

الجمهورية الجزائرية الديمقراطية الشعبية

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**

وزارة التعليم العالي والبحث العلمي

**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**

جامعة أبي بكر بلقايد - تلمسان

Université Aboubakr Belkaïd – Tlemcen –

Faculté de TECHNOLOGIE



## **MEMOIRE**

Présenté pour l'obtention du **diplôme** de **MASTER**

**En** : Télécommunications

**Spécialité** : Réseaux Et Télécommunications

**Par** :

**SENOUCI BEREKSI** Wahiba & **SELADJI** Mohammed Chakib

**Sujet**

**Création Du Module De Gestion Du Dossier Médical Du Patient  
(DMP) Sur Odoo**

Soutenu publiquement, le 30 / 06 / 2022 , devant le jury composé de :

Mr. ZERROUKI Hadj	MCA	Université de Tlemcen	Président
Mme. BOUSSALAH Fayza	MCB	Université de Tlemcen	Examinateur
Mr. MOUSSAOUI Djilali	MCB	Université de Tlemcen	Encadreur

Année universitaire : 2021 /2022

## Remerciements

*Tout d'abord, nous remercions Allah le tout puissant de nous avoir donné le courage et la volonté d'achever ce travail et sans Lequel il n'aurait jamais été accompli.*

*Toute notre reconnaissance et notre gratitude vont à notre Encadreur de mémoire  
Mr. MOUSSAOUI Djilali pour sa présence, son écoute.*

*Nous remercions les membres de jury pour l'honneur qu'ils nous ont fait en acceptant  
d'examiner notre travail Et de l'enrichir par leurs propositions.*

*Enfin, nous remercions toutes nos familles pour leurs soutiens, encouragements, et leurs  
conseils.*

## *Dédicaces*

*Je dédie ce travail à :*

*Mes très chers parents, pour tous leurs sacrifices, leur amour, leur tendresse, leur soutien et leurs prières tout au long de mes études.*

*A la personne qui a ajouté du plus à ma vie et qui Dieu me l'a envoyé pour partager mon bonheur.*

*A toute ma famille et mes proches.*

*A Assia et à tous mes amis sans exception.*

*Je vous dédie ce travail et vous souhaite un avenir radieux et plein de belles promesses.*

*Toutes les personnes qui ont cru en moi et qui me donnent l'envie d'aller de l'avant, je vous remercie tous.*

*Votre soutien et vos encouragements me donnent la force de continuer.*

*Senouci Bereksi Wahiba*

## *Dédicaces*

*Je dédie ce modeste travail à :*

*Toute ma famille*

*Mes amis*

*Seladji Mohammed Chakib*

# Liste Des Figures

<i>Figure I 1</i> : Modules de base d'Odoo développés par Odoo SA .....	10
<i>Figure I 2</i> : Vue du CMS Odoo.....	13
<i>Figure I 3</i> : L'architecture à trois niveaux d'Odoo .....	15
<i>Figure I 4</i> : Structure du répertoire des modules Odoo.....	16
<i>Figure I 5</i> : Exemple d'un fichier <code>__manifest__.py</code> .....	17
<i>Figure I 6</i> : Exemple d'un modèle d'héritage.....	18
<i>Figure I 7</i> : Exemple d'une vue en liste (en haut) et d'une vue en formulaire (en bas) .....	19
<i>Figure I 8</i> : Exemple de code QWeb Template avec héritage.....	19
<i>Figure II 9</i> : Découpage du Système d'Information hospitalier (SIH).....	24
<i>Figure II 10</i> : Vue générale de DMP .....	25
<i>Figure III 11</i> : Diagramme de cas d'utilisation « gestion des utilisateurs ».....	35
<i>Figure III 12</i> : Diagramme de cas d'utilisation « gestion des diagnostics, d'ordonnance et des consultations ».....	36
<i>Figure III 13</i> : Diagramme de cas d'utilisation « créer un dossier patient » .....	38
<i>Figure III 14</i> : Diagramme de cas d'utilisation « prendre des RDV ».....	39
<i>Figure III 15</i> : Diagramme de séquence prendre rendez-vous.....	41
<i>Figure III 16</i> : Diagramme de séquence de recherche du médecin.....	42
<i>Figure III 17</i> : Diagramme de séquence de création du compte patient.....	43
<i>Figure III 18</i> : Diagramme de séquence « gestion des utilisateurs ».....	43
<i>Figure III 19</i> : Diagramme de séquence de gestion des diagnostics et d'ordonnance.....	44
<i>Figure III 20</i> : Diagramme de classe.....	45
<i>Figure III 21</i> : Structure d'un module ERP.....	48

## *Liste Des Tableaux*

Tableau III 1 : Identification des acteurs et des cas d'utilisations .....	34
Tableau III 2 : description de cas d'utilisation « Gestion des utilisateurs » .....	35
Tableau III 3 : description de cas d'utilisation « Gestion des diagnostics » .....	37
Tableau III 4 : description de cas d'utilisation de création du dossier patient .....	38
Tableau III 5 : description du cas d'utilisation de prise de rendez-vous .....	39
Tableau III 6 : description du cas d'utilisation de recherche de médecin .....	40

# Tables Des Matières

Remerciements.....	1
Dédicaces .....	2
Liste Des Figures .....	4
Liste Des Tableaux .....	5
Tables Des Matières.....	6
Introduction générale .....	10
Chapitre I: Odoo	
I.    Introduction : .....	11
II.   Odoo : .....	12
1.  Histoire :.....	12
2.  Open source : .....	13
3.  Fonctionnalité : .....	14
4.  Modularité et personnalisation :.....	15
5.  Caractéristiques techniques et guide de développement : .....	16
a.  Architecture : .....	16
b.  Créer un module Odoo :.....	17
c.  Modèles :.....	19
d.  Vues et fichiers de données : .....	20
e.  Contrôleurs de Web : .....	21
f.  Javascript :.....	22
III.  Conclusion : .....	22
Chapitre II : Dossier Médical De Patient.	
I.    Introduction : .....	23
II.   Hospital :.....	23
1.  Définition d'un hôpital :.....	23
2.  Définition d'un système d'information hospitalier : .....	24
3.  Objectifs du système d'information hospitalier :.....	24
4.  Les composantes d'un SIH : .....	24
III.  Dossier Médicale De Patient (DMP) : .....	25
1.  Définition du DMP : .....	25
2.  Vue générale du DMP : .....	26
3.  Historique du dossier médical : .....	27
4.  Bénéfice de dossier médical : .....	27

1)	L'intérêt pour les professionnels de santé :.....	28
2)	L'intérêt pour le patient :.....	28
a.	Modalités de la demande :.....	28
b.	Délai et mode de communication :.....	28
c.	Des règles particulières :.....	29
IV.	Le dossier médical informatisé :.....	29
1.	Définition :.....	29
2.	Utilité d'un dossier patient informatisé :.....	30
3.	Les informations à recueillir dans le dossier :.....	31
V.	Conclusion :.....	32
<b>Chapitre III :</b>		
I.	Introduction :.....	33
II.	Conception :.....	33
1.	Définition de UML :.....	33
2.	Diagramme de cas d'utilisation du module :.....	34
a.	Les diagrammes des cas d'utilisation de l'application :.....	35
b.	Diagramme de cas d'utilisation de directeur :.....	35
c.	Diagramme de cas d'utilisation de médecin :.....	37
d.	Diagramme de cas d'utilisation de secrétaire :.....	38
4.	Diagramme de séquence :.....	41
I.	Diagramme de séquence de prise de rendez-vous :.....	42
II.	Diagramme de séquence de recherche du médecin :.....	43
III.	Diagramme de séquence « créer un dossier patient » :.....	44
IV.	Diagramme de séquence « gestion des utilisateurs » :.....	44
V.	Diagramme de séquence de gestion des diagnostics et d'ordonnance :.....	45
5.	Diagramme de classe :.....	45
III.	Réalisation :.....	47
1.	Introduction :.....	47
2.	Les outils utilisés :.....	47
a)	Odoo v13 :.....	47
b)	PostgreSQL :.....	47
c)	PyCharm :.....	48
d)	Python :.....	48
e)	XML :.....	49
f)	Qweb :.....	50
3.	Présentation de l'application modulaire :.....	50
A.	Présentation des interfaces :.....	52
a)	Gestion des patients :.....	52

b) Gestion des rendez-vous : .....	52
c) Inventaire des médicaments :.....	53
d) Gestion des ordonnances : .....	54
e) Gestion des diagnostics :.....	54
f) Médecins enregistrés : .....	55
B. Les accès pour les différents acteurs : .....	55
1. Directeur : .....	55
2. Médecine : .....	55
3. Secrétaire : .....	55
IV. Conclusion : .....	56
Conclusion Générale .....	57
Bibliographies.....	58
<i>Résumé</i> .....	61

# *Introduction Générale*

## Introduction générale

Les ordinateurs sont au centre de toute activité humaine aujourd'hui. Qu'il s'agisse d'un simple achat en magasin, du paiement d'une facture, d'une transaction bancaire, des systèmes d'information simples ou complexes sont concernés. Quel que soit le domaine, nous avons toujours besoin d'un système de traitement automatisé de l'information.

Avant l'invention de l'ordinateur, les informations de divers types d'entreprises et d'entreprises étaient saisies manuellement dans des fiches. Ces formulaires sont archivés et chaque fois qu'une information est nécessaire, le personnel parcourt les archives une par une, ce qui entraîne des problèmes tels que la perte de temps et la dégradation du support d'information.

Comme tous les domaines, le secteur médical a décidé d'informatiser ses services pour faciliter ses tâches. La clinique faisait face à un certain nombre de problèmes, à savoir un accès difficile à l'analyse des patients, la lenteur de l'évolution des dossiers des patients, la disparition de certains documents archivés et des difficultés dans le traitement de l'information, compte tenu de la croissance de sa population hospitalière, et la quantité importante d'informations traitées à son niveau de service est encore une gestion manuelle prédominante. On a alors rapidement ressenti le besoin d'informatiser les services rendus aux usagers et de tenir un dossier médical pour chaque patient de la clinique, c'est pourquoi elle veut gérer toutes ses informations. Autour du même système d'information, un progiciel de gestion intégré en abrégé ERP est utilisé.

Enterprise Resource Planning (ERP) est un système d'information composé de manière modulaire de multiples applications liées aux différentes fonctions d'une entreprise, sachant que ces modules sont interdépendants entre eux et définis par des systèmes d'automatisation prédéfinis partageant une base de données unique et identique.

C'est dans ce cadre que s'inscrit notre projet de fin d'études qui a pour objectif de mettre en place un module de gestion du patient, pour ce faire, nous avons organisé notre travail en trois chapitres :

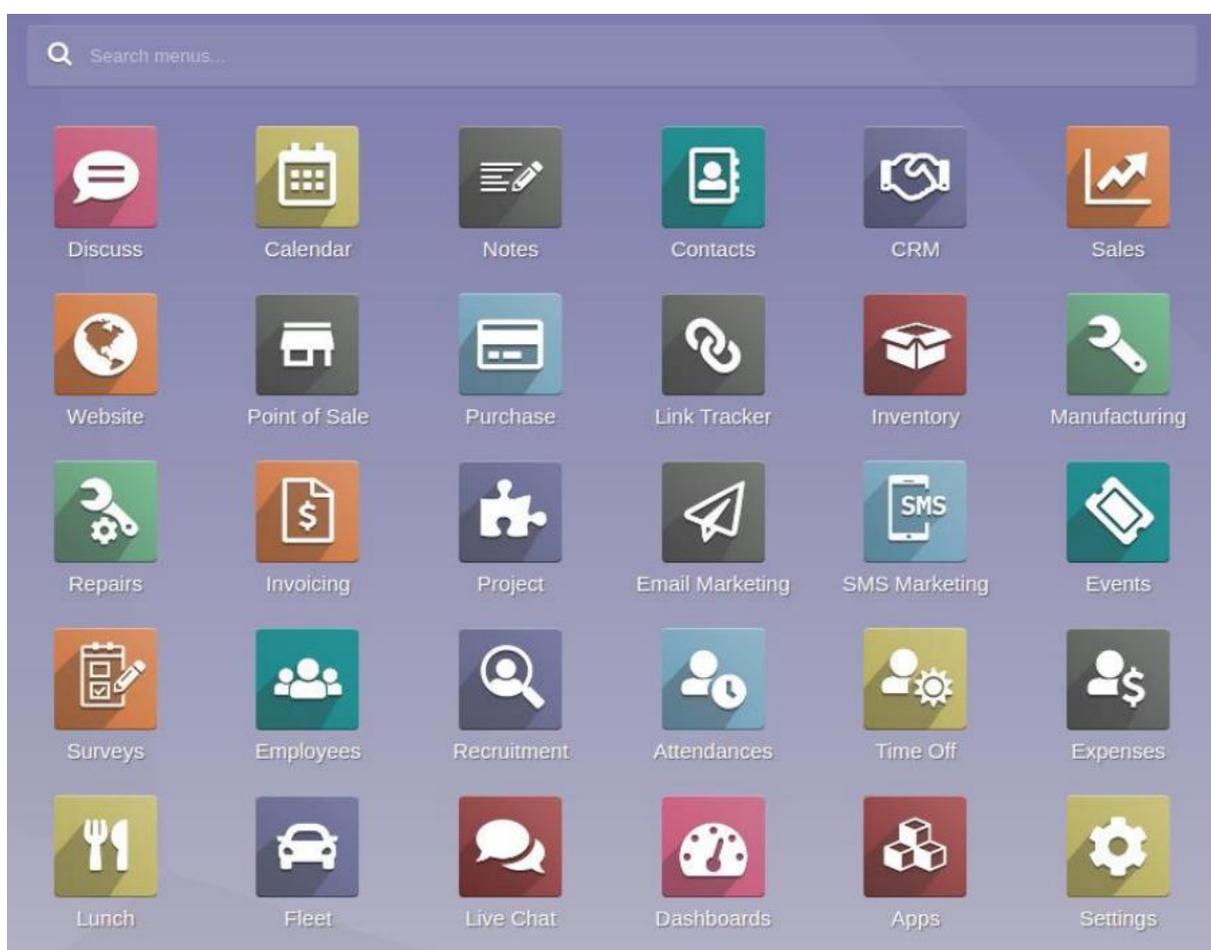
- ◇ Le premier chapitre intitulé : « Odoo », est consacré à la présentation de Odoo ;
- ◇ Le deuxième chapitre intitulé : « Dossier Médical De Patient », s'est accentué sur une présentation générale de Dossier Médical De Patient ;

Le troisième et dernier chapitre intitulé : « Conception et réalisation du module Odoo » il présente les étapes de la conception et les résultats obtenus après implémentation, les différentes interfaces et les outils utilisés pour son développement

# *Chapitre I: Odo*

## I. Introduction :

Odoo est une application commerciale open source, qui comprend à la fois un système de planification des ressources de l'entreprise (ERP) et un système de gestion de la relation client (CRM). Il est conçu comme un système modulaire capable de répondre aux besoins des petites et moyennes entreprises et organisations. Chaque installation d'Odoo est construite sur une sélection de modules adaptés aux besoins des clients. Un module est généralement une application unique ou une fonctionnalité qui est ajoutée à la plateforme de base d'Odoo. Parmi les exemples de ces modules, citons un module d'événement pour gérer les événements et les inscriptions, un module de vente pour faciliter le processus de vente et un module de site Web pour créer un site Web visible par le public pour le client (figure 1). La modularité permet à chaque client de ne disposer que des fonctionnalités dont il a besoin tout en étant facilement gérable et extensible à l'avenir.[1]



**Figure I 1 :** Modules de base d'Odoo développés par Odoo SA.

## II. Odoo :

Odoo est une suite open source de logiciels de gestion d'entreprise développée par une entreprise belge société belge Odoo SA. [2]

### 1. Histoire :

Odoo n'a pas toujours porté le nom susmentionné. Elle a débuté sous le nom de TinyERP en 2005. TinyERP était un logiciel créé par un entrepreneur belge ambitieux qui avait pour objectif de dépasser SAP dans le domaine des ERP. Trois ans plus tard, il a été rebaptisé OpenERP pour éviter que le logiciel ne soit considéré à tort comme minuscule. Le nom OpenERP s'explique très probablement par la nature open source du logiciel, mais cela n'a pas été confirmé par le créateur. OpenERP a continué à se développer au cours des années suivantes en tant qu'entreprise et en tant que suite logicielle. [3]

En tant qu'entreprise, OpenERP a connu une croissance régulière de 100% par an et a reçu le titre de huitième entreprise à la croissance la plus rapide de Belgique par Deloitte en 2013, car elle a connu une croissance du chiffre d'affaires de 1549% au cours des cinq années précédentes. En tant que produit logiciel, OpenERP a continué à se développer et à améliorer sa technologie. Il est devenu le logiciel de gestion le plus installé au monde avec 1 000 installations par jour et jusqu'à 60 nouveaux modules publiés chaque mois [3]

OpenERP a continué à se développer et à étendre son logiciel au-delà de la portée des systèmes ERP traditionnels en implémentant des fonctionnalités telles que CMS, e-commerce, Point of Sale, un moteur de Business Intelligence intégré, et bien plus encore dans leur nouvelle version 8. À ce stade, OpenERP était beaucoup plus qu'une minuscule ou un simple système ERP, de sorte que le nom de la société et du logiciel a été changé en Odoo pour se distancer du terme "ERP". [3]

Aujourd'hui, Odoo est à une version 14 avec un total de 7 millions d'utilisateurs dans le monde. Odoo, en tant qu'entreprise, emploie plus de 1 700 personnes. La communauté Odoo a contribué à plus de 16 000 modules qui couvrent une grande variété de besoins commerciaux. [2]

## 2. Open source :

La définition de l'open source n'est pas aussi claire qu'il n'y paraît à première vue. Elle signifie plus que l'accès au code source et il existe de multiples points de vue concurrents sur ce qui constitue l'open source. Parfois, le terme "open source" est associé à l'expression "free and open source" qui fait référence à la signification de "Free software" par GNU et la Free Software Foundation. "Logiciel libre" signifie un logiciel qui respecte la liberté et la communauté des utilisateurs. La Free Software Foundation (2021) en résume la signification : " - - - Grossièrement, cela signifie que les utilisateurs ont la liberté d'exécuter, de copier, de distribuer, d'étudier, de modifier et d'améliorer le logiciel. "

L'open source et le logiciel libre n'ont pas la même signification selon la Free Software Foundation (2021). Le logiciel libre a un poids idéologique et philosophique et ne concerne pas seulement l'accès au code source mais aussi les libertés et les droits des utilisateurs du logiciel. En pratique, la différence entre le logiciel libre et le logiciel open source est que le logiciel open source a généralement des critères plus souples et n'inclut pas toujours le copyleft. [4]

Le copyleft signifie qu'un logiciel qui est sous licence copyleft libre peut être modifié et étendu tant que la nouvelle version du logiciel est publiée sous une licence copyleft libre également. [5]. Odoo est souvent commercialisé et désigné comme un logiciel open source mais il s'inscrit dans l'idée de logiciel libre car il est actuellement sous licence GNU Lesser General Public License (LGPLv3).

