



UNIVERSITÉ POLITEHNIQUE DE BUCHAREST

<< Faculté d'Ingénierie En Langues Étrangères (FILS) >>

Équipe : Bankoue Tankoua Christian & Fohom Yann Cedric

GROUPE : 1230F

Titre : Java-Health (système médical innovant)

1. ENONCE DU PROBLEME

1.1. Identification des acteurs

1.2. Opérations (fonctionnalités) du système
--

1.3. Règles métier (business rules)

2. Matrice RASIC

3. Diagrammes UML

3.1. Diagramme de cas d'utilisation, description détaillée de chaque cas d'utilisation (tableau)
--

3.2. Diagrammes d'activités (au moins 3)
--

3.3. Diagrammes de séquence (au moins 3)
--

3.4. Diagramme de paquets

4. Description du modèle architecturale

4.1. Identification des composantes

4.2. Diagramme de classes et – si nécessaire – diagramme de composantes

4.3. Design patterns – au moins 2

5. Diagramme (schéma) de la base de données

6. Description des tests unitaires

1. Énoncé du problème

Le Ministère de la Santé, en collaboration avec les hôpitaux, les cabinets et les cliniques publiques ou privées, a décidé que la Roumaine a besoin d'un nouveau système de e-health, plus innovant et flexible. Ils ont demandé à votre compagnie de développer un système logiciel pour gérer les processus métier associés. Les demandes d'un tel système sont les suivantes:

- La plateforme a plusieurs types d'utilisateurs : personnel médical (médecins, assistants, infirmières), patients, pharmaciens, fournisseurs (médicaments, appareils médicaux, équipements), administrateurs.
- Pour avoir accès à l'application, chaque utilisateur doit premièrement s'enregistrer en ligne, cela signifie créer un compte en fournissant ses informations personnelles (incluant un code unique – par exemple un code unique pour chaque membre du personnel médical / le CNP pour les patients), adresse d'email, mot-de-passe.
- Chaque patient peut faire une programmation pour une consultation, en choisissant le domaine médical correspondant (contrôle ophtalmologique, ORL, contrôle de routine etc.), le jour et l'heure de la visite. Le système va lui afficher les cabinets / hôpitaux et les médecins disponibles.
- Chaque médecin peut ajouter des consultations dans le système, incluant les informations du patient, la date de la visite, les maladies, le traitement offert. Les médecins et les assistants peuvent aussi libérer des recettes contenant les médicaments indiqués au patient. Le pharmacien va recevoir la recette (en format image) et donne les médicaments indiqués (ou des substituts) au patient.
- Les fournisseurs maintiennent des catalogues mises à jour avec les informations de leurs produits (description, photos, prix), qui peuvent être accédés soit par le personnel médical, soit par les deux, personnel médical et patients.
- Chaque administrateur a le droit d'effacer des utilisateurs inactifs ou des produits médicaux qui n'ont pas toutes les informations essentielles mises à jour.

1.1. Identification des acteurs

Les acteurs du système sont les patients, les pharmaciens, les fournisseurs, le personnel médical et les administrateurs.

1.2. Opérations (fonctionnalités) du système

Les opérations permises par la plate-forme sont les suivantes :

- Création d'un compte
- Programmation de consultation
- Choix du domaine médical
- Mise à jour du dossier médical du patient
- Suppression des utilisateurs

1.3. Règles métier (business rules)

[RM1] La plateforme a plusieurs types d'utilisateurs.

[RM2] La plate-forme peut être utilisée seulement par les personnes enregistrées.

[RM3] Seuls les assistants et les médecins peuvent faire un diagnostic et envoyer une ordonnance (recette).

[RM4] Les administrateurs peuvent effacer un compte d'utilisateur si l'utilisateur n'a pas été actif les 2 derniers mois.

[RM5] Un médecin ne peut ajouter une consultation que si le patient est dans son domaine.

[RM6] Les résultats des mesures des signes vitaux du patient sont inscrits dans son dossier médical contenant toutes les données personnelles et médicales de celui-ci.

[RM7] Le patient peut choisir le domaine médical selon les maux qu'il rencontre.

[RM8] Les fournisseurs maintiennent des catalogues mises à jour avec les informations de leurs produits (description, photos, prix), qui peuvent être accédés soit par le personnel médical, soit par les deux, personnel médical et patients.

