

Université Abdelmalek Essaadi

Faculté des sciences et techniques de Tanger

Cycle d'ingénieur

Logiciels & Systèmes Intelligents



Année académique : 2020/2021

Projet de fin de module de Conception Orienté Objet UML :

Sujet :

Gestion d'un cabinet médical

Réalisé par : Fatima TLEMZI

Samir AIT-ABOU

Nassima EL JAZOULI

Encadré par : Monsieur Chaker EL AMRANI

Remerciements :

C'est avec toute nos profondes grâtes que nous exprimons nos sincères remerciements à notre professeur du module de conception orienté objet Monsieur Chaker EL AMRANI d'être généreux en partageant avec nous son expérience riche dans le monde des entreprises et d'avoir dirigé ce projet de fin de module. Sans sa disponibilité, sa patience, les conseils qu'il nous a prodigués tout au long de ce semestre, ce travail n'aurait pas pu, sans doute, être mené à son terme.

Table des matières :

INTRODUCTION :	5
A. OUTIL DE MODELISATION :	6
B. TRADUCTION DES BESOINS DU CLIENT :	7
A. REGLES DE GESTION :	7
B. REGLES D'ORGANISATION :	9
C. DEFINITION DU SYSTEME D'INFORMATION :	11
D. MODELISATION AVEC LES DIAGRAMMES DE LA CONDUITE DES PROJETS :	12
A. DIAGRAMME DE PACKAGE :	12
B. DIAGRAMME DE CLASSE PROVISoire POUR CHAQUE PACKAGE:	12
C. DIAGRAMME D'ACTIVITE POUR CHAQUE PACKAGE :	15
D. DIAGRAMME DE CAS D'UTILISATION POUR CHAQUE PACKAGE :	17
E. DIAGRAMME DE SEQUENCE POUR CHAQUE CAS D'UTILISATION :	19
F. DIAGRAMME DE CLASSE POUR CHAQUE PACKAGE :	25
G. DIAGRAMME DE CLASSE GENERAL :	27
H. DIAGRAMME D'ETATS-TRANSITIONS :	28
E. VERIFICATION DES DIFFERENTS DIAGRAMMES :	31
F. MAPPING OBJET-RELATIONNEL :	32
G. LISTAGE DES FONCTIONS DES DIFFERENTS DIAGRAMMES:	33
CONCLUSION :	35

Introduction :


L'informatique étant une science de traitement automatique de données s'avère bénéfique dans tous les domaines qu'ils soient scientifiques ou professionnels, privés ou publics.


Et comme n'importe quel type de projet, un projet informatique nécessite une phase d'analyse, suivi d'une étape de conception. Dans la phase d'analyse, on cherche d'abord à bien comprendre et à décrire de façon précise les besoins des utilisateurs ou des clients. Puis dans la phase de conception, on apporte plus de détails à la solution et on cherche à clarifier les aspects techniques.

Pour cela on utilise UML, qui fait partie des notations les plus utilisées de nos jours, afin de réaliser une conduite de projet pour la gestion d'un cabinet médical.

Ce projet nous permet de concrétiser et améliorer nos connaissances informatiques et théoriques récoltées durant le semestre par le développement d'un projet pratique pris du monde réel.

A. Outil de modélisation :

 **Visual Paradigm** (VP-UML) est un outil UML CASE prenant en charge UML 2, SysML et Business Process Modeling Notation (BPMN) de l'Object Management Group (OMG). En plus de la prise en charge de la modélisation, il fournit des capacités de génération de rapports et d'ingénierie de code, y compris la génération de code. Il peut effectuer une ingénierie inverse des diagrammes à partir du code et fournir une ingénierie aller-retour pour divers langages de programmation.

 **Creatly** est l'outil en ligne qu'on a utilisé pour construire le mapping objet/relationnel.

B. Traduction des besoins du client :

a. Règles de gestion :

L'objectif de ce projet est de créer une application pour gérer un cabinet médical de telle sorte que les tâches du médecin, la secrétaire et l'infirmier soient automatisées.

Les attributs de chaque acteur (le médecin, la secrétaire, l'infirmière) vont être bénéfiques juste au niveau de la consultation du profil, et au cas dont l'un des salariés démissionnera de son travail dans le futur, le médecin –notre acteur principal peut garder leurs dossiers dans l'historique.

Le Médecin :

- Image
- CIN
- Nom
- Prénom
- Nom_utilisateur
- Mot_de_passe
- Fixe
- Email
- Téléphone
- Adresse
- Diplôme

La Secrétaire :

- Image
- CIN
- Nom
- Prénom

- Nom_utilisateur
- Mot_de_passe
- Fixe
- Email
- Téléphone
- Adresse
- Diplôme

L'Infirmière :

- Image
- CIN
- Nom
- Prénom
- Nom_utilisateur
- Mot_de_passe
- Fixe
- Email
- Téléphone
- Adresse
- Diplôme

Les informations collectées concernant les patients vont être stockées au niveau de notre base de données.

Le Patient :

- Id_patient
- CIN
- Nom
- Prénom
- Date_Naissance
- Adresse

- Code_mutuelle
- Téléphone
- Date_visite
- Type_visite (Consultation ou Contrôle)
- Taille
- Poids
- Tension
- Antécédents_médicaux (Toute maladie dont le patient souffre déjà)
- Médicaments_utilisés
- Diagnostic (Médical)
- Ordonnance
- Certificat (pour prouver que le patient a effectivement pris le certificat)
- Analyses
- Dossier_médical (pour prouver que le patient a effectivement pris son dossier médical)
- Facture

b. Règles d'organisation :

- Le médecin c'est l'acteur principal de l'application qu'on veut créer, pour cela il va avoir les droits de l'administrateur, ce qui va lui permettre de modifier ou supprimer ou même ajouter une secrétaire ou une infirmière.
- L'utilisation de l'application doit être facile pour les non-informaticiens.
- Chaque utilisateur de l'application (y compris le médecin, l'infirmière et la secrétaire) doit s'authentifier avant d'accéder à l'application.
- Chaque acteur va remplir ou modifier les informations selon sa spécialité (par exemple dans l'interface de la secrétaire, la case du diagnostic n'est pas modifiable parce qu'elle est réservée au médecin).
- La secrétaire doit gérer les rendez-vous selon la disponibilité du médecin.
- La secrétaire doit saisir le code de mutuelle si le patient le possède.
- Au cas où le patient est en visite de contrôle, l'encaissement s'annule, mais s'il bénéficie des services de santé supplémentaire, le patient doit absolument payer.

- Au cas où le patient a déjà visité le cabinet, une consultation du dossier est nécessaire.

Un cabinet médical contient normalement trois acteurs principaux : le Médecin, l'Infirmière, et la Secrétaire. Chacun d'entre eux a ses propres activités :

La secrétaire a plusieurs rôles dans le cabinet médical. Notamment :

- Gérer les dossiers des patients.
- planifier les horaires de médecin.
- Gérer les rendez-vous.
- Consulter les informations des patients.
- Gérer les encaissements.

Le médecin a un contact direct avec le patient tout en disposant d'un dossier médical déjà rempli par la secrétaire dans les limites de son travail. Son activité principale se repose sur :

- Gérer les consultations.
- Gérer les ordonnances et les certificats.
- Suivre les dossiers des patients.
- Gérer le financement.

L'infirmière joue un rôle important dans le cabinet, on peut dire qu'elle est la main droite du médecin, elle essaye de faciliter le travail au médecin en faisant les tâches suivantes :

- Consulter le dossier d'un patient.
- Participer à des actions de santé.
- Modifier le dossier médical d'un patient.

C. Définition du système d'information :

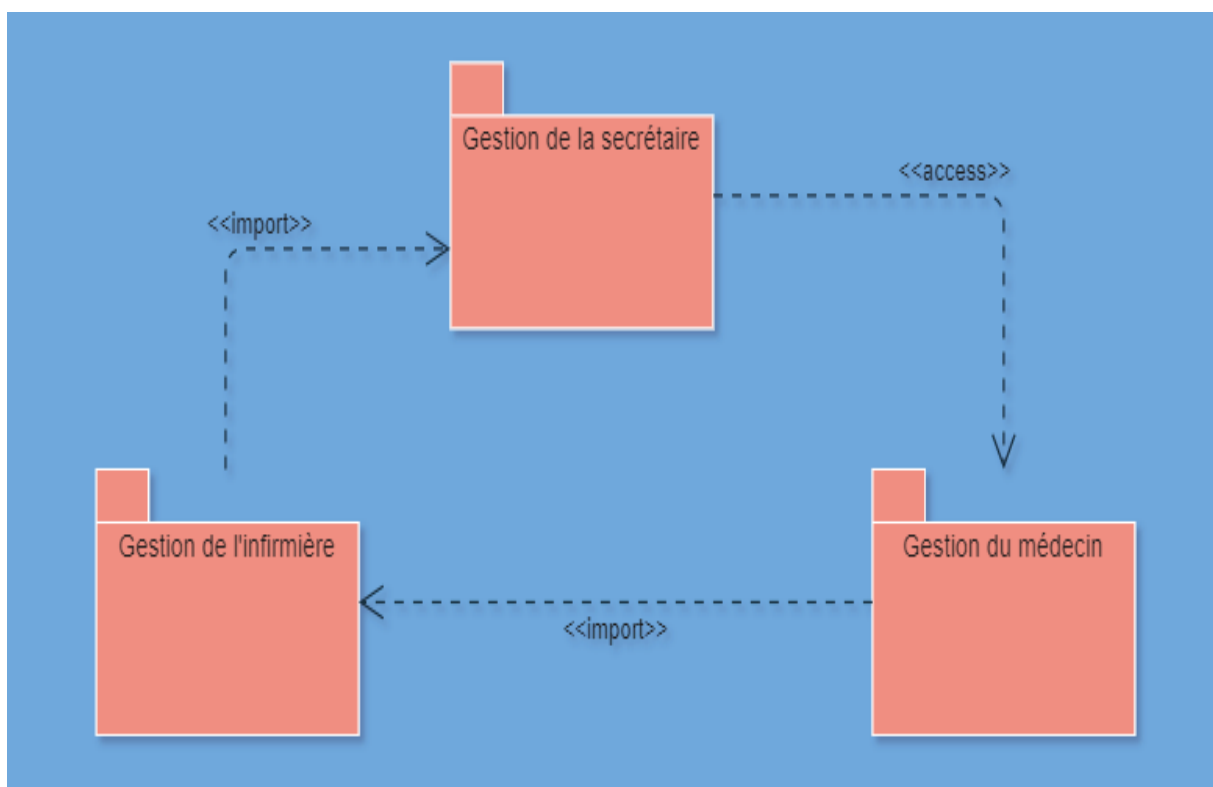
D'après avoir bien analysé les besoins du client et les traduire en règles de gestion et d'organisation : on a pu déterminer l'objectif sur lequel on va travailler qui sera la gestion d'un cabinet médical en automatisant les tâches des différents acteurs (médecin, infirmière et secrétaire) pour mieux traiter les données des patients.

D. Modélisation avec les diagrammes de la conduite des projets :

a. Diagramme de package :

Les diagrammes de packages sont utilisés pour refléter l'organisation de packages et de leurs éléments. Dans le but de diviser pour mieux régner, on décompose notre Système d'Information en des sous-systèmes pour rendre le projet plus clair et bien structurer les diagrammes qui vont venir par la suite.