Le créateur et fondateur d'Odoo avait le rêve de diriger le marché de la gestion d'entreprise avec un logiciel entièrement open source et, dès le début, Odoo a fourni sa suite logicielle de base en tant que source libre et gratuite. Jusqu'à la version 8, Odoo a été publié avec une licence publique générale GNU Affero (AGPLv3). À partir de la version 9, Odoo a fait la transition vers un modèle de noyau ouvert, qui fournit une version propriétaire basée sur un abonnement premium en plus d'une version open source. La version open source a été publiée sous la licence GNU Lesser General Public License (LGPLv3) avec ce changement [6].

En 2013, l'Odoo Community Association, OCA, a été créée en tant qu'organisation à but non lucratif. L'énoncé de mission de l'Odoo Community Association dans leurs mots est le suivant : " - - - Établir et soutenir une communauté Open Source et collaborative pour le développement et la promotion des fonctionnalités et modules Odoo. ".[7].

### 3. Fonctionnalité :

Odoo est une application web qui fonctionne sur un serveur et à laquelle on accède à l'aide d'un navigateur web. Cela permet d'accéder à une installation d'Odoo depuis n'importe quel endroit disposant d'une connexion Internet et sur tout appareil prenant en charge un navigateur Web moderne.

L'objectif principal d'Odoo est d'être un système de planification des ressources de l'entreprise (ERP). Les systèmes ERP sont destinés à gérer les processus commerciaux et à suivre les informations les concernant. Ils comprennent généralement des processus tels que les ventes, les achats, le stock, la comptabilité et la gestion de projets. Un problème commun aux entreprises, qu'il s'agisse de start-ups ou de grandes entreprises, est la gestion et le suivi des données liées à leurs processus commerciaux. L'objectif des systèmes ERP est de centraliser ces informations dans une base de données ou un emplacement unique pour en faciliter l'accès [8].

Odoo peut également fournir une gestion de la relation client (CRM), c'est-à-dire l'administration des interactions entre une entreprise ou une organisation et ses clients. Ces données sont enregistrées pour analyser et améliorer les relations avec les clients, ce qui conduit à une croissance des ventes et à la fidélisation des clients. [9]

En plus de fournir des systèmes ERP et CRM, Odoo dispose également d'un système de gestion de contenu ou CMS (Figure 2).

Le CMS est un outil qui permet à l'utilisateur de créer, modifier et publier du contenu sur le Web. Habituellement, les logiciels CMS essaient d'avoir une interface utilisateur intuitive pour permettre la modification et la construction de pages Web. WordPress est l'un des logiciels CMS les plus populaires, avec plus de 40% de tous les sites Web créés à l'aide de ce logiciel [10].

Bien qu'Odoo ne puisse pas directement concurrencer WordPress en termes de parts de marché des CMS, il permet aux entreprises et aux organisations d'avoir des sites web présentables tout en restant dans le même système que leurs autres processus commerciaux.[1]

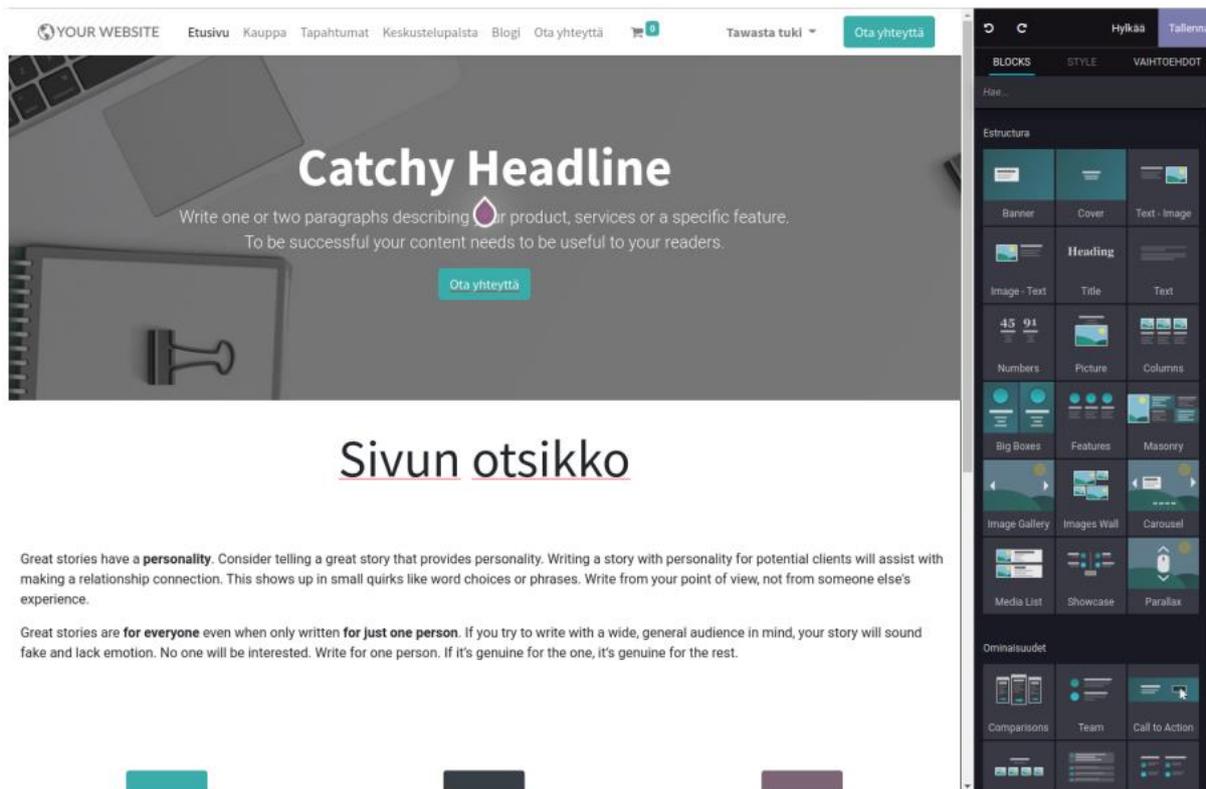


Figure I 2 : Vue du CMS Odoo.

#### 4. Modularité et personnalisation :

À la base, Odoo fournit un serveur d'applications et un cadre pour construire des applications d'entreprise. Chaque application est un module qui implémente ou étend une fonctionnalité spécifique. L'installation de base d'Odoo comprend 30 modules officiels créés et maintenus par Odoo SA et la communauté open-source. Il existe également d'innombrables modules supplémentaires construits par la communauté et d'autres tiers. Certains sont gratuits et d'autres sont payants.

Grâce à ces modules, chaque système Odoo peut être adapté à un besoin commercial spécifique tout en laissant la possibilité d'ajouter des fonctionnalités supplémentaires en installant d'autres modules. Peut-être qu'au début, une startup veut un simple site web utilisant le CMS Odoo, mais plus tard le système peut être étendu pour gérer le commerce électronique et les inscriptions aux événements.

Les modules Odoo sont également divisés en deux catégories : les applications et les composants. Les applications sont des modules qui peuvent inclure plusieurs composants plus petits et fournir une application autonome avec de nombreuses fonctionnalités. Le module Événement est un exemple d'application car il fournit des fonctionnalités pour créer des événements, gérer les inscriptions, envoyer des e-mails aux inscrits, etc. Les modules de composants étendent une application existante en y ajoutant certaines fonctionnalités. Cela peut être aussi simple qu'un module qui ajoute un bouton de raccourci.

Souvent, les applications de base d'Odoo ne correspondent pas entièrement au flux de travail d'une entreprise et les applications et les fonctionnalités doivent être personnalisées pour le client. Cette personnalisation d'Odoo est effectuée par les modules de composants susmentionnés pour ajouter des fonctionnalités, des raccourcis ou automatiser des processus. La personnalisation d'Odoo pour répondre aux besoins exacts de l'entreprise réduira le temps et les efforts consacrés à l'utilisation du système. Cela permettra de soutenir et de rationaliser les processus, au lieu de ralentir le flux de travail en devant lutter contre un système incomplet.

Certaines entreprises peuvent déjà utiliser d'autres systèmes qu'elles préfèrent conserver, mais elles souhaitent également pouvoir transférer des données et des processus entre les anciens systèmes et Odoo. Ce type d'intégration de plusieurs systèmes est possible à personnaliser pour Odoo avec une API (interface de programmation d'application) XML-RPC intégrée ou une API REST personnalisée.[1]

### **5. Caractéristiques techniques et guide de développement :**

Le backend du serveur Odoo et les modules sont principalement écrits en Python avec JavaScript et XML pour rendre le frontend.[1]

#### **a. Architecture :**

Odoo utilise une architecture à trois niveaux (figure 3), qui est couramment utilisée par les applications Web. Dans une architecture à trois niveaux, la présentation visuelle, la logique métier et le stockage des données sont séparés. Le niveau de présentation est la partie de l'application qui est visible pour l'utilisateur final. Son but est de montrer les résultats et les tâches de manière à ce que l'utilisateur puisse les comprendre. Le niveau de présentation de Odoo est écrit en XML, JavaScript et Sass/CSS. [3]

La couche suivante de l'application est le niveau logique de l'entreprise. C'est dans cette couche que sont traitées toutes les décisions logiques, les calculs, les évaluations et les commandes. Elle est également responsable du transfert des données entre les couches de présentation et de données. La couche logique de Odoo est écrite en Python. [2]

La couche inférieure de cette architecture à trois niveaux est le niveau des données, où toutes les informations sont stockées et récupérées. La couche de données peut être une base de données ou un système de fichiers et dans le cas d'Odoo, cette couche utilise une base de données PostgreSQL. [2]

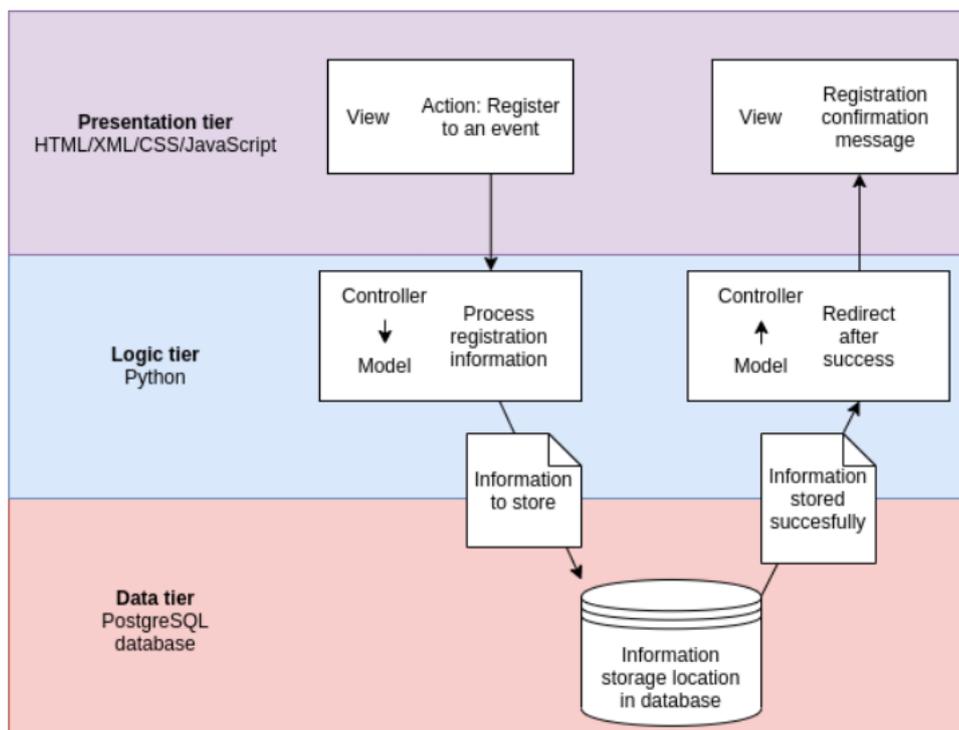


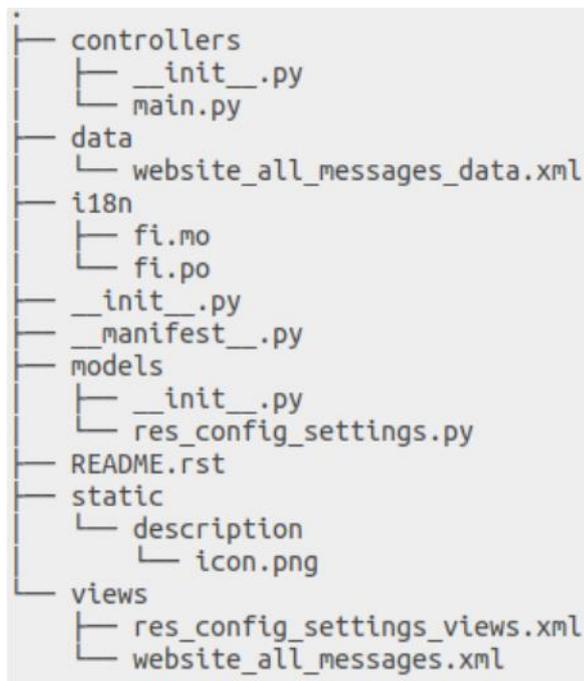
Figure I 3 : L'architecture à trois niveaux d'Odoo.

## b. Créer un module Odoo :

Les modules Odoo contiennent les éléments suivants :

- Les objets métiers ou modèles, sont des classes Python qui contiennent la logique du programme et les champs qui mappent à la base de données en utilisant une couche ORM.
- Les vues d'objets, qui créent l'interface utilisateur. Il s'agit de fichiers XML qui sont convertis en pages HTML à l'aide du moteur de création de modèles QWeb.
- Les fichiers de données, qui déclarent les données du modèle. Ces fichiers de données comprennent des fichiers XML et CSV qui créent l'interface utilisateur du backend, ajoutent des groupes et des règles de sécurité, définissent les paramètres du système, ajoutent des données de démonstration au système et ajoutent la localisation de différentes langues.
- Les contrôleurs Web, qui gèrent les demandes et les redirections du navigateur.
- Les données web statiques, telles que les images, les fichiers CSS, Sass ou JavaScript.

