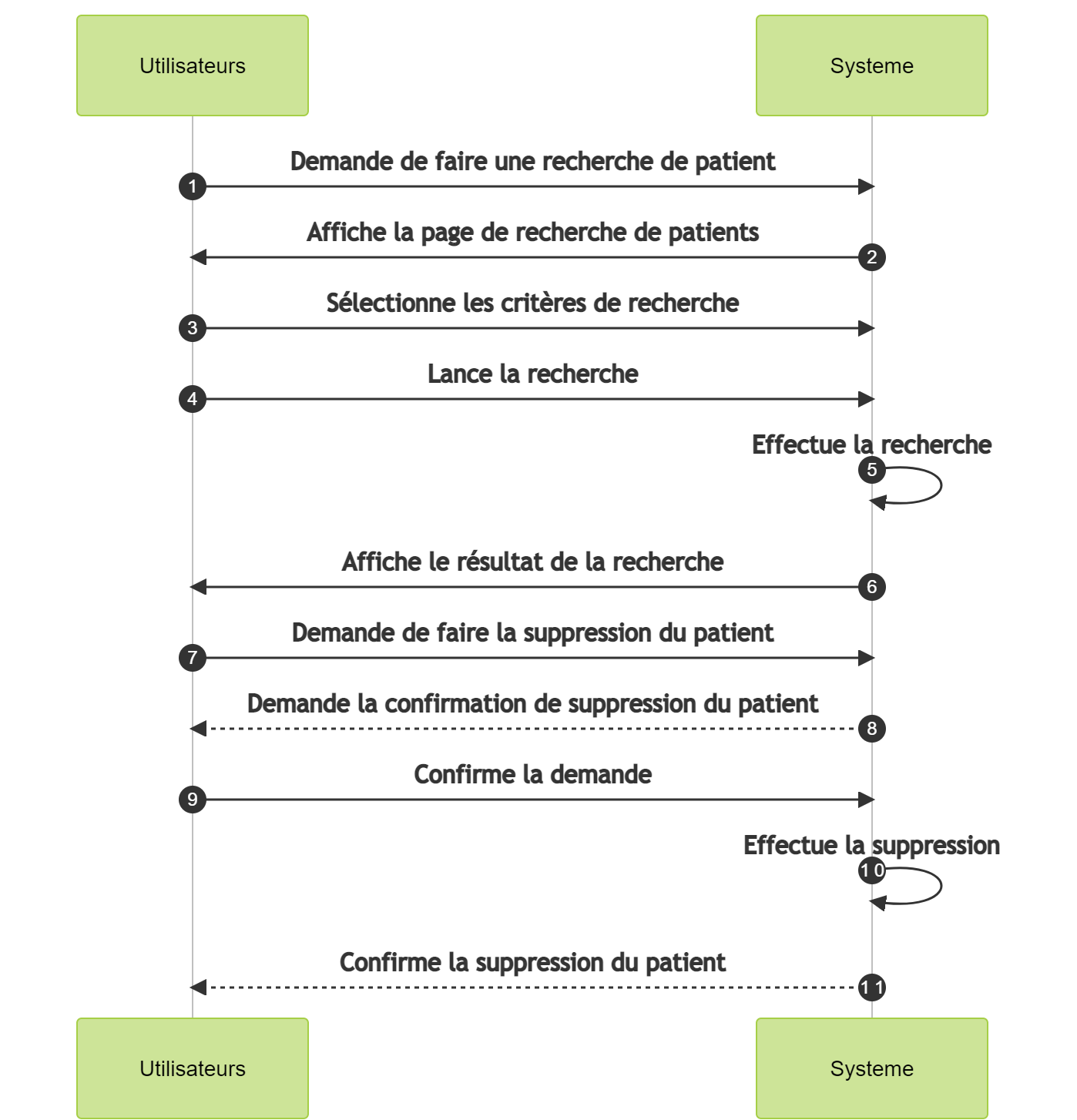
1. Diagramme de séquence pour le cas d’utilisation : Rechercher un patient



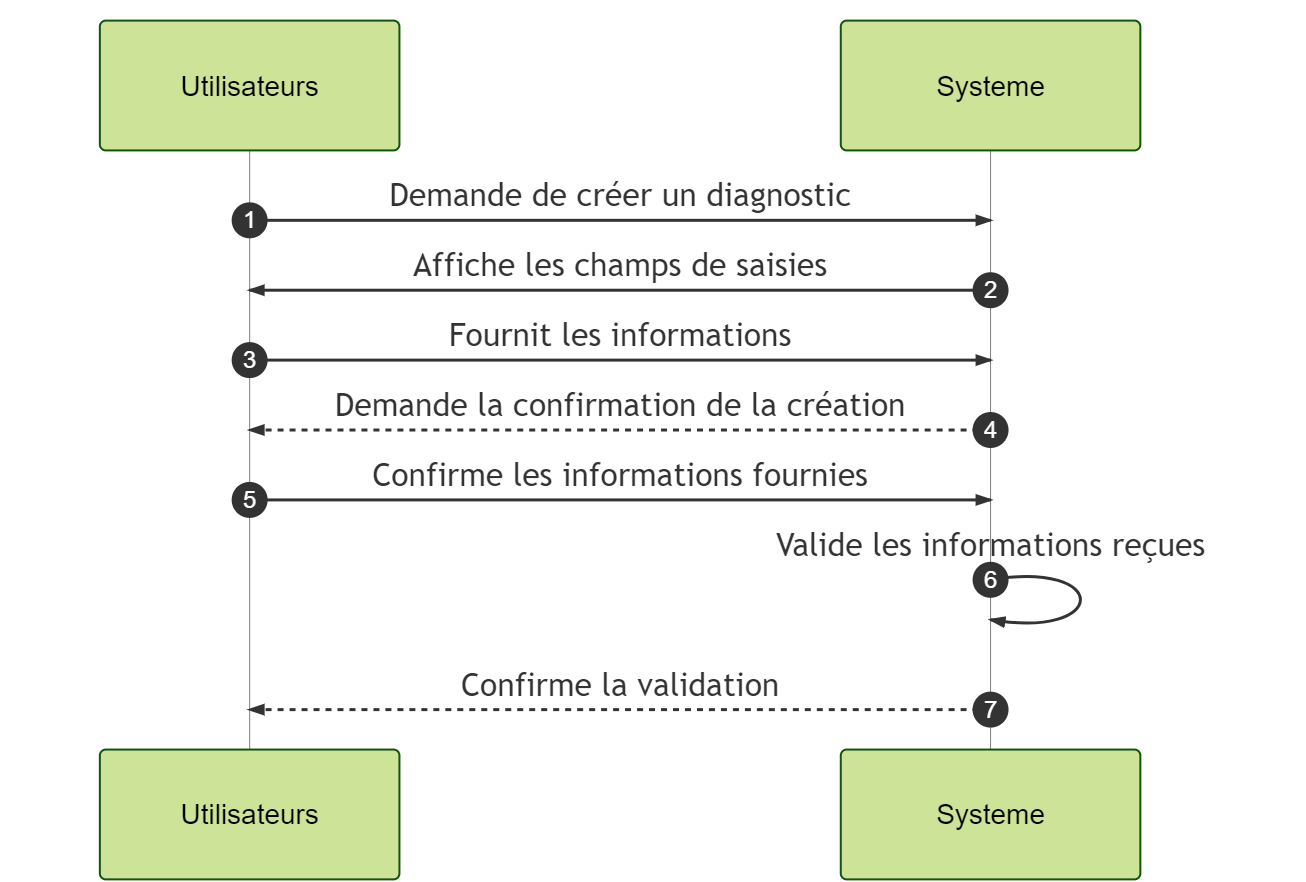
1. Diagramme de séquence pour le cas d’utilisation: Modifier un patient