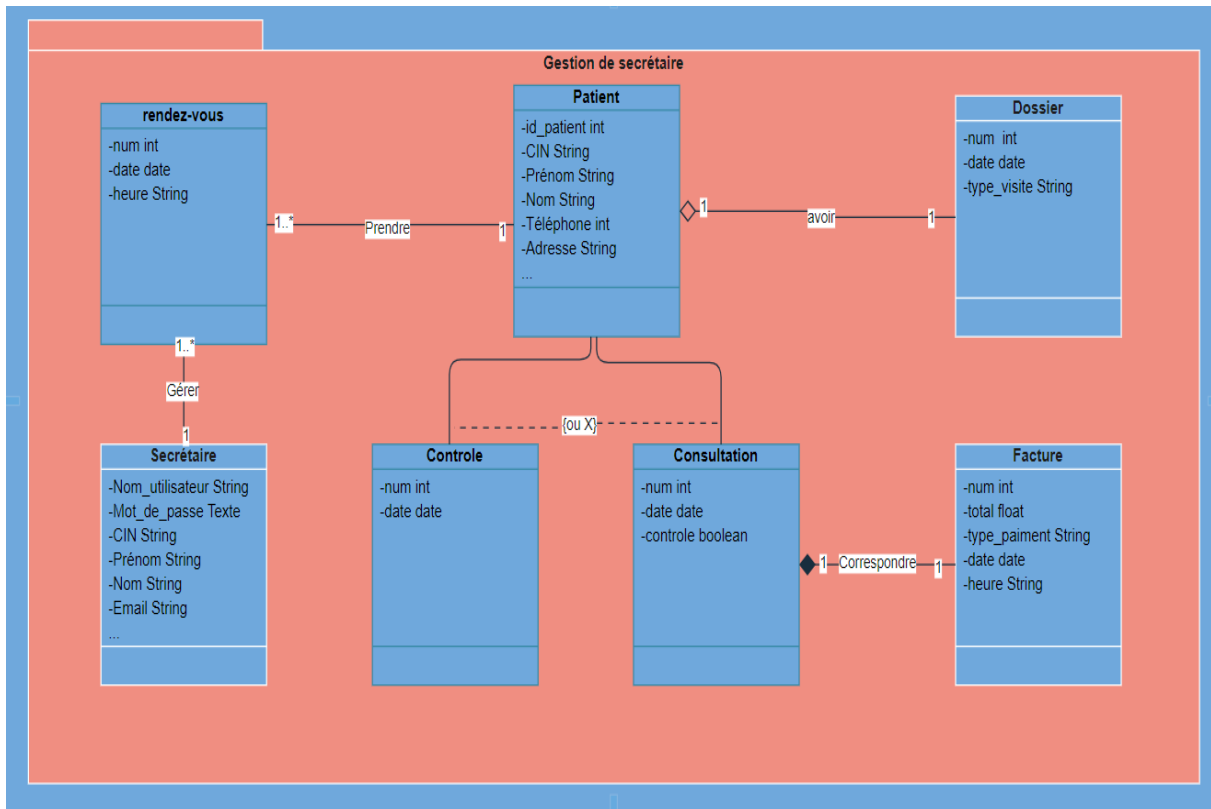
Le diagramme qui suit présente le diagramme de package de notre système d'information :



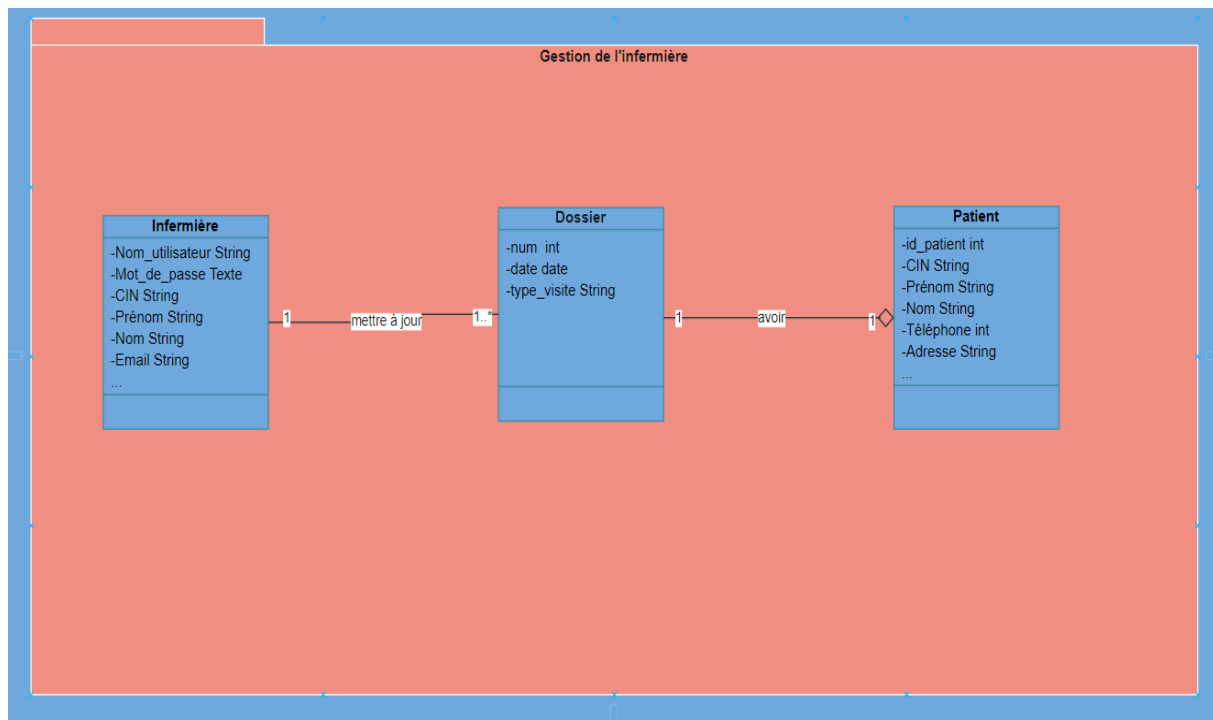
b. Diagramme de classe provisoire pour chaque package:

Dans cette étape, on va faire un diagramme de classe provisoire pour chaque package, dont le but est d'avoir une vue générale sur le projet sur lequel on travaille.

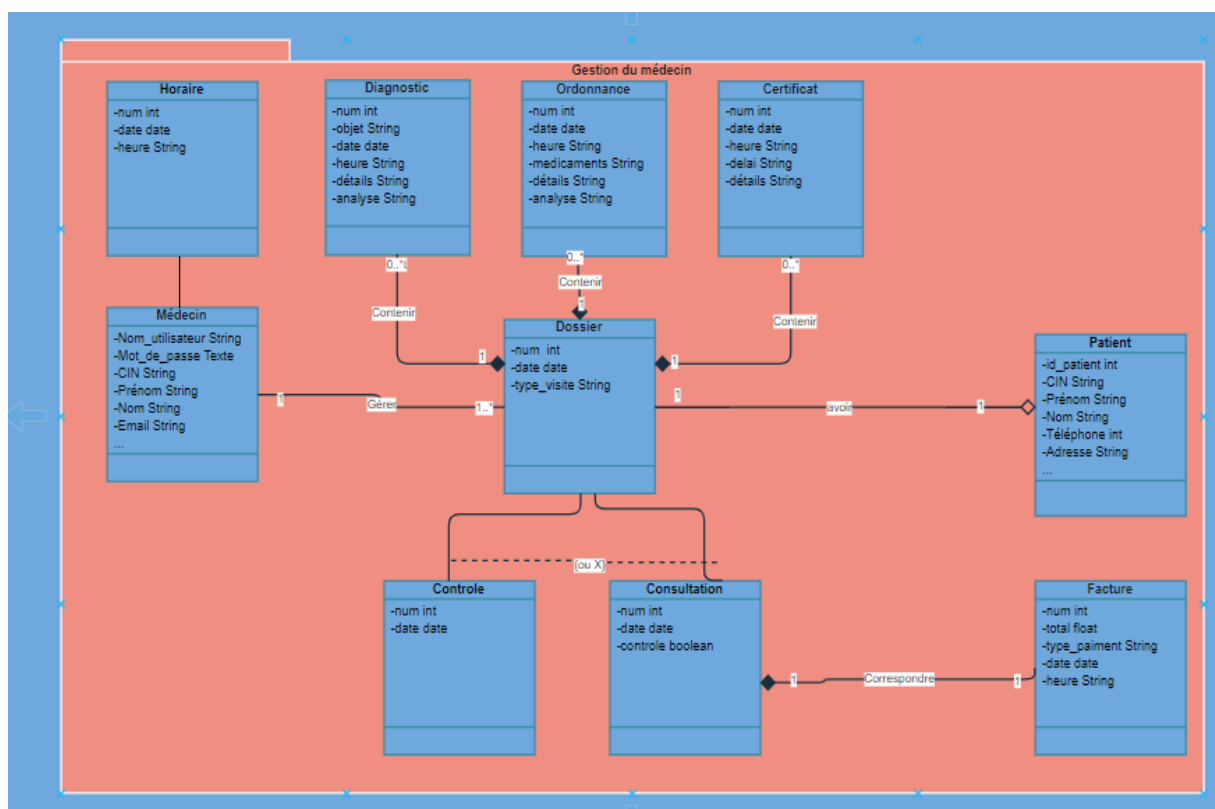
Le diagramme de classe provisoire pour le package gestion de la secrétaire est le suivant :



Le diagramme de classe provisoire pour le package gestion de l'infirmière est le suivant :



Le diagramme de classe provisoire pour le package gestion du médecin est le suivant :

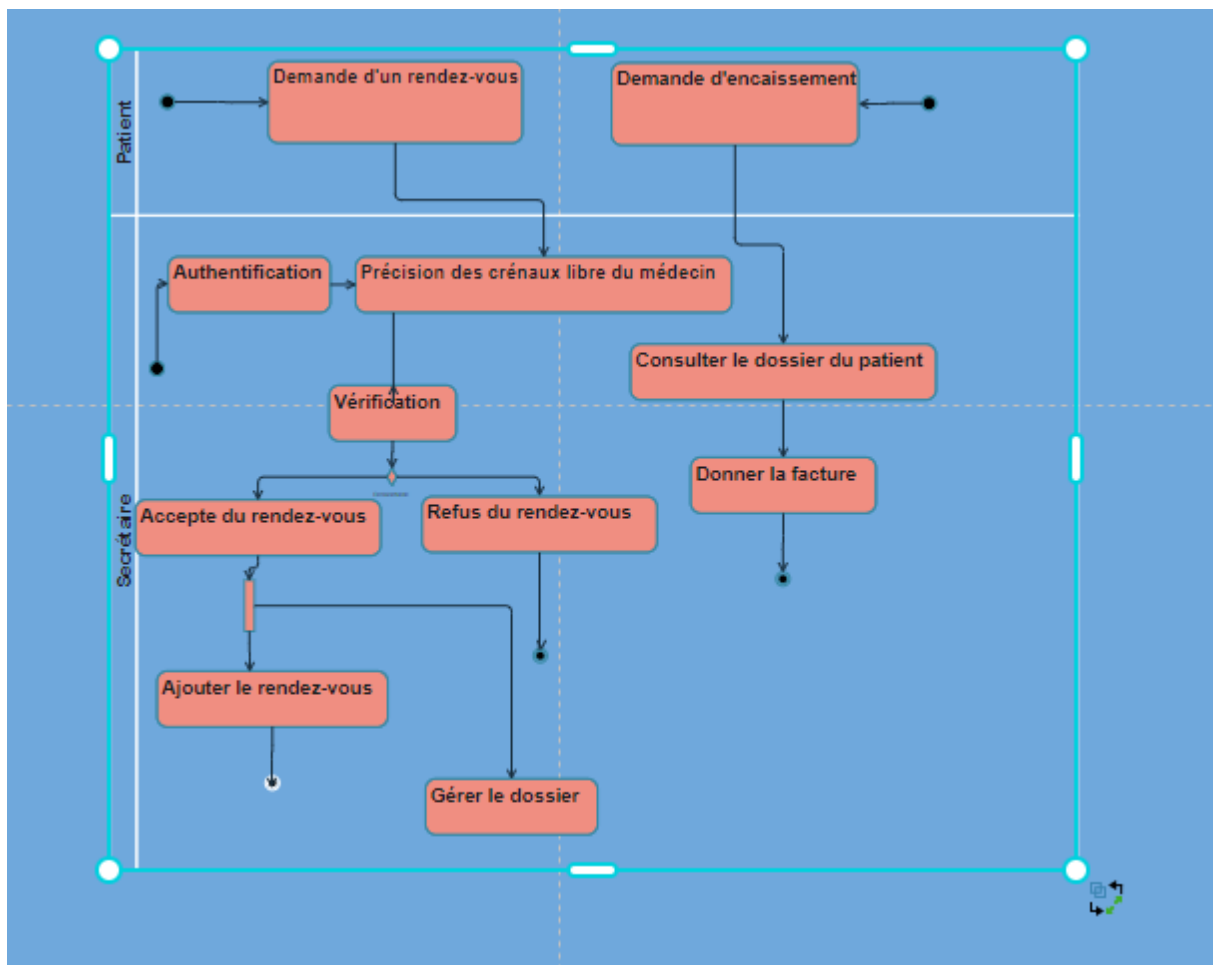


c. Diagramme d'activité pour chaque package :