Un même module ne doit pas nécessairement utiliser tous ces éléments et aucun d'entre eux n'est obligatoire en soi. Pour créer un module, il faut créer un nouveau répertoire sous un répertoire de module (Figure 4). Ce nom de répertoire sera le nom technique du module.[2]



**Figure I 4.** Structure du répertoire des modules Odoo.

Le fichier le plus important d'un module est le fichier `__manifest__.py`, qui déclare le module et définit le nom du module, les dépendances, les fichiers inclus et d'autres informations (Figure 5). Outre le fichier manifeste, Odoo a également besoin d'un fichier `__init__.py` pour indiquer à Python les fichiers à charger. Si un module n'a pas de fichiers Python, ce fichier init peut être vide, mais il doit quand même être inclus.[1]

```

-- 23 |
22 |     "name": "Website Event Waiting List",
21 |     "summary": "Adds a waiting list functionality to Events.",
20 |     "version": "14.0.1.2.1",
19 |     "category": "Events",
18 |     "website": "https://gitlab.com/tawasta/odoo/event",
17 |     "author": "Tawasta",
16 |     "license": "AGPL-3",
15 |     "application": False,
14 |     "installable": True,
13 |     "depends": ["website_event_cancellation"],
12 |     "data": [
11 |         "wizard/waiting_mail_list_wizard.xml",
10 |         "security/ir.model.access.csv",
9 |         "data/email_template_views.xml",
8 |         "data/email_template_data.xml",
7 |         "views/assets.xml",
6 |         "views/event_views.xml",
5 |         "views/event_ticket_views.xml",
4 |         "views/event_templates_page_registration.xml",
3 |         "views/event_templates_page_waiting_list.xml",
2 |         "wizard/waiting_mail_list_message.xml",
1 |     ],
44 |

```

**Figure I 5.** Exemple d'un fichier `__manifest__.py`.

### c. Modèles :

L'un des composants clés de Odoo est l'API ORM (Object Relational Mapping), qui relie le niveau logique au niveau des données. Elle présente une structure hiérarchique, une cohérence et une validation des contraintes, un traitement optimisé par requête, une optimisation des permissions, des objets persistants avec la base de données PostgreSQL, un système de cache à plusieurs niveaux, deux mécanismes d'héritage différents et une variété de types de champs tels que les types classiques entier, char, booléen et relationnel ainsi que des types de champs fonctionnels.[2]

Cette API ORM évite au développeur d'avoir à écrire lui-même la plupart des requêtes SQL, ce qui apporte extensibilité et sécurité. Les modèles sont créés en tant que classes Python qui étendent le modèle Odoo (Figure 6). Ils sont ainsi intégrés dans la couche ORM. [2]

Les champs du modèle définissent ce que le modèle peut stocker et où, et ces champs sont définis comme des attributs dans la classe du modèle. Les champs peuvent être divisés en deux grandes catégories : les champs simples et les champs relationnels. Les champs simples sont stockés directement dans la table du modèle et les champs relationnels relient les enregistrements. Les champs simples sont par exemple des champs de type date, texte ou booléen. Les champs relationnels peuvent être des champs Many2one qui relient un seul ou aucun enregistrement à un autre champ, des champs One2many qui relient tous les enregistrements d'un seul champ ou des champs Many2many qui peuvent relier plusieurs enregistrements à un autre champ. [2]

```

13 # 1. Standard library imports:
12 # 2. Known third party imports:
11 # 3. Odoo imports (openerp):
10 from odoo import fields
9  from odoo import models
8
7 # 4. Imports from Odoo modules:
6
5 # 5. Local imports in the relative form:
4
3 # 6. Unknown third party imports:
2
1
33 class Survey(models.Model):
1  # 1. Private attributes
2  _inherit = "survey.survey"
3
4  # 2. Fields declaration
5  is_event_survey = fields.Boolean(
6  "Is an Event Survey",
7  help="If checked, this survey can be used as an Event Survey.",
8  )
9
10 # 3. Default methods
11
12 # 4. Compute and search fields, in the same order that fields declaration
13
14 # 5. Constraints and onchange
15
16 # 6. CRUD methods
17
18 # 7. Action methods
19
20 # 8. Business methods

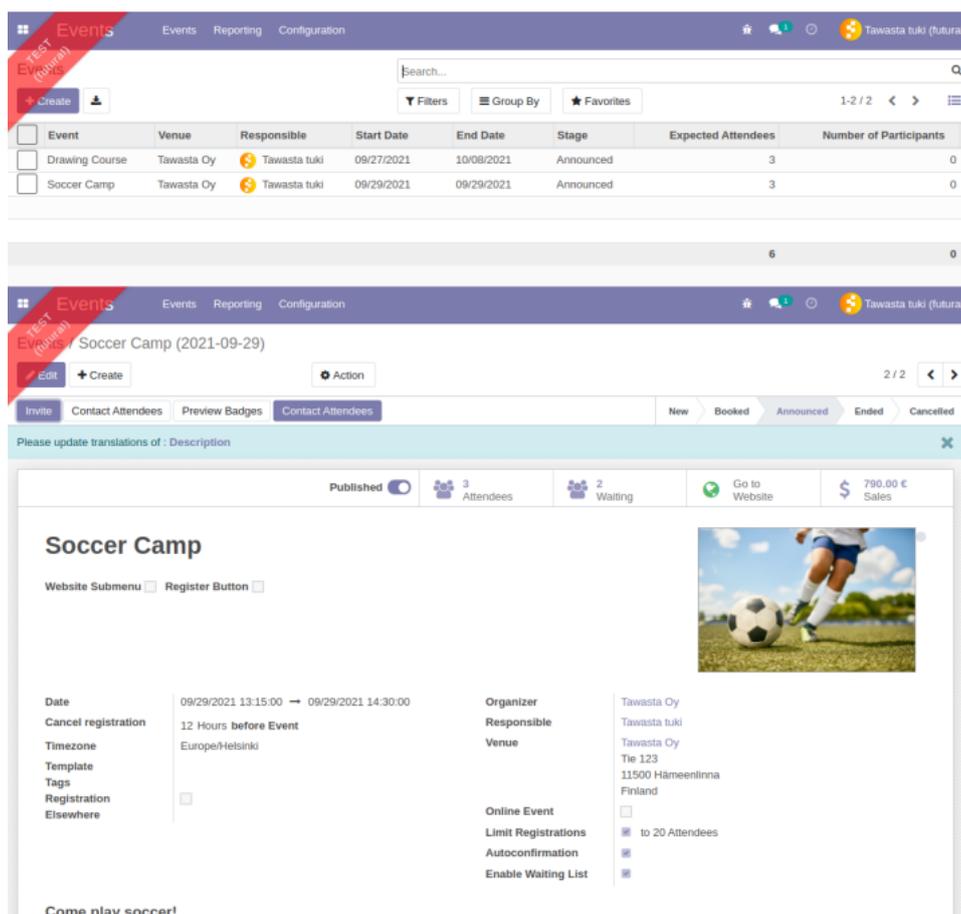
```

Figure I 6. Exemple d'un modèle d'héritage

#### d. Vues et fichiers de données :

Les vues définissent la manière dont les enregistrements sont affichés dans le système back-end Odoo et font partie du niveau de présentation. Ces vues sont écrites en XML et sont hautement personnalisables. Il existe plusieurs types de vues, comme les formulaires, les listes et les kanbans. [2]

Les vues de type liste et kanban affichent généralement une vue d'ensemble de plusieurs enregistrements et une vue de type formulaire affiche les détails d'un seul enregistrement (Figure 7).



**Figure I 7.** Exemple d'une vue en liste (en haut) et d'une vue en formulaire (en bas).

Parfois appelés à tort "vues", les modèles QWeb gèrent l'affichage des pages Web en rendant le XML en pages HTML. Le moteur de création de modèles QWeb peut inclure des instructions conditionnelles, des boucles et des attributs dans les pages Web. Il peut également produire des données d'enregistrement Odoo dans différents formats. Les modèles QWeb sont également dotés d'une fonction d'héritage, ce qui rend triviale l'extension et la réutilisation de snippets précédents (Figure 8). [2]

```

95  <!-- Confirmation page add waiting list -->
1   <template id="registration_complete" inherit_id="website_event.registration_complete">
2   <xpath expr="//div[@class='col-12']" position="replace">
3   <div class="col-12">
4     <t t-if="attendees[0].state == 'wait'">
5       <h3>Joined waiting list!</h3>
6     </t>
7     <t t-else="">
8       <h3>Registration confirmed!</h3>
9     </t>
10    <span class="h4 text-muted" t-esc="event.name"/>
11    <t t-if="attendees[0].state == 'wait'">
12      <p class="py-2">You will be contacted if the event has more seats available.</p>
13    </t>
14  </div>
15 </xpath>
16 </template>

```

**Figure I 8.** Exemple de code QWeb Template avec héritage

La principale façon dont Odoo gère la mise en page et le style des pages Web est avec l'aide de Bootstrap. Bootstrap est un cadre CSS gratuit et open-source qui permet le développement de sites web réactifs. Ces classes Bootstrap sont définies dans le code XML de QWeb. Pour personnaliser davantage la mise en page des pages Web, Odoo prend également en charge les fichiers CSS simples ou les fichiers Sass. [1]

Les fichiers de données dans Odoo jouent un rôle important dans la valeur d'un module. Ils sont chargés lors de l'installation d'un module et définissent des éléments tels que les droits d'accès, les rapports et les modèles d'e-mail. Ils peuvent également ajouter des données simples pour remplir un module avec des données de démonstration ou des valeurs par défaut raisonnables. Les fichiers de données sont principalement définis par des fichiers de données XML, mais les droits d'accès sont définis dans un fichier CSV. [2]

### e. Contrôleurs de Web :

Les contrôleurs Web de Odoo acheminent et dirigent les utilisateurs vers les URL d'un site Web et gèrent les demandes et les réponses Web. Les contrôleurs font partie du niveau logique de Odoo et ils relient les actions effectuées par l'utilisateur final dans le frontend à une action ou un calcul dans le niveau logique. Les contrôleurs web de Odoo sont créés en tant que classes Python qui héritent du contrôleur principal. Ces contrôleurs peuvent définir quels modèles QWeb doivent être rendus et fournir le contexte sur les pages.

Les contrôleurs peuvent appeler des fonctions de modèle sur les enregistrements sélectionnés et lancer un processus tel que l'inscription à un événement après qu'un utilisateur ait rempli un formulaire d'inscription sur le site Web.[2]

## **f. Javascript :**

Odoo dispose également d'un petit Framework Javascript lui permettant d'interagir avec les modèles et les enregistrements de la base de données. Les trois principales utilisations du Framework Javascript sont dans le client web backend, dans le site web et dans le point de vente. Dans Odoo, JavaScript est utilisé dans le niveau de présentation, mais il peut également appeler des actions à partir de modèles ou de contrôleurs Python.

Les fichiers Javascript sont chargés à partir de fichiers de données XML où ils sont définis. Le Framework Javascript de Odoo est utilisé pour charger les fichiers Javascript dans le bon ordre. Dans certains cas, un traitement asynchrone est nécessaire, ce que Odoo gère en renvoyant des promesses.

Le cadre Javascript de Odoo fournit également des widgets utiles qui aident au rendu des données et fournissent des blocs interactifs utiles tels qu'un sélecteur de date et d'heure. [2]

## **III. Conclusion :**

Dans ce chapitre nous avons défini Odoo comme un progiciel adapté à toutes les entreprises. Très modulable et sans coût de licence, il répond aux besoins d'un déploiement rapide à un coût maîtrisé. Nous avons vu aussi les Caractéristiques, l'architecture, le domaine d'application. Les composantes d'Odoo.

*Chapitre I.I: Dossier  
Médical De Patient.*

## I. Introduction :

Les systèmes d'information coordonnent les activités d'une organisation par le biais de l'information afin qu'elle puisse atteindre ses objectifs. C'est l'outil de communication de l'organisation. Il représente toutes les ressources (personnes, matériels et logiciels) au sein d'une organisation qui collectent, stockent, traitent et communiquent des informations.

Un hôpital est une organisation relativement complexe qui associe une variété d'unités avec des fonctions différentes et une certaine autonomie (unités de soins infirmiers, unités de technologie médicale, ... unités afin de traiter au mieux les patients. Elle est assurée par le système d'information hospitalier. Tâches de coordination entre les différentes unités d'un hôpital.

## II. Hospital :

### 1. Définition d'un hôpital :

Un hôpital est un établissement de santé dont la tâche essentielle est de soigner et, si possible, de guérir les malades. Pour accomplir sa mission, l'hôpital dispose d'un certain nombre de ressources humaines, matérielles et logicielles. Un hôpital est en effet une union de sous-systèmes fonctionnellement distincts mais non disjoints. Ces sous-systèmes pourraient ressembler à ceci :

- ✚ Sous-système médical : Concentrez-vous sur les activités mises en place par l'équipe médicale pour répondre aux besoins du patient.
- ✚ Sous-système logistique : comprenant tous les processus générés par les activités médicales : prescription, résultats, transfert, classement...
- ✚ Recherche et sous-système de recherche : documents de groupe pour évaluer et améliorer la qualité des soins fournis. En outre, il fournit des informations pour les sous-systèmes de planification et de gestion.
- ✚ Le sous-système administration : C'est le sous-système de l'hôpital qui s'intéresse à la facturation, à la gestion du personnel, à la gestion des stocks et à la comptabilité d'une manière générale.
- ✚ Le sous-système planification : Il s'appuie sur les activités ou les études de morbidité hospitalière pour engager des décisions d'investissements structurels, matériels et humains. C'est le sous-système qui est en interaction avec l'extérieur (autorités et autres établissements de santé).[11]

## 2. Définition d'un système d'information hospitalier :

Un Système d'Information Hospitalier (SIH) est un ensemble d'éléments en interaction dont le but est de générer, traiter et fournir les informations nécessaires aux activités hospitalières. Il s'agit d'un système d'information pour les professions de santé, en particulier les établissements de santé. Il contient l'ensemble des informations et traitements nécessaires à l'accomplissement de la mission de l'agence. Le SIH est l'une des composantes du Système d'Information Sanitaire (SIS), et les principaux établissements de santé pouvant mettre en place le SIS sont :[12]

- + Les hôpitaux.
- + Les cliniques.
- + Les centres d'analyses.
- + Les centres de soins.
- + Les cabinets médicaux.

## 3. Objectifs du système d'information hospitalier :

- Les deux objectifs principaux d'un SIH sont :[13]
  - + L'amélioration de la qualité des soins : Amélioration de la communication, réduction des délais, aide à la prise de décision.
  - + La maîtrise des coûts : Réduction des durées de séjours, réduction des tâches administratives, diminution du personnel.
- La réussite du SIH dépend essentiellement de trois facteurs importants :
  - + Une analyse approfondie du système d'information de l'hôpital.
  - + Une stratégie logicielle et matérielle adaptée.
  - + Estimation juste des ressources nécessaires.

## 4. Les composantes d'un SIH :

Le système d'information hospitalier est divisé en 3 sous-systèmes :

- o Le sous-système médico-administratif : ce sous-système remplit trois fonctions principales, à savoir :
  - Gestion des patients : admission des patients, gestion des activités hospitalières (lits, transfert interservices), sortie administrative des patients, et frais d'hébergement.
  - Gestion comptable financière : comptabilité fournisseurs, comptabilité clients (tels que frais d'hospitalisation, d'hébergement), gestion des immobilisations.
  - Gestion des actifs : gestion des stocks de matériel, gestion des stocks de médicaments, gestion des stocks de matériel médical, gestion de la ligne de ligature.

- Sous-système médical : Le sous-système médical est considéré comme le cœur du système d'information, autour duquel s'organisent les deux autres sous-systèmes. Il regroupe l'ensemble des unités de soins de l'hôpital où s'exercent la plupart des activités médicales (prescription, signalement, soins infirmiers, visites, etc.).
- Sous-système de technologie médicale : Le sous-système de technologie médicale comprend approximativement toutes les plateformes d'inspection (laboratoire d'analyses, centre d'imagerie médicale, centre d'exploration de fonctions, etc.).[13]

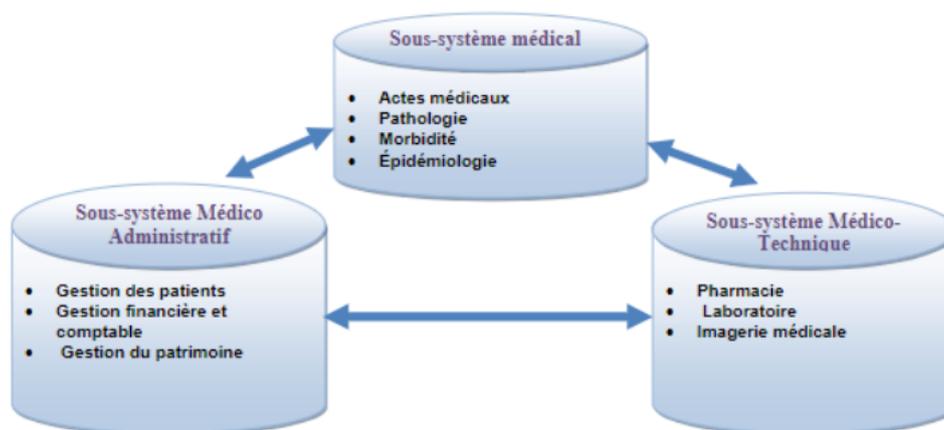


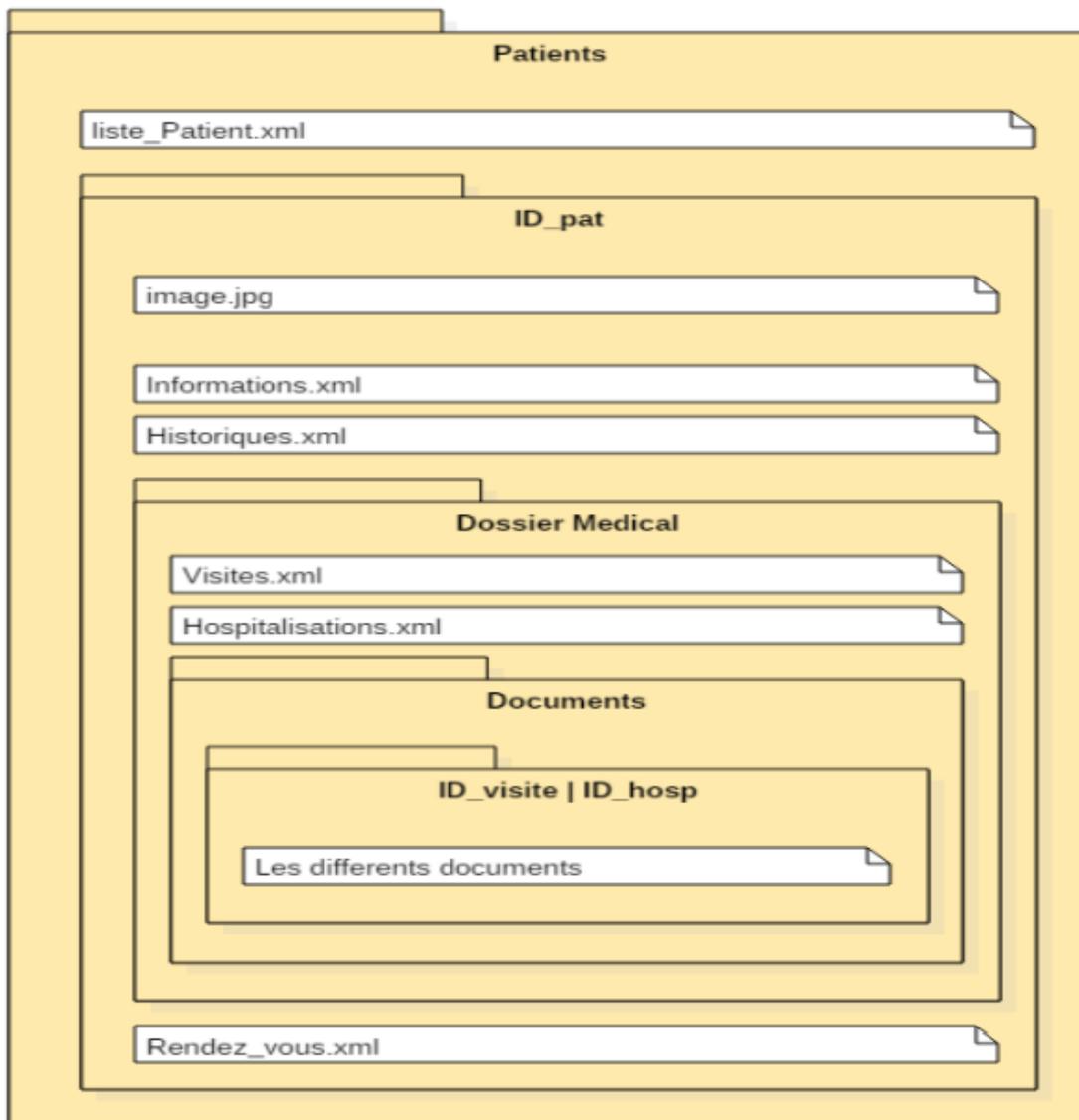
Figure II 9 : Découpage du Système d'Information hospitalier (SIH).

