

4.1 Introduction

Ce chapitre constitue le premier pas dans la réalisation de notre projet. Il est nécessaire de fixer quelques choix sur l'approche à suivre et sur le langage ou le paradigme à utiliser pour représenter la solution proposée. Ensuite, nous allons détailler la problématique en essayant de comprendre le système existant et de reprendre les notions théoriques acquises durant l'étude théorique pour pouvoir proposer une solution.

4.2 Cycle de vie en Génie Logiciel

4.2.1 Définition

Il s'agit du processus de gestion d'une application. Il regroupe l'ensemble des phases par lesquelles passe généralement un produit partant de l'identification des besoins jusqu'à la clôture du projet. L'objectif d'un tel découpage est de permettre de déterminer la validité du projet et la vérification de son processus de développement. Le but est de faciliter la préparation de la mission, ainsi que son bon déroulement. Cette structure offre aussi la possibilité d'identifier les grandes étapes du planning.

4.2.2 Activité d'un cycle de vie

Chaque projet est différent mais généralement chacun comprend au minimal les activités suivantes :

Définition du besoin : Au commencement, cette étape permet de définir les objectifs du projet et sa stratégie globale, elle sera appelée à être enrichie au cours de la mission.

Conception générale : Élaboration théorique de la solution et la préparation de l'architecture générale du système et de définir chaque sous-ensemble du logiciel à produire.

Conception détaillée : C'est la phase durant laquelle les exigences sont transformées en caractéristiques spécifiques ou en spécification d'un produit, d'un procédé ou d'un système avant sa mise en œuvre.

Développement (Implémentation ou programmation) : Il s'agit d'une traduction des fonctionnalités définies dans la phase de conception par son implémentation en utilisant une technologie adéquate.

Etape de clôture : C'est la dernière étape et dans la quelle le système développé est déployé en production.

Ces étapes de base sont organisées suivant le système, la nature de ses besoins, et les contraintes techniques et organisationnelles. Ainsi, nous avons plusieurs processus proposés tels que : cycle de vie en V, cycle de vie en spirale, cycle de vie en cascade, etc. Dans le cadre de ce projet, nous avons les contraintes suivantes :

- Environnement stable : contrairement à un environnement hostile et dynamique, la bibliothèque de la faculté des sciences exactes et d'informatique peut être considérée comme un environnement stable avec peu à pas de changements.
- Nos besoins fonctionnels sont stables : vu la durée de ce projet, des changements importants des besoins sont peu probables.
- Pas d'exigences techniques : à part la nécessité d'utiliser des technologies web (imposées par la nature du web 2.0), aucune autre contrainte technique n'est imposée. Nous notons aussi l'absence d'un code de base, ainsi, nous pouvons entamer l'analyse et la conception sans une attention particulière au choix technique.
- Une seule itération majeure : en tenant compte de la durée de réalisation ainsi que la taille du système d'un point de vu organisationnel (peu de postes de travail), une seule itération

majeure est planifiée. Il sera par la suite possible d'apporter des mises à jour limitées (amélioration de l'IHM, statistiques) mais sans un changement réel sur la structure de la base de données ou bien au niveau des fonctionnalités.

Par conséquent, nous avons opté pour le processus en cascade.

4.2.3 Processus en Cascade

Le modèle en cascade (waterfall model) est un modèle de gestion linéaire qui divise les processus de développement en phases de projet successives. Contrairement aux modèles itératifs, chaque phase est effectuée une seule fois. Les sorties de chaque phase antérieure sont intégrées comme entrées de la phase suivante. Le modèle en cascade est principalement utilisé dans le développement de logiciels (voir fig : 4.1).

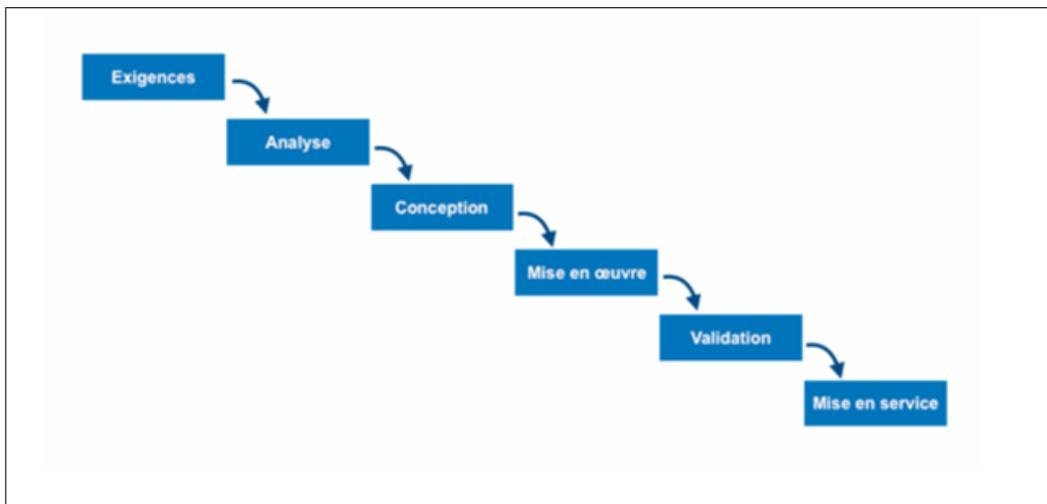


FIG. 4.1: Modèle en cascade

4.3 Langage de modélisation : UML

4.3.1 Historique d'UML

L'UML est le résultat de la fusion de précédents langages de modélisation objet : Booch, OMT, et OOSE. Principalement issu des travaux de Grady Booch, James Rumbaugh et Ivar Jacobson, UML est à présent un standard adopté par l'Object Management Group (OMG). UML 1.0 a été normalisé en janvier 1997; UML 2.0 a été adopté par l'OMG en juillet 2005. La dernière version de la spécification validée par l'OMG est UML 2.5.1.

4.3.2 Présentation d'UML

Le Langage de Modélisation Unifié, de l'anglais Unified Modeling Language (UML), est un langage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système. Il est couramment utilisé en développement logiciel et en conception orientée objet. La notation UML est un langage visuel constitué d'un ensemble de schémas, appelés des diagrammes, qui donnent chacun une vision différente du projet à traiter. UML nous fournit donc des diagrammes pour représenter le logiciel à développer : son fonctionnement, sa mise en route, les actions susceptibles d'être effectuées par le logiciel, etc. UML utilise l'approche objet en présentant un langage de description universel. Il est utilisé pour spécifier, visualiser, modifier et construire les documents nécessaires au bon développement d'un logiciel orienté objet. UML offre un standard de modélisation, pour représenter l'architecture logicielle il est très explicite. Les différents éléments représentables sont : acteurs, processus, activités d'un objet/logiciel, et composants logiciels. Il peut être utilisé pour la modélisation des bases de données de type objet (SGBDO). Grâce aux outils de modélisation UML, il est également possible de générer automatiquement tout ou partie du code d'une application logicielle, par exemple en langage Java, à partir des divers documents réalisés.

4.3.3 Points forts d'UML

UML est un langage formel et normalisé. Il permet aussi le gain de précision et de stabilité. Il est considéré comme un support de communication performant grâce à sa richesse. En association à un processus de génie logiciel, il cadre l'analyse et facilite la compréhension de représentations abstraites et complexes. Son caractère polyvalent et sa souplesse en font un langage universel. Ainsi, il sera notre choix pour ce projet.

4.4 Système existant et Problématique

Problématique : L'utilisation des technologies de l'information et de la communication s'est généralisée sur tous les domaines de la vie économique et sociale. La bibliothèque universitaire a trouvé dans ces technologies un meilleur moyen pour gérer ses ressources et suivre son activité quotidienne. Cependant, la bibliothèque universitaire a encore besoin d'introduire un niveau plus élevé de technologie à son système pour poursuivre la tendance de ces technologies.

C'est le cas de notre bibliothèque universitaire (bibliothèque de la faculté des sciences exactes et d'informatique) qui manque des technologies dites du web 2.0. Notre objectif est de proposer une solution qui permet ce passage d'une bibliothèque traditionnelle à une bibliothèque 2.0.

Système existant : La bibliothèque de notre faculté a un catalogue OPAC utilisé principalement pour localiser les livres et autres documents disponibles dans la bibliothèque. La recherche est faite par la saisie du nom du document ou son matricule ou bien le nom du sujet que l'utilisateur désire trouver (voir fig : 4.2). C'est l'unique cas d'utilisation offert actuellement par le catalogue. Nous avons été contactés par les bibliothécaires pour proposer une solution pour inclure les technologies web 2.0 aux différents services adaptables à ces technologies.

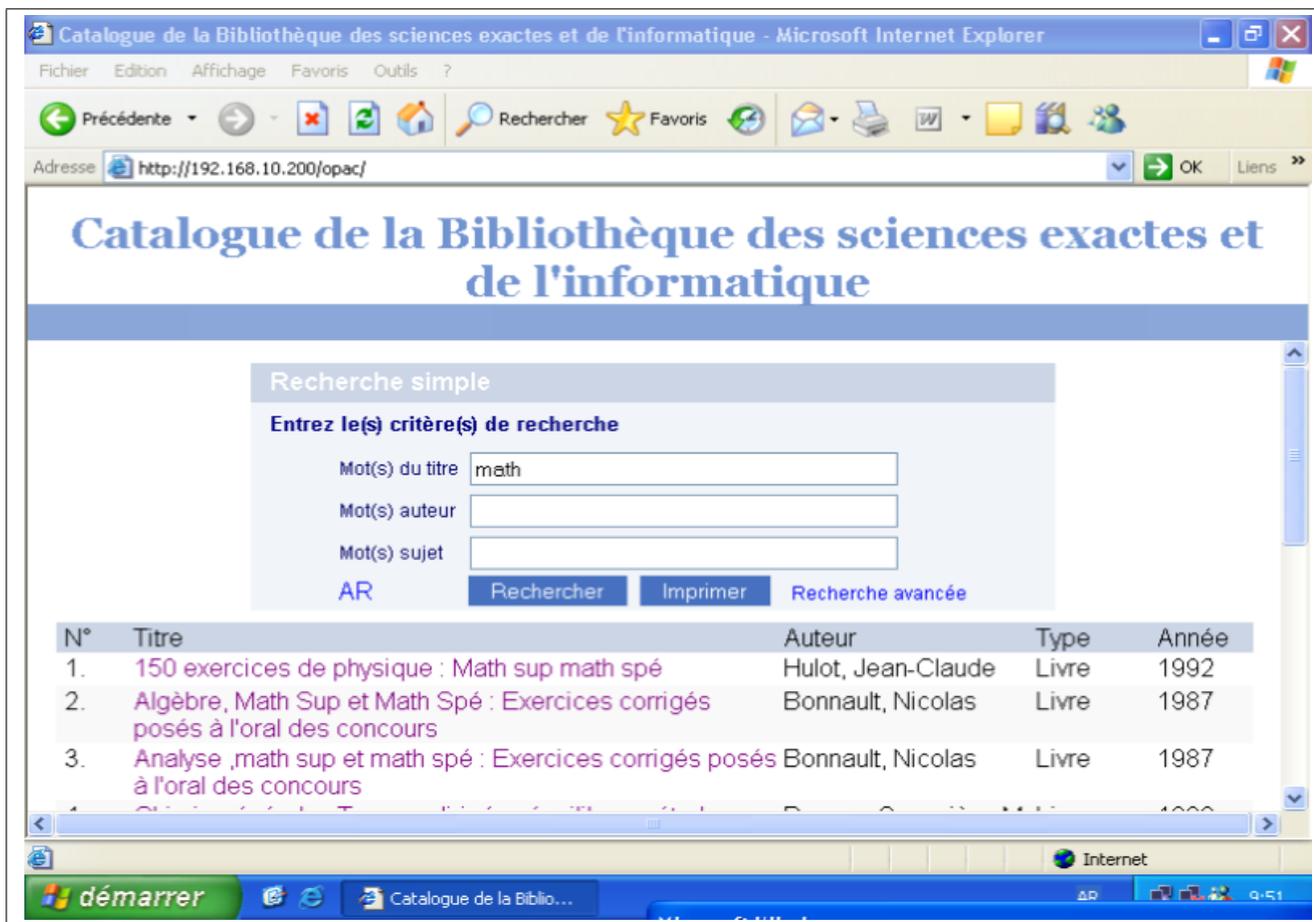


FIG. 4.2: Catalogue actuel de la faculté

Fonctionnalités désirés : Nous nous sommes rendues à la bibliothèque et nous avons effectué un interview pour pouvoir reconnaître les fonctionnalités désirées en priorité pour la nouvelle bibliothèque 2.0. L'interviewé nous a confirmé que le système actuel souffre d'une grave insuffisance de développement technologique surtout pour toutes les technologies en relation

avec le web 2.0 comme les wikis, blogs, chat, RSS, podcast, vodcast, et les réseaux sociaux. Il a également souligné la nécessité d'avoir un espace de collaboration et d'interactivité entre les étudiants mais aussi entre les bibliothécaires et les étudiants. Ainsi, il est acceptable de voir l'ensemble de besoins exprimés comme une volonté de passer vers une bibliothèque 2.0 avec un support particulier à la communication ; la chose qui est complètement absente dans la version actuel du OPAC. Par conséquent, notre vision doit se diriger vers le développement d'un portail web qui regroupe un ensemble d'outils web 2.0. Le choix des outils nous est laissé. Pour choisir parmi les différents existants, nous avons pris en compte les éléments suivants :

- Absence totale de tout composant web 2.0, ainsi, il est possible de commencer par les outils les plus simples.
- Nécessité de travailler sur des composants internes, par conséquent, le composant « réseau social » est écartée dans un premier temps.
- Il est prioritaire d'intégrer un mécanisme de communication directe entre les bibliothécaires et les étudiants, ainsi, un mécanisme de chat synchrone doit être considéré dans le système.