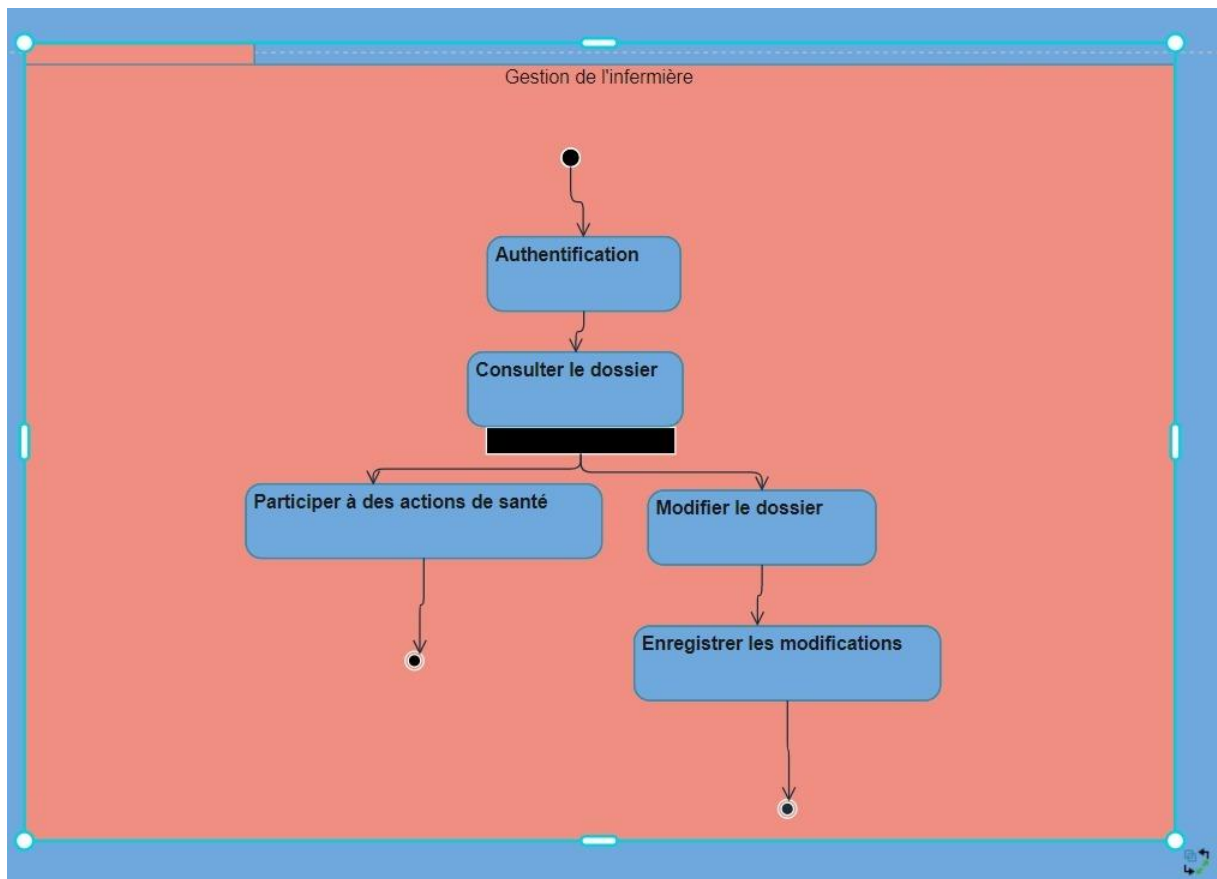
L'objectif du diagramme d'activité est de trouver les différentes opérations et fonctions, il décrit le fonctionnement du projet et permet de le mieux comprendre.

Cette partie sera consacrée pour la présentation des différents diagrammes d'activités pour les différents packages présentés précédemment :

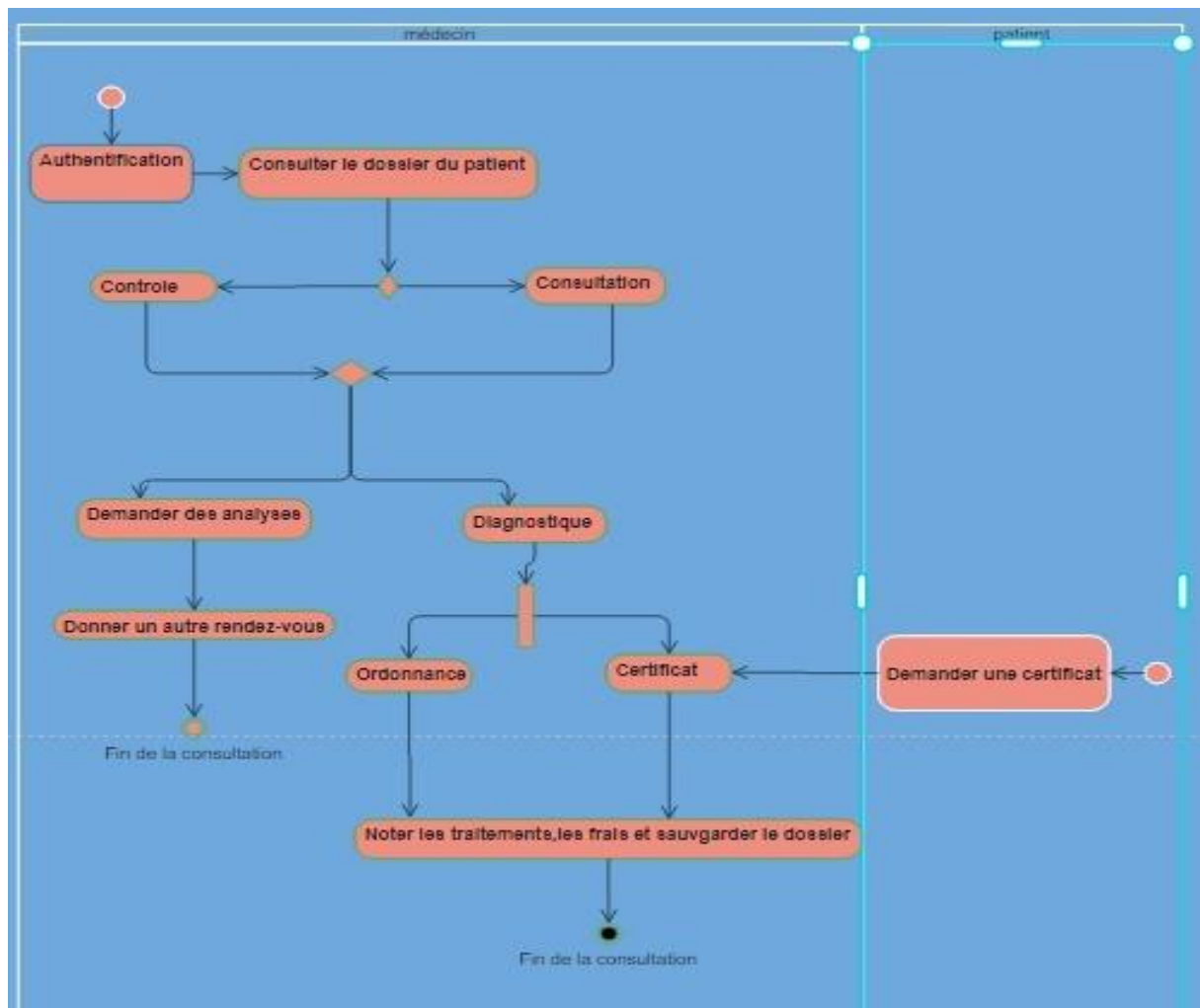
Le diagramme suivant va être relatif au package gestion secrétariat :



Le diagramme ci-dessous est un diagramme d'activité du package gestion d'infirmierie :



Le diagramme suivant est un diagramme d'activité du package gestion du médecin :

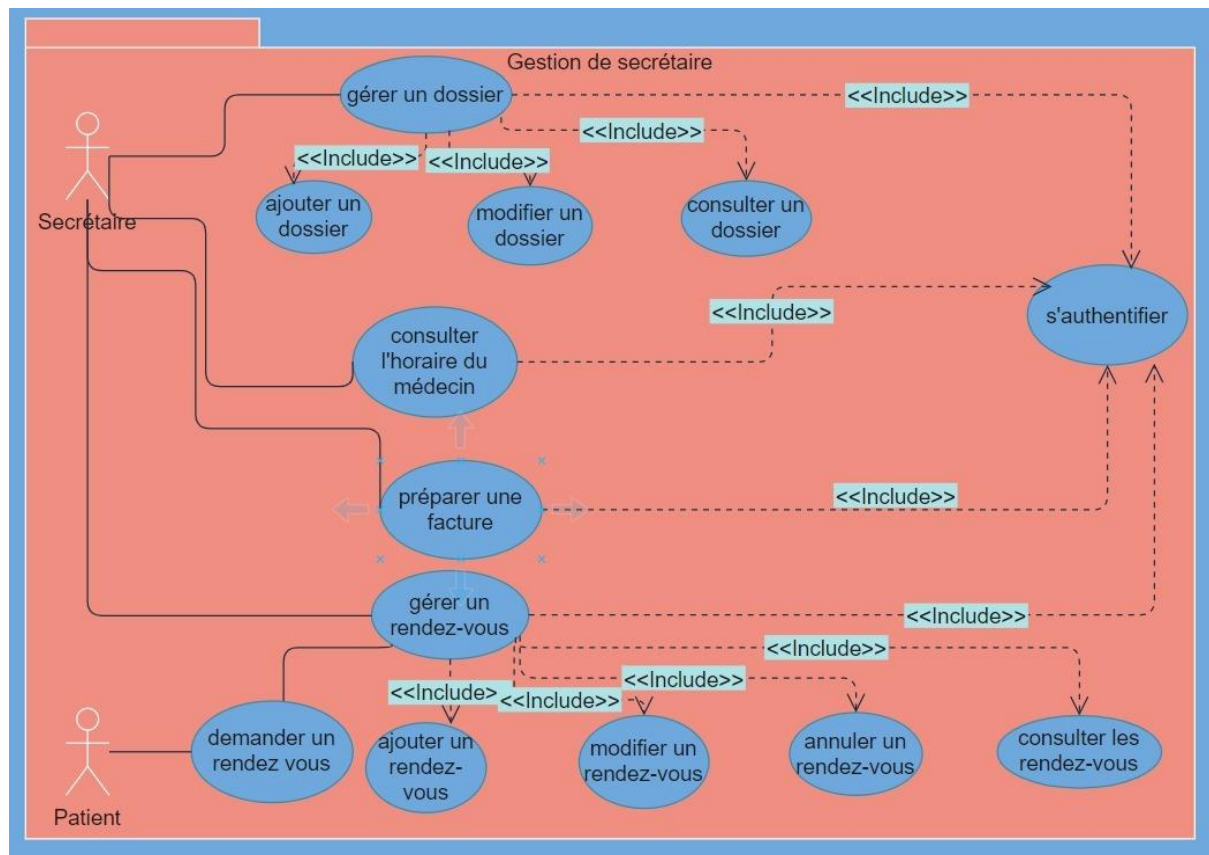


d. Diagramme de cas d'utilisation pour chaque package :

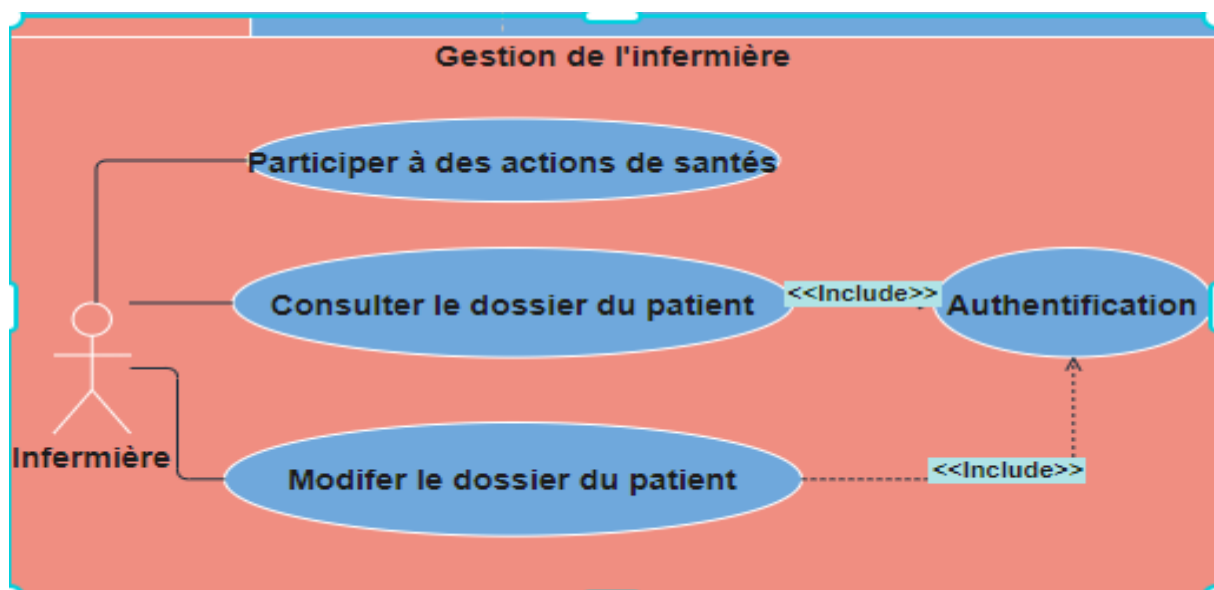
Le diagramme de cas d'utilisation a un but très important à la conduite des projets qui est la recherche des différents acteurs qui agissent dans une partie du système d'information et en quoi agissent-t-ils, c'est à dire les types de services qu'ils présentent au-dedans du système d'information.