### III. Dossier Médicale De Patient (DMP) :

#### 1. Définition du DMP :

Le **dossier médical** d'une personne est un ensemble de documents qui retrace l'histoire d'une maladie ou de l'ensemble des épisodes ayant affecté la santé de cette personne. Ces documents (lettre, compte-rendu, résultats de laboratoire, film radiologique, ...) sont regroupés dans un dossier, une chemise, un classeur détenu par le patient et/ou le médecin et/ou le service hospitalier ou la clinique.[14]

## 2. Vue générale du DMP :



**Figure II 10 :** Vue générale de DMP

Le dossier 'Patients' contient la liste des patients, lorsqu'un nouveau patient est enregistré, il sera ajouté à la liste des patients (List Patients). Ce dossier contient

- Image du patient (image.jpg),
- Ses informations (information.xml),
- Son historique (history.xml)
- Et son dossier médical, qui contient
  - Liste des visites (visites.xml)
  - Liste d'hospitalisation (hospitalizations.xml)
  - Dossier Documents, qui contient différents documents regroupés par identifiant pour chaque visite ou hospitalisation,
- Enfin il contient également le rendez-vous du patient (rendez\_vous.xml). [15]

### **3. Historique du dossier médical :**

Dans le passé, les observations des soignants consistaient simplement à prendre des notes et à observer. Au 14<sup>ème</sup> siècle, le concept de "dossiers des patients" a émergé. En tant que support écrit, il est utilisé pour les réunions et la conservation (archivage) des notes des médecins. En d'autres termes, cela évite au médecin d'oublier les antécédents médicaux du patient. De plus, ces informations peuvent être partagées avec d'autres médecins, équipes médicales et/ou membres de la famille. A la fin du 17<sup>ème</sup> siècle, le "dossier médical personnel" de chaque patient apparaît. Il sert alors de registre à l'Hôtel-Dieu de Paris. Mais le contenu est très concis. Complexité du traitement, optimisation En matière de droits des patients hospitalisés, la qualité des soins et l'évolution sociale contribuent à renforcer le dossier médical.

Les dossiers infirmiers font partie des dossiers médicaux depuis 1970. L'ensemble est devenu un outil d'échange et de transfert de données entre professionnels de santé, quel que soit leur sport (hôpitaux et secteurs libres).

En résumé, les dossiers médicaux sont constitués de notes prises par les médecins lors des consultations et sont aujourd'hui des documents médico-légaux et administratifs. [16]

### **4. Bénéfice de dossier médical :**

La traçabilité du suivi permet une analyse de la qualité des prises en charge. Il est alors un moyen d'améliorer la qualité des soins, ceci dans le but d'une meilleure gestion de la santé de population. De plus, l'accès du patient à son dossier médical permet de lui fournir une information éclairée de son état et de sa prise en charge. Par conséquent l'intérêt est double, pour le professionnel et le patient.[17]

### **1) L'intérêt pour les professionnels de santé :**

C'est un moyen permettant le suivi au long cours, une approche globale et la synthèse de toutes les pathologies retrouvées chez le patient et enfin, la transmission des données aux divers intervenants dans le cadre de la continuité des soins. Également, la traçabilité exigée dans le dossier favorise la mise en place des actions préventives et éducatives concernant le patient et elle simplifie la gestion du patient dans le système de santé. Par ailleurs, la bonne tenue du dossier médical oblige les personnels médicaux à adapter leurs pratiques aux contraintes médico-légales.[17]

### **2) L'intérêt pour le patient :**

La loi de 4 mars 2002, relative aux droits des malades et la qualité de système de santé et les recommandations de la Haute Autorité de Santé précisent que toute personne a désormais accès à son dossier médical. « Toute personne a accès à l'ensemble des informations concernant sa santé détenues par des professionnels et établissements de santé, qui sont formalisées et ont contribué à l'élaboration et au suivi du diagnostic et du traitement ou d'une action de prévention, ou ont fait l'objet d'échanges écrits entre professionnels de santé, notamment des résultats d'examen, comptes rendus de consultation, d'intervention, d'exploration ou d'hospitalisation, des protocoles et prescriptions thérapeutiques mis en œuvre, feuilles de surveillance, correspondances entre professionnels de santé, à l'exception des informations mentionnant qu'elles ont été recueillies auprès de tiers n'intervenant pas dans la prise en charge thérapeutique ou concernant un tel tiers ». L'accès peut être autorisé directement ou par l'intermédiaire d'un médecin et/ou d'une tierce personne (représentant légal), ceci dans des conditions définies par voie réglementaire.[17]

#### **a. Modalités de la demande :**

Une demande doit être formulée auprès du médecin ou de l'établissement de santé (détenteur des informations). Cette demande peut provenir de la part de la personne concernée, son ayant droit (en cas de décès), la personne ayant l'autorité parentale, le tuteur ou le médecin désigné comme intermédiaire [17]

#### **b. Délai et mode de communication :**

La communication doit s'effectuer dans un délai de 48 heures à huit jours suivant la demande. Ce délai est prolongé de deux mois si la demande concerne un dossier datant de cinq ans et plus. Deux modalités de communication sont accessibles :[17]

- Consultation sur place avec la remise de copies de documents,
- Envoi de copies des documents.

### **c. Des règles particulières :**

#### **- Accès au dossier d'un patient décédé**

La loi précise que :

« Le secret médical ne fait pas obstacle à ce que les informations concernant une personne décédée soient délivrées à ses ayants droit, dans la mesure où elles leur sont nécessaires pour leur permettre de connaître les causes de la mort, de défendre la mémoire du défunt ou de faire valoir leurs droits, sauf volonté contraire exprimée par la personne avant son décès. »

#### **Cas spécifique de mineur**

Elle précise que :

« Le médecin peut se dispenser d'obtenir le consentement du ou des titulaires de l'autorité parentale sur les décisions médicales à prendre lorsque le traitement ou l'intervention s'impose pour sauvegarder la santé d'une personne mineure, dans le cas où cette dernière s'oppose expressément à la consultation du ou des titulaires de l'autorité parentale afin de garder le secret sur son état de santé. Toutefois, le médecin doit dans un premier temps s'efforcer d'obtenir le consentement du mineur à cette consultation. Dans le cas où le mineur maintient son opposition, le médecin peut mettre en œuvre le traitement ou l'intervention. Dans ce cas, le mineur se fait accompagner d'une personne majeure de son choix. » [17]

## **IV. Le dossier médical informatisé :**

La gestion de l'information liée aux soins aux patients a profondément changé depuis le début des années 2000. Auparavant conservées sous forme de dossier médical papier et de dossier individuel de soins, avec la numérisation progressive de nos hôpitaux, les données patients ont été regroupées sous forme de Dossiers Médicaux Électroniques (DSE, Dossier de Santé Électronique). Aujourd'hui, son développement et sa bonne maîtrise constituent un thème important dans le système de santé.[18]

### **1. Définition :**

Un dossier médical informatisé se rattachant à un clinicien, un cabinet ou une organisation. C'est le dossier dans lequel les cliniciens consignent les données sur leurs propres patients telles que les renseignements sociodémographiques, les antécédents médicaux, le profil pharmaceutique et les diagnostics (résultats de laboratoire et d'imagerie diagnostique). Il est souvent intégré à d'autres logiciels servant à gérer d'autres fonctions telles que la facturation et la gestion des rendez-vous. Le dossier médical informatisé est une des composantes d'un système d'information en réseaux.

Il est précisé dans la loi du 4 mars 2002 relative aux droits des malades que le dossier médical concerne l'élaboration des suivis de diagnostic, les traitements, mais aussi plus généralement tous les échanges écrits entre les professionnels de santé. Le dossier médical informatisé est donc constitué d'informations administratives et médicales nominatives qui forment une base de données dans le sens où il s'agit d'un "recueil d'œuvres, de données, ou d'autres éléments indépendants, disposés de manière systématique ou méthodique et individuellement accessibles par des moyens électroniques ou par tout autre moyen" (loi du 1 juillet 1998).

Selon le Conseil Supérieur des Systèmes d'Information de Santé (CSSIS), organisme chargé d'évaluer les perspectives et modalités de mise en œuvre du dossier médical et des échanges d'informations nécessaires à la meilleure prise en charge des personnes, il est préférable d'utiliser le terme de dossier santé qui est moins restrictif que celui de dossier médical dans le sens où il englobe les soins de l'ensemble des professionnels de la santé, du médecin à l'infirmière en passant par le dentiste.[19]

### **2. Utilité d'un dossier patient informatisé :**

Dans son rapport d'activité 1999, le Conseil Supérieur des Systèmes d'Information de Santé indique que le dossier se place au cœur du système de santé dans le sens où il se situe au carrefour de quatre grandes évolutions [19]

- Le développement de la "Médecine assistée par ordinateur"
- La recherche d'une efficacité sanitaire nouvelle
- Le besoin de transparence et d'information
- Le besoin de sécurité sanitaire

Informatiser un dossier de santé permet :

- De faciliter la coordination des soins entre les différents professionnels de santé. Le dossier de santé doit permettre une prise en charge partagée du patient au sein des différentes structures de soins d'un réseau.
- De faciliter l'exercice professionnel quotidien par la fourniture d'outils de classification permettant de retrouver les informations rapidement selon plusieurs critères : par nature des données (cliniques, biologiques, imagerie), par ordre chronologique, par nom, par âge, par lieu de domiciliation, par type d'affection.
- D'apporter une aide à la décision, à l'évaluation et aux études cliniques en permettant l'utilisation de protocoles de prise en charge prédéfinis établis à partir des référentiels de pratiques. Ces protocoles comporteront des formulaires de saisie correspondant aux données structurées nécessaires à l'évaluation de la qualité des soins dispensés dans le réseau, à la recherche clinique coopérative régionale, aux études épidémiologiques et à la traçabilité du parcours du patient dans le système de soins.

- Le dossier de santé par Internet offre en plus de ces services, la possibilité pour le patient d'accéder à son dossier à n'importe quel endroit du monde et en plusieurs langues. De plus, il permet de favoriser la prise de conscience et la prise en charge par le patient lui-même de sa santé grâce par exemple à l'implémentation de messages d'alerte automatique (rappel de vaccinations obligatoires, de consultations annuelles ou d'examens complémentaires à effectuer).[19]

### 3. Les informations à recueillir dans le dossier :

- Identification.
- Nom complet actualisé
- Sexe
- Date de naissance
- Numéro du dossier
- Symbole pour signaler les homonymes
- Informations administratives
- Adresse
- Téléphone
- Profession
- Numéro de sécurité sociale
- Affection de longue durée (ALD)
- Tuteur, curateur, tiers payeurs
- Mutuelle
- Données d'alertes (dont allergies et intolérances médicamenteuses)
- Rencontre
- Nom du médecin
- Date de la rencontre
- Type de contact
- Données significatives de la rencontre
- Conclusion/synthèse de la rencontre
- Décisions
- Histoire médicale actualisée et facteurs de santé
- Antécédents personnels
- Antécédents familiaux
- Facteurs de risques
- Vaccinations et autres actions de prévention et de dépistage
- Evènements biographiques significatifs.[19]

## **V. Conclusion :**

En résumé, le dossier médical du patient est un dossier qui exige des normes et des standards plus élevés. D'une part, il est un outil de soins et de traçabilité, et un outil d'études pathologiques lorsqu'il s'agit d'une maladie rare ; d'autre part, il représente l'image et la qualité des soins dans un hôpital.

*Chapitre III:*  
*Conception Et Réalisation*  
*Du Module Odoo*

### **I. Introduction :**

Pour une meilleure organisation et un meilleur contrôle des projets informatiques, il est nécessaire de suivre une démarche méthodologique rigoureuse.

Pour cela, le choix du formalisme de conception est très important. Pour notre développement ERP, nous avons choisi une approche de conception orientée objet basée sur la modélisation UML. Nous commencerons par des diagrammes de cas d'utilisation et des diagrammes de séquence, qui nous donneront une vision plus claire et plus détaillée des différentes fonctions du système, puis nous modéliserons des diagrammes de classes.

Dans la partie réalisation, nous allons créer et implémenter les différents programmes, interfaces et bases de données utilisés pour construire notre module et ses fonctionnalités. Nous décrivons le système de construction et l'environnement de la base de données, puis montrerons certaines des interfaces résultantes.

### **II. Conception :**

La recherche dans le domaine des soins de santé repose, outre les données d'étude primaires acquises, sur des données provenant de sources secondaires très différentes. Afin de relier, comparer et analyser les sources de données, des modèles et des méthodes uniformes sont nécessaires. Cela pourrait être facilité par une description plus structurée des exigences, des modèles et des méthodes de la recherche en soins de santé que celles utilisées actuellement. Des méthodes de présentation appropriées ont été recherchées dans une approche de cet objectif et le langage de modélisation unifié (UML) a été identifié comme une alternative possible.

#### **1. Définition de UML :**

UML (Unified Modeling Language) est conçu comme un langage de modélisation visuel polyvalent, sémantiquement et syntaxiquement riche. Il est destiné à être utilisé dans l'architecture, la conception et la mise en œuvre de systèmes logiciels complexes à travers sa structure et son comportement. UML a des applications au-delà du développement logiciel, en particulier pour le flux de processus dans l'industrie. Il est similaire aux plans utilisés dans d'autres domaines et se compose de différents types de graphiques. En général, les diagrammes UML décrivent les limites, la structure et le comportement d'un système et des objets qu'il contient.

UML n'est pas un langage de programmation, mais il existe des outils qui peuvent être utilisés pour générer du code dans plusieurs langages à partir de diagrammes UML. UML a une relation directe avec l'analyse et la conception orientées objet. [20]

## 2. Diagramme de cas d'utilisation du module :

Un cas d'utilisation est une liste d'étapes qui définissent les interactions entre un acteur (un humain qui interagit avec le système ou un système externe) et le système lui-même. Les diagrammes de cas d'utilisation décrivent les spécifications d'un cas d'utilisation et modélisent les unités fonctionnelles d'un système. Ces diagrammes aident les équipes de développeurs à comprendre les besoins de leur système, notamment le rôle des interactions humaines et les différences entre plusieurs cas d'utilisation. Un diagramme d'utilisation peut illustrer tous les cas d'utilisation du système ou seulement un groupe de cas d'utilisation ayant des fonctionnalités similaires.[21]

- ✧ Pour commencer un diagramme de cas d'utilisation, ajoutez une forme ovale au centre du dessin.
  - ✧ Saisissez le nom du cas d'utilisation à l'intérieur de la forme ovale.
  - ✧ Représentez les acteurs par des bonshommes disposés près du diagramme, puis utilisez des lignes pour modéliser les relations entre acteurs et cas d'utilisation. [21]
- Relation "Contain" : la relation de confinement du cas d'utilisation A par rapport au cas d'utilisation B signifie qu'une instance de A contient le comportement décrit dans B.
  - Relation "Extend" La relation d'extension du cas d'utilisation B au cas d'utilisation A signifie que les instances de A peuvent être étendues par le comportement décrit dans B.

**a. Les diagrammes des cas d'utilisation de l'application :**

- **Identification des acteurs et des cas d'utilisation :**

Les acteurs sont des entités externes qui interagissent directement avec le système et communiquent avec lui en envoyant et en recevant des messages Le tableau suivant résume les acteurs et les cas d'utilisation :

Cas d'utilisation	Acteur
Gérer les rendez-vous	Médecin
Consulter les statistiques	
Ajouter un diagnostic	
Gérer les dossiers patients	
Ajouter une ordonnance	
Chercher médecin	Secrétaire
Créer un dossier patient	
Prendre un rendez-vous	
Consulter l'état des RDV	
Gérer les comptes des utilisateurs	Administrateur

**Tableau III 1 :** Identification des acteurs et des cas d'utilisations

**b. Diagramme de cas d'utilisation de directeur :**

- Le directeur est le seul qui a le droit d'accepter ou rejeter un médecin.
- Créer un compte utilisateur.
- Supprimer un utilisateur.
- Modifier les droits d'accès d'un utilisateur.