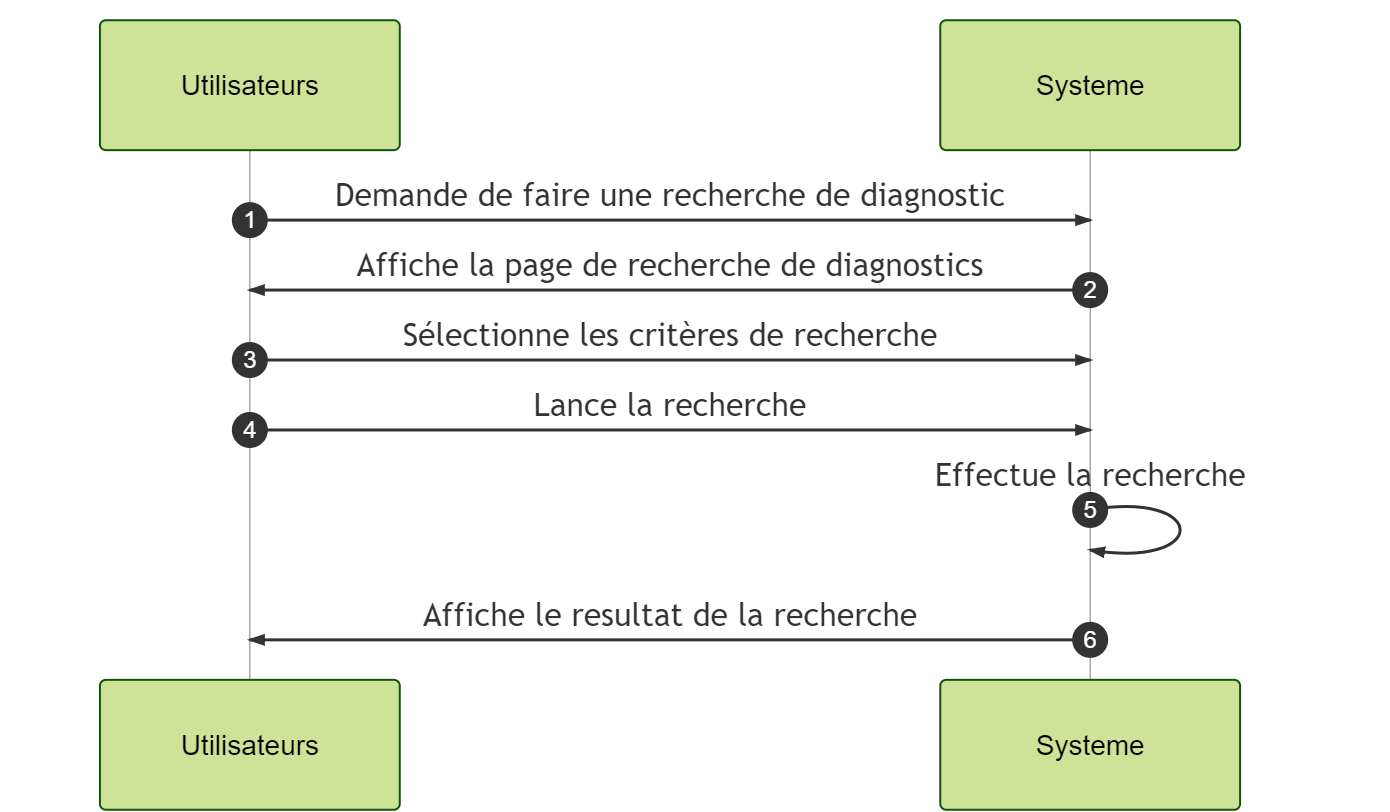
1. Diagramme de séquence pour le cas d’utilisation : Supprimer un patient



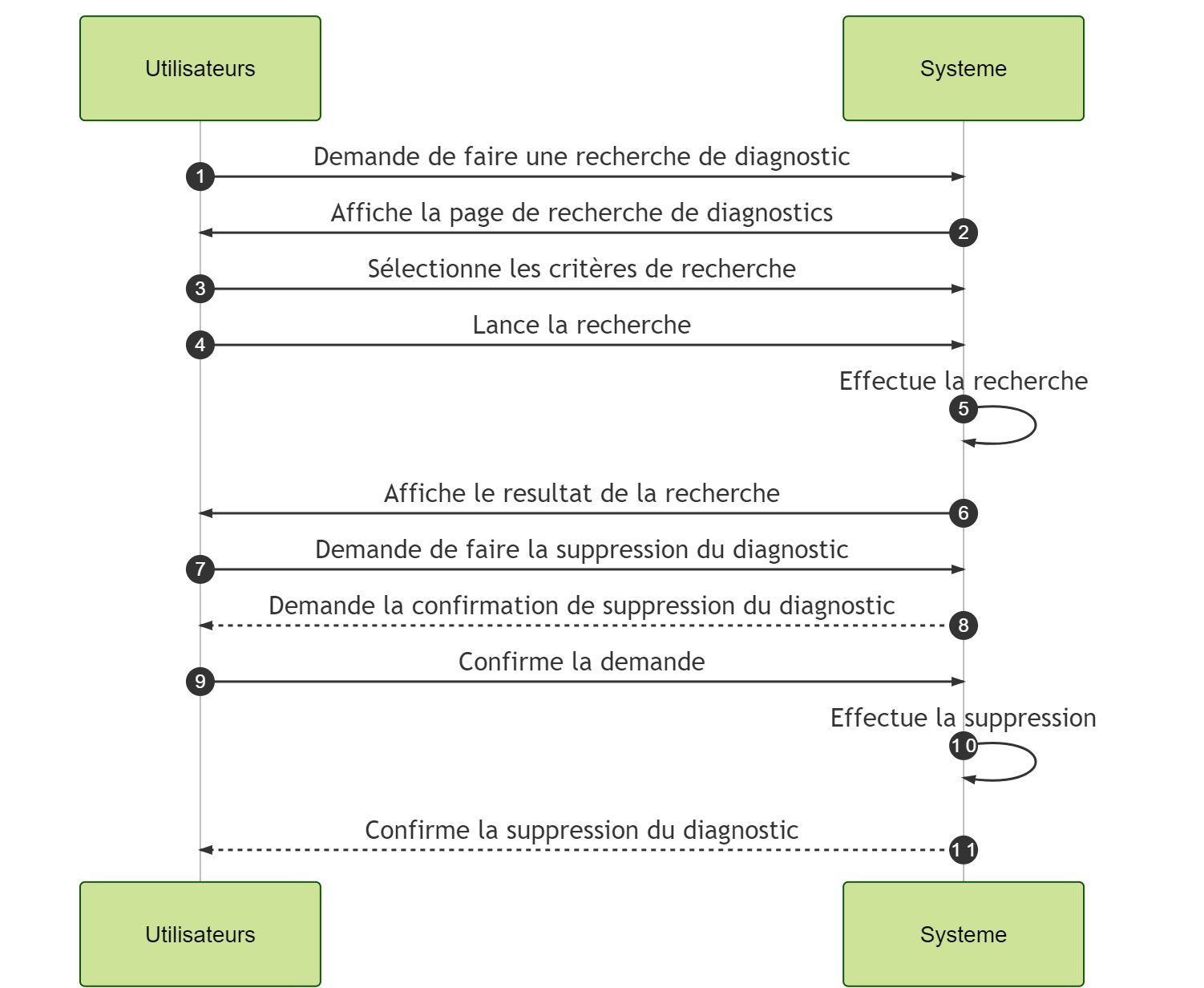
1. Diagramme de séquence pour le cas d’utilisation : Créer un diagnostic



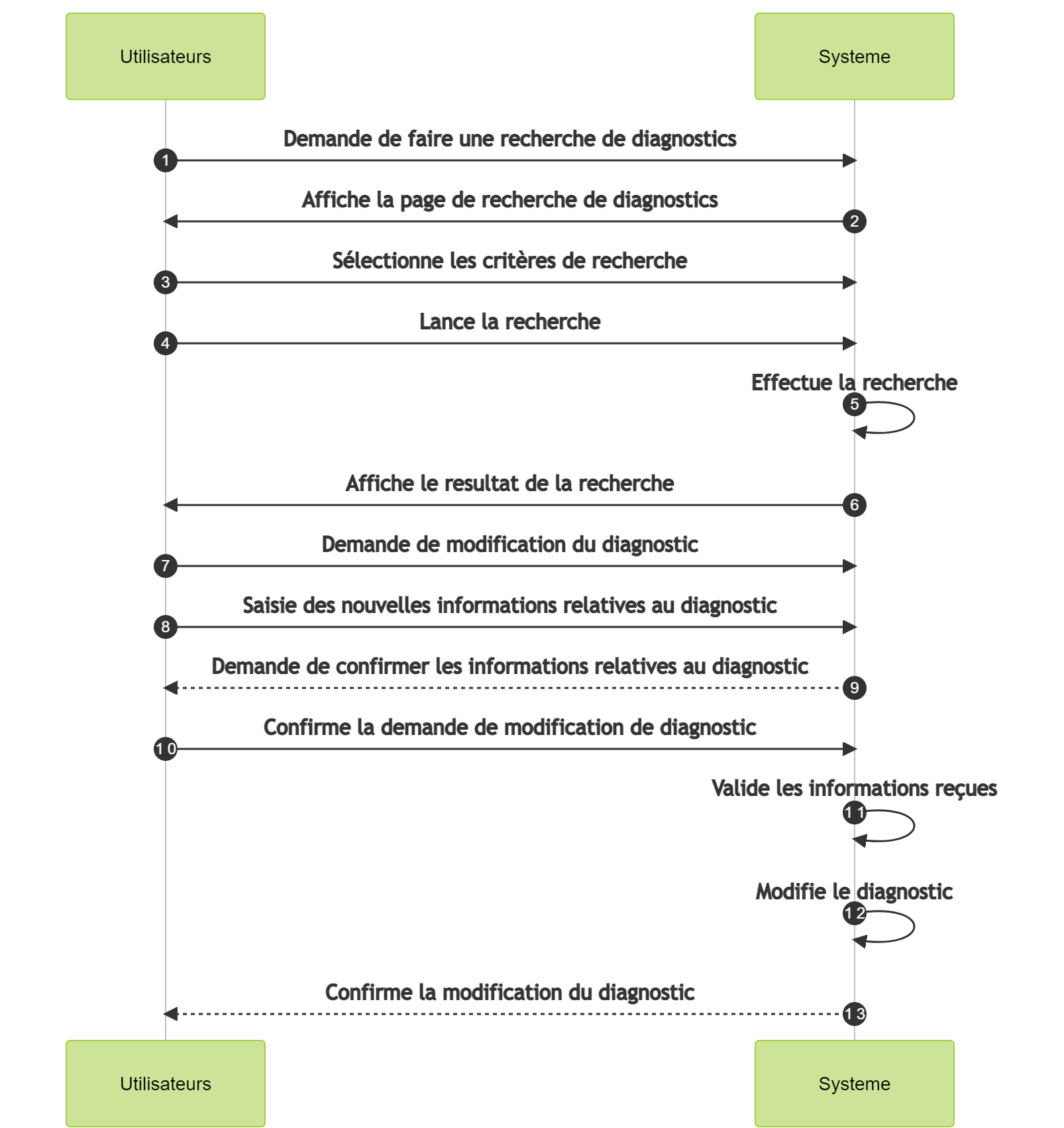
1. Diagramme de séquence pour le cas d’utilisation: Rechercher un diagnostic