Les diagrammes qui suivent vont illustrer les diagrammes de cas d'utilisation pour les trois diagrammes de packages qu'on a déjà trouvé dans une étape précédente :

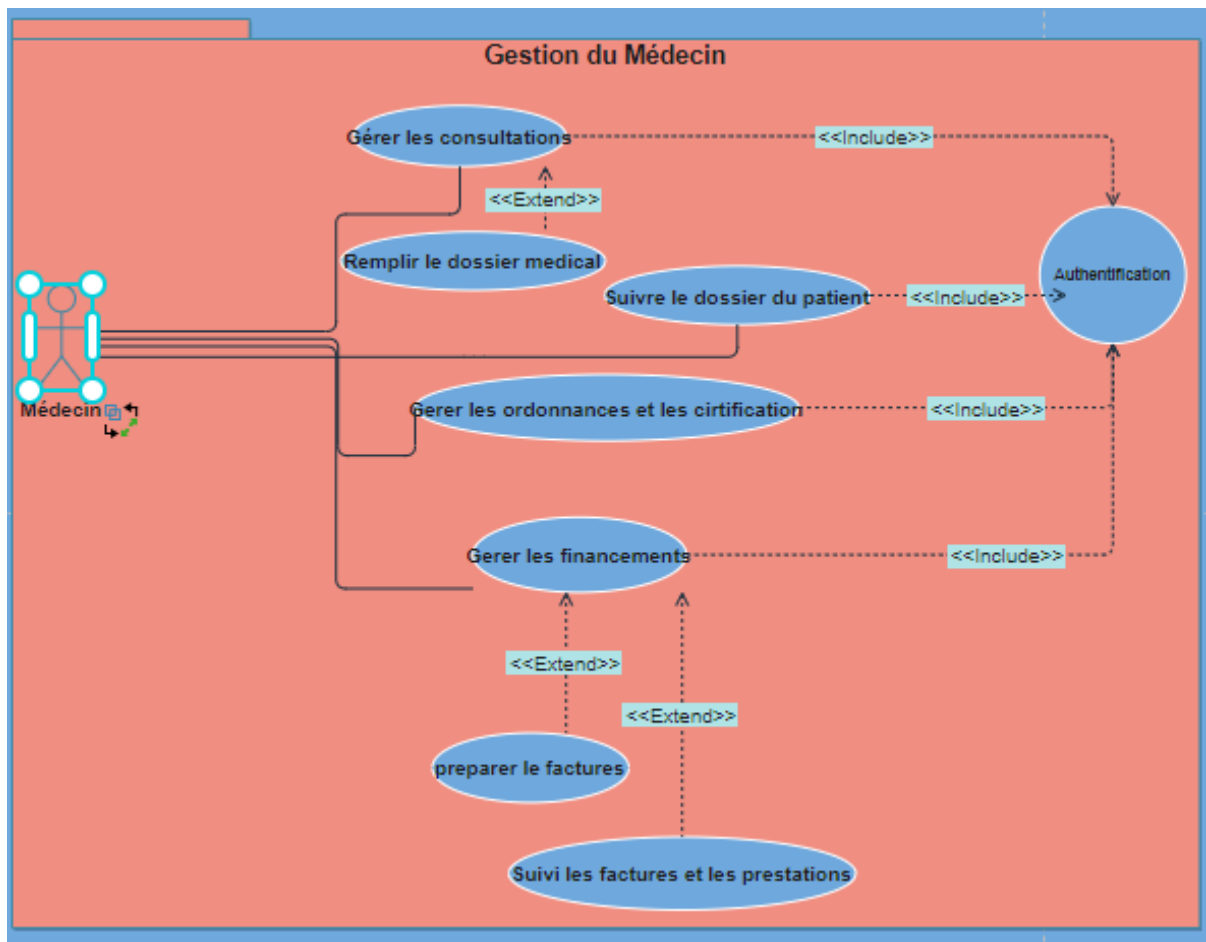
Le diagramme de cas d'utilisation pour le package gestion secrétariat :



Le diagramme de cas d'utilisation pour le package gestion d'infirmierie :



Le diagramme de cas d'utilisation pour le package gestion du médecin :



e. Diagramme de séquence pour chaque cas d'utilisation :

Le diagramme de séquence, décrit les scénarios de chaque cas d'utilisation en mettant l'accent sur la chronologie des opérations en interaction avec les objectifs. Il montre une interférence présentée en séquence dans le temps. En particulier, il montre aussi les objets qui participent à l'interaction par leur «la barre d'activation» et les messages qu'ils échangent présentés en séquence dans l'axe temporel. Voici quelques notions de base du diagramme :

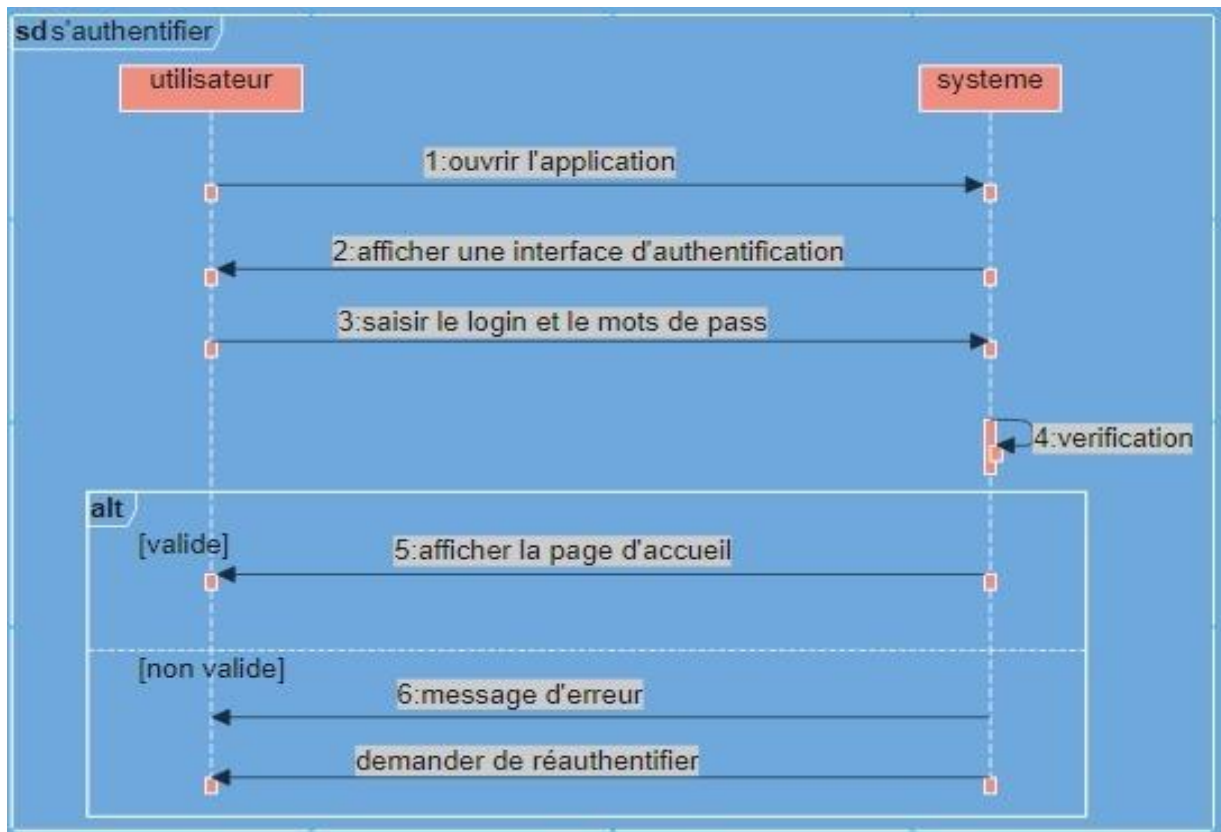
Scénario : une liste d'actions qui décrivent une interaction entre un acteur et le système.

Interaction : un comportement qui comprend un ensemble de messages échangés par un ensemble d'objets dans un certain contexte pour accomplir une certaine tâche.

Message : Un message définit une communication particulière entre des lignes de vie (objets ou acteurs).

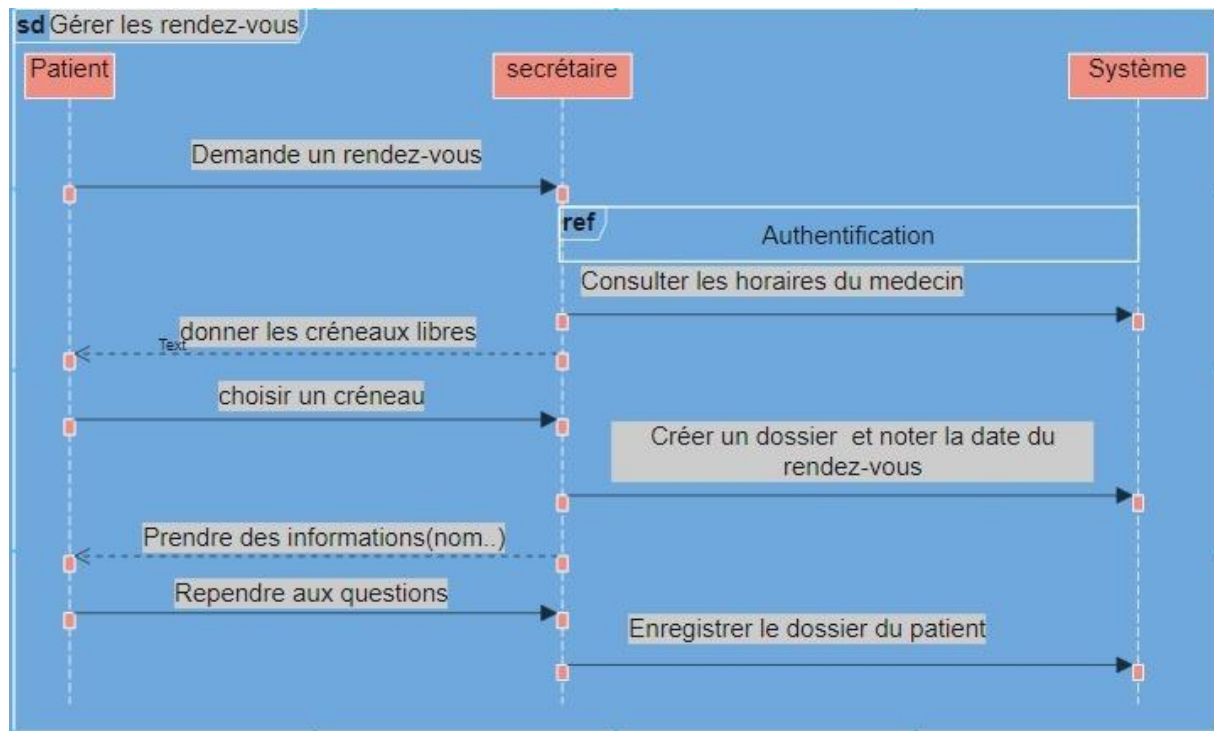
On va essayer par la suite de construire un diagramme de séquence pour chaque cas d'utilisation déjà trouvée dans les diagrammes précédents :

Le diagramme qui suit présente l'enchainement de la phase «authentification» qui concerne tous les autres diagrammes :