→ Diagramme de cas d'utilisation « gestion des utilisateurs »

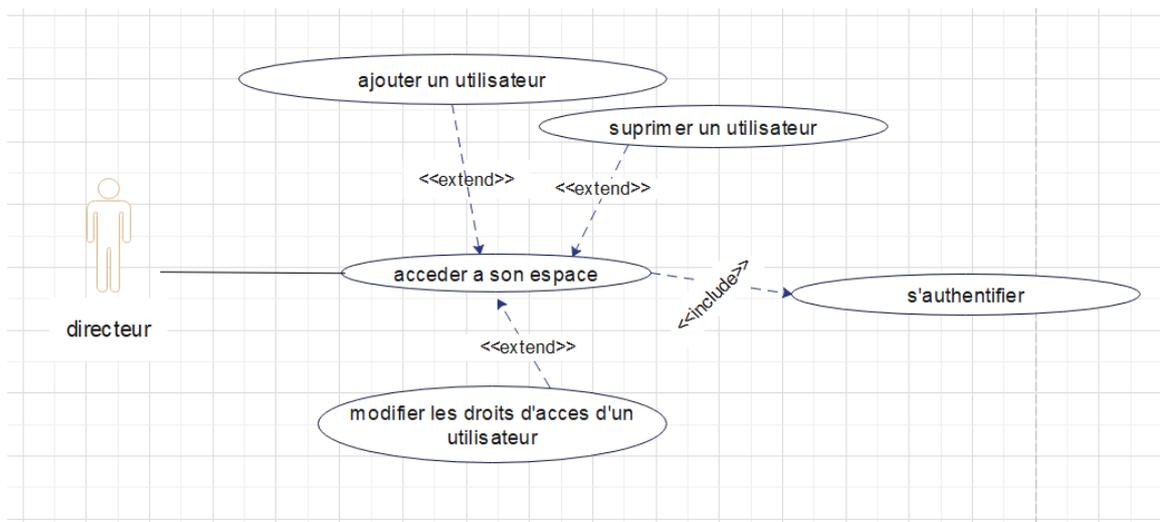


Figure III 11 : Diagramme de cas d'utilisation « gestion des utilisateurs »

✚ Description du cas d'utilisation « Gestion des utilisateurs »

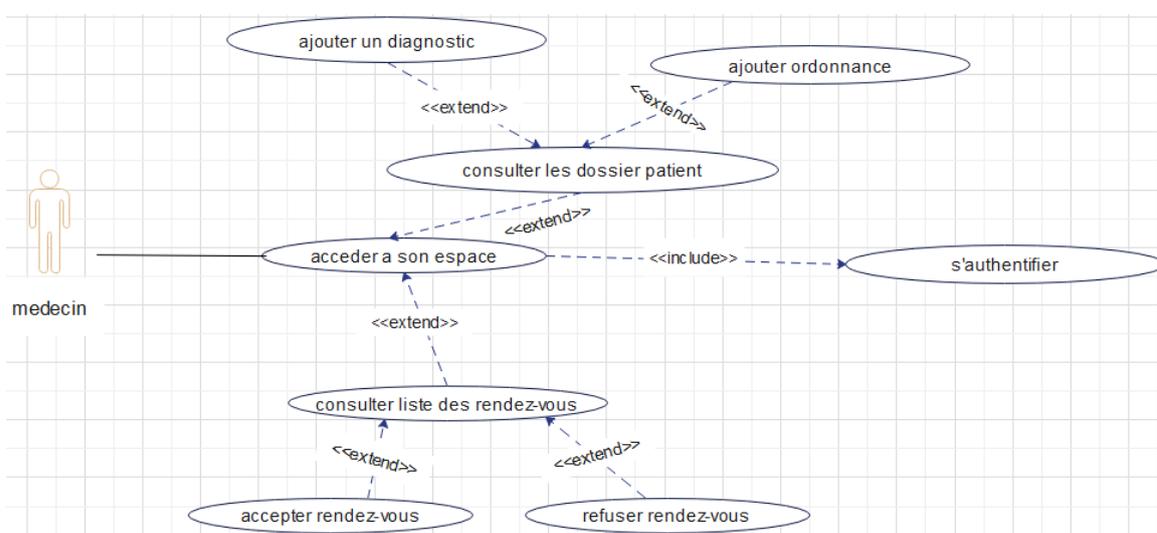
<u>« Gestion des utilisateurs »</u>	
<b>Cas d'utilisation</b>	
<b>Résumé</b>	Ce cas permet au directeur d'ajouter et modifier les comptes des utilisateurs.
<b>Acteurs</b>	Directeur
<b>Scénario nominale</b>	<ul style="list-style-type: none"> <li>• Le directeur rechercher compte médecin</li> <li>• Le système afficher le compte</li> <li>• Le directeur ajouter ou modifier le compte</li> <li>• Le système enregistre les informations dans la base de données</li> <li>• Le directeur rechercher compte secrétaire</li> <li>• Le système afficher le compte</li> <li>• Le directeur ajouter ou modifier le compte</li> <li>• Le système enregistre les informations dans la base de données</li> </ul>
<b>Post condition</b>	Créer un compte utilisateur Modifier un compte utilisateur

Tableau III 2 : description de cas d'utilisation « Gestion des utilisateurs »

**c. Diagramme de cas d'utilisation de médecin :**

- Le médecin peut accepter ou refuser un rendez-vous.
- Le médecin peut ajouter ou modifier un diagnostic.
- Le médecin peut ajouter et modifier ordonnance.
- Si un rendez-vous est accepté, le médecin peut gérer le dossier du patient : il peut ajouter un diagnostic ou/et une ordonnance.
- Le médecin peut aussi consulter les statistiques des rendez-vous.

→ Diagramme de cas d'utilisation « Gestion des diagnostics »



**Figure III 12 :** Diagramme de cas d'utilisation « Gestion des diagnostics »

✚ Description du cas d'utilisation « Gestion des diagnostics »

<u>« Gestion des diagnostics »</u>	
<b>Cas d'utilisation</b>	
<b>Résumé</b>	Ce cas permet au médecin d'ajouter et modifier les diagnostics.
<b>Acteurs</b>	Médecin
<b>Scénario nominale</b>	<ul style="list-style-type: none"> <li>• Le médecin rechercher dossier patient</li> <li>• Le système afficher le dossier</li> <li>• Le médecin ajouter ou modifier le diagnostic</li> <li>• Le système enregistre les informations dans la base de données</li> <li>• Le médecin ajouter une ordonnance</li> <li>• Le système enregistre les informations dans la base de données</li> </ul>
<b>Post condition</b>	<p>Diagnostic crée</p> <p>Ordonnance crée</p>

**Tableau III 3 :** description de cas d'utilisation « Gestion des diagnostics »

**d. Diagramme de cas d'utilisation de secrétaire :**

- Pour pouvoir accéder aux différentes fonctionnalités de l'application, le secrétaire doit se connecter s'il possède déjà un dossier, sinon il doit créer un dossier.
- Le secrétaire peut chercher un médecin par son nom ou par sa spécialité.
- Une fois le médecin a été sélectionné, le secrétaire peut prendre un rendez-vous selon la disponibilité du médecin.
- Après avoir pris un rendez-vous, le secrétaire peut consulter la liste de ses rendez-vous (acceptés, refusés et annuler). Tant que le rendez-vous est encore en attente, le secrétaire a la possibilité de l'annuler.
- Le secrétaire peut aussi changer les informations de dossier de patient.

→ Diagramme de cas d'utilisation « créer un dossier patient »

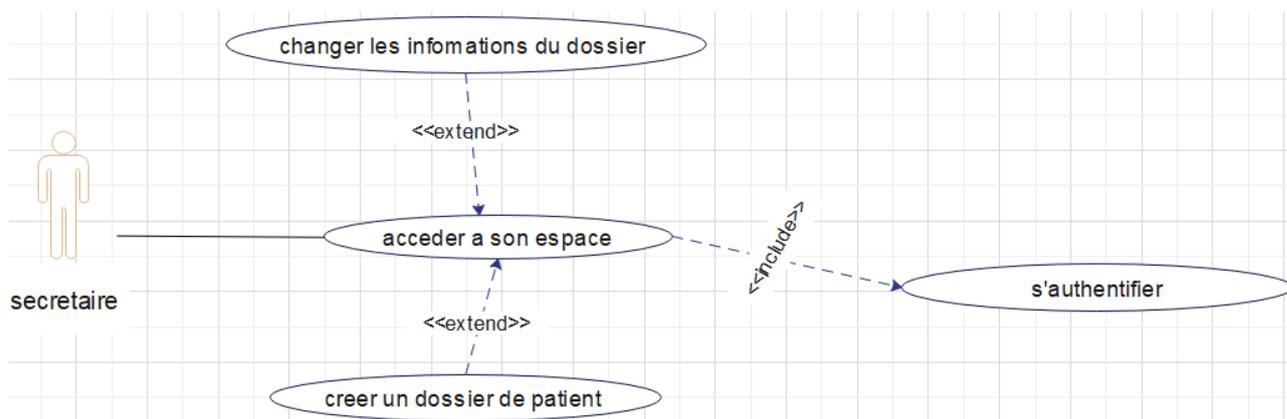


Figure III 13 : Diagramme de cas d'utilisation « créer un dossier patient »

✚ Description du cas d'utilisation de création du dossier patient

<b><u>Créer un dossier patient</u></b>	
<b>Cas d'utilisation</b>	
<b>Résumé</b>	Ce cas permet au secrétaire de créer un dossier
<b>Acteurs</b>	Secrétaire
<b>Scénario nominale</b>	<ul style="list-style-type: none"> <li>• Le Secrétaire saisie les données</li> <li>• Le système reçoit les données saisies</li> <li>• Si les données sont valides un jeton de sécurité sera créé</li> <li>• Le système enregistre les informations dans la base de données</li> <li>• Le système envoie le jeton de sécurité au patient</li> </ul>
<b>Scénario d'erreur</b>	Le Secrétaire saisit des données erronées <ul style="list-style-type: none"> <li>• Le système envoie un message d'erreur au Secrétaire</li> <li>• Le Secrétaire sera redirigé à l'interface de création du dossier</li> </ul>
<b>Post condition</b>	Dossier crée

Tableau III 4 : description de cas d'utilisation de création du dossier patient

→ Diagramme de cas d'utilisation « prendre des RDV »

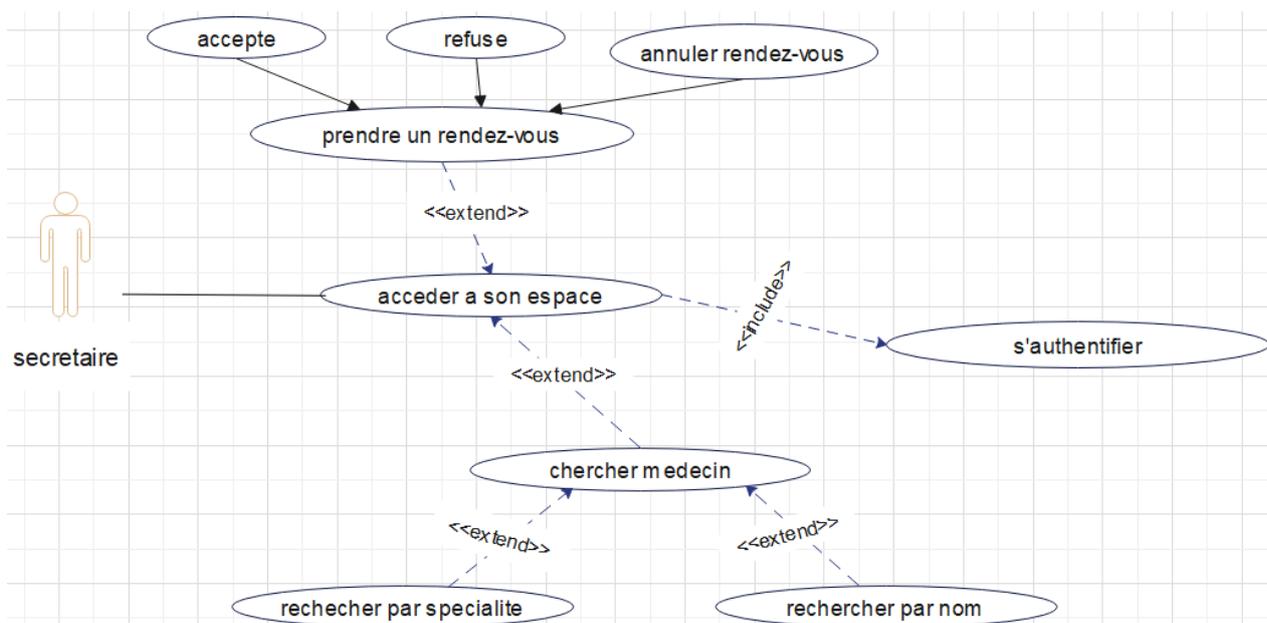


Figure III 14 : Diagramme de cas d'utilisation « prendre des RDV »

🚩 Description du cas d'utilisation de prise de rendez-vous

<u><b>Prendre rendez-vous</b></u>	
<b>Cas d'utilisation</b>	
<b>Résumé</b>	Ce cas permet au secrétaire de prendre un rendez-vous avec un médecin
<b>Acteurs</b>	Secrétaire
<b>Scénario nominale</b>	<ul style="list-style-type: none"> <li>• Le secrétaire sélectionne la date du rendez-vous</li> <li>• Le secrétaire sélectionne l'heure du rendez-vous</li> <li>• Le système vérifie la disponibilité du médecin</li> <li>• Le rendez-vous est enregistré dans la base de données</li> </ul>
<b>Scénario d'erreur</b>	<ul style="list-style-type: none"> <li>• La date ne correspond pas à la disponibilité du médecin</li> </ul>
<b>Précondition</b>	Choisir un médecin Être authentifié
<b>Post condition</b>	Le rendez-vous enregistré

Tableau III 5 : description du cas d'utilisation de prise de rendez-vous

✚ Description du cas d'utilisation de recherche de médecin

<b>Cas d'utilisations</b>	<b><u>Chercher un médecin</u></b>
<b>Résumé</b>	Ce cas permet au secrétaire de chercher un médecin
<b>Acteurs</b>	Secrétaire
<b>Scénario nominale</b>	<ul style="list-style-type: none"> <li>• Le secrétaire choisit un médecin sur la carte</li> <li>• Le système affiche le profil du médecin</li> <li>• Le secrétaire choisit une spécialité</li> <li>• Le système cherche tous les médecins avec la spécialité choisie</li> </ul>
	<ul style="list-style-type: none"> <li>• Le secrétaire choisit un médecin de la liste</li> <li>• Le système affiche le profil du médecin sélectionné</li> <li>• Le secrétaire saisit le nom du médecin</li> <li>• Le système cherche tous les médecins avec ce nom</li> <li>• Le secrétaire choisit un médecin de la liste</li> <li>• Le système affiche le profil du médecin</li> </ul>
<b>Scénario d'erreur</b>	<ul style="list-style-type: none"> <li>• Le secrétaire ne choisit aucune spécialité</li> <li>• Le secrétaire ne choisit aucun médecin</li> <li>• Le secrétaire saisie un nom erroné</li> <li>• Le secrétaire ne saisit aucun nom</li> <li>• Le secrétaire ne choisit aucun médecin</li> </ul>
<b>Précondition</b>	Patient authentifié
<b>Post condition</b>	Profil de médecin affiché

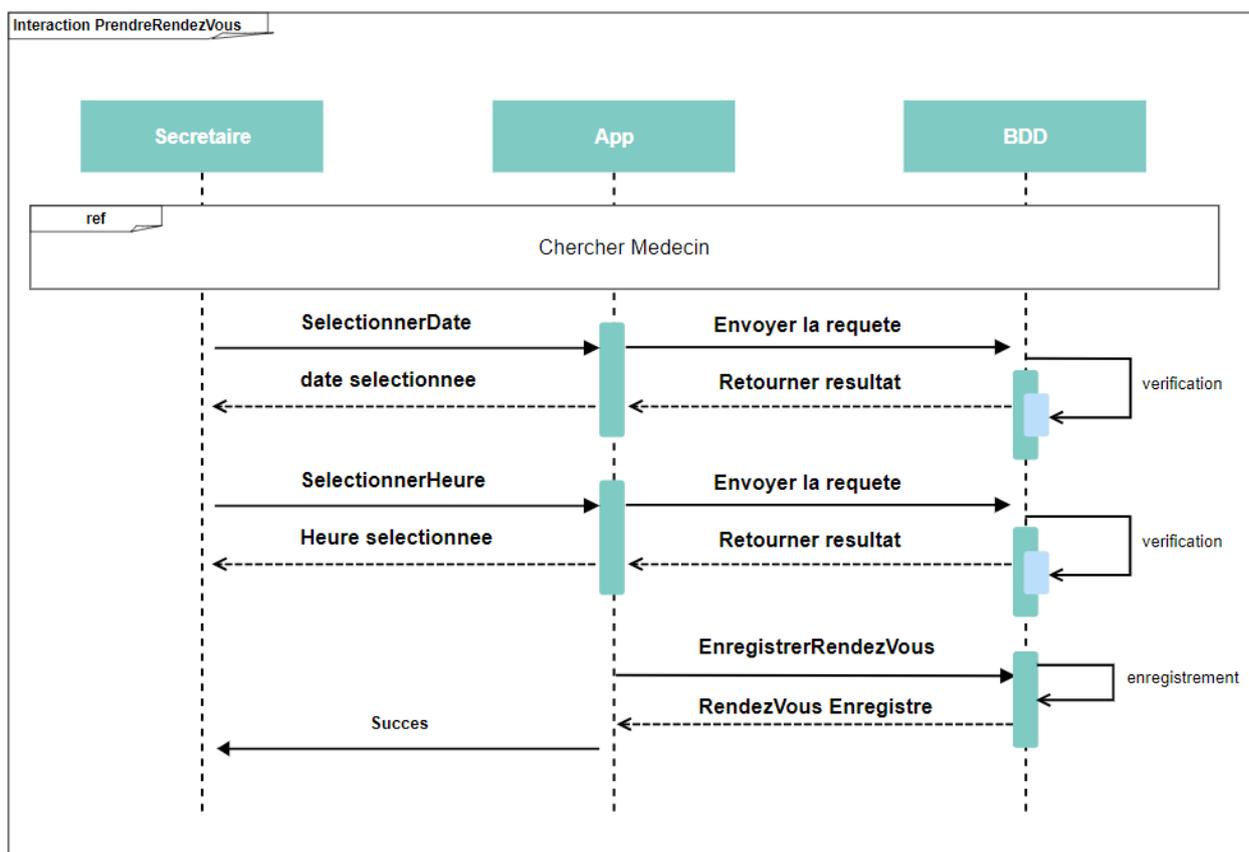
**Tableau III 6** : description du cas d'utilisation de recherche de médecin

**4. Diagramme de séquence :**

Les diagrammes de séquence, également appelés diagrammes d'événements, illustrent la façon dont des processus interagissent entre eux en montrant les appels entre différents objets dans une séquence. Ces diagrammes ont deux dimensions : verticale et horizontale. Les lignes verticales montrent la séquence de messages et d'appels dans l'ordre chronologique, et les éléments horizontaux montrent les instances d'objets où les messages sont relayés.[22]

- ✧ Pour créer un diagramme de séquence, écrivez le nom de l'instance de classe et le nom de la classe dans une case rectangulaire.
- ✧ Dessinez des lignes entre les instances de classes pour représenter l'émetteur et le destinataire des messages.
- ✧ Utilisez des pointes de flèches pleines pour symboliser les messages synchrones, des pointes de flèches ouvertes pour les messages asynchrones et des lignes en pointillés pour les messages de réponse.