2. Matrice RASIC

Nous aurons 2 **matrices RASIC** dans notre projet à savoir :

a. Membres de l'équipe + Educateurs

Matrice RASIC	Educateurs	Yann	Christian
Identification des acteurs	C	A	R
Fonctionnalités du système	C	R	A
Business Rules	C	R	A
Règles métier	C	A	R
Cas d'utilisation	C	A	R
Modèle architecturale	C	R	A
Schéma de la base de données	C	A	R
L'implémentation	C	R	A

b. Fonctionnalités et acteurs

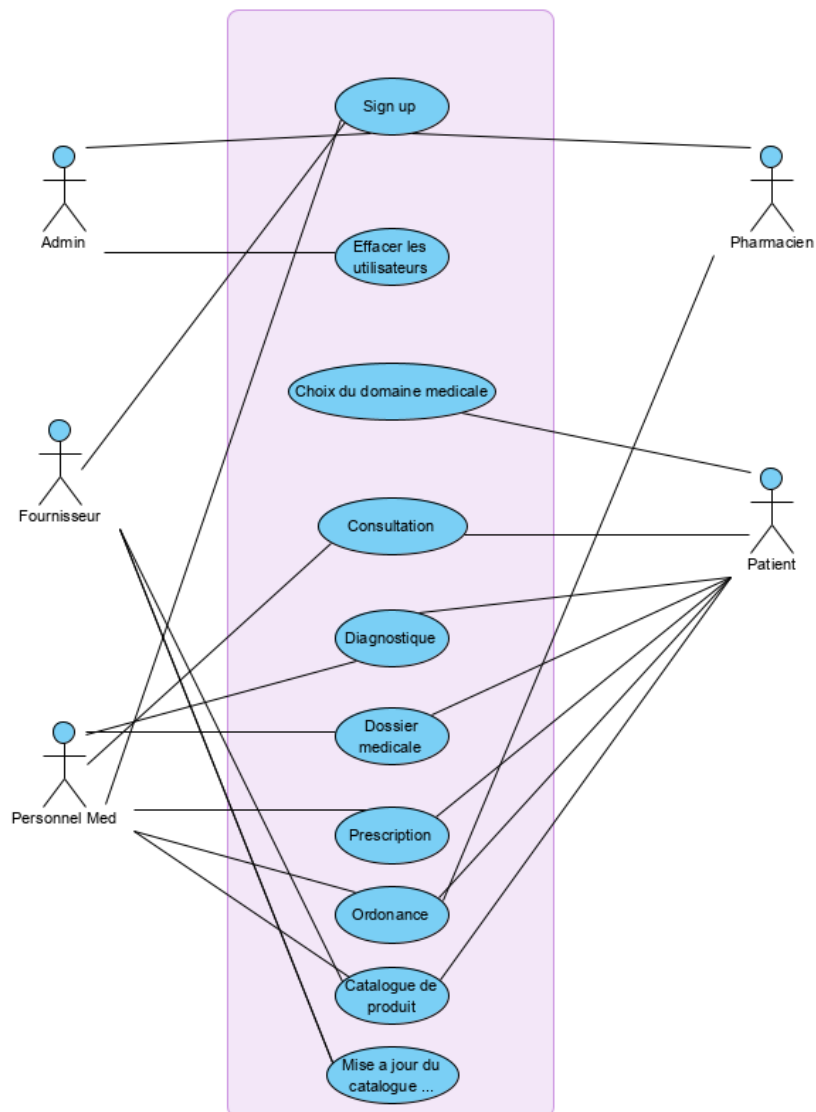
Matrice RASIC	Admn	Manager	Médecins	Assistants	Nurse	Patients	Pharmaciens	Fournisseurs
Inscription (Création d'un compte)	C	A	C	C	C	C	C	C
Programmation de consultation	I	A	I			R		
Choix du domaine médicale	I	A	I			R		
Mise à jour du dossier médicale du patient	S	A	R	S	I			
Ordonnance (Recettes)	I	A	R	C	I	I	I	
Transmission Des produits aux patients	I	A	I	I	I		R	
Modification des utilisateurs	R	A						
Catalogue de produits (médicaments, appareils médicaux, équipements)	A	A	I	I	I	I	I	R
Suppression D'utilisateurs	R	A						

3. Diagrammes UML

3.1. Diagramme de cas d'utilisation et description détaillée de chaque cas d'utilisation (tableau)

3.2.

a. Diagramme de cas d'utilisation



b. Description détaillée de chaque cas d'utilisation (tableau)

