

République Algérienne Démocratique et Populaire
Ministère De l'Enseignement Supérieur et de la Recherche Scientifique
Université Abderrahmane Mira - Béjaia -
Faculté des Sciences Exactes
Département Informatique



*Mémoire de fin de cycle
En vue de l'obtention du diplôme Master professionnel en Informatique
Spécialité : Génie Logiciel*

Thème

Conception et réalisation d'une application
mobile pour le Suivi des Patients à Distance

Réalisé par :

M^{elle} ALLAOUA Lydia

M^{elle} MAOUCHE Nada

Déposé le 13-10-2020 devant le jury composé de :

Examinatrice 1 : M^{elle} OUADA Sara Farah M.C.B Université de Bejaia.

Examinatrice 2 : M^{elle} TIAB Amal M.C.B Université de Bejaia.

Promotrice : M^{me} AIT ABDELOUHAB Karima, M^{me} AZOUI Aicha.

Promotion 2019-2020

- *Remerciements* -

La réalisation du présent travail a été rendue possible malgré la pandémie grâce au soutien et aux conseils de plusieurs personnes que nous tenons remercier ici.

Nos vifs remerciements s'adressent tout naturellement à **Madame AIT ABDELOUHAB Karima** et **Madame AZOUI Aicha** pour leur grande disponibilité, leur esprit de rigueur et de méthode, leurs conseils et leurs remarques pertinentes. Nous avons particulièrement apprécié leur soutien sans relâche ainsi que leurs critiques constructives qu'elles nous ont fournies à tout moment du déroulement de ce travail.

Nous tenons également à remercier tous les membres de jury pour avoir accepté d'évaluer notre travail.

Nous adressons nos sincères remerciements à tous ceux qui ont contribué par de diverses manières à l'aboutissement de ce travail. De tout coeur nous exprimons nos profondes gratitudeux aux membres de nos familles, nos parents, nos frères, soeurs et petits neveux, pour leur soutien tout au long de notre parcours.

En dernier lieu, nous pensons à tous nos amis qui nous ont soutenu d'une manière constante, et aux personnes chères à nos yeux qui veillent sur nous tout là haut.

- Dedicaces-

Je dédie ce travail

A ma petite maman, celle qui s'est toujours sacrifiée pour me voir réussir, celle qui a toujours été la pour moi, qui m'a soutenu et encouragé durant toutes ces années d'études.

A mon papa qui ne m'a jamais laissé manquer de quoi que ce soit, qui m'a toujours poussé et motivé dans mes études et ma vie quotidienne.

A ma soeur chérie, mon exemple, celle qui a toujours les bons mots pour me reconforter et me rendre le sourire.

A mes petits neveux MEISSANE et NASSIM, ma source de bonheur et de joie quotidienne.

A la mémoire de mon frère Nassim, qui est toujours présent dans mon cœur, j'aurai tellement aimé partager la joie de ma réussite en ta présence.

A tout mes amis qui ont toujours été là pour moi.

Table des matières

Table des matières	i
Table des figures	iii
Liste des Tableaux	v
Introduction générale	1
1 État de l’art sur la télésurveillance médicale	2
1.1 Introduction	3
1.2 Système d’information médicale	3
1.2.1 Acteurs impliqués dans un système d’information médicale	3
1.3 Définition d’un système de surveillance à distance	4
1.4 La télémédecine	4
1.4.1 Avantages de la telemedecine	4
1.4.2 Inconvenients de la télémédecine	5
1.4.3 Actes de la télémédecine	6
1.5 La télésurveillance médicale	6
1.5.1 Freins de la télésurveillance médicale	7
1.6 Technologies utilisées dans la Télésurveillance	7
1.6.1 Systèmes de capteurs	8
1.6.1.1 Personal Sensor Network (PSN)	8
1.6.1.2 Body Sensor Network (BSN)	8
1.6.2 Technologies de communication	10
1.6.2.1 Technologie Zigbee (IEEE 802.15.4)	10
1.6.2.2 Technologie WiFi (IEEE 802.11)	10
1.6.2.3 Technologie Bluetooth (IEEE 802.15.1)	10
1.6.2.4 Technologie Bluetooth à faible consommation d’énergie (BLE IEEE 802.15.1)	11
1.7 Travaux existants dans la Télésurveillance	11
1.7.1 Projet HIS (Habitat Intelligent pour la Santé)	12
1.7.2 Projet d’AMINE BEN YAHIA	12
1.7.3 Projet d’AILISA	13

1.7.4	Projet TISSAD	14
1.7.5	Synthèse sur les travaux existants	14
1.8	Proposition d'une architecture globale pour la télésurveillance à distance	15
1.9	Présentation du cas d'étude : CORONAVIRUS	17
1.10	Démarche de développement	20
1.10.1	Processus Unifié-UP	20
1.10.2	Caractéristiques du processus unifié	21
1.10.3	Activités et Phases	22
1.10.4	Les activités	22
1.10.5	Les Phases	23
1.11	Langage de modélisation	24
1.12	Conclusion	26
2	Spécification des Besoins	27
2.1	Introduction	28
2.2	Expression des besoins	28
2.2.1	Besoins fonctionnels	28
2.2.2	Besoins non fonctionnels	28
2.3	Identification des acteurs	29
2.4	Identification des cas d'utilisation	30
2.5	Diagramme de cas d'utilisation	31
2.5.1	Diagramme de cas d'utilisation de notre application mobile	32
2.5.2	Diagrammes de cas d'utilisation détaillés	33
2.6	Maquettes	46
2.6.1	Les principales interfaces (maquettes)	46
2.7	Conclusion	50
3	Analyse et Conception	51
3.1	Introduction	52
3.2	Diagrammes d'analyse	52
3.2.1	Diagramme de séquence système	52
3.2.1.1	Diagramme de séquence « Inscription »	54
3.2.1.2	Diagramme de séquence « Authentification »	55
3.2.1.3	Diagramme de séquence « Gérer patient »	56
3.2.1.4	Diagramme de séquence « Consulter mesures »	57
3.2.1.5	Diagramme de séquence « Générer alerte »	58
3.2.1.6	Diagramme de séquence « Consulter l'historique »	59
3.2.1.7	Diagramme de séquence « Consulter profile patient »	60
3.2.1.8	Diagramme de séquence « Rédiger un rapport »	61
3.2.1.9	Diagramme de séquence « Envoyer un rapport »	62
3.3	Conception	63

3.3.1	Diagrammes d'interaction	63
3.3.1.1	Diagramme d'interaction « Authentification »	64
3.3.1.2	Diagramme d'interaction « Inscription »	65
3.3.1.3	Diagramme d'interaction « Gérer patient »	66
3.3.1.4	Diagramme d'interaction « Consulter mesures »	67
3.3.1.5	Diagramme d'interaction « Consulter Historique »	68
3.3.1.6	Diagramme d'interaction « Consulter profile patient»	69
3.3.1.7	Diagramme d'interaction « Rédiger rapport »	70
3.3.1.8	Diagramme d'interaction «Envoyer un rapport »	71
3.3.2	Diagramme de classe	72
3.3.2.1	Diagramme de classe globale de notre application mobile	72
3.3.3	Dictionnaire de données	73
3.3.4	Passage relationnel	76
3.4	Conclusion	77
4	Réalisation	78
4.1	Introduction	79
4.2	Outils et langages de développement	79
4.2.1	Outils de développement	79
4.2.2	Langages de développement	81
4.3	Réalisation	82
4.4	Conclusion	105
	Bibliographie	107

Table des figures

1.1	Acteurs du système d'information médicale[13]	3
1.2	Architecture globale du système de surveillance à distance	15
1.3	Schéma résumant les différentes étapes de notre système de surveillance à distance	16
1.4	Symptômes du COVID-19 [6]	18
1.5	Taux de mortalité selon l'âge [6]	19
1.6	Comparaison entre le covid-19 et autres gripes [6]	20
1.7	Processus Unifié [14]	21
1.8	Les différents diagrammes UML [15]	25
2.1	Diagramme de cas d'utilisation de notre application	32
2.2	Diagramme de cas d'utilisation « Inscription »	33
2.3	Diagramme de cas d'utilisation « Authentification »	34
2.4	Diagramme de cas d'utilisation « Consulter espace patient »	35
2.5	Diagramme de cas d'utilisation « Consulter les Mesures »	36
2.6	Diagramme de cas d'utilisation « Consulter Historique »	37
2.7	Diagramme de cas d'utilisation « Générer Alerte »	38
2.8	Diagramme de cas d'utilisation « Gérer Patient »	40
2.9	Diagramme de cas d'utilisation « Rédiger un Rapport »	41
2.10	Diagramme de cas d'utilisation « Envoyer un rapport »	42
2.11	Diagrammes de cas cas d'utilisation « Collecter les données »	43
2.12	Diagramme de cas d'utilisation « Envoyer les données »	44
2.13	Diagramme de cas d'utilisation « Stocker les données »	45
2.14	Maquette de l'interface d'accueil	47
2.15	Maquette de l'interface Authentification	47
2.16	Maquette de l'interface d'inscription	48
2.17	Maquette de l'interface représentant le type d'utilisateurs	48
2.18	Maquette de l'interface Espace patient	49
2.19	Maquette de l'interface Espace médecin	49
2.20	Maquette de l'interface Dossier médicale	50
3.1	Diagramme de séquence « Inscription »	54
3.2	Diagramme de séquence « Authentification »	55

3.3	Diagramme de séquence « Gérer patient »	56
3.4	Diagramme de séquence « Consulter mesures »	57
3.5	Diagramme de séquence « Générer alerte »	58
3.6	Diagramme de séquence « Consulter l'historique »	59
3.7	Diagramme de séquence « Consulter profile patient »	60
3.8	Diagramme de séquence « Rédiger un rapport »	61
3.9	Diagramme de séquence « Envoyer un rapport »	62
3.10	Diagramme d'interaction « Authentification »	64
3.11	Diagramme d'interaction « Inscription »	65
3.12	Diagramme d'interaction « Gérer patient »	66
3.13	Diagramme d'interaction « Consulter mesures »	67
3.14	Diagramme d'interaction « Consulter Historique »	68
3.15	Diagramme d'interaction « Consulter profile patient »	69
3.16	Diagramme d'interaction « Rediger un rapport »	70
3.17	Diagramme d'interaction « Envoyer un rapport »	71
3.18	Diagramme de classe globale	73
4.1	Interface d'Accueil	82
4.2	Interface Choix de l'utilisateur	83
4.3	Interface d'Authentification	84
4.4	Interface d'Inscriptions	85
4.5	Espace Patient	86
4.6	Interface Mesures du patient	87
4.7	Interface Mesures de la Fievre	88
4.8	Interface Mesures du Rythme Cardiaque	89
4.9	Interface Mesures de la fièvre	90
4.10	Notification d'Alerte	91
4.11	Interface Choix des mesures	92
4.12	Interface Historique du Rythme Cardiaque	93
4.13	Interface Espace Médecin	94
4.14	Interface Ajouter un patient	95
4.15	Interface Supprimer un patient	96
4.16	Interface Profil du Patient	97
4.17	Interface choix des mesures	98
4.18	Interface mesures du Rythme Cardiaque	99
4.19	Interface mesures de la Fièvre	100
4.20	Interface choix de l'Historique	101
4.21	Interface Historique du Rythme cardiaque	102
4.22	Interface Rédiger un rapport	103
4.23	Interface Envoyer un rapport	104

Liste des tableaux

1.1	Les différents capteurs et leurs fonctions	9
1.2	Technologies de communication	11
2.1	Identification des acteurs du système	30
2.2	Identification des cas d'utilisation du système	31
2.3	Description textuelle du cas d'utilisation « Inscription »	33
2.4	Description textuelle du cas d'utilisation « Authentification »	35
2.5	Description textuelle du cas d'utilisation « Consulter espace patient »	36
2.6	Description textuelle du cas d'utilisation « Consulter les Mesures »	37
2.7	Description textuelle du cas d'utilisation « Consulter Historique »	38
2.8	Description textuelle du cas d'utilisation « Générer une Alerte »	39
2.9	Description textuelle du cas d'utilisation « Gérer Patient »	41
2.10	Description textuelle du cas d'utilisation « Rédiger un rapport »	42
2.11	Description textuelle du cas d'utilisation « Envoyer un rapport »	43
2.12	Description textuelle du cas d'utilisation « Collecter les données »	44
2.13	Description textuelle du cas d'utilisation « Envoyer les données »	45
2.14	Description textuelle du cas d'utilisation « Stocker les données »	46
3.1	Dictionnaire de données	75
3.2	Description des classes de l'application a réaliser	76

Introduction générale

La population mondiale est en croissance exponentielle et le nombre de personnes malades et âgées augmente assez rapidement, les médecins ne peuvent plus assurer leur travail de la même façon qu'il y a un siècle.

Ce pendant les progrès réalisés dans les technologies de l'information ont ouvert de nouvelles perspectives pour la création de réseaux corporels sans fil qui ont le potentiel de compléter le rôle du personnel médical et d'améliorer la qualité des soins et services apportés aux patients.

Les systèmes de surveillance à distance permettent une surveillance omniprésente des signes vitaux des patients à l'hôpital et dans les environnements de soins à domicile et leur permettre de se concentrer sur leurs besoins en assurant les soins et services médicaux adéquats.

Le but de ce travail est de réaliser une application mobile sous android pour le suivi des patients à distance afin d'améliorer la qualité de vie des patients particulièrement les personnes âgées, devenant ainsi plus indépendants.

Notre application aura comme objectif majeur de rendre possible la surveillance des patients partout et à tout moment permettant ainsi la détection instantanée d'anomalies de santé des patients, la prédiction et la prévention de problèmes sérieux et critiques de leur état de santé.

Afin d'aboutir à la réalisation de notre application, notre travail est reparti en quatre chapitres qu'on peut décrire comme suit :

Le premier chapitre « État de l'art » présente une vue d'ensemble sur les systèmes de surveillance médicale à base de capteurs.

Le deuxième chapitre « Spécification des besoins » décrit les besoins fonctionnels et non fonctionnels de notre application en présentant les cas d'utilisation sous différents diagrammes.

Le troisième chapitre « Analyse et conception » aborde la phase de conception détaillée de notre application.

Le quatrième chapitre « Réalisation » évoque tous les outils et langages de développement utilisés pour la réalisation de notre application.

Enfin nous terminons ce travail par une conclusion générale résumant les différents points de notre mémoire et dégageons quelques perspectives pour améliorer notre application.

Chapitre 1

État de l'art sur la télésurveillance médicale

1.1 Introduction

Les systèmes de surveillance médicale à base de capteurs sont utilisés dans de différentes applications médicales, pour les quelles diverses systèmes et prototypes ont été développés à ce jour, telles que la surveillance quotidienne des patients à domicile, à l'hôpital encore à l'extérieur.

L'objectif de ce chapitre est de décrire globalement les systèmes de télésurveillance médicale, tout en citant et présentant quelques travaux existants dans ce domaine. Par la suite, nous présenterons notre approche.

Pour finir, nous terminons ce chapitre par une conclusion.

1.2 Système d'information médicale

Un système d'information médical est un système conçu pour gérer les données relatives aux informations médicales et administratives au sein d'un hôpital. Il permet de collecter, stocker, traiter et gérer le dossier médical électronique d'un patient et la gestion opérationnelle d'un hôpital. Il peut être utilisé pour améliorer les résultats des patients, informer la recherche et influencer à la prise de décision [1].

En effet, l'objectif majeur d'un système d'information médical consiste à optimiser la prise en charge de l'activité de soins en améliorant la gestion de l'information ainsi que la coordination des tâches médicales, administratives et logistiques effectuées au sein de l'établissement médical [1].

1.2.1 Acteurs impliqués dans un système d'information médicale

La figure 1.1 représente les différents acteurs impliqués dans le système d'information médical :

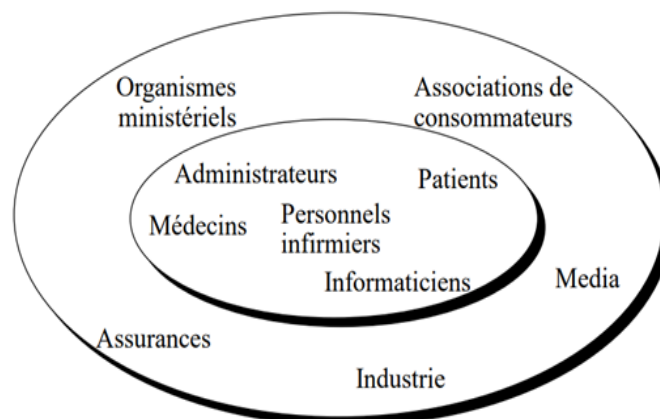


FIGURE 1.1 – Acteurs du système d'information médicale[13]

Les acteurs qui jouent un rôle le plus important dans le système d'information médicale sont les suivants :

1. **Acteurs médicales (Medecins)** : ont pour ambition de prendre en charge les patients et de contribuer à leur état de santé et ainsi de répondre à toutes leurs détresses , quelque soit leur âge et leurs problèmes de santé.
2. **Acteurs paramédicales (Infirmiers)** : ont pour rôle de soigner les malades et s'en occuper, sous la direction des médecins.
3. **Acteurs administratifs (Secrétaires)** : ont pour rôle de garder le contact avec les patients à partir d'une réception physique ou par téléphone des patients ,permet ainsi de gérer l'accueil et les dossiers administratifs .

1.3 Définition d'un système de surveillance à distance

Souvent utilisé dans de différentes applications médicales, le système de surveillance à distance à base de capteurs permet la collecte, la communication, et l'analyse des données. Grâce au nœud de capteurs, les signaux vitaux des patients à l'hôpital et dans les environnements de soins à domicile peuvent être détectés et traités améliorant ainsi le système de santé [2].

1.4 La télémédecine

La télémédecine est l'échange d'informations médicales d'un endroit à un autre par communication électronique, qui améliore l'état de santé des patients. La télémédecine a de multiples applications et peut être utilisée pour différents services, qui comprennent les outils sans fil, le courrier électronique, les téléphones intelligents et d'autres méthodes de technologie des télécommunications. La télémédecine a commencé il y a plus de 40 ans avec les hôpitaux qui ont étendu leurs services aux patients des régions éloignées. Elle s'est rapidement développée et est devenue une partie intégrante des départements spécialisés, des hôpitaux, des cabinets médicaux privés, des soins de santé à domicile, ainsi que du domicile et du lieu de travail du consommateur [3].

1.4.1 Avantages de la telemedecine

La télémédecine est de plus en plus populaire car elle présente quatre grands avantages [3] :

1. **Un meilleur accès aux soins** : Elle permet d'assurer la continuité de l'accès aux soins pour une partie importante de la population, elle est un outil précieux pour traiter des urgences médicales et enfin sa mise en œuvre favorise les réseaux multidisciplinaires dont la médecine de demain aura besoin.
2. **Un meilleur rapport coût-efficacité** : L'une des raisons les plus importantes d'inclure la technologie de la télésanté est la capacité de contenir ou de réduire le coût des soins de santé. La télémédecine est en mesure de réduire les coûts des soins de santé grâce à son efficacité, à une meilleure gestion des maladies chroniques, à la réduction des temps de déplacement, à la diminution des séjours à l'hôpital et au partage du personnel professionnel de santé.

3. **Améliorer la qualité des soins de santé** : De nombreuses études ont montré qu'il y a une amélioration de la qualité des soins de santé lorsque les services de télé-médecine sont utilisés. Ils sont tout aussi bons que les services fournis lors de consultations en personne. Dans certains cas, tels que les soins intensifs ou la santé mentale, la télé-médecine surpasse même les services traditionnels. Les patients obtiennent de meilleurs résultats et sont plus satisfaits de la télé-médecine.
4. **La télé-médecine est populaire auprès des consommateurs** : Certains des impacts les plus profonds de la télé-médecine se font sentir sur le patient, sa famille et sa communauté. Les patients ont moins de stress et de temps de déplacement avec l'utilisation de la télé-médecine. Au cours des 15 dernières années, de nombreuses études ont montré la satisfaction des patients et un soutien accru aux services de télé-médecine. Ces services permettent aux patients d'avoir accès à des prestataires qui seraient autrement inaccessibles, et d'accéder à des services médicaux sans avoir à se déplacer sur de longues distances.

1.4.2 Inconvénients de la télé-médecine

Si la télé-médecine promet de se développer rapidement au cours de la prochaine décennie et présente des avantages évidents, elle pose encore certains problèmes techniques et pratiques aux prestataires de soins de santé [3].

Dans ce qui suit, nous présentons deux inconvénients.

1. **Formation technique et équipement** :

La restructuration des responsabilités du personnel informatique et l'achat d'équipements prennent du temps et coûtent de l'argent. La formation est essentielle pour mettre en place un programme de télé-médecine efficace. Les médecins, les responsables de cabinet et les autres membres du personnel médical doivent être formés aux nouveaux systèmes pour assurer un solide retour sur investissement. Donc, les besoins en termes de personnel peuvent être diminués.

2. **Réduction de la continuité des soins** :

Dans les cas où les patients utilisent des services de télé-médecine à la demande qui les mettent en relation avec un prestataire de soins aléatoire, la continuité des soins en souffre. Le prestataire de soins d'un patient peut ne pas avoir accès aux dossiers de ces autres visites et se retrouver avec des antécédents incomplets pour le patient. Le remaniement des prestataires de services augmente le risque qu'un médecin ne connaisse pas les antécédents d'un patient ou n'ait pas de notes sur les routines de soins. Étant donné qu'une continuité des soins réduite peut diminuer la qualité des soins, les fournisseurs de télé-médecine grand public doivent appliquer des solutions de données solides pour maintenir les dossiers des patients adéquats et accessibles. À mesure que les prestataires de soins de santé adopteront des solutions de télé-santé à utiliser avec leurs propres patients, la continuité des soins augmentera probablement, réduisant le risque que les patients se retrouvent dans une clinique ou un centre de soins d'urgence lorsqu'ils ont besoin de soins rapides.

1.4.3 Actes de la télémédecine

Selon le décret numéro 2010-1229 datant du 19 octobre 2010, il a été défini 5 champs d'application pour la télémédecine qu'on peut représenter comme suit [3] :

1. **La téléconsultation :**

Le médecin donne une consultation à distance à un patient, lequel a la possibilité, s'il le souhaite, d'être assisté d'un professionnel de santé. Le patient, seul ou accompagné du professionnel, fournissent les informations et le médecin à distance pose un diagnostic.

2. **La télé-expertise :**

Un médecin sollicite à distance l'avis d'un ou de plusieurs confrères sur la base d'informations médicales liées à la prise en charge d'un patient.

3. **La télésurveillance médicale :**

Un médecin surveille et interprète à distance les paramètres médicaux d'un patient. L'enregistrement et la transmission des données peuvent être automatisés ou réalisés par le patient lui-même ou par un professionnel de santé.

4. **La téléassistance médicale :**

Un médecin assiste à distance un autre professionnel de santé au cours de la réalisation d'un acte.

Dans le cadre de notre travail, nous nous intéressons plus particulièrement à la télésurveillance médicale.

1.5 La télésurveillance médicale

Le but de la télésurveillance médicale à domicile est de promouvoir les auto soins par la formation des patients, de manière à ce que ces derniers puissent prendre l'habitude de se surveiller eux-mêmes et qu'ils aient le pouvoir de jouer un rôle plus actif dans la gestion de la variation de leurs traitements journaliers. Cela va influencer le rôle de l'industrie, les délégués et les salariés de l'industrie du matériel médical devront être directement en contact avec les patients (exemple : implanter un dispositif personnalisé pour donner des informations sur le fonctionnement du dispositif et pour le réparer). Les systèmes de télésurveillance médicale sont généralement composés en deux modules distants [3] :

1. Une unité locale, au niveau de chaque habitat, de traitement des signaux reçus des capteurs, de gestion d'une base de connaissances relative au patient, et à l'origine de l'émission de messages et d'alarmes.
2. un centre de télé-vigilance pour le traitement des messages reçus des habitats ainsi qu'un ensemble d'acteurs (personne médicale, patient et membre de sa famille) pouvant accéder aux données du système.

La télésurveillance apporte de nombreux bénéfices pour les patients et les professionnels de santé. L'enjeu majeur, qui découlera de la mise en place d'une télésurveillance médicale, sera l'amélioration du suivi des patients. Ceci permettra de traiter les anomalies plus tôt et ainsi modifier le traitement administré afin de réduire les traitements médicamenteux lourds, le besoins d'hospitalisation et les coûts de transport inutiles [3].

1.5.1 Freins de la télésurveillance médicale

En plus des freins cités pour la télé-médecine (§ 1.4.2), dans le cadre particulier de la télésurveillance médicale on retrouve un certain nombre de contraintes telles que des contraintes éthiques, sociales et individuelles (Respect de la vie privée, confidentialité des données). Les données de santé relèvent de l'intimité et de la vie privée et appellent à une protection renforcée. Les règles relatives à la sécurité ont toujours pris une place importante dans le domaine médical. Avec la multiplication des transmissions de données médicales et l'accroissement du nombre de personnes susceptibles d'accéder aux réseaux informatiques, la sécurité doit être une priorité renforcée [3].

1.6 Technologies utilisées dans la Télésurveillance

La plupart des systèmes de santé ont la capacité d'interagir avec des catégories de personnes différentes (les patients, les personnes âgées, etc.) en utilisant un ou plusieurs types de capteurs. Les capteurs sont responsables de l'acquisition des données et sont soit des capteurs fixes déployés dans les environnements domestiques (**Personal sensor Network-PSN**), soit des capteurs portables portés par la personne surveillée (**Body Sensor Network-BSN**). Ces deux catégories de capteurs recueillent des informations contextuelles sur la personne de manière constante ou périodique. Les capteurs relaient ces données à travers un point d'accès ou une station de base, vers un serveur ensuite vers des appareils mobiles via des technologies de réseau disponibles en permanence (Internet, Bluetooth, Zigbee, Wifi, etc.).

Au niveau de l'application, les acteurs d'un système de surveillance de la santé (patients ou prestataires de soins) utilisent une interface utilisateur graphique. Selon la complexité du système, les interfaces graphiques peuvent être utilisées pour la surveillance et l'évaluation de l'état de santé des personnes, des paramètres de santé, et peuvent notifier des alertes aux soignants en cas de détection d'une anomalie ou d'une situation d'urgence.

Divers paramètres et données recueillis par les capteurs peuvent être impliqués, comme l'environnement (par exemple, la température, l'humidité et la détection sonore), le suivi des mouvements et de la localisation (par exemple, les gestes et la pression) et les signes vitaux (par exemple, le rythme cardiaque, la saturation en oxygène et la pression sanguine). Ces données fournissent une vue sur l'état de santé de la personne et sur son environnement et son cadre de vie [4].

1.6.1 Systèmes de capteurs

L'acquisition de données est la première étape dans les environnements de télésurveillance dans lesquels diverses sources sont utilisées pour recueillir les informations relatives à l'état physique de la personne (patient), son comportement, l'environnement, les activités réalisées, etc. Aujourd'hui, les systèmes et les projets de surveillance des soins de santé utilisent des capteurs dans le but de recueillir des données brutes pour surveiller les individus et leur environnement. Il existe deux grandes classes de réseaux interconnectés qui sont souvent utilisés dans ce domaine : Réseau de capteurs personnels (PSN), réseau de capteurs corporels (BSN). Chaque capteur est responsable d'une ou plusieurs tâches en même temps.

Dans ce qui suit, nous détaillerons ces deux types de capteurs.

1.6.1.1 Personal Sensor Network (PSN)

Le PSN ou les capteurs environnementaux sont chargés de saisir et de récupérer les données contextuelles relatives à la personne et à l'environnement. Les PSN peuvent être placés dans un lieu de vie ou attachés à des objets domestiques afin de détecter les activités de la personne. Ces objets peuvent être un canapé, une table, un lit, des chaises ou des meubles équipés de capteurs de pression. Les interactions personnelles avec les objets domestiques dans un endroit précis, combinées aux observations environnementales, peuvent indiquer la performance de la personne dans ses activités de la vie quotidienne. Par exemple, si un capteur de mouvement identifie l'emplacement actuel de l'utilisateur comme étant la cuisine et qu'un capteur d'objets de cuisson (par exemple, gaz, four, grille-pain ou plaque de cuisson) est allumé, et qu'il y a une consommation d'eau ou que la porte du réfrigérateur est ouverte (en utilisant un capteur de robinet mélangeur ou des interrupteurs à contact), alors les données détectées indiquent que l'activité de préparation des repas a lieu. Par conséquent, les capteurs environnementaux peuvent fournir des informations contextuelles riches pour détecter les activités quotidiennes et observer le comportement [4].

1.6.1.2 Body Sensor Network (BSN)

Les BSN utilisent des capteurs portables portés par les personnes surveillées telles que les personnes âgées et les patients. Ces capteurs sont utilisés pour fournir des informations en continu. Ils sont souvent intégrés dans des accessoires tels que des vêtements, des ceintures, des montres ou des lunettes. Les BSN utilisent souvent des unités de mesure inertielle telles que des accéléromètres pour détecter les activités ambulatoires ou des dispositifs de signes vitaux tels que des capteurs de fréquence cardiaque pour surveiller l'état de santé. Les accéléromètres et les gyroscopes sont les capteurs inertiels les plus couramment utilisés pour surveiller les mouvements et les postures du corps comme la position debout, assise et la marche. Le système de positionnement global (GPS) peut également être utilisé comme un capteur portable pour surveiller les activités basées sur la localisation dans un environnement ouvert ou mobile. Par exemple, pour connaître les emplacements des patients et prédire les mouvements ou pour apprendre et déduire le mode de transport de l'utilisateur en se basant sur les données enregistrées par le GPS. Plusieurs biocapteurs sont utilisés

pour surveiller les signes vitaux des patients et des personnes âgées tels que le rythme cardiaque, la saturation en oxygène, la pression sanguine, la glycémie, la température du corps, le poids, etc. On peut citer les capteurs portables intégrés dans les montres, les chemises et les ceintures. Ces capteurs fournissent des paramètres physiologiques en temps réel et des valeurs liées à l'état de santé du sujet surveillé. Il existe divers biocapteurs, tels que les capteurs d'électrocardiographie (ECG) utilisés pour surveiller l'activité cardiaque, les capteurs d'électroencéphalographie (EEG) utilisés pour surveiller l'activité cérébrale, les capteurs d'électromyographie (EMG) utilisés pour surveiller l'activité musculaire et les capteurs d'électrooculographie (EOG) utilisés pour surveiller les mouvements des yeux. Les oxymètres de pouls sont utilisés pour mesurer le niveau d'oxygène du sang (c'est-à-dire la saturation en oxygène) tandis que les capteurs de photopléthysmographie (EPG) sont utilisés pour surveiller le taux d'oxygène du sang [4].

Le tableau 1.1 résume les différents capteurs ainsi que leurs fonctions [4] :

Catégorie	Nom	Fonction
PSN	PIR	Détection de mouvement
	Pressure	Identifier le lieu
	Ultrasonic	Localisation et posture
	Contact switches	Détection d'ouverture/fermeture (par exemple, portes)
	Light	Utilisation de la lumière et de son intensité
	Temperature	Mesure de la température ambiante
	Weight	Poids des personnes âgées
	Humidity	Mesure de l'humidité ambiante
	Power	mesure de la consommation d'énergie
BSN	Accelormeter	Mesure de l'accélération, détection des chutes, localisation et posture
	Gyroscope	sMesure de l'orientation, détection de mouvement
	GPS	Détection de mouvement et localisation
	ECG	Moniteur d'activité cardiaque
	EEG	Mesure des ondes cérébrales
	EOG	Surveiller le mouvement des yeux
	EMG	Surveiller l'activité musculaire
	PPG	Fréquence cardiaque et vitesse du sang
	Pulse	oximeter Mesure de la saturation en oxygène du sang
	Blood pressure	Mesure de la pression artérielle
	SKT	Mesure de température de la peau

TABLE 1.1 – Les différents capteurs et leurs fonctions

1.6.2 Technologies de communication

Pour servir les transmissions de données intérieures et extérieures entre les capteurs, les stations de base et le serveur dans les réseaux de zones personnelles et corporelles, plusieurs technologies de communication sans fil sont disponibles. Les technologies les plus populaires et les plus utilisées sont les protocoles sans fil à courte portée tels que Zigbee, Bluetooth, Bluetooth LowEnergy et le WiFi.

1.6.2.1 Technologie Zigbee (IEEE 802.15.4)

ZigBee est un Protocol de réseau sans fil à faible débit de données, à faible consommation d'énergie et à faible coût, destiné aux applications d'automatisation et de contrôle à distance. Il est utilisé pour les réseaux personnels sans fil (WPAN) avec une communication radio à courte portée, il est plus simple et moins coûteux que d'autres réseaux WPAN comme Bluetooth. ZigBee peut fonctionner avec un débit de données allant de 20 Kbps à 250 Kbps. Il prend en charge trois types de topologies : maillage, étoile et arbre à amas. IEEE et ZigBee Alliance ont travaillé en étroite collaboration pour spécifier l'ensemble de la pile de protocoles. La norme IEEE802.15.4 se concentre sur la spécification des deux couches inférieures du protocole (couche physique et couche de liaison de données). D'autre part, ZigBee Alliance vise à fournir les couches supérieures de la pile de protocoles (de la couche réseau à la couche application) pour la mise en réseau de données interopérables, les services de sécurité et une gamme de solutions sans fil de contrôle des maisons et des bâtiments, à fournir des tests de conformité à l'interopérabilité, à commercialiser la norme et à faire évoluer la norme grâce à une ingénierie avancée [4].