### I. Diagramme de séquence de prise de rendez-vous :



**Figure III 15 :** Diagramme de séquence prendre rendez-vous

Pour prendre rendez-vous, le secrétaire sélectionne la date et l'heure du rendez-vous souhaité. Après vérification de la disponibilité du médecin, le rendez-vous est enregistré dans la base de données et un message de confirmation est affiché au secrétaire. Si la date et/ou l'heure sélectionnée ne correspond pas au temps libre du médecin, le système envoie un message d'erreur au secrétaire l'informant que son rendez-vous n'a pas été enregistré et lui demande de sélectionner une autre date.

## II. Diagramme de séquence de recherche du médecin :

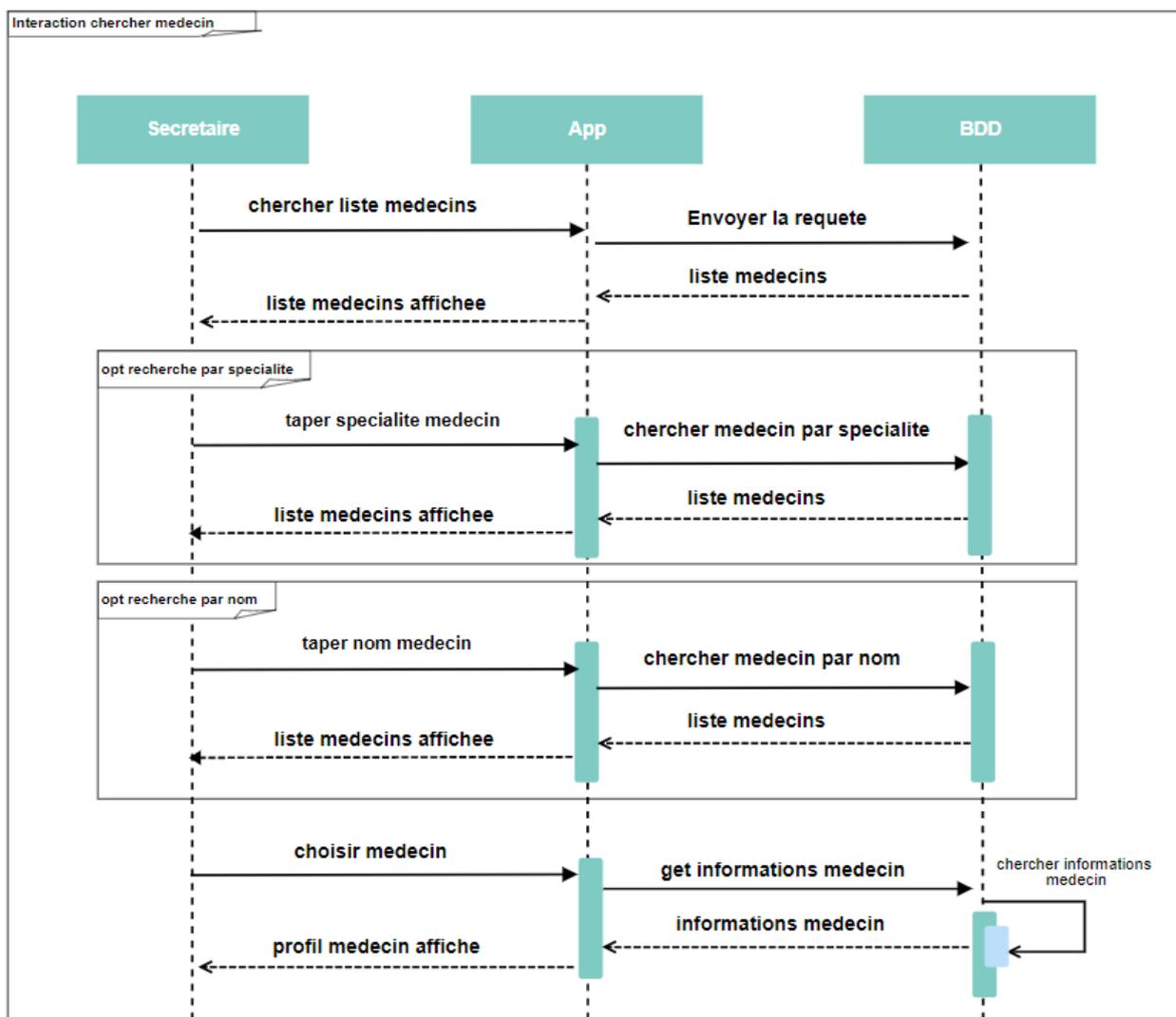


Figure III 16 : Diagramme de séquence de recherche du médecin

- Le secrétaire peut rechercher des médecins par nom et spécialité.
- Pour rechercher des médecins par spécialité, sélectionnez simplement une spécialité existante dans la base de données et le système affichera une liste de médecins dans cette spécialité.
- De plus, le secrétaire peut rechercher des médecins par leur nom. Pour ce faire, entrez simplement un nom et le système affichera une liste de médecins portant ce nom.

### III. Diagramme de séquence « créer un dossier patient » :

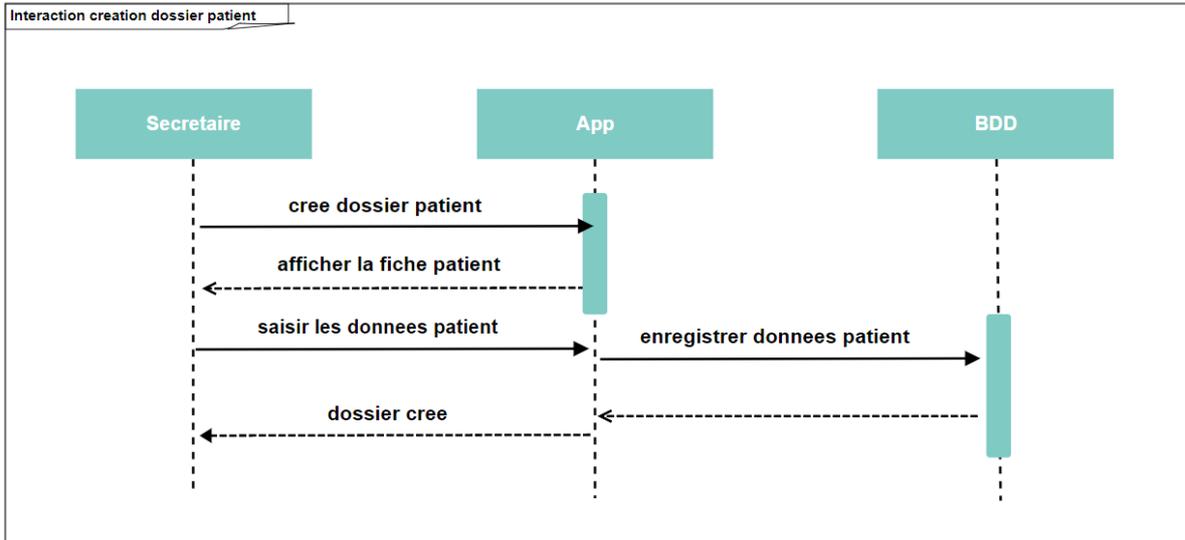


Figure III 17 : Diagramme de séquence « créer un dossier patient »

### IV. Diagramme de séquence « gestion des utilisateurs » :

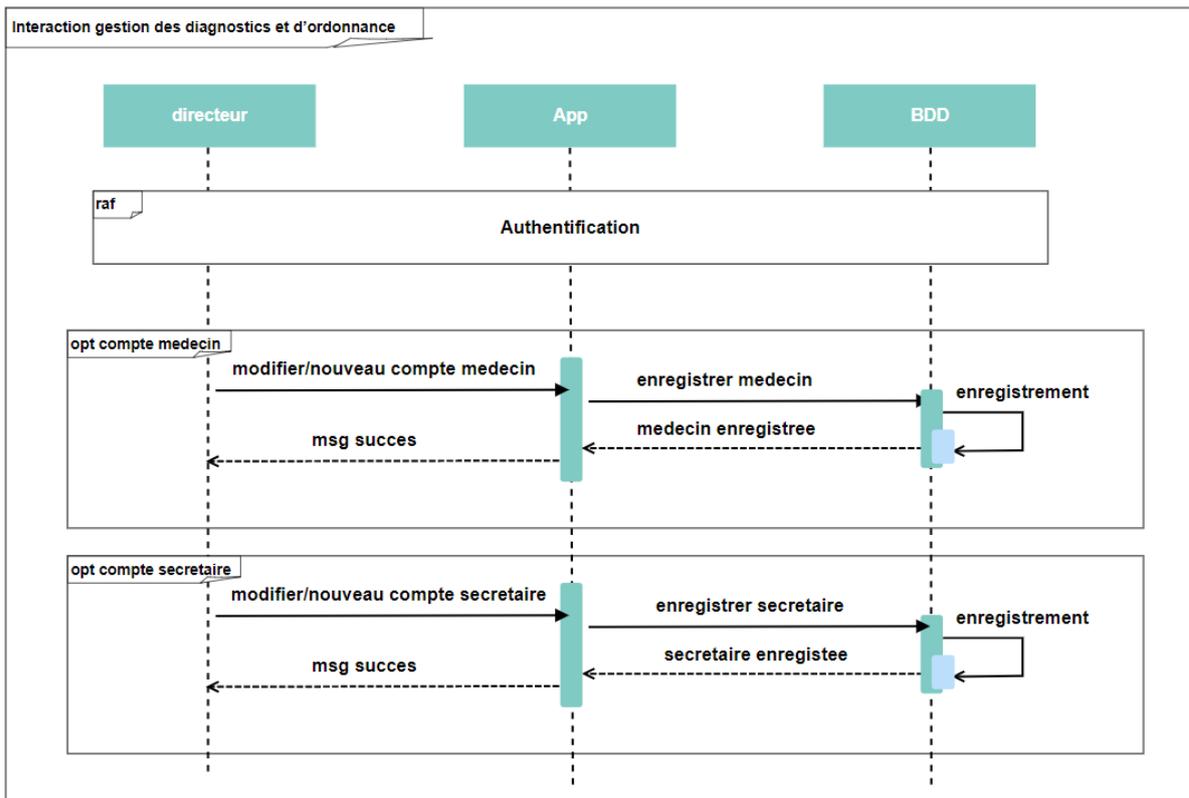


Figure III 18 : Diagramme de séquence « gestion des utilisateurs »

## V. Diagramme de séquence de gestion des diagnostics et d'ordonnance :

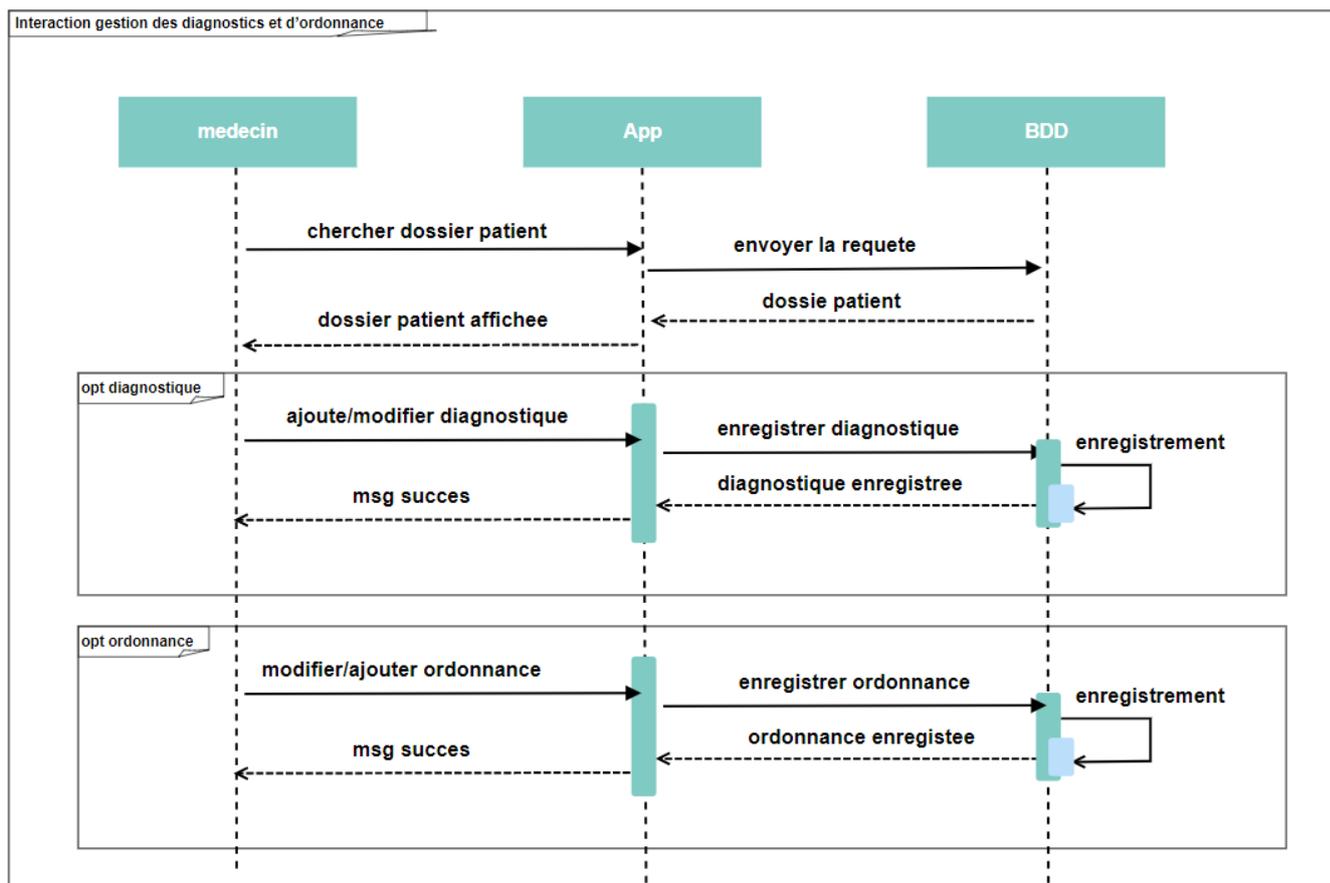


Figure III 19 : Diagramme de séquence de gestion des diagnostics et d'ordonnance

### 5. Diagramme de classe :

Les diagrammes de classes représentent les structures statiques d'un système, y compris ses classes, attributs, opérations et objets. Un diagramme de classes peut afficher des données statistiques ou des données d'organisation, respectivement sous la forme de classes d'implémentation et de classes logiques. Il peut y avoir chevauchement entre ces deux groupes.

