

FACULTE
des SCIENCES ET TECHNIQUES MARRAKECH

INGENIERIE EN RESEAUX ET SYSTEMES
D'INFORMATION

**Document d'architecture technique du
projet UpStudy**

Réalisé par :

AIT ELFATMI Najah

ELHYATE Dina

BEN-YADI Ouijdane

Encadré par:

M. Bencharef

M. Kaloun

1. Introduction

Ce document d'architecture technique vise à décrire les composantes clés, les choix technologiques et l'organisation du projet d'un site web de bibliothèque en ligne. Ce site permet aux utilisateurs de rechercher, consulter, et télécharger des articles, tout en fournissant à l'administrateur des droits complets de gestion.

2. Fonctionnalités Principales

2.1 Authentification et gestion des utilisateurs

- **Inscription** : Les utilisateurs peuvent créer un compte avec un email unique et un mot de passe.
- **Connexion** : Accès au compte utilisateur avec vérification des identifiants.
- **Gestion des sessions** : Maintenance d'une session sécurisée.

2.2 Gestion des rôles et des permissions

- **Utilisateur** :
 - Accès à ses propres activités.
 - Consultation des articles disponibles.
 - Téléchargement des articles.
- **Administrateur** :
 - Gestion complète des utilisateurs et des articles.
 - Ajout, modification et suppression des articles.
 - Gestion des rôles des utilisateurs.

2.3 Gestion des articles

- Affichage et recherche des articles.
- Filtrage par catégories, auteurs, ou mots-clés.
- Téléchargement des fichiers PDF.

3. Architecture Logicielle

3.1 Modèle MVC (Model-View-Controller)

L'architecture repose sur le modèle MVC pour séparer les responsabilités :

- **Model** : Gestion des données et interaction avec la base de données MySQL.
- **View** : Interface utilisateur (UI) construite avec ReactJS.
- **Controller** : Traitement des requêtes et logique d'application, implémentée avec Node.js.

3.2 Composants Principaux

Frontend :

- Framework : ReactJS
- Bibliothèque UI : Material-UI

Backend :

- Framework : Node.js

Base de données :

- SGBD : MySQL

3.3 Schéma de la Base de Données

Tables Principales :

- **Users :**
 - id (PK), name, email (unique), password, role (admin/user).
- **Articles :**
 - id (PK), title, author, category, file_path.
- **Logs :**
 - id (PK), user_id (FK), action, timestamp.

4. Architecture Technique

4.1 Schéma de l'Architecture

1. **Client** (ReactJS) \Rightarrow **API REST** (Node.js) \Rightarrow **Base de données** (MySQL).
2. **Authentification.**
3. **Gestion des fichiers .**

5. Sécurité

- **Communication :** HTTPS pour les échanges entre client et serveur.
- **Autorisation :** Middleware pour vérifier les rôles et permissions.

6. Outils et Technologies

- **Versionnage :** Git et GitHub.
- **Intégration continue :** GitHub Actions.
- **Tests :** Test avec Selenium (frontend) et les test unitaires avec Jenkins(backend).
- **Monitoring :** Postman pour le débogage des API .

7. Plan de Mise en Œuvre

7.1 Étapes de Développement

1. Configuration de l'environnement de développement.
2. Mise en place de la base de données.
3. Implémentation du backend : Authentification, gestion des utilisateurs et des articles.
4. Développement du frontend : Pages d'authentification, tableau de bord et gestion des articles.
5. Intégration frontend-backend.
6. Tests unitaires et d'intégration.
7. Déploiement et tests en production.