1. Diagramme de séquence pour le cas d’utilisation: Supprimer un diagnostic



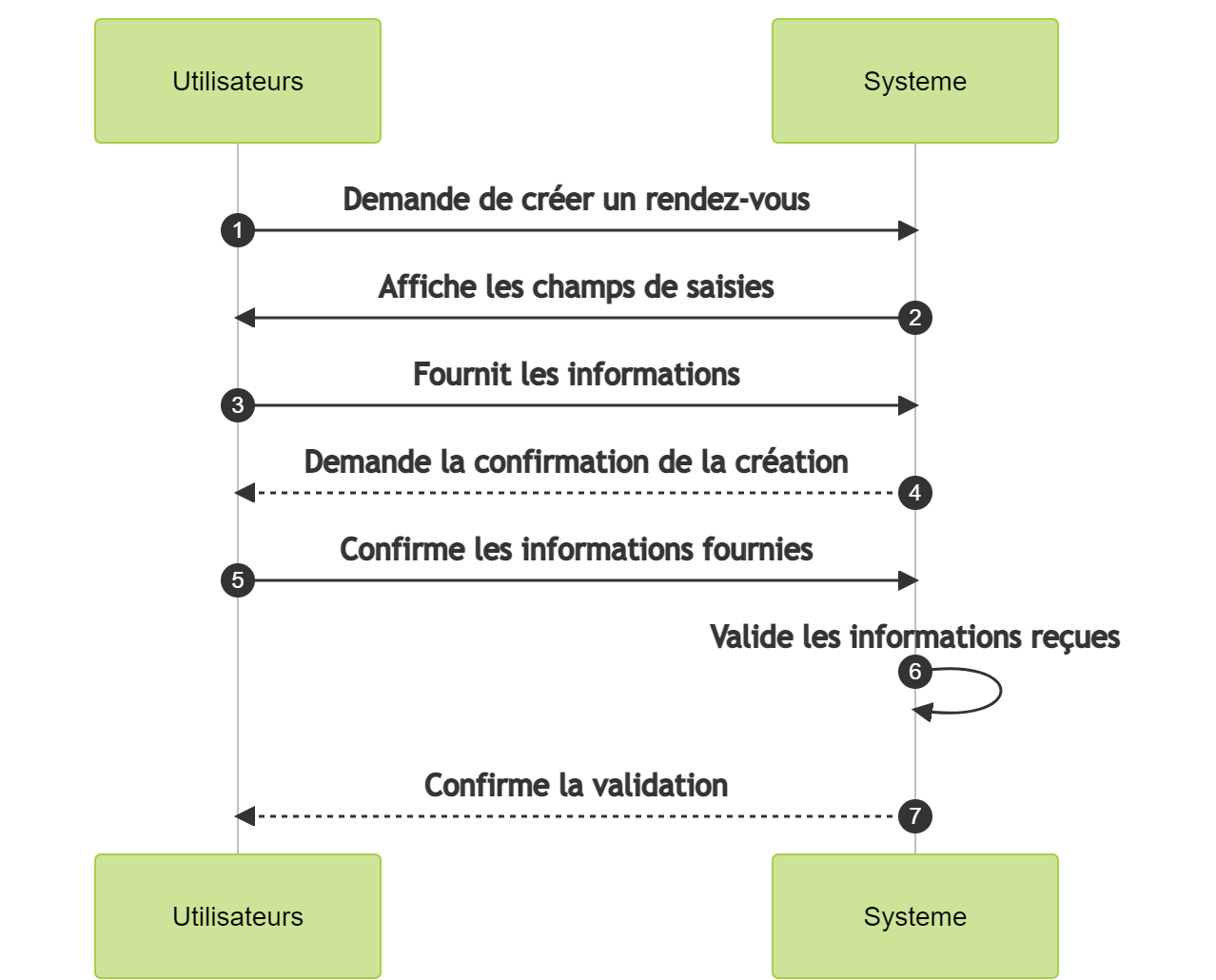
1. Diagramme de séquence pour le cas d’utilisation : Modifier un diagnostic



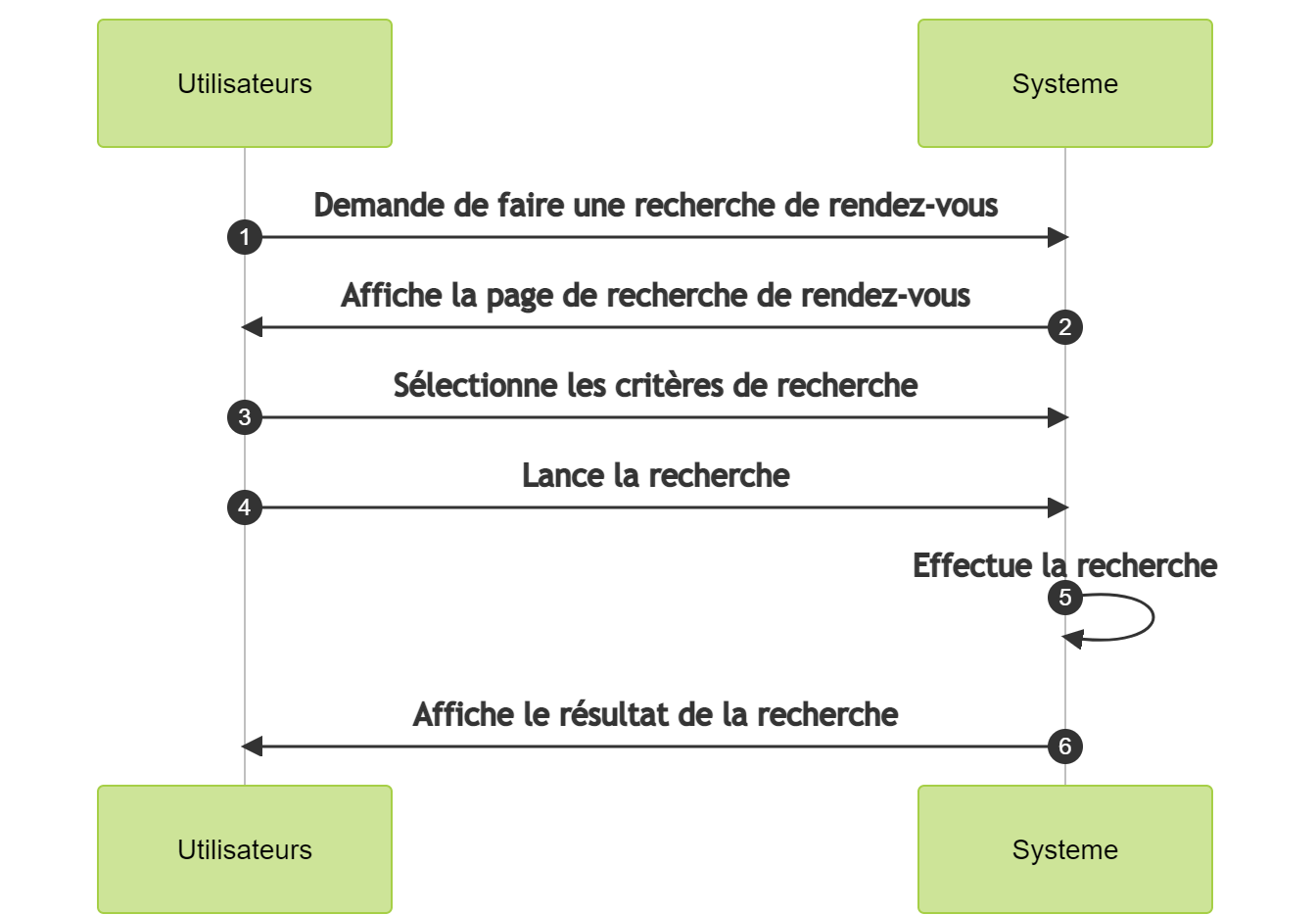
1. Diagramme de séquence pour le cas d’utilisation : Afficher les rendez-vous



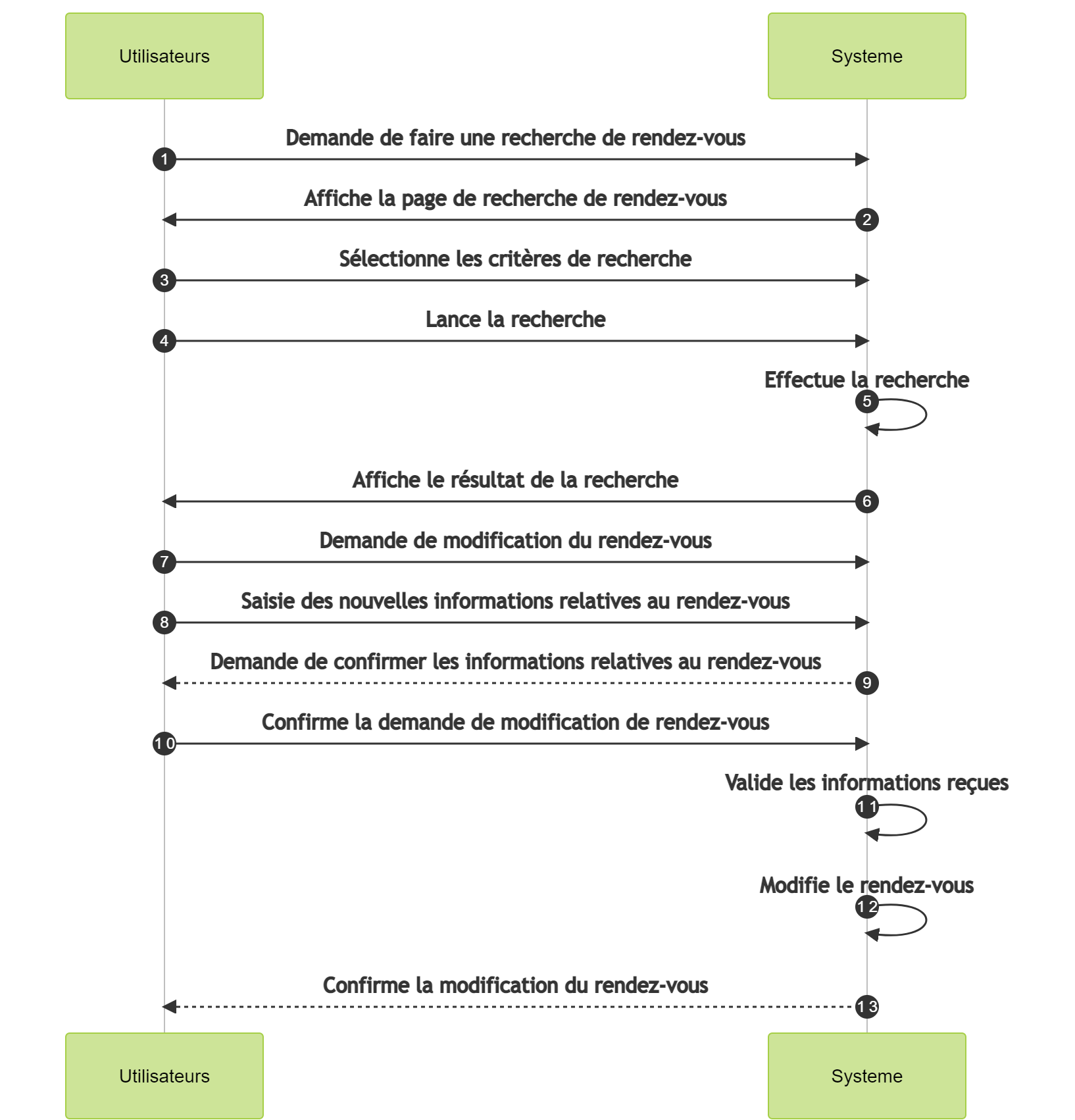
1. Diagramme de séquence pour le cas d’utilisation : Créer un rendez-vous



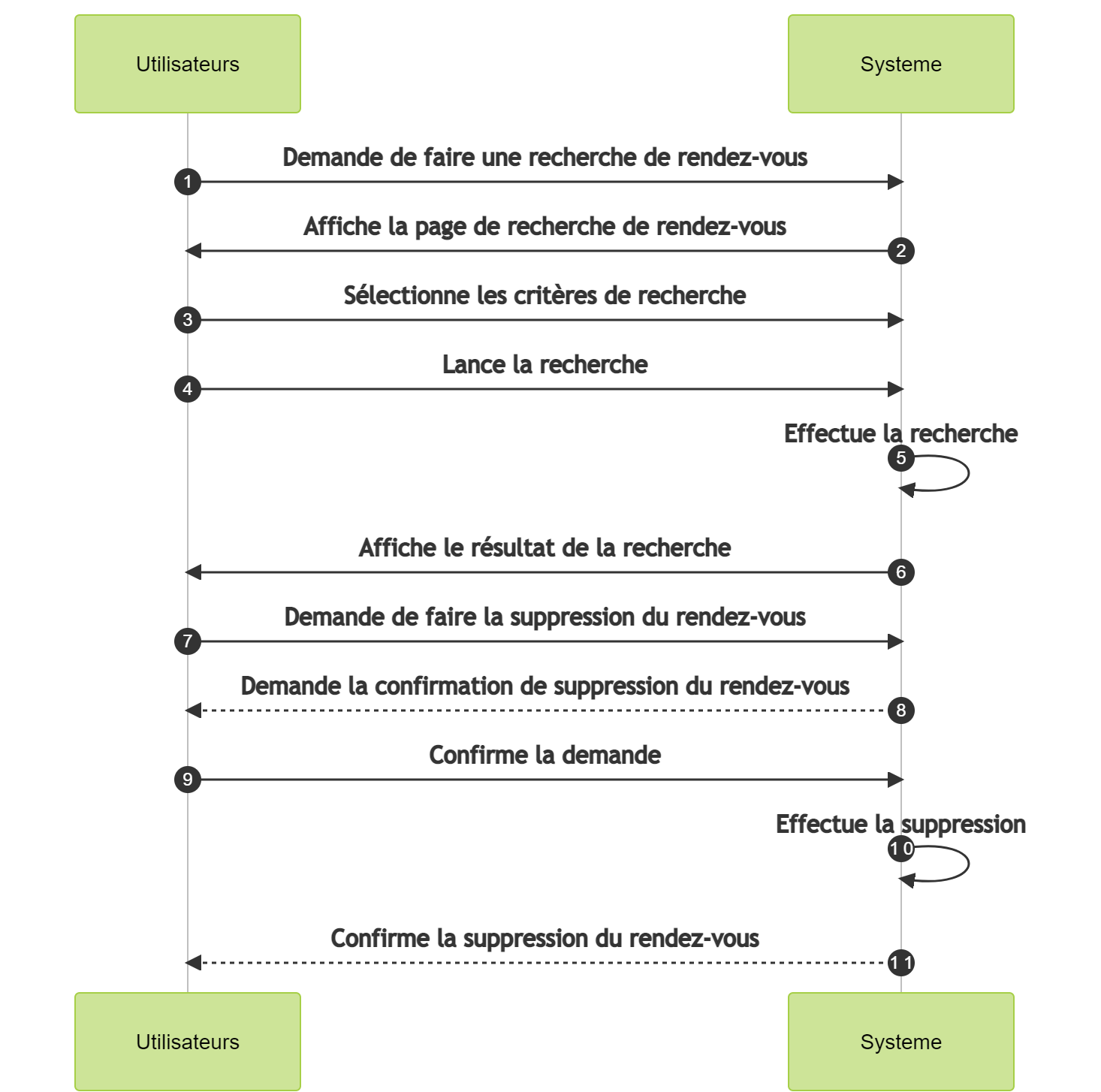
1. Diagramme de séquence pour le cas d’utilisation: Rechercher un rendez-vous



1. Diagramme de séquence pour le cas d’utilisation : Modifier un rendez-vous



1. Diagramme de séquence pour le cas d’utilisation : Supprimer un rendez-vous



### 3.3.4 - Diagramme de classes

Un diagramme de classes est un type de diagramme de structure statique qui décrit la structure d'un système en montrant les classes du système, leurs attributs, opérations (c’est à dire leurs méthodes) et relations entre les objets.

La figure suivante est un exemple de classe simple:

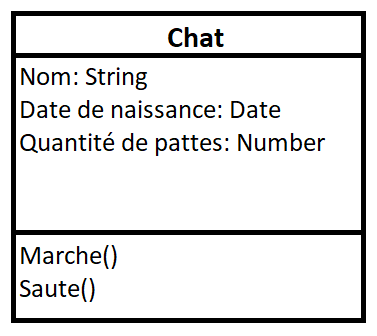


Figure : Diagramme de classes simple avec attributs et opérations

Dans l'exemple, une classe appelée « Chat » est représentée. Les classes dans les diagrammes de classes sont représentées par des cases qui sont divisées en trois :

1. La partition supérieure contient le nom de la classe.
2. La partie centrale contient les attributs de la classe.
3. La partition du bas montre les opérations possibles associées à la classe.

L'exemple montre comment une classe peut encapsuler toutes les données pertinentes d'un objet particulier d'une manière très systématique et claire. Un diagramme de classes est une collection de classes similaire à celle ci-dessus.