- ✧ Les classes sont représentées par une forme rectangulaire qui est divisée en trois. La partie supérieure affiche le nom de la classe, tandis que la section du milieu contient les attributs de classe. La partie inférieure fait figurer les opérations de classe (également connues sous le nom de méthodes).
- ✧ Ajoutez des formes de classe à votre diagramme de classes pour modéliser la relation entre ces objets. Vous devrez peut-être également ajouter des sous-classes.
- ✧ Utilisez des lignes pour représenter l'association, l'héritage, la multiplicité et d'autres relations entre les classes et sous-classes. Votre style de notation préféré influencera la représentation de ces lignes.[23]

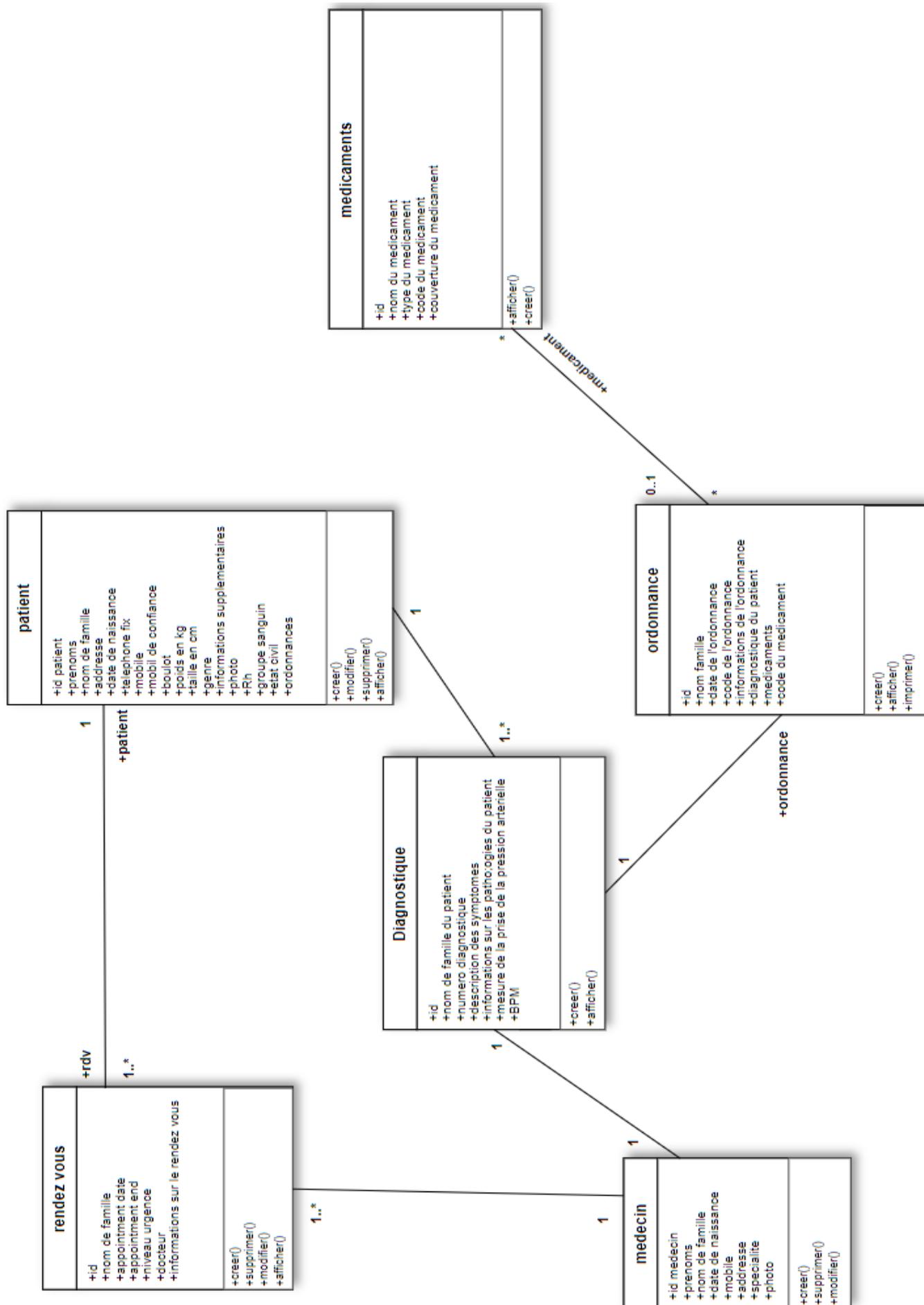


Figure III 20 : Diagramme de classe.

### III. Réalisation :

#### 1. Introduction :

Après avoir terminé la partie d'analyse et conception, nous allons, maintenant, entamer une phase qui consiste à la mise en pratique de ce qui a été présenté dans les chapitres précédents.

Dans ce chapitre, nous allons d'abord vous présenter l'environnement et les outils utilisés, puis nous passerons aux différentes fonctionnalités apportées par les modules développés par Odoo sous forme de captures d'écran avec descriptions, qui vous permettront de mieux comprendre le travail dans la mise en œuvre du projet dans le processus.

#### 2. Les outils utilisés :

##### a) Odoo v13 :

Gestion d'entreprise rapide, simple et efficace - telle est la devise d'Odoo, le principal ERP open source du monde. Et c'est ce qui fait d'Odoo le choix le plus important et le plus préféré des entreprises. Avec la sortie de Odoo 13, l'ERP open source est devenu encore plus adapté et robuste pour répondre aux besoins diversifiés des entreprises. Avec Odoo 13, les utilisateurs peuvent bénéficier de meilleures conceptions et personnalisations. [24]



##### b) PostgreSQL :

PostgreSQL est un système de base de données objet-relationnel puissant et open-source qui utilise et étend le langage SQL combiné à de nombreuses fonctionnalités permettant de stocker et de faire évoluer en toute sécurité les charges de travail de données les plus complexes. Les origines de PostgreSQL remontent à 1986, dans le cadre du projet POSTGRES de l'Université de Californie à Berkeley. La plate-forme de base est développée activement depuis plus de 30 ans.



PostgreSQL s'est forgé une solide réputation pour son architecture éprouvée, sa fiabilité, l'intégrité de ses données, son ensemble de fonctionnalités robustes, son extensibilité et l'engagement de la communauté open-source derrière le logiciel à fournir constamment des solutions performantes et innovantes. PostgreSQL fonctionne sur tous les principaux systèmes d'exploitation, est conforme à la norme ACID depuis 2001 et dispose de puissants modules complémentaires, tels que la populaire extension de base de données géospatiale PostGIS. Il n'est pas surprenant que PostgreSQL soit devenu la base de données relationnelle open-source de choix pour de nombreuses personnes et organisations.[25]

### c) PyCharm :

PyCharm est un environnement de développement intégré (IDE) Python dédié qui fournit une large gamme d'outils essentiels pour les développeurs Python, étroitement intégrés pour créer un environnement pratique pour le développement productif de Python, du web et des sciences des données.[26]



✚ Présentation du langage utilisé

### d) Python :

Python est un langage de programmation puissant et facile à apprendre. Il possède des structures de données de haut niveau efficaces et une approche simple mais efficace de la programmation orientée objet. La syntaxe élégante et le typage dynamique de Python, ainsi que sa nature interprétée, en font un langage idéal pour le développement de scripts et d'applications rapides dans de nombreux domaines et sur la plupart des plates-formes.



L'interpréteur Python et la bibliothèque standard étendue sont disponibles gratuitement sous forme de source ou de binaire pour toutes les principales plates-formes sur le site Web de Python, <https://www.python.org/>, et peuvent être distribués librement. Le même site contient également des distributions et des pointeurs vers de nombreux modules, programmes et outils Python tiers gratuits, ainsi qu'une documentation complémentaire.

L'interpréteur Python est facilement extensible avec de nouvelles fonctions et de nouveaux types de données implémentés en C ou C++ (ou d'autres langages appelables depuis le C). Python convient également comme langage d'extension pour les applications personnalisables.[27]

#### - Python est un langage :

- ✧ Vise à générer du code de haute qualité, portable et facile à intégrer : grâce à sa syntaxe claire, cohérente et concise, Python permet aux développeurs de générer du code de haute qualité, lisible et maintenable.
- ✧ Des modules de test sont fournis dès le début, Python est un langage agile. Le terme agile vient à l'origine de la méthode de programmation agile, très proche de la programmation itérative. Cette approche réduit les risques liés à la conception de logiciels et introduit, entre autres, le principe du test continu du code.

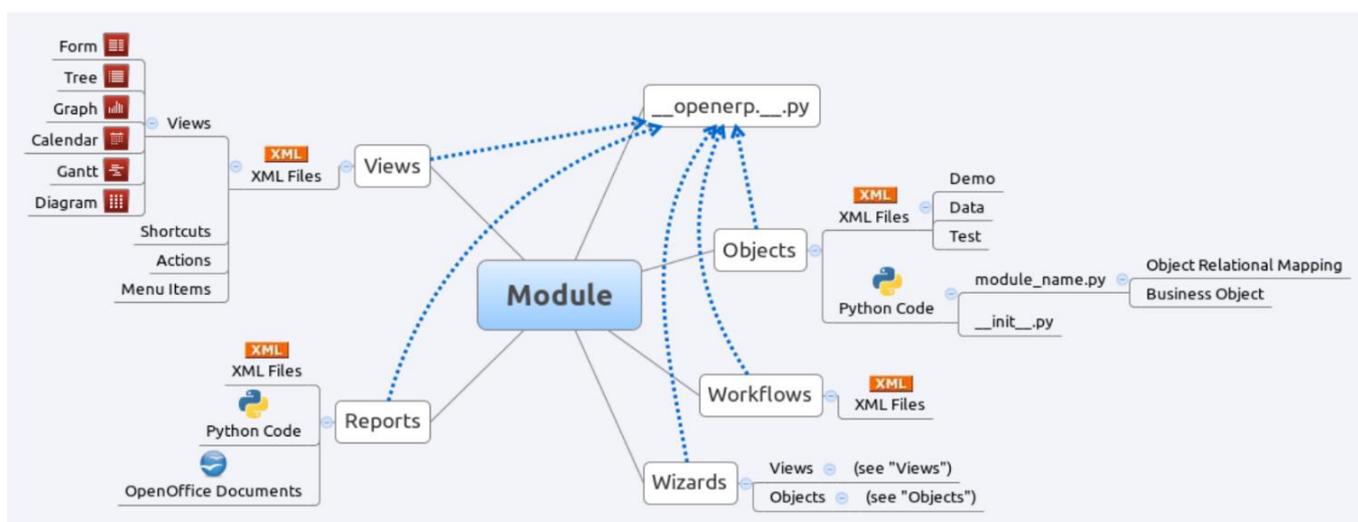
- ✧ Avancé, orienté objet et totalement gratuit : Python permet la programmation orientée objet, même sans l'imposer. Tous les mécanismes objets de base sont implémentés, et toutes les données manipulées sont des instances de classes, tout comme Small Talk ou le langage Ruby.
- ✧ Dynamique : Dans la plupart des implémentations, le code source n'est pas compilé comme dans des langages comme C ou Pascal, mais exécuté dynamiquement. C'est ce qu'on appelle un langage interprété. Ce mode de fonctionnement rend la programmation plus souple car le programme peut être modifié en cours d'exécution.

**e) XML :**

Le XML (Extensible Markup Language) est un langage de balisage similaire au HTML, mais sans balises prédéfinies à utiliser. Au lieu de cela, vous définissez vos propres balises conçues spécifiquement pour vos besoins. Il s'agit d'un moyen puissant de stocker des données dans un format qui peut être conservé, recherché et partagé. Plus important encore, le format fondamental du XML étant normalisé, si vous partagez ou transmettez du XML entre systèmes ou plates-formes, localement ou sur Internet, le destinataire peut toujours analyser les données grâce à la syntaxe XML normalisée.[28]



- Les fichier XML qui contient les vues, les actions et les menus.



**Figure III 21 : Structure d'un module ERP.**

**f) Qweb :**



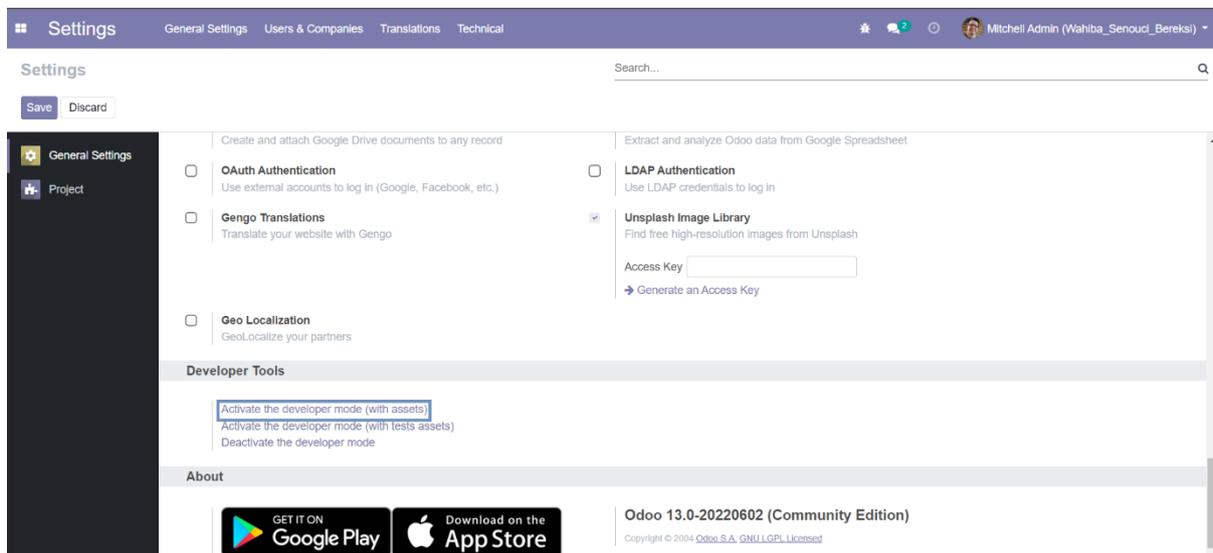
Qweb est le principal moteur de création de modèles utilisé par Odoo. Il s'agit d'un moteur de modèles XML utilisé principalement pour générer des fragments et des pages HTML. Qweb fournit plusieurs outils pour créer un rapport. En utilisant Qweb, nous pouvons manipuler les données très facilement.[29]

**3. Présentation de l'application modulaire :**

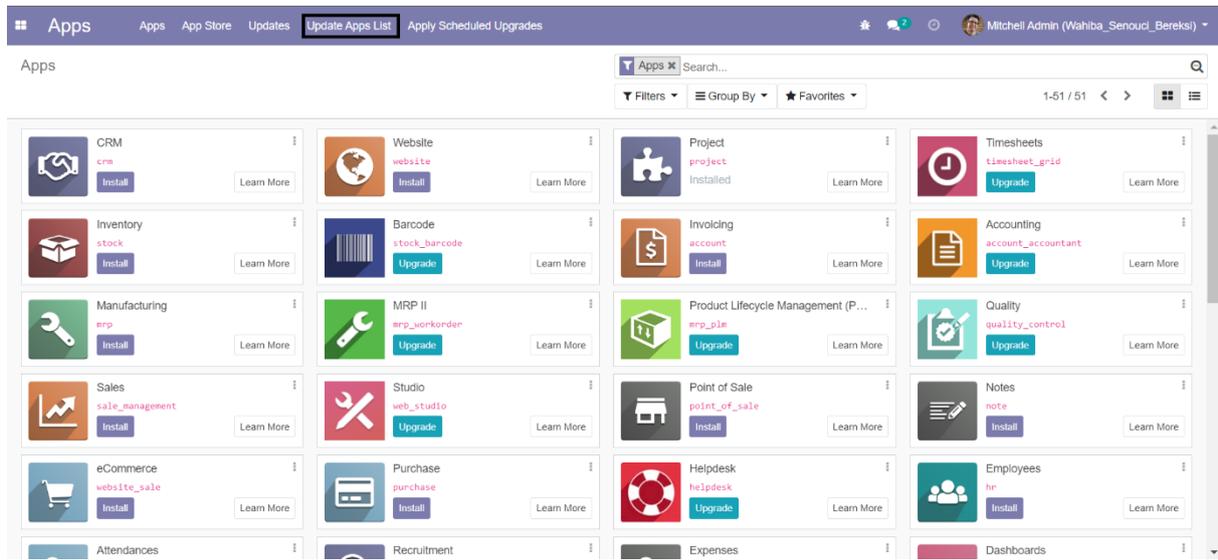
 Création d'un module Odoo

Etapes pour créer/installer le module chez Odoo :

- Créez un dossier puis définissez le nom technique du module en vous assurant qu'il ne contient pas d'espace, par exemple : " gestion\_des\_patients ".
- Créez ensuite un fichier "\_\_init\_\_.py" à l'intérieur du dossier, vous pouvez définir des paquets/fichiers python dans ce fichier.
- Créez ensuite un fichier "\_\_openerp\_\_.py", vous pouvez définir le nom du module (s'il n'est pas défini, le nom sera 'Unnamed') et quelques autres métadonnées.
- Maintenant, connectez-vous à votre Odoo.
- Ensuite, ouvrez le mode développeur de votre Odoo à partir de "settings" de Odoo.



- Puis allez dans le menu "Apps".
- Dans le menu "Apps", vous verrez un bouton "Update App List".