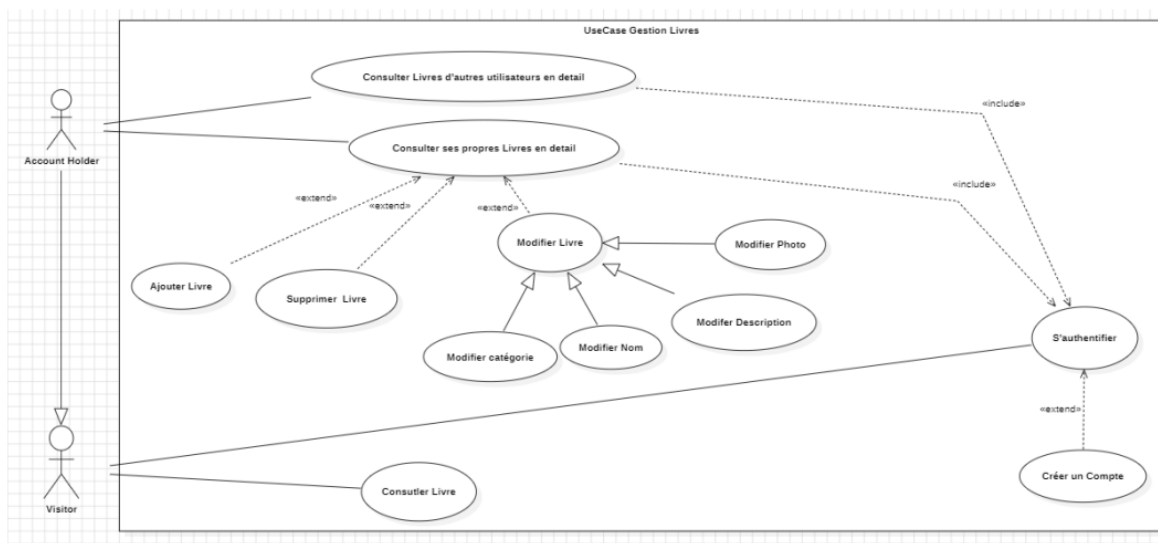
7.2 Chronogramme

Étape	Durée estimée
Conception et planification	1 semaine
Développement backend	2 semaines
Développement frontend	3 semaines
Tests et débogage	1 semaine
Déploiement	1 semaine