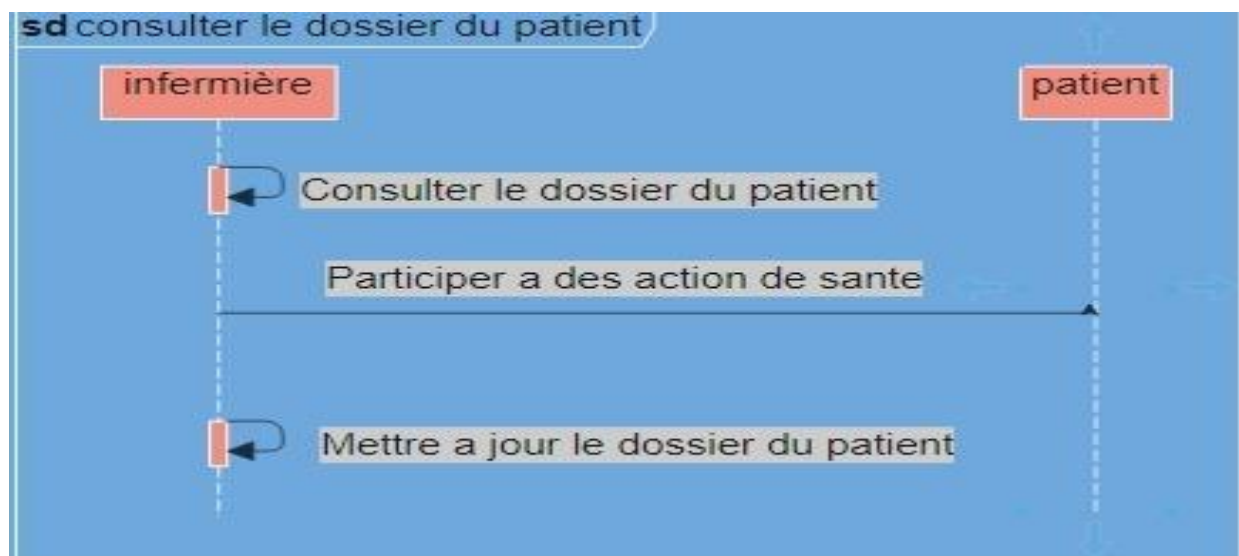


Pour accéder à l'application, l'utilisateur doit tout d'abord s'identifier par son login et son mot de passe via le système qui prend en charge la vérification des champs saisis. S'il est accepté, donc il aura un accès au système et aux applications du menu correspondant. Sinon, il doit vérifier ses données et s'identifier de nouveau.

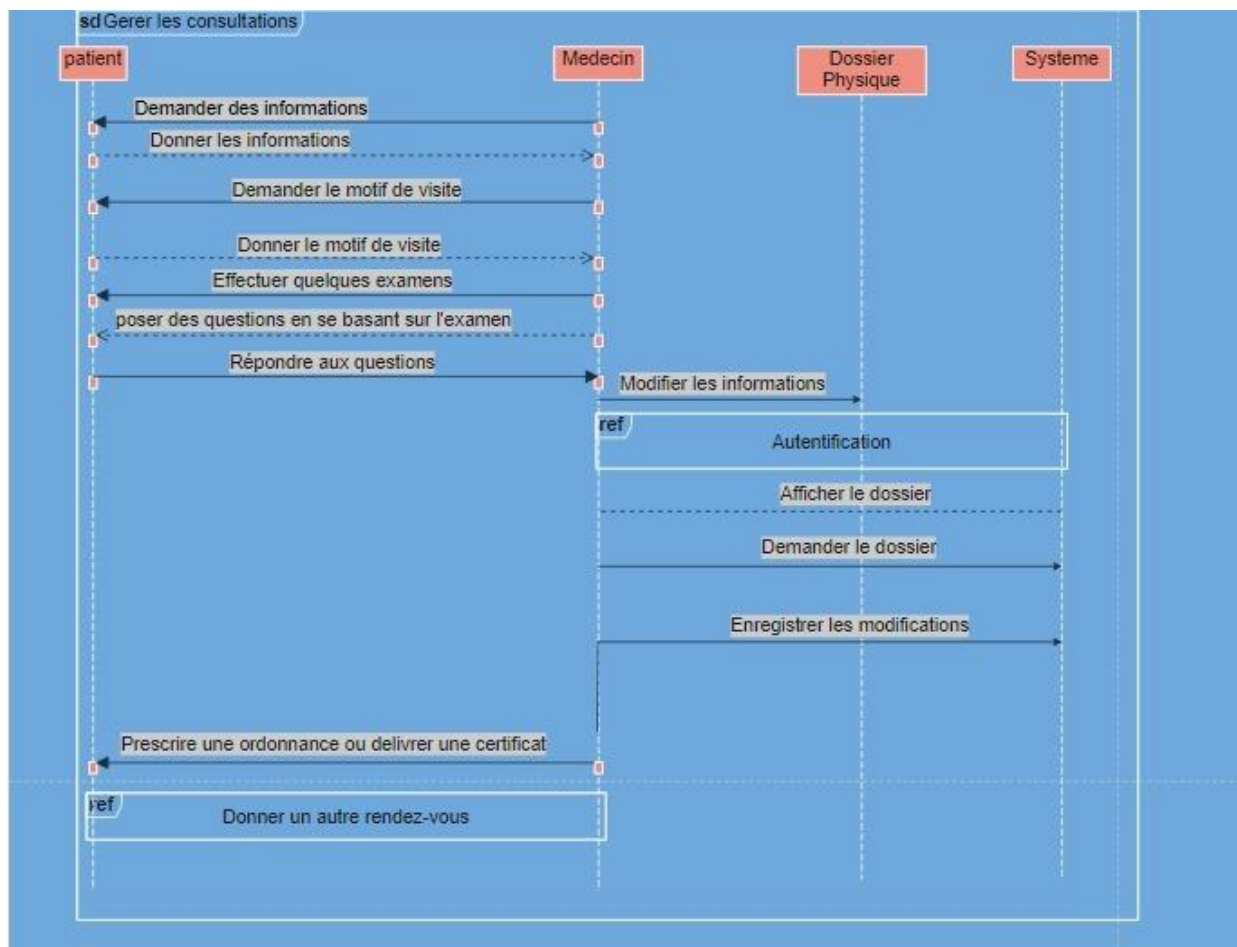
Le diagramme suivant concerne le cas d'utilisation de «Gérer les rendez-vous» :



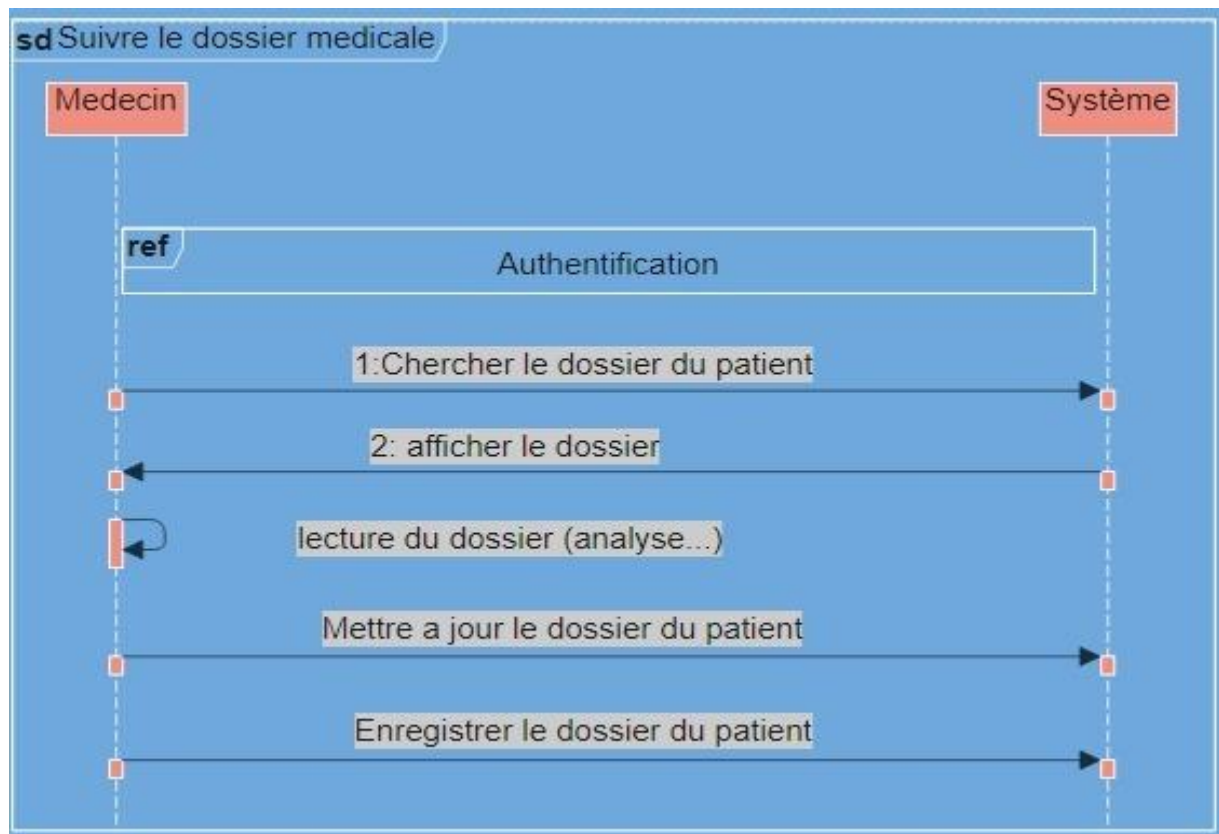
Le diagramme suivant concerne le cas d'utilisation de «Consulter le dossier du patient» :



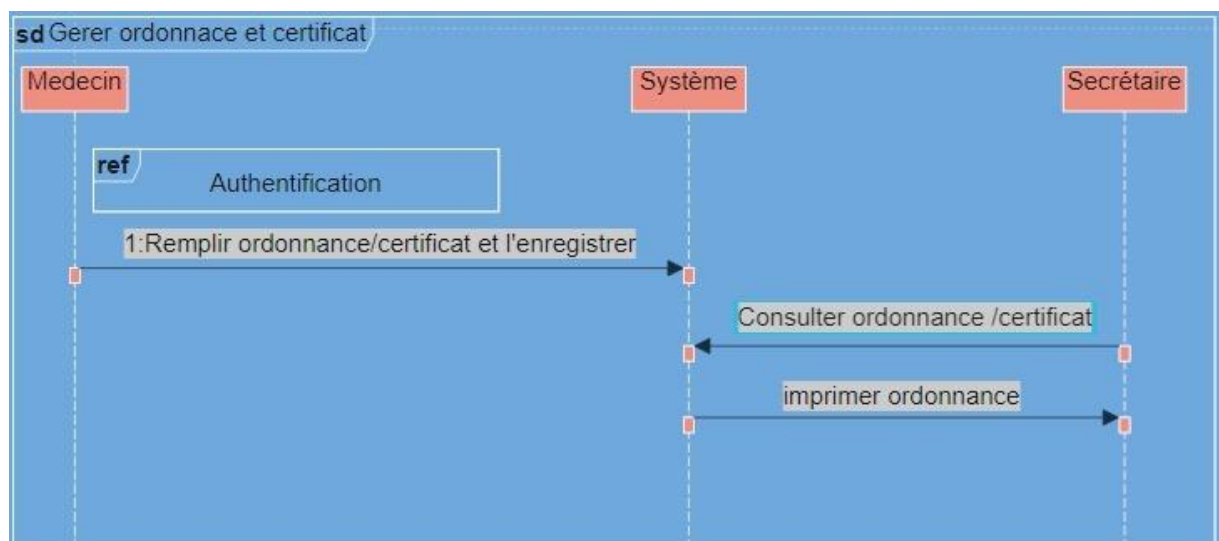
Le diagramme suivant concerne le cas d'utilisation de «Gérer les consultations» :