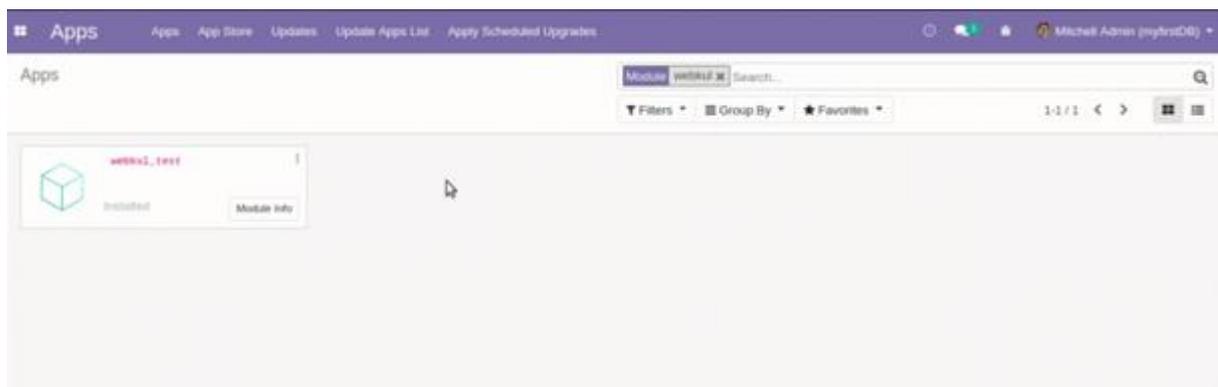
## Module Update

### Module Update Result

Click on Update below to start the process...



- Cliquez sur ce bouton.
- Ensuite, vérifiez le module en utilisant le nom du module.
- Après cela, vous trouverez votre module dans la liste des applications.
- Cliquez ensuite sur le bouton "Installer" afin d'installer votre module au niveau d'Odoo.



## A. Présentation des interfaces :

### a) Gestion des patients :



### Gerer les patients

Interface qui stock tout les dossiers des patients entrer par la reception ou le docteur.

Gestion des Patients

Search...

Filters Group By Favorites 1-6 / 6 < >

ID	Nom de famille	Genre	Date de naissance
1	Selka	Homme	02/14/1979
2	Diden	Homme	06/01/1960
3	Ghizlene	Femme	06/08/1971
4	Salim	Homme	01/30/1980
5	Yasmine	Femme	05/02/1990
6	Imene	Femme	05/02/1980

### b) Gestion des rendez-vous :



### La reception peut planifier des rendez-vous

Les agents de la reception peuvent planifier un rendez-vous pour le patient afin que le medecin puisse les consulter.

Gestion des rendez vous

Search...

Filters Group By Favorites 14 / 4 < >

ID	Nom de famille	Niveau urgence	Appointment Date	Appointment End	Docteur
1	Charlie	Normal	06/18/2022 14:35:00	06/19/2022 14:35:00	Psi
2	Alpha	Urgence	06/18/2022 14:35:27	06/21/2022 14:35:28	Mu
3	Sigma	Normal	06/18/2022 14:35:45	06/22/2022 14:35:45	Psi
4	Theta	Urgence medicale	06/18/2022 14:36:04	06/20/2022 14:36:04	Tau

Gestion des rendez vous (Jun 12 – 18, 2022)

Search...

Today Day Week Month Filters Favorites

Week 25	Sun 12	Mon 13	Tue 14	Wed 15	Thu 16	Fri 17	Sat 18
All day							
7:00							
8:00							
9:00							2022-06-18 06:35:27
10:00							
11:00							
12:00							
13:00							
14:00							
15:00							2022-06-18 12:36:04
16:00							
17:00							2022-06-18 14:35:00
18:00							2022-06-18 15:35:45
19:00							

Jun 2022

Sun	Mon	Tue	Wed	Thu	Fri	Sat
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

Nom de famille

- Charlie
- Alpha
- Sigma
- Theta

## c) Inventaire des médicaments :



### Medicaments.

Gestion de l'inventaire medical afin de faciliter la generation des ordonnances.

Inventaire des medicaments

Search...

Create Import Filters Group By Favorites 1-5 / 5

<input type="checkbox"/> Nom du medicament	Code du medicament	Type du medicament	Couverture du medicament
<input type="checkbox"/> Doliprane 500 gelules	0101	Anti inflammatoire	Medicament couvert
<input type="checkbox"/> Augmentin	0201	Anti biotique	Medicament couvert
<input type="checkbox"/> Smecta sachets	0102	Anti inflammatoire	Medicament non couvert
<input type="checkbox"/> Tirlor	0301	Anthistaminique	Medicament couvert
<input type="checkbox"/> Doliprane 1000 gelules	0103	Anti inflammatoire	Medicament non couvert

### d) Gestion des ordonnances :



**Ordonnances.**

Generer des ordonnances suivant le diagnostique donne par le medecin. Interface qui stock tout les dossiers des patients entrer par la reception ou le docteur.

Gestion des ordonnances

Search...

Create Import

Filters Group By Favorites 1-4 / 4 < >

<input type="checkbox"/>	Date de l'ordonnance	Date de l'ordonnance	Informations de l'ordonnance	Medicaments
<input type="checkbox"/>	06/08/2022	06/08/2022	Sachets a prendre tout les matins a jeun.	1 record
<input type="checkbox"/>	06/05/2022	06/05/2022	Un comprimé de chaque a prendre apres les repas.	2 records
<input type="checkbox"/>	06/03/2022	06/03/2022	Un sachet a prendre le prendre et un autre le soir apres avoir mange.	1 record
<input type="checkbox"/>	06/15/2022	06/15/2022	Un comprimé a prendre chaque matin apres le café.	1 record

### e) Gestion des diagnostiques :



**Diagnostiques.**

Le docteur peut emettre un diagnostique pour ensuite l'insérer dans l'ordonnance.

Gestion des diagnostiques

Search...

Create Import

Filters Group By Favorites 1-4 / 4 < >

<input type="checkbox"/>	Numero diagnostique	Description des symptomes	Informations sur les pathologies du patient
<input type="checkbox"/>	01	Nez qui coule et yeux qui grattent.	Allergie saisonniere.
<input type="checkbox"/>	02	Vomissements et mal de ventre.	Intoxication alimentaire.
<input type="checkbox"/>	03	Fievre et mal de tete.	Grippe saisonniere.
<input type="checkbox"/>	04	Mal de tete.	Aucune pathologie detectee.

## f) Médecins enregistrés :



### Medecins.

Gestion de l'effectif qui est gere par le directeur.

Medecins enregistrés

Search...

Create Import

Filters Group By Favorites

1-5 / 5

<input type="checkbox"/>	Nom de famille	Prenoms	Date de naissance	Adresse	Mobile	specialite
<input type="checkbox"/>	Benyelles	Wahiba	07/30/1979	86 rue xxx	557.894.865	Cardiologue
<input type="checkbox"/>	Gaouar	Mustapha	04/08/1989	51 rue xxx	558.956.324	Dermatologue
<input type="checkbox"/>	Lallam	Feryel	05/08/1956	12 rue xxx	558.963.745	Generaliste
<input type="checkbox"/>	Nacef	Abderrahmene	02/05/1980	26 rue xxx	552.689.412	Nephrologue
<input type="checkbox"/>	Bouabdellah	Abedeslam	11/25/1970	39 rue xxx	554.897.632	Pediatre

## B. Les accès pour les différents acteurs :

### 1. Directeur :

Il peut gérer les utilisateurs en plus de pouvoir gérer les médecins enregistrés au cabinet, avec un accès total au module sans restriction au niveau des interfaces.



### 2. Médecine :

Le docteur peut gérer les patients, les rendez-vous, l'inventaire des médicaments, les ordonnances et les diagnostics à l'exception de la gestion des médecins enregistrés.



### 3. Secrétaire :

Les agents de la réception peuvent juste gérer les patients ainsi que les rendez-vous.



#### **IV. Conclusion :**

A la fin de ce chapitre, nous avons conçu et analysé notre application. Nous suivons une solution basée sur la conception de la modélisation, en utilisant le langage de modélisation UML pour spécifier les cas d'utilisation, concevoir des diagrammes de séquence et développer le diagramme de classe.

Nous avons défini les différentes tables de la base de données et leurs relations, le fonctionnement des différents outils et systèmes de l'environnement de développement.

À la fin de ce chapitre, les différents outils et logiciels qui nous permettent de créer des applications, les langages de programmation utilisés et quelques interfaces pour développer des applications sont introduits.

## *Conclusion Générale*

Dans notre projet de fin d'études, il incluait la création d'un module Odoo qui gère le suivi des patients, dont l'objectif est de fournir aux utilisateurs un outil leur facilitant la tâche. Pour mettre en œuvre ce projet, nous avons d'abord introduit les définitions et les concepts de Odoo et de l'ERP, puis nous avons étudié la situation existante, c'est-à-dire comprendre le fonctionnement et mettre en évidence les lacunes ou les carences du système actuel et le besoin de l'utilisateur de faciliter le fonctionnement de l'application. Pour la conception, nous utilisons UML comme langage de modélisation, ce qui nous permet de décrire la fonctionnalité complète du système. Enfin, nous exposons les différentes fonctionnalités de notre application à travers l'interface lors de la phase d'implémentation.

La réalisation de ce projet de fin d'études est une aubaine pour nous, compte tenu de ses nombreux reportages sur l'accès au savoir. En effet, cela nous a permis d'approfondir notre compréhension des systèmes d'information d'entreprise, de découvrir le domaine médical, de comprendre et de garder un œil attentif sur la complexité des tâches auxquelles nos employés sont confrontés, car cela nous a familiarisés avec les outils de développement d'Odoo (PYTHON, XML, PyCharm, POSTGRESQL). Maîtriser les concepts et les outils de la conception orientée objet et de la méthode de conception UML. Notre application proposée offre à divers utilisateurs une plus grande flexibilité et efficacité lors de l'exécution de tâches. Notre module propose diverses fonctionnalités, mais après avoir maîtrisé l'ERP Odoo, certaines améliorations et extensions peuvent être apportées à ce module

# Bibliographie

- [1] Miika Nissi, Bachelor's Thesis Improving Event Management Implementation on Odoo Business Management Software Year 2021.
- [2] Odoo, About Us. Retrieved August 25, 2021, from Odoo: <https://accounts.odoo.com/page/about-us> , Jun 2022.
- [3] Pinckaers, F. (2013, April 16). The Odoo Story. <https://www.odoo.com/blog/odoo-news5/the-odoo-story-56>, Jun 2022.
- [4] Système d'exploitation GNU, Free Software Foundation. (2021, August 21). What is Free Software? <https://www.gnu.org/philosophy/free-sw.html> , Jun 2022.
- [5] Système d'exploitation GNU, Free Software Foundation. (2018, December 15). What is Copyleft? <https://www.gnu.org/licenses/copyleft> , Jun 2022.
- [6] Odoo, (2015, February 3). Adapting our open-source license. <https://www.odoo.com/blog/odoo-news-5/adapting-our-open-source-license-245>
- [7] © 2022 GitHub, Inc, Odoo Community Association. (2018, August 24). OCA Guidelines. <https://github.com/OCA/odoo-> , Jun 2022.
- [8] Oracle, what is ERP? <https://www.oracle.com/erp/what-is-erp/> , Jun 2022.
- [9] Techterms, CMS (Content Management System) Definition. <https://techterms.com/definition/cms> , Jun 2022.
- [10] web Technology surveys, Q-Success. (2021, August 27). Usage statistics and market share of WordPress. <https://w3techs.com/technologies/details/cm-wordpress> , Jun 2022.
- [11] Wikipedia, Hôpital, <https://fr.wikipedia.org/wiki/H%C3%B4pital> , Jun 2022.
- [12] Wikipédia, Système d'information hospitalier [https://fr.wikipedia.org/wiki/Syst%C3%A8me\\_d%27information\\_hospitalier](https://fr.wikipedia.org/wiki/Syst%C3%A8me_d%27information_hospitalier) , Jun 2022.
- [13] Mlle Khadija MASSKOUB, « Mise en place d'un système d'information hospitalier intégré dans OpenERP, Module médical » Rapport PFE 2010/2011, pg 17, 18.
- [14] conseil national médecin, Le dossier du patient, <https://www.conseil-national.medecin.fr/medecin/exercice/dossier-patient#:~:text=Le%20dossier%20du%20patient%20contient,au%20patient%2C%20%C3%A0%20sa%20demande> , Jun 2022.

- [15] BENOUADAH Ali et GUENDOOUSSI Norelhouda, Conception et réalisation d'une application pour la gestion du dossier médical personnel (Etude de cas : CHU Algérien) , 2017.
- [16] Université Médicale Virtuelle Francophone, - Support de Cours (Version PDF) - « le dossier médical », Comité éditorial pédagogique de l'UVMaF, cours 2011-2012, page 3.
- [17] - Université Médicale Virtuelle Francophone - campus cerimes, Le dossier médical , Mise à jour : 07/01/2013 - Mentions légales - © 2011-2012 UVMF Comité éditorial pédagogique UVMaF [http://campus.cerimes.fr/maieutique/UE-sante-societe-humanite/dossier\\_medical/site/html/3.html#:~:text=2%20%2D%20B%C3%A9n%C3%A9fice%20du%20dossier%20m%C3%A9dical,de%20la%20sant%C3%A9%20de%20population](http://campus.cerimes.fr/maieutique/UE-sante-societe-humanite/dossier_medical/site/html/3.html#:~:text=2%20%2D%20B%C3%A9n%C3%A9fice%20du%20dossier%20m%C3%A9dical,de%20la%20sant%C3%A9%20de%20population).Jun 2022.
- [18] BERNER.ES, DETMER.DE, SIMBORG.D. Will the wave finally break? A brief view of the adoption of electronic medical records in the United States. Journal of the American Medical Informatics Association. 2005.: p (3-7).
- [19] caducee.net, par A.j. Meille,, M. L. Dassie , Dr. L. Labreze , Le dossier médical informatisé, <https://www.caducee.net/DossierSpecialises/systeme-information-sante/dmi.asp> , Jun 2022.
- [20] lucidchart, Qu'est-ce que le langage UML, <https://www.lucidchart.com/pages/fr/langage-uml> , Jun 2022.
- [21] lucidchart, Qu'est-ce que le langage UML, Diagramme de cas d'utilisation <https://www.lucidchart.com/pages/fr/langage-uml> , Jun 2022.
- [22] lucidchart, Qu'est-ce que le langage UML, Diagramme de séquence <https://www.lucidchart.com/pages/fr/langage-uml> , Jun 2022.
- [23] lucidchart, Qu'est-ce que le langage UML, Diagramme de classe <https://www.lucidchart.com/pages/fr/langage-uml> , Jun 2022.
- [24] Wikipedia, Odoo, <https://fr.wikipedia.org/wiki/Odoo> , Jun 2022.
- [25] PostgreSQL, Copyright © 1996-2022 The PostgreSQL Global Development Group, New to PostgreSQL? <https://www.postgresql.org/> , Jun 2022.
- [26] JetBrains, Get started, <https://www.jetbrains.com/help/pycharm/quick-start-guide.html> , Jun 2022.
- [27] rune book, Le tutoriel Python, <https://runebook.dev/fr/docs/python/tutorial/index> , Jun 2022.

[28] developer Mozilla, XML introduction [https://developer.mozilla.org/en-US/docs/Web/XML/XML\\_introduction](https://developer.mozilla.org/en-US/docs/Web/XML/XML_introduction) , Jun 2022.

[29] Cypress Technologies , Basic Qweb Operations <https://www.cybrosys.com/blog/basic-qweb-operations#:~:text=Qweb%20is%20a%20template%20engine,manipulate%20the%20data%20very%20easily>. Jun 2022.

## ***Résumé***

Cet article appartient au domaine des applications en santé. Dans ce cas, il nous a été proposé de concevoir et de créer un module de gestion des dossiers médicaux des patients basé sur Odoo.

Dans ce travail, nous présentons Odoo de manière générale, où nous définissons le profil médical d'un patient de manière globale.

Pour modéliser nos modules, nous avons choisi la forme UML. Nous avons choisi ce dernier pour sa simplicité, ses performances et son adaptabilité en termes de design.

Pour implémenter ce module, nous avons choisi le principal ERP open source du monde Odoo v13, nous utilisons le langage de programmation python, PyCharm pour le développement productif de Python, et PostgreSQL pour l'implémentation de l'espace de travail. Dans l'application, nous utilisons XML pour la structuration des données.

**Mots-Clés :** Odoo, ERP Open source, module, python, PyCharm, PostgreSQL, DMP Dossier Médical de patient.

## ***Abstract***

This article belongs to the field of health applications. In this case, we have been proposed to design and create a module for managing patient medical records based on Odoo.

In this work, we present Odoo in a general way, where we define the medical profile of a patient in a global way.

To model our modules, we have chosen the UML form. We chose the latter for its simplicity, its performance and its adaptability in terms of design.

To implement this module, we chose the world's leading open-source ERP Odoo v13, we use the python programming language, PyCharm for productive Python development, and PostgreSQL for the workspace implementation. In the application, we use XML for data structuring.

**Keywords:** Odoo, ERP Open source, module, python, PyCharm, PostgreSQL, DMP Patient Medical Record.

## ملخص

هذه المقالة تنتمي إلى مجال التطبيقات الصحية. في هذه الحالة، عُرض علينا تصميم وإنشاء وحدة إدارة السجلات الطبية للمرضى استنادًا إلى Odoo.

في هذا العمل، نقدم Odoo بشكل عام، حيث نقوم بتعريف الملف الطبي للمريض على مستوى العالم. لنمذجة الوحدات النمطية الخاصة بنا، اخترنا نموذج UML. لقد اخترنا هذا الأخير لما يتميز به من بساطة وأداء وقدرة على التكيف من حيث التصميم.

لتنفيذ هذه الوحدة، اخترنا ERP Odoo v13 الرائد في العالم مفتوح المصدر، ونستخدم لغة برمجة Python و PyCharm لتطوير Python المنتج و PostgreSQL لتنفيذ مساحة العمل. في التطبيق، نستخدم XML لهيكل البيانات.

**الكلمات الرئيسية:** Odoo، ERP مفتوح المصدر، الوحدة النمطية، python، PyCharm، PostgreSQL، السجل الطبي لمرضى DMP.