8. Diagrammes

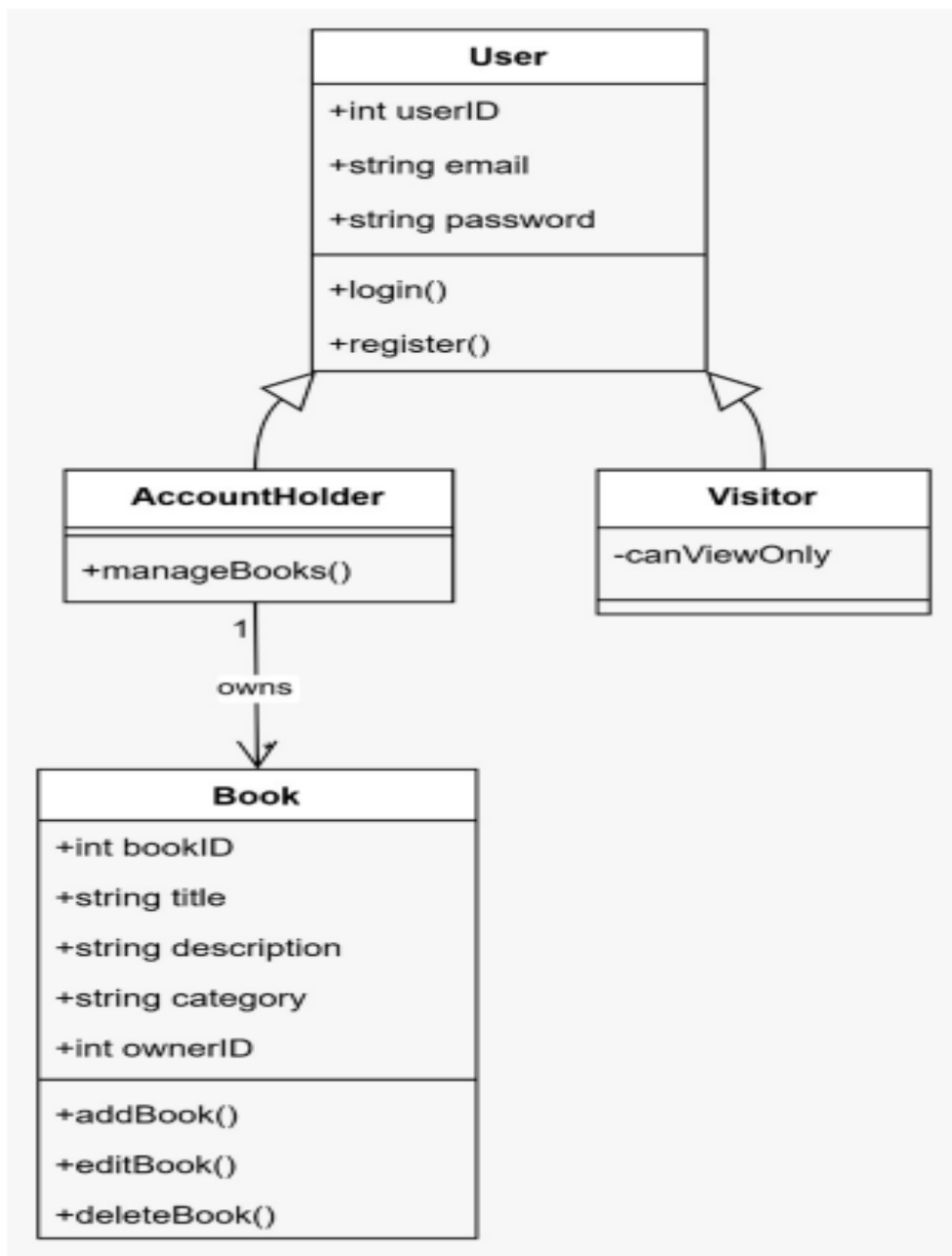
8.1 Diagramme de Cas d'Utilisation

Ce diagramme illustre les interactions principales entre les acteurs (Utilisateur, Administrateur) et le système (Site de Bibliothèque).



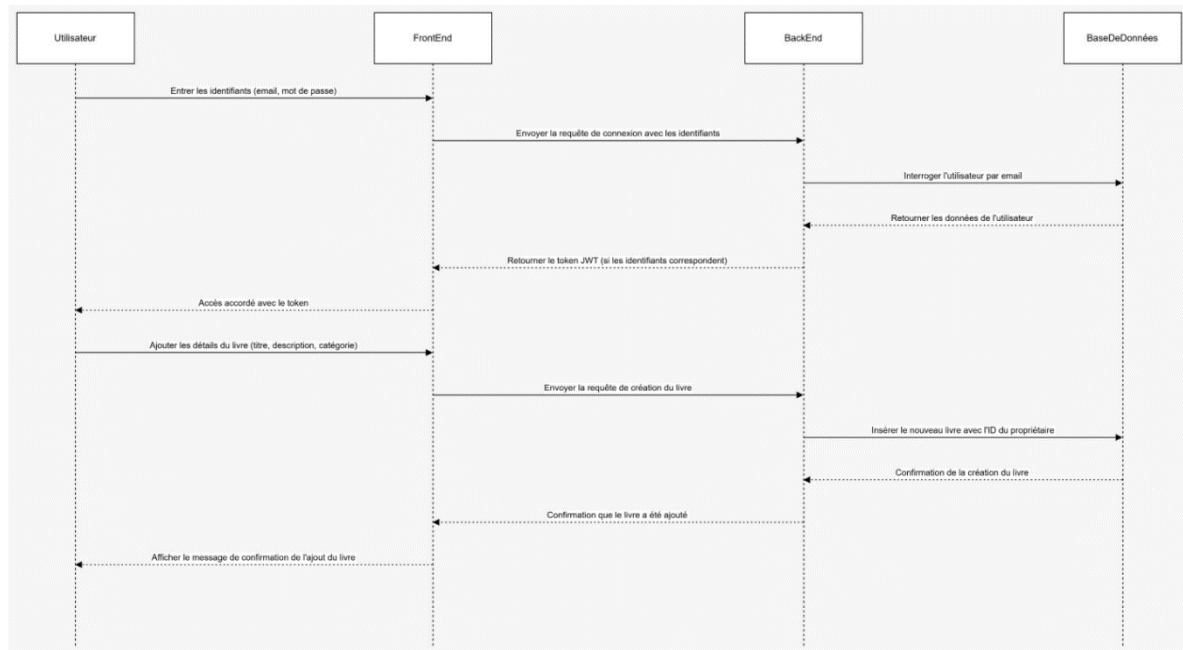
8.2 Diagramme de Classe

Le diagramme de classe présente les relations entre les entités principales comme Users, Articles, et Logs.



8.3 Diagramme de Séquence

Ce diagramme montre les échanges entre les composants pour une fonctionnalité clé, comme l'authentification ou la gestion d'un article.



9. Spécifications Fonctionnelles

9.1 Authentification

- Inscription :
 - Validation de l'email.
- Connexion :
 - Utilisation de JWT pour la gestion des sessions.
 - Vérification des identifiants avec la base de données.

9.2 Gestion des Livres

- **Ajout de livre :**
 - Renseignement des détails comme le titre, la description, et la catégorie.
- **Modification de livre :**
 - Mise à jour des détails des livres créés par l'utilisateur.
- **Suppression de livre :**
 - Suppression des livres ajoutés par l'utilisateur.

9.3 Visualisation

- Consultation des livres publiés par d'autres utilisateurs.
- Les utilisateurs authentifiés peuvent ajouter

10. Conclusion

Cette architecture fournit une base robuