



ECOLE MAROCAINE DES
SCIENCES DE L'INGENIEUR
Membre de
HONORIS UNITED UNIVERSITIES

Mini Projet

Année : 2023/2024

Réalisation du cahier des charges de
l'application *Mendeley*

Ingénierie Informatique et Réseaux

Réalisé par :

ZOUIZZA KHALIL

OUBAALLA MEHDI

Chapitre 1 : Contexte générale du projet

1. Problématique :

La gestion des références bibliographiques est une tâche essentielle pour les chercheurs, les étudiants et d'autres professionnels de la recherche. Cependant, de nombreux utilisateurs rencontrent des difficultés pour organiser, stocker, et récupérer efficacement leurs références bibliographiques. Les outils existants présentent souvent des limitations en termes de convivialité, de collaboration, et de gestion de différents formats de références. Par conséquent, il est impératif de développer une solution complète et conviviale pour résoudre ces problèmes.

2. Objectifs du projet :

Les objectifs du projet Mendeley sont les suivants :

- Développer une application web et des applications mobiles pour Android et iOS permettant la gestion complète des références bibliographiques.
- Intégrer une base de données robuste pour stocker les informations sur les utilisateurs, les références et les métadonnées associées.
- Offrir une interface utilisateur conviviale pour faciliter l'ajout, la gestion et la recherche de références.
- Fournir des fonctionnalités de synchronisation en ligne pour un accès aux données depuis différents appareils.
- Permettre l'importation et l'exportation de références à partir de formats courants.
- Générer automatiquement des citations conformes aux styles académiques et la possibilité de personnaliser ces styles.
- Permettre la collaboration entre utilisateurs sur des références et des dossiers de recherche.

3. Utilisateurs Cibles :

- Chercheurs.
- Étudiants.
- Bibliothécaires.
- Professeurs.

4. Conclusion :

Le projet Mendeley vise à répondre à la problématique de la gestion de références bibliographiques en offrant une solution complète, conviviale et collaborative. En tenant compte des besoins des utilisateurs, il s'efforce de fournir une application de gestion de références de haute qualité, qui permettra aux chercheurs, aux étudiants, aux bibliothécaires et à d'autres professionnels de la recherche de gérer efficacement leurs références bibliographiques. Les chapitres suivants de ce cahier des charges détailleront les exigences techniques, fonctionnelles et de conception pour atteindre ces objectifs.

Chapitre 2 : Étude Préliminaire

1. Introduction :

Ce chapitre examine en profondeur les besoins fonctionnels et non-fonctionnels essentiels de l'application Mendeley. Il s'agit d'une phase clé dans la définition des spécifications du projet, couvrant les attentes des utilisateurs, la performance, la sécurité, l'expérience utilisateur et l'évolutivité. Cette étude préliminaire jettera les bases du développement de Mendeley pour en faire une solution complète et de haute qualité.

2. Besoins Fonctionnels :

2.1. Gestion des Références :

- Ajouter une référence bibliographique avec des champs pour le titre, les auteurs, la source, la date, etc.
- Organiser les références dans des dossiers personnalisables.
- Permettre la recherche et le filtrage des références par différents critères.
- Autoriser la modification et la suppression des références existantes.

2.2. Génération de Citations :

- Générer automatiquement des citations et des bibliographies conformes à des styles académiques variés, tels que APA, MLA, Chicago, etc.
- Personnaliser les styles de citation selon les préférences de l'utilisateur.

2.3. Collaboration :

- Partager des références et des dossiers avec d'autres utilisateurs pour permettre la collaboration.
- Permettre la synchronisation en temps réel des modifications apportées aux références partagées.

2.4. Synchronisation Multiplateforme :

- Offrir un accès aux données de références depuis différents appareils (navigateurs web, applications mobiles Android et iOS).
- Assurer une synchronisation en ligne transparente des données entre les plateformes.

2.5. Gestion de Fichiers PDF :

- Associer des fichiers PDF aux références pour permettre un accès rapide aux documents source.
- Permettre l'annotation et la prise de notes sur les fichiers PDF.