4.5 Analyse et choix architecturaux

Choix architectural : Sur ce que nous avons appris, nous avons pris les choix architecturaux suivants :

- La fonctionnalité principale est la recherche des livres à travers un OPAC, néanmoins, nous visons à améliorer l'interface actuelle ainsi que les technologies de base pour permettre d'intégrer d'autres fonctionnalités et de naviguer entre les différents composants du système.
- Implémenter la communication entre les étudiants à travers les commentaires sur les livres. Ce mécanisme est utilisé dans la grande partie des réseaux sociaux spécialisés où les internautes communiquent à travers des commentaires postés sur un objet central. À titre d'exemple, Yelp permet la communication autour des restaurants ; les internautes communiquent leurs expériences, conseils et recommandations à travers des commentaires

postés sur des restaurants. Un autre exemple est GoodReads où les objets centraux sont des livres.

- Donner la possibilité de postuler une évaluation sur un livre. Nous nous inspirons ici aussi des réseaux sociaux Yelp et GoodReads.
- Ajouter un support de communication synchrone direct sous forme d'un système de Chat.
- Ajouter un blog pour donner la possibilité d'une communication dans le cadre des services d'extension et pour des sujets qui ne concernent pas les livres d'une manière directe.
- Mettre au point un wiki pour former le personnel et les étudiants sur le nouveau portail. Ce wiki doit être enrichi par un contenu multimédia pour garantir une meilleure exploitation des possibilités techniques et pour des meilleurs résultats auprès des utilisateurs.
- Mettre au point un flux RSS vu qu'il est le mode de diffusion le plus adéquat en comparaison aux mailing-lists.

Nous envisageons l'utilisation de ces composants suivants les scénarios suivants :

- Nous allons aussi construire un catalogue modern qui permet la recherche des livres par leurs noms, il sera possible pour les membres de la bibliothèque d'afficher les détails des livres, les évaluer et d'exprimer leurs avis concernant ces livres.
- Le blog, qui est un outil de publication géré par les administrateurs, est très facile à utiliser. Il contiendra des annonces, des articles et des sondages sur lesquels les autres membres peuvent commenter. Son utilisation dans notre portail sera pour que le bibliothécaire puisse informer les autres membres des mises à jour et des nouveaux ajouts au catalogue. Il sera aussi possible aux étudiants d'« aimer » ou « pas aimer » une publication.
- Le chat figure souvent dans différents réseaux sociaux et est utilisé pour établir des discussions instantanées entre deux ou plusieurs personnes pour communiquer des pensées, des idées, des opinions et des questions entre les participants. Nous le réaliserons dans notre portail pour que le bibliothécaire et les étudiants puissent communiquer entre eux pour poser des questions ou avoir plus d'explications en ce qui concerne la bibliothèque et cela de manière instantanée.

- Les wikis sont supposés être des pages publiques qui servent à communiquer des informations et garantir la collaboration de l'administrateur et du lecteur donc créer une base de connaissance commune en ce qui concerne le portail pour avoir une vue détaillée sur son contenu et la manière d'utiliser ses services.

La présentation générale du projet :

L'objectif du projet est de construire et réaliser un catalogue modern pour la bibliothèque de la faculté des sciences exactes et d'informatique et utilisation des technologies web 2.0. Nous résumons l'ensemble des opérations globales possibles sur le portail comme suit :

- Gestion des utilisateurs et des droits d'accès (inscription et authentification).
- Parcourir le catalogue dans but de trouver un livre et afficher les détails qui lui correspondent avec la possibilité de le commenter et de l'évaluer.
- Accéder au site pour obtenir des informations qui concernent la bibliothèque et le site sans avoir à s'authentifier.
- Contacter le bibliothécaire après authentification ou inscription pour poser des questions concernant la bibliothèque.
- Discuter avec le bibliothécaire en utilisant le service de messagerie de la bibliothèque.
- Voir les articles, publications, et les annonces, avec la possibilité de les commenter et de les évaluer.

4.6 Conclusion

Dans ce chapitre nous avons présenté la problématique principale de ce projet. Nous avons aussi exploité les connaissances acquises durant l'étude théorique pour proposer une solution sous forme d'un portail avec plusieurs composants où chaque composant représente un outil web 2.0. L'étape suivante est la conception du système, c'est-à-dire traduire cette analyse en unesolution informatique. C'est l'objectif du chapitre suivant.

5.1 Introduction

Dans ce chapitre, nous allons traduire la solution proposée sous forme d'une solution informatique. Pour cela, nous allons faire appel au langage UML choisi comme langage de représentation pour la solution à développer. Dans ce chapitre, nous allons définir d'une manière précise et explicite tous les cas d'utilisation du système. Nous allons aussi décrire chaque cas d'utilisation en utilisant un tableau de description ainsi qu'un diagramme de séquence. En progressant, nous allons construire à fur et à mesure le diagramme de classes de notre système.

5.2 Rappel sur les notions UML utilisées

Un acteur : Un acteur représente un ensemble cohérent de rôles joués par des entités externes (utilisateur humain, matériel ou autre système) qui interagissent avec le système.

Un cas d'utilisation : Le cas d'utilisation est une description des interactions qui vont permettre à l'acteur d'atteindre son objectif en utilisant le système. Il définit « une façon d'utiliser le système pour accomplir une tâche ».

Diagramme des cas d'utilisation : En langage de modélisation unifié (UML), un diagramme de cas d'utilisation peut servir à résumer les informations des utilisateurs du système (également appelés acteurs) et leurs interactions avec ce dernier. Il regroupe les cas d'utilisation du système ou de l'un de ses sous-systèmes. La création de ce type de diagramme requiert un ensemble de

symboles et de connecteurs spécifiques.

Lorsqu'ils sont bien conçus, les diagrammes de cas d'utilisation peuvent aider les équipes à collaborer et représenter :

- Les scénarios dans lesquels le système ou application interagit avec des personnes, des organisations ou des systèmes externes.
- Les objectifs que le système ou application permet aux acteurs d'atteindre.

Diagramme de séquence : Un diagramme de séquence est un diagramme UML (Unified Modeling Language) qui représente la séquence de messages entre les objets au cours d'une interaction. Un diagramme de séquence comprend un groupe d'objets, représentés par des lignes de vie, et les messages que ces objets échangent lors de l'interaction. Les diagrammes de séquence représentent la séquence de messages transmis entre des objets. Ils peuvent également représenter les structures de contrôle entre des objets. Dans le suite de ce rapport, nous allons prendre le système comme une boîte noire. Les détails internes vont dépendre de la technologie et l'approche choisies pour développer le système.

Diagramme de classes : Le diagramme de classes est un schéma utilisé pour présenter les classes et les interfaces des systèmes ainsi que les différentes relations entre celles-ci. Ce diagramme fait partie de la partie statique d'UML car il fait abstraction des aspects temporels et dynamiques. Une classe décrit les responsabilités, le comportement et le type d'un ensemble d'objets. Les éléments de cet ensemble sont les instances de la classe. En général, un diagramme de classes peut contenir les éléments suivants :

1. Une classe : Représente la description formelle d'un ensemble d'objets ayant une sémantique et des caractéristiques communes. Elle est représentée en utilisant un rectangle divisé en trois sections. La section supérieure est le nom de la classe, la section centrale définit les propriétés de la classe alors que la section du bas énumère les méthodes de la classe [12].

2. Un attribut : Représente les données encapsulées dans les objets des classes. Chacune de ces informations est définie par un nom, un type de données, une visibilité et peut être initialisé. Le nom de l'attribut doit être unique dans la classe [12].

3. Une association : Une association représente une relation entre deux classes. Une association indique que les objets d'une classe ont une relation avec des objets d'une autre classe, dans lesquels cette connexion a une signification spécifiquement définie.

Diagramme de paquetage : En UML, un paquetage (package en anglais) est un mécanisme qui permet d'organiser des éléments de modélisation en groupes.

5.3 Définition des acteurs

En se basant sur l'analyse effectuée, nous avons défini les acteurs suivants :

Visiteur : Cet acteur désigne un utilisateur qui n'est pas inscrit dans le système, il peut accéder à certaines fonctionnalités du portail comme : le wiki, le catalogue et le blog. Il peut aussi s'inscrire pour devenir un membre.

Membre : Cet acteur représente tout utilisateur inscrit dans le système et qui peut accéder au chat et interagir avec d'autres utilisateurs par les commentaires sur les livres ou les articles du blog.

Bibliothécaire : C'est un membre avec des privilèges plus élevés. Il est responsable de la gestion du blog et du catalogue. Il doit aussi veiller sur le contenu du portail (rôle de modérateur).

5.4 Cas d'utilisation et leurs descriptions

Organisation des diagrammes des cas d'utilisation : Pour simplifier les diagrammes des cas d'utilisation, nous avons opté pour une représentation par sous-système. Chaque diagramme regroupe les cas d'utilisation de l'un des sous-systèmes définis après l'analyse de la problématique.

5.4.1 Diagramme des cas d'utilisation globaux

La figure ci-dessous présente les cas d'utilisation globaux. Il s'agit des cas d'utilisation du visiteur avec les cas d'utilisation simples des autres utilisateurs (les cas d'utilisation qui ne représente pas un sous-système complexe).

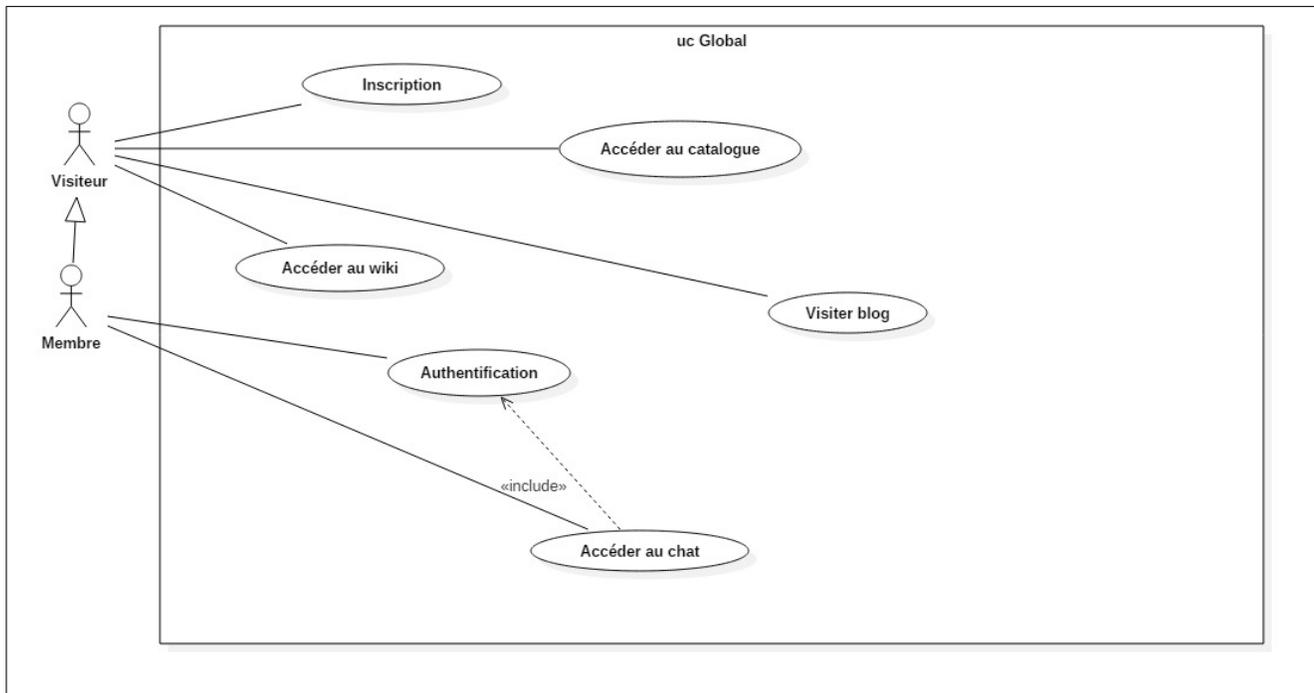


FIG. 5.1: Diagramme des cas d'utilisation globaux

5.4.1.1 Diagrammes de séquence associés aux cas d'utilisation globaux

Cas d'utilisation : Inscription

Pour ce cas d'utilisation, le choix conceptuel majeur qui a été fait est de laisser l'inscription ouverte. C'est-à-dire, la création du profil et des informations de connexion est la responsabilité de l'utilisateur et non pas de l'administration du système. L'inconvénient de cette approche est la possibilité d'avoir des personnes non-autorisées qui s'inscrivent sur la plate-forme. Pour résoudre ce problème, l'administrateur doit valider chaque inscription avant de pouvoir accéder au système.

Inscription
Objectif : inscrire un visiteur pour qu'il devient membre
Acteur principal : le visiteur
Précondition : accéder au site
Déroulement :
<ul style="list-style-type: none"> • Demander le formulaire à remplir • Le système affiche le formulaire • Le visiteur introduit ses informations • Le système affiche une interface avec un message du succès.
Exception :
<ul style="list-style-type: none"> • Les informations sont incorrectes : le système affiche un message d'erreur.
Postcondition : L'ajout d'un nouveau compte

TAB. 5.1: FT de diagramme de séquence de l'UC Inscription

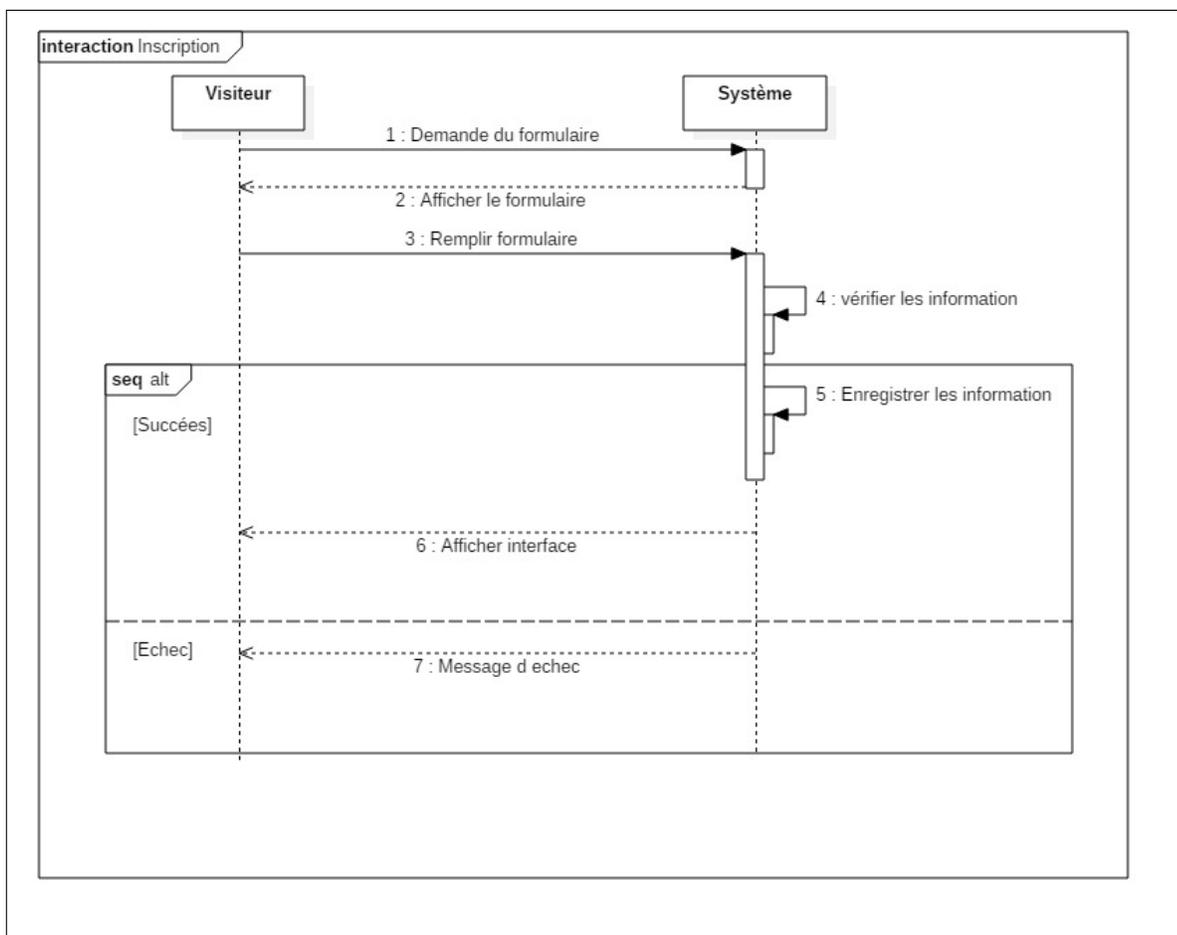


FIG. 5.2: Diagramme de séquence de l'inscription

Cas d'utilisation : Authentification

Le choix conceptuel effectué pour ce cas est le choix entre mettre le formulaire d'authentification sur la page d'accueil ou bien réserver une page d'authentification séparée. Nous avons opté pour

le deuxième choix pour laisser la page d'accueil minimale et est centralisé sur la recherche des livres.

Authentification
Objectif : authentifier un membre du site web
Acteur principal : le membre (etudiant, bibliothécaire)
Précondition : doit accéder au site
Déroulement :
<ul style="list-style-type: none"> • Demander le formulaire a remplir • Le système affiche le formulaire • Le membre introduit ses informations • Le système affiche une interface
Exception :
<ul style="list-style-type: none"> • Information incorrectes ou le compte n'existe pas ou il n'est pas activé : le système affiche un message d'erreur
Postcondition : L'utilisateur est authentifié

TAB. 5.2: FT de diagramme de séquence de l'UC Authentification

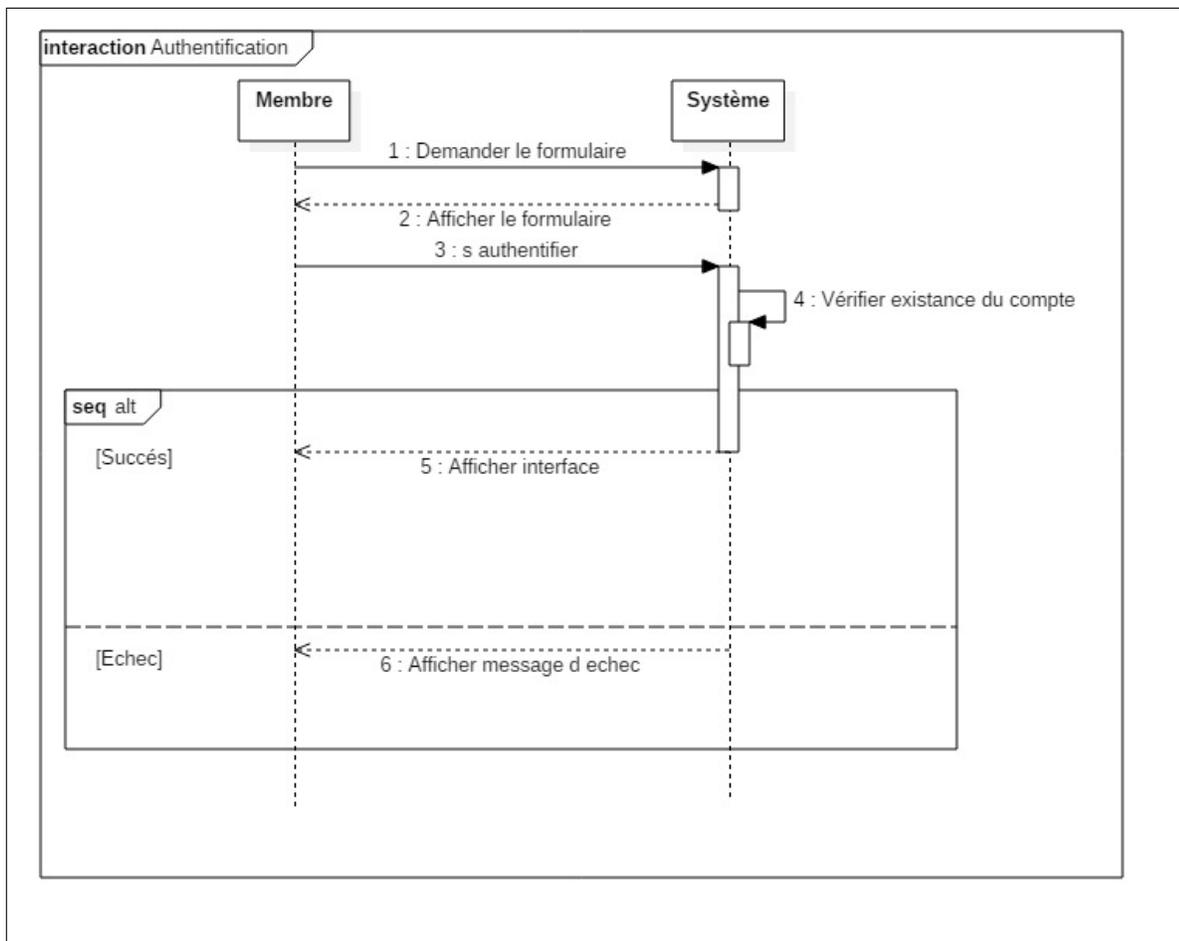


FIG. 5.3: Diagramme de séquence de l'authentification

Les autres cas d'utilisation sont intuitifs avec des interactions minimales. Par exemple, pour visiter le blog il suffit de cliquer sur son lien dans le menu pour que le système l'affiche. Aucune exception côté utilisateur ne peut être commise. Cette description correspond aussi à l'accès au wiki.

5.4.2 Diagramme des cas d'utilisation du sous-système Catalogue

La figure ci-dessous présente les cas d'utilisation du sous-système Catalogue :

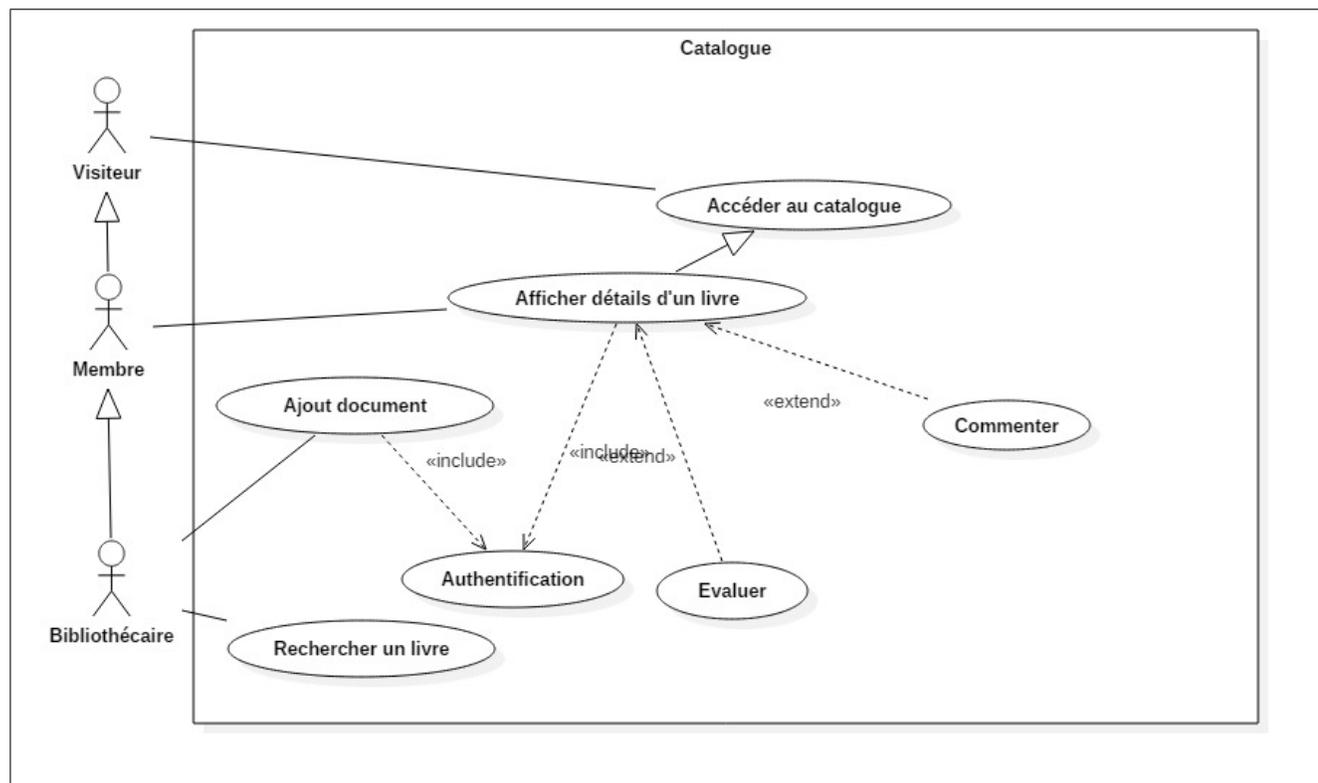


FIG. 5.4: Diagramme des cas d'utilisation du catalogue

5.4.2.1 Diagrammes de séquence associés aux cas d'utilisation du sous-système Catalogue

Cas d'utilisation : Ajout document

C'est le cas d'utilisation qui permet d'enrichir le catalogue de la bibliothèque. Aucun choix conceptuel a été pris à part le suivi d'une approche standard.

Ajout document
Objectif : ajouter un document au catalogue de la bibliothèque
Acteur principal : bibliothécaire
Précondition : s'authentifier autant que bibliothécaire
Déroulement : <ul style="list-style-type: none"> • Demander le formulaire d'ajout d'un livre. • Le système affiche le formulaire • L'utilisateur remplit le formulaire en introduisant les données du livre. • Le système envoie un message de succès.
Exception : <ul style="list-style-type: none"> • Information mal saisies : le système affiche un message d'erreur
Postcondition : ajout du livre au catalogue

TAB. 5.3: FT de diagramme de séquence de l'UC Ajout document

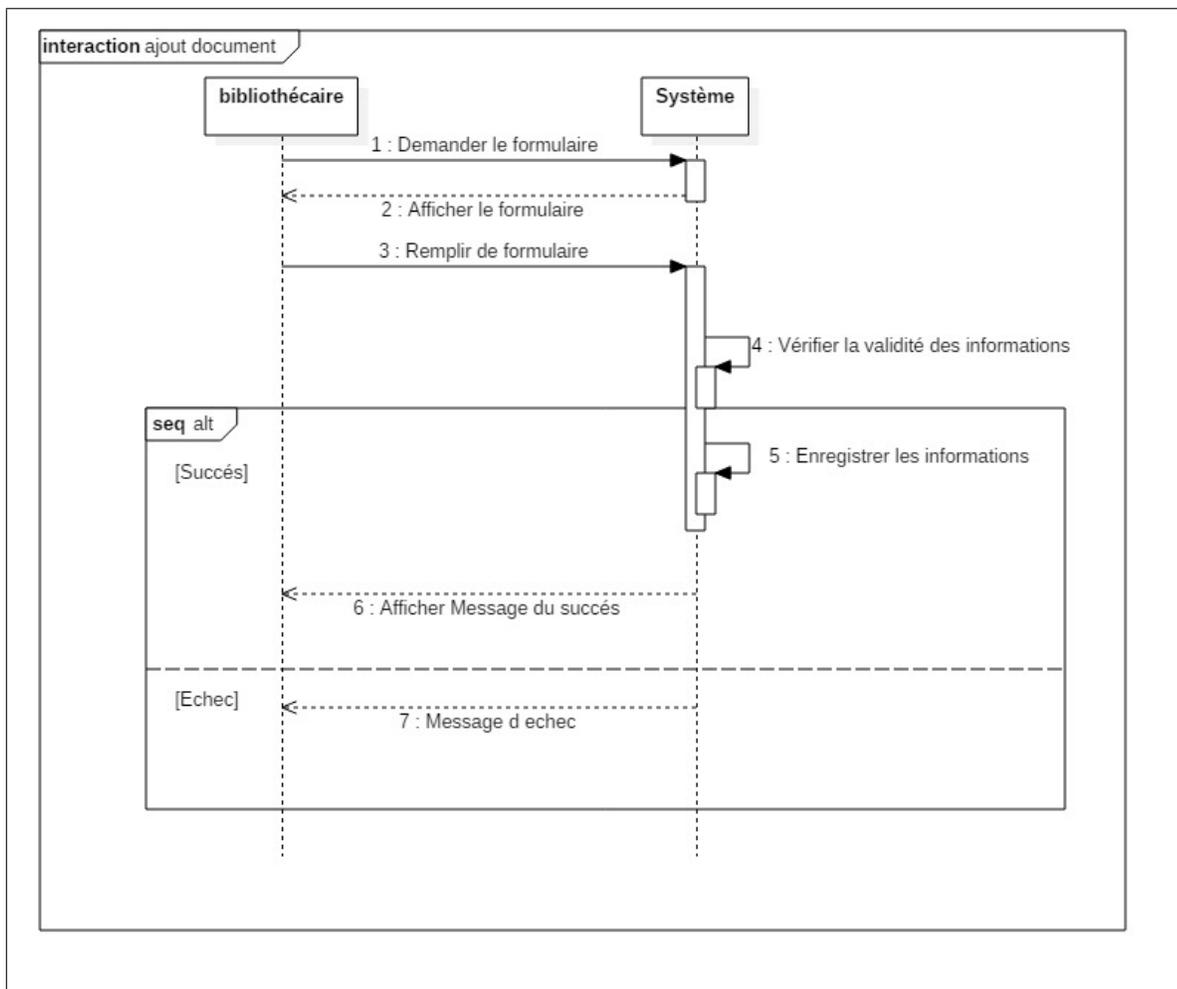


FIG. 5.5: Diagramme de séquence d'ajout d'un document

Cas d'utilisation : Rechercher un livre

C'est le cas d'utilisation principal du catalogue. Ainsi, il est simplifié d'une manière très simi-

lares aux moteurs de recherche utilisés quotidiennement.

Rechercher un livre
Objectif : Rechercher un livre au catalogue du site
Acteur principal : visiteur
Précondition : accéder au portail
Déroulement : <ul style="list-style-type: none"> • Saisir le titre ou le matricule ou l’auteur du livre désiré dans la barre de recherche • Si le résultat n’est pas vide : le système l’affiche son nom complet , son matricule et son auteur • Sinon : le système affiche de message pour informer l’utilisateur.
Exception : <ul style="list-style-type: none"> • aucune
Postcondition : aucune

TAB. 5.4: FT de diagramme de séquence de l’UC Rechercher un livre

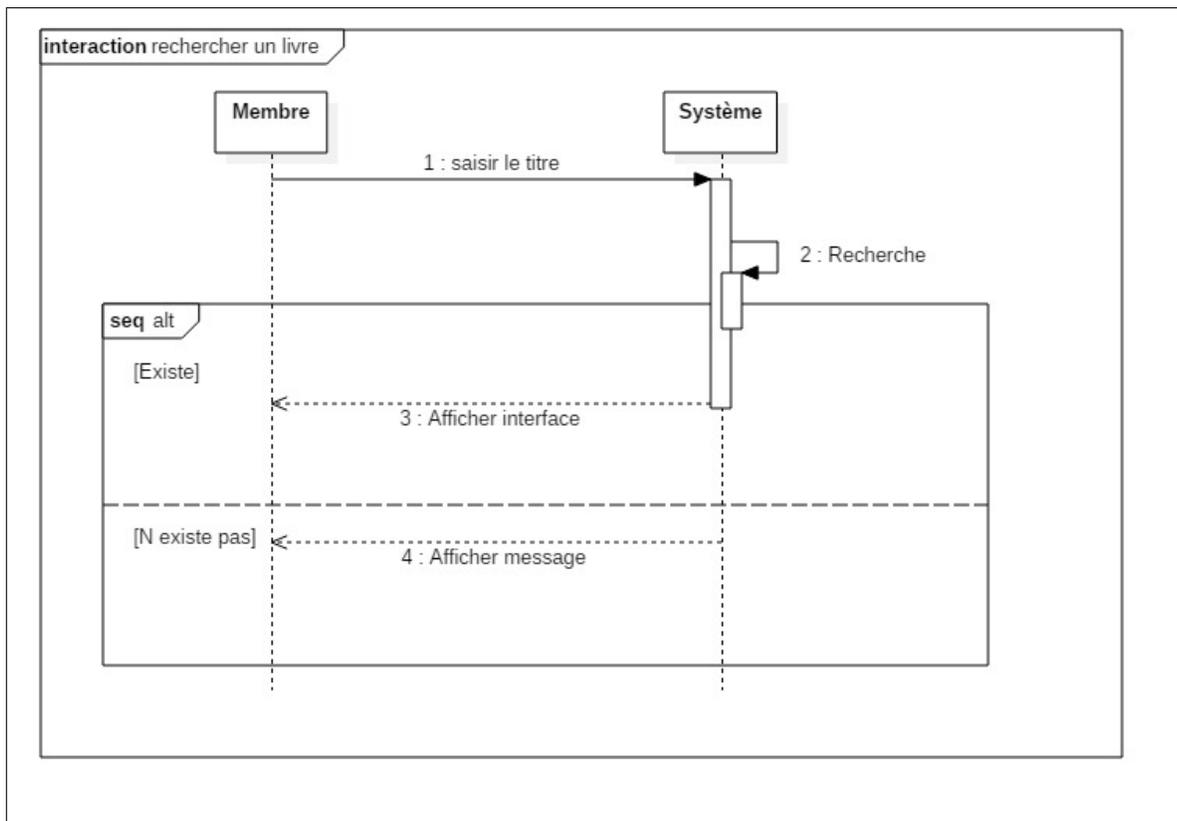


FIG. 5.6: Diagramme de séquence du recherche d’un livre

Les autres cas d’utilisation sont conçus pour simplifier l’expérience d’utilisateur. Par exemple, commenter et évaluer un livre ne nécessite pas un rechargement de la page ce qui réduit considérablement les interactions visibles à l’utilisateur.

5.4.2.2 Diagramme de classes du sous-système Catalogue

Ce diagramme représente les objets manipulés durant les cas d'utilisation liés au sous-système Catalogue. Les deux classes principales sont l'utilisateur et le livre.

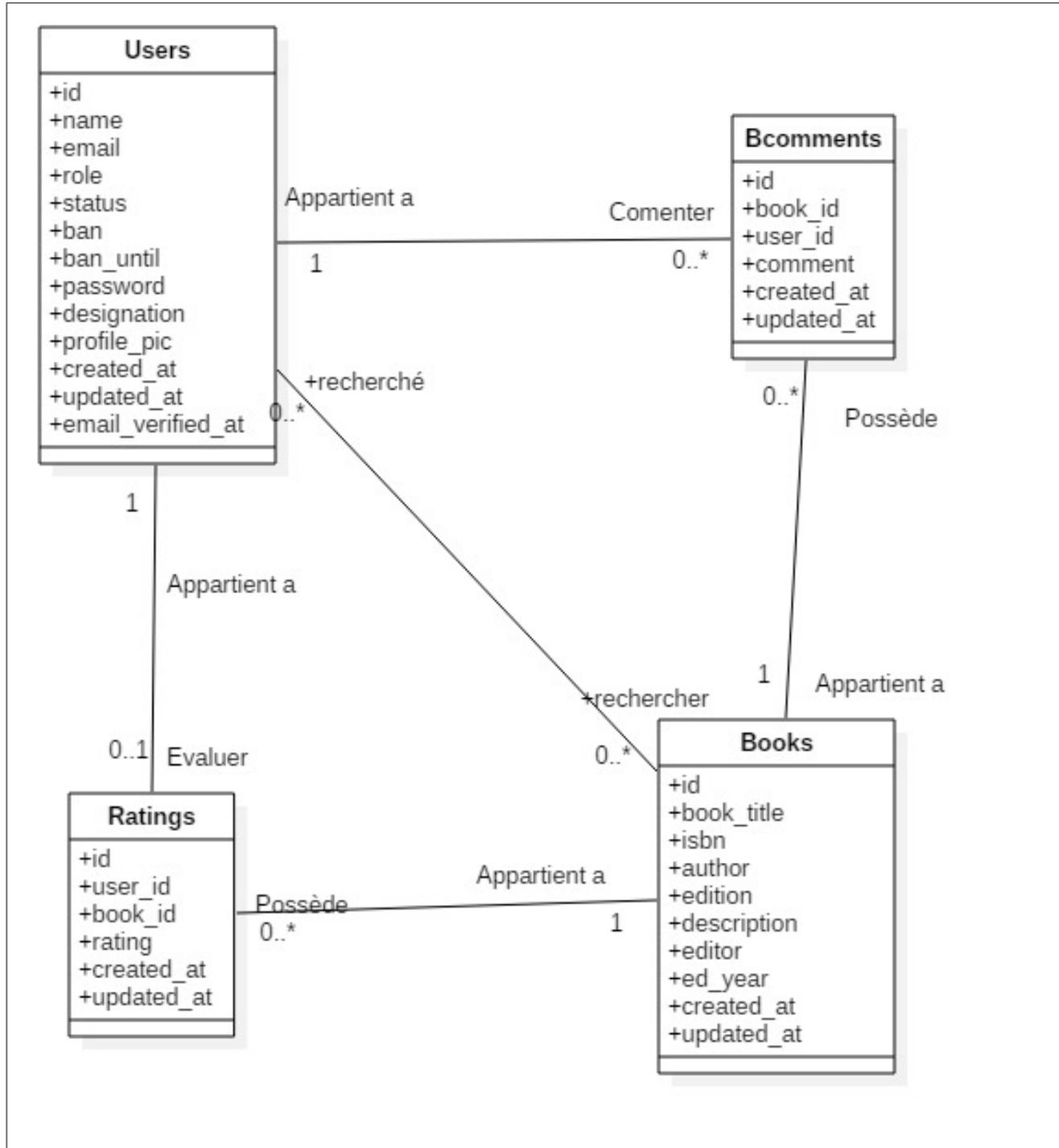


FIG. 5.7: Diagramme de classes du catalogue

5.4.3 Diagramme des cas d'utilisation liés au sous-système Blog

La figure ci-dessous présente les cas d'utilisation du sous-système Blog :

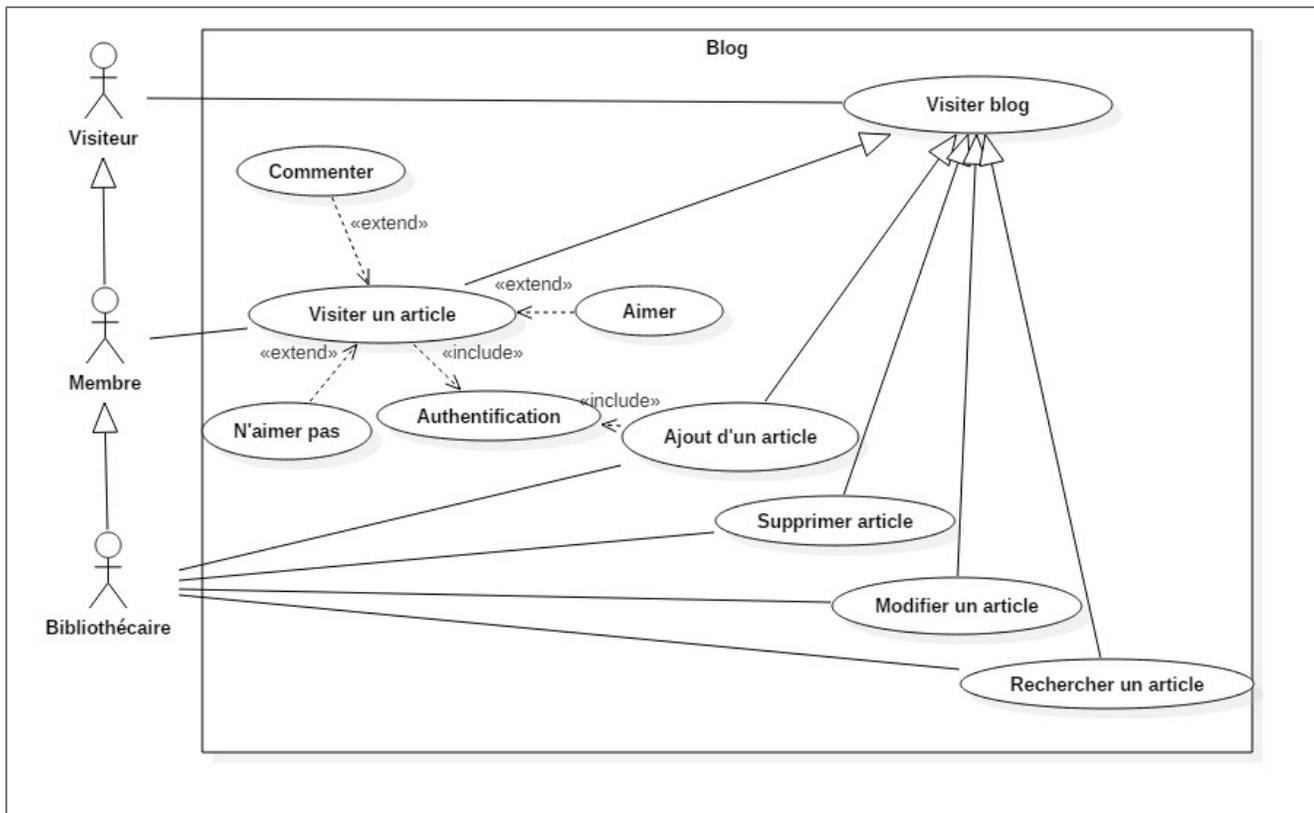


FIG. 5.8: Diagramme des cas d'utilisation du blog

5.4.3.1 Diagrammes de séquence associés aux cas d'utilisation du sous-système Blog

Cas d'utilisation : Ajout d'un article

Comme pour l'ajout d'un document et en utilisant la même approche, le bibliothécaire peut ajouter un article au blog. Il est nécessaire d'opter pour les mêmes séquences et la même IHM pour garder l'homogénéité du portail et simplifier la formation des bibliothécaires.