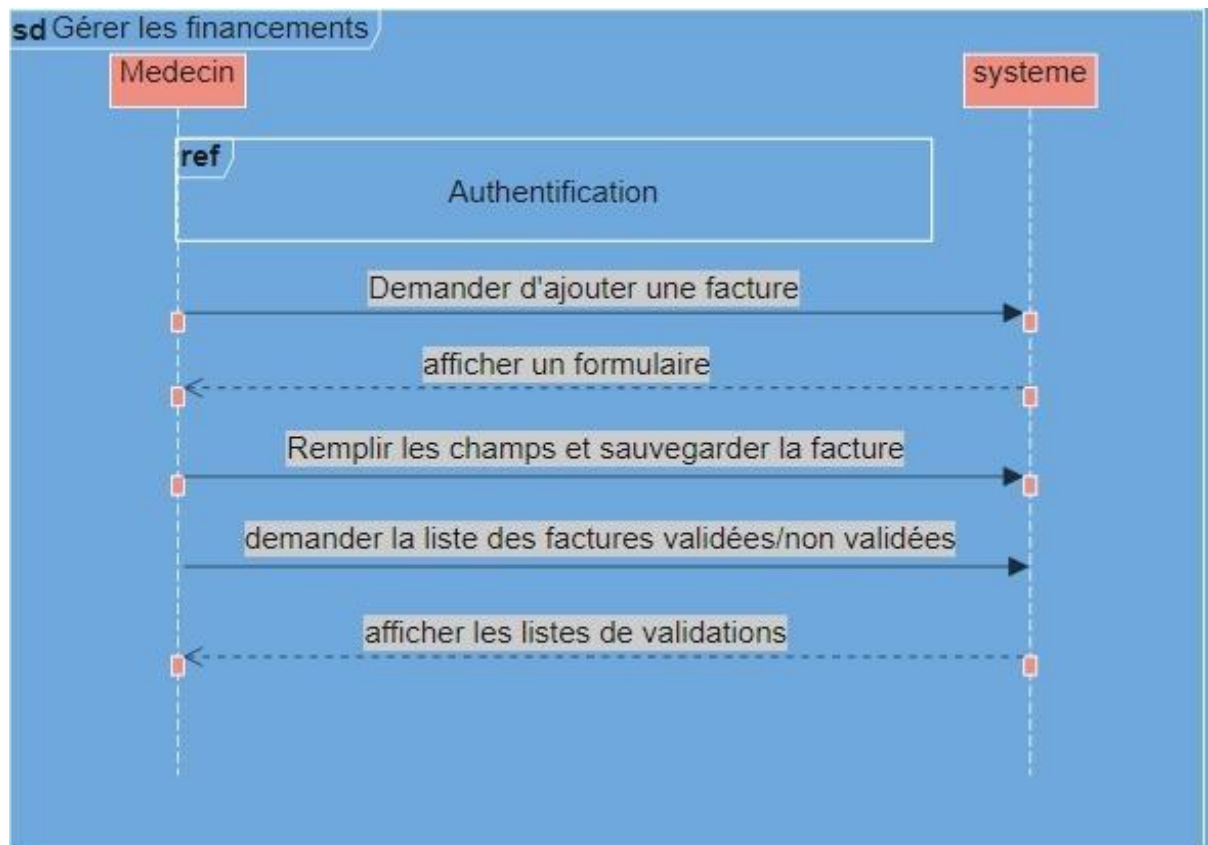
Le diagramme suivant concerne le cas d'utilisation de «Suivre le dossier médical» :



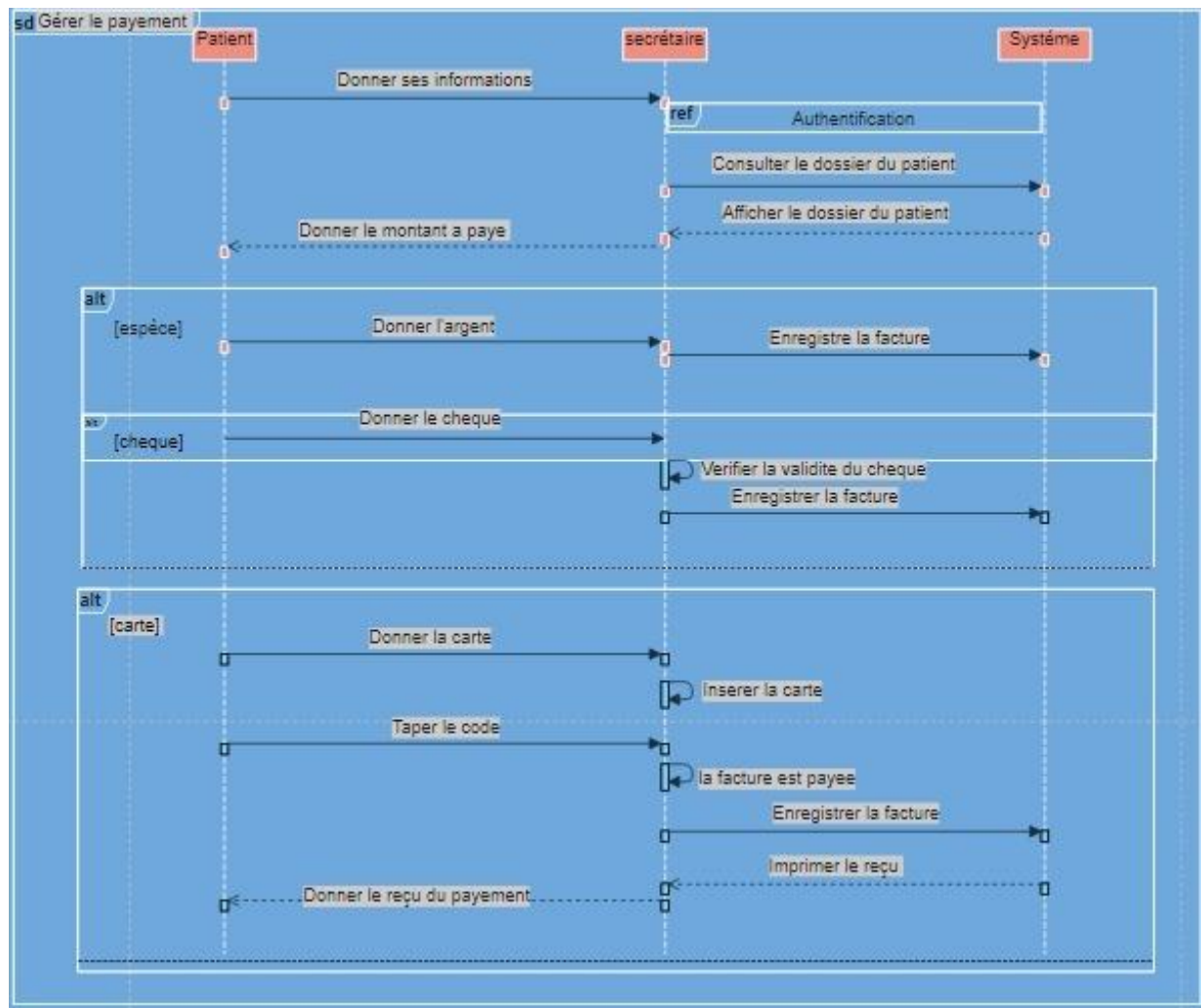
Le diagramme suivant concerne le cas d'utilisation de «Gérer les ordonnances et les certificats» :



Le diagramme suivant concerne le cas d'utilisation de «Gérer les financements» :



Le diagramme suivant concerne le cas d'utilisation de «Gérer le paiement» :

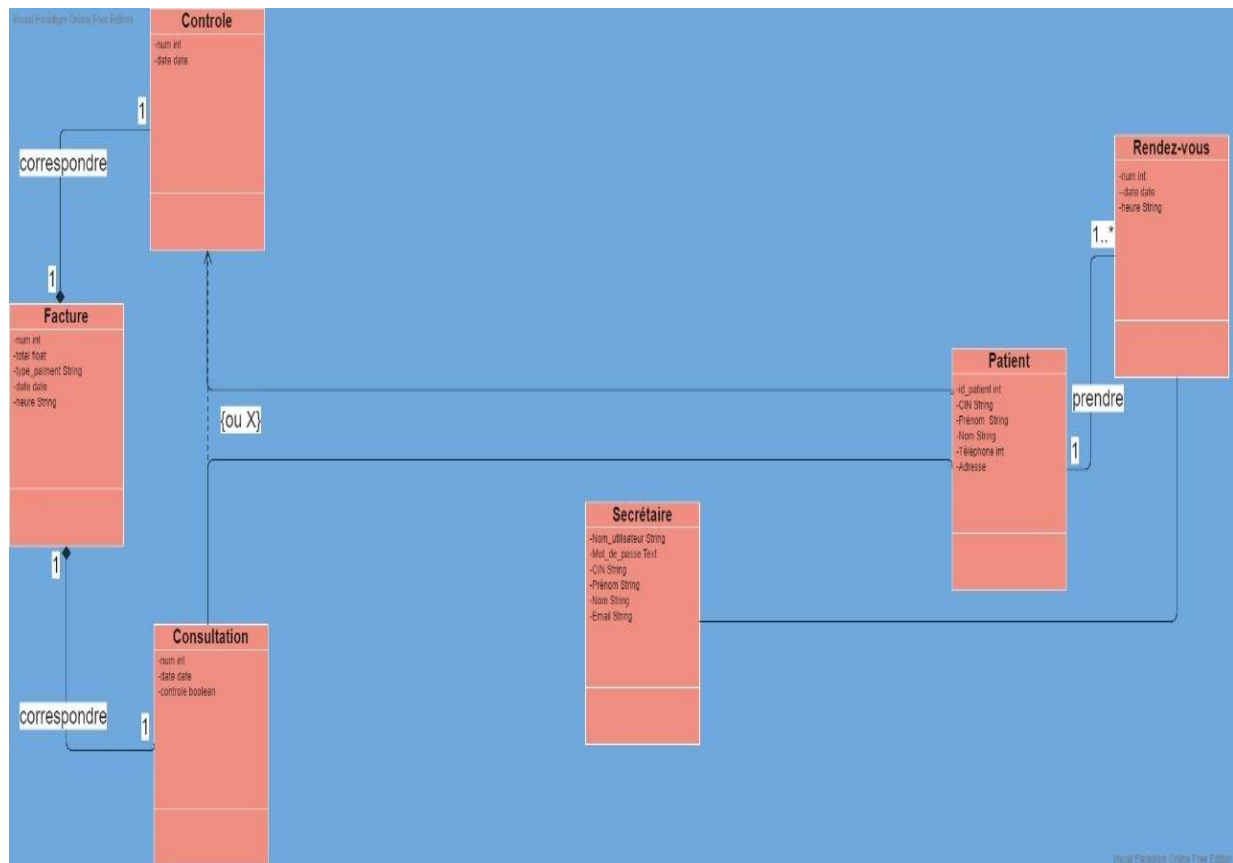


f. Diagramme de classe pour chaque package :

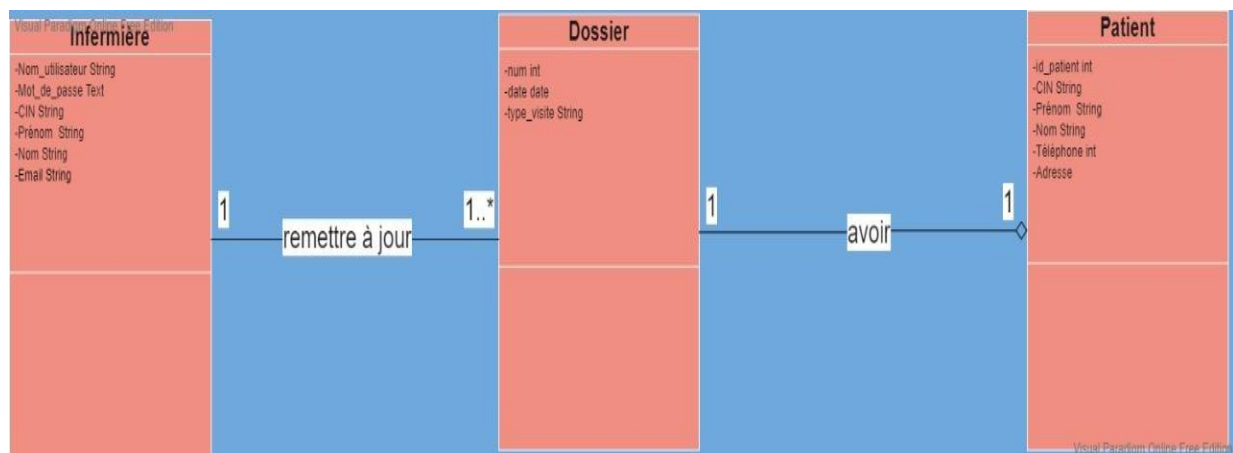
Les diagrammes de classe sont l'un des types de diagrammes UML les plus utiles, car ils décrivent clairement la structure d'un système particulier en modélisant ses classes, ses attributs, ses opérations et les relations entre ses objets. La réalisation de ce diagramme Fournir une description indépendante de l'implémentation des types utilisés dans un système, qui sont ensuite transmis entre ses composants.

Le diagramme de classe pour chaque package réalisé au début c'était un diagramme provisoire, mais ces diagrammes qu'on va faire toute de suite vont nous servir pour créer notre diagramme de classe général dans une étape prochaine :

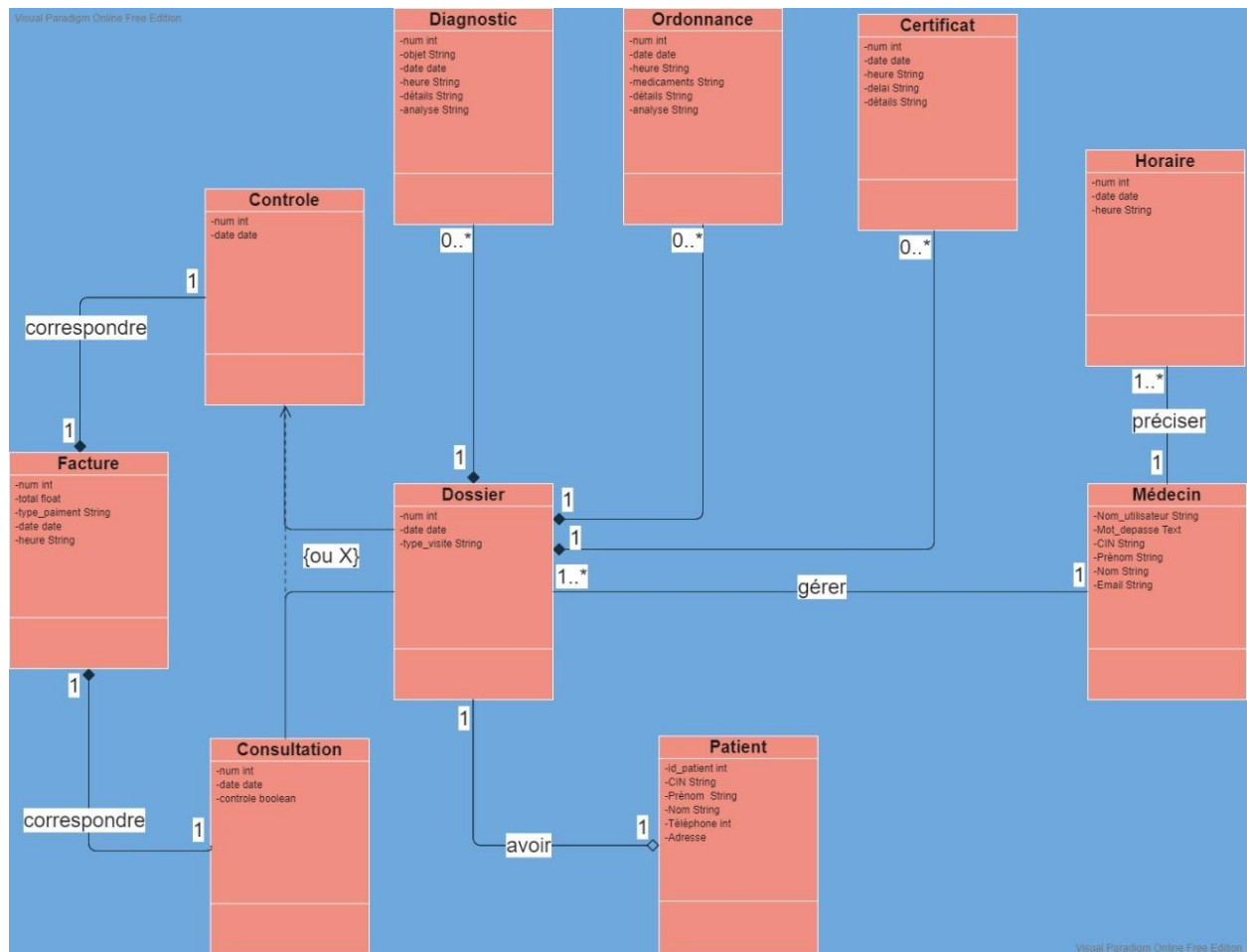
Le diagramme de classe pour le package de gestion secrétariat :



Le diagramme de classe pour le package de gestion d'infirmierie :



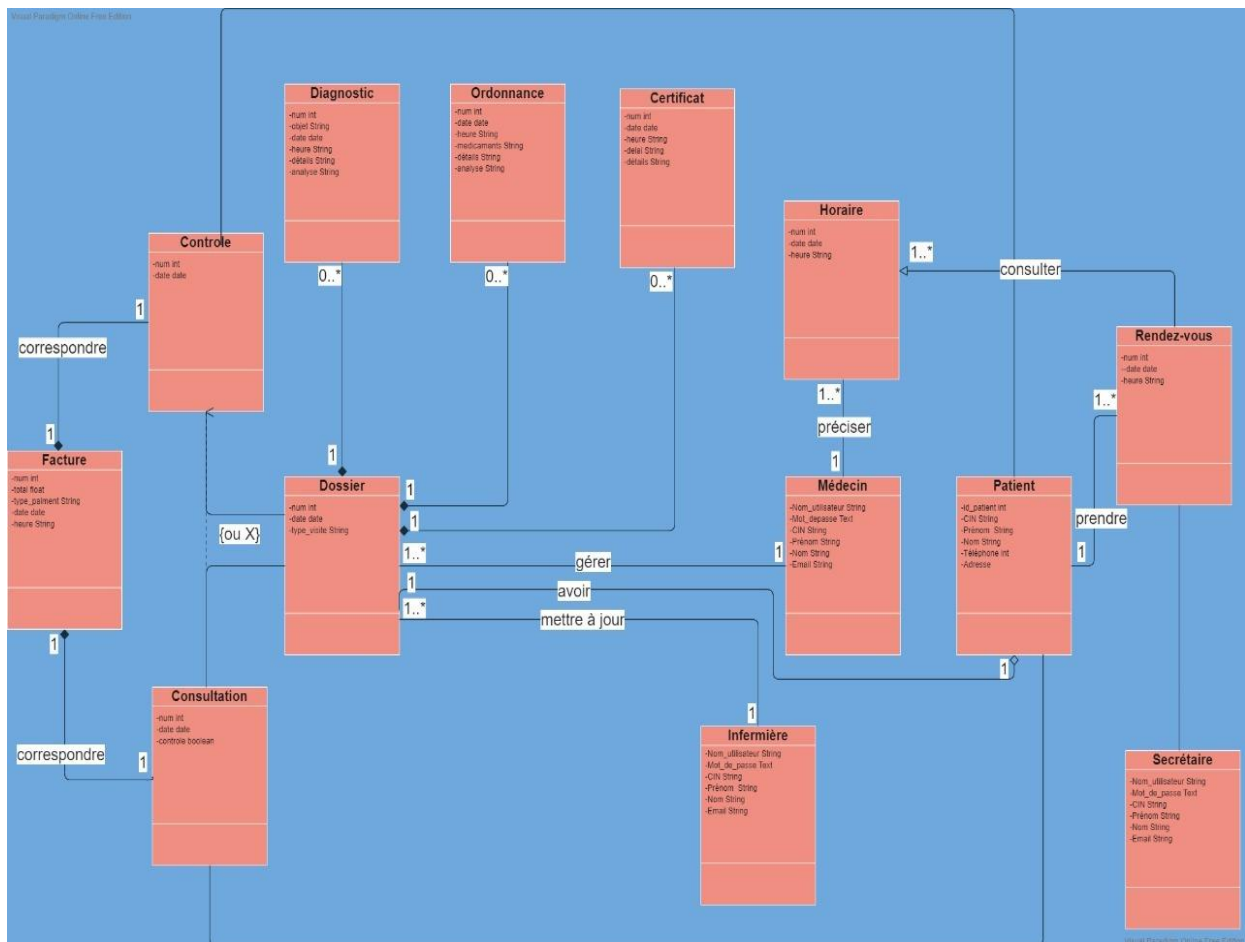
Le diagramme de classe pour le package de gestion du médecin :



g. Diagramme de classe général :

En identifiant les concepts importants du système d'information, on a réalisé un diagramme de classe général pour représenter ces concepts et leurs associations.

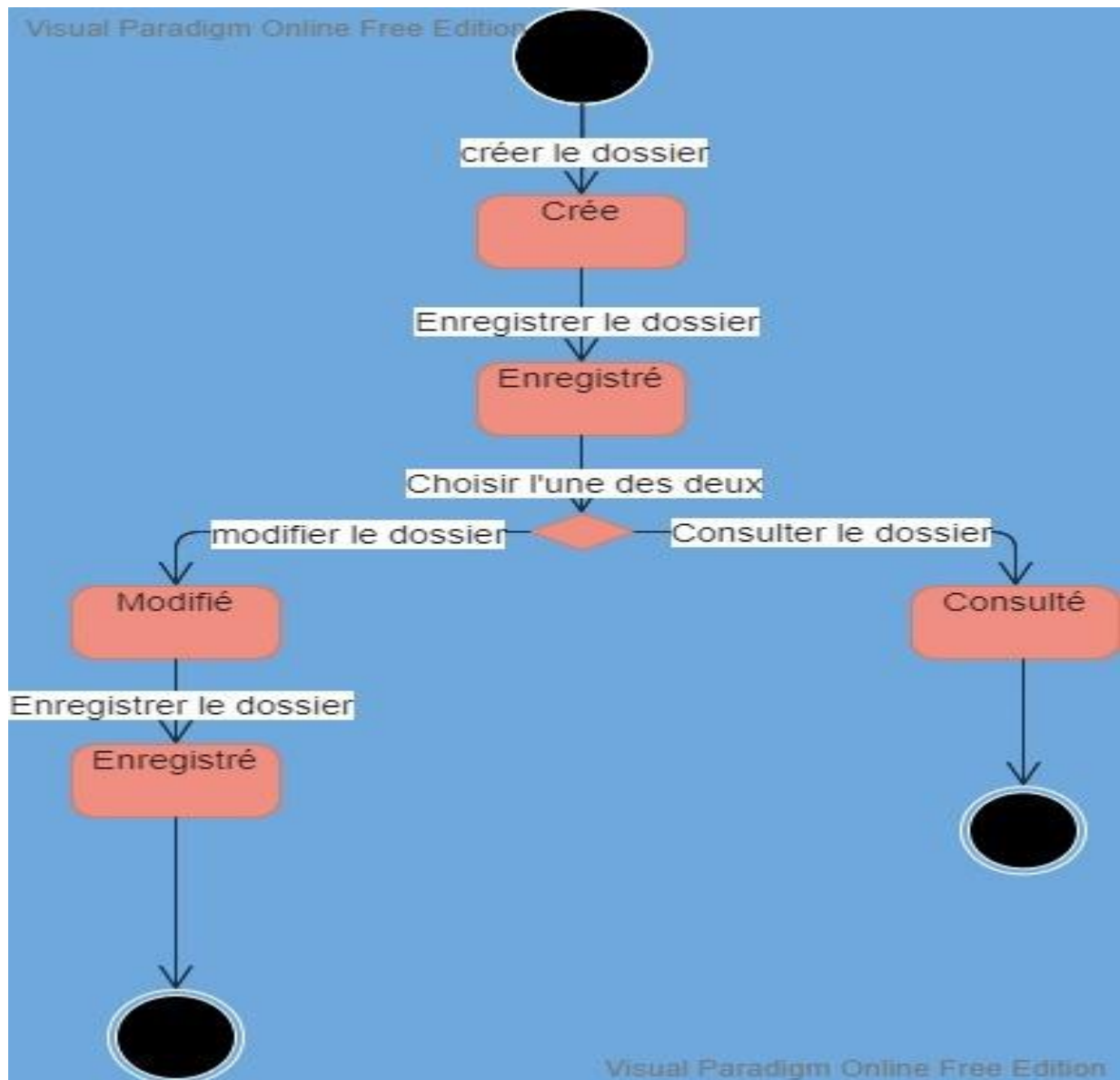
La figure suivante présente notre diagramme de classe générale :