1.6.2.2 Technologie WiFi (IEEE 802.11)

Le WiFi est une technologie populaire de communication et de transfert de données sans fil. Elle est basée sur la série de normes IEEE 802.11 utilisées pour les communications sans fil dans un réseau local (LAN). La vitesse de transmission élevée est le principal avantage du WiFi. Le réseau prend en charge les topologies en étoile et point à point où les appareils sont interopérables entre eux. La couverture WiFi peut inclure plusieurs appareils électroniques capables de se connecter au réseau local ou à l'internet via un point d'accès au réseau (Access Point-AP) sans fil d'une distance moyenne de 100 mètres et d'une vitesse de transmission à large bande allant jusqu'à 54 Mbps selon la norme IEEE utilisée. L'inconvénient de cette technologie est la consommation d'énergie relativement importante. Le WiFi représente un bon candidat pour les capteurs et les dispositifs qui sont déployés dans les habitats pour assurer une surveillance continue [4].

1.6.2.3 Technologie Bluetooth (IEEE 802.15.1)

La technologie Bluetooth (Bluetooth Technology-BT) est conçue pour les communications sans fil à courte distance. C'est une technologie de communication sans fil ouverte basée sur la norme IEEE 802.15.1. Elle est utilisée pour connecter divers appareils pour les transmissions de données

et de voix dans le réseau WPAN. Le nombre de dispositifs BT peut être de deux ou plus jusqu'à huit dans la topologie de réseau à courte distance [4].

1.6.2.4 Technologie Bluetooth à faible consommation d'énergie (BLE IEEE 802.15.1)

La technologie Bluetooth LowEnergy (Bluetooth LE) consomme moins d'énergie que la technologie Bluetooth standard et est utilisée dans des appareils tels que les traqueurs de fitness, les montres intelligentes et d'autres appareils connectés afin de transmettre des données sans fil sans compromettre fortement la puissance de la batterie du téléphone de l'utilisateur. Le Bluetooth utilise les ondes radio ultra hautes fréquences (Ultra High Frequency – UHF) pour le transfert de données. Cette technologie a été normalisée à l'origine sous la référence IEEE 802.15.1 [4].

Le tableau 1.2 représente un résumé de ces technologies [4] :

Type	Zigbee	Wifi	Bluetooth	Bluetooth LE
Standard	IEEE 802.15.4	IEEE 802.11	IEEE 802.15.1	IEEE 802.15.1
Fréquence de transmission	868 Mhz, 915 Mhz, and 2.4 Ghz	2.4 and 5 Ghz	2.4 Ghz	2.4 Ghz
Topologie	Maille, étoile et amas	arbre Étoile et point à point	Piconet	Piconet
Débit de données	20, 40 and 250 Kbs	11 and 54 Mbs	1 to 3 Mbs	1 Mbs
Portée(mètre)	10 à 100	100	10	10
Consommation d'énergie	Très faible	Elevée	Moyenne	Très faible
Coût	Faible	Elevé	Elevé	Faible
Durée de Vie de la batterie	Mois ou années	Heures	Jours	Mois ou années
Sécurité	128-bit AES	SSID authentication	64-128 bit	128-bit AES
Optimisé pour	Faible puissance et faible coût,fiabilité et évolutivité	Rapidité, exhibilité	Commodité	Faible coût et faible puissance

TABLE 1.2 – Technologies de communication

1.7 Travaux existants dans la Télésurveillance

Des nombreux projets variés avec différents concepts et objectifs sont menés à travers le monde. Ils visent par exemple à définir une architecture générique pour de tels systèmes de surveillance, à

expérimenter un système de télésurveillance sur une catégorie spécifique de patients (insuffisants cardiaques et pulmonaires, asthmatiques, diabétiques, patients souffrant de la maladie d'Alzheimer, etc.), ou encore à concevoir des appartements, des capteurs, des systèmes d'alarmes adaptés aux exigences de la télésurveillance médicale.

Un ensemble de projets et de concepts relatifs au domaine de la télésurveillance médicale à domicile sont présentés dans cette section du chapitre.

1.7.1 Projet HIS (Habitat Intelligent pour la Santé)

a. Description :

Ce projet HIS de mise en place d'un Habitat Intelligent pour la Santé, développé au sein de l'équipe AFIRM (Acquisition, Fusion d'Informations et Réseaux pour la Médecine) du laboratoire TIMC-IMAG à Grenoble concerne la problématique de surveillance de personnes à domicile et investigate en particulier le concept d'habitat intelligent. Il s'inscrit cependant dans un projet plus vaste définissant le système d'information complet associé à un habitat télé-surveillé dans son environnement médico-social : SIC-HIS (Système d'Information et de Communication de l'Habitat Intelligent pour la Santé) [5].

b. Objectif :

L'objectif du projet HIS est de détecter les besoins du patient, les situations inhabituelles et urgentes (chutes, malaises, hypotension, etc.) nécessitant l'intervention d'un centre de soins. L'habitat intelligent fait partie d'un projet plus vaste définissant le système d'information complet associé au logement télé-surveillé dans son environnement médico-social : SIC-HIS (« Système d'Information et de Communication de l'Habitat Intelligent pour la Santé »). Il intègre sur un même réseau médical plusieurs habitats de patients, des centres de télé-vigilance, des postes de médecins et d'autres membres du corps médical et des postes de consultants occasionnels. En effet, l'objectif ultime est de pouvoir gérer simultanément les alertes émanant de plusieurs patients à domicile [5].

1.7.2 Projet d'AMINE BEN YAHIA

a. Description :

L'auteur propose un processus méthodologique afin de faciliter l'analyse et la conception de systèmes de télésurveillance médicale pour la détection précoce de signes précurseurs à une complication. Le processus doit permettre d'identifier les aspects génériques et spécifiques de chaque système et les architectures ainsi conçues doivent prendre en compte l'ensemble des données du patient, son profil, ses antécédents, les médicaments prescrits, les données physiologiques et comportementales ainsi que les données de son environnement. Ces architectures doivent également être ouvertes pour s'adapter à de nouvelles sources de données [3].

b. Objectif :

Les informations représentent une banque de données privées qu'il faut sécuriser par des protocoles d'accès et de transmission. Elle peut être utilisée par les professionnels de santé dans leur travail quotidien en y accédant via un portail de service. Pour gérer le vocabulaire des différents professionnels de santé (médecin, infirmier, etc.), la méthodologie proposée doit prendre en compte la construction et l'intégration de telles ontologies de domaine. L'utilité des ontologies de domaines est de partager une sémantique et garder une certaine cohérence des données. Cette méthodologie doit pro-poser au personnel soignant un accès aux données des patients pour le suivi et la mise à jour. La télésurveillance étant basée sur une architecture distribuée, le module installé chez le patient doit avoir un certain niveau d'autonomie et de réactivité. Cela permettra la détection de situations dangereuses sans passer par le module distant (par exemple, dans le cas d'absence de connexion). Pour la détection d'anomalies, cette méthodologie propose un système expert basé sur des règles d'inférences construites en collaboration avec les experts médicaux. Ces règles doivent être génériques avec la capacité d'évoluer avec l'état du patient [3].

1.7.3 Projet d'AILISA

a. Description :

Le projet AILISA a pour but de mettre en place des plateformes pérennes pour l'évaluation de technologies de télésurveillance médicale et d'assistance en gérontologie. Il a démarré au début de 2004 grâce à un financement du RNTS (Réseau National Technologies pour la Santé) dans le cadre de l'Institut de la Longévité. Les plateformes seront installées dans deux services gériatriques : l'un à l'hôpital Charles Foix (Ivry-sur-Seine) et l'autre au CHU La Grave (Toulouse), et dans deux appartements d'un foyer logement pour personnes âgées (Grenoble). Les sites d'évaluation disposeront de trois technologies mises au point dans les laboratoires de la recherche publique française : L'Habitat Intelligent pour la Santé (TISSAD), le vêtement de Télé-assistance Médicale Nomade (VTAMN) et le robot déambulateur (MONIMAD). Il s'agit ici d'évaluer ces technologies sur les plans technologiques, médicaux et aussi sur le plan de l'usage et de l'éthique. Les plateformes d'expérimentations s'appuient sur un système d'information pour faire circuler des informations entre le lieu où séjournent le patient et les lieux où se trouvent les acteurs, médecins et scientifiques, qui le prennent en charge à distance. Ce système d'information collecte des données sur le patient et son environnement, au travers d'un réseau local de capteurs, relié à un ordinateur (PC) lui même connecté au réseau Internet. L'ordinateur est doté des logiciels pour communiquer avec les capteurs et vers l'extérieur, mais aussi pour traiter les données des capteurs, pour les mettre en forme dans une base de données, et enfin pour délivrer des indicateurs sur l'état du patient [5].

b. Objectifs :

Les objectifs du projet AILISA sont à la fois ambitieux et pragmatiques [5] :

- Mettre en place, dans des environnements contrôlés, des plateformes pour l'évaluation médicale, technique et d'usage, de technologies pour le maintien à domicile de personnes

âgées dépendantes, en prenant en compte très tôt la dimension éthique de la prise en charge de la santé par des moyens technologiques.

- Créer et pérenniser des lieux de validation qui permettront d'accumuler l'expérience et d'augmenter la connaissance en toute sécurité.
- Rédiger des préconisations (matériels, logiciels, mise en œuvre) pour les dispositifs technologiques de maintien à domicile de personnes âgées vivant seules.
- Organiser et encourager le transfert technologique des solutions qui auront été validées sur ces plateformes, en s'appuyant sur des professionnels de l'industrialisation.

1.7.4 **Projet TISSAD**

a. **Description :**

Le projet TISSAD développé par Thomesse et al a pour objectif principale la définition d'une architecture générique, modulaire et ouverte pour les systèmes de télésurveillance, adaptable à diverses pathologies traitées à domicile (suivi de personnes âgées, d'insuffisants cardiaques et rénaux) [5].

Le projet TISSAD a été centré sur l'utilisateur en regroupant ses données dans 4 classes :

- identification,
- historique des prescriptions,
- historique clinique,
- les données médicales.

b. **Objectif :**

L'objectif principal de ce projet est de prévenir des accidents ou des aggravations de l'état de santé de patients âgés.

En effet, le système est basé sur la récupération de données vitales ou comportementales et les sauvegarde dans une base de données. Puis grâce à un module intelligent, il fait de l'aide au diagnostic [5].

1.7.5 **Synthèse sur les travaux existants**

Tout les projets cités ont comme objectif principale, la surveillance de l'état de santé du patient grâce à de différentes technologies de la télésurveillance médicale, afin de prévenir toute anomalie chez le patient.

Cependant, ces derniers ne permettent pas une consultation quotidienne et fréquente de l'état de santé du patient. De plus le contacte médecin / patient est inexistant.

Partant de cette idée nous avons proposé une architecture d'un système permettant aux patients ainsi qu'aux médecins traitants de :

- Consulter l'état de santé du patient à tout moment ;
- Consulter l'historique des mesures du patient ;

- Faciliter l'appel aux urgences ;
- La détection de toute anomalie chez le patient ;
- Permettre une communication directe entre le médecin et son patient ;
- Faciliter aux médecins la gestion de leurs patients ;

Dans ce qui suit, nous allons détailler l'architecture proposée pour ensuite passer à son implémentation.

1.8 Proposition d'une architecture globale pour la télésurveillance à distance

L'objectif principal de notre travail est de proposer un système de surveillance médicale à distance pour la détection des symptômes du Covid-19 (Respiration rapide, pneumonie, Fièvre, etc.) qui est notre cas d'étude. Le système doit permettre d'identifier l'ensemble des données du patient, son état actuel, son profil, ses antécédents, les données physiologiques et comportementales ainsi que les données de son environnement ainsi de permettre à un personnel médicale de détecter une hospitalisation urgente.

La figure 1.2 représente l'architecture globale du système de surveillance à distance :

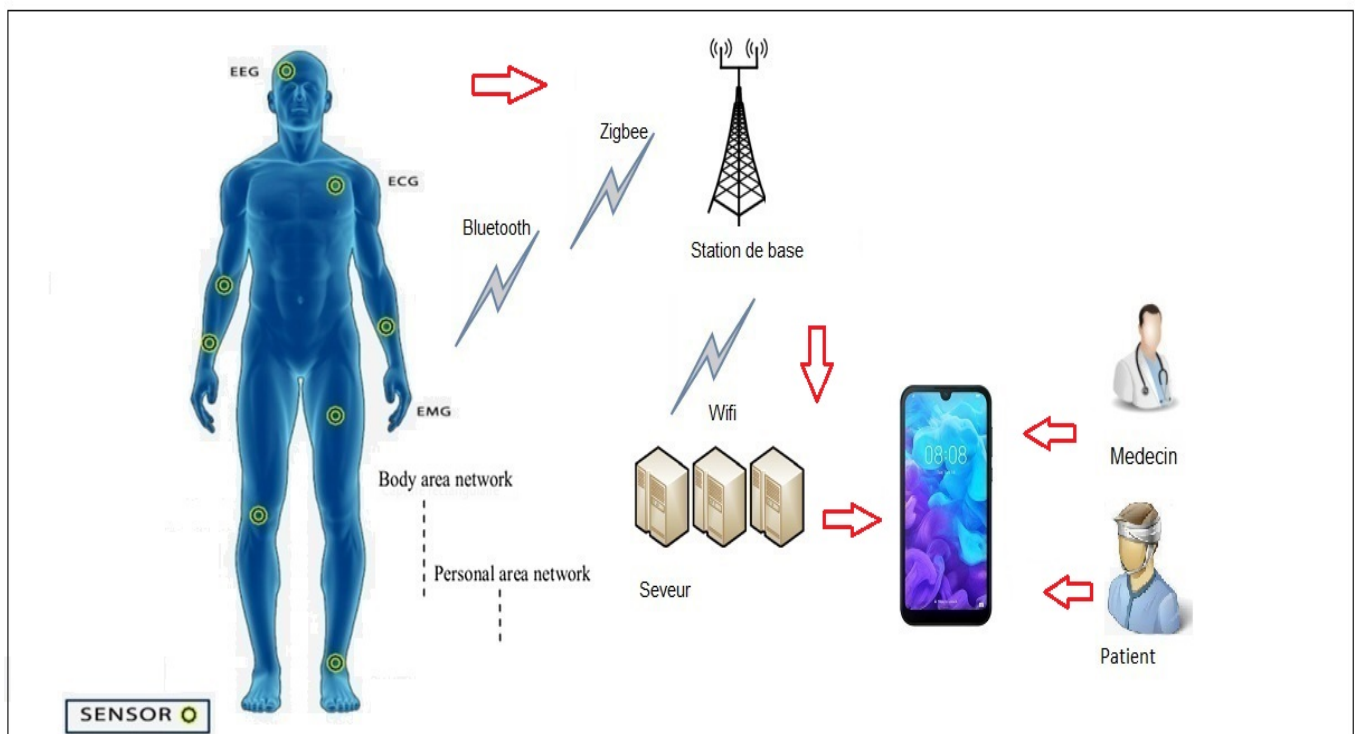


FIGURE 1.2 – Architecture globale du système de surveillance à distance

Dans ce qui suit, nous présentons les différentes étapes de notre système de surveillance médicale

à distance qui consiste en une chaîne de traitement de l'information. La figure 1.3 schématise les différentes étapes du système de surveillance à distance :

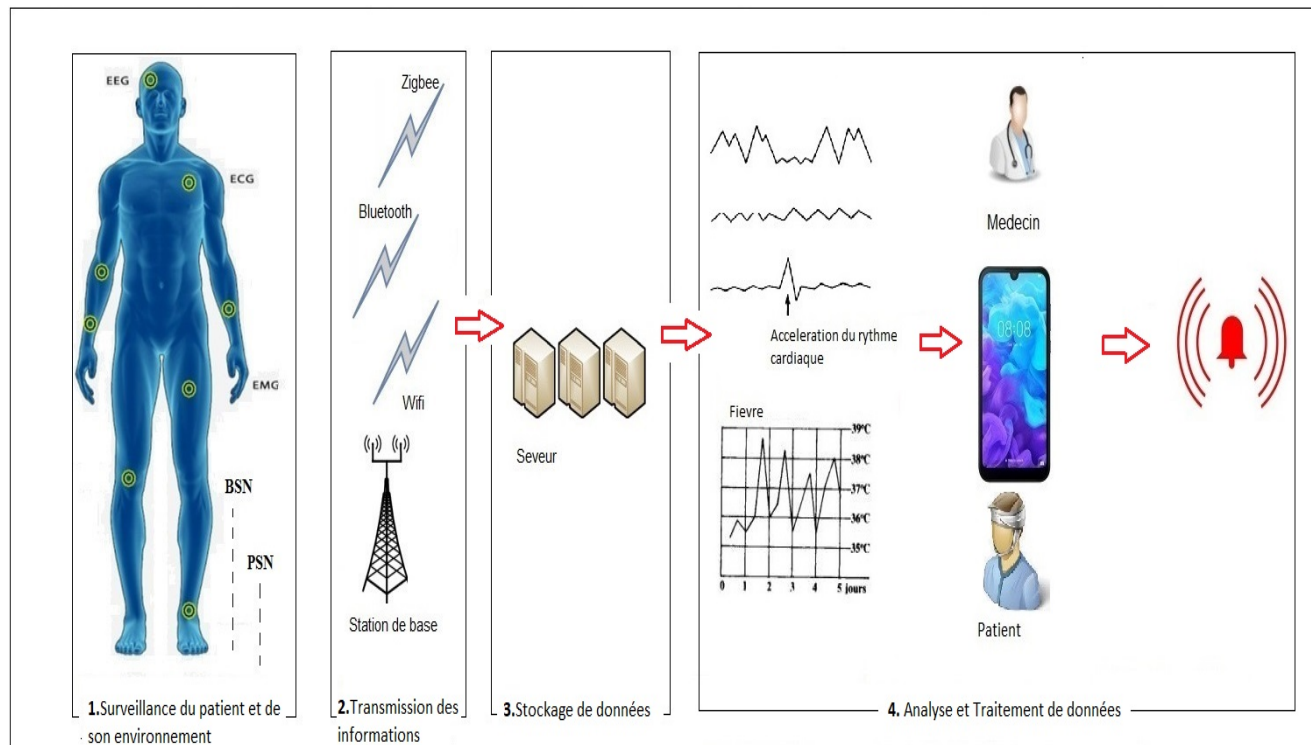


FIGURE 1.3 – Schéma résumant les différentes étapes de notre système de surveillance à distance

— **Etape 1 : Surveillance du patient et de son environnement**

Cette étape a pour objectif principal de collecter les données concernant l'état (évolutif) du patient et de son environnement et ceci grâce aux différents types de capteurs installés dans son appartement (Personal Area Network) ou des capteurs portés par le patient à domicile (Body Area Network).

— **Etape 2 : Transmission des informations**

Cette étape a pour but de transmettre les données enregistrées et collectées par les différents capteurs vers une station de base ensuite vers un serveur grâce à de différentes technologies de communication (zigbee, bluetooth, wifi, etc).

— **Etape 3 : Stockage des données**

Une fois que les données sont envoyées par les capteurs et la station de base, ces dernières sont enregistrées dans un serveur dédié pour être par la suite exploitées par le medecin et le patient.

— **Etape 4 : Analyse et traitement de données**

Cette étape consiste à analyser et traiter les données enregistrées dans le serveur. Ces données se composent des signaux envoyés par des capteurs en temps réel et de leur évolution

temporelle, ainsi que d'autres données connues du système (dossier médical du patient). Il s'agit de les fusionner afin d'avoir à tout instant une représentation de l'état global de la personne à domicile en terme médical. L'objectif est la détection tout événement critique pour le patient (Fièvre aiguë, difficulté respiratoire par exemple). Il s'agit également d'identifier toute détérioration de l'état de la personne sur un plus long terme, pouvant traduire la nécessité de soins ou bien d'une hospitalisation.

Les données médicales (fièvre, rythme cardiaque par exemple) obtenues sont affichées sous forme de courbes pour représenter l'état du patient. Le médecin traitant dispose ainsi à distance des données lui permettant d'émettre un diagnostic sur l'état de la personne à domicile. En cas de détection d'une anomalie (Fièvre élevée ou rythme cardiaque insuffisant par exemple) chez le patient une notification d'alerte est affichée et un appel d'urgence est effectué.

1.9 Présentation du cas d'étude : CORONAVIRUS

1. Qu'est-ce qu'un coronavirus ?

Les coronavirus forment une vaste famille de virus qui peuvent être pathogènes chez l'homme et chez l'animal. On sait que, chez l'être humain, plusieurs coronavirus peuvent entraîner des infections respiratoires dont les manifestations vont du simple rhume à des maladies plus graves comme le syndrome respiratoire du Moyen-Orient (MERS) et le syndrome respiratoire aigu sévère (SRAS). Le dernier coronavirus qui a été découvert est responsable de la maladie coronavirus 2019 (COVID-19) [6].

2. Qu'est-ce que la COVID-19 ?

La COVID-19 est la maladie infectieuse causée par le dernier coronavirus qui a été découvert. Ce nouveau virus et cette maladie étaient inconnus avant l'apparition de la flambée à Wuhan (Chine) en décembre 2019 [6].

3. Quels sont les symptômes de la COVID-19 ?

Ainsi que le notait l'OMS (Organisation Mondiale de la Santé) fin février dans son rapport d'observation de la maladie en Chine, la plupart des signes cliniques sont non spécifiques, c'est-à-dire qu'ils peuvent recouper ceux observés avec certains états grippaux (Figure 1.3) Un malade ne va pas forcément tous les présenter systématiquement. L'OMS énumère les symptômes suivants, par ordre de fréquence : fièvre (88%), toux sèche (67%), fatigue (38 %), production de mucus au niveau des poumons (34%), essoufflement (19), douleurs musculaires ou articulaires (15%), mal de gorge (14%), maux de têtes (14%), frissons (11,4%), nausées (5%), et enfin congestion nasale (5%, ce qui est très faible par rapport au rhume classique) et diarrhée (4%) [6].

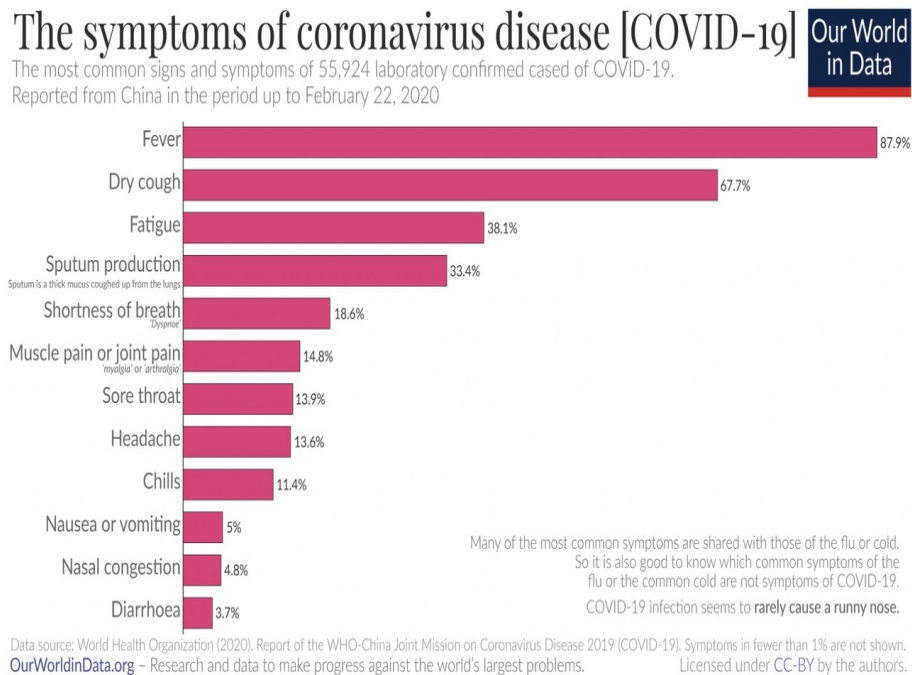


FIGURE 1.4 – Symptômes du COVID-19 [6]

Les symptômes précédents sont peu graves, et à eux seuls, ne justifient pas une prise en charge hospitalière. Les signes de gravité sont jugés par les médecins en fonction de plusieurs critères [6] :

- Pneumonie grave (la pneumonie légère n'étant pas forcément un facteur d'hospitalisation).
- Baisse de la saturation de l'oxygène dans le sang (hypoxémie).
- Respiration rapide (plus de 20 par minute).
- Existence d'autres pathologies chroniques risquant de décompenser (par exemple diabète).
- Ou encore évolution clinique rapide et défavorable.

L'hospitalisation permet dans ce cadre de surveiller l'état général des patients, éventuellement de leur fournir de l'oxygène à l'aide d'un masque en cas d'hypoxémie. Cet état peut brutalement basculer en syndrome de détresse respiratoire aigu, qui peut alors nécessiter une assistance respiratoire urgente et une intubation.

4. Qui risque d'être atteint d'une forme grave de la maladie ?

Même si nous devons encore approfondir nos connaissances sur la façon dont le COVID-19 affecte les individus, jusqu'à présent, les personnes âgées et les personnes déjà atteintes d'autres maladies (comme l'hypertension artérielle, les maladies pulmonaires, le cancer, le diabète ou les cardiopathies) semblent être gravement atteintes plus souvent que les autres. Le risque de mortalité serait nettement plus élevé chez les patients âgés de plus de 70 ans, probablement car nombre d'entre eux ont des problèmes de santé préexistants. Les patients

atteints de coronavirus et de maladies cardiaques, par exemple, auraient un taux de mortalité d'environ 10%, selon l'étude, tandis que ceux atteints de diabète auraient un taux de mortalité d'environ 7%. Environ trois quarts des patients n'avaient pas de problèmes de santé préexistants, le taux de mortalité pour ce groupe serait légèrement inférieur à 1% [6].

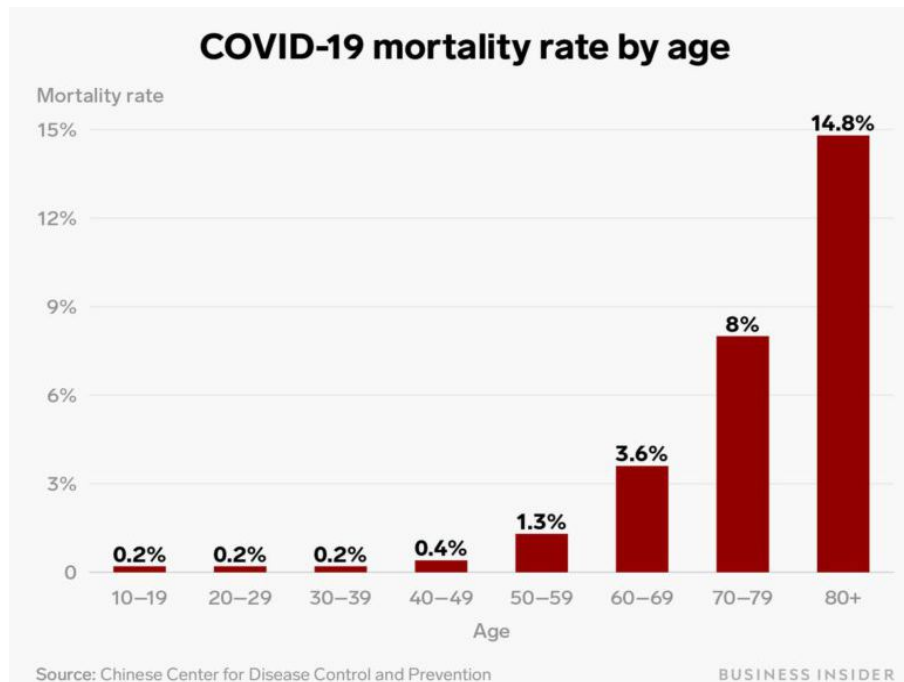


FIGURE 1.5 – Taux de mortalité selon l'âge [6]

5. Comparaison entre le covid et les autres grippes :

L'Organisation mondiale de la santé a confirmé mardi 3 mars que le taux de mortalité pour le nouveau coronavirus était de 3,4% au niveau mondial. En comparaison, la grippe saisonnière tue moins de 1% des personnes infectées. "Le Covid-19 se propage moins efficacement que la grippe, sa transmission ne semble pas être provoquée par des personnes qui ne sont pas malades, il provoque une maladie plus grave que la grippe, il n'y a pas encore de vaccins ni de traitement et il peut être contenu", selon Tedros Adhanom Ghebreyesu, le directeur général de l'OMS, lors d'une conférence de presse, rapporte Reuters [6].

2019 novel coronavirus compared to other major viruses

VIRUS	YEAR IDENTIFIED	CASES	DEATHS	FATALITY RATE	NUMBER OF COUNTRIES
Ebola**	1976	33,577	13,562	40.4%	9
Nipah	1998	513	398	77.6%	2
SARS	2002	8,096	774	9.6%	29
MERS*	2012	2,494	858	34.4%	28
COVID-19**	2020	82,548	2,810	3.4%	45

Sources: Johns Hopkins, CDC, World Health Organization, New England Journal of Medicine, Malaysian Journal of Pathology, CGTN

*As of November 2019 **As of February 27, 2020

BUSINESS INSIDER

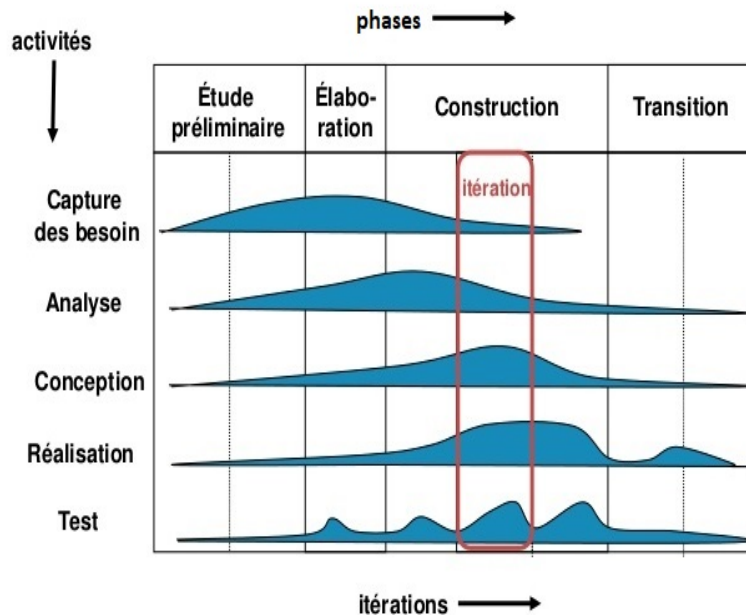
FIGURE 1.6 – Comparaison entre le covid-19 et autres gripes [6]

1.10 Démarche de développement

Afin de concevoir et de réaliser notre application, nous avons opté pour la démarche de développement UP (Unified Process). Notre choix s'est posé sur ce dernier car c'est un processus de développement moderne, itératif, efficace sur des projets informatiques de toutes tailles. Très complet, il couvre l'ensemble des activités, depuis la conception du projet jusqu'à la livraison de la solution. Intégrant une organisation de projet type, une méthodologie utilisant UML et un ensemble de bonnes pratiques cohérentes entre elles, il permet de circonvier aux problèmes récurrents que rencontrent nombre de réalisations : dérive des coûts et des délais, qualité insuffisante, réponse incomplète aux attentes des utilisateurs.

1.10.1 Processus Unifié-UP

Le processus unifié est un processus de développement logiciel construit sur UML. Il est itératif, centré sur l'architecture, piloté par des cas d'utilisation et orienté vers la diminution des risques. Il regroupe les activités à mener pour transformer les besoins d'un utilisateur en système logiciel. C'est un patron de processus pouvant être adapté à une large classe de systèmes logiciels, à différents domaines d'application, à différents types d'entreprises, à différents niveaux de compétences et à différentes tailles de l'entreprise [7].



2011-2012 / Yannick Prié - Université Claude Bernard Lyon 1

FIGURE 1.7 – Processus Unifié [14]

1.10.2 Caractéristiques du processus unifié

- UP est piloté par les cas d'utilisation :** L'objectif principal d'un système logiciel est de rendre service à ses utilisateurs ; il faut par conséquent bien comprendre les désirs et les besoins des futurs utilisateurs. Le processus de développement sera donc centré sur l'utilisateur. Le terme utilisateur ne désigne pas seulement les utilisateurs humains mais également les autres systèmes. L'utilisateur représente donc une personne ou une chose dialoguant avec le système en cours de développement. Les cas d'utilisation permettent d'illustrer les besoins. Ils détectent puis décrivent les besoins fonctionnels (du point de vue de l'utilisateur), et leur ensemble constitue le modèle de cas d'utilisation qui dicte les fonctionnalités complètes du système [7].
- UP est centré sur l'architecture :** Dès le démarrage du processus, on aura une vue sur l'architecture à mettre en place. L'architecture d'un système logiciel peut être décrite comme les différentes vues du système qui doit être construit. L'architecture logicielle équivaut aux aspects statiques et dynamiques les plus significatifs du système. L'architecture émerge des besoins de l'entreprise, tels qu'ils sont exprimés par les utilisateurs et autres intervenants et tels qu'ils sont reflétés par les cas d'utilisation [7].
- UP est itératif et incrémental :** L'itération est une répétition d'une séquence d'instructions ou d'une partie de programme un nombre de fois fixé à l'avance ou tant qu'une condition définie n'est pas remplie, dans le but de reprendre un traitement sur des données différentes. Elle qualifie un traitement

ou une procédure qui exécute un groupe d'opérations de façon répétitive jusqu'à ce qu'une condition bien définie soit remplie. Une itération prend en compte un certain nombre de cas d'utilisation et traite en priorité les risques majeurs. Une itération désigne donc la succession des étapes de l'enchaînement d'activités, tandis qu'un incrément correspond à une avancée dans les différents stades de développement [7].

4. **UP est piloté par la gestion des risques :**

La gestion des risques consiste à prévoir ou à anticiper les situations risquées et mettre en œuvre un plan d'actions composé :

- Actions de réduction : réduire l'influence des facteurs de risques (réduire la probabilité)
Les technologies utilisées sont bien connues des développeurs, ce qui devrait réduire les risques de non-aboutissement.
- Actions préventives : ne pas se mettre dans une situation (actions sur le déclenchement des facteurs de risques) L'intervention d'un ergonome réduira les risques de non-acceptance.
- Actions de couverture : limiter les conséquences des risques (réduire la gravité) Un développement itératif, incrémental permettra de limiter la prise de risque technologique
- Attitude de NO GO : le projet est abandonné [7].

1.10.3 Activités et Phases

L'objectif d'un processus unifié est de maîtriser la complexité des projets informatiques en diminuant les risques. UP gère le développement selon deux axes qu'on peut décrire comme suit :

- L'axe vertical représente les principaux enchaînements d'activités, qui regroupent les activités selon leur nature. Cette dimension rend compte l'aspect statique du processus qui s'exprime en terme décomposant, de processus, d'activités, d'enchaînements, d'artefacts et de travailleurs ;
- L'axe horizontal représente le temps et montre le déroulement du cycle de vie du processus ; cette dimension rend compte de l'aspect dynamique du processus qui s'exprime en terme de cycles, de phases, d'itérations et de jalons [7].

1.10.4 Les activités

1. **Capture des besoins :**

L'expression des besoins permet de :

- inventorier les besoins principaux et fournir une liste de leurs fonctions ;
- recenser les besoins fonctionnels (du point de vue de l'utilisateur) qui conduisent à l'élaboration des modèles de cas d'utilisation ;
- appréhender les besoins non fonctionnels (technique) et livrer une liste des exigences. Le modèle de cas d'utilisation présente le système du point de vue de l'utilisateur et représente sous forme de cas d'utilisation et d'acteur, les besoins du client [7].

2. **Analyse :**

L'objectif de l'analyse est d'accéder à une compréhension des besoins et des exigences du client. Il s'agit de réaliser des spécifications permettant de concevoir la solution. Un modèle d'analyse livre une spécification complète des besoins issus des cas d'utilisation et les structures sous une forme qui facilite la compréhension (scénarios), la préparation (définition de l'architecture), la modification et la maintenance du futur système. Il peut être considéré comme une première ébauche du modèle de conception [7].

3. **Conception :**

La conception permet d'acquérir une compréhension approfondie des contraintes liées au langage de programmation, à l'utilisation des composants et au système d'exploitation. Elle détermine les principales interfaces et les transcrit à l'aide d'une notation commune. Elle constitue un point de départ à l'implémentation [7] :

- elle décompose le travail d'implémentation en sous-système ;
- elle crée une abstraction transparente de l'implémentation.

4. **Implémentation :**

L'implémentation est le résultat de la conception pour implémenter le système sous forme décomposant, c'est-à-dire, de code source, de scripts, de binaires, d'exécutables et d'autres éléments du même type. Les objectifs principaux de l'implémentation sont de planifier les intégrations des composants pour chaque itération, et de produire les classes et les sous-systèmes sous formes de codes sources [7].

5. **Test :**

Les tests permettent de vérifier des résultats de l'implémentation en testant la construction. Pour mener à bien ces tests, il faut les planifier pour chaque itération, les implémenter en créant des cas de tests, effectuer ces tests et prendre en compte le résultat de chacun [7].

1.10.5 Les Phases

1. **Analyse des besoins :**

L'analyse des besoins donne une vue du projet sous forme de produit fini. Cette phase porte essentiellement sur les besoins principaux (du point de vue de l'utilisateur), l'architecture générale du système, les risques majeurs, les délais et les coûts (On met en place le projet). Elle répond aux questions suivantes [7] :

- que va faire le système ? Par rapport aux utilisateurs principaux, quels services va-t-il rendre ?
- quelle va être l'architecture générale (cible) de ce système ?
- quels vont être : les délais, les coûts, les ressources, les moyens à déployer ?

2. **Elaboration :**

L'élaboration reprend les éléments de la phase d'analyse des besoins et les précise pour arriver à une spécification détaillée de la solution à mettre en œuvre. L'élaboration permet de

préciser la plupart des cas d'utilisation, de concevoir l'architecture du système et surtout de déterminer l'architecture de référence. Au terme de cette phase, les chefs de projet doivent être en mesure de prévoir les activités et d'estimer les ressources nécessaires à l'achèvement du projet. Les tâches à effectuer dans la phase élaboration sont les suivantes [7] :

- créer une architecture de référence ;
- identifier les risques, ceux qui sont de nature à bouleverser le plan, le coût et le calendrier ;
- définir les niveaux de qualité à atteindre ;
- formuler les cas d'utilisation pour couvrir les besoins fonctionnels et planifier la phase de construction ;
- élaborer une offre abordant les questions de calendrier, de personnel et de budget [7].

3. **Construction :**

La construction est le moment où l'on construit le produit (architecture= produit complet). Le produit contient tous les cas d'utilisation que les chefs de projet, en accord avec les utilisateurs ont décidé de mettre au point pour cette version [7].

4. **Transition :**

Un groupe d'utilisateurs essaye le produit et détecte les anomalies et défauts. Cette phase suppose des activités comme la formation des utilisateurs clients, la mise en œuvre d'un service d'assistance et la correction des anomalies constatées [7].

1.11 Langage de modélisation

Afin d'optimiser la compréhension ainsi que la réalisation du projet on a choisi comme langage de modélisation l'UML (Unified Modeling Language).

UML est un langage de modélisation graphique destiné à visualiser, analyser, spécifier, construire des logiciels orientés objets.

UML est aujourd'hui considéré comme un standard autant dans le milieu industriel qu'académique. Il propose un ensemble de diagrammes afin de couvrir l'ensemble des besoins de modélisation potentiellement nécessaires à la conception des logiciels, ce qui le rend relativement complet et générique.

Ainsi, au travers des 14 types de diagrammes (figure 1.8), UML permet de modéliser les aspects statiques et dynamiques des systèmes complexes et de couvrir la plupart des phases du développement logiciel (analyse, conception, implantation, déploiement, etc.) [8].

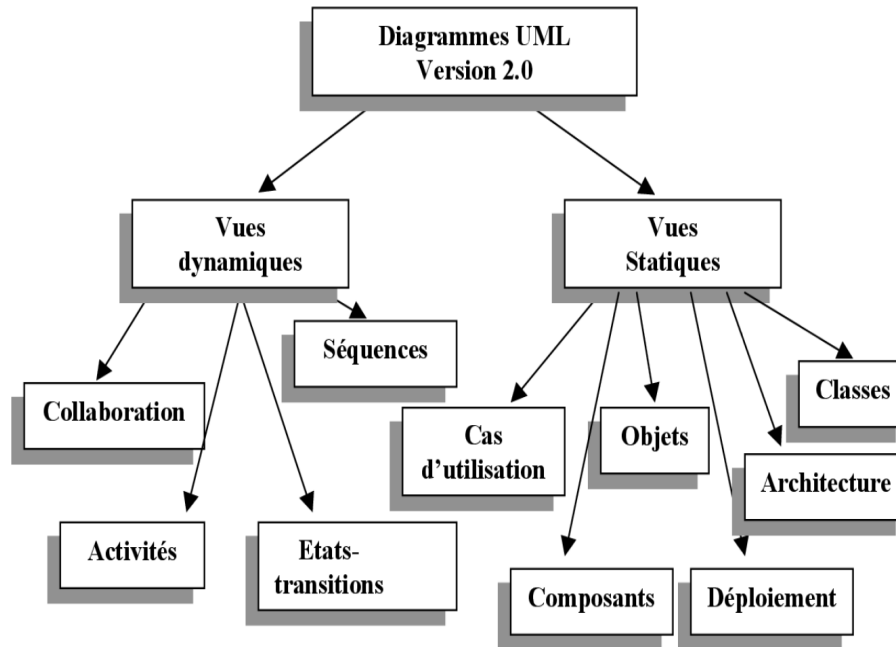


FIGURE 1.8 – Les différents diagrammes UML [15]

1. Vues statiques :

- Les diagrammes de cas d'utilisation décrivent le comportement et les fonctions d'un système du point de vue de l'utilisateur ;
- Les diagrammes de classes décrivent la structure statique, les types et les relations des ensembles d'objets ;
- Les diagrammes d'objets décrivent les objets d'un système et leurs relations ;
- Les diagrammes de composants décrivent les composants physiques et l'architecture interne d'un logiciel ;
- Les diagrammes de déploiement décrivent la répartition des programmes exécutables sur les différents matériels.

2. Vues dynamiques :

- Les diagrammes de collaboration décrivent les messages entre objets (liens et interactions) ;
- Les diagrammes d'états-transitions décrivent les différents états d'un objet ;
- Les diagrammes d'activités décrivent les comportements d'une opération (en termes d'actions) ;
- Les diagrammes de séquence décrivent de manière temporelle les interactions entre objets et acteur.

Pour la conception de notre application, nous avons choisi les diagrammes suivants :

- diagramme de cas d'utilisation ;

- diagramme de classe ;
- diagramme de séquence.

1.12 Conclusion

Dans ce chapitre, nous avons évoqué l'approche de la télésurveillance médicale ses avantages ainsi que ses inconvénients. Ensuite, nous avons cité quelques travaux existant dans ce domaine. Pour finir nous avons proposé une architecture pour une mise en œuvre d'un système de télésurveillance médicale.

Le prochain chapitre fera l'objet de la Spécification des besoins du système à mettre en œuvre.

Chapitre 2

Spécification des Besoins

2.1 Introduction

La spécification des besoins est la première étape du cycle de vie de développement d'un projet, c'est une étape primordiale car elle permet de mieux comprendre le travail demandé en dégagant les besoins des utilisateurs que le système doit accomplir.

Dans ce chapitre, nous présentons en les besoins fonctionnels et non fonctionnels, les différents acteurs qui interagissent avec notre système ainsi que le diagramme de cas d'utilisation et les maquettes qui donnent une vue globale du système. Enfin, nous terminons ce chapitre par une conclusion.

2.2 Expression des besoins

Le but d'un projet est de satisfaire un besoin. Il faut l'exprimer clairement avant d'imposer une solution généralement formulé sous formes d'exigences fonctionnelles et non fonctionnelles.

La suite de cette section du chapitre définit les besoins fonctionnels et non fonctionnels de notre application mobile.

2.2.1 Besoins fonctionnels

Les besoins fonctionnels représentent les principales fonctionnalités du système dont l'utilisateur ne peut s'en passer.

Pour la réalisation de notre application mobile, nous avons extrait les besoins fonctionnels suivants :

1. Consulter l'espace patient ;
2. Consulter l'état du patient ;
3. Consulter l'évolution de l'état du patient ;
4. Consulter le profile du patient ;
5. Gérer les patients (ajout, suppression, recherche) ;
6. Générer les rapports pour chaque patient ;
7. Envoyer un rapport ;
8. Générer une notification d'alerte en cas d'anomalie ;
9. Envoyer des interventions aux patients ;

2.2.2 Besoins non fonctionnels

Ce sont des besoins en relation avec la performance du système en temps de réponse et de stockage mémoire, la sécurité, la facilité d'utilisation et l'ergonomie de l'interface graphique. Pour notre application mobile, on s'est basé sur les points suivants :

1. Permettre un accès sécurisé à l'application via une authentification ;

2. Confidentialité des données ;
3. Fiabilité : probabilité de n'avoir aucune défaillance pendant la durée ;
4. Ergonomie : améliorer les interactions Homme-Machine, la facilité d'utilisation et d'apprentissage des produits interactifs ;

2.3 Identification des acteurs

Un acteur représente l'abstraction d'un rôle joué par une entité externe, il peut être un utilisateur humain ou un dispositif matériel ou autre system connexe. Il interagit directement avec le système en consultant ou en modifiant son état. Il existe quatre types d'acteurs :

1. Les acteurs principaux : Ce sont les acteurs qui vont réaliser le cas d'utilisation.
2. Les acteurs secondaires : Ce sont ceux qui font que recevoir des informations à l'issue de la réalisation d'un cas d'utilisation ;
3. Périphériques externes : Les dispositifs matériels qui font partie du domaine de l'application et qui doivent absolument être utilisés. Exemple : capteur, horloge externe, etc ;
4. Système externe : Les systèmes avec lesquels le système interagit.

Dans notre cas, nous avons défini cinq acteurs suivants : patient, médecin, capteur, serveur de base de données et l'application.

Le tableau 2.1 résume les différents types d'acteurs qui interagissent avec le système ainsi que leurs rôles.

Acteur	Type	Rôle
Patient	Acteur principale	-Inscription ; -Authentification ; -Accéder à son espace ; -Consulter ses mesures ; -Consulter l'historique de ses mesures.
Médecin	Acteur principale	-Inscription ; -Authentification ; -Accéder à son espace ; -Gérer patient (ajouter/supprimer/rechercher) ; -Consulter l'état du patient ; -Consulter les mesures du patient ; -Consulter l'historique des mesures du patient ; -Générer des rapports ; -Réception des alertes ; -Envoyer des interventions.
Capteur	Périphérique externe	-Collecter les données ; -Envoyer les données.
Serveur BDD	Système externe	Stoker les données
Application	Système externe	-Traitement de données ; -Générer les alertes.

TABLE 2.1 – Identification des acteurs du système

2.4 Identification des cas d'utilisation

Un cas d'utilisation est une unité cohérente représentant une fonctionnalité visible de l'extérieur. Il réalise un service de bout en bout, avec un déclenchement, un déroulement et une fin, pour l'acteur qui l'initie. Un cas d'utilisation modélise donc un service rendu par le système, sans imposer le mode de réalisation de ce service [8].

Le diagramme de cas d'utilisation est un diagramme UML utilisé pour donner une vision globale du comportement fonctionnel d'un système logiciel, il permet de représenter graphiquement les cas d'utilisation.

Le tableau 2.2 ci-dessous représente l'identification des différents cas d'utilisation :

Numéro cas d'utilisation	Cas d'utilisation	Acteurs
1	Authentification	Médecin/Patient
2	Inscription	Médecin/Patient
3	Consulter l'espace patient	Patient
4	Consulter le profil patient	Medecin
5	Consulter les mesures du patient	Patient/Medecin
6	Consulter historique des mesures	Médecin/Patient
7	Rédiger des rapports	Médecin
8	Envoyer des rapports	Médecin
9	Gérer les patients	Médecin
10	Envoyer une intervention	Médecin
11	Recevoir alerte	Médecin/Patient
12	Traitement de données	Application
13	Générer notification d'alerte	Application
14	Collecter données	Capteurs
15	Envoyer données	Capteurs
16	Stocker données	Serveur BDD
17	Recevoir une Alerte	Medecin/Patient
18	Envoyer une intervention	Medecin

TABLE 2.2 – Identification des cas d'utilisation du système

2.5 Diagramme de cas d'utilisation

Bien souvent, la maîtrise d'ouvrage et les utilisateurs ne sont pas des informaticiens. Il leur faut donc un moyen simple d'exprimer leurs besoins. C'est précisément le rôle des diagrammes de cas d'utilisation qui permettent de recueillir, d'analyser et d'organiser les besoins, et de recenser les grandes fonctionnalités d'un système. Il s'agit donc de la première étape UML d'analyse d'un système.

Un diagramme de cas d'utilisation capture le comportement d'un système, d'un sous-système, d'une classe ou d'un composant tel qu'un utilisateur extérieur le voit. Il scinde la fonctionnalité du système en unités cohérentes (les cas d'utilisation) ayant un sens pour les acteurs. Les cas d'utilisation permettent d'exprimer le besoin des utilisateurs d'un système, ils sont donc une vision orientée utilisateur de ce besoin au contraire d'une vision informatique.

Il ne faut pas négliger cette première étape pour produire un logiciel conforme aux attentes des utilisateurs. Pour élaborer les cas d'utilisation, il faut se fonder sur des entretiens avec les utilisateurs [8].

2.5.1 Diagramme de cas d'utilisation de notre application mobile

La figure 2.1 représente le diagramme de cas d'utilisation global de notre système.

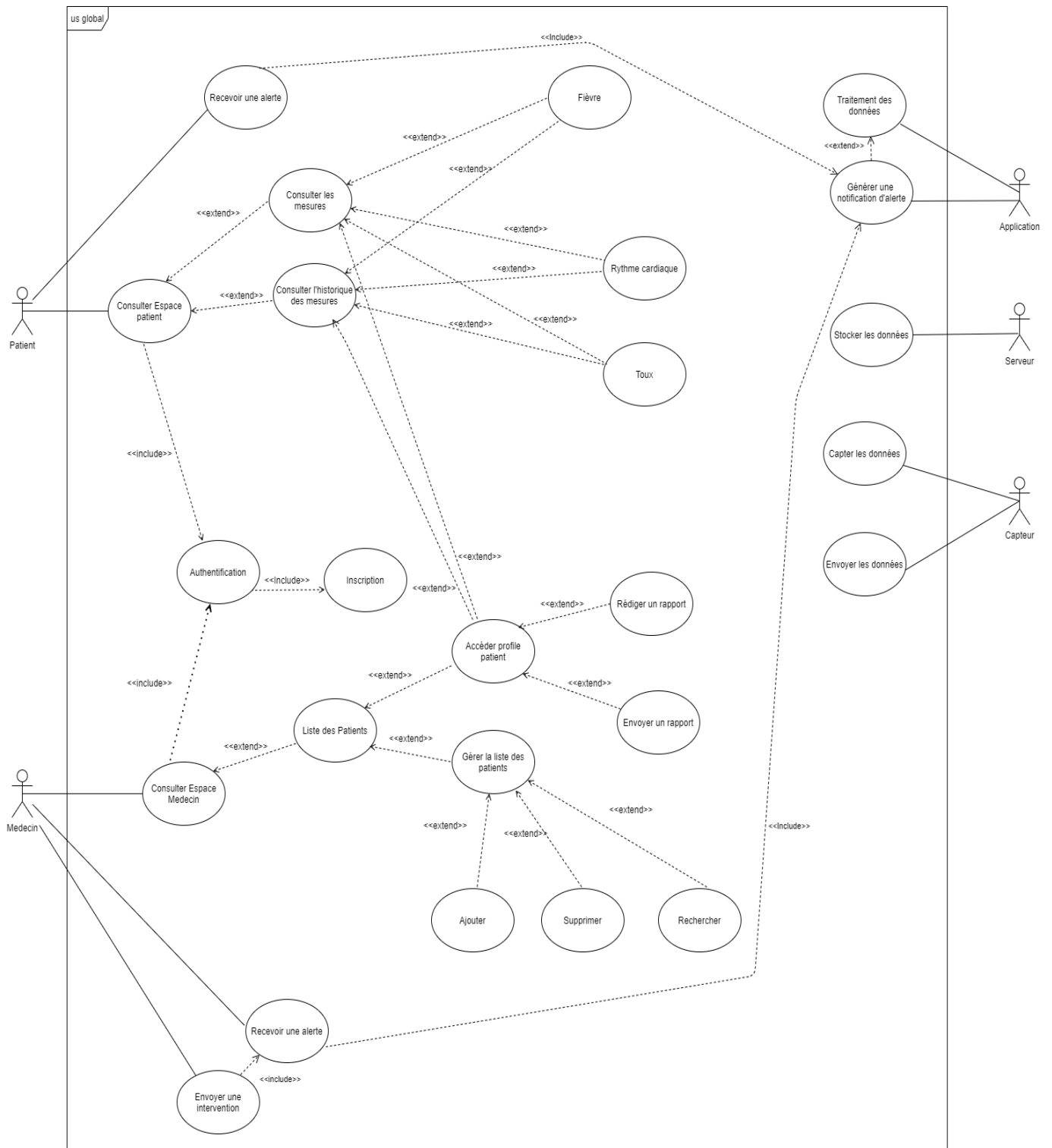


FIGURE 2.1 – Diagramme de cas d'utilisation de notre application

2.5.2 Diagrammes de cas d'utilisation détaillés

Dans ce qui suit, nous allons détailler le diagramme de cas d'utilisation général de notre système. De plus, nous allons donner leurs descriptions textuelles.

1. Diagramme de cas d'utilisation : « Inscription »

La figure 2.2 représente le diagramme de cas d'utilisation « Inscription »

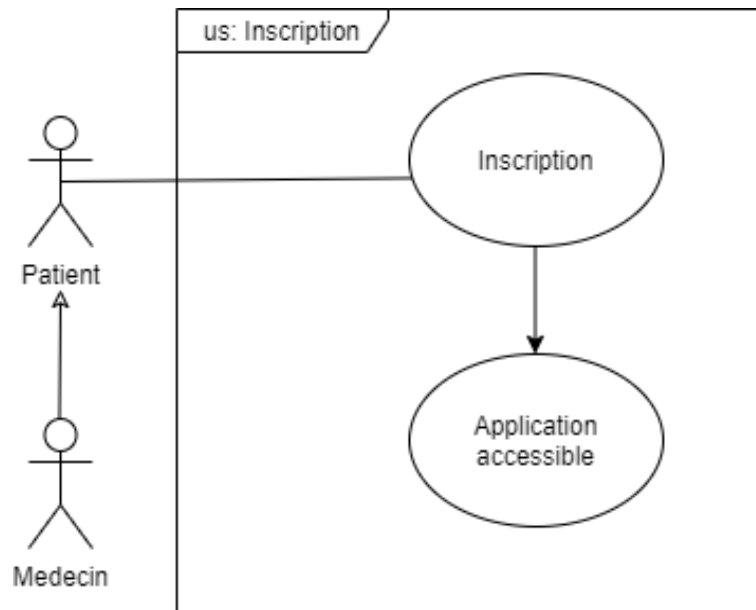


FIGURE 2.2 – Diagramme de cas d'utilisation « Inscription »

La description textuelle du cas d'utilisation « Inscription » est représentée par le tableau 2.3 ci-dessous :

Cas d'utilisation Inscription	
Titre	Inscription
Résumé	L'Inscription permet un accès contrôlé et sécurisé au système.
Acteurs	Médecin, Patient.
Description des scénarios	
Pré-condition	Application installée.
Scénario Nominal	-L'utilisateur introduit ses coordonnées et les valide ; -Le système vérifie la validité des données saisies ; -Le système affiche l'espace qui correspond à l'utilisateur.
Enchaînement d'erreurs	-Utilisateur existe déjà ; -Données introduits erronées.
Post-condition	Inscription réussie.

TABLE 2.3 – Description textuelle du cas d'utilisation « Inscription »

2. Diagramme de cas d'utilisation : « Authentification »

La figure 2.3 représente le diagramme de cas d'utilisation « Authentification »

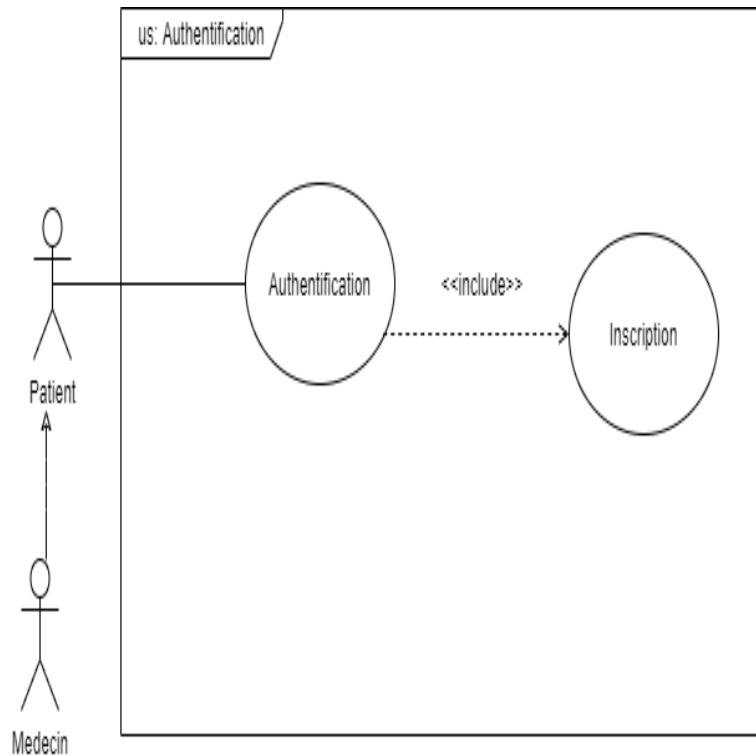


FIGURE 2.3 – Diagramme de cas d'utilisation « Authentification »

La description textuelle du cas d'utilisation « Authentification » est représentée par le tableau 2.4 ci-dessous :

Cas d'utilisation Authentification	
Titre	Authentification.
Résumé	L'authentification permet un accès contrôlé et sécurisé au système, de tel sorte que chaque acteur accède directement après son authentification à l'espace qui lui est dédié.
Acteurs	Médecin, Patient.
Description des scénarios	
Pré-condition	Inscriptions.
Scénario Nominal	-L'utilisateur introduit son login ainsi que son mot de passe et valide ; -Le système vérifie l'existence de l'utilisateur et la validité du mot de passe ; -Le système affiche la fenêtre qui correspond à l'utilisateur.
Enchaînement d'erreurs	Aucun utilisateur ne correspond à l'identifiant et au mot de passe saisis, le system renvoi un message d'erreur.
Post-condition	Connexion réussie.

TABLE 2.4 – Description textuelle du cas d'utilisation « Authentification »

3. Diagramme de cas d'utilisation : « Consulter l'espace du patient »

La figure 2.4 représente le diagramme de cas d'utilisation « Consulter espace patient »

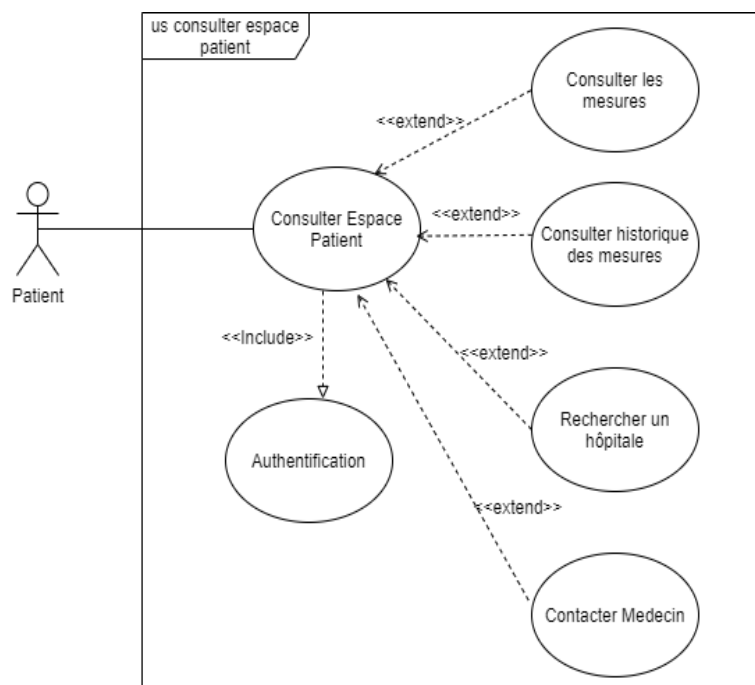


FIGURE 2.4 – Diagramme de cas d'utilisation « Consulter espace patient »

La description textuelle du cas d'utilisation « Consulter espace patient » est représentée par le tableau 2.5 ci-dessous :

Cas d'utilisation Consulter Espace du patient	
Titre	Consulter Espace-Patient
Résumé	La consultation de l'espace d'un patient permet de consulter et de contrôler son état de santé.
Acteurs	Patient.
Description des scénarios	
Pré-condition	Authentification réussie
Scénario Nominal	-Le patient s'authentifie et accède directement à son espace ;
Enchaînement d'erreurs	Erreur d'authentification.
Post-condition	Espace patient affiché

TABLE 2.5 – Description textuelle du cas d'utilisation « Consulter espace patient »

4. Diagramme de cas d'utilisation : « Consulter Mesures »

La figure 2.5 représente le diagramme de cas d'utilisation « Consulter les mesures »

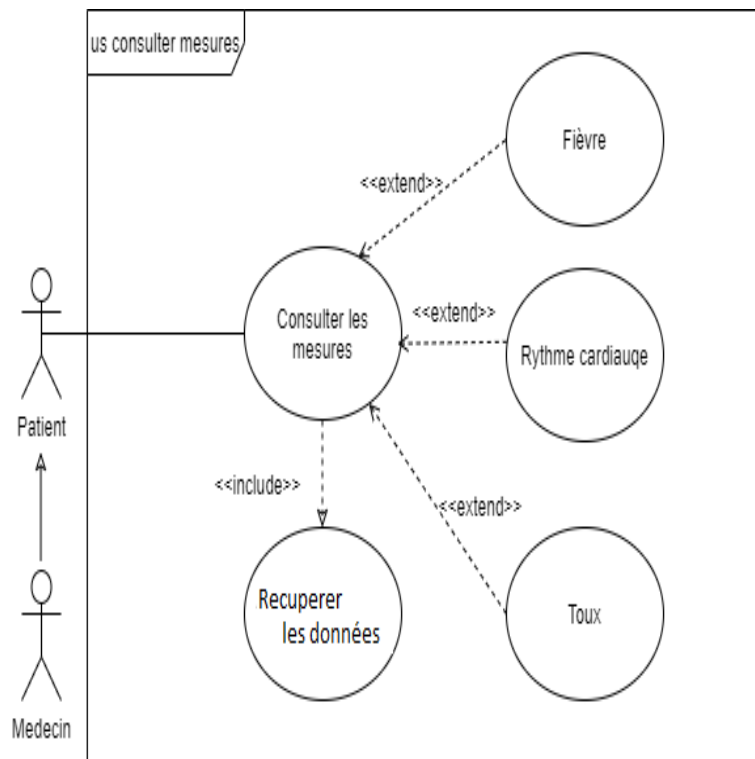


FIGURE 2.5 – Diagramme de cas d'utilisation « Consulter les Mesures »

La description textuelle du cas d'utilisation « Consulter les Mesures » est représentée par le tableau 2.6 ci-dessous :

Cas d'utilisation Consulter les Mesures	
Titre	Consulter Mesures
Résumé	Consulter les mesures du patient permet de consulter son état de santé.
Acteurs	Medecin/ Patient
Description des scénarios	
Pré-condition	Mesures récupérées.
Scénario Nominal	<ul style="list-style-type: none"> - Le Patient/Medecin accède à l'espace/profile du patient ; - Le Patient/Medecin accède à l'interface dédiée à ses mesures ; - Le Patient/Medecin choisi les mesures qu'il souhaite consulter ; - Le système affiche les valeurs enregistrées.
Enchainement d'erreurs	<ul style="list-style-type: none"> -Erreur d'authentification. -Données non enregistrées. -Données non reçues.
Post-condition	Mesures affichées.

TABLE 2.6 – Description textuelle du cas d'utilisation « Consulter les Mesures »

5. Diagramme de cas d'utilisation : « Consulter Historique »

La figure 2.6 représente le diagramme de cas d'utilisation « Consulter historique »

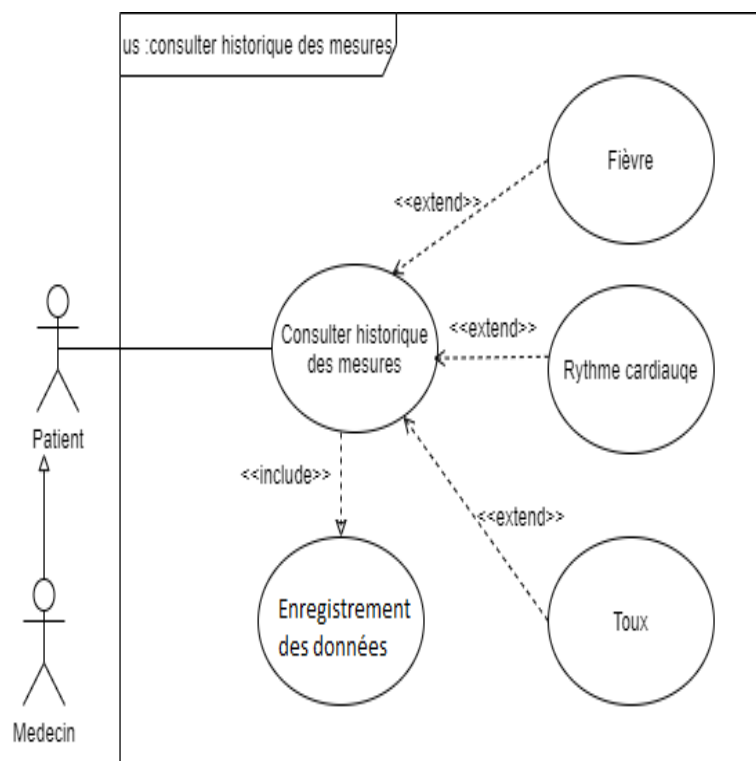


FIGURE 2.6 – Diagramme de cas d'utilisation « Consulter Historique »

La description textuelle du cas d'utilisation « Consulter Historique » est représentée par le tableau 2.7 ci-dessous :

Cas d'utilisation Consulter Historique	
Titre	Consulter Historique
Résumé	Consulter l'historique du patient permet de voir l'évolution de son état de santé.
Acteurs	Médecin, Patient
Description des scénarios	
Pré-condition	Mesures enregistrées.
Scénario Nominal	<ul style="list-style-type: none"> - Le Médecin/ Patient accède à l'espace du patient ; - Le Médecin/ Patient accède à l'interface dédiée à l'historique ; - Le Médecin/ Patient choisit l'historique des mesures qu'il souhaite consulter ; - Le système affiche les valeurs enregistrées.
Enchaînement d'erreurs	<ul style="list-style-type: none"> -Erreur d'authentification. -Données non enregistrées.
Post-condition	Historique affiché.