Ajout d'un article
Objectif : ajouter un article au blog du site
Acteur principal : bibliothécaire
Précondition : être authentifié comme bibliothécaire
Déroulement : <ul style="list-style-type: none"> • Demander le formulaire d'ajout d'un article • Le système affiche le formulaire • L'utilisateur remplit le formulaire en introduisant les données de l'article de blog • Le système envoie un message de succès et affiche l'interface qui comprend tous les articles.
Exception : <ul style="list-style-type: none"> • Les informations sont mal saisies : le système affiche un message d'erreur
Postcondition : un nouvel article est ajouté au blog

TAB. 5.5: FT de diagramme de séquence de l'UC Ajout d'un article

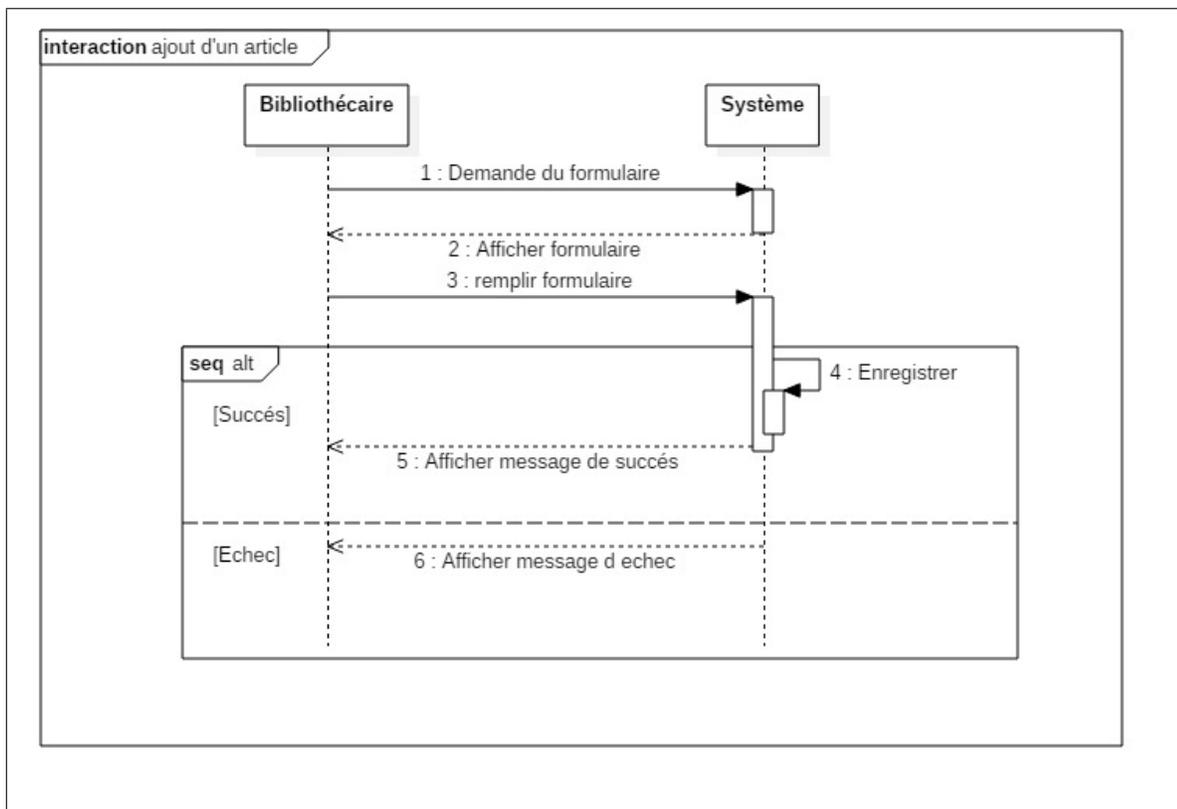


FIG. 5.9: Diagramme de séquence d'ajout d'un article

Cas d'utilisation : Rechercher un article

Comme pour le catalogue, il est possible de faire des recherches sur le contenu du blog. A ce niveau aussi, nous visons l'homogénéité du système.

Rechercher un article
Objectif :rechercher un article au blog du site
Acteur principal : visiteur
Précondition : accéder au blog
Déroulement : <ul style="list-style-type: none"> • Accéder à la page du blog et saisir dans la barre de recherche le nom de l'article • Si le résultat n'est pas vide : le système affiche la liste des articles correspondant à la requête • Sinon : le système affiche un message pour informer l'utilisateur
Exception : <ul style="list-style-type: none"> • aucune
Postcondition : aucune

TAB. 5.6: FT de diagramme de séquence de l'UC Rechercher un article

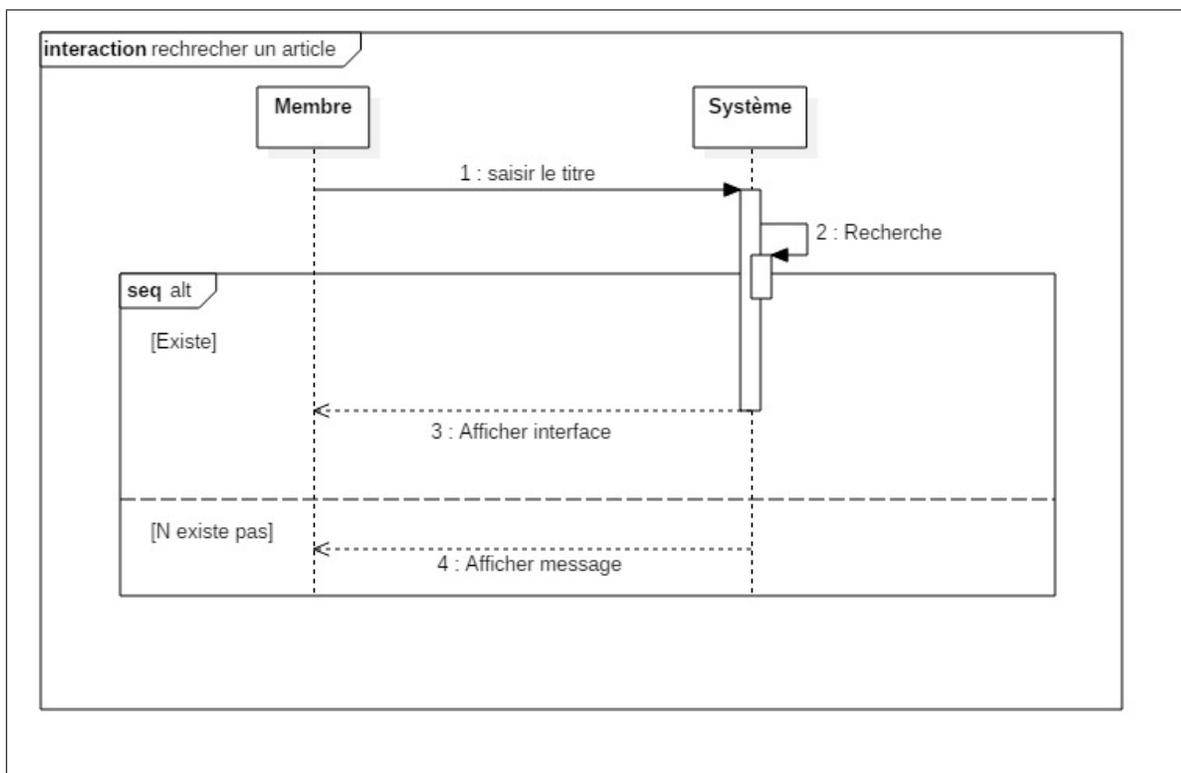


FIG. 5.10: Diagramme de séquence rechercher un article

Cas d'utilisation : Modifier un article

Respectant toujours le principe d'homogénéité, le cas d'utilisation " modifier article " reprend les grandes lignes du cas d'utilisation "Ajouter article"

Modifier un article
Objectif : modifier un article au blog du site
Acteur principal : bibliothécaire
Précondition : être authentifié comme bibliothécaire
Déroulement : <ul style="list-style-type: none"> • Demander le formulaire de modification d'article • Le système affiche le formulaire rempli avec les données de l'article • L'utilisateur modifier les données et valide le formulaire. • Le système envoie un message de succès et affiche l'interface qui comprend les informations de l'article
Exception : <ul style="list-style-type: none"> • Les informations sont mal saisies : le système affiche un message d'erreur
Postcondition : l'article sélectionné est mis à jour

TAB. 5.7: FT de diagramme de séquence de l'UC Modifier un article

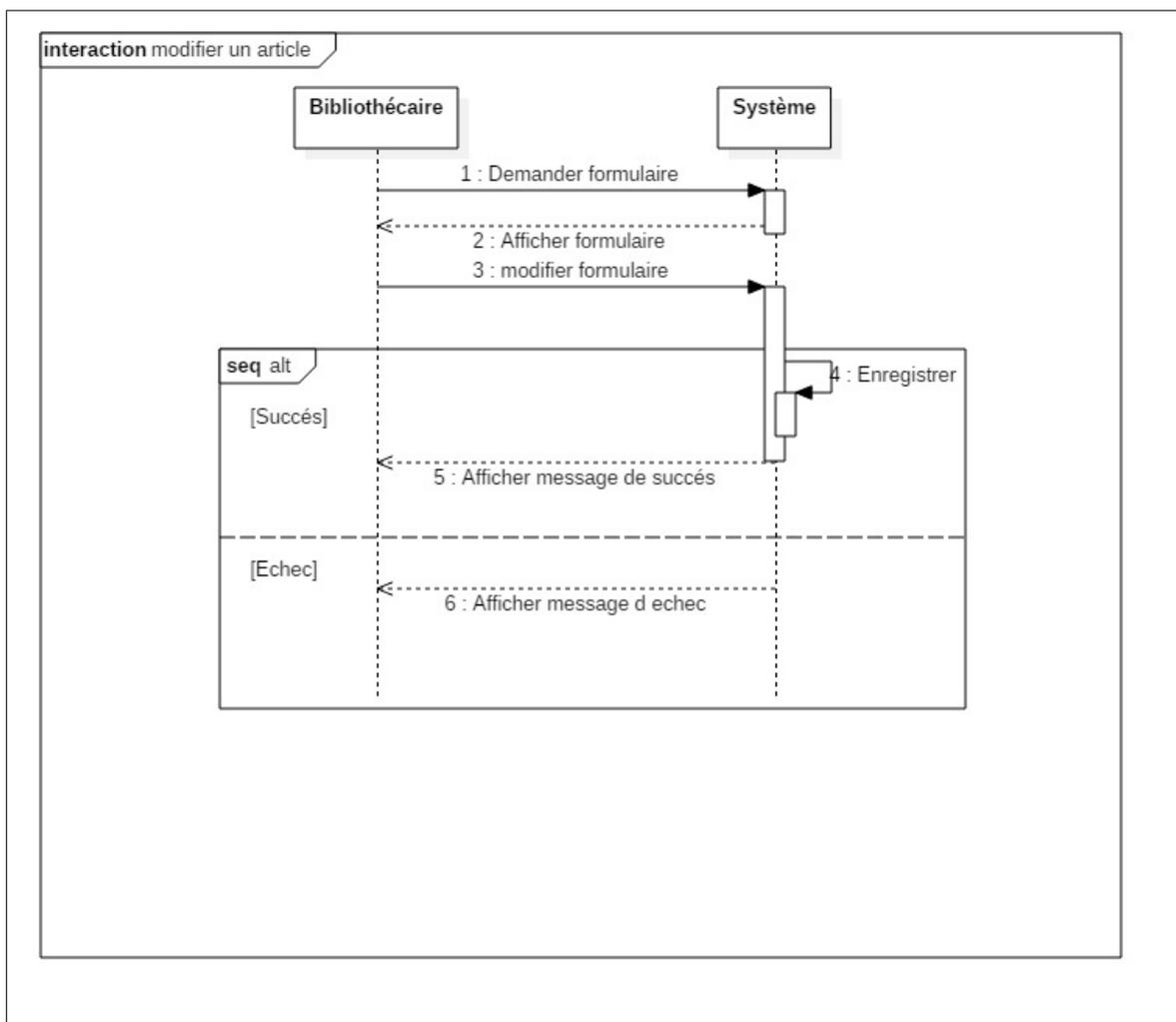


FIG. 5.11: Diagramme de séquence modifier un article

Les autres cas d'utilisation présentent peu d'interaction. Ils permettent à l'utilisateur de com-

menter et d'évaluer un article sur la même interface qui l'affiche. L'évaluation à titre d'exemple doit garantir un effort minimal (à la limite en un seul clique).