3. Besoins Non-fonctionnels :

3.1. Performance :

- L'application doit offrir des temps de réponse rapides pour les opérations courantes.
- Les performances du système doivent être maintenues même en cas d'augmentation de la charge.

3.2. Sécurité :

- Les données des utilisateurs, y compris les références et les fichiers PDF, doivent être stockées de manière sécurisée.
- L'authentification des utilisateurs doit être robuste et sécurisée.

3.3. Évolutivité :

- L'application doit être conçue pour évoluer avec l'ajout de nouvelles fonctionnalités ou une augmentation du nombre d'utilisateurs.

3.4. Expérience Utilisateur :

- L'interface utilisateur doit être intuitive, conviviale et accessible sur différentes plateformes.
- L'application doit offrir une expérience fluide et agréable à l'utilisateur.

3.5. Intégration :

- L'application doit pouvoir s'intégrer avec des bases de données académiques et d'autres sources externes pour l'import de références.
- L'intégration avec des services de stockage en ligne, tels qu'AWS S3, doit être prise en charge pour la gestion des fichiers PDF.

4. Conclusion :

L'étude préliminaire approfondie de ce cahier des charges a permis d'identifier de manière claire et détaillée les besoins fonctionnels et non-fonctionnels essentiels pour le développement de Mendeley. Ces exigences serviront de base solide pour la création d'une application complète et performante dans le domaine de la gestion de références bibliographiques. Mendeley s'engage à simplifier la vie des chercheurs et à contribuer à la recherche académique en fournissant une solution de haute qualité.

Chapitre 3 : Analyse et conception :

1. Démarche et conception :

1.1. Introduction :

Ce chapitre expose la démarche et la conception de l'application, en mettant en avant les étapes clés du processus de développement et les choix de conception effectués pour assurer une application fonctionnelle et efficace.

1.2. Présentation UML :



Le langage UML a été pensé pour être un langage de modélisation visuelle commun, et riche sémantiquement et syntaxiquement. Il est destiné à l'architecture, la conception et la mise en œuvre de systèmes logiciels complexes par leur structure aussi bien que leur comportement. L'UML a des applications qui vont au-delà du développement logiciel, notamment pour les flux de processus dans

l'industrie. Il ressemble aux plans utilisés dans d'autres domaines et se compose de différents types de diagrammes. Dans l'ensemble, les diagrammes UML décrivent la limite, la structure et le comportement du système et des objets qui s'y trouvent. L'UML n'est pas un langage de programmation, mais il existe des outils qui peuvent être utilisés pour générer du code en plusieurs langages à partir de diagrammes UML. L'UML a une relation directe avec l'analyse et la conception orientées objet.

2. Le choix UML :

Pour une modélisation cohérente qui fournit une clarté permettant la réalisation d'une application qui répond parfaitement au besoin, sans négliger aucuns aspects, on fait appel à UML sa richesse en termes de possibilités, telles que :