TABLE 2.7 – Description textuelle du cas d'utilisation « Consulter Historique »

6. Diagramme de cas d'utilisation : « Générer Alerte »

La figure 2.7 représente le diagramme de cas d'utilisation « Générer Alerte »

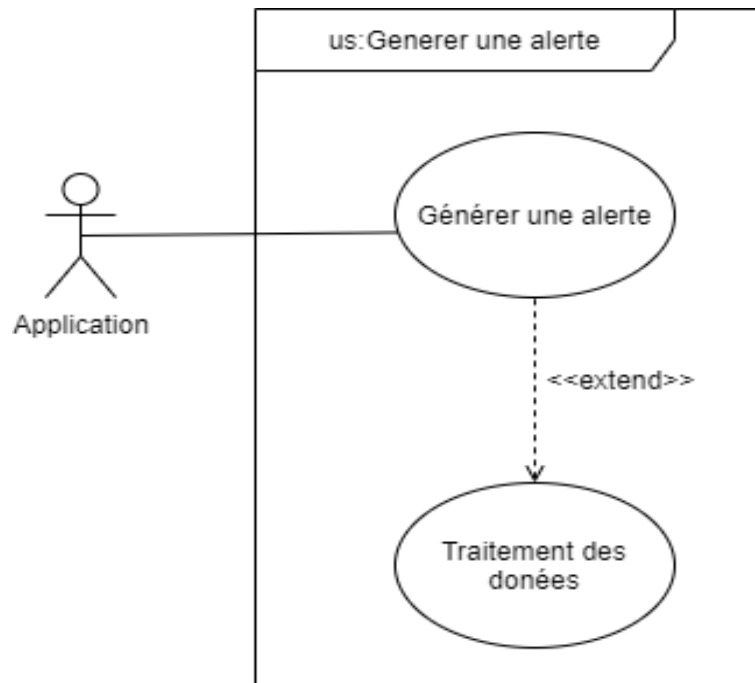


FIGURE 2.7 – Diagramme de cas d'utilisation « Générer Alerte »

La description textuelle du cas d'utilisation « Générer Alerte » est représentée par le tableau 2.8 ci-dessous :

Cas d'utilisation Générer Alerte	
Titre	Générer Alerte
Résumé	Gestion d'une alerte permet de prévenir le médecin et le patient d'une anomalie lors du traitement des données reçus par les capteurs ceci permettra d'éviter de lourdes conséquences chez le patient.
Acteurs	Application
Description des scénarios	
Pré-condition	Traitement des données.
Scénario Nominal	<ul style="list-style-type: none"> - Le système reçoit les données des capteurs ; - Le système traite ces données ; - Le système détecte une anomalie lors du traitement des données ; - Le système envoie une alerte (sous forme de notification) au patient et au médecin traitant.
Enchaînement d'erreurs	<ul style="list-style-type: none"> -Erreur d'authentification ; -Données du patients non disponibles.
Post-condition	Alerte générée.

TABLE 2.8 – Description textuelle du cas d'utilisation « Générer une Alerte »

Algorithme décrivant le processus d'alerte :

```

Algorithme Alerte;
Entrées : Pss, ValeurTemp, EtatSante;
Sorties: Alerte generée;

Debut
  TQ surveillance Active faire
    ValeurTemp ← valeur lue à partir d'un capteur-temperature ;
    Si (ValeurTemp > 40 ) ou (ValeurTemp < 36 ) alors
      -EtatSanté ← "anormale";
      -Envoyer notification;
    Sinon
      -EtatSanté ← " normale";
      -Attendre la prochaine lecture de temperature;
    Finsi;
  FinTQ;
Fin.

```


7. Diagramme de cas d'utilisation : « Gérer Patient »

La figure 2.8 représente le diagramme de cas d'utilisation « Gérer Patient »

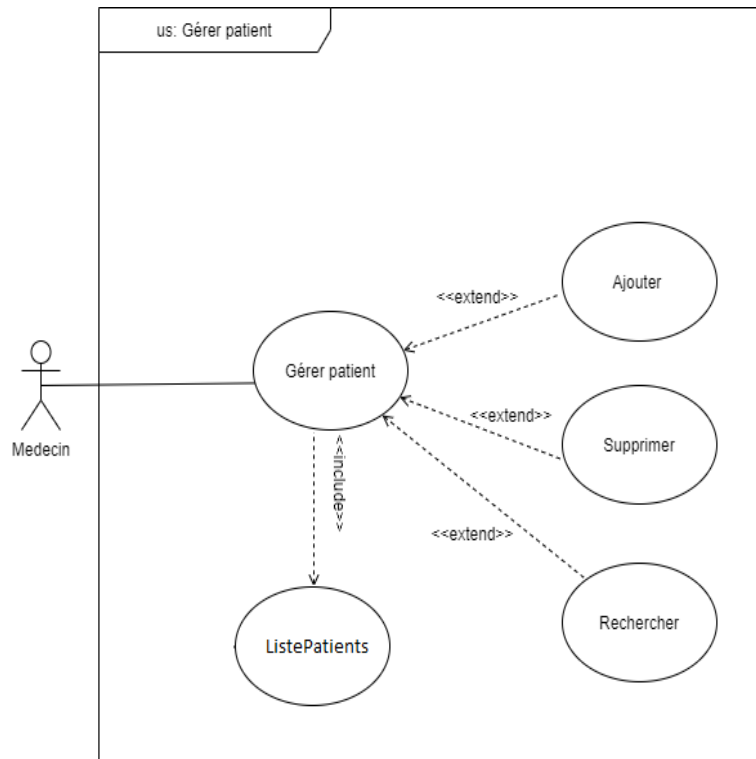


FIGURE 2.8 – Diagramme de cas d'utilisation « Gérer Patient »

La description textuelle du cas d'utilisation « Gérer Patient » est représentée par le tableau 2.9 ci-dessous :

Cas d'utilisation Gérer Patient	
Titre	Gérer patient
Résumé	La gestion des patients permet d'ajouter, rechercher ou supprimer des patients.
Acteurs	Médecin.
Description des scénarios	
Pré-conditions	Accéder à la liste des patients
Scénario Nominal	<ul style="list-style-type: none"> -Le médecin Accède à son espace ; -Le médecin ajoute un patient dans une liste en saisissant le nom d'utilisateur de ce dernier ; -Le médecin peut rechercher un patient en saisissant son nom sur la barre de recherche ; -Le médecin supprime un patient en cliquant sur le bouton «supprimer».
Enchaînement d'erreurs	<ul style="list-style-type: none"> -Erreur d'authentification ; -Patient non inscrit ; -Patient déjà ajouté (Ajout) ; -Patient non trouvé (Recherche).
Post-conditions	Les mises à jour ont été effectuées avec succès

TABLE 2.9 – Description textuelle du cas d'utilisation « Gérer Patient »

8. Diagramme de cas d'utilisation : « Rédiger un rapport »

La figure 2.9 représente le diagramme de cas d'utilisation « Rédiger un Rapport »

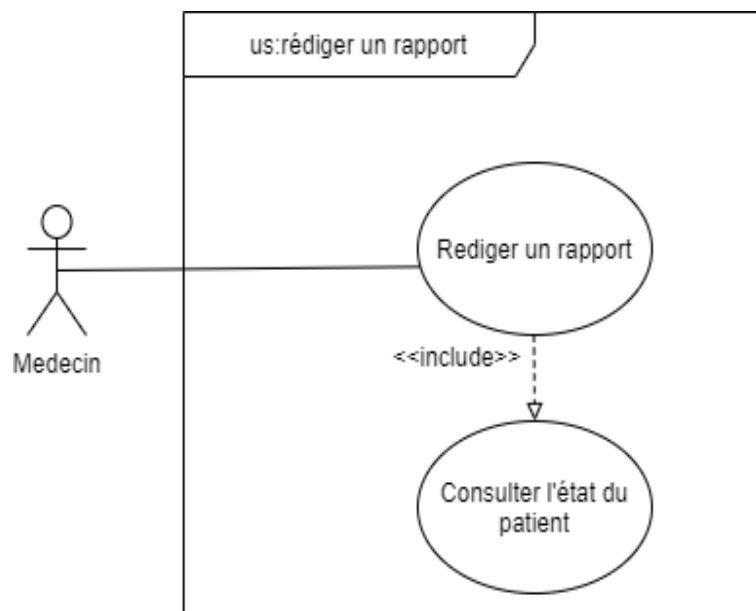


FIGURE 2.9 – Diagramme de cas d'utilisation « Rédiger un Rapport »

La description textuelle du cas d'utilisation « Rédiger un rapport » est représentée par le tableau 2.10 ci-dessous :

Cas d'utilisation Rédiger un rapport	
Titre	Rédiger un rapport
Résumé	Rédiger un rapport permet regrouper toutes les informations médicales essentielles sur l'état de santé du patient.
Acteurs	Médecin
Description des scénarios	
Pré-condition	Consulter l'état du patient.
Scénario Nominal	-Le Médecin accède à son espace ; -Le Médecin accède au profil du patient ; -Le Médecin accède à l'interface dédiée à la rédaction d'un rapport ; -Une fois le rapport est rédigé il est ensuite enregistré.
Enchaînement d'erreurs	Erreur d'authentification.
Post-condition	Rapport rédigé et enregistré.

TABLE 2.10 – Description textuelle du cas d'utilisation « Rédiger un rapport »

9. Description textuelle du cas d'utilisation : « Envoyer un rapport »

La figure 2.10 représente le diagramme de cas d'utilisation « Envoyer un rapport »

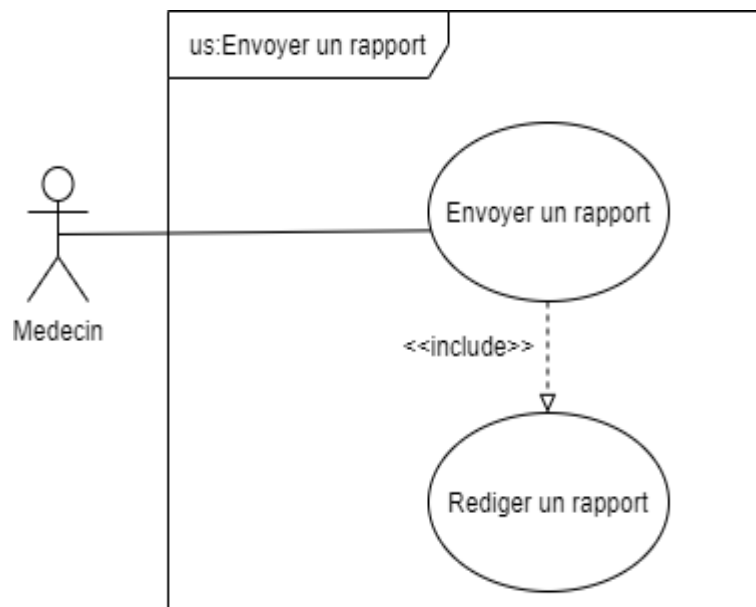


FIGURE 2.10 – Diagramme de cas d'utilisation « Envoyer un rapport »

La description textuelle du cas d'utilisation « Envoyer un rapport » est représentée par le tableau 2.11 ci-dessous :

Cas d'utilisation Envoyer un rapport	
Titre	Envoyer un rapport
Résumé	Envoyer un rapport au patient concerné permet de l'informer de l'état et l'évolution de son état de santé.
Acteurs	Médecin
Description des scénarios	
Pré-condition	Rapport rédigé.
Scénario Nominal	<ul style="list-style-type: none"> -Le Médecin accède à son espace ; -Le Médecin accède au profil du patient ; -Le Médecin accède à l'interface dédiée à l'envoi d'un rapport par mail ; -Le système recherche le document demandé ; -Le Medecin insert les infomations essentielles à l'envoi d'un mail ; -Le système vérifie la validité des champs remplies ; -Le système envoie le mail.
Enchaînement d'erreurs	Erreur d'authentification.
Post-condition	Rapport envoyé.

TABLE 2.11 – Description textuelle du cas d'utilisation « Envoyer un rapport »

10. Diagramme de cas d'utilisation : « Collecter les données »

La figure 2.11 représente le diagramme de cas d'utilisation « Capter les données »

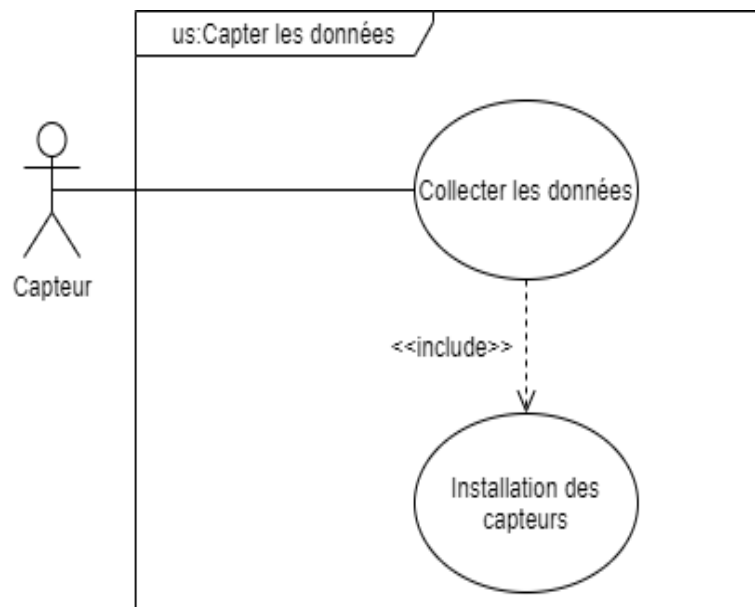


FIGURE 2.11 – Diagrammes de cas cas d'utilisation « Collecter les données »

La description textuelle du cas d'utilisation « Collecter les données » est représentée par le tableau 2.12 ci-dessous :

Cas d'utilisation Collecter les données	
Titre	Collecter les données
Résumé	Les capteurs (BSN,PSN) collectent les données du patient et de son environnement.
Acteurs	Capteurs
Description des scénarios	
Pré-conditions	Capteurs installés
Scénario Nominal	Les capteurs (BSN,PSN) collectent les données du patient et de son environnement et sont ensuite enregistrées et envoyées au serveur.
Enchaînement d'erreurs	Données non collectées.
Post-conditions	Les données sont collectées avec succès

TABLE 2.12 – Description textuelle du cas d'utilisation « Collecter les données »

11. Diagramme de cas d'utilisation : « Envoyer les données »

La figure 2.12 représente le diagramme de cas d'utilisation « Envoyer les données »

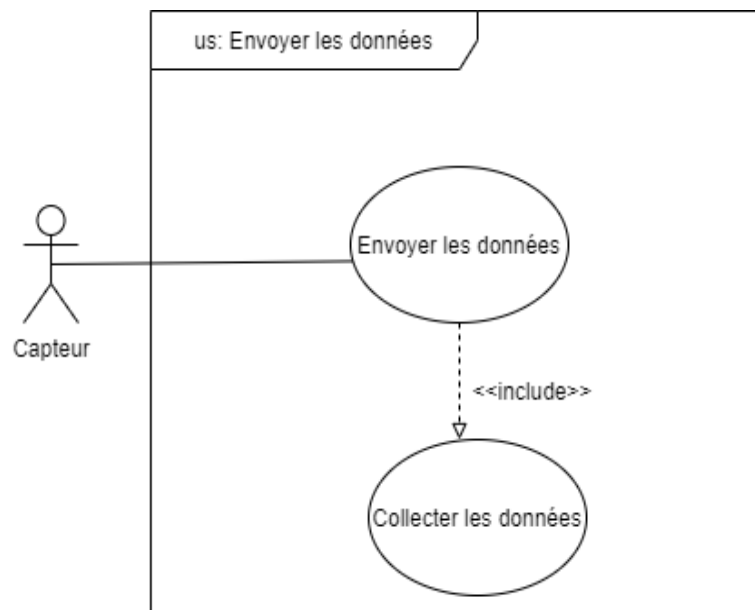


FIGURE 2.12 – Diagramme de cas d'utilisation « Envoyer les données »

La description textuelle du cas d'utilisation « Envoyer les données » est représentée par le tableau 2.13 ci-dessous :

Cas d'utilisation Envoyer les données	
Titre	Envoyer les données
Résumé	Les capteurs (BSN,PSN) envoient les données collectées du patient et de son environnement.
Acteurs	Capteurs
Description des scénarios	
Pré-conditions	Données collectées
Scénario Nominal	Les capteurs (BSN,PSN) envoient grâce au différentes technologies de communication à la station de base les données collectées du patient et de son environnement, cette dernière envoie ces mêmes données au serveur afin d'être stockées ensuite traitées par l'application.
Enchaînement d'erreurs	Échec d'envoi des données.
Post-conditions	Les données sont envoyées avec succès

TABLE 2.13 – Description textuelle du cas d'utilisation « Envoyer les données »

12. Diagramme de cas d'utilisation : « Stocker les données »

La figure 2.13 représente le diagramme de cas d'utilisation « Stocker les données »

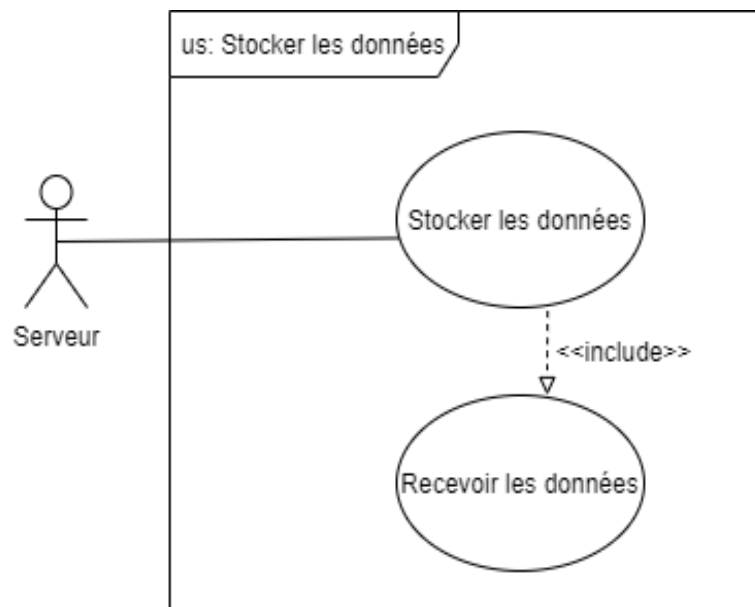


FIGURE 2.13 – Diagramme de cas d'utilisation « Stocker les données »

La description textuelle du cas d'utilisation « Stocker les données » est représentée par le tableau 2.14 ci-dessous :

Cas d'utilisation Stocker les données	
Titre	Stocker les données
Résumé	Le serveur stock les données reçues par la station de base.
Acteurs	Serveur BDD
Description des scénarios	
Pré-conditions	Données reçues
Scénario Nominal	Le serveur stock les données envoyées par la station de base via WIFI, ces dernières sont ensuite traitées par le système.
Enchaînement d'erreurs	Données non reçues.
Post-conditions	Les données sont reçues et stockées avec succès

TABLE 2.14 – Description textuelle du cas d'utilisation « Stocker les données »

2.6 Maquettes

Une maquette est un schéma définissant les zones et composants de l'interface d'une application ou d'un site internet. Il ne s'agit d'autre chose que d'un croquis qui présente la structure de la future application.

Il existe plusieurs logiciels pour créer les maquettes d'une application mobile, nous citons : AXURE RP, BALSAMIQUE MOCKUPS , FRAME BOX, etc.

Pour la représentation des maquettes de notre application mobile, nous avons choisi d'utiliser le logiciel « Balsamiq Mockups » qui est simple et facile à utiliser. Dans ce qui suit, nous allons présenter les différentes interfaces de notre application mobile à développer.

2.6.1 Les principales interfaces (maquettes)

1. Maquette de l'interface d'accueil

La figure 2.14 ci-dessous représente la maquette de l'interface d'accueil.

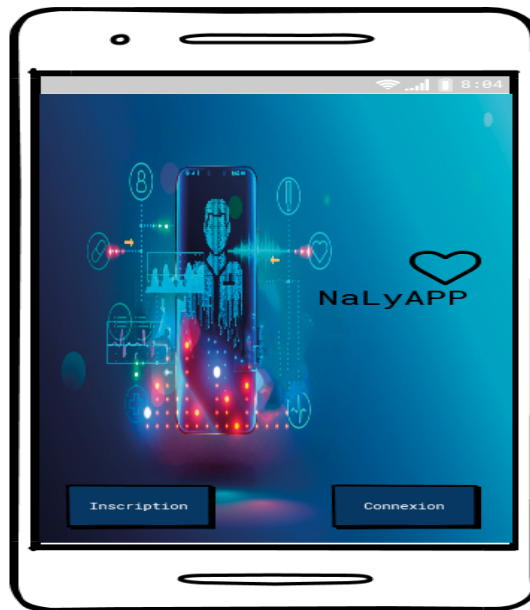


FIGURE 2.14 – Maquette de l'interface d'accueil

2. Maquette de l'interface Authentification

La figure 2.15 ci-dessous représente la maquette de l'interface de connexion pour les utilisateurs ayant déjà un compte.



FIGURE 2.15 – Maquette de l'interface Authentification

3. Maquette de l'interface d'inscription

La figure 2.16 ci-dessous représente la maquette de l'interface d'inscription.



FIGURE 2.16 – Maquette de l'interface d'inscription

4. Maquette de l'interface représentant le type d'utilisateurs

La figure 2.17 ci-dessous représente la maquette de l'interface représentant le type d'utilisateur.



FIGURE 2.17 – Maquette de l'interface représentant le type d'utilisateurs

5. Maquette de l'interface Espace Patient

La figure 2.18 ci-dessous représente la maquette de l'interface Espace Patient.



FIGURE 2.18 – Maquette de l'interface Espace patient

6. Maquette de l'interface Espace Médecin

La figure 2.19 ci-dessous représente la maquette de l'interface Espace Médecin.



FIGURE 2.19 – Maquette de l'interface Espace médecin

7. Maquette de l'interface Dossier médicale

La figure 2.20 ci-dessous représente la maquette de l'interface Dossier médicale.

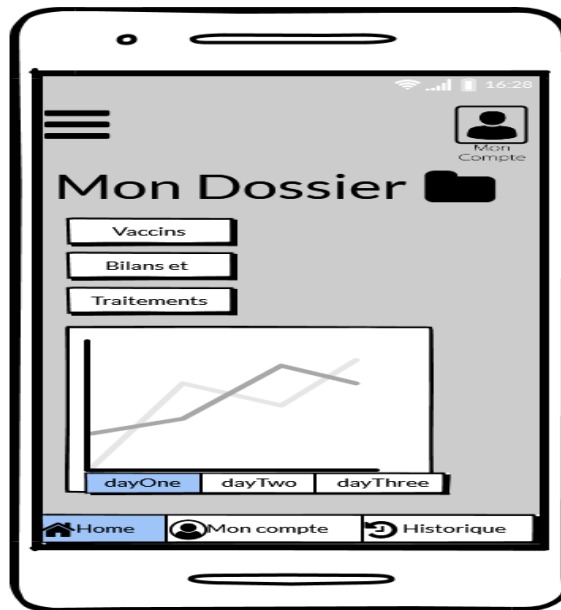


FIGURE 2.20 – Maquette de l'interface Dossier médicale

2.7 Conclusion

Ce chapitre, nous a permis de définir les fonctionnalités de notre application, ce qui nous mène à entamer la phase de conception pour assurer une bonne mise en œuvre d'un système fonctionnel répondant aux besoins cités.

Chapitre 3

Analyse et Conception

3.1 Introduction

Le modèle d'analyse nous permet de faire une représentation transitoire entre l'expression des besoins d'une part et le modèle de conception d'autre part. Il permet de reformuler les besoins sous une forme proche de ce que sera la conception et la réalisation mais tout en s'abstrayant de leurs contraintes techniques.

Dans la suite de ce chapitre, nous allons décrire dans un premier temps les différents diagrammes de notre système à savoir les diagrammes séquence, diagrammes d'interaction et le digramme de classes. Ensuite, nous allons appliquer les règles de passage au relationnel ainsi que les règles de gestion de l'application et nous terminons par une conclusion.

3.2 Diagrammes d'analyse

3.2.1 Diagramme de séquence système

Le diagramme de séquence permet de représenter des collaborations entre des objets pour réaliser une fonctionnalité tout en mettant l'accent sur les relations temporelles. De nombreuses notations annexes permettent de préciser la nature des messages (Appel de procédures, simple signal, message répétitif, etc.) et les données véhiculées.

Le diagramme de séquence fait partie des diagrammes comportementaux (dynamique) et plus précisément des diagrammes d'interactions, il permet de [8] :

- Représenter des échanges entre les différents objets et acteurs du système en fonction du temps.
- A moins que le système à modéliser soit extrêmement simple, nous ne pouvons pas modéliser la dynamique globale du système dans un seul diagramme. Nous ferons donc appel à un ensemble de diagrammes de séquences chacun correspondant à une sous fonction du système, généralement d'ailleurs pour illustrer un cas d'utilisation.

Dans ce qui suit, nous donnons les différents éléments d'un diagramme de séquence [8] :

1. **L'objet** : Dans un diagramme de séquence, l'objet a la même représentation que dans le diagramme des objets. C'est-à-dire un rectangle dans lequel figure le nom de l'objet.
2. **La ligne de vie** : Comme il représente la dynamique du système, le diagramme de séquence fait entrer en action les instances de classes intervenant dans la réalisation d'un cas d'utilisation particulier.
 - A chaque objet est associé une ligne de vie (en trait pointillés à la verticale de l'objet).
 - La ligne de vie indique les périodes d'activité de l'objet (généralement, les moments où l'objet exécute une de ces méthodes).
3. **Les messages** : Un message définit une communication particulière entre des lignes de vie. Ainsi, un message est une communication d'un objet vers un autre objet. Plusieurs types de messages existent, les plus communs sont :

- **Les messages synchrones :** La réception d'un message synchrone doit provoquer chez le destinataire le lancement d'une de ses méthodes.
- **Les messages asynchrones :** Dans le cas d'un message asynchrone, l'expéditeur n'attend pas la fin de l'activation de la méthode invoquée chez le destinataire.

Dans ce qui suit, nous présenterons les diagrammes de séquences des cas d'utilisation les plus pertinents de notre système.

3.2.1.1 Diagramme de séquence « Inscription »

La figure 3.1 représente le diagramme de séquence correspondant au cas d'utilisation « Inscrition »

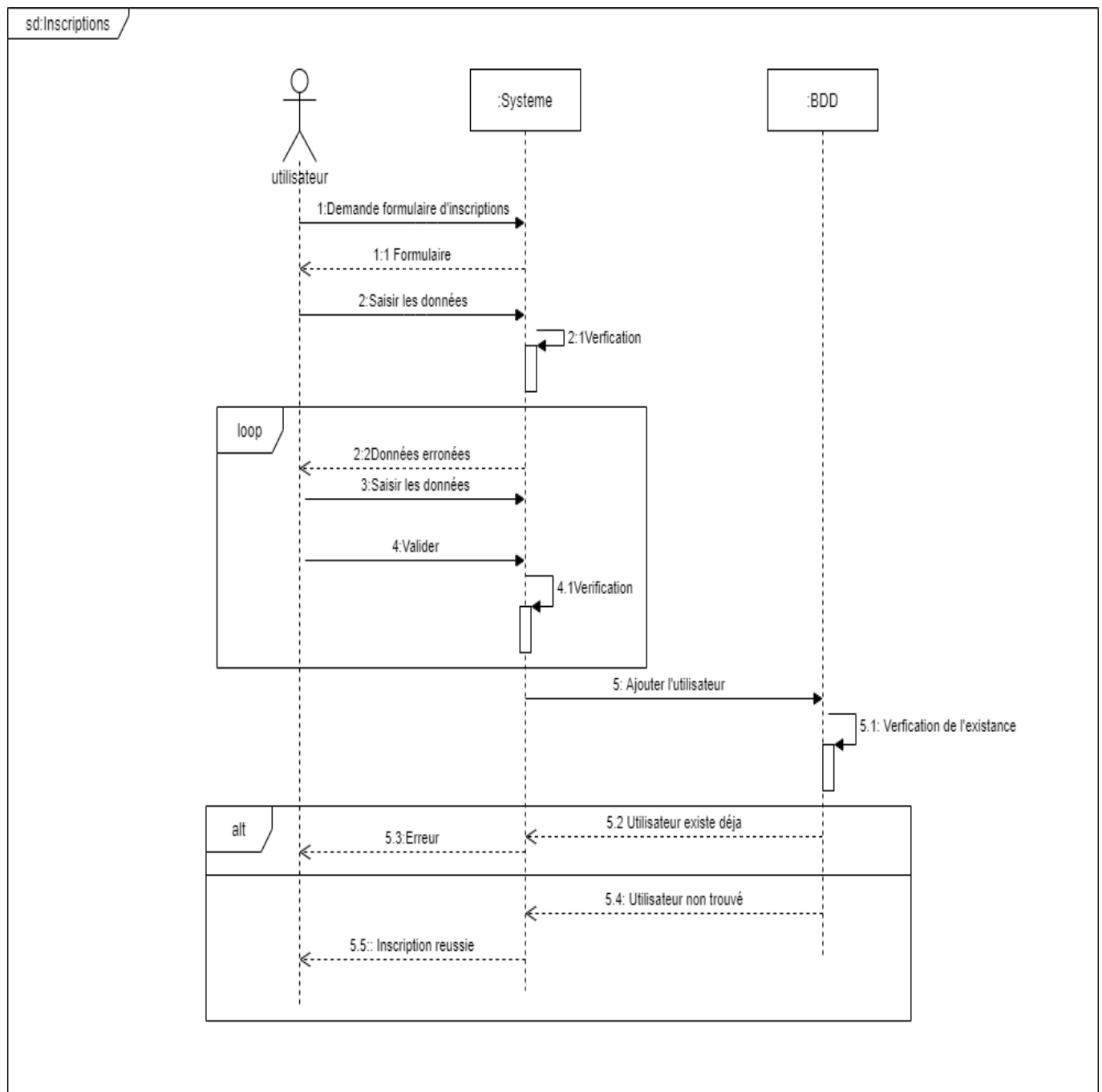


FIGURE 3.1 – Diagramme de séquence « Inscrition »

3.2.1.2 Diagramme de séquence « Authentification »

La figure 3.2 représente le diagramme de séquence correspondant au cas d'utilisation « Authentification »

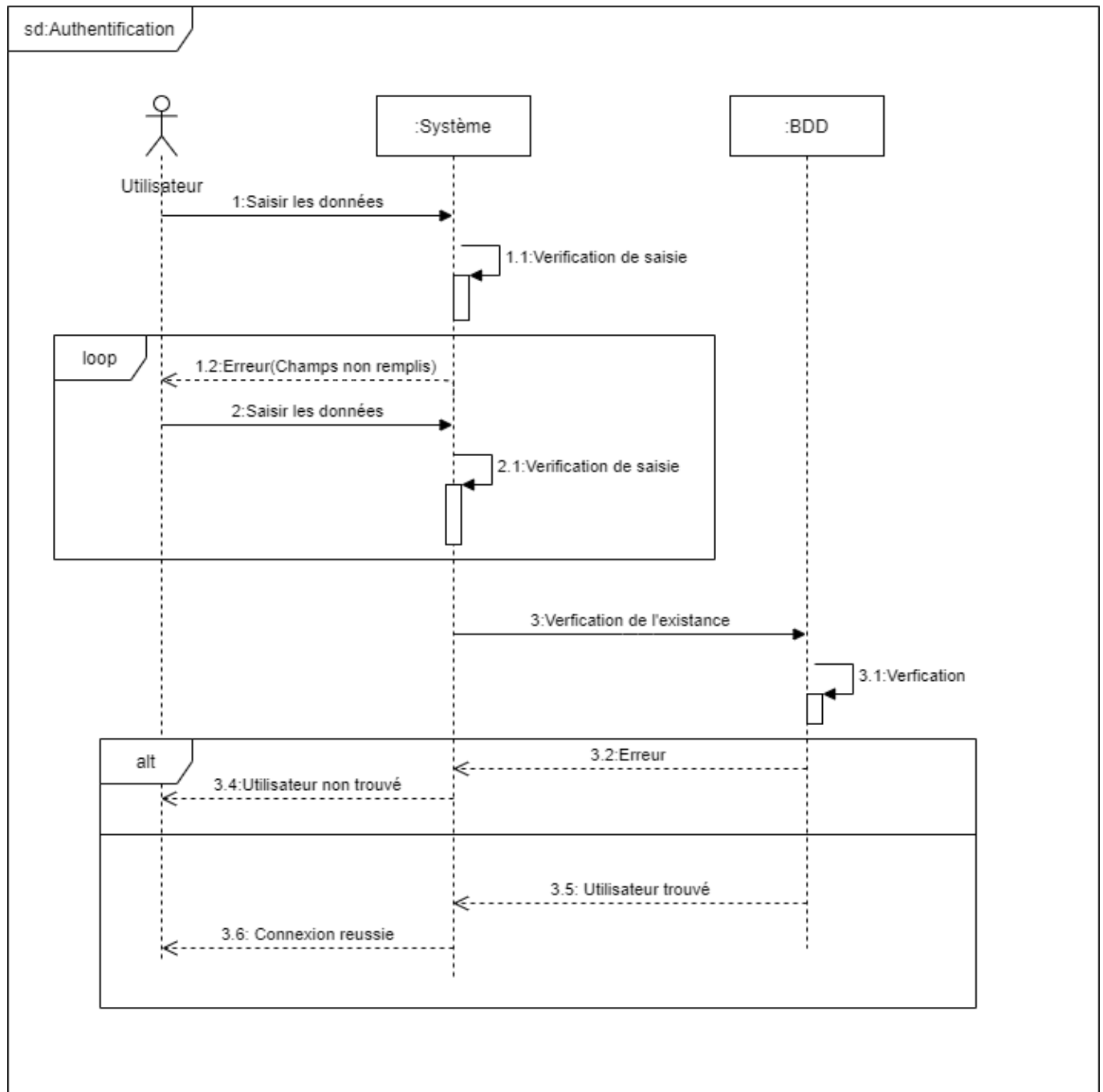


FIGURE 3.2 – Diagramme de séquence « Authentification »

3.2.1.3 Diagramme de séquence « Gérer patient »

le diagramme de séquence « Gérer patient » est représenté dans la figure 3.3 suivante :

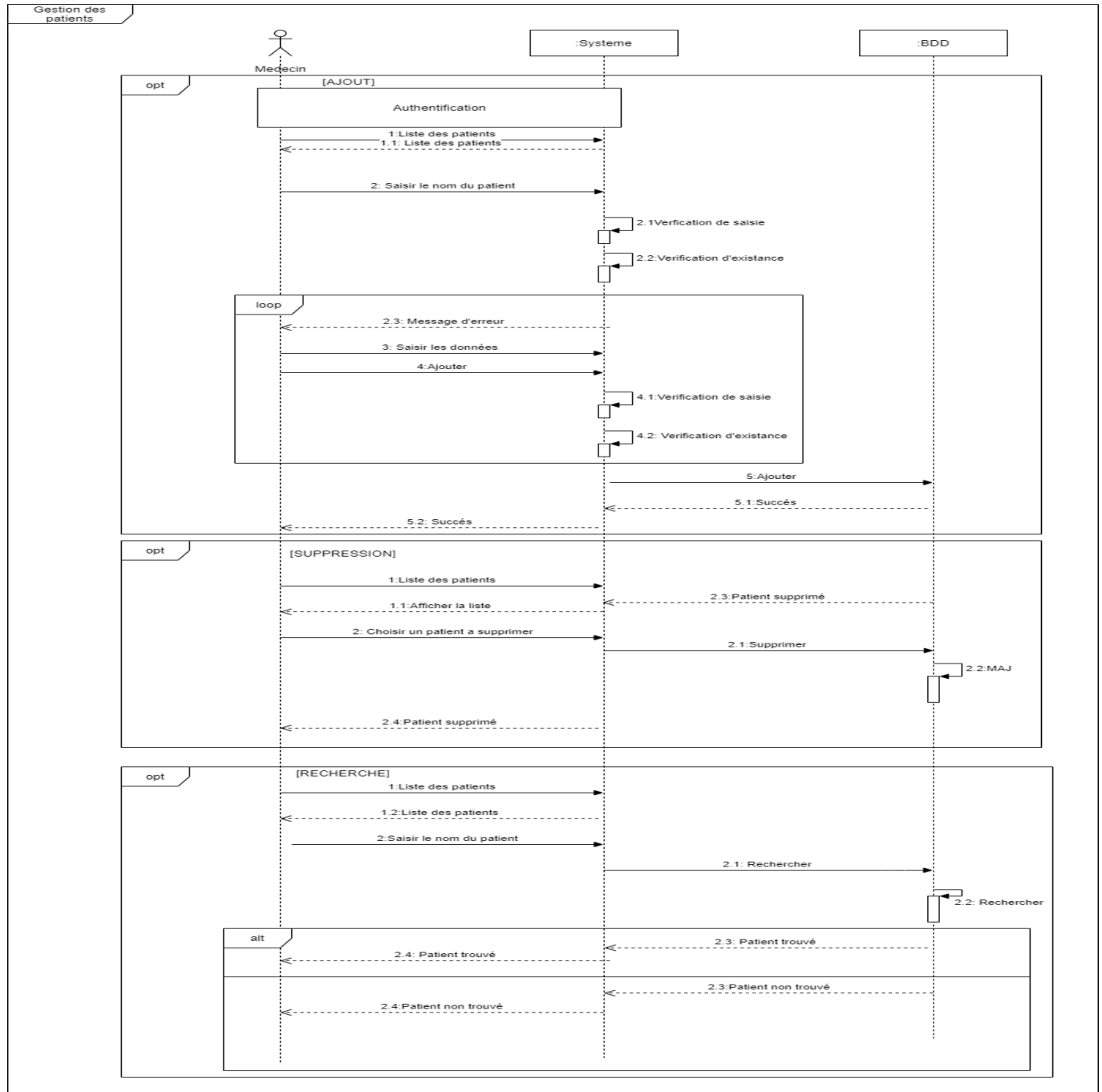


FIGURE 3.3 – Diagramme de séquence « Gérer patient »

3.2.1.4 Diagramme de séquence « Consulter mesures »

La figure 3.4 représente le diagramme de séquence correspondant au cas d'utilisation « Consulter mesures »

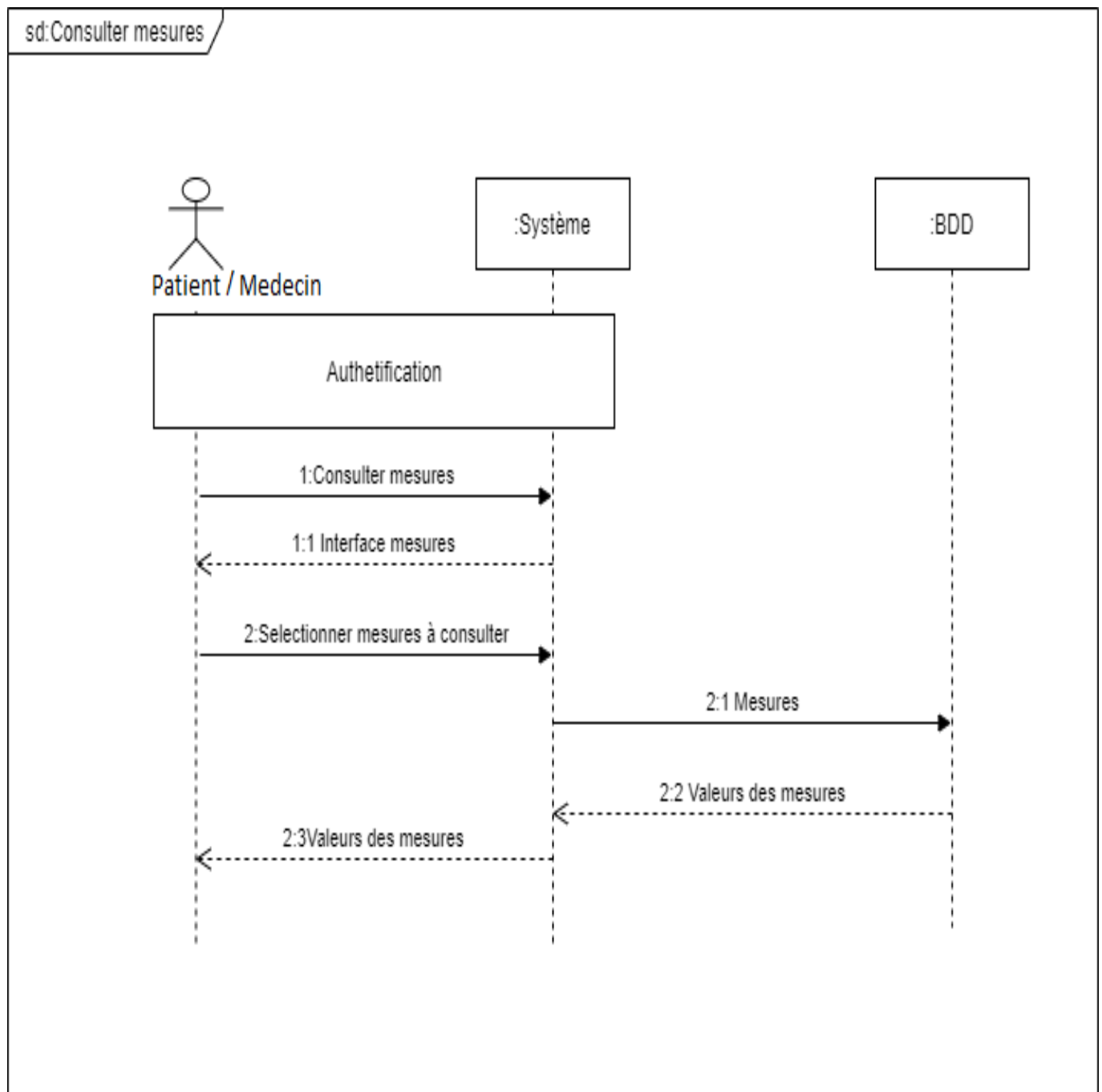


FIGURE 3.4 – Diagramme de séquence « Consulter mesures »

3.2.1.5 Diagramme de séquence « Générer alerte »

La figure 3.5 représente le diagramme de séquence correspondant au cas d'utilisation « Générer alerte ».

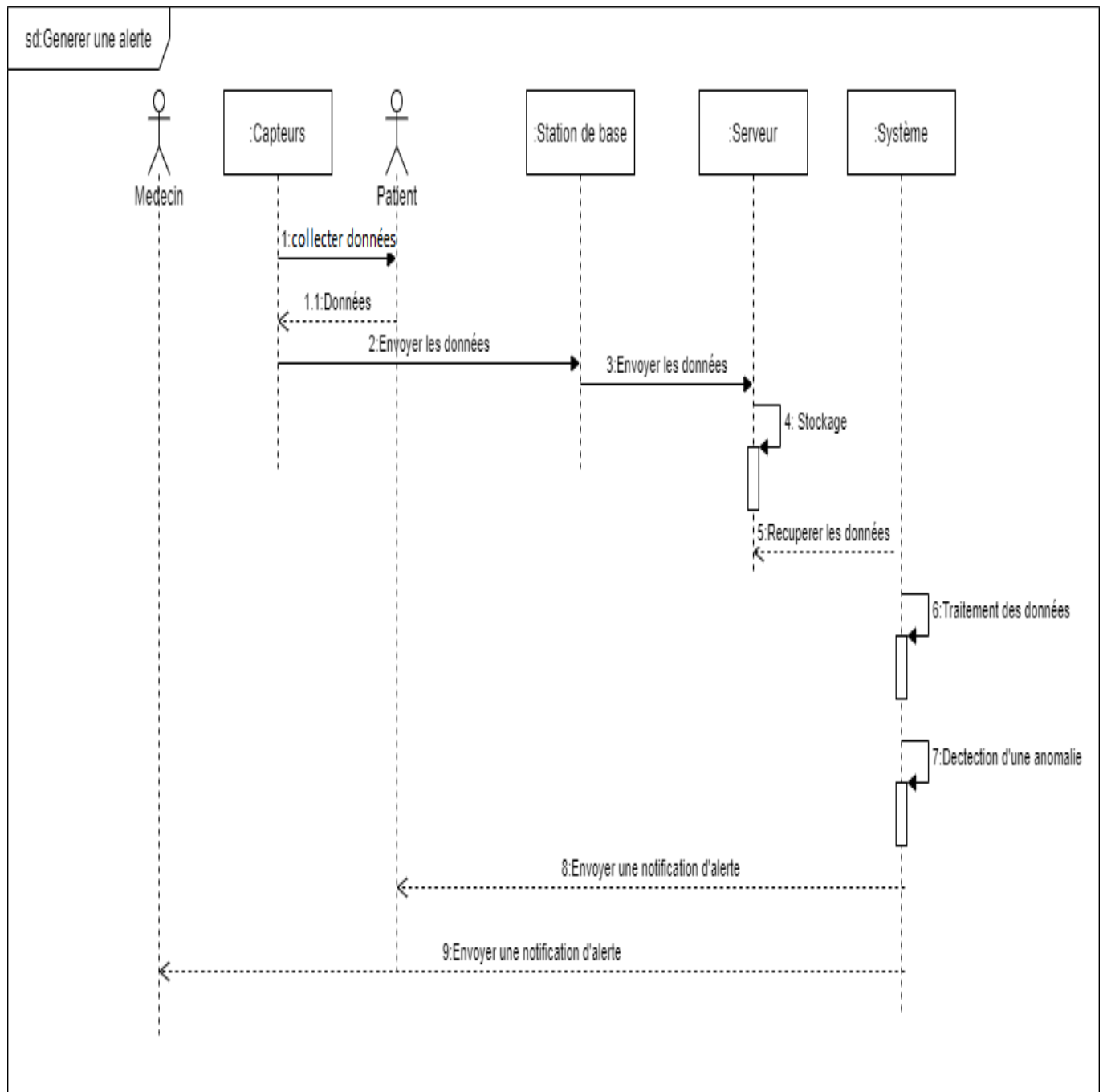


FIGURE 3.5 – Diagramme de séquence « Générer alerte »

3.2.1.6 Diagramme de séquence « Consulter l'historique »

La figure 3.6 représente le diagramme de séquence correspondant au cas d'utilisation « Consulter Historique »

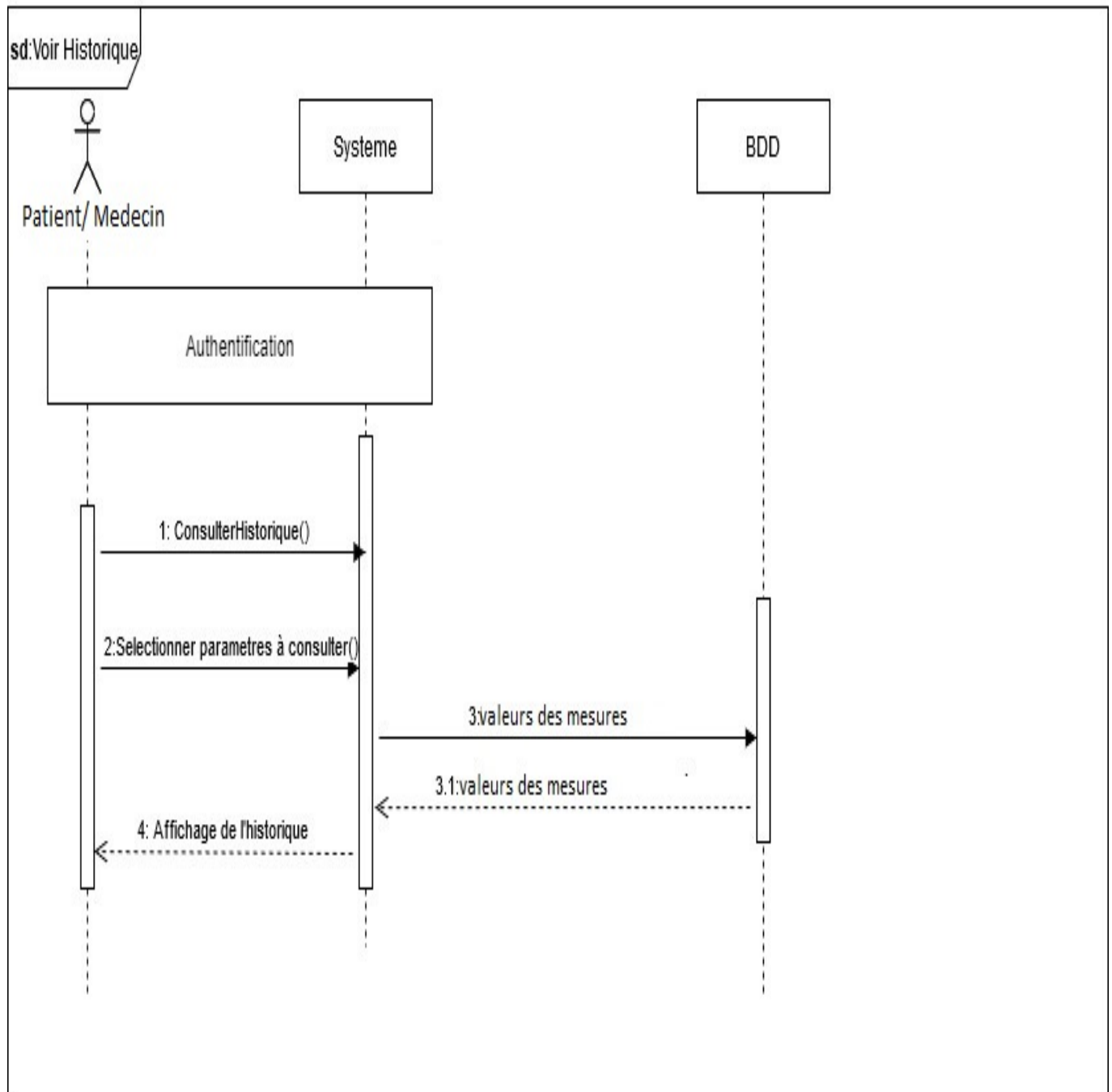


FIGURE 3.6 – Diagramme de séquence « Consulter l'historique »

3.2.1.7 Diagramme de séquence « Consulter profile patient »

La figure 3.7 représente le diagramme de séquence correspondant au cas d'utilisation « Consulter profile patient »

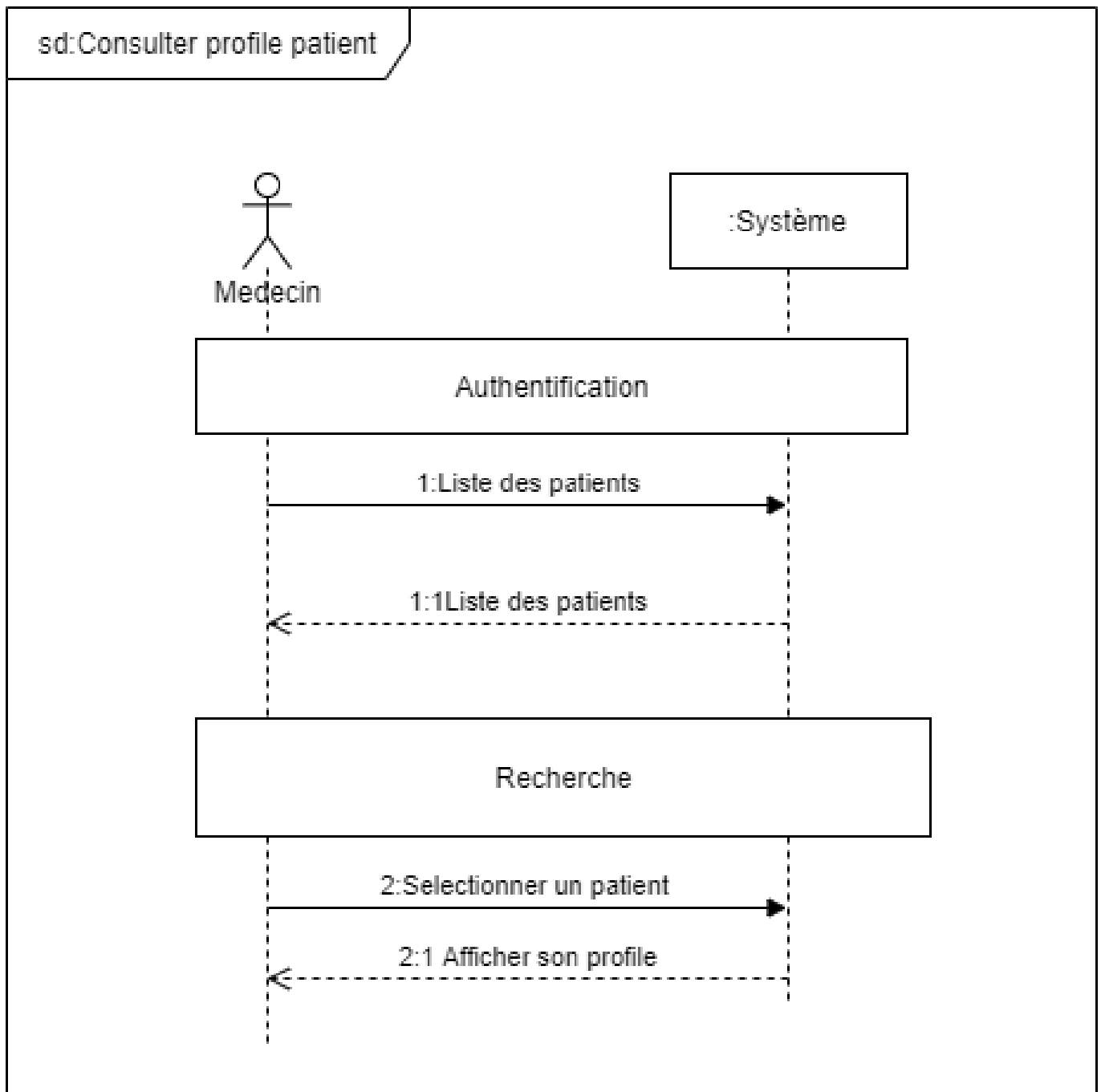


FIGURE 3.7 – Diagramme de séquence « Consulter profile patient »

3.2.1.8 Diagramme de séquence « Rédiger un rapport »

La figure 3.8 représente le diagramme de séquence correspondant au cas d'utilisation « Rédiger un rapport »

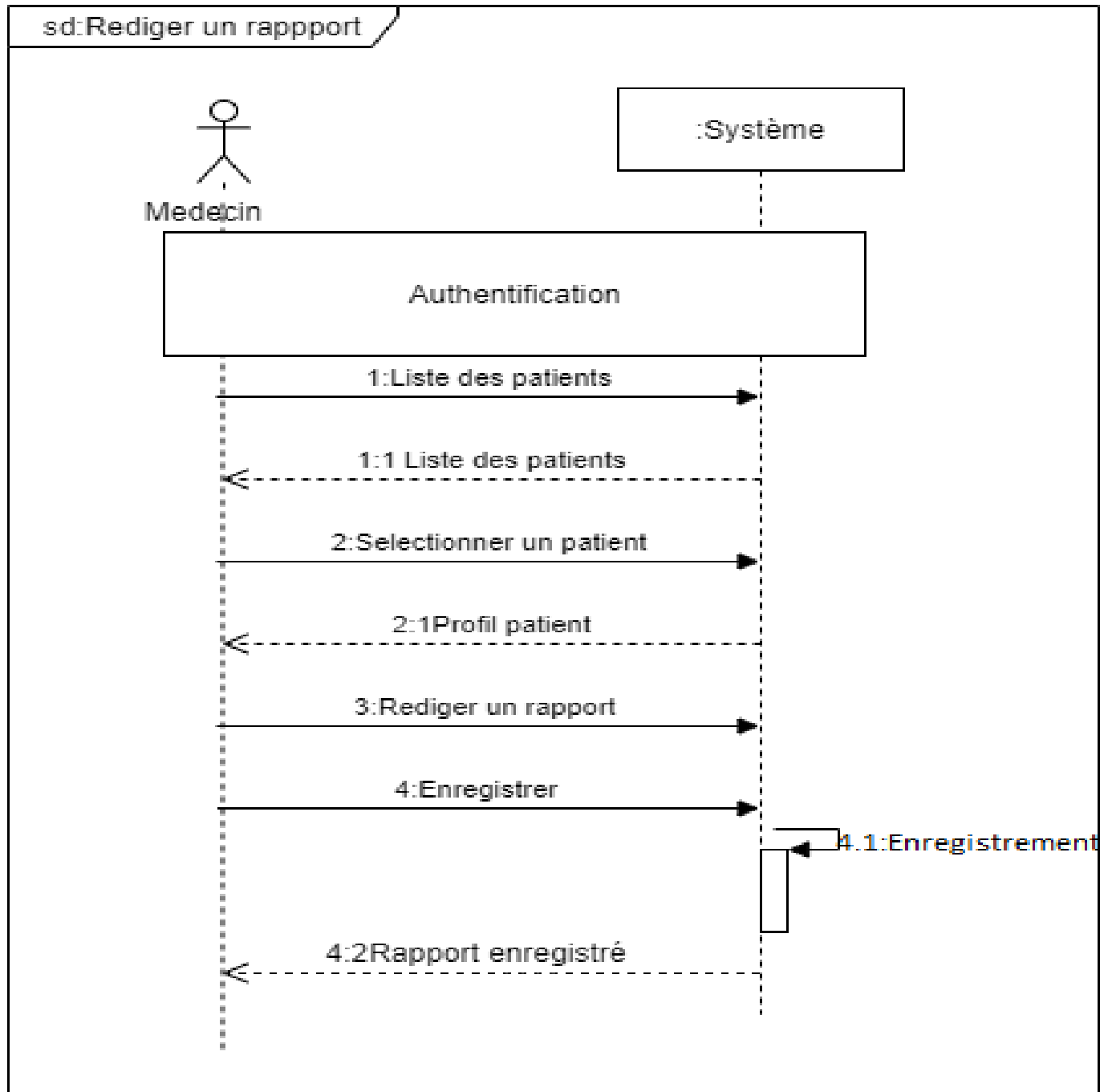


FIGURE 3.8 – Diagramme de séquence « Rédiger un rapport »

3.2.1.9 Diagramme de séquence « Envoyer un rapport »

le diagramme de séquence « Envoyer un rapport » est représenté dans la figure 3.9 suivante :

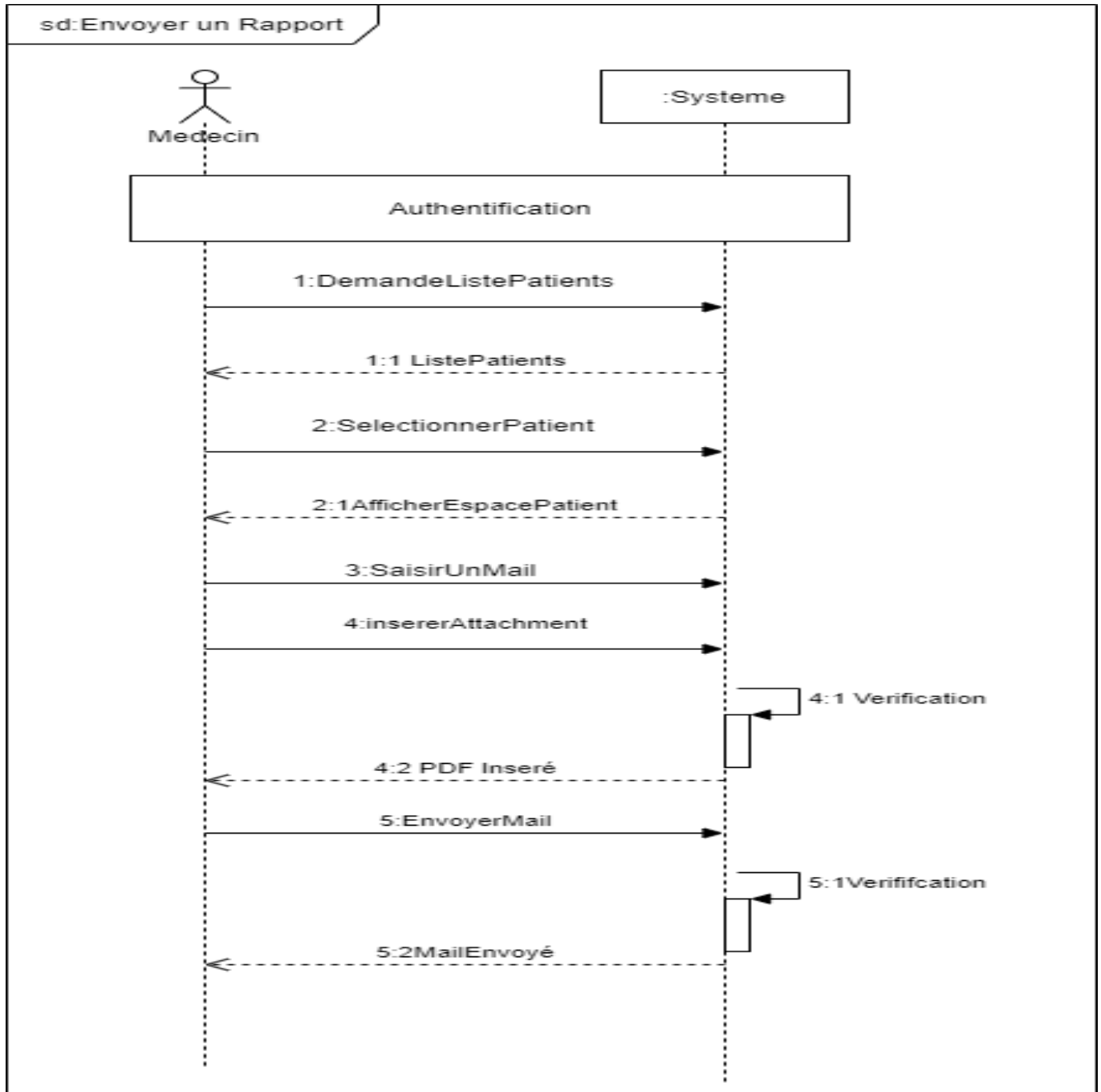


FIGURE 3.9 – Diagramme de séquence « Envoyer un rapport »

3.3 Conception

3.3.1 Diagrammes d'interaction

Les diagrammes d'interaction permettent d'établir un lien entre les diagrammes de cas d'utilisation et les diagrammes de classes : ils montrent comment des objets (i.e. des instances de classes) communiquent pour réaliser une certaine fonctionnalité. Ils apportent ainsi un aspect dynamique à la modélisation du système [7]. Dans cette section, nous allons présenter les diagrammes d'interactions de notre système.

3.3.1.1 Diagramme d'interaction « Authentification »

Le diagramme d'interaction du cas d'utilisation «Authentification» est illustré par la figure 3.10 :

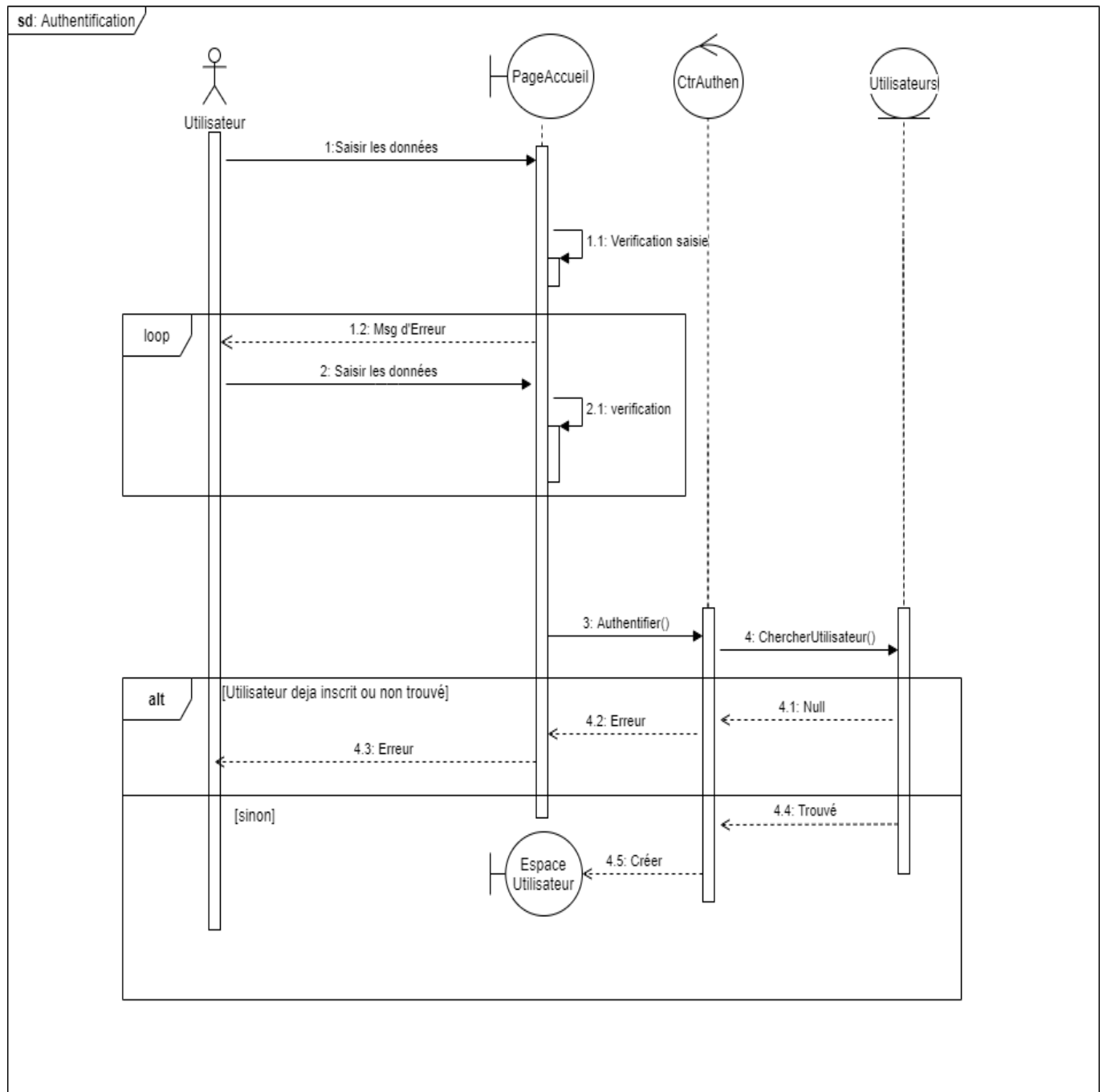


FIGURE 3.10 – Diagramme d'interaction « Authentification »

3.3.1.2 Diagramme d'interaction « Inscription »

La figure 3.11 représente le diagramme d'interaction « Inscription ».

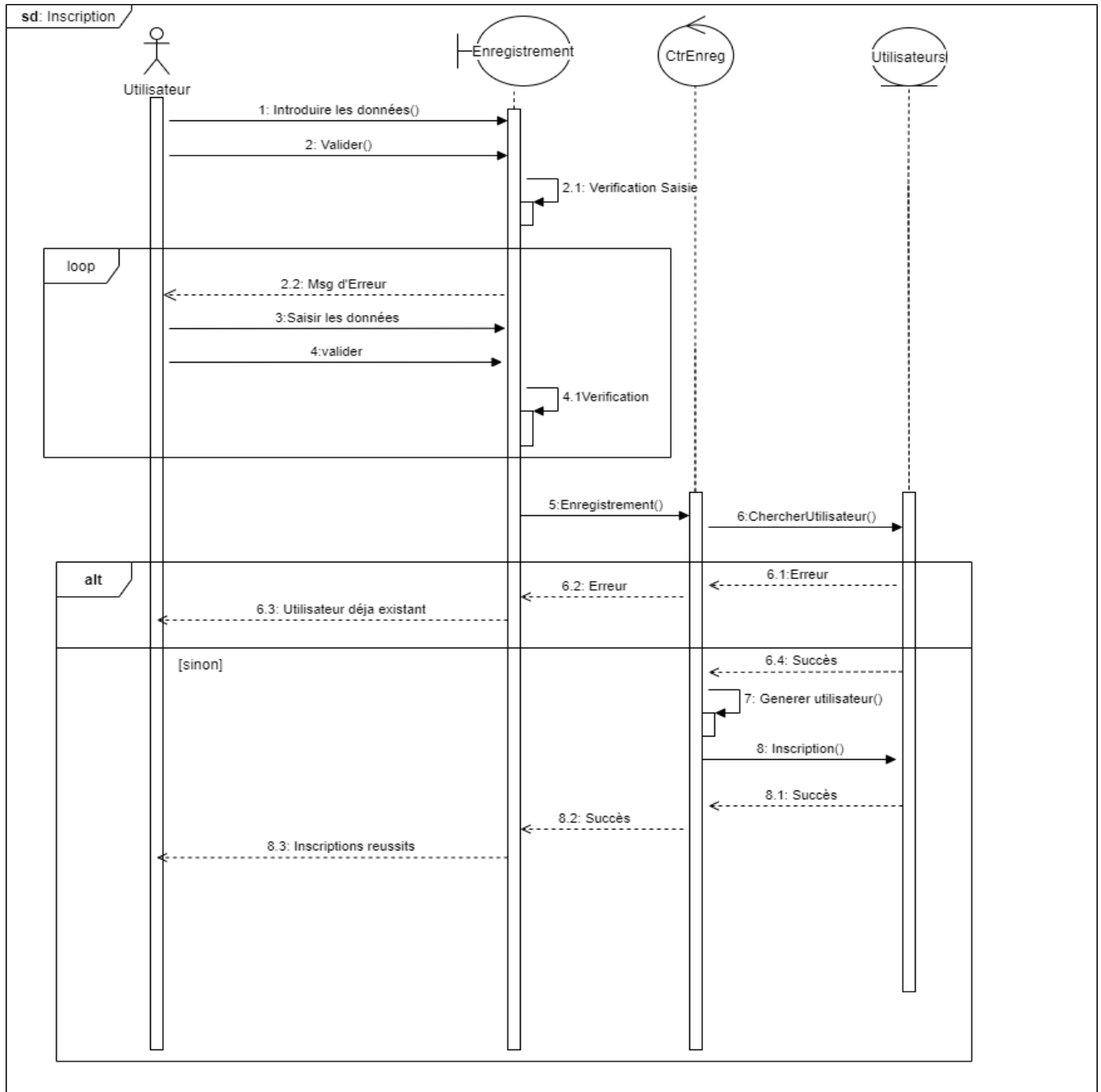


FIGURE 3.11 – Diagramme d'interaction « Inscription »

3.3.1.3 Diagramme d'interaction « Gérer patient »

La figure 3.12 représente le diagramme d'interaction « Gérer patient ».

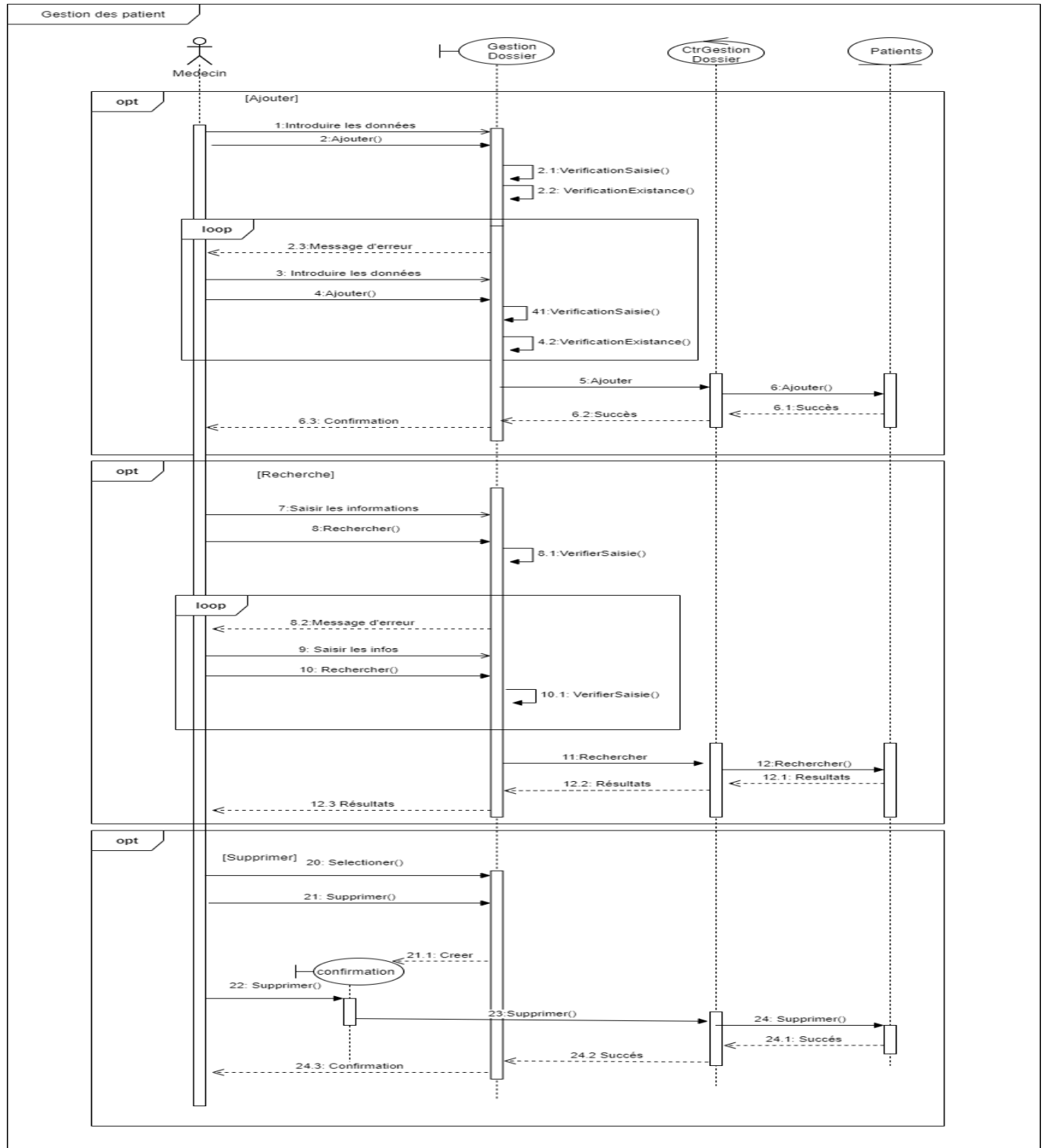


FIGURE 3.12 – Diagramme d'interaction « Gérer patient »

3.3.1.4 Diagramme d'interaction « Consulter mesures »

La figure 3.13 représente le diagramme d'interaction « Consulter mesures ».

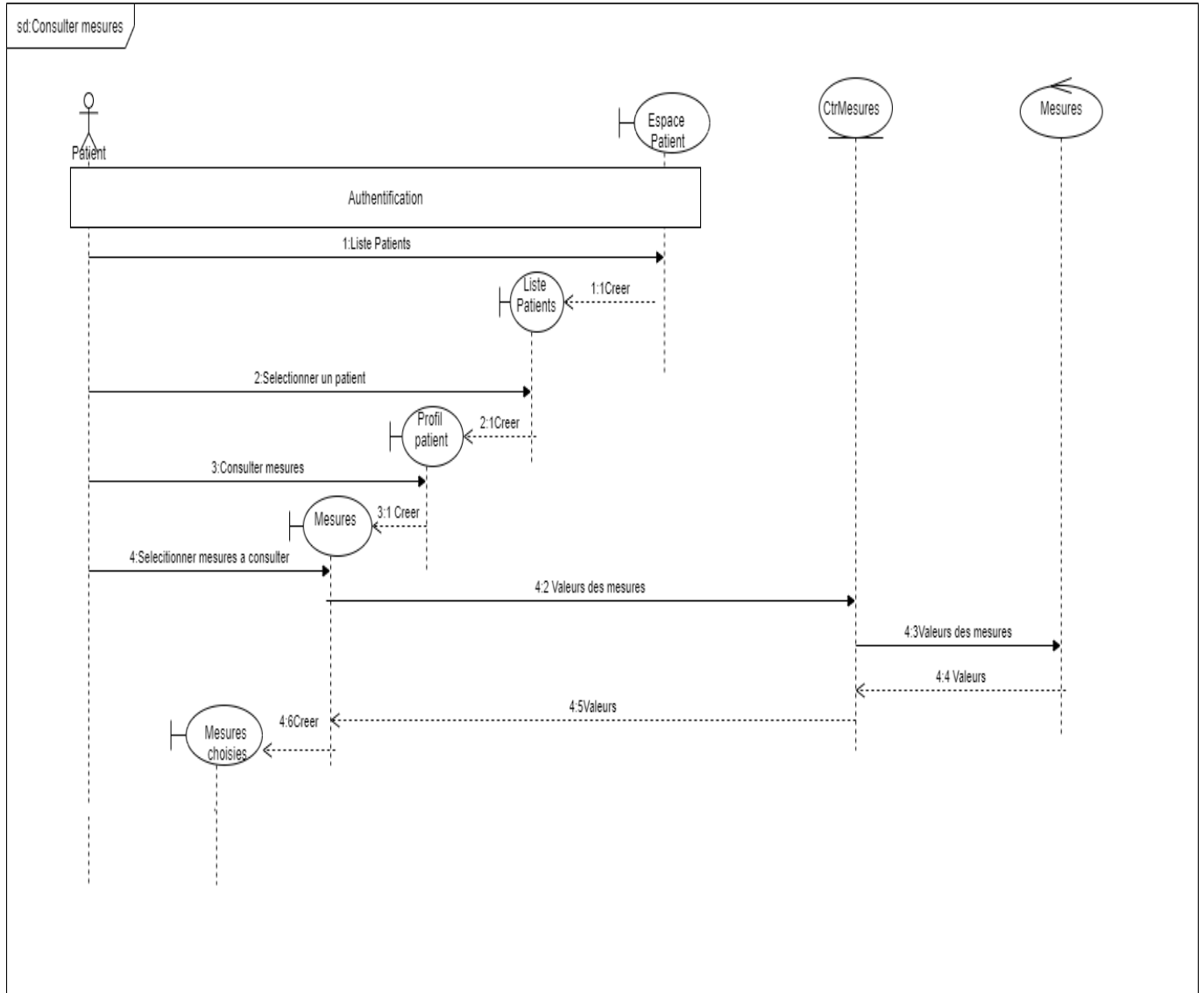


FIGURE 3.13 – Diagramme d'interaction « Consulter mesures »

3.3.1.5 Diagramme d'interaction « Consulter Historique »

La figure 3.14 représente le diagramme d'interaction « Consulter historique ».

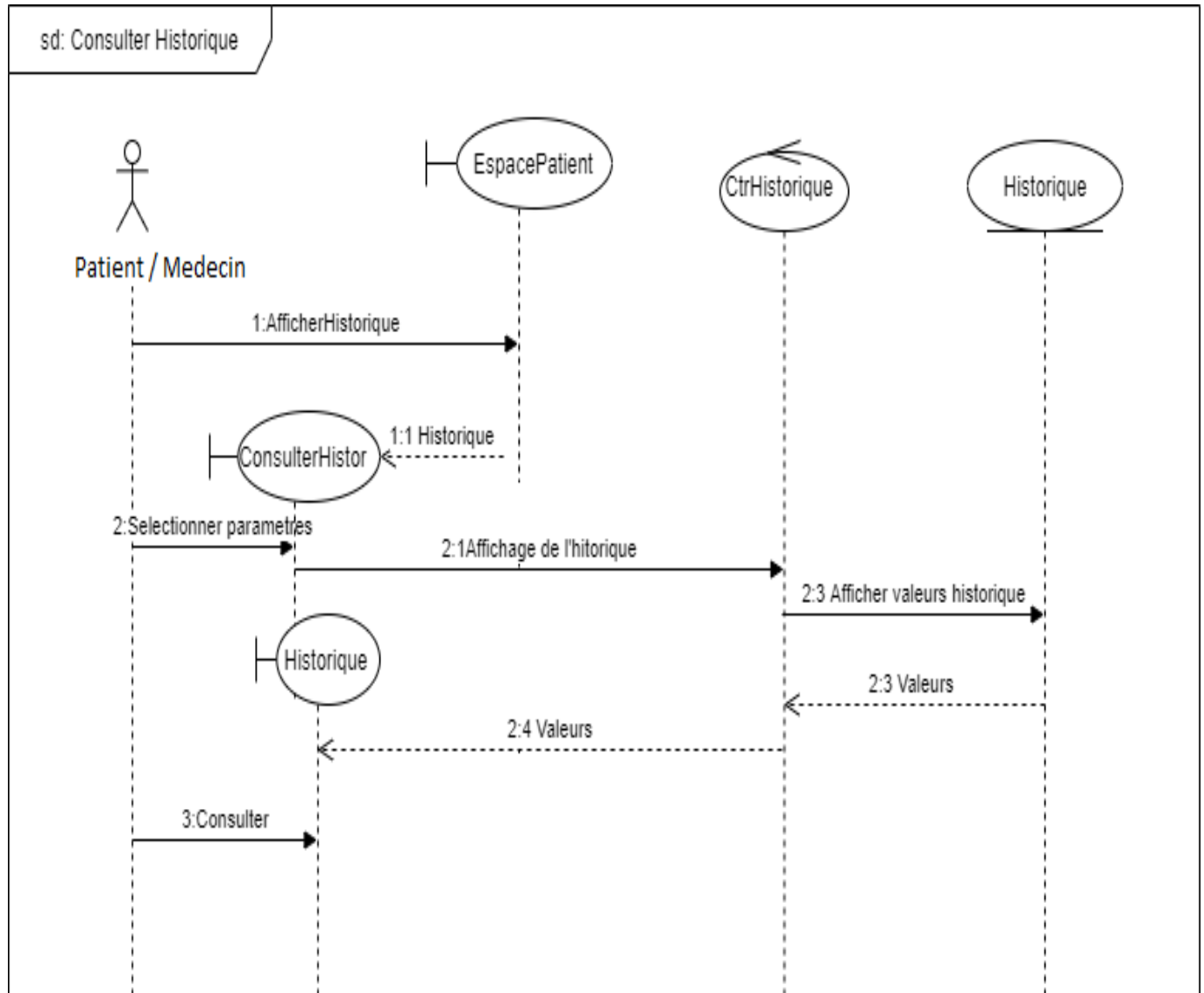


FIGURE 3.14 – Diagramme d'interaction « Consulter Historique »

3.3.1.6 Diagramme d'interaction « Consulter profile patient »

La figure 3.15 représente le diagramme d'interaction « Consulter profile patient ».

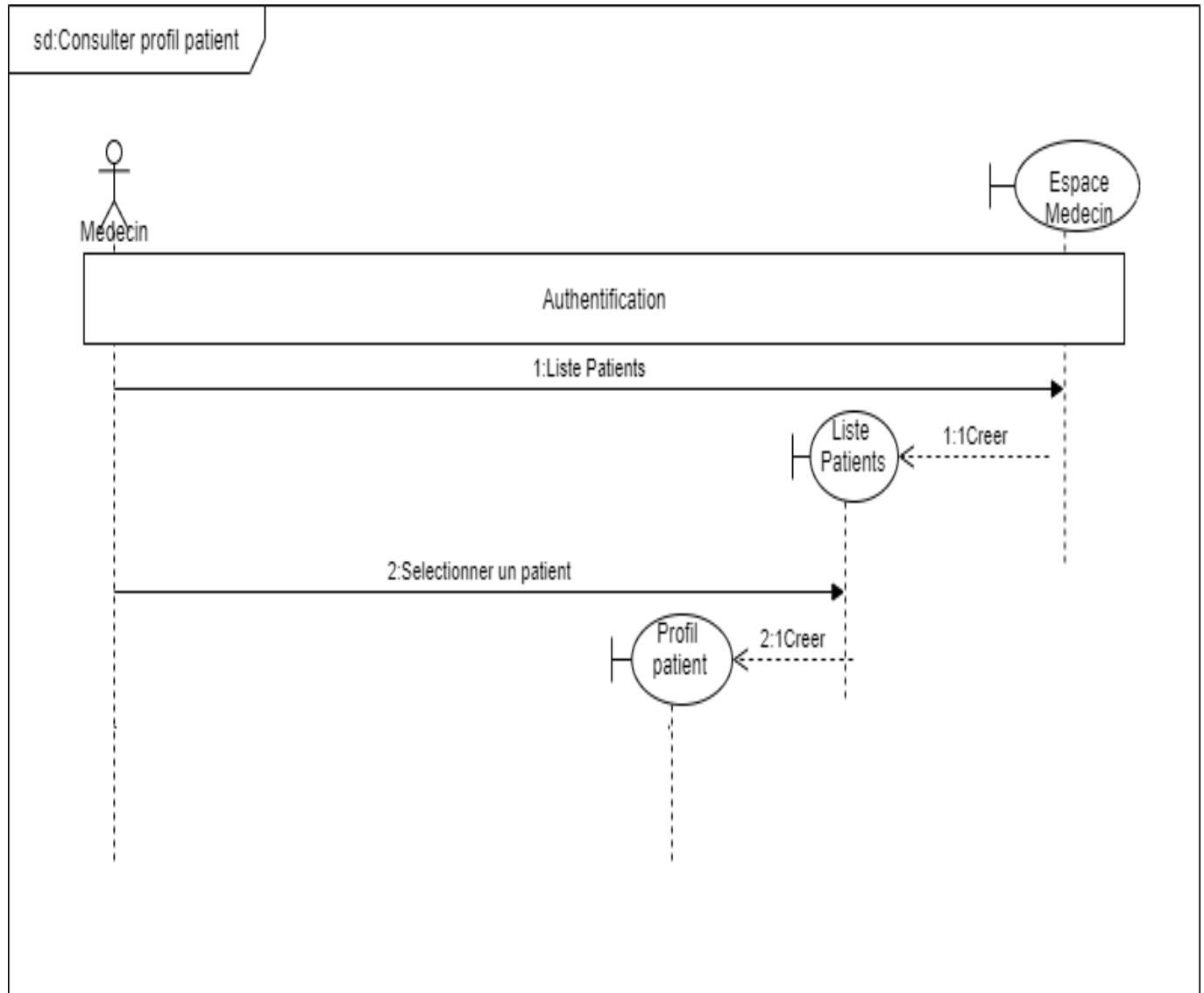


FIGURE 3.15 – Diagramme d'interaction « Consulter profile patient »

3.3.1.7 Diagramme d'interaction « Rédiger rapport »

La figure 3.16 représente le diagramme d'interaction « Rediger un rapport ».

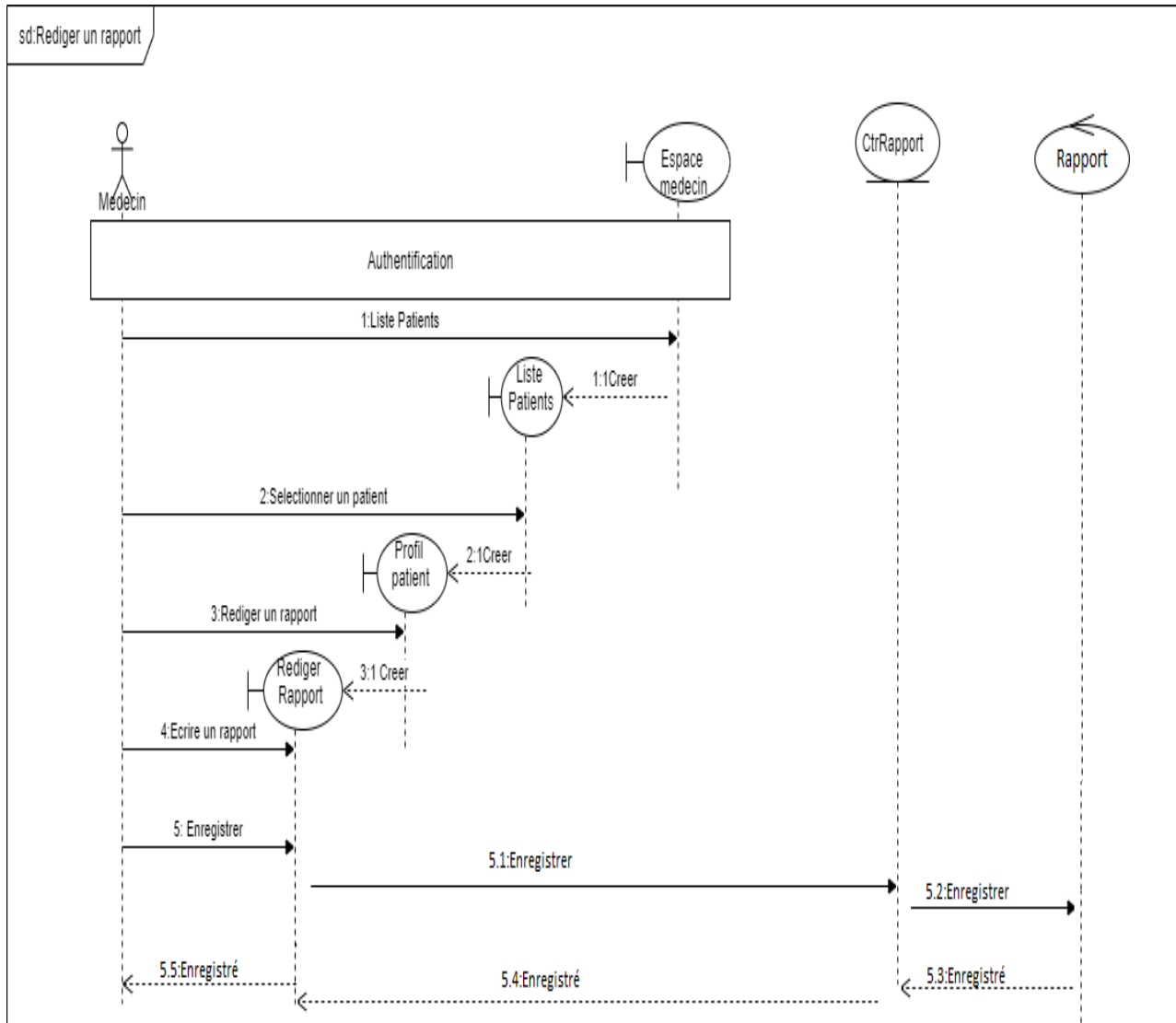


FIGURE 3.16 – Diagramme d'interaction « Rediger un rapport »

3.3.1.8 Diagramme d'interaction «Envoyer un rapport »

La figure 3.17 représente le diagramme d'interaction « Envoyer un rapport ».

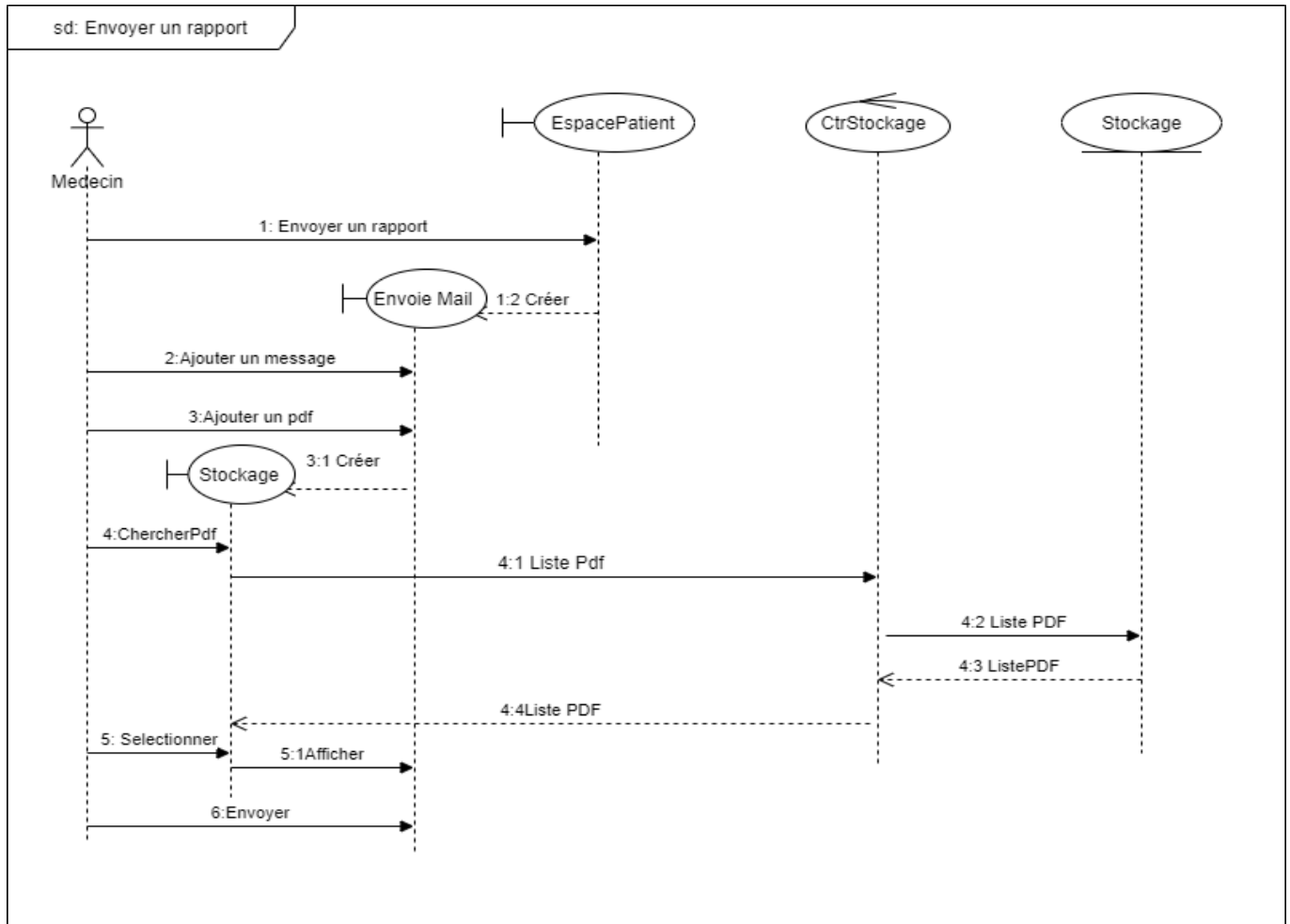


FIGURE 3.17 – Diagramme d'interaction « Envoyer un rapport »

3.3.2 Diagramme de classe

Le diagramme de classes est considéré comme le plus important de la modélisation orientée objet, il est le seul obligatoire lors d'une telle modélisation.

Alors que le diagramme de cas d'utilisation montre un système du point de vue des acteurs, le diagramme de classe en montre la structure interne. Il permet de fournir une représentation abstraite des objets du système qui vont interagir pour réaliser les cas d'utilisation.

Il s'agit d'une vue statique, car on ne tient pas compte du facteur temporel dans le comportement du système. Le diagramme de classe modélise les concepts du domaine d'application ainsi que les concepts internes créés de toutes pièces dans le cadre de l'implémentation d'une application[8].

Les principaux éléments de cette vue statique sont les classes et leurs relations :

- Association ;
- Généralisation ;

3.3.2.1 Diagramme de classe globale de notre application mobile

La figure 3.18 représente le diagramme global de notre système.

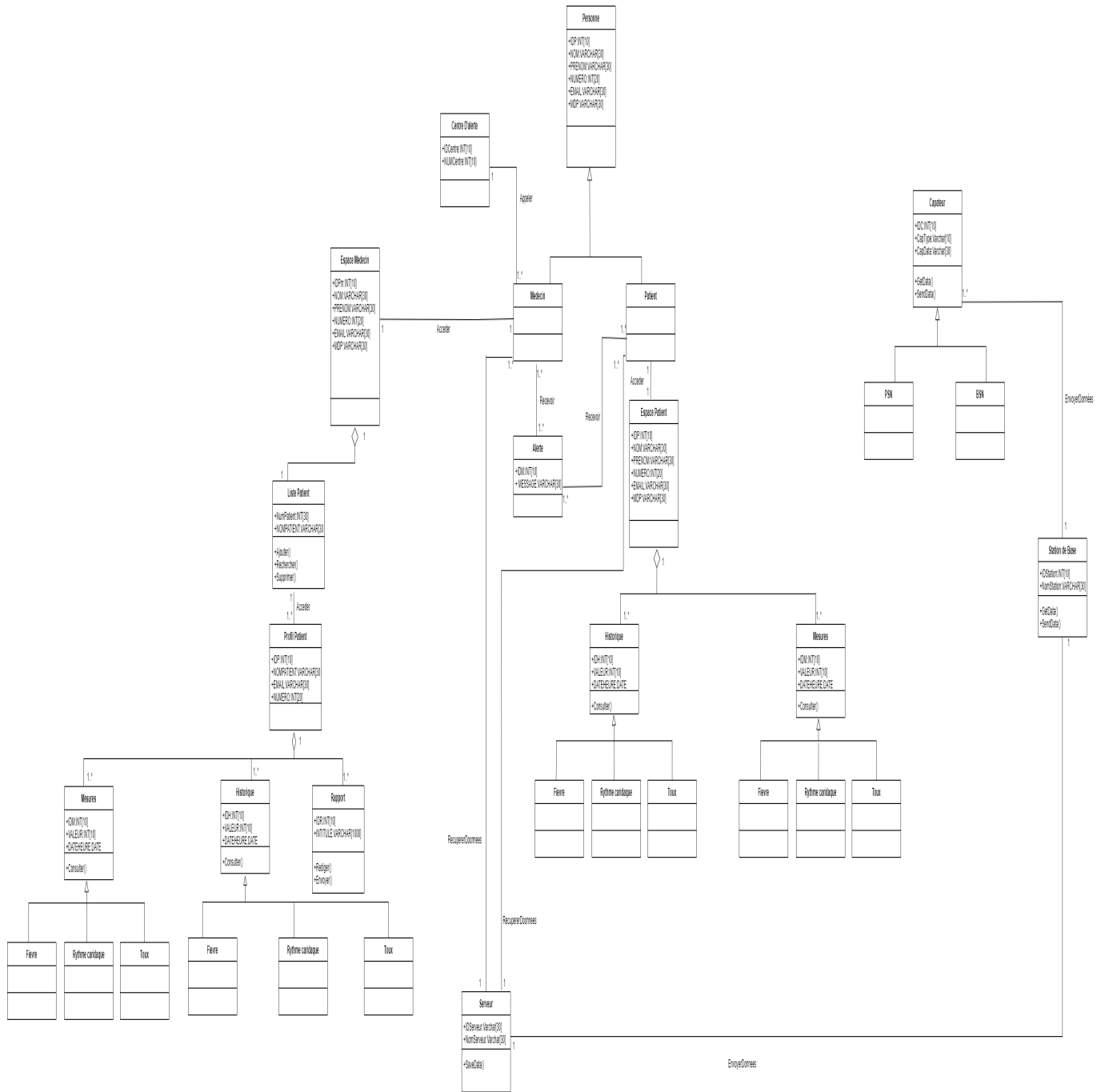


FIGURE 3.18 – Diagramme de classe globale

3.3.3 Dictionnaire de données

Le dictionnaire de données du diagramme de classes illustré ci-dessus est représenté dans le tableau 3.1 suivant :

L'attribut	Désignation	Type
Personne		
IdPersonne	Identifiant d'une personne.	INT
Nom	Nom de la personne.	VARCHAR
Prenom	Prénom de la personne.	VARCHAR
Email	Email de la personne	VARCHAR
Mot de passe	Mot de passe de la personne	VARCHAR
Espace Patient		
NomPatient	Nom du patient	VARCHAR
Age	Age du patient	INT
Email	Email de la personne	VARCHAR
Numero	Numero du patient	INT
Espace Medecin		
NomMedecin	Nom du medecin	VARCHAR
Email	Email du medecin	VARCHAR
Numero	Numero du medecin	INT
Liste Patient		
IDPatient	Identifiant du patient	INT
NOMPATIENT	Nom du patient	VARCHAR
Profil Patient		
NomPatient	Nom du Patient	VARCHAR
Email	Email du Patient	VARCHAR
Age	Age du Patient	INT
Numero	Numero du Patient	INT
Historique		
ID	Identifiant de l'historique	INT
DateHeure	Date et heure des mesures du patient	Date
Valeur	Valeur des mesures du patient	INT
Mesures		
IDM	Identifiant des mesures	INT
DateHeure	Date et heure des mesures du patient	Date
Valeur	Valeur des mesures du patient	INT
Alerte		
IDALERTE	Identifiant de l'alerte	INT
MESSAGE	Message d'alerte	VARCHAR

Centre d'Alerte		
IDCentre	Identifiant du centre d'alerte	INT
NumCentre	Numero du centre d'alrte	INT
Rapport		
IDRapp	Identifiant du rapport	INT
Contenu	Contenu du rapport	VARCHAR
Station de base		
IDStation	Identifiant de la station de base.	INT
NomStation	Nom de la station	VARCHAR
Serveur		
IDServeur	Identifiant du serveur	INT
NomServeur	Nom du serveur	VARCHAR
Capteur		
IDC	Identifiant du capteur	INT
CapType	Type du capteur	VARCHAR
CapData	Données du capteur	VARCHAR

TABLE 3.1 – Dictionnaire de données

Les classes sur lesquelles se porte notre application mobile sont représentées par le tableau 3.2 suivant :

Classe	Description
Centre d'alerte	C'est la classe responsable de la réception de l'appel d'urgence
Alerte	C'est la classe qui permet de prévenir le médecin et le patient d'une anomalie chez le patient
Capteurs	C'est la classe qui permet la réception et l'envoi des différents paramétrés du patient
Personne	C'est la classe mère qui comporte les attributs des classes medecin et patient
Espace Patient	C'est la classe que seul le patient peut y accéder, elle comporte les attributs et méthodes propres au patient
Espace medecin	C'est la classe que seule le medecin peut y acceder, elle comporte les attributs et les méthodes propres au medecin
Liste Patient	C'est la classe qui permet d'ajouter, supprimer ou rechercher un patient
Profil Patient	C'est la classe que seul le patient peut y accéder pour consulter l'état du patient
Mesures	C'est la classe qui comporte les informations concernant les mesures du patient
Historique	C'est la classe qui comporte les informations concernant les mesures enregistrées du patient
Rapport	C'est la classe qui permet la rédaction et l'envoi d'un rapport
Serveur	C'est la classe qui permet de stocker les différentes valeurs des paramétrés du patient
Station de base	C'est la classe qui reçoit et transmet les données des capteurs

TABLE 3.2 – Description des classes de l'application a réaliser

3.3.4 Passage relationnel

Afin de pouvoir implémenter une base de données, il faut pouvoir traduire le modèle conceptuel en modèle logique. Cela signifie qu'il faut pouvoir convertir un modèle UML en modèle relationnel.

Les modèles conceptuels sont suffisamment formels pour que ce passage soit systématisé dans la plupart des cas.

Après avoir appliqué les règles de passage du diagramme de classes au modèle relationnel, nous avons abouti au schéma relationnel de la base de données suivant :

Personne(IDP, Nom, Prenom, Email, Numero, MDP).

Medecin(IDP, Nom, Prenom, Email, Numero, MDP, # IDCentre).

Recevoir(# IDP, # IDM).
Acceder(# IDPm, # IDP).
AccederEspace(# IDPA, # IDP).
ListePatient(NomPatient, # IDPm).
Profile Patient(IDP, NomPatient, Email, Numero, # NumPatient).
Rapport(# IDR, Intitule, # IDP).
Historique(IDH, Valeur, DateHeure, # IDP).
Mesures(IDM, Valeur, DateHeure, # IDP).
Capteur(IDC, CapType, CapData, # IDStation).
EnvoyerDonnées(IDS, IDStation).

3.4 Conclusion

Tout au long de ce chapitre, nous avons proposé une modélisation avec UML à travers les diagrammes de séquence, diagrammes d'interactions et diagramme de classe qu'on a traduit en modèle relationnel afin de concevoir le schéma de la base de données. Le prochain chapitre fera l'objet de l'implémentation et réalisation de notre application mobile.

Chapitre 4

Réalisation

4.1 Introduction

A ce stade du processus les cas d'utilisation sont bien cernés, le problème a été analysé en profondeur et nous avons défini une conception appropriée aux besoins de l'application. Nous pouvons alors entreprendre l'implémentation, en présentant l'architecture de la solution proposée, les outils de développement, et quelques captures d'écran sur l'application réalisée.

4.2 Outils et langages de développement

A fin de réaliser et d'implémenter notre application nous avons eu recours à un ensemble d'outils et de langages de développement.

4.2.1 Outils de développement

1. Visual Paradigm :

Visual Paradigm est un logiciel de création de diagrammes dans le cadre d'une programmation. Tout en un, il possède plusieurs options permettant une large possibilité de modélisation en ULM [9]. Ses principales fonctionnalités sont :

- Modélisation : le logiciel Visual Paradigm offre de nombreux outils pour créer différents types de schémas comme les diagrammes d'exigences et de cas d'utilisation. Il possède bon nombre de navigateurs permettant de personnaliser chaque élément.
- Analyse et manipulation de codes sources : Visual Paradigm permet de générer des codes sources en divers langages comme le Java ou C++ à partir du modèle créé. Inversement, il permet de produire un modèle à partir de codes sources.

2. Photoshop :

Photoshop est un logiciel de traitement d'images. A ce titre il permet en autres de :

- Corriger les défauts d'une photographie ;
- Retirer un élément indésirable sur une image ;
- Ajuster une photo trop sombre (sous-exposée) ou trop claire(sur-exposée) ;
- Créer de nouvelles images ;
- Montage de plusieurs images.

3. Android Studio :

Android Studio est un nouvel environnement pour le développement et programmation entièrement intégré qui a été récemment lancé par Google pour les systèmes Android, il a été conçu pour fournir un environnement de développement et une alternative à Eclipse qui est l'IDE le plus utilisé. Android est gratuit, pour les constructeurs d'appareils souhaitant l'utiliser, et partiellement opensource. Concrètement, ce système d'exploitation s'appuie sur un noyau Linux optimisé pour un usage mobile, et sur une machine virtuelle Java assez fortement modifiée, nommée Dalvik JVM. Celle-ci n'exécute pas les (.class) habituels, mais des

fichiers portant l'extension (.dex), compilés différemment et optimisés par le SDK Android. Le développement se fait donc en Java, mais sur cette JVM spécifique à Android. Il n'est, du fait, pas possible d'utiliser n'importe quelle librairie Java dans une application Android [10].

Afin de développer des applications sous Android, un ensemble d'outils est nécessaire :

— **Le JDK de Oracle :**

Java Development Kit, distribué par Oracle désigne un ensemble de bibliothèques logicielles de base du langage de programmation Java, ainsi que les outils avec lesquels le code Java peut être compilé, transformé en bytecode destiné à la machine virtuelle Java.

— **Le SDK (Software Development Kit) Android :**

Software Development Kit ou Kit de développement logiciel est un ensemble de fichiers et d'applications distribué par Google afin de permettre la compilation d'application pour le système d'exploitation Android.

— **AVD Manager :**

Un appareil virtuel Android (AVD) est une configuration d'émulateur qui permet aux développeurs de tester l'application en simulant les capacités réelles de l'appareil. Nous pouvons configurer l'AVD en spécifiant les options matérielles et logicielles. Le gestionnaire AVD permet de créer et de gérer facilement l'AVD grâce à son interface graphique. Nous pouvons créer autant d'AVD que nécessaire, en fonction des types d'appareils que nous voulons tester[10].

— **SQLite :**

SQLite est une bibliothèque en langage C qui implémente un petit moteur de base de données SQL rapide, autonome, hautement fiable et complet. SQLite est le moteur de base de données le plus utilisé au monde. Il est intégré à tous les téléphones mobiles et à la plupart des ordinateurs et est intégré à d'innombrables autres applications que les gens utilisent chaque jour. Plus d'information... Le format de fichier SQLite est stable, multiplateforme et rétrocompatible et les développeurs s'engagent à le conserver au moins jusqu'en 2050. Les fichiers de base de données SQLite sont couramment utilisés comme conteneurs pour transférer un contenu riche entre les systèmes et comme format d'archivage à long terme des données. Il existe plus de 1 billion de bases de données SQLite en cours d'utilisation.

Le code source de SQLite est dans le domaine public et est gratuit pour tout le monde à utiliser dans n'importe quel but[11].

4. **DB Browser :**

DB Browser for SQLite est un outil visuel open source de haute qualité pour créer, concevoir et modifier des fichiers de base de données compatibles avec SQLite, il est destiné aux utilisateurs et aux développeurs qui souhaitent créer, rechercher et modifier des bases de données.

Il utilise une interface familière de type feuille de calcul et les commandes SQL complexes n'ont pas besoin d'être apprises[11].

4.2.2 Langages de développement

1. JAVA :

JAVA est un langage de programmation orienté objet, développé par Sun Microsystems. Il permet de créer des logiciels compatibles avec de nombreux systèmes d'exploitation (Windows, Linux, Macintosh, Solaris). Java donne aussi la possibilité de développer des programmes pour téléphones portables et assistants personnels. Enfin, ce langage peut être utilisé sur internet pour des petites applications intégrées à la page web (applet) ou encore comme langage serveur (JSP)[12].

2. SQL (Structured Query Language :)

Le langage SQL (Structured Query Language) est un langage informatique utilisé pour exploiter des bases de données. Il permet de façon générale la définition, la manipulation et le contrôle de sécurité de données. Dans la pratique, le langage SQL est utilisé pour créer des tables, ajouter des enregistrements sous forme de lignes, interroger une base de données, la mettre à jour, ou encore gérer les droits d'utilisateurs de cette base de données. Il est bien supporté par la très grande majorité des systèmes de gestion de base de données (SGBD). Créé au début des années 1970 par Donald D. Chamberlin et Raymond F. Boyce, tous deux chez IBM, le langage SQL est aujourd'hui reconnu comme une norme internationale[11].

3. XML (Extensible Markup Language :)

Le XML, pour Extensible Markup Language, désigne un langage informatique (ou métalangage pour être plus précis). Ce langage de description a pour mission de formaliser des données textuelles. Il s'agit, en quelque sorte, d'une version améliorée du langage HTML avec la création illimitée de nouvelles balises. Comme le langage HTML, le XML permet la mise en forme de documents via l'utilisation de balises. Développé et standardisé par le World Wide Web Consortium à la fin des années 1990, il répondait à l'objectif de définition d'un langage simple, facile à mettre en application.

Le XML se classe dans la catégorie des langages de description (il n'est ni un langage de programmation, ni un langage de requêtes). Il est donc naturellement utilisé pour décrire des données en s'appuyant sur des balises et des règles personnalisables[12].

4. JSON (JavaScript Object Notation) :

Il s'agit d'un format de fichier open-standard permettant de stocker les données de manière organisée et lisible par un humain tout en y facilitant l'accès. Les données sont présentées dans un format textuel constitué de paires " key / value ".

Etroitement lié à JavaScript, ce format peut toutefois être généré et lu par la plupart des langages de programmation. Cette universalité lui a permis de devenir une façon très populaire de stocker, organiser, lire et partager des données dans les applications et services web[12].

4.3 Réalisation

Dans cette partie, nous allons présenter quelques interfaces, sous forme d'un guide utilisateur. La figure 4.1 ci dessous, représente l'interface d'accueil.



FIGURE 4.1 – Interface d'Accueil

Pour accéder à notre application l'utilisateur doit s'authentifier en précisant son identité (Médecin ou Patient).

La figure 4.2 ci dessous, représente l'interface choix de l'utilisateur.

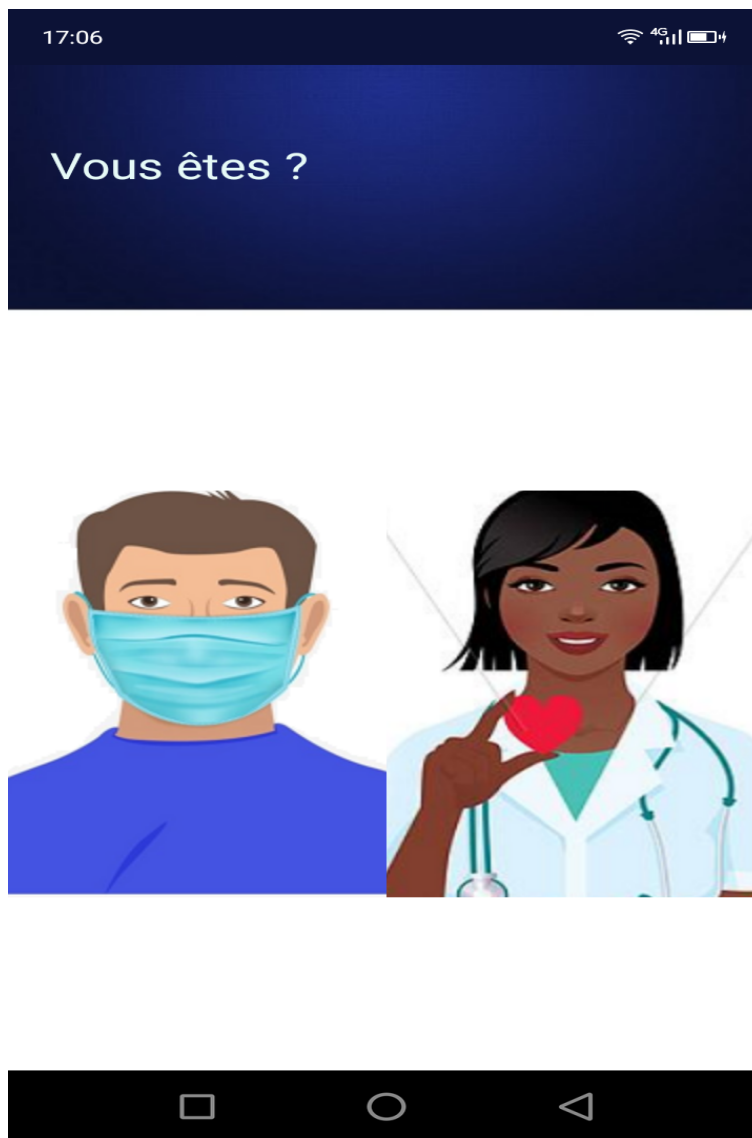


FIGURE 4.2 – Interface Choix de l'utilisateur

Comme toute application, la sécurité d'accès est nécessaire. La figure 4.3 donne l'interface à travers laquelle l'utilisateur s'identifie, il saisit son login et son password puis le serveur vérifie la validité de ces informations.

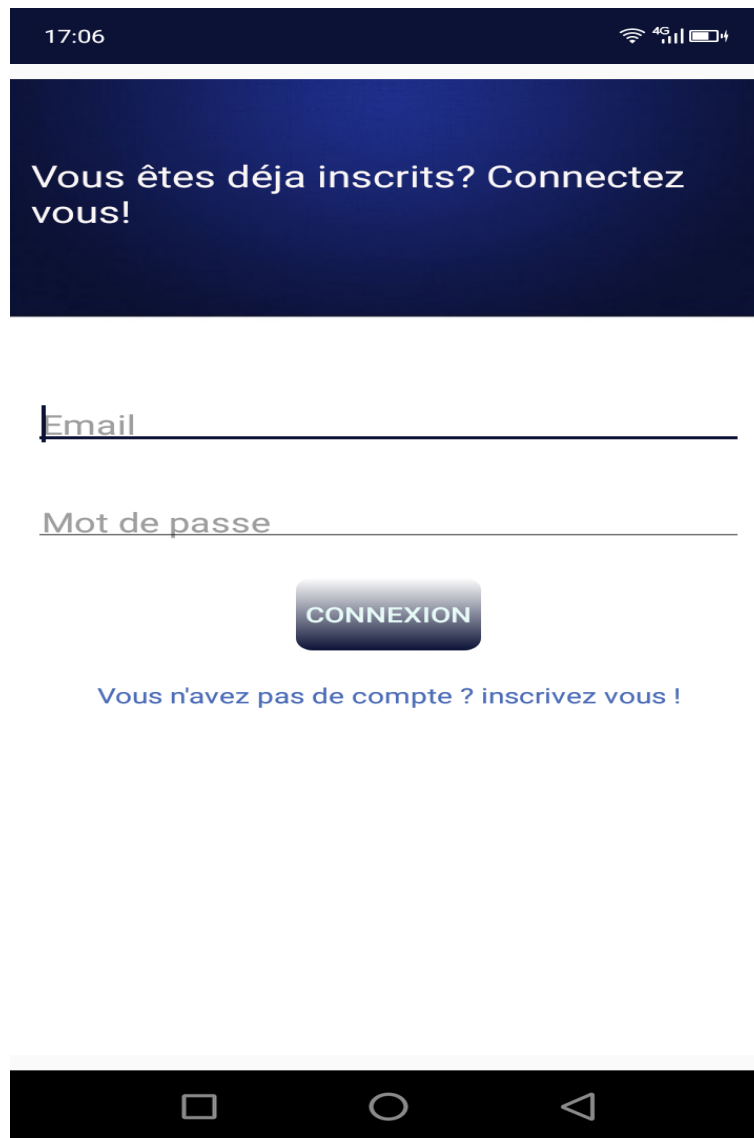
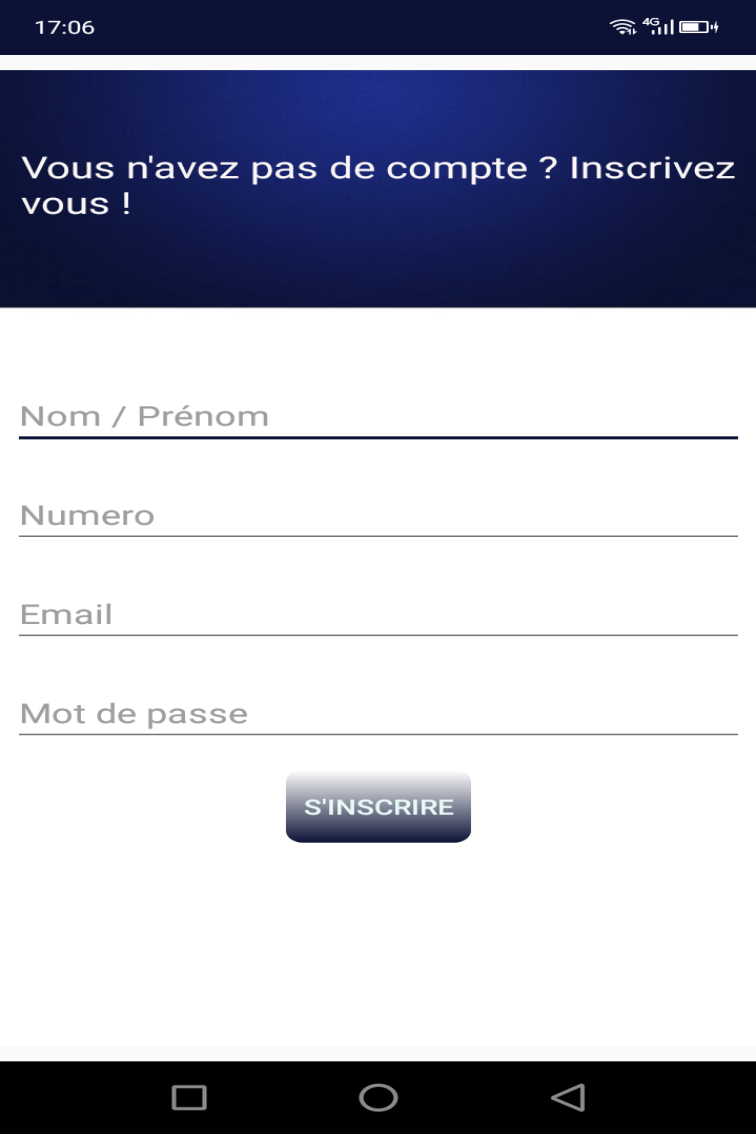


FIGURE 4.3 – Interface d'Authentification

Si l'utilisateur est nouveau il sera dirigé vers une page d'inscriptions à fin de créer un compte , les informations sont vérifiées par le serveur, en cas d'erreur de saisie un message d'erreur sera affiché.

La figure 4.4 ci dessous, représente l'interface d'Inscription.



17:06

Vous n'avez pas de compte ? Inscrivez vous !

Nom / Prénom

Numero

Email

Mot de passe

S'INSCRIRE

FIGURE 4.4 – Interface d'Inscriptions

1. Interfaces dédiées au patient :

Une fois le patient authentifié, il est dirigé vers son espace. La figure 4.5 représente l'espace dédié au patient.

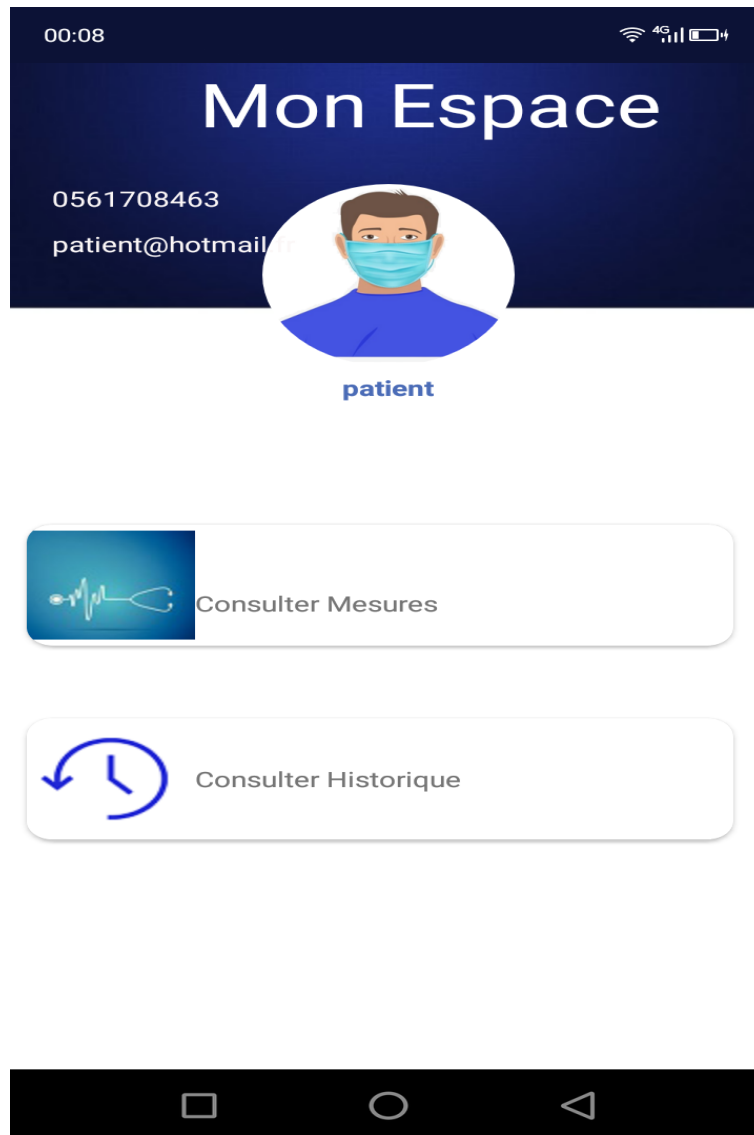


FIGURE 4.5 – Espace Patient

Si un patient souhaite consulter ses différentes mesures enregistrées, il est dirigé vers l'interface représentée par la figue 4.6 ci après, il pourra ensuite choisir le type de mesures qu'il souhaite consulter.

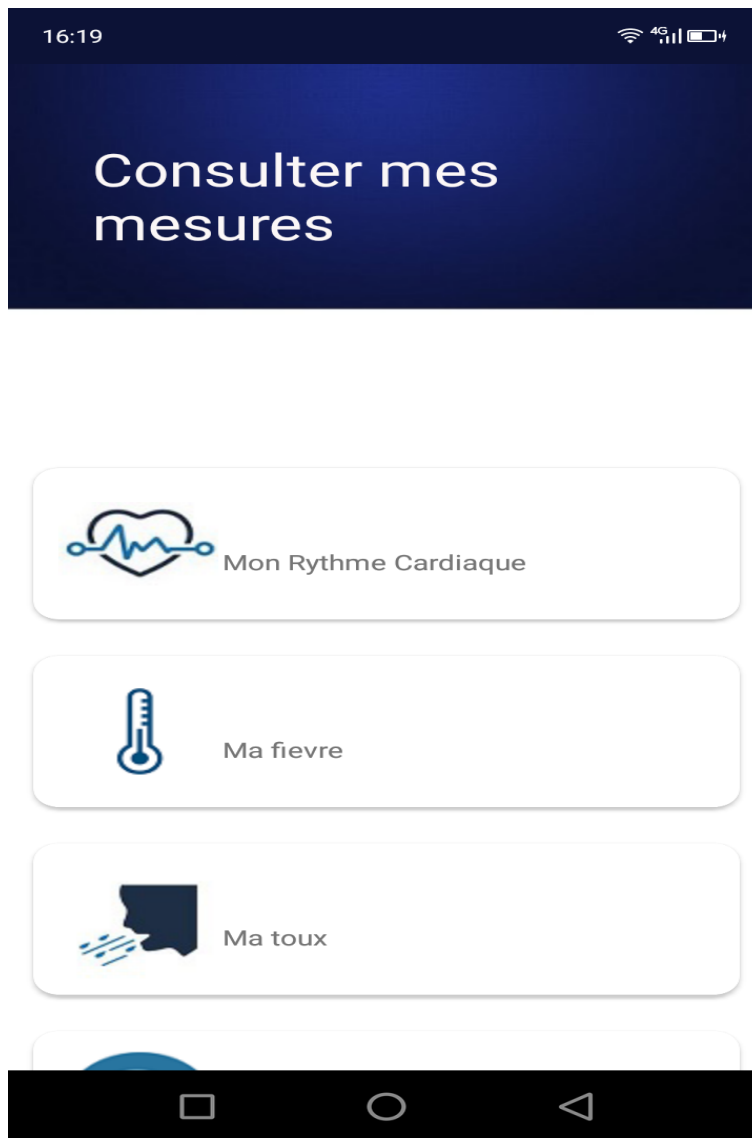


FIGURE 4.6 – Interface Mesures du patient

La figure 4.7 représente l'interface des mesures de la Fièvre, les valeurs de la fièvre du patient sont enregistrées selon la date et l'heure.

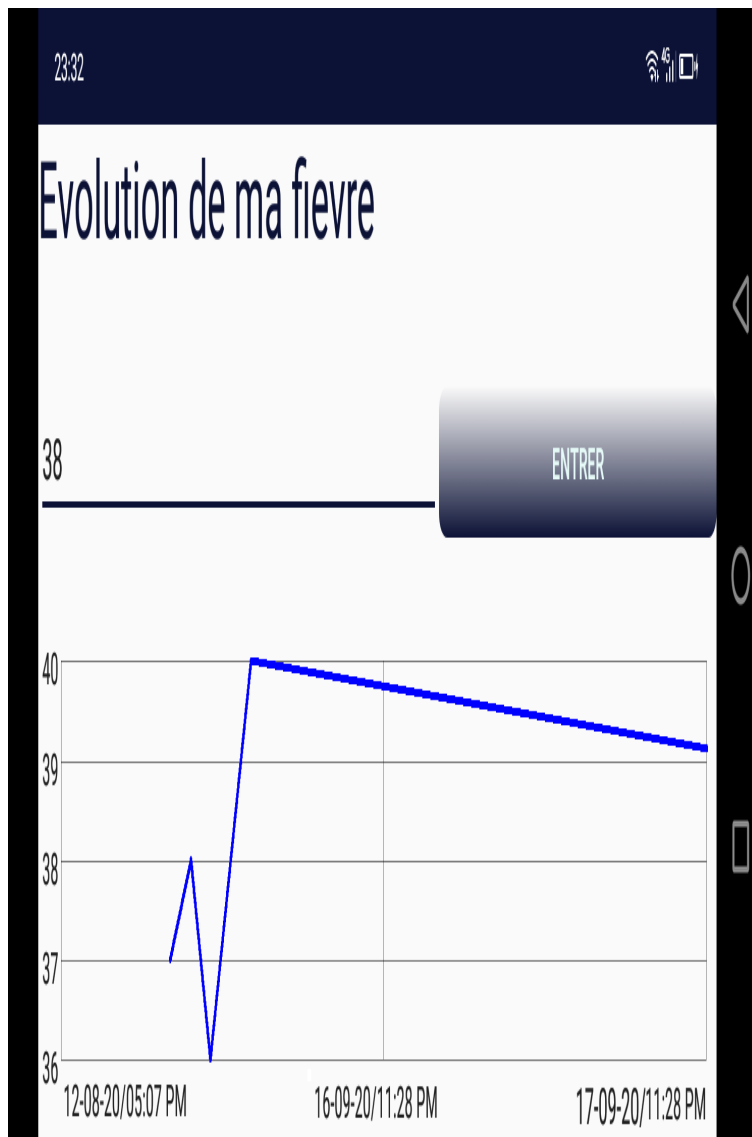


FIGURE 4.7 – Interface Mesures de la Fièvre

La figure 4.8 représente l'interface des mesures du Rythme Cardiaque, les valeurs du rythme cardiaque du patient sont enregistrées selon la date et l'heure.

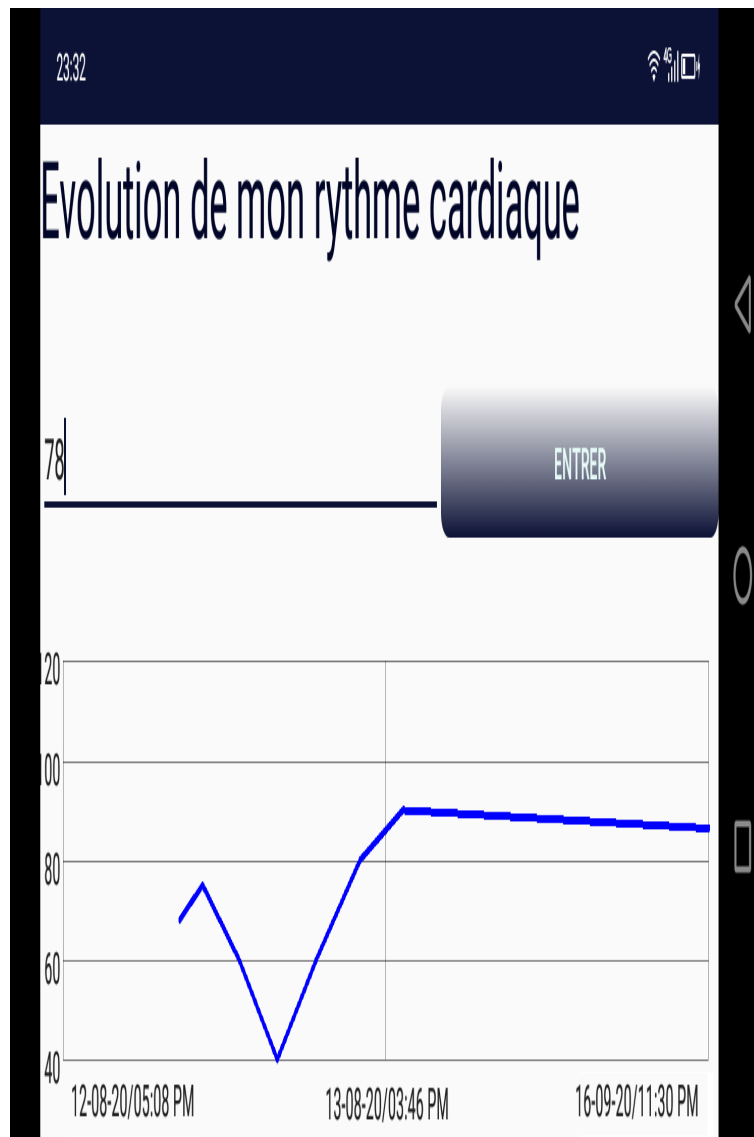


FIGURE 4.8 – Interface Mesures du Rythme Cardiaque

Si les paramètres enregistrés ne rentrent pas dans les mesures normales, la couleur du graphe devient rouge et une notification d'alerte est affichée (Pour le Médecin et le Patient), en cliquant dessus un appel d'urgence sera effectué (Figure 4.9).