Diagramme de classes du sous-système Blog

La figure suivante représente le diagramme de classes correspondant au cas d'utilisation du blog :

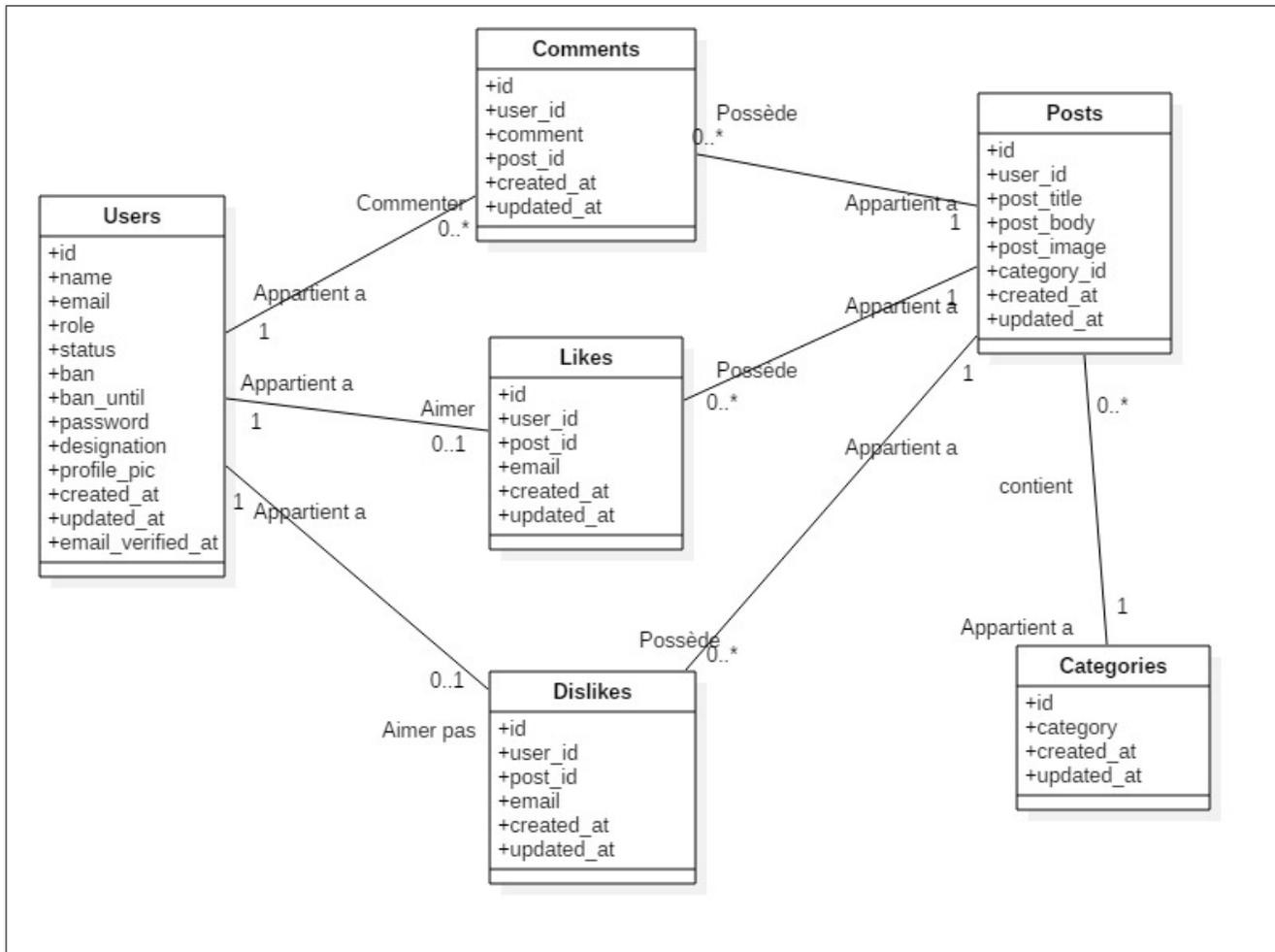


FIG. 5.12: Diagramme de classes du blog

5.4.4 Diagramme des cas d'utilisation pour le sous-système Chat

La figure suivante représente les cas d'utilisation du chat. La simplicité d'un système de chat est cruciale pour sa réussite. Le sous-système doit être simple avec peu de cas d'utilisation mais au même temps efficace.

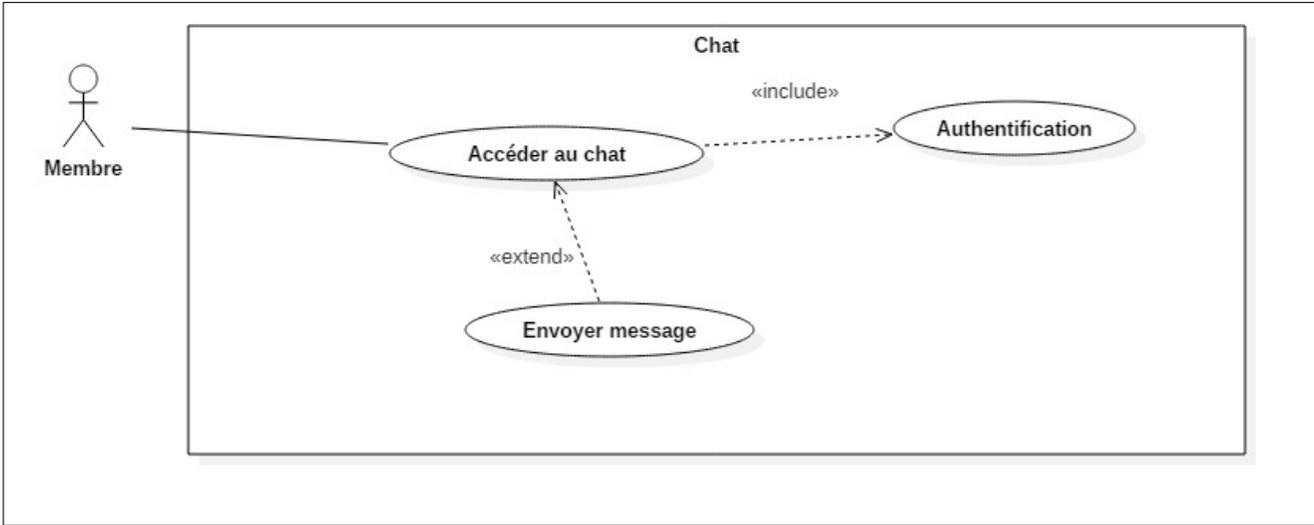


FIG. 5.13: Diagramme des cas d'utilisation de Chat

5.4.4.1 Diagramme de classes pour le sous-système Chat

La figure suivante représente le diagramme de classes correspondant au cas d'utilisation du chat :

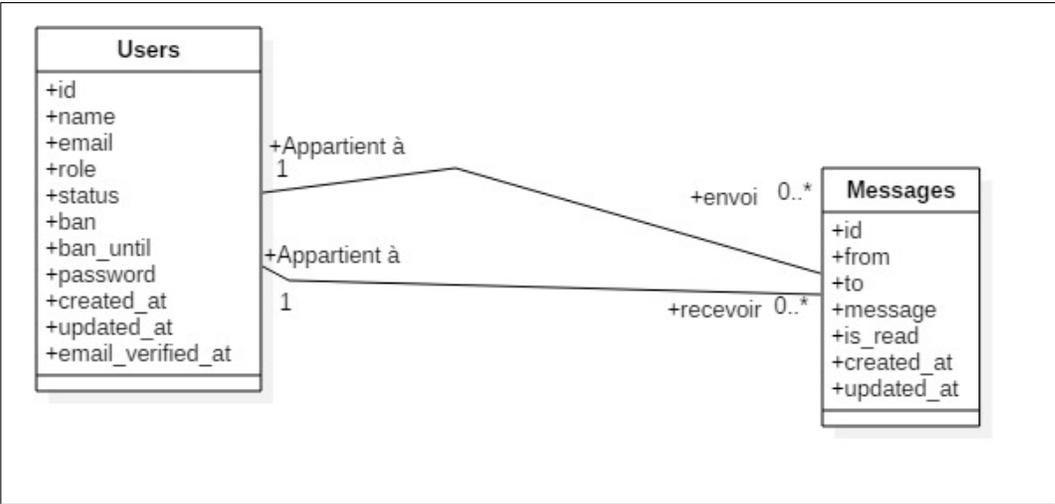


FIG. 5.14: Diagramme de classes de Chat

5.4.5 Diagramme des cas d'utilisation du sous-système Administration

Malgré que ces besoins ne sont pas explicitement exigés par les bibliothécaires, avoir un système multiutilisateur et collaboratif impose l'implémentation d'un ensemble d'outils pour

gérer et veiller sur le bon fonctionnement du système. La figure suivante représente les cas d'utilisation d'administration :

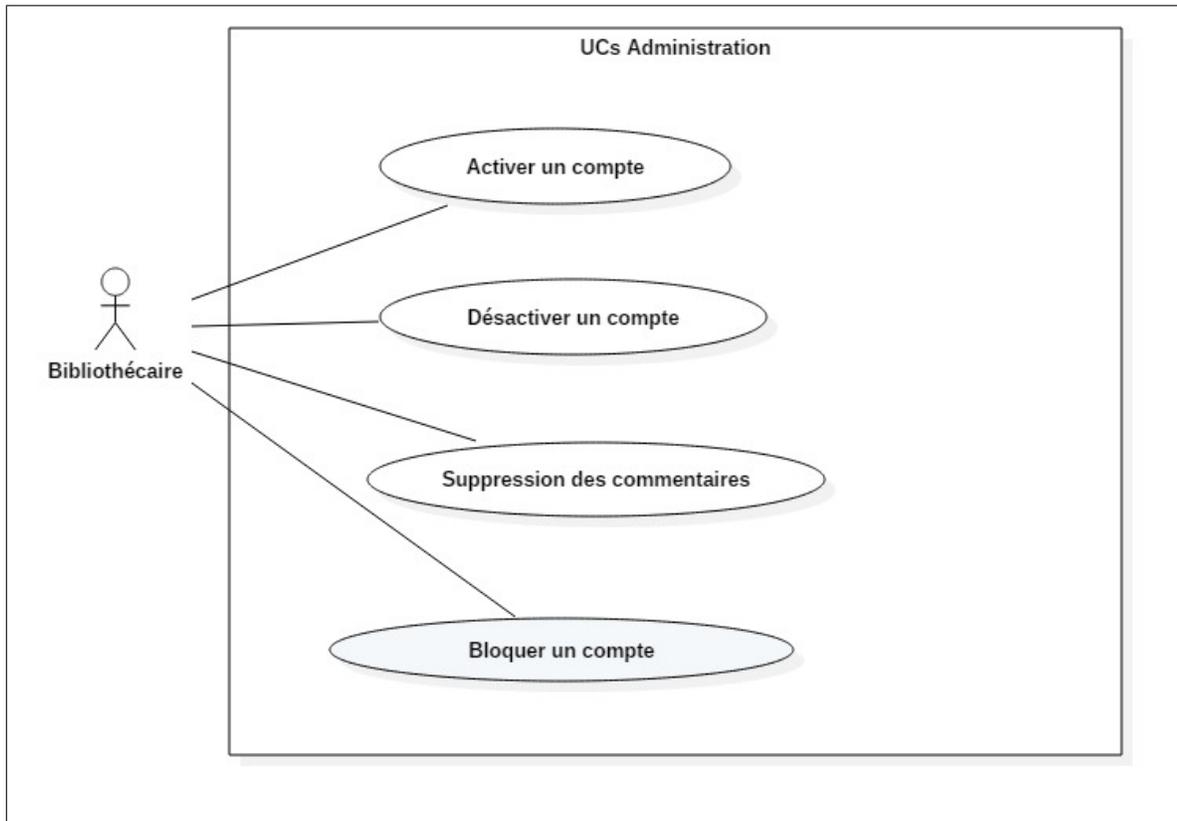


FIG. 5.15: Diagramme des cas d'utilisation d'administration

5.4.5.1 Diagrammes de séquences associés aux cas d'utilisation du sous-système Administration

Cas d'utilisation : Activer un compte

En association avec le choix effectuer pour le cas d'utilisation « Inscription », le cas d'utilisation s'impose pour le compléter. Ainsi, l'administration active le compte d'un utilisateur après son inscription. Autrement, des personnes non-autorisées auront accès à la plate-forme.

Activer un compte
Objectif : activer un compte utilisation
Acteur principal : bibliothécaire
Précondition : être authentifié comme bibliothécaire
Déroulement : <ul style="list-style-type: none"> • Demander la page des utilisateurs. • Le système affiche la page d'utilisateurs • Le bibliothécaire clique sur "activer" pour le compte qu'il veut activer. • Le système envoie un message de succès.
Exception : <ul style="list-style-type: none"> • aucune
Postcondition : le nouveau compte est activé

TAB. 5.8: FT de diagramme de séquence de l'UC Activer un compte

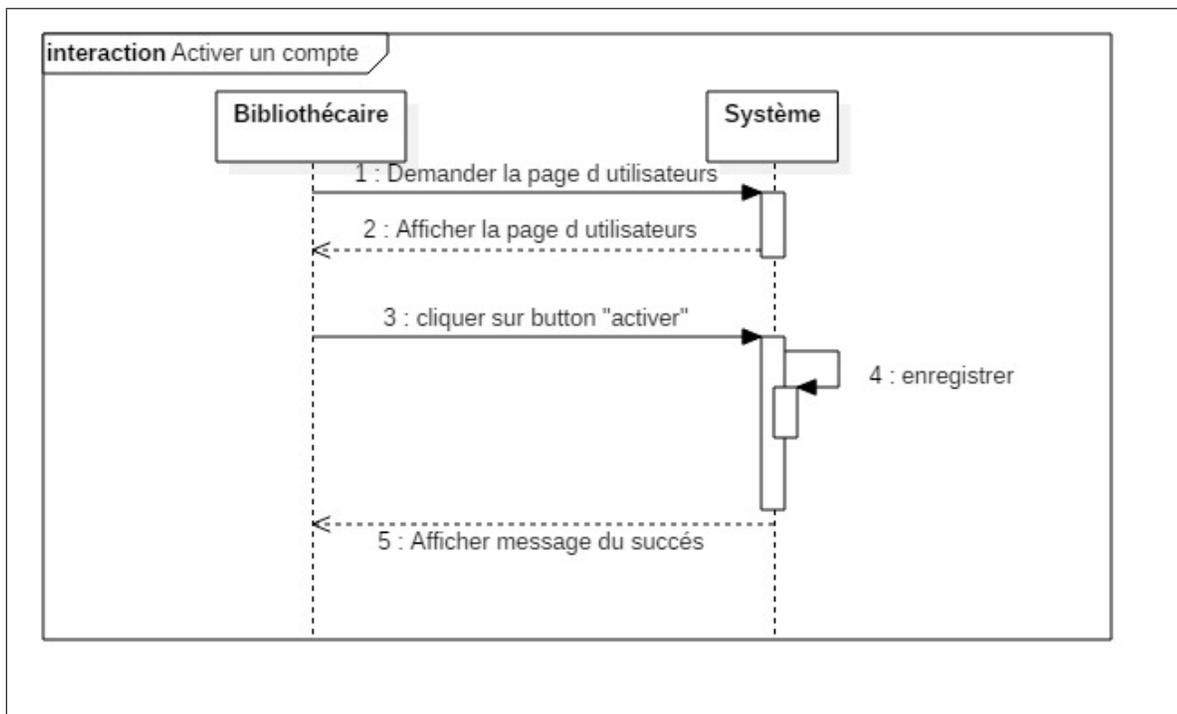


FIG. 5.16: Diagramme de séquence d'activation d'un compte

Cas d'utilisation : Désactiver un compte

Ce cas est commun à tous les systèmes multitilateurs avec un panneau d'administration. Pour différentes raisons, le compte d'un utilisateur peut être désactivé pour lui empêcher d'utiliser la plate-forme. Dans le cas d'une bibliothèque, il est essentiellement utile pour désactiver les comptes des étudiants qui ont terminé leurs études.