[UC1] Création compte	
Acteurs	Patient / Pharmacien / Fournisseur / Personnel médical / Administrateur
Règles métiers	[RM1] / [RM2]
Sommaire	Ce cas d'utilisation décrit la création du compte de tout utilisateur de la plateforme
Précondition	<p>L'application doit contenir toutes les informations personnelles du patient C'est-à-dire que Le patient doit avoir un compte lie à une adresse email ou un réseau social (Facebook, Twitter, WhatsApp).</p> <p>De plus l'application doit contenir un nombre de comptes plus petit que la capacité maximale de l'application.</p>
Post-condition	Un nouveau compte est ajouté dans la base de données
Pas	<ol style="list-style-type: none"> 1. Sélectionner l'option « Créez un compte ». 2. Le système affiche la fenêtre correspondante a cette option. 3. L'utilisateur introduit : un nom d'utilisateur, une adresse de courriel, un mot de passe et la confirmation du mot de passe et le code captcha pour s'assurer que c'est bien l'œuvre d'un humain. 4. L'utilisateur envoie le formulaire pour la vérification. 5. Le système vérifie les données introduites. 6. Le système affiche le message « compte crée avec succès » et « la page d'accueil du profile de l'utilisateur ».
Exceptions	<p>Au pas 5, si les données introduites sont erronées, le système affiche un message d'erreur et renvoi automatiquement l'utilisateur au pas 3.</p> <ol style="list-style-type: none"> 5.a. S'il y a un champ vide. 5.b. Si l'adresse d'email n'existe pas / n'est pas valide. 5.c. Si un utilisateur avec le même email / même compte social existe déjà dans la bd. 5.d. Si le mot de passe ne respecte pas les demandes min. 5.e. Si les deux champs du mot de passe contiennent des infos différentes. 5.f. Le mot de passe introduit n'est pas valide – trop peu de caractères ou des caractères non-acceptés. 5.g. Le nom d'utilisateur est déjà associé à un autre compte.
[Croquis interface]	Annexe 1. Création de compte.



Annexe 1. Création compte

[UC2] Effacer des Utilisateurs	
Acteurs	Administrateur
Règles métiers	[RM4]
Sommaire	Ce cas d'utilisation décrit le processus de suppression de compte
Précondition	Un utilisateur est connecté en tant qu'administrateur de la plateforme
Post-condition	Le compte sélectionné est effacé de la base de données
Pas	<ol style="list-style-type: none"> 1. Sélectionner l'option « Effacer Compte » 2. Le système affiche une fenêtre de validation 3. Le système efface le compte de la base de données 4. Le système affiche le message « Compte supprimé avec succès » 5. Le système envoie un courriel à l'utilisateur qui détenait le Compte pour lui informer de sa suppression
Exceptions	Au pas 3, s'il y a une erreur pendant suppression du Compte, le système affiche un message d'erreur et on revient au pas 1.

[UC3] Choix du domaine médical	
Acteurs	Patient
Règles métiers	[RM7]
Sommaire	Ce cas d'utilisation décrit le choix du domaine médical du client sur la plateforme
Précondition	<p>L'application doit contenir une description brève de la maladie du patient C'est-à-dire que Le patient doit fournir des détails concernant sa maladie afin que le système lui donne la possibilité de choisir un domaine médical.</p> <p>De plus l'application doit contenir un nombre de domaine médical plus petit que la capacité maximale de l'application.</p>
Postcondition	Un nouveau domaine médical est ajouté dans la base de données
Pas	<ol style="list-style-type: none"> 1. Sélectionner l'option « choix du domaine médical ». 2. Le système affiche la fenêtre correspondante a cette option. 3. L'utilisateur introduit : un code captcha pour s'assurer que c'est bien l'œuvre d'un humain et fournie une description de sa maladie. 4. L'utilisateur envoie le formulaire pour la vérification. 5. Le système vérifie les données introduites. 6. Le système affiche le message « domaine médical valide avec succès ».
Exceptions	<p>Au pas 5, si les données introduites sont erronées, le système affiche un message d'erreur et renvoi automatiquement l'utilisateur au pas 3.</p> <ol style="list-style-type: none"> 5.a. S'il y a un champ vide. 5.b. Si le code captcha n'est pas valide. 5.c. Si le patient insère des symptômes d'une autre domaine différent du domaine qu'il a choisi. 5.d. Si les symptômes ne sont pas répertoriés dans notre système,

[UC4] Consultation	
Acteurs	Patient / personnel médical
Règles métiers	[RM5]
Sommaire	Ce cas d'utilisation décrit la programmation d'une consultation sur la plateforme
Précondition	L'application doit contenir un agenda détaillé du personnel médical. C'est-à-dire que Le médecin ne peut ajouter une consultation que si le patient est dans son domaine, et que ce dernier accepte la date proposer et l'horaire proposés. De plus l'application doit contenir un nombre limite de consultation plus petit que la capacité maximale de l'application.
Postcondition	Une nouvelle consultation est ajoutée dans la base de données
Pas	<ol style="list-style-type: none"> 1. Sélectionner l'option « consultation ». 2. Le système affiche la fenêtre correspondante a cette option. 3. Le patient sélectionne la date qui le convient selon l'agenda fourni par la plateforme qui présente bien entendu les disponibilités du personnel médical. 4. L'utilisateur envoie le formulaire pour la vérification. 5. Le système vérifie les données introduites. 6. Le système affiche le message « consultation programmée ».
Exceptions	<p>Au pas 5, si les données introduites sont erronées, le système affiche un message d'erreur et renvoi automatiquement l'utilisateur au pas 3.</p> <ol style="list-style-type: none"> 5.a. S'il y a un champ vide. 5.b. Si la date sélectionnée n'est pas valide. 5.c. Si le patient sélectionne une date qui n'est pas répertorié dans l'agenda 5.d. Si le patient a sélectionné une date déjà occupée.

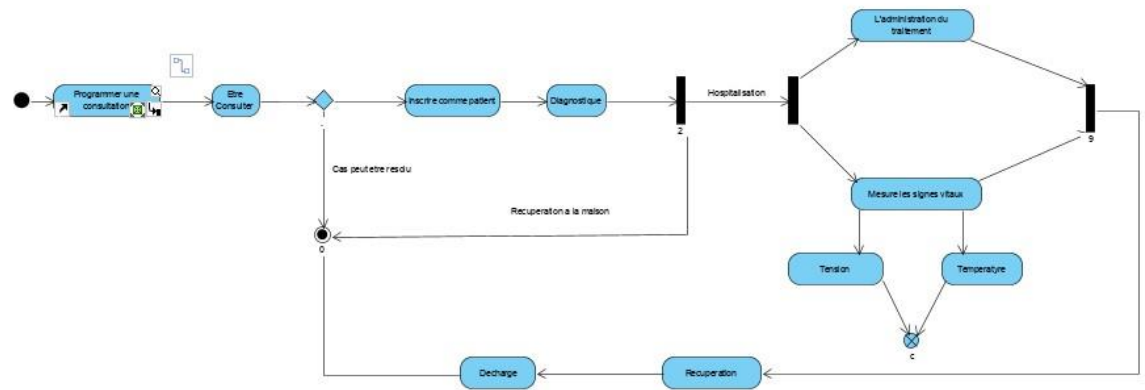
[UC5] Diagnostique	
Acteurs	Personnel médical / Patient
Règles métiers	[RM3]
Sommaire	Ce cas d'utilisation décrit le diagnostic de la situation sur la plateforme
Précondition	L'application doit contenir toutes les informations requises. C'est-à-dire que Le patient doit fournir des détails concernant sa maladie et le personnel médical doit fournir un feedback et établir un rapport.
Postcondition	Un nouveau diagnostique est ajouté à la base de données
Pas	<ol style="list-style-type: none"> 1. Sélectionner l'option « Diagnostique ». 2. Le système affiche la fenêtre correspondante a cette option. 3. L'utilisateur introduit : un code captcha pour s'assurer que c'est bien l'œuvre d'un humain et tous les détails nécessaires. 4. L'utilisateur envoie le formulaire pour la vérification. 5. Le système vérifie les données introduites. 6. Le système affiche le message « Diagnostique établi avec succès ».
Exceptions	<p>Au pas 5, si les données introduites sont erronées, le système affiche un message d'erreur et renvoi automatiquement l'utilisateur au pas 3.</p> <ol style="list-style-type: none"> 5.a. S'il y a un champ vide. 5.b. Si le code captcha n'est pas valide. 5.c. Si les informations mentionnées ne sont pas valides

[UC6] Prescription/Ordonnance	
Acteurs	Personnel médical
Règles métiers	[RM3]
Sommaire	Ce cas d'utilisation décrit la prescription du traitement au patient sur une ordonnance virtuelle.
Précondition	L'application doit contenir un document détaillé du personnel médical avec toutes recommandations faites aux patients. C'est-à-dire que le personnel médical doit établir une ordonnance comportant toutes les indications nécessaires pour aider le patient à bien suivre son traitement.
Postcondition	Une nouvelle Ordonnance est ajoutée dans la base de données
Pas	<ol style="list-style-type: none"> 1. Sélectionner l'option « Prescription/Ordonnance ». 2. Le système affiche la fenêtre correspondante a cette option. 3. Le personnel médical doit établir.la prescription et les informations nécessaires pour le patient. 4. Le personnel médical envoie le formulaire pour la vérification. 5. Le système vérifie les données introduites. 6. Le système affiche le message « Prescription/Ordonnance crée ».
Exceptions	<p>Au pas 5, si les données introduites sont erronées, le système affiche un message d'erreur et renvoi automatiquement l'utilisateur au pas 3.</p> <ol style="list-style-type: none"> 5.a. Si la prescription ne correspond pas a la maladie du patient. 5.b. Si l'ordonnance n'est pas établie et signée par un membre du personnel médical.

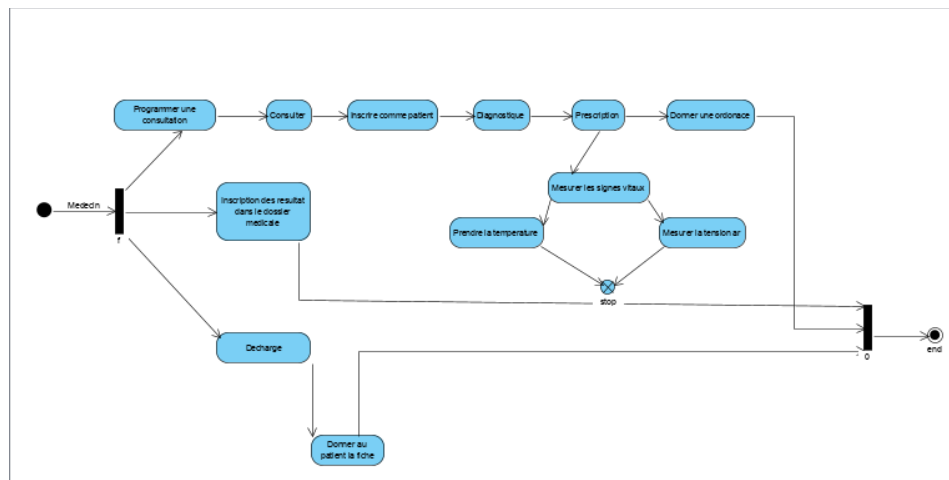
[UC7] Catalogue de produits	
Acteurs	Les fournisseurs
Règles métiers	[RM7]
Sommaire	Ce cas d'utilisation présente la maintenance du catalogue des produits médicaux par les fournisseurs
Précondition	Un fournisseur doit être au préalable avoir un compte sur la plateforme
Postcondition	Les informations des produits (description, photos, prix) sont mises à jour
Pas	<ol style="list-style-type: none"> 1. Sélectionner l'option « Catalogue de produits » 2. Le système affiche une fenêtre de validation 3. Le système démarre la mise à jour dans la base de données 4. Le système affiche le message « Mise à jour du catalogue médical avec succès » 5. Le système envoie un courriel aux autres utilisateurs pour les informer de la disponibilité des produits.
Exceptions	Au pas 3, s'il y a une erreur pendant la mise à jour, le système affiche un message d'erreur et on revient au pas 1.