Les classes sont liées les unes aux autres de manière spécifique. En particulier, les relations dans les diagrammes de classes incluent différents types de connexions logiques. Voici les types de connexions logiques possibles en UML :

1. Association : Est un terme large qui englobe à peu près n'importe quelle connexion logique ou relation entre les classes.



1. Association dirigée : Fait référence à une relation directionnelle représentée par une ligne avec une pointe de flèche. La pointe de flèche représente le flux directionnel contenu dans un conteneur.



1. Association réflexive : Cela se produit lorsqu'une classe peut avoir plusieurs fonctions ou responsabilités. La notation 0 .. \* dans le diagramme signifie "zéro à plusieurs".



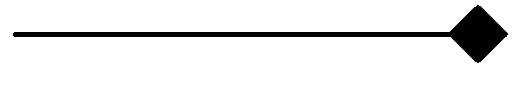
1. Multiplicité : Est l'association logique active lorsque la cardinalité d'une classe par rapport à une autre est représentée. La notation 0 .. \* dans le diagramme signifie "zéro à plusieurs".



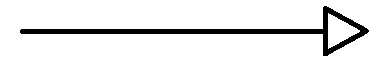
1. Agrégation : Fait référence à la formation d'une classe particulière suite à l'agrégation ou à la construction d'une classe en tant que collection. Dans l'agrégation, les classes contenues ne dépendent pas fortement du cycle de vie du conteneur. L'agrégation dans un diagramme est un une ligne de la classe parente à la classe enfant avec une forme de losange près de la classe parente.



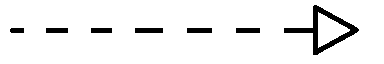
1. Composition : La relation de composition est très similaire à la relation d'agrégation. La seule différence étant son objectif principal de souligner la dépendance de la classe contenue au cycle de vie de la classe de conteneur. Autrement dit, la classe contenue sera effacée lorsque la classe conteneur est détruit.



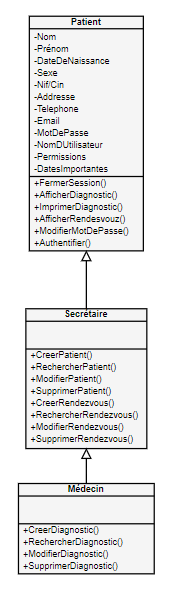
1. Héritage / généralisation : Fait référence à un type de relation dans lequel une classe associée est un enfant d'une autre car elle assume la même fonctionnalité de la classe parent. En d'autres termes, la classe enfant est un type spécifique de classe parent. L'héritage dans un diagramme UML, une ligne continue de la classe enfant à la classe parente est dessinée à l'aide d'une pointe de flèche non remplie.



1. La concrétisation : Désigne l'implémentation de la fonctionnalité définie dans une classe par une autre classe.



Voici le diagramme de classe du système :



# Partie IV

# Dossier de Conception

## 4.1 - Interface Homme-Machine

L'introduction du concept d'interface homme-machine est un moyen de concevoir des systèmes informatiques ergonomiquement efficaces, simples à utiliser et à utiliser afin d'interagir avec les utilisateurs.

Notre système se compose d’un modules majeurs: un module local et un module distribué concernant le registre de pré-étude. Ensuite, l'interaction du module distribué avec l'homme-machine, puis le module local a été proposé d'être présenté.

## 4.2 - Schémas de base de données

Les schémas de base de données définissent la structure des données. Nous présentons dans les sections qui suivent les modèles conceptuel, logique et physique de la base de données qui stocke les données l’application développée pour notre client.

## 4.2.1- Modèle conceptuel de données

Le Modèle Conceptuel de Données (MCD) permet d’établir une représentation claire des données. Il permet également de définir les dépendances fonctionnelles de ces données entre elles

Les concepts utilisés pour la formalisation de notre MCD :

– Entité (ou individu) : Objet abstrait ou concret du monde réel perçu.

- Propriété (ou rubrique) : Attribut que l’on peut associer à une entité

– Association (relation) : Relation perçue entre entités.

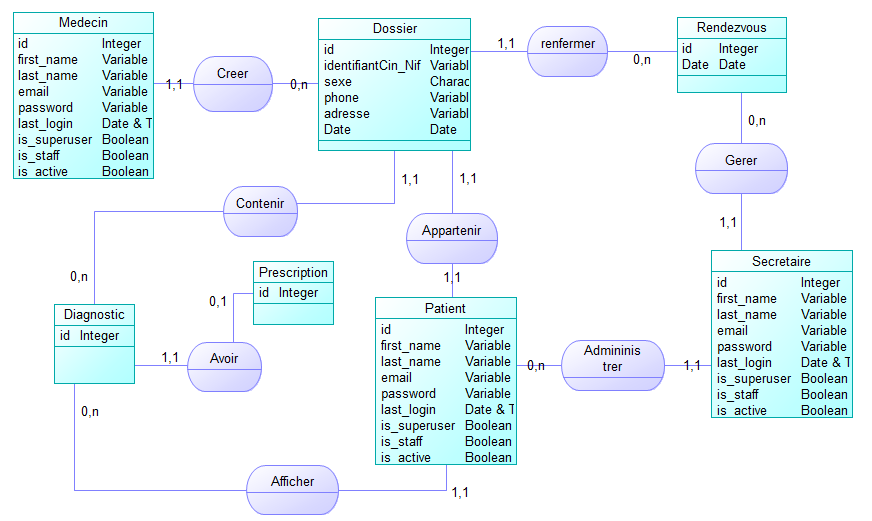
**Liste des entités du MCD**

* Médecin
* secrétaire
* patient
* diagnostic
* rendez-vous
* dossier
* prescription

**Liste des associations**

* créer
* contenir
* voir
* renfermer
* appartenir
* concerner
* gérer
* administrer

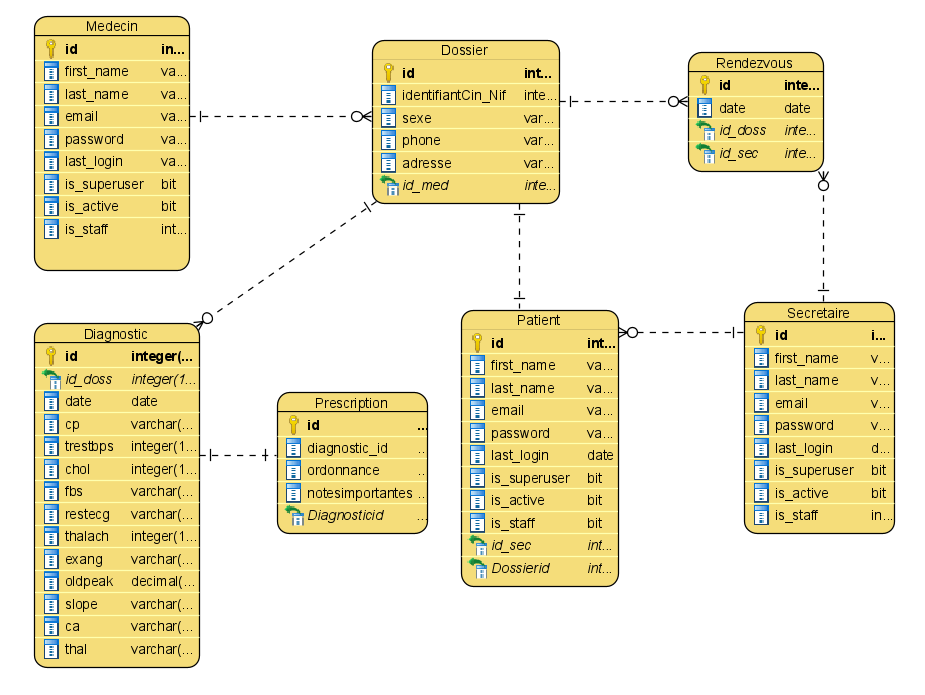
**Schéma du Modèle Conceptuel de Données**



4.2.2- Modèle logique de données

Le modèle de données découle du modèle conceptuel mais le raffine pour tenir compte des caractéristiques du type de SGBD utilisé pour la réalisation de la base de données. Le modèle logique de données spécifie un schéma pour une base de données relationnelle, dans ce schéma : les entités du modèle conceptuel deviennent des tables ou relations, les attributs deviennent les champs de chaque table et leurs propriétés, les identifiants deviennent la clé primaire des tables, les clés étrangères assurant les liaisons entre les tables et les contraintes d’intégrité portant sur ces liaisons

**Schéma du Modèle Conceptuel de Données**



4.2.3- Modèle physique de données

class Dossier(models.Model):

utilisateur = models.OneToOneField(User,

on\_delete=models.CASCADE,

primary\_key=True,

help\_text='')

dateDeNaissance = models.DateField('Date de naissance',

default=timezone.now,

null=False,

blank=False)

SEX\_VALUE = (

('0', 'M'),

('1', 'F'),

)

sex = models.CharField(

'Sexe',

max\_length=1,

choices=SEX\_VALUE,

blank=True,

default='0',

help\_text='(0 = male; 1 = female)',

)

phone = models.CharField(

'Téléphone',

max\_length=200,

default='',

help\_text='xxx xx xx xxxx'