Désactiver un compte
Objectif : Désactiver un compte utilisation
Acteur principal : bibliothécaire
Précondition : être authentifié comme bibliothécaire
Déroulement : <ul style="list-style-type: none"> • Demander la page des utilisateurs. • Le système affiche la page d'utilisateurs • Le bibliothécaire clique sur "désactiver" pour le compte qu'il veut désactiver. • Le système envoie un message de succès.
Exception : <ul style="list-style-type: none"> • aucune
Postcondition : le compte choisi est désactivé

TAB. 5.9: FT de diagramme de séquence de l'UC Désactiver un compte

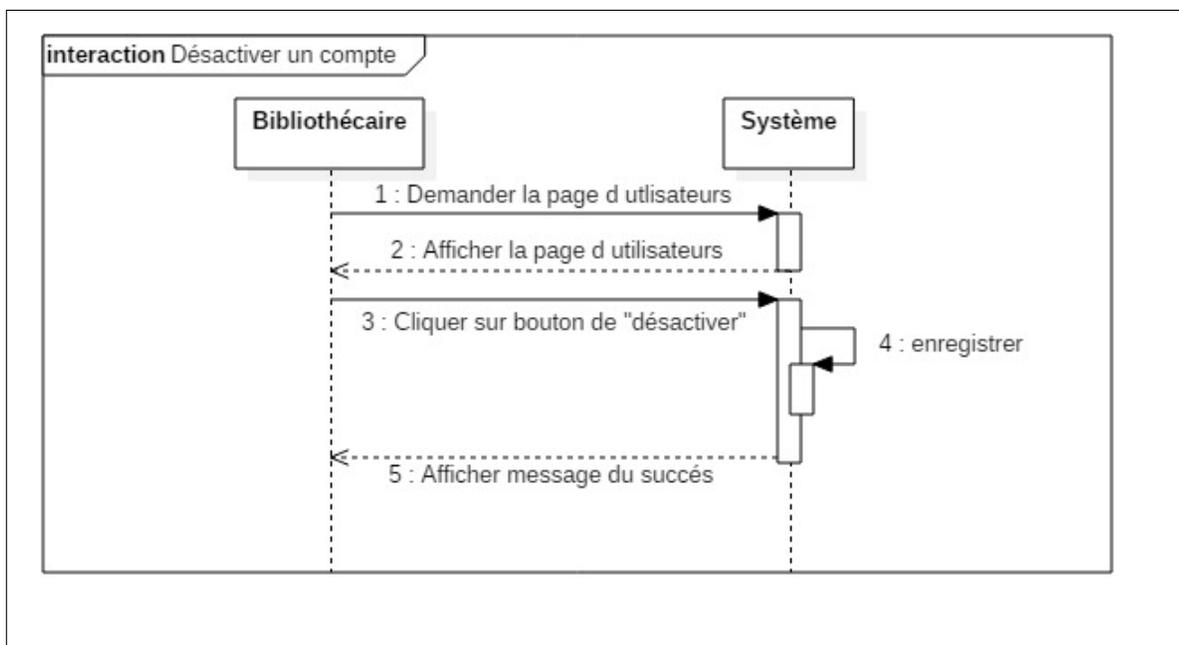


FIG. 5.17: Diagramme de séquence de désactivation d'un compte

Cas d'utilisation : Bloquer un compte

Ce cas d'utilisation est utile lorsqu'il s'agit des pénalités et des sanctions au niveau de la bibliothèque. Un compte est bloqué pour empêcher l'utilisateur de se connecter à la plate-forme pour une durée donnée.

Bloquer un compte
Objectif : Bloquer un compte utilisation
Acteur principal : bibliothécaire
Précondition : être authentifié comme bibliothécaire
Déroulement : <ul style="list-style-type: none"> • Demander la page des utilisateurs. • Le système affiche la page d'utilisateurs. • Le bibliothécaire clique sur "bloquer" pour le compte qu'il veut bloquer. • Le système envoie un message de succès.
Exception : <ul style="list-style-type: none"> • aucune
Postcondition : le compte est bloqué

TAB. 5.10: FT de diagramme de séquence de l'UC Bloquer un compte

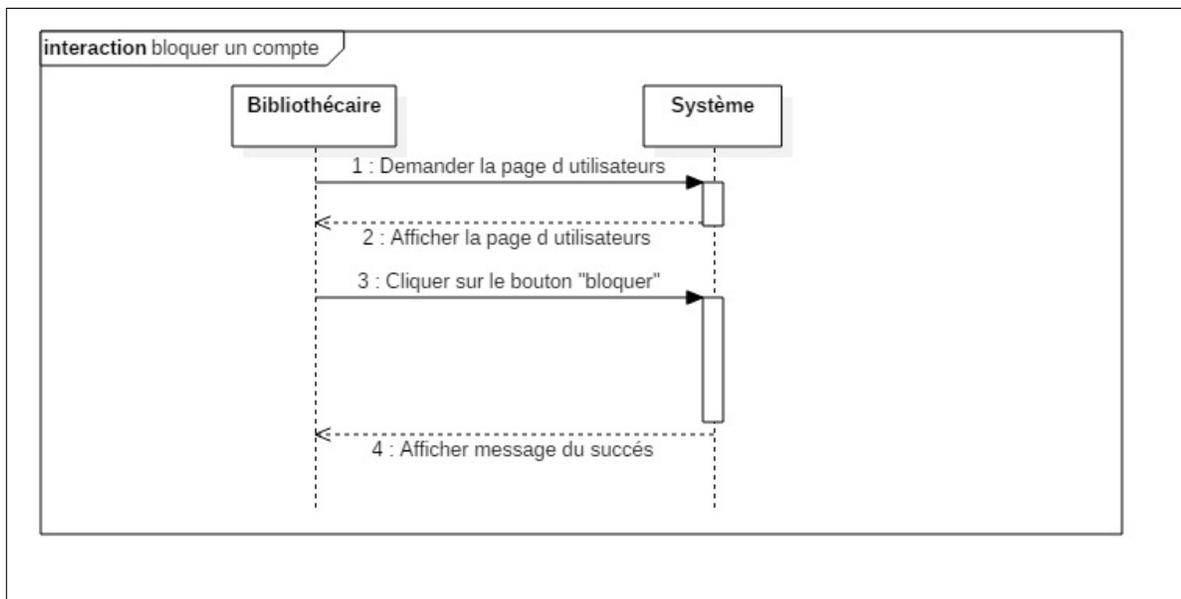


FIG. 5.18: Diagramme de séquence bloquer un compte

A ce niveau aussi, l'homogénéité de la page d'administration est visée. Le système doit garantir ces multiples cas d'utilisation en relation avec la gestion des utilisateurs sur la même interface et d'une manière similaire. L'administrateur se charge aussi de la surveillance des commentaires. En effet, les utilisateurs ont la possibilité de commenter les livres et les articles du blog. L'administrateur peut visualiser tous les commentaires pour supprimer les commentaires inappropriés.

5.4.5.2 Diagramme de classes manipulés par le sous-système Administrateur

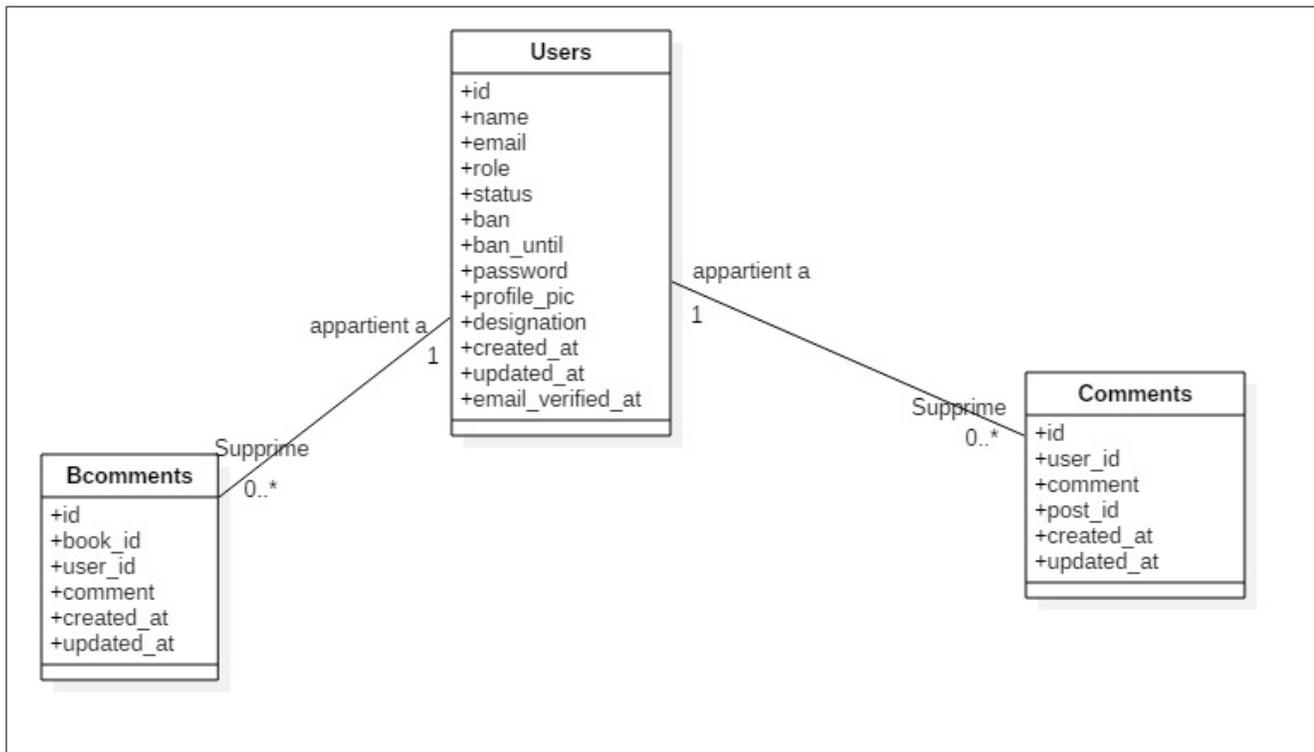


FIG. 5.19: Diagramme de classes d'administration

5.5 Diagramme de paquetage

En prenant en compte tout ce qui a été obtenu dans les phases d'analyse et de conception, nous pouvons structurer les paquets du système comme suit :

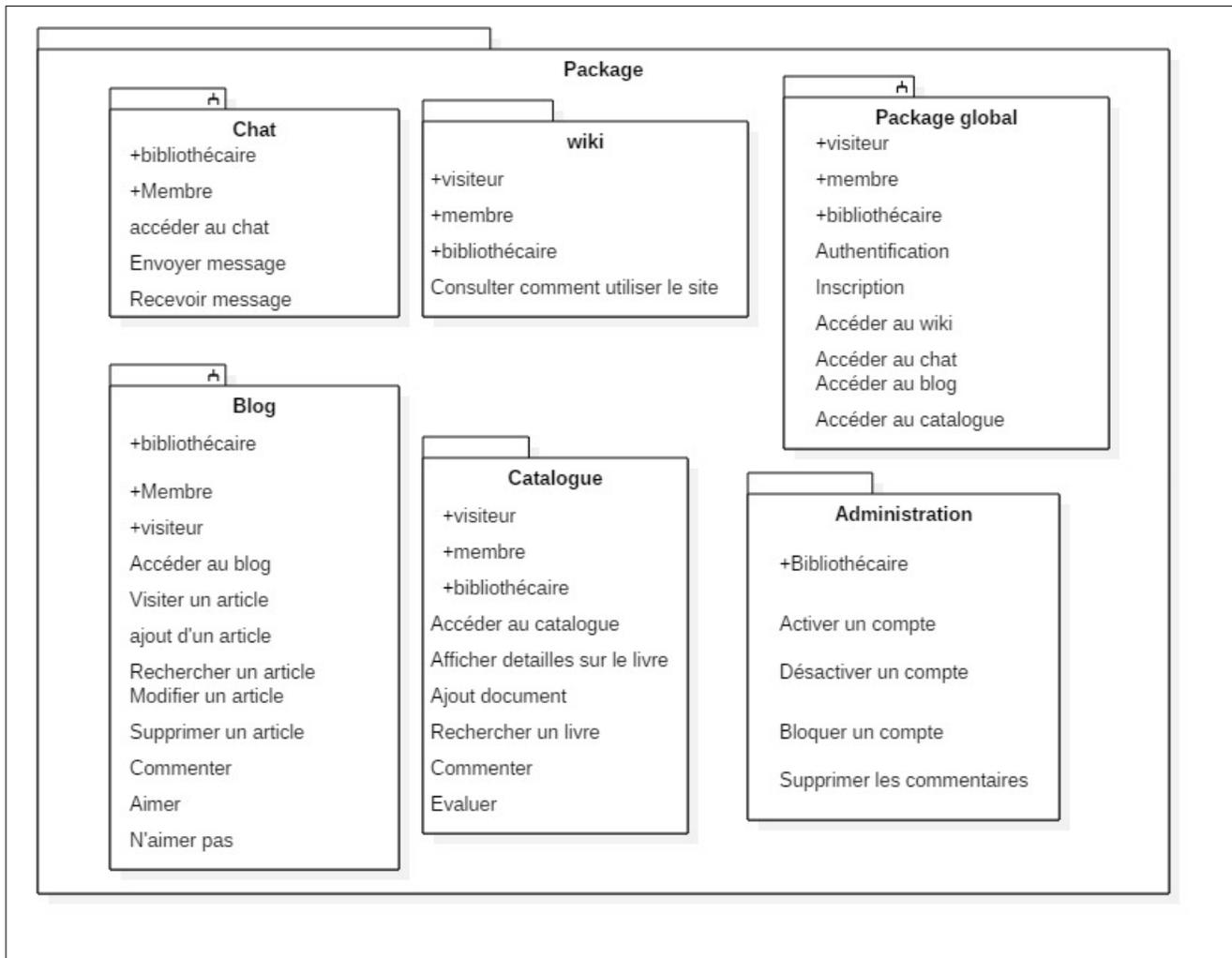


FIG. 5.20: Diagramme de Paquetage

5.6 Diagramme de classes

En parallèle avec la conception des cas d'utilisation, nous avons construit progressivement les différentes classes de notre système. Le diagramme de classes global regroupe toutes les classes proposées et permet de représenter toutes les données manipulées par le système.