3.3. Diagrammes d'activités (au moins 3)

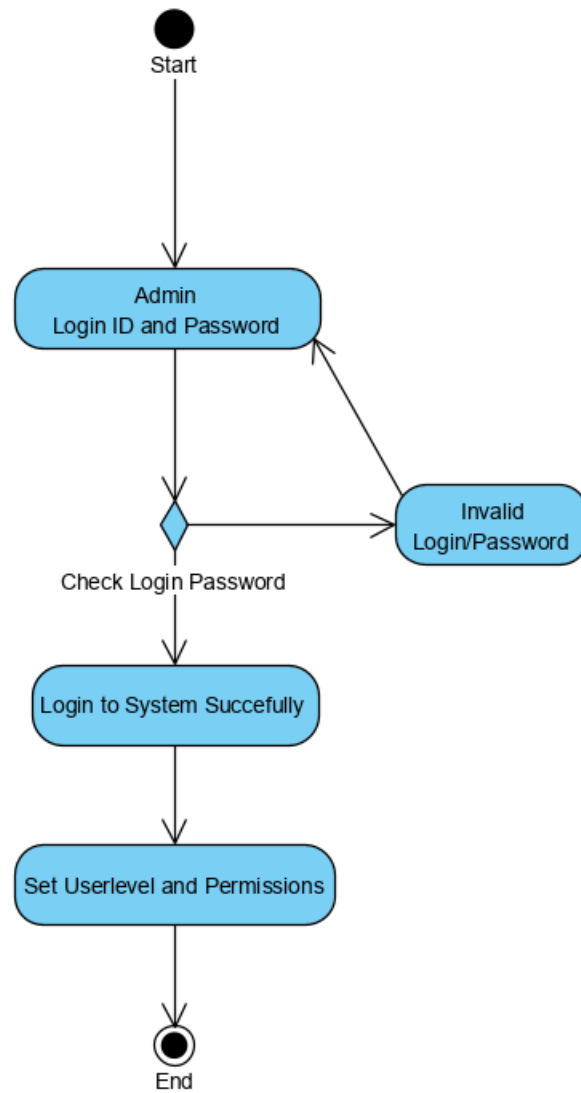
a. Patient



b. Personnel Medical

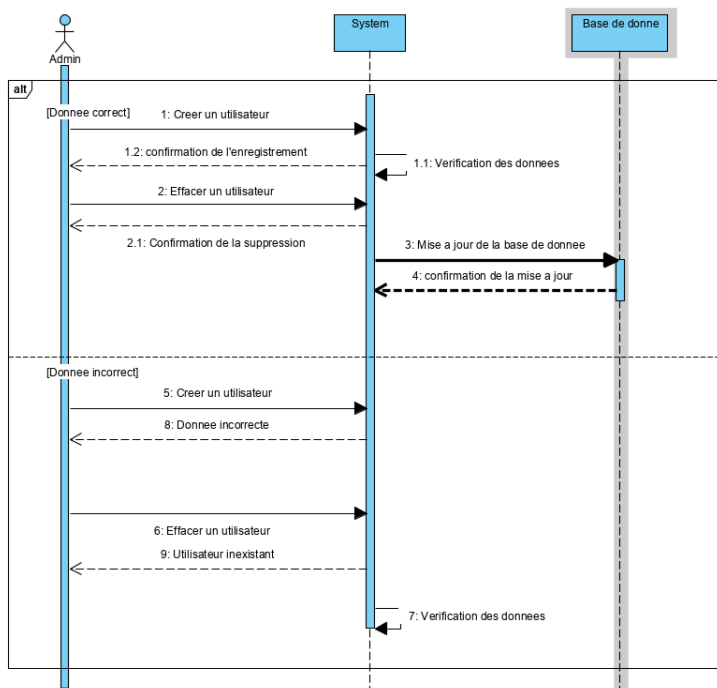


c. Connexion de l'administrateur

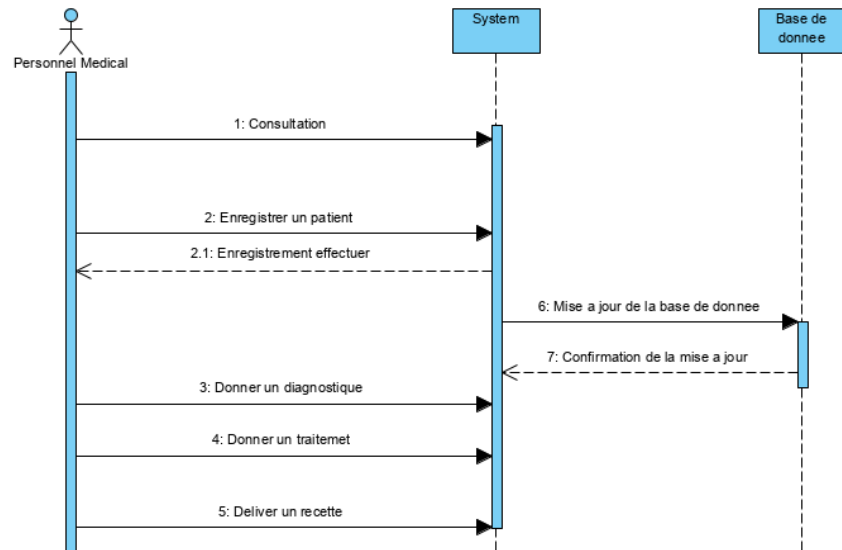


3.4. Diagrammes de séquence (au moins 3)

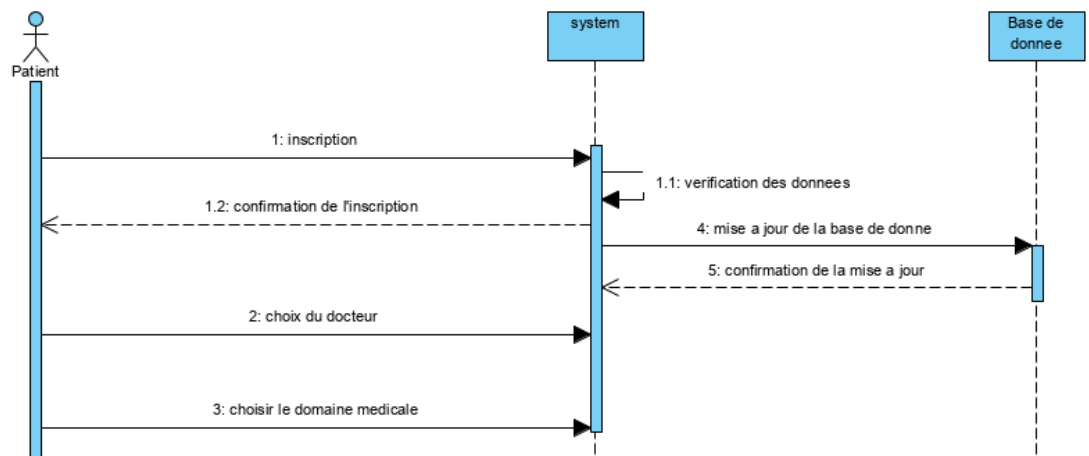
a. Administrateur