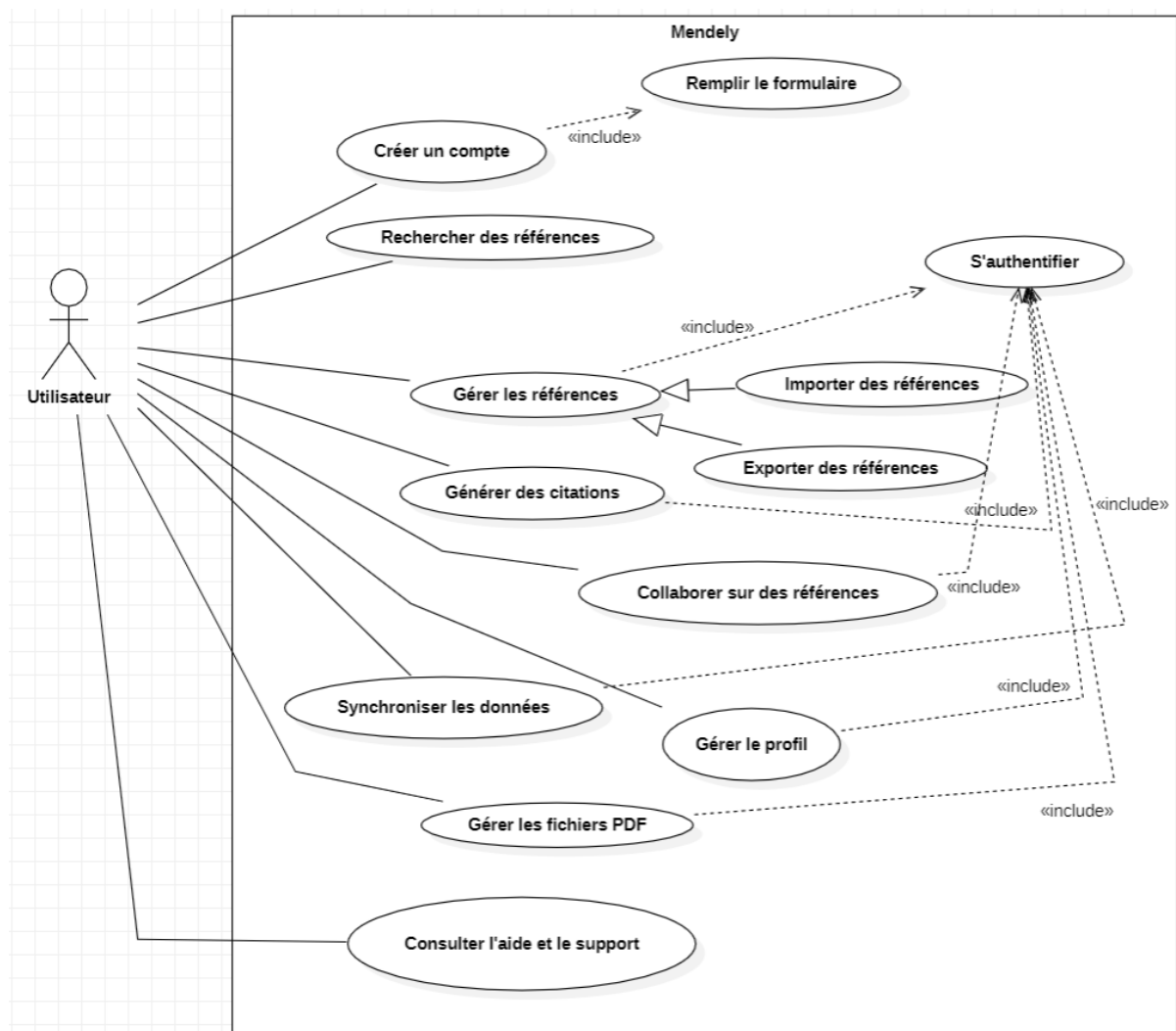
- Obtenir une modélisation de très haut niveau indépendante des langages et des environnements.
- Faire collaborer des participants de tous horizons autour d'un même document de synthèse.

- Exprimer dans un seul modèle tous les aspects statiques, dynamiques, spécifications, etc...
- Générer automatiquement la partie logicielle d'un système.