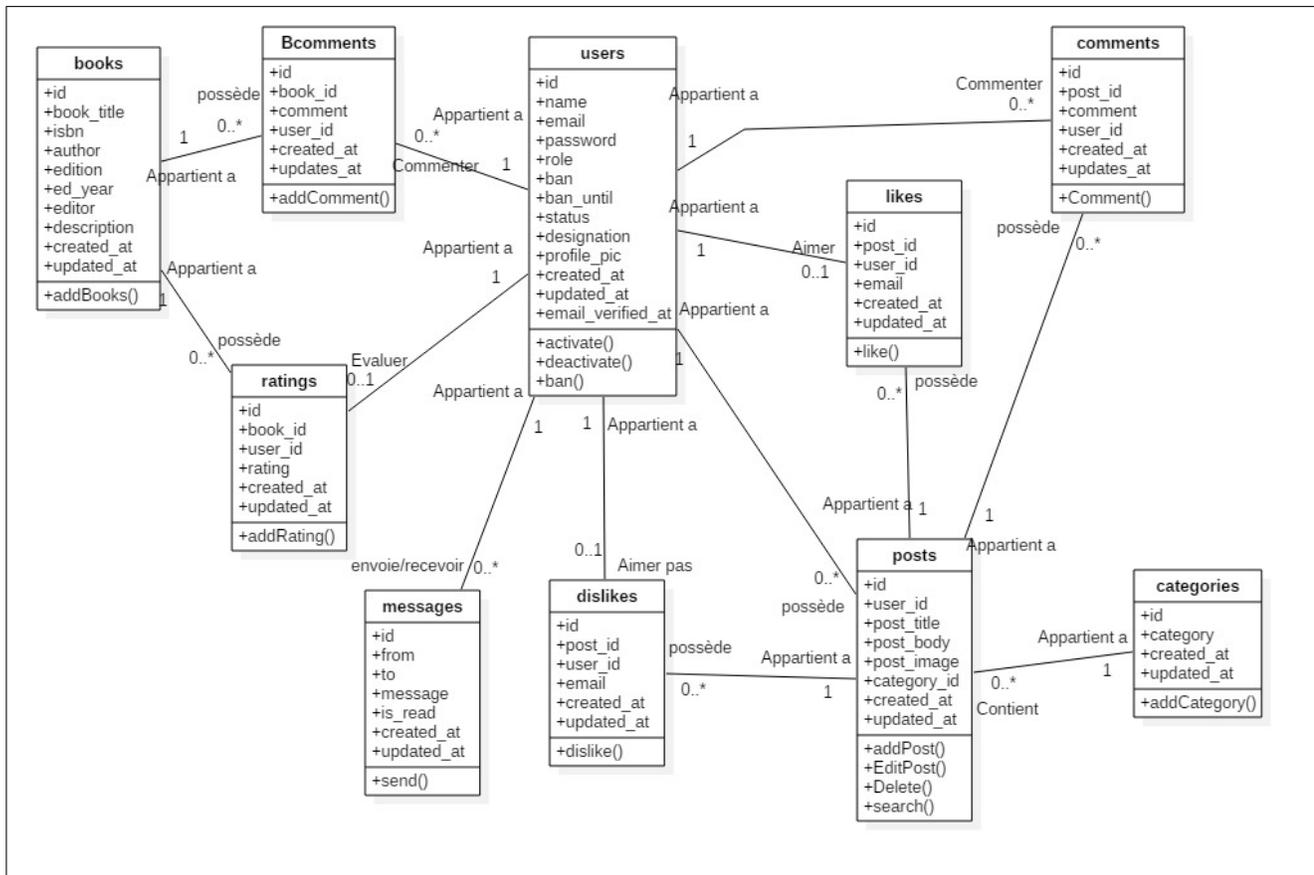


FIG. 5.21: Diagramme de classes global

5.7 Schéma relationnel

Importance du modèle relationnel

Un SGBD est dit relationnel lorsque les informations sont présentées selon le modèle relationnel. Le modèle logique des données utilisé pour décrire une base relationnelle est appelé Schéma relationnel. Le schéma relationnel est une représentation théorique des tables constituant physiquement une base des données ; une relation peut s'écrire en notation fonctionnelle, on souligne sa clé pour la mettre en évidence.

Le modèle relationnel doit refléter le Model Conceptuelle de Donnée issu de l'analyse, et donc les éléments présents dans celui-ci (entités, propriétés, associations) doivent se retrouver dans le modèle relationnel. Pour produire le modèle relationnel à partir du model conceptuelle on devra :

- Chaque entité du Modèle Conceptuelle de Donnée est transformée en table.

- Les propriétés de l'entité deviennent les attributs de la table.
- L'identifiant de l'entité devient clé primaire.

Le schéma relationnel est particulièrement important à construire. En effet, la plupart des technologies web reposent sur des SGBD Relationnels, ainsi, pour pouvoir stocker les différentes données du système, le modèle basé sur les classes dans UML doit être traduit au modèle relationnel supporté et utilisé par les SGBD Relationnels.

Le schéma relationnel du notre système Nous obtenons le schéma relationnel suivant :

Users(id,name,email,role,status,ban,email_verified_at,password,remember_token,created_at,updated_at,ban_until)

Books(id,isbn,book_title,author,editor,edition,image,ed_year,description,created_at,updated_at)

Bcomments (id,user_id,book_id,comment, created_at,updated_at)

Posts (id,user_id,post_title,post_body,post_image,category_id, created_at,updated_at)

Categories(id,category, created_at,updated_at)

Comments(id,user_id,post_id,comment, created_at,updated_at)

Dislike(id,user_id,post_id,email, created_at,updated_at)

Likes(id,user_id,post_id,email, created_at,updated_at)

Ratings(id,user_id,book_id,rating,email, created_at,updated_at)

Profiles (id,user_id,name,designation,profile_pic,created_at,updated_at)

Messages(id,from,to,message,is_read,created_at,updated_at)

5.8 Influence de la pandémie CoVid-19 sur la conception du système

5.8.1 Pandémie CoVid-19

La pandémie de Covid-19 est une pandémie d'une maladie infectieuse émergente, appelée la maladie à coronavirus 2019 ou Covid-19, provoquée par le coronavirus SARS-CoV-2, apparue à Wuhan le 17 novembre 2019, dans la province de Hubei (en Chine centrale), avant

de se propager dans le monde. L'Organisation mondiale de la santé (OMS) alerte dans un premier temps la République populaire de Chine et ses autres États membres, puis prononce l'état d'urgence de santé publique de portée internationale le 30 janvier 2020. Le 11 mars 2020, l'épidémie de Covid-19 est déclarée pandémie par l'OMS, qui demande des mesures de protection essentielles pour prévenir la saturation des services de soins intensifs et renforcer l'hygiène préventive (suppression des contacts physiques, bises et poignées de mains, fin des attroupements et des grandes manifestations ainsi que des déplacements et voyages non indispensables, promotion du lavage des mains, mise en application de quarantaine, etc.). L'Algérie, comme la majorité des pays du monde, a pris au sérieux les alertes de l'OMS et a procédé à appliquer un ensemble de mesures pour faire face à la pandémie. Parmi ces mesures, les études supérieures en présentiel (en face-à-face) ont été suspendues. L'objectif est de :

- Limiter le déplacement des personnes entre des zones géographiques distantes,
- Limiter les assemblés de personnes dans des lieux fermés, sachant qu'un amphi peut assembler des centaines d'étudiants durant les séances du cours,
- Limiter le contact physique entre les étudiants surtout que la plupart des services universitaires sont de nature collective.

Influence sur notre projet : L'effet négatif principal de la pandémie était l'absence du contact avec les bibliothécaires. Par conséquent, nous n'avons pu récupérer d'une manière pertinente et suffisante les détails de deux exigences :

1. Wiki

Choix conceptuels : Le wiki est la technologie que l'on emploie pour faire communiquer un contenu aux utilisateurs du portail. L'objectif de son intégration à notre projet est d'offrir un support de formation pour les nouveaux étudiants. Durant les entretiens accomplis, l'utilisation des supports multimédias et de streaming été encouragée, néanmoins, nous n'avons pas pu effectuer une étude détaillée pour ce besoin. Ainsi, nous avons opté pour les choix suivants :

- Une page de bienvenue qui contient une présentation.
- Une page pour le catalogue qui contient l'espace visiteur et étudiant chacun d'eux comporte une explication et vidéo appropriée.
- Une page pour le blog qui contient l'espace visiteur et étudiant chacun d'eux comporte

une explication et vidéo appropriée.

- Une page pour la messagerie qui contient l'espace visiteur et étudiant chacun d'eux comporte une explication et vidéo appropriée.

Notre avons opté pour un Wiki statique dans un premier temps parce qu'il satisfait tout les besoins en relation avec la formation sur les différentes fonctionnalités du portail.

Diagramme des cas d'utilisation pour le sous-système Wiki

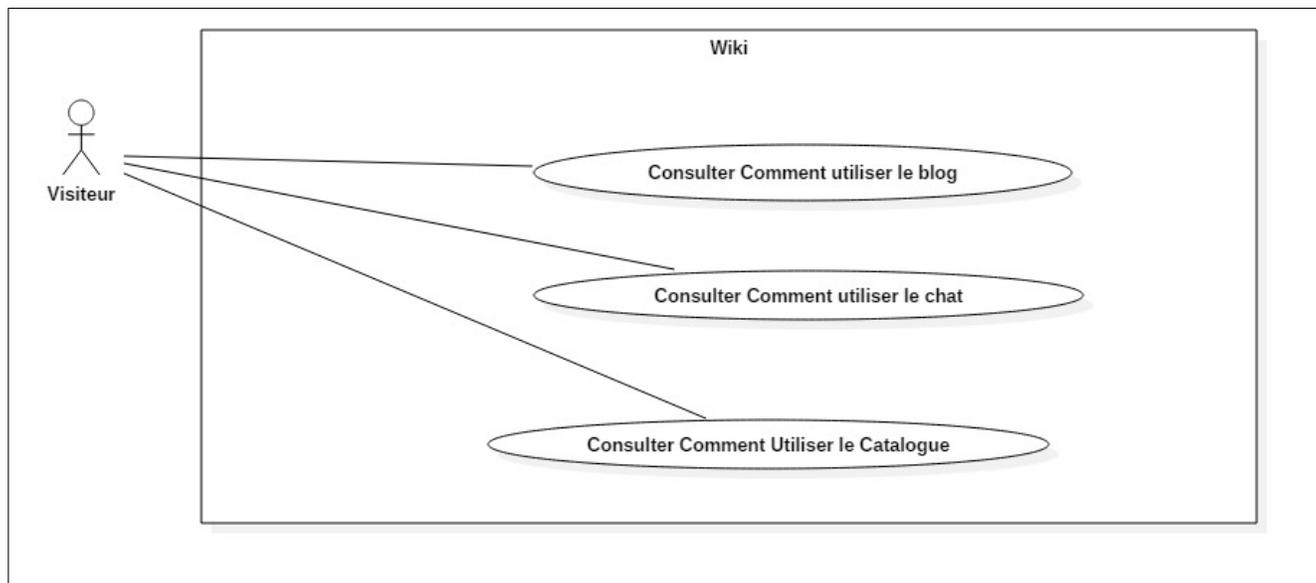


FIG. 5.22: Diagramme des cas d'utilisation du wiki

2. Besoins d'analyse et de reporting : Planifié pour être préparé à la fin de la phase de conception, le reporting est important pour permettre aux bibliothécaires d'avoir une vue globale sur l'activité du nouveau portail. Malheureusement, nous n'avons pas récupéré les besoins exacts en termes de statistiques et d'indicateurs décisionnels désirés par les bibliothécaires. Ainsi, nous avons opté pour les statistiques suivantes :

- Les utilisateurs les plus actifs récemment.
- Les livres les plus consultés et commentés récemment.
- Les articles de blog les plus consultés et commentés récemment.

Malgré ses limitations, le développement d'une page de reporting facilitera une adaptation future et une prise en compte rapide des besoins des bibliothécaires une fois récupérés.

5.9 Conclusion

Dans ce chapitre, nous avons repris les choix conceptuels et les solutions proposés dans la phase d'analyse pour concevoir une solution complète. Le langage de modélisation utilisé était UML et la conception est organisée suivant les différents sous-systèmes à concevoir. L'étape suivante est de traduire cette conception à une solution informatique fonctionnelle. Le chapitre suivant sera dédié à la présentation des technologies utilisées et la solution développée.