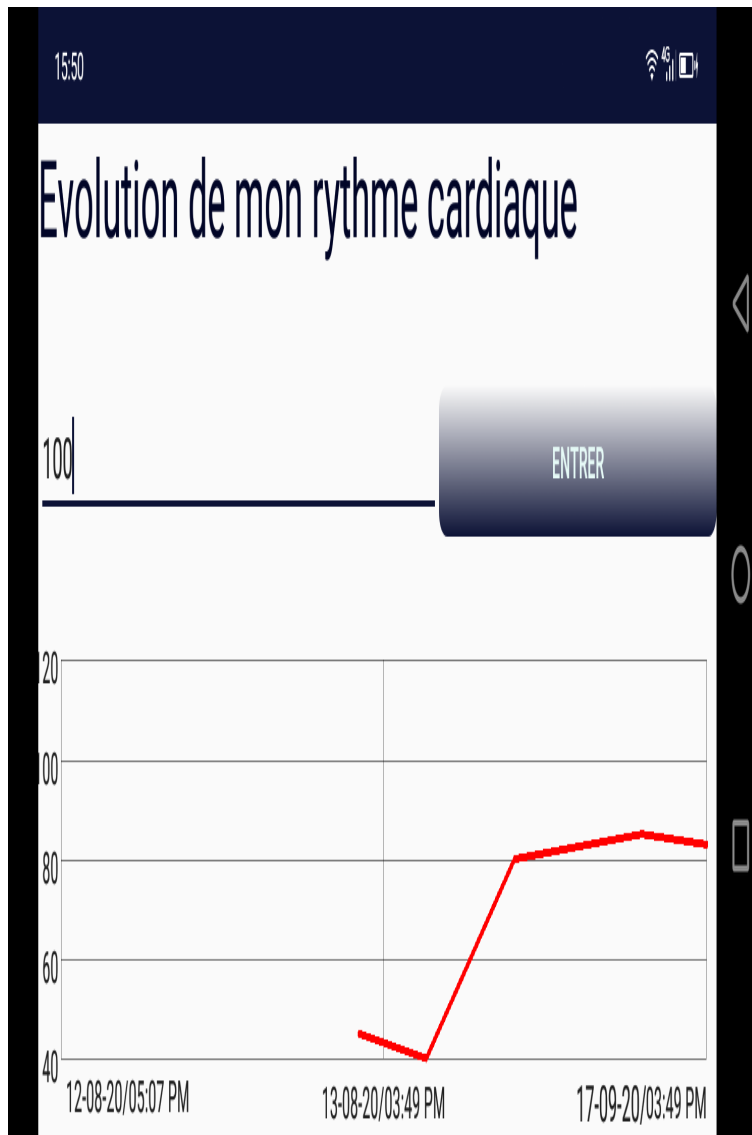


FIGURE 4.9 – Interface Mesures de la fièvre

La figure 4.10 représente la notification d'alerte envoyée en cas d'une anomalie enregistrée.

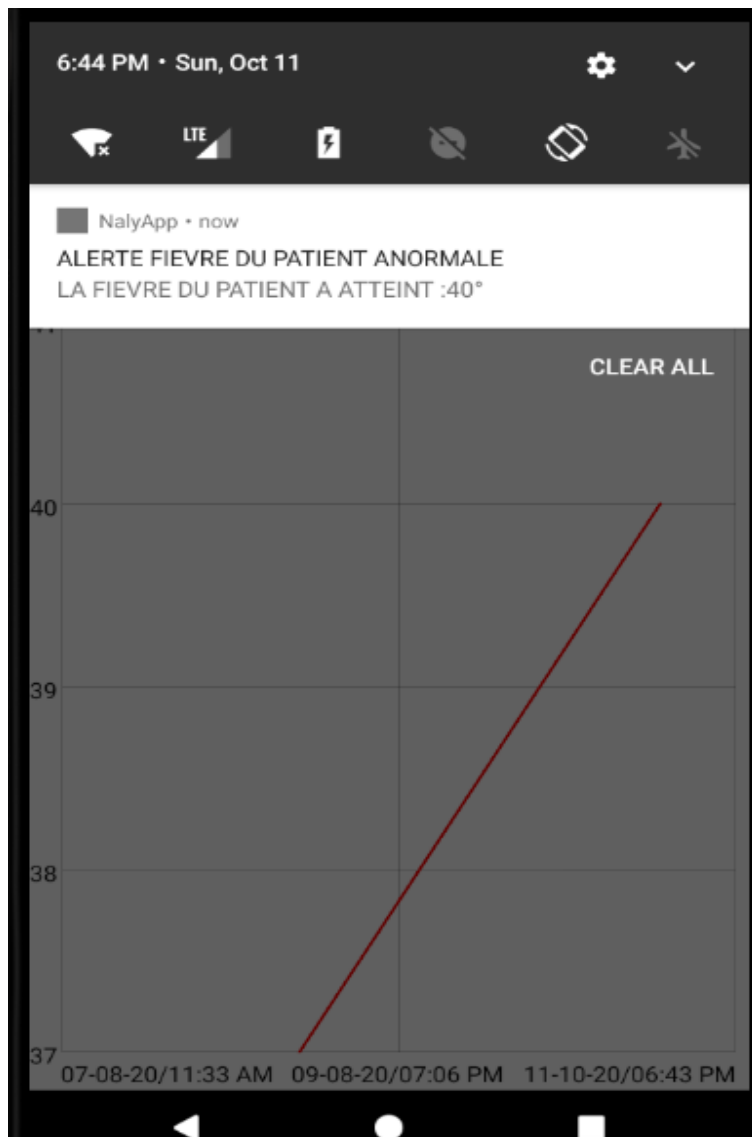


FIGURE 4.10 – Notification d'Alerte

Si un patient souhaite consulter l'évolution de son état de santé une page d'historique de toutes ses mesures est à sa portée. La figure 4.11 représente l'interface choix de l'historique.

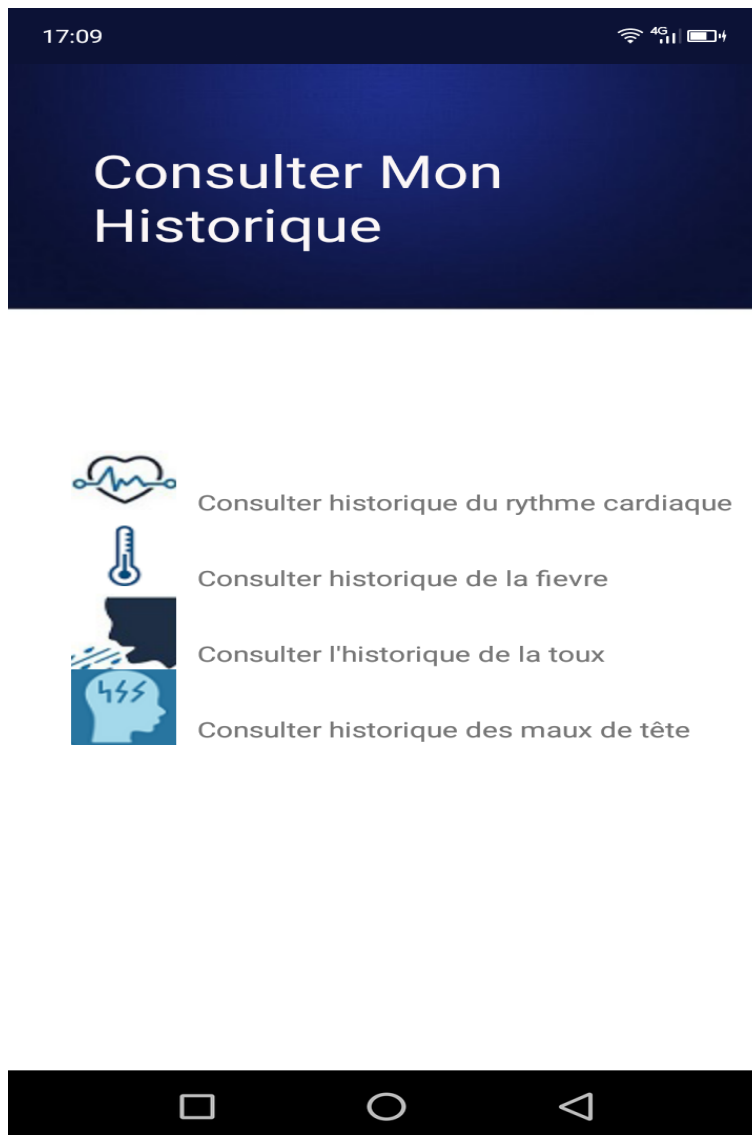
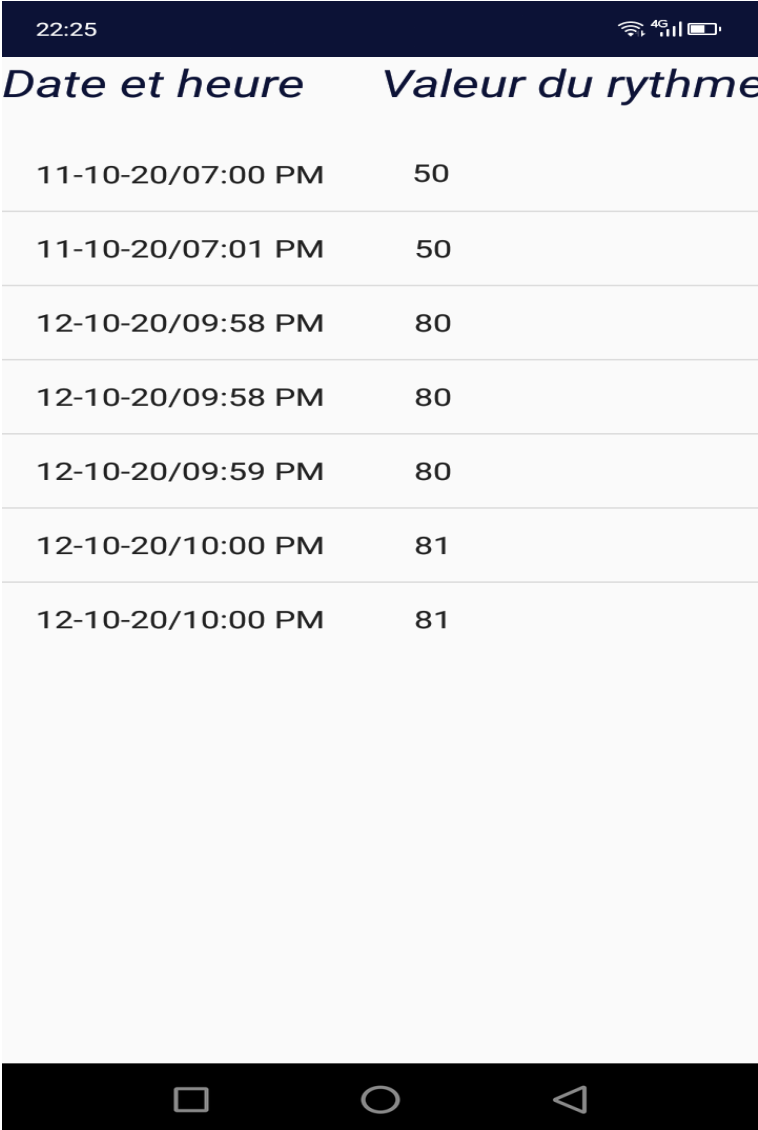


FIGURE 4.11 – Interface Choix des mesures

Une fois le patient aurait effectué son choix, les valeurs enregistrées sont affichées. La figure 4.12 représente l'interface de l'historique du Rythme Cardiaque.



<i>Date et heure</i>	<i>Valeur du rythme</i>
11-10-20/07:00 PM	50
11-10-20/07:01 PM	50
12-10-20/09:58 PM	80
12-10-20/09:58 PM	80
12-10-20/09:59 PM	80
12-10-20/10:00 PM	81
12-10-20/10:00 PM	81

FIGURE 4.12 – Interface Historique du Rythme Cardiaque

2. Interfaces dédiées au Médecin :

Après avoir effectué son authentification le médecin est dirigé vers son espace qui est représenté par la figure 4.13.

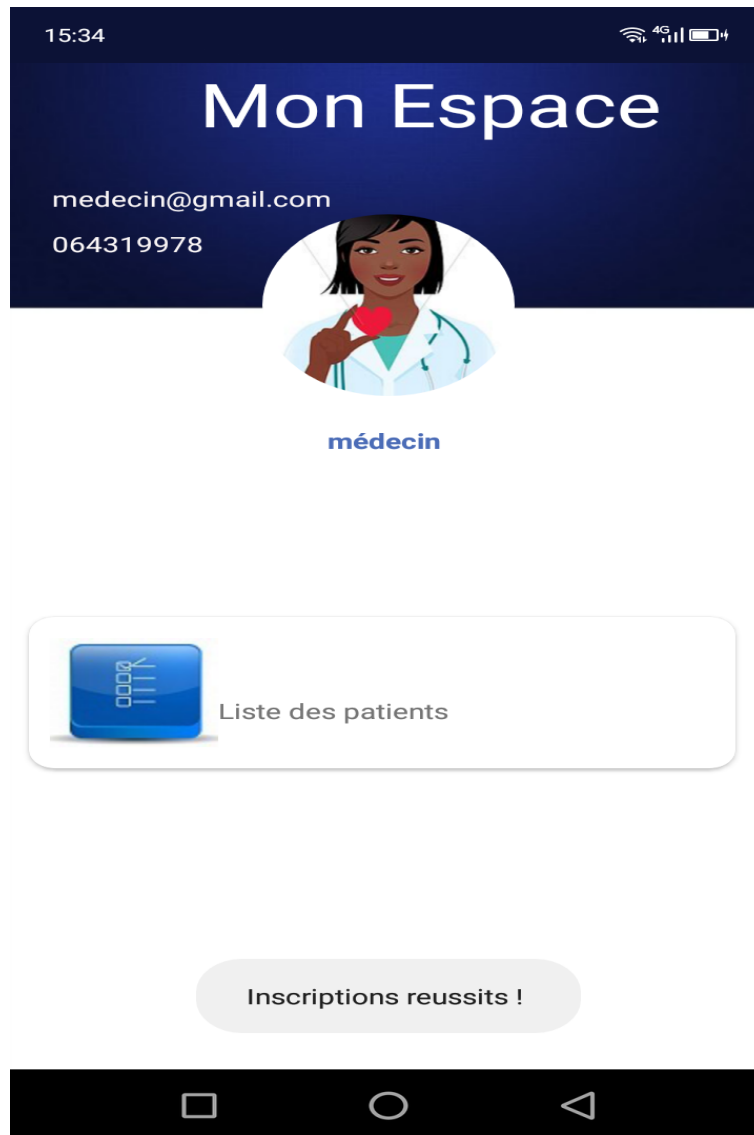


FIGURE 4.13 – Interface Espace Médecin

En cliquant sur la liste des patients, le médecin est dirigé vers une interface où il pourra ajouter, supprimer ou rechercher un patient. Pour ajouter un patient il doit être déjà inscrit sur notre application, dans le cas contraire, un message d'erreur sera affiché. Un patient déjà ajouté ne peut pas être ajouté une seconde fois !

La figure 4.14 représente l'interface d'Ajout.

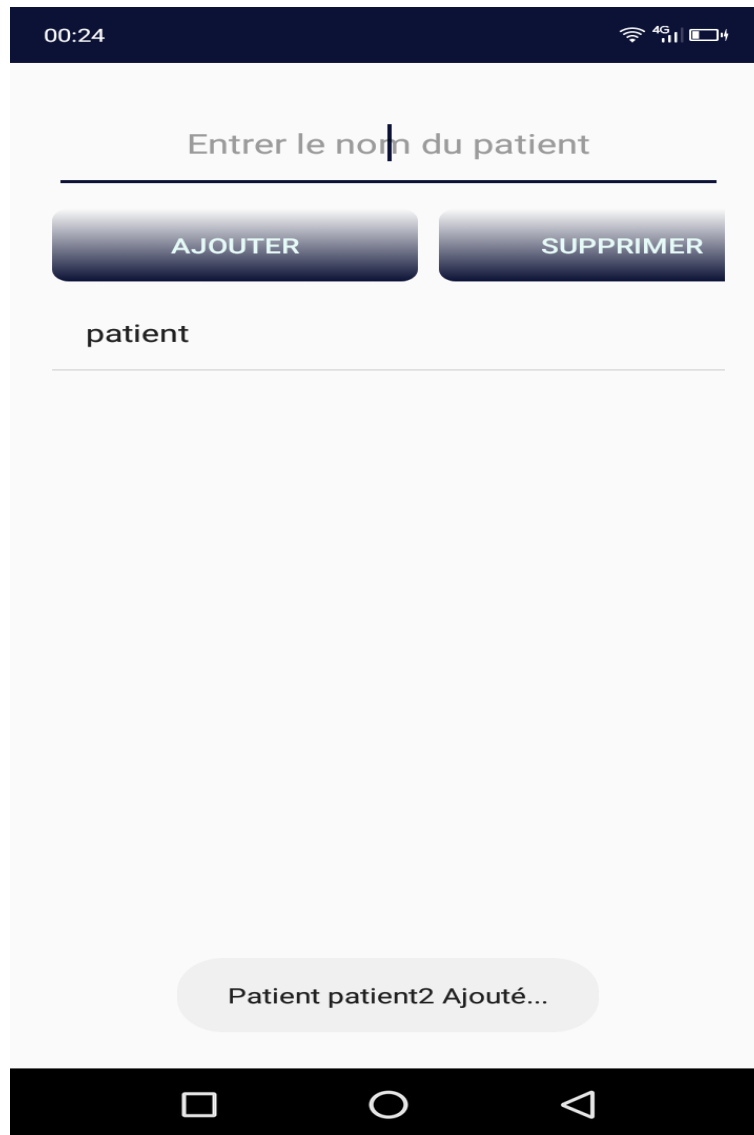


FIGURE 4.14 – Interface Ajouter un patient

La figure 4.15 représente l'interface de suppression.

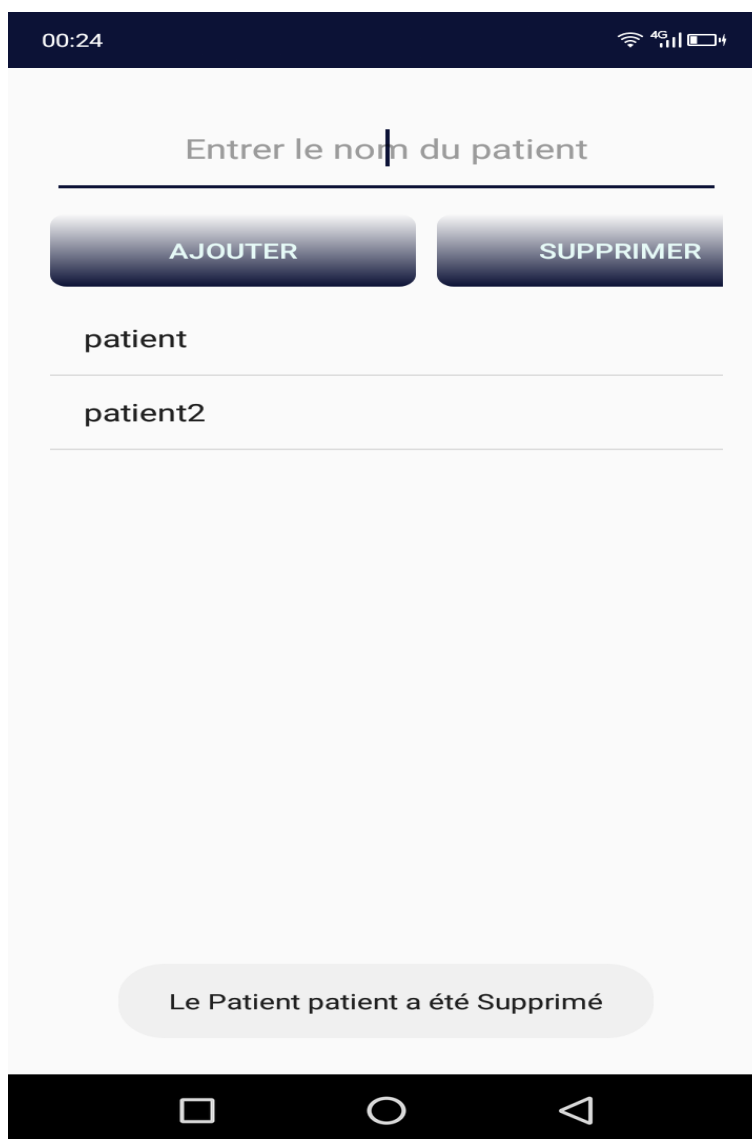


FIGURE 4.15 – Interface Supprimer un patient

En cliquant sur l'un des noms de ses patients, le médecin est dirigé vers le profil de son patient où il peut effectuer plusieurs actions !

La figure 4.16 représente l'interface Profil du patient.



FIGURE 4.16 – Interface Profil du Patient

Un medecin peut consulter les mesures de son patient pour voir l'état de sa santé. La figure 4.17 représente l'interface Choix des mesures.



FIGURE 4.17 – Interface choix des mesures

Une fois le médecin aurait effectué son choix des mesures à consulter, il est dirigé vers une interface où les mesures enregistrées du patient sont affichées selon la date et l'heure. La figure 4.18 représente l'interface des mesures du Rythme Cardiaque.



FIGURE 4.18 – Interface mesures du Rythme Cardiaque

La figure 4.19 représente l'interface des mesures de la Fièvre.



FIGURE 4.19 – Interface mesures de la Fièvre

Un médecin peut consulter l'historique de toutes les mesures sauvegardées du patient afin de voir l'évolution de son état de santé. La figure 4.20 représente l'interface choix de l'historique.

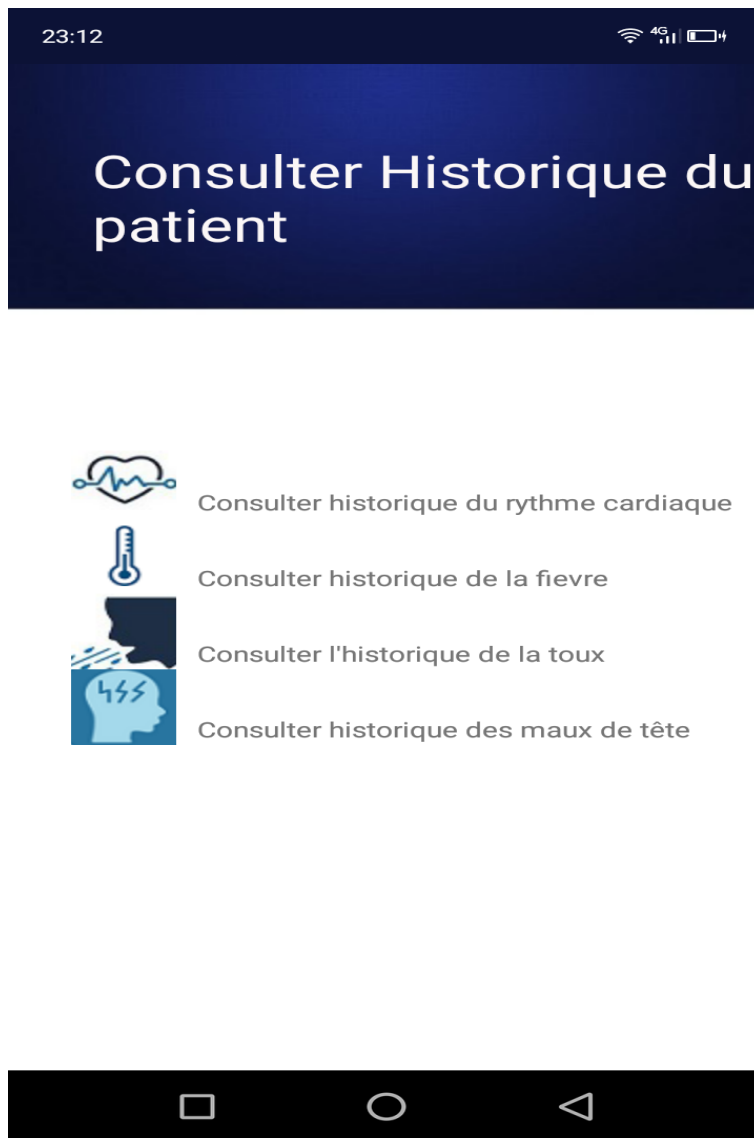
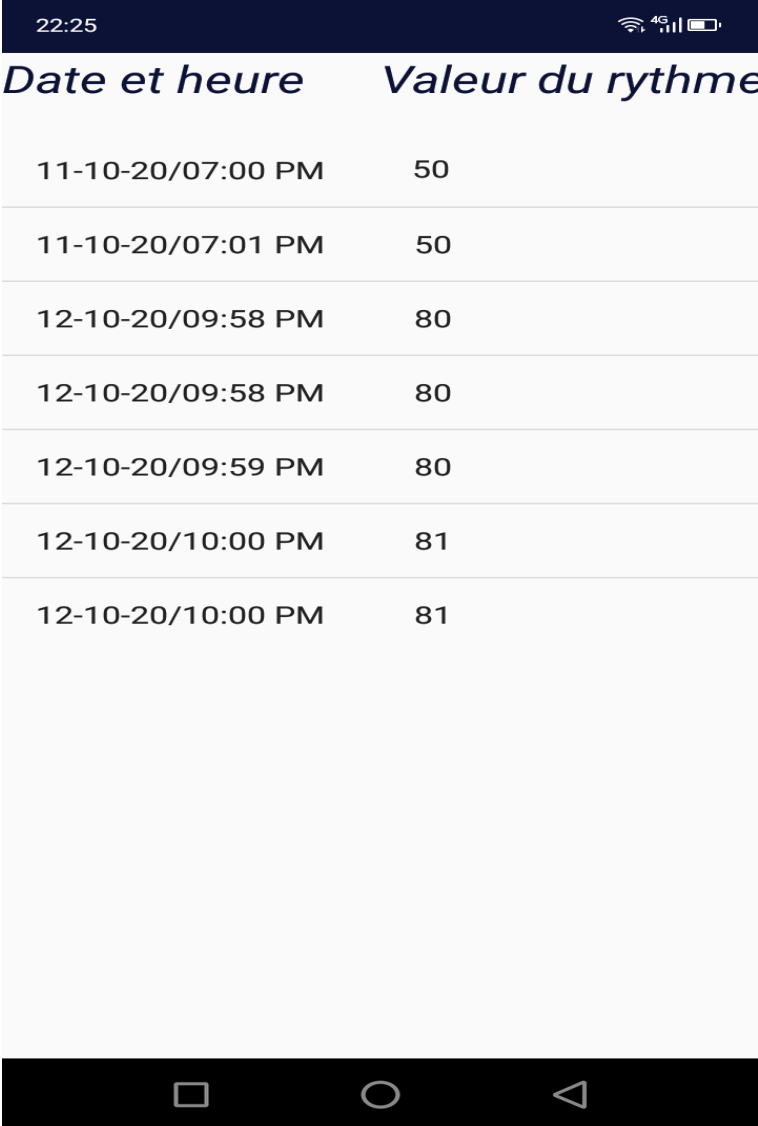


FIGURE 4.20 – Interface choix de l'Historique

Une fois que le médecin aurait effectué son choix de l'historique à consulter, il est dirigé vers une interface où les valeurs sont affichées selon la date et l'heure. La figure 4.21 représente l'interface historique du Rythme Cardiaque.



<i>Date et heure</i>	<i>Valeur du rythme</i>
11-10-20/07:00 PM	50
11-10-20/07:01 PM	50
12-10-20/09:58 PM	80
12-10-20/09:58 PM	80
12-10-20/09:59 PM	80
12-10-20/10:00 PM	81
12-10-20/10:00 PM	81

FIGURE 4.21 – Interface Historique du Rythme cardiaque

Une fois que le médecin ait consulté l'état de son patient, il peut donc rédiger son rapport. La figure 4.22 représente l'interface Rédiger un rapport.



FIGURE 4.22 – Interface Rédiger un rapport

Une fois que le rapport rédigé et enregistré il peut être envoyé par mail. La figure 4.23 représente l'interface Envoyer un Rapport.



FIGURE 4.23 – Interface Envoyer un rapport

4.4 Conclusion

Ce chapitre a été consacré à la phase de réalisation. Cette phase est le fruit de nos efforts tout au long de la durée du projet.

Dans ce chapitre, nous avons décrit brièvement le processus de réalisation de notre application en spécifiant l'environnement, les outils et les langages de développement associés à notre système. En effet, nous avons achevé l'implémentation tout en respectant la conception élaborée, de ce fait on peut considérer notre travail comme étant achevé.

Conclusion et Perspectives

Au cours de ce mémoire, nous avons présenté les différentes étapes de la conception et réalisation de notre application pour le suivi des patients à distance, qui n'est qu'une initiation au système que nous souhaitons réaliser, par faute de moyens la réalisation complète de ce dernier a été impossible.

Nous avons commencé la conception en utilisant le formalisme UML, et cela, en suivant le cycle de vie du processus de développement UP. La réalisation a été faite avec le logiciel ANDROID STUDIO qui inclue tous les outils nécessaires pour la réalisation d'une application mobile. La rédaction de ce document a été faite avec le langage de structuration des documents LATEX.

Le présent travail, nous a permis de mettre en pratique toutes nos connaissances théoriques acquises durant notre parcours universitaire, mais aussi d'enrichir d'avantage notre expérience notamment dans les différents outils et langages dédiés à la programmation mobile. Nous avons retenu également que la réalisation d'une application mobile demande une bonne organisation et une cohérence entre les différents acteurs du projet.

Comme perspective, nous espérons voir notre application évoluer, ceci en mettant en place tout le matériel nécessaire (Capteurs, Serveurs, etc) afin d'aboutir à une application utilisable qui améliorera la qualité de vie des patients et du personnel médical.

Nous espérons enfin que le travail que nous avons effectué a été à la hauteur de la confiance qui nous a été donnée.

Bibliographie

- [1] A.Abdo, "Lecturer of Systems and information", Faculty of computers and information Mansoura University, (2019)
- [2] A.BOULEMTAFES, "Mobilité contextuelle dans un système de surveillance médicale à distance", MAGISTERE, Université A. MIRA - BEJAIA, (2016)
- [3] A.ABENYAHY, Etude d'une méthodologie pour la construction d'un système de télésurveillance médicale, thèse de doctorat, Université de Technologie de Belfort-Montbéliard, (2015).
- [4] H.MSHALI, "Services e-Santé sensibles au contexte dans les espaces intelligents", pour obtenir le grade de DOCTEUR, L'UNIVERSITÉ DE BORDEAUX École Doctorale de Mathématiques et Informatique, (2017).
- [5] R.Amri, R.Badi, "Système d'Information Pour la Télésurveillance Médicale", Mémoire de fin d'étude Master 2, Faculté des Mathématiques, d'Informatique et des Sciences de la matière Université de Guelma
- [6] <https://www.who.int/fr/emergencies/diseases/novel-coronavirus-2019/>, consulté le : 19/04/2020.
- [7] UP : Unified Process - FTPftp-developpez.com/sabricole/UP.PDF, consulté le : 10/02/2020.
- [8] [<https://laurent-audibert.developpez.com/Cours-UML>], consulté le : 17/02/2020.
- [9] [<https://www.commentcamarche.net/>], consulté le 20/07/2020.
- [10] [<https://developer.android.com/>], consulté le : 30/07/2020.
- [11] [<https://www.sqlite.org/about.html>], consulté le : 30/07/2020.
- [12] [<https://www.journaldunet.fr/>], consulté le : 1/08/2020.
- [13] 2012-10 Article SIH - Degoulet-Venot.pdf, consulté le : 10/12/2019.
- [14] Y.Prié, "Processus de conception de SI 2/3 Processus Unifié", Département d'informatique, Université Claude Bernard Lyon1, (2011-2012).
- [15] <https://www.semanticscholar.org/paper/Informatisation-d'une-mediatheque-a-travers-la-UML-Sbihi>, consulté le : 20/02/2020.

RÉSUMÉ

Ce mémoire de fin d'études, présente la conception et réalisation d'une application mobile "Android" pour le suivi des patients à distance.

Le système de santé Algérien a plus besoin d'informatiser le suivi des patients. Pour atteindre cet objectif, il nous a été proposé de concevoir et d'implémenter une application assurant le suivi à distance de l'état de santé des patients principalement les personnes âgées ainsi faciliter la vie de ces derniers.

Pour ce faire, nous avons choisi de modéliser notre système avec le formalise UML, notre choix s'est porté sur ce dernier par rapport à sa simplicité et sa performance en matière de conception.

La réalisation a été faite avec Android Studio qui inclue tous les outils nécessaires à la réalisation d'une application mobile.

Mots Clés : Télésurveillance médicale, Suivi patient, Informatique médicale, Application mobile, UML, UP.

ABSTRACT

This thesis presents the design and production of an "Android" mobile application for remote patient monitoring.

The Algerian health system no longer needs to computerize patient monitoring. To achieve this objective, we were asked to design and implement an application ensuring the remote monitoring of the state of health of patients and thus making life easier for the latter, mainly the elderly.

To do this, we have chosen to model our system with the UML formalization, our choice fell on the latter in relation to its simplicity and its performance in terms of design.

The realization was made with Android Studio which includes all the tools necessary for the realization of a mobile application.

Key Words : Medical remote monitoring, Patient monitoring, Medical IT, Mobile application, UML, UP.