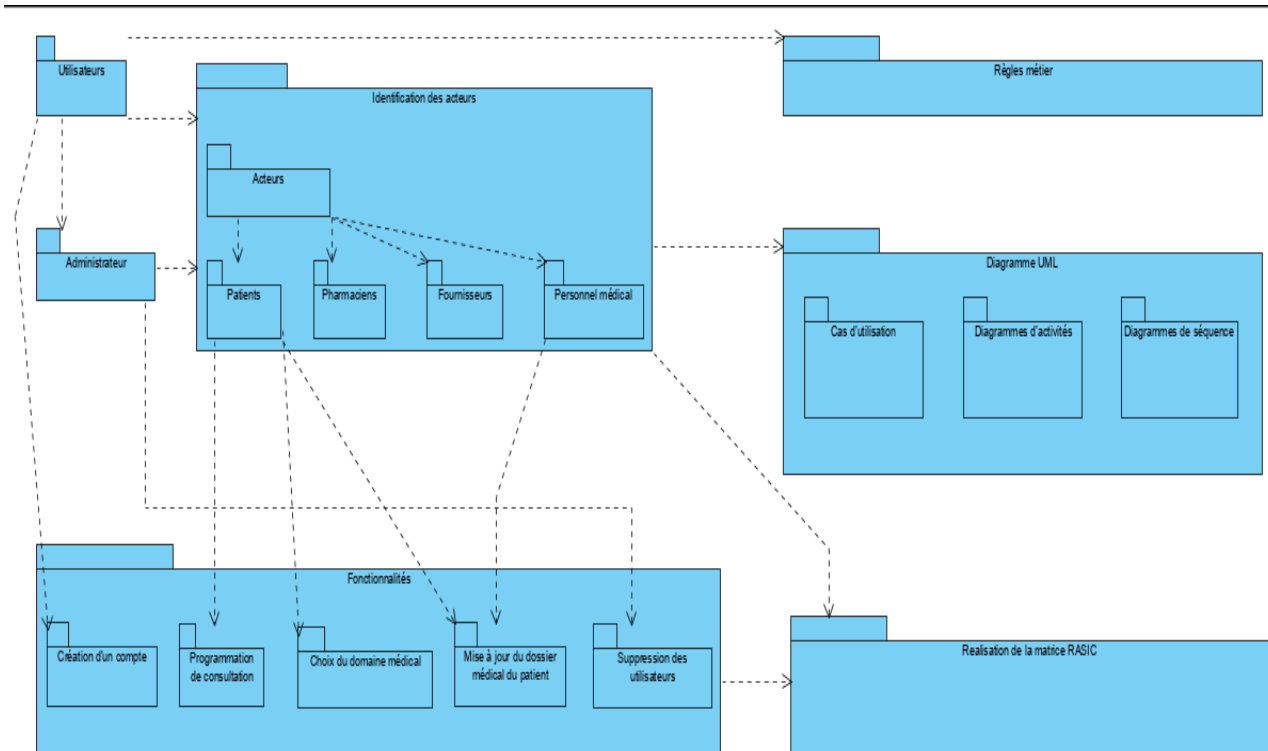
b. Personnel médical



c. Patient



3.5. Diagramme de paquets



4. Description du modèle architecturale

Dans notre Project, nous avons choisi le modèle architectural **3-tier**. Ce modèle est approprié parce qu'il reflète la séparation des couches existante dans notre projet.