2.1. Les diagrammes utilisés :

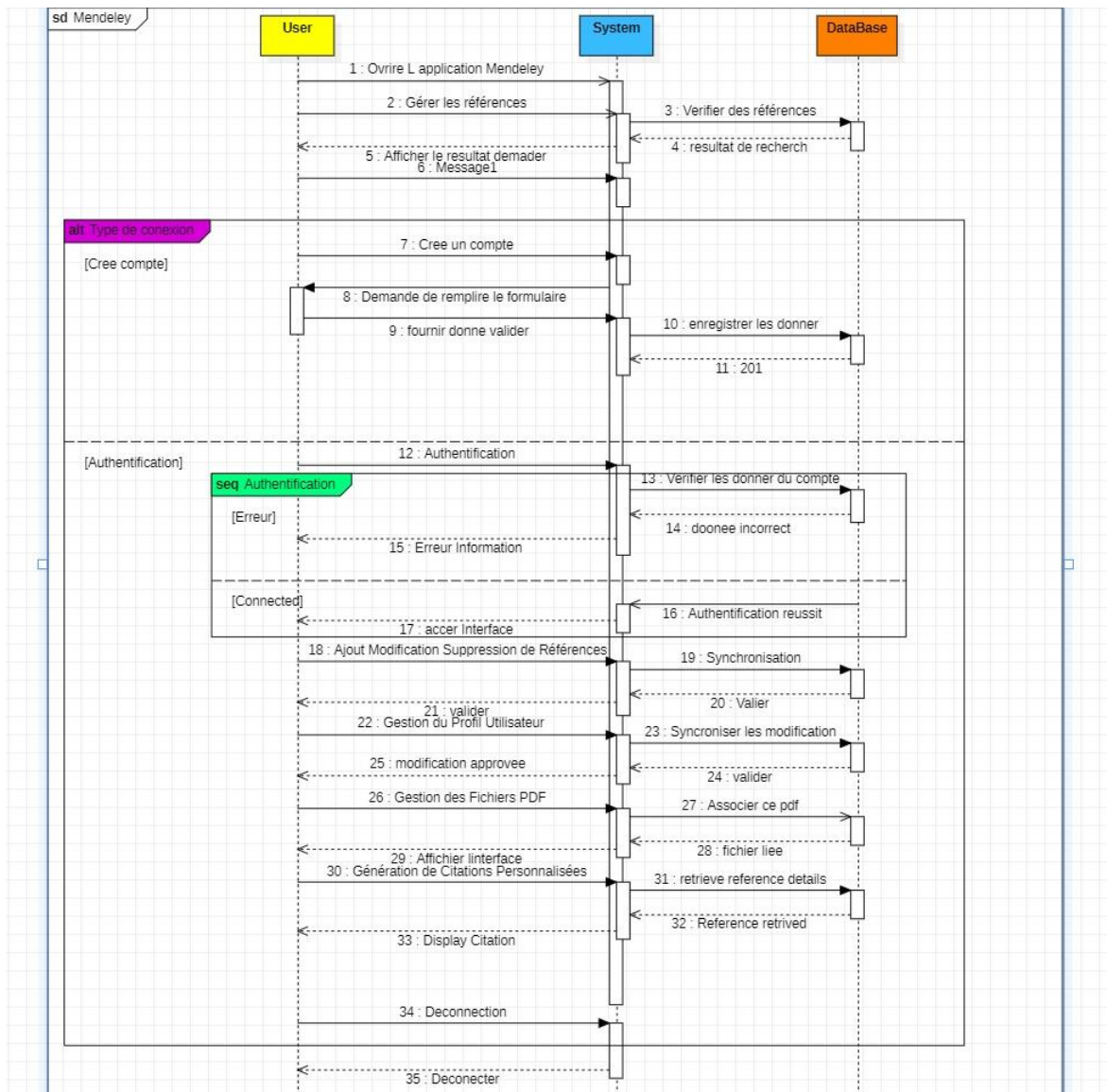
2.1.1. Diagramme de cas d'utilisations :

Diagrammes de cas d'utilisation sont des diagrammes UML utilisés pour une représentation du comportement fonctionnel d'un système logiciel. Un cas d'utilisation définit une manière d'utiliser un système qui a une valeur ou une utilité pour les acteurs impliqués. Le cas d'utilisation correspond à un ensemble d'actions réalisées par le système en interaction avec les acteurs en vue d'une finalité. L'ensemble des cas d'utilisation permet ainsi de décrire les exigences fonctionnelles d'un système en adoptant le point de vue et le langage de l'utilisateur final.