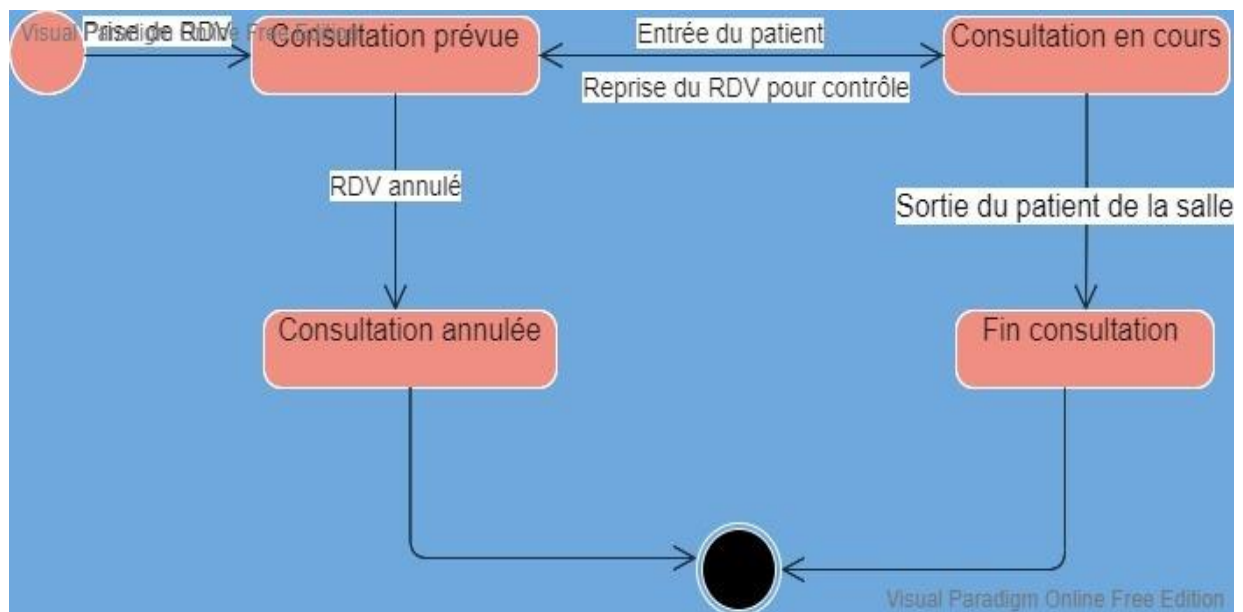
h. Diagramme d'états-transitions :

A partir de notre diagramme de classe général, on a pu extraire les classes dont leurs objets changent d'états pour construire notre diagramme d'états-transitions qui a pour objectif de trouver d'autres fonctions et opérations à travers la recherche des fonctions qui changent d'états pour mieux décrire le système d'information.

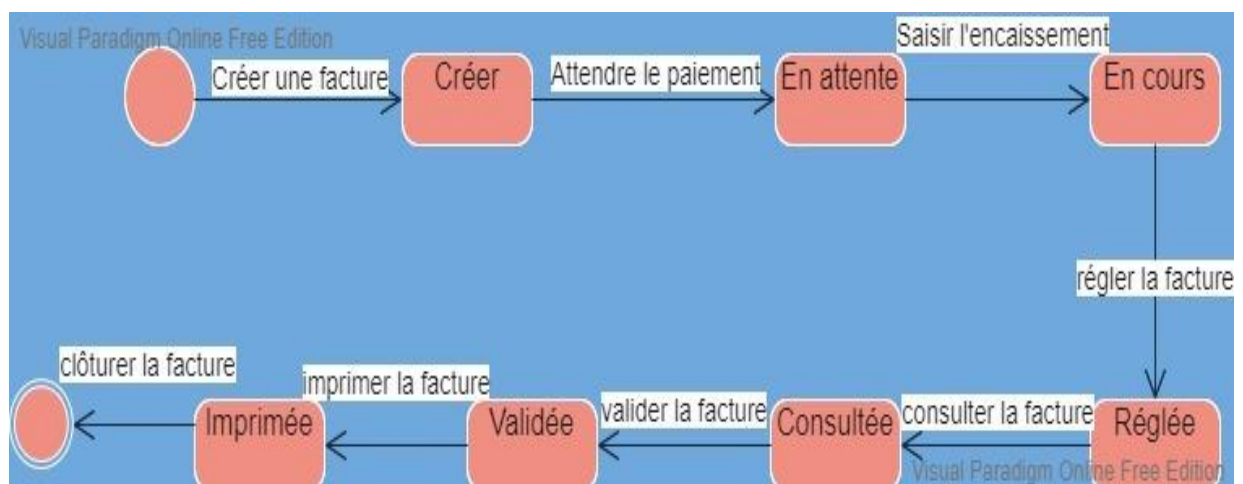
Le diagramme d'états-transitions pour la classe «dossier» :



Le diagramme d'états-transitions pour la classe «consultation» :



Le diagramme d'états-transitions pour la classe «facture» :



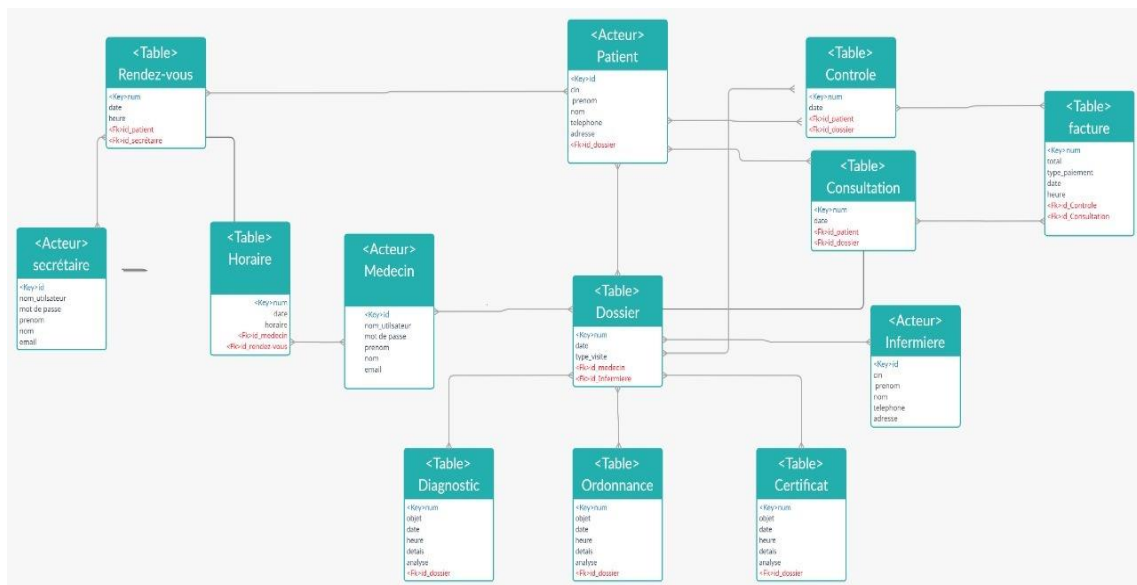
E. Vérification des différents diagrammes :

Dans cette étape de conduite de projet, le but est de revenir à tous les étapes précédentes, plus précisément tous les diagrammes qu'on a construit à partir du diagramme de package et vérifier la compatibilité de ces diagramme en passant d'une étape à une autre.

Comme c'est déjà mentionné au début de chaque sous-titre l'objectif de chaque diagramme et étape, il faut bien constater que deux diagrammes qui font le même travail ne veut pas dire qu'ils permettent de présenter la même vue du projet. Donc c'est assez normal que le progresse dans la conduite de projet de ce sujet va développer plus le travail et va nous offrir plus de fonctions, mais ça ne sort pas la cohérence et la compatibilité entre les différents éléments de cette conduite.

F. Mapping objet-relationnel :

Dans cette étape, on va s'intéresser à faire le Mapping objet/relationnel et générer notre base de données à travers le MLD :



G.Listage des fonctions des différents diagrammes:

Dans ce point-là, on va synthétiser nos résultats dans le tableau suivant qui présente les différentes fonctions trouvées dans les différents diagrammes d'activité, de séquence et d'états-transitions :

Nom de la fonction	Type du diagramme	Titre du diagramme
Authentification	Diagramme d'activité	Package gestion secrétariat/infirmier/médecin
Ajouter un rendez-vous	Diagramme d'activité	Package gestion secrétariat
Gérer un dossier	Diagramme d'activité	Package gestion secrétariat /médecin
Consulter le dossier	Diagramme d'activité	Package gestion infirmier/médecin
Modifier le dossier	Diagramme d'activité	Package gestion infirmier/médecin
Enregistrer les modifications	Diagramme d'activité	Package gestion infirmier/médecin
Demander des analyses	Diagramme d'activité	
Demander un autre rendez-vous	Diagramme d'activité	Package gestion médecin
Demander un certificat	Diagramme d'activité	Package gestion médecin
Noter des traitements	Diagramme d'activité	Package gestion médecin
Consulter les horaires du médecin	Diagramme de séquence	Package gestion secrétariat : Gérer les rendez-vous
Prendre des informations	Diagramme de séquence	Package gestion secrétariat : Gérer les rendez-vous
Mettre à jour le dossier du patient	Diagramme de séquence	Package gestion infirmier : Consulter le dossier du patient
Demander des informations	Diagramme de séquence	Package gestion médecin: Gérer les consultations
Donner des informations	Diagramme de séquence	Package gestion médecin: Gérer les consultations
Prescrire une ordonnance	Diagramme de séquence	Package gestion médecin: Gérer les consultations
Délivrer un certificat	Diagramme de séquence	Package gestion médecin: Gérer les consultations
Chercher le dossier du patient	Diagramme de séquence	Package gestion médecin: Suivre le dossier médical
Lire le dossier du patient	Diagramme de séquence	Package gestion médecin: Suivre le dossier médical
Remplir une ordonnance	Diagramme de séquence	Package gestion médecin: Gérer les ordonnances et les certificats
Remplir un certificat	Diagramme de séquence	Package gestion médecin: Gérer les ordonnances et les certificats

Consulter une ordonnance	Diagramme de séquence	Package gestion médecin: Gérer les ordonnances et les certificats
Consulter un certificat	Diagramme de séquence	Package gestion médecin: Gérer les ordonnances et les certificats
Imprimer ordonnance / certificat	Diagramme de séquence	Package gestion médecin: Gérer les ordonnances et les certificats
Demander d'ajouter une facture	Diagramme de séquence	Package gestion médecin: Gérer les financements
Afficher un formulaire	Diagramme de séquence	Package gestion médecin: Gérer les financements
Demander la liste des factures	Diagramme de séquence	Package gestion médecin: Gérer les financements
Afficher les listes des factures	Diagramme de séquence	Package gestion médecin: Gérer les financements
Enregistrer la facture	Diagramme de séquence	Package gestion secrétariat : Gérer les paiements
Vérifier la validité du chèque	Diagramme de séquence	Package gestion secrétariat : Gérer les paiements
Insérer la carte	Diagramme de séquence	Package gestion secrétariat : Gérer les paiements
Taper le code	Diagramme de séquence	Package gestion secrétariat : Gérer les paiements
Valider le paiement	Diagramme de séquence	Package gestion secrétariat : Gérer les paiements
Imprimer le reçu	Diagramme de séquence	Package gestion secrétariat : Gérer les paiements
Donner le reçu	Diagramme de séquence	Package gestion secrétariat : Gérer les paiements
Annuler le rendez-vous	Diagramme d'états-transitions	Consultation
Attendre le paiement	Diagramme d'états-transitions	Facture
Saisir l'encaissement	Diagramme d'états-transitions	Facture
Régler la facture	Diagramme d'états-transitions	Facture
Clôturer la facture	Diagramme d'états-transitions	Facture

Conclusion :

Ce projet fût très intéressant à réaliser. Il nous a permis de bien comprendre la démarche à suivre pour la réalisation de futurs grands projets. Il semble très efficace de se tenir à cette méthode car elle prend en compte la plupart des problématiques à résoudre. Nous avons vu que UML permet également une étude détaillée grâce par exemple aux diagrammes d'états-transitions et de cas d'utilisation qui sont très bénéfiques à la compréhension du logiciel.

Le langage UML est de plus très visuel, ce qui est bien sûr un à tout lors de la conception de l'application. Ce travail nous a beaucoup intéressés car il nous a fait découvrir une approche en matière d'analyse de projet. Il nous a permis de nous familiariser avec langage de modélisation UML et de mieux maîtriser ce concept. Il a aussi favorisé le travail en équipe comme on pourra le rencontrer au sein d'une entreprise.