De plus, il a l'avantage qu'un peut modifier un des couches sans interférer avec les autres. La représentation des couches est la suivante :

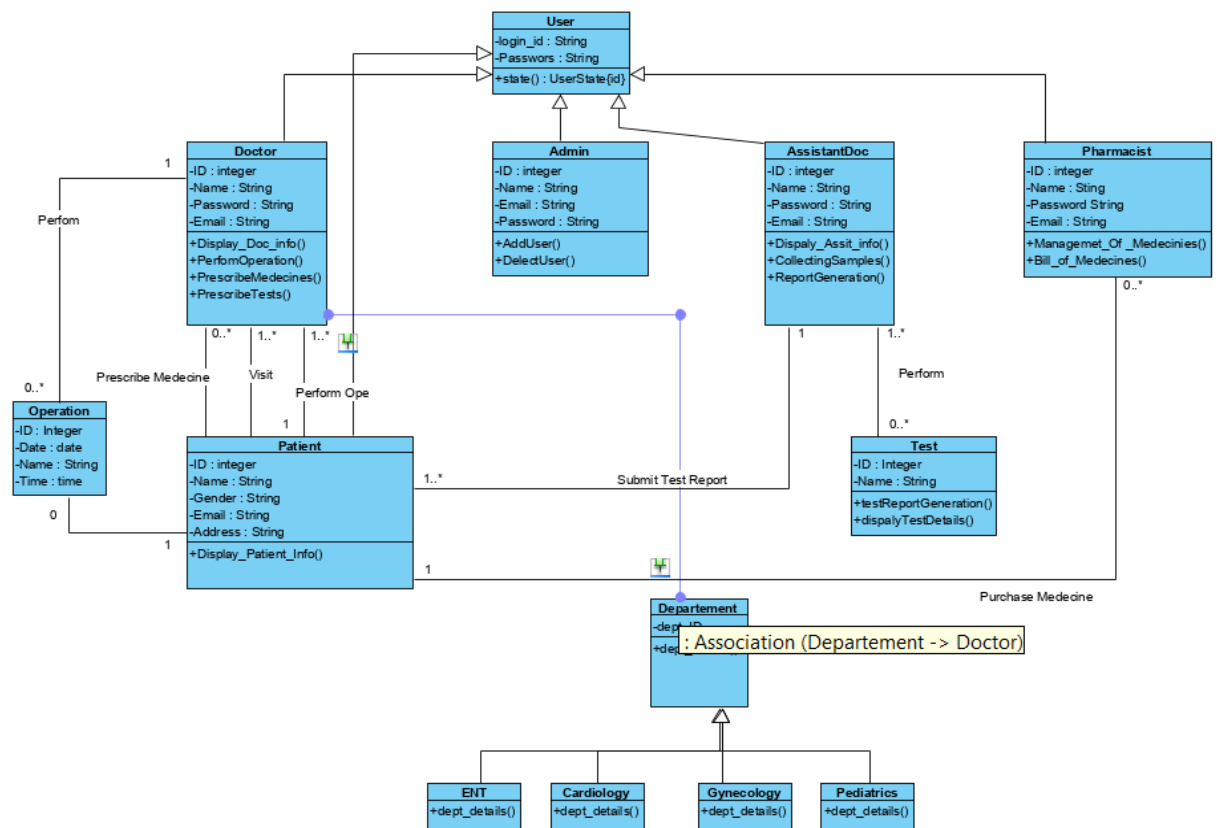
- Couche de présentation : l'interface graphique de l'application
- Couche logique : le traitement des données introduites, des actions des utilisateurs et de leurs commandes
- Couche de stockage : la base de données avec les utilisateurs et les livres

4.1. Identification des composantes

Dans notre projet, les composantes principales sont les suivantes :

- l'interface : est réalisé avec **JavaFX** dans des fichiers FXML
- le **controller** : fait la liaison entre l'interface et l'application ; il prend les données introduites et les autres informations et les envoie vers la couche logique de l'application pour être traitées ; si ce dernier retourne quelque chose, le **controller** convertit les données dans des éléments d'interface graphique et les affiche.
- le **service** : c'est la couche logique de l'application ; c'est le niveau où a lieu le traitement des données et aussi où il y a la seule liaison proprement dite avec la base de données.
- la **base de données** : réalisé avec **SQL**, elle contient les utilisateurs de la plateforme, et des détails concernant ces derniers ; la connexion avec la base de données est faite au moment où on ouvre l'application, dans la fenêtre de connexion

4.2. Diagramme de classes



4.3. Design patterns – au moins 2

Comme Design patterns nous avons premièrement **Façade** car il simplifie l'interface client.

C'est l'objectif principal de ce modèle qui est facile à mettre en œuvre et simple à ajouter du code. De plus c'est également un **re-factoring Pattern**.

Comme autre design patterns nous pouvons aussi citer dans notre cas **Singleton**, parce que l'on veut s'assurer qu'on a une seule instance de connexion de la base de données et que toutes les classes du programme utilisent la même base de données afin de faciliter l'exécution.

5. Diagramme (schéma) de la base de données