2.1.2. Diagramme de séquence :

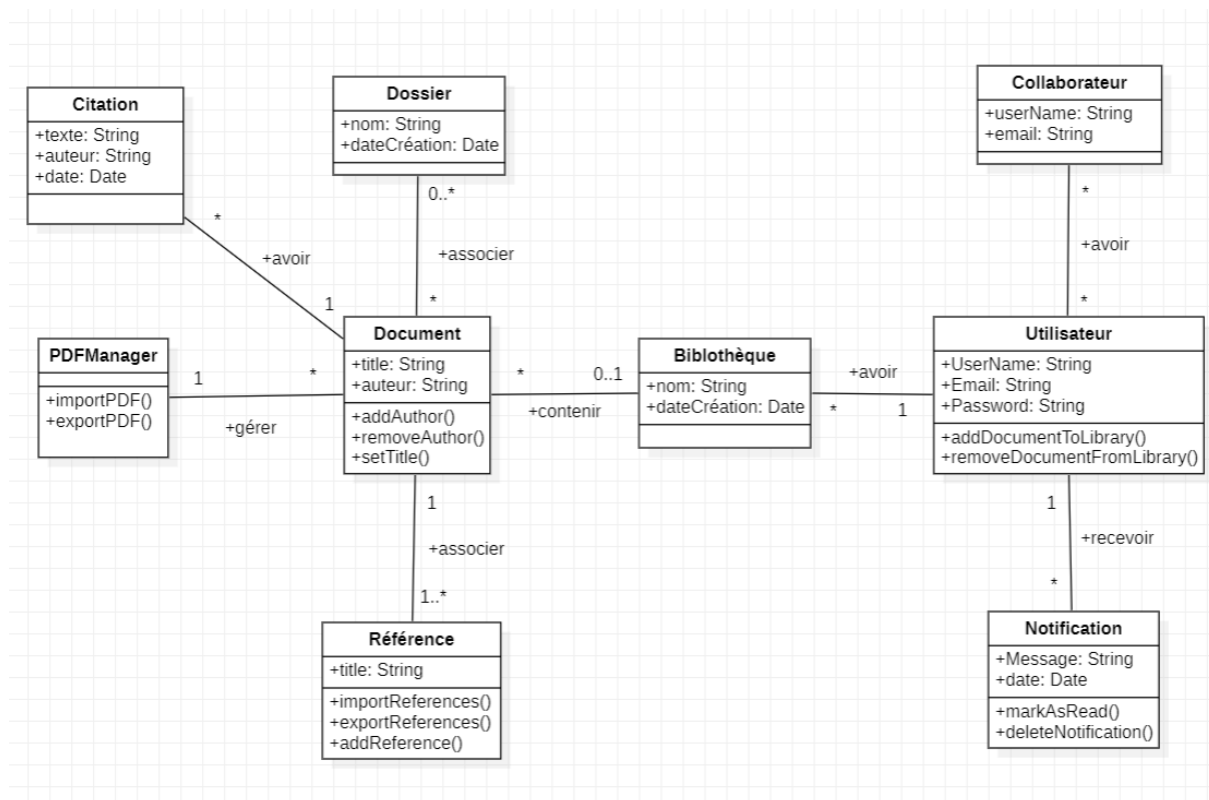
Le diagramme de séquence permet de montrer les interactions d'objets dans le cadre d'un scénario d'un diagramme des cas d'utilisation. Dans un souci de simplification, on représente l'acteur principal à gauche du diagramme, et les acteurs secondaires éventuels à droite du système. Le but est de décrire comment se déroulent les interactions entre les acteurs ou objets.



2.1.3. Diagramme de classe :

C'est un schéma utilisé pour présenter les classes et les interfaces des systèmes ainsi que leurs relations. Ce diagramme fait partie de la partie statique d'UML, ne s'intéressant pas aux 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.

Le diagramme de classes associé à notre système est représenté ci-dessus, il représente d'une manière générale les entités utilisées à l'application.



3. Conclusion :

Grâce à la phase d'analyse et de conception, nous avons pu formaliser les étapes initiales du développement de notre système et obtenir une compréhension claire des différentes interactions entre l'utilisateur et le système, afin de répondre aux besoins précédemment identifiés. Dans la suite de ce document, nous présenterons les différents outils techniques utilisés pour la mise en œuvre de ce projet.