)

addresse = models.CharField(max\_length=200,

null=True,

help\_text='Numero, Rue')

identifiantCin\_Nif = models.CharField('Cin ou Nif',

max\_length=200,

null=True,

help\_text='')

class Diagnostic(models.Model):

dossier = models.ForeignKey('Dossier',

on\_delete=models.CASCADE,

)

date = models.DateField('Date du diagnostic',

default=timezone.now,

null=False,

blank=False)

CP\_VALUE = (

('0', 'Valeur: 0'),

('1', 'Valeur: 1'),

('2', 'Valeur: 2'),

('3', 'Valeur: 3'),

)

cp = models.CharField(

'Chest pain type',

max\_length=13,

choices=CP\_VALUE,

default='1',

help\_text='(4 values)',

)

trestbps = models.IntegerField(

'Resting blood pressure',

default=130,

help\_text=

'Resting blood pressure (in mm Hg on admission to the hospital)')

chol = models.IntegerField('Serum cholestoral',

default=236,

help\_text='mg/dl')

FBS\_VALUE = (

('1', '> 120 mg/dl'),

('0', '<= 120 mg/dl'),

)

fbs = models.CharField(

'Fasting blood sugar',

default='0',

max\_length=13,

choices=FBS\_VALUE,

help\_text='(fasting blood sugar > 120 mg/dl) (1 = true; 0 = false)')

RESTECG\_VALUE = (

('0', 'Valeur: 0'),

('1', 'Valeur: 1'),

('2', 'Valeur: 2'),

)

restecg = models.CharField('Resting electrocardiographic results',

default='0',

choices=RESTECG\_VALUE,

max\_length=13,

help\_text=' (values 0,1,2)')

thalach = models.IntegerField('Maximum heart rate achieved',

default=174,

help\_text='')

EXANG\_VALUE = (

('1', 'Yes'),

('0', 'NO'),

)

exang = models.CharField(

'Exercise induced angina',

default='0',

choices=EXANG\_VALUE,

max\_length=13,

help\_text='exercise induced angina (1 = yes; 0 = no)')

oldpeak = models.DecimalField(

'Oldpeak',

max\_digits=2,

decimal\_places=1,

default=0.0,

help\_text='Oldpeak = ST depression induced by exercise relative to rest'

)

SLOPE\_VALUE = (

('0', 'Valeur: 0'),

('1', 'Valeur: 1'),

('2', 'Valeur: 2'),

)

slope = models.CharField(

'Slope',

default='1',

choices=SLOPE\_VALUE,

max\_length=13,

help\_text='The slope of the peak exercise ST segment')

CA\_VALUE = (

('0', 'Valeur: 0'),

('1', 'Valeur: 1'),

('2', 'Valeur: 2'),

('3', 'Valeur: 3'),

)

ca = models.CharField(

'Number of major vessels',

default='1',

choices=CA\_VALUE,

max\_length=13,

help\_text='Number of major vessels (0-3) colored by flourosopy')

THAL\_VALUE = (

('1', 'Valeur: 1'),

('2', 'Valeur: 2'),

('3', 'Valeur: 3'),

)

thal = models.CharField(

'Thal',

default='2',

choices=THAL\_VALUE,

max\_length=13,

help\_text='Thal: 3 = normal; 6 = fixed defect; 7 = reversable defect')

class Prescription(models.Model):

diagnostic = models.OneToOneField('Diagnostic',

on\_delete=models.CASCADE,

primary\_key=True,

help\_text='')

ordonnance = models.CharField(max\_length=500, help\_text='')

notesImportantes = models.CharField('Notes importantes',max\_length=200, help\_text='')

class RendezVous(models.Model):

dossier = models.ForeignKey('Dossier',

on\_delete=models.CASCADE,

)

date = models.DateField(null=True, blank=True)

## 4.3 - Dictionnaire de données

| **Champs** | **Description** | **Type de données** | **Contraintes** |
| --- | --- | --- | --- |
| id | identifiant utilisateur | entier | clé primaire |
| first\_name | prenom de l’utilisateur | varchar(150) | non nulle |
| last\_name | nom de l’utilisateur | varchar(150) | non nulle |
| username | nom d’utilisateur | varchar(150) | non nulle |
| email | email de l’utilisateur | varchar(150) | non nulle |
| password | mot de passe de l’utilisateur | varchar(150) | non nulle |
| last\_login | date de dernière login | datetime |  |
| is\_superuser |  | booléen | non nulle |
| is\_active |  | booléen | non nulle |
| is\_staff |  | booléen | non nulle |
| user\_id | identifiant du patient | entier | non nulle |
| dateDeNaissance | date de naissance de l’utilisateur | dateheure | nulle |
| sexe | sexe du patient | varchar(1) | non nulle |
| phone | téléphone du patient | varchar(200) | non nulle |
| adresse | adresse du patient | varchar(200) | non nulle |
| identifiantCin\_Nif | cin ou nif du patient | varchar(200) | non nulle |
| dagnostic\_id | identifiant du diagnostic | entier | clé primaire |
| ordonnances | ordonnance du médecin pour le diagnostic |  | nulle |
| notesImportantes | notes du médecin au patient |  | nulle |
| id | identifiant d’un rendez-vous | entier | clé primaire |
| dossier\_id | identifiant du dossier patient | entier | clé étrangère |
| date | date du rendez-vous | dateheure | non nulle |
| id | identifiant d’un diagnostic | entier | clé primaire |
| date | date du diagnostic | dateheure | non nulle |
| cp | La douleur thoracique ressentie (Valeur 1 : angine typique, Valeur 2 : angine atypique, Valeur 3 : douleur non angineuse, Valeur 4 : asymptomatique) | varchar(13) | non nulle |
| trestbps | La pression artérielle au repos de la personne (mm Hg à l'admission à l'hôpital) | entier | non nulle |
| chol | La mesure du cholestérol de la personne en mg/dl | entier | non nulle |
| fbs | La glycémie à jeun de la personne (> 120 mg/dl, 1 = vrai ; 0 = faux) | varchar(13) | non nulle |
| restecg | Mesure électrocardiographique au repos (0 = normale, 1 = présentant une anomalie de l'onde ST-T, 2 = présentant une hypertrophie ventriculaire gauche probable ou certaine selon les critères d'Estes) | varchar(13) | non nulle |
| thalach | (Maximum heart rate achieved) Fréquence cardiaque maximale atteinte | entier | non nulle |
| exang | Angine induite par l'exercice (1 = oui ; 0 = non) | varchar(13) | non nulle |
| oldpeak | Dépression ST induite par l'exercice par rapport au repos ("ST" fait référence aux positions sur le tracé ECG) | decimal(2, 1) | non nulle |
| slope | la pente du segment ST du pic d'exercice (valeur 1 : pente ascendante, valeur 2 : pente plate, valeur 3 : pente descendante) | varchar(13) | non nulle |
| ca | Le nombre de vaisseaux principaux (0-3) | varchar(13) | non nulle |
| thal | Un trouble sanguin appelé thalassémie (3 = normal ; 6 = anomalie fixe ; 7 = anomalie réversible). | varchar(13) | non nulle |

## 

## 4.4 - Architecture du système

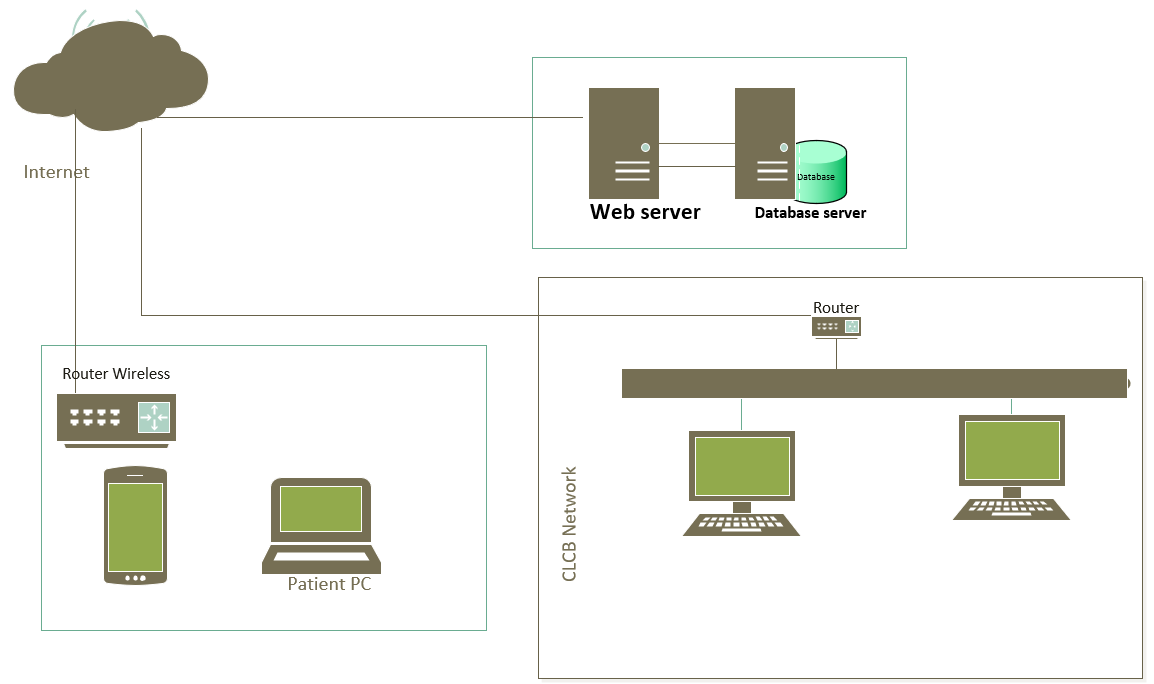
L’architecture des systèmes d’information correspond à l’installation d’un nombre d’équipements nécessaire et suffisant pour assurer la bonne circulation des informations et répondre aux fonctionnalités demandées. Elle consiste également à intégrer, lors de la conception, une certaine marge d’évolution stratégique pour accompagner la croissance de l’entreprise.

### 4.4.1 - Architecture matérielle

L'architecturematérielle est une vue tournée sur le choix et l'organisation des différents composants électroniques d'un appareil informatique.

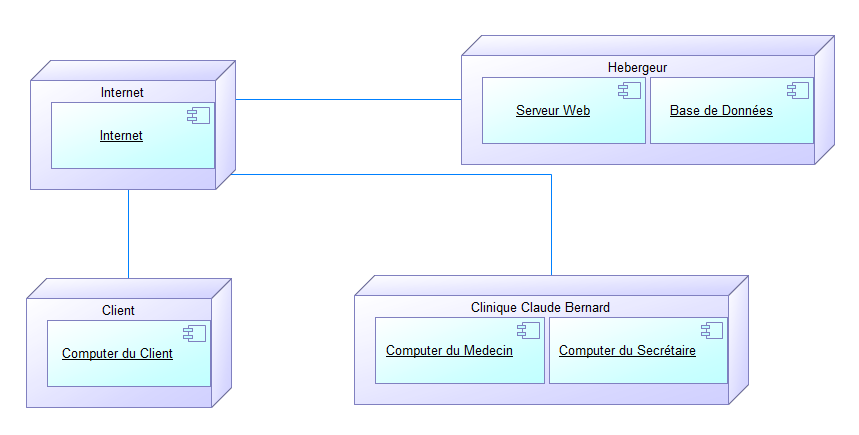
### 4.4.2 - Architecture physique

L’architecture physique décrit l'ensemble des composants matériels supportant l'application. Ces composants peuvent être des calculateurs, des postes de travail, des équipements de stockage, des équipements de sauvegarde, des équipements réseaux. Nous présentons dans l’image ci-dessous le schéma physique de l’architecture tenant en compte l’architecture physique supportant l’application développée pour notre client.



### 4.4.3 - Le diagramme de déploiement

Le diagramme de déploiement se rapproche de la réalité physique puisqu'il identifie les éléments matériels (PC, Modem, Serveur, etc.), leur disposition physique (connexions) et la disposition des exécutables (représentés par des composants) sur ces éléments matériels. Chaque ressource étant matérialisée par un nœud.



### 4.4.4 - Architecture logicielle

L'architecturelogicielle est une vue tournée sur l'organisation interne et le découpage en couches et modules du ou des logiciels du système informatique. Les responsabilités de chaque module et la nature et la structure des relations entre modules.

# Partie V

# Développement

## 5.1 - Méthode de développement

La méthodologie sur laquelle reposent les analyses, la conception, la programmation et les tests de la CLCB doit être établie, et nous avons donc choisi la méthode dite Agile.

Agile est une méthodologie par laquelle une équipe peut gérer un projet dans sa répartition en plusieurs étapes et sa communication constante, son développement et son itération à chaque étape. La méthodologie Agile commence avec les clients décrivant comment le produit final sera utilisé et quel problème il résoudra, cela clarifie les attentes du client vis-à-vis de l'équipe de projet.

Agile est une méthodologie largement utilisée en raison de son adaptabilité à la réalisation de tâches difficiles. Une fois le travail commencé, les équipes passent par un processus de planification, d'exécution et d'évaluation, qui pourrait si nécessaire modifier le produit final, pour mieux répondre aux besoins du client. Une collaboration continue est essentielle, à la fois entre les membres de l'équipe et avec les parties prenantes du projet, pour prendre des décisions éclairées, car cette méthodologie met en évidence le travail d'équipe, la polyvalence, les progrès continus et les performances positives.

En ce qui concerne le rôle de cette méthodologie, Agile commence souvent par identifier les utilisateurs et de documenter un plan stratégique sur une variété de défis, de possibilités et de valeurs. Afin de concrétiser cette idée, le Chef du projet capture et opère avec une équipe multidisciplinaire.

Les utilisateurs sont définis souvent avec des personas d'utilisateurs pour illustrer différents rôles dans un flux de travail pris en charge par le logiciel ou différents types de besoins et de comportements des clients.

Le processus de développement agile lui-même commence avec quelqu'un qui, y compris les parties prenantes internes, doit être la voix du client. Cette personne distille toutes les pensées, idées et suggestions pour créer une vision du produit. Ces visions du produit sont pour la plupart courtes et concises mais montrent néanmoins qui est le client, les principes discutés et un plan pour les traiter.

Cette personne est nommée le propriétaire du produit ou Product Owner, sa responsabilité est de définir cette vision puis de travailler avec une équipe de développement pour en faire une réalité. Pour collaborer avec l'équipe de développement, le Product Owner décompose la perspective du produit en une série de descriptions simples d'attentes, appelées user stories, qui illustrent plus en profondeur qui est l'utilisateur final, quel défi est résolu pour lui, pourquoi la résolution est importante pour lui. et quelles conditions et critères d'acceptation décrivent la solution. Ces user stories sont hiérarchisées par le product owner, revues par l'équipe pour s'assurer qu'ils ont une compréhension commune de ce qui leur est demandé.

En agile, les responsabilités de l’équipe de développement et de ses membres diffèrent de celles du développement logiciel traditionnel. Les équipes sont multidisciplinaires, composées d'un groupe diversifié de personnes ayant les compétences nécessaires pour faire le travail. Parce que l'accent est mis sur la livraison de logiciels fonctionnels, l'équipe doit compléter des applications fonctionnelles de bout en bout. Ainsi, la base de données, la logique métier et l'interface utilisateur d'une partie de l'application sont développées, puis la présentation de prototypes. Pour ce faire, les membres de l'équipe doivent collaborer.

Ils se réunissent fréquemment pour s'assurer que tout le monde est aligné sur ce qu'ils construisent, sur qui fait quoi et sur la manière exacte dont le logiciel est développé. En plus des développeurs, les équipes de développement logiciel peuvent inclure des ingénieurs d'assurance qualité, d'autres ingénieurs (comme pour les bases de données et les systèmes back-end), des concepteurs et des analystes, selon le type de projet logiciel.

## 5.2-Planification

La planification méthodologie Agile est construite par six « livrables » clés. C'est-à-dire que ces livrables nous aident à tracer les progrès et à construire le produit, afin que le projet puisse être transparent et observable.

Les livrables sont listés ainsi :

1. Énoncé de vision du produit :

Un résumé qui articule les objectifs du produit.

1. Feuille de route du produit :

La vue d'ensemble des exigences nécessaires pour réaliser la vision du produit.

1. Backlog de produit :

Trié par priorité, il s'agit de la liste complète de ce qu'il faut faire pour mener à bien le projet.

1. Plan de sortie :

Un calendrier pour la sortie d'un produit opérationnel.

1. Backlog de sprint :

Les exigences, les objectifs et les tâches liées au sprint actuel.

1. Incrément :

La fonctionnalité du produit en état de marche qui est présentée aux parties prenantes à la fin du sprint et qui pourrait potentiellement être donnée au client.

Notre équipe a pour objectif de fournir le produit en respectant la méthodologie énoncée, de ce fait des rôles ont été répartis à chacun de nous. Le tableau ci-dessous en affiche la répartition :

| Jean Maurice Dumas | Product Owner (Directeur du projet) |
| --- | --- |
| Nick Anderson Azemar | Développeur d’IA |
| Cyprien Jean Maxo | Webmaster, Database Manager |
| Cadet Wadson | Tester, Bug Tracker |

## 5.3 - Outils de développement

CLCB a été développée par l'utilisation d’un ensemble de technologies qui sont énoncées dans cette liste ci-dessous :

1. Pycharm:

Pycharm est un environnement de développement intégré pour Python.

1. Python:

Python est un langage de programmation interprété dynamiquement interactif, orienté objet.

1. Django:

Django est un framework de haut niveau pour Python qui facilite le développement de sites Web rapides et stables.

1. WampServer:

Wampserver est un environnement de développement de sites web.

1. Html:

Html est un langage de développement web utilisé pour le fond des pages web.

1. CSS:

CSS est un langage de développement web utilisé pour la forme des pages web.

1. MySql:

MySql est un langage de programmation de création des bases de données.

1. Visual Paradigm:

Visual Paradigm est un outil de conception de diagramme UML.

1. Mermaid:

Mermaid est un outil de conception de diagramme UML.

# Partie VI

# Conclusion et Perspective

Ce projet s’est révélé très enrichissant dans la mesure où il consiste en une approche concrète du métier d’ingénieur en informatique. De plus, il nous a permis d’appliquer nos connaissances en gestion de projet et la programmation informatique à un domaine pratique. Nous comptons approfondir notre apprentissage et nos recherches sur l’apprentissage machine (Machine Learning), premièrement dans l’objectif de permettre à ce projet d’atteindre un niveau de maturité et de perfection nous permettant de le proposer sur le marché médical, deuxièmement dans la perspective de proposer des solutions innovantes dans d’autres spécialités du domaine médical, et en dernier lieu dans d’autres métiers qui pourront bénéficier de cet apport important que pourrait leur offrir cette technologie.

## Annexe

### A. Acronymes et Abréviations

**CLCB**: Clinique Claude Bernard

**FTP**: File Transfer Protocol (Protocol de transfert de fichiers)

**HP**: Hewlett-Packard

**HTTPS**: HyperText Transfer Protocol Secure (Protocole de transfert HyperText sécurisé)

**IA**: Intelligence Artificielle

**IBM**: International Business Machine Corporation

**OMT**: Object modeling technique

**OOD**: Object oriented design (Conception orientée objet)

**OOSE**: Object oriented software engineering (Ingénierie logicielle orientée objet)

**MCD**: Modèle Conceptuel de Données

**ML**: Machine Learning (Apprentissage automatique)

**RAID**: Redundant array of inexpensive drives (réseau redondant de disques bon marché)

**RUP**: Rational unified process (Processus unifié rationnel)

**SGBD**: Système de gestion de base de données

**SQL**: Structured query language (Langage de requêtes structurées)

**SSL**: Secure sockets layer (couche de sockets sécurisée)

**TFE**: Travail de fin d'études

**UML**: Unified modeling language (Langage de modélisation unifié)

**VPS:** Virtual private server (Serveur privé virtuel)

### B. Glossaire

**A**

**Administrateur:** Responsable de la gestion du système.

**Application:** Un logiciel qui peut être utilisé pour effectuer une certaine tâche ou une opération pour un ordinateur ou un système informatique.

**Authentification:** Procédure de sécurité pour l'identification des utilisateurs.

**B**

**Base de données:** Une compilation de données structurées qui permet de modéliser des objets dans un environnement spécifique et sert de support logiciel.

**C**

**Client:** Il s'agit d'ordinateurs connectés au réseau utilisant divers services sur d'autres périphériques réseau appelés serveurs.

**Codes Sources:** Ensemble d'instructions rédigées dans un langage machine clairement compréhensible par un humain dans le but d’obtenir un programme informatique.

**I**

**Intelligence Artificielle:** Imitation de l'intelligence humaine par des processus informatiques.

**M**

**Maladie cardiovasculaire:** Une maladie cardiovasculaire est un trouble cardiaque et de la circulation sanguine.

**Mot de passe :** Un jeu de caractères connu de l'utilisateur pour se connecter au système.

**N**

**Navigateur:** Logiciel permettant de visualiser des sites Web.

**S**

**Serveur:** Un ordinateur qui fournit ses ressources à d'autres ordinateurs qui lui sont connectés par une connexion réseau.

**Système de gestion de base de données (SGBD):** Logiciel spécialisé pour les saisies des données spécifiques, les mis-à-jour de données, et la formulation de requêtes dans une base de données.

**U**

**Utilisateur:** Quiconque peut utiliser le programme.

### D. Bibliographie

Gilles, Roy. <<Le modèle conceptuel de données>>, In Conception de bases de données avec UML. Presses de l’Université du Québec, Quebec,2009, pp.29-141

Gilles, Roy. <<Le modèle Logique de données>>, In Conception de bases de données avec UML. Presses de l’Université du Québec, Quebec,2009, pp.143-238

Pascal, Roques. <<Modélisation fonctionnelle : étude de cas>>, In UML2 par la pratique, Editions Eyrolles, France, 2006, pp. 20-54.

Pascal Roques. <<Modélisation statique : étude de cas>>, In UML2 par la pratique, Editions Eyrolles, France, 2006, pp. 80-121.

Pascal Roques. <<Modélisation dynamique : étude de cas>>, In UML2 par la pratique, Editions Eyrolles, France, 2006, pp. 156-183.

### E. Webographie

1. About us. (2017). Ada. <https://ada.com/about/>
2. Artificial intelligence has learned to probe the minds of other computers. (2018, août 16). Science | AAAS. <https://www.sciencemag.org/news/2018/07/computer-programs-can-learn-what-other-programs-are-thinking>
3. cairn.info. (s. d.). cairn. <https://www.cairn.info/>
4. Col, A. (2018, 20 février). Google veut diagnostiquer les maladies cardiovasculaires en analysant les yeux des patients. www.pourquoidocteur.fr. <https://www.pourquoidocteur.fr/Articles/Question-d-actu/24605-Google-veut-diagnostiquer-maladies-cardiovasculaires-analysant-les-yeux-patients>
5. Fedak, V. (2019, 4 juillet). Top 10 Most Popular AI Models. Dzone. <https://dzone.com/articles/top-10-most-popular-ai-models>
6. H. (2018, 20 février). Google veut diagnostiquer les maladies cardiovasculaires en analysant les yeux des patients. Pourquoi docteur. <https://www.pourquoidocteur.fr/Articles/Question-d-actu/24605-Google-veut-diagnostiquer-maladies-cardiovasculaires-analysant-les-yeux-patients>
7. Neveu, L. (2020a, février 19). Quel avenir pour l’intelligence artificielle dans l’automobile ? Futura. <https://www.futura-sciences.com/tech/questions-reponses/voiture-avenir-intelligence-artificielle-automobile-12806/>
8. Neveu, L. (2020b, février 19). Quel avenir pour l’intelligence artificielle dans l’automobile ? Futura. <https://www.futura-sciences.com/tech/questions-reponses/voiture-avenir-intelligence-artificielle-automobile-12806/>
9. OpenEdition : four platforms for electronic resources in the humanities and social sciences : OpenEdition Books, OpenEdition Journals, Hypotheses, Calenda. (2020, 21 octobre). OpenEdition. <https://www.openedition.org/?lang=en>
10. Rougetet, L. (2016a, février 20). Un ordinateur champion du monde d’Échecs : histoire d’un affronteme. . . OpenEdition. <https://journals.openedition.org/sdj/598#tocto2n4>
11. Rougetet, L. (2016b, février 20). Un ordinateur champion du monde d’Échecs : histoire d’un affrontement homme-machine. OpenEdition Journals. <https://journals.openedition.org/sdj/598#tocto2n4>
12. Veiga, A. (2020, 21 décembre). 10 Best AI Based Healthcare Apps You Can Try in 2020. SwissCognitive - The Global AI Hub. <https://swisscognitive.ch/2020/03/27/10-best-ai-based-healthcare-apps-you-can-try-in-2020/>
13. WebMD - Better information. Better health. (2005). WebMD. <https://www.webmd.com/>
14. World Health Organization : WHO. (2017, 17 mai). Maladies cardiovasculaires. <https://www.who.int/fr/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)>

1. Joshi, N. (2019, 25 juin). 7 Types Of Artificial Intelligence. *Forbes*. <https://www.forbes.com/sites/cognitiveworld/2019/06/19/7-types-of-artificial-intelligence/#1c290fa3233e> [↑](#footnote-ref-0)
2. Rougetet, L. (2016, 20 février). *Un ordinateur champion du monde d’Échecs : histoire d’un affrontement homme-machine*. OpenEdition Journals. <https://journals.openedition.org/sdj/598#tocto2n4> [↑](#footnote-ref-1)
3. Fedak, V. (2019, 4 juillet). *Top 10 Most Popular AI Models*. Dzone. <https://dzone.com/articles/top-10-most-popular-ai-models> [↑](#footnote-ref-2)
4. Neveu, L. (2020, 19 février). *Quel avenir pour l’intelligence artificielle dans l’automobile ?* Futura. <https://www.futura-sciences.com/tech/questions-reponses/voiture-avenir-intelligence-artificielle-automobile-12806/> [↑](#footnote-ref-3)
5. *Artificial intelligence has learned to probe the minds of other computers*. (2018, 16 août). Science | AAAS. <https://www.sciencemag.org/news/2018/07/computer-programs-can-learn-what-other-programs-are-thinking> [↑](#footnote-ref-4)
6. *Artificial intelligence has learned to probe the minds of other computers*. (2018, 16 août). Science | AAAS. <https://www.sciencemag.org/news/2018/07/computer-programs-can-learn-what-other-programs-are-thinking> [↑](#footnote-ref-5)
7. World Health Organization : WHO. (2017, 17 mai). *Maladies cardiovasculaires*. <https://www.who.int/fr/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)> [↑](#footnote-ref-6)
8. *About us*. (2017). Ada. https://ada.com/about/ [↑](#footnote-ref-7)
9. *WebMD - Better information. Better health.* (2005). WebMD. https://www.webmd.com/ [↑](#footnote-ref-8)
10. a remplir [↑](#footnote-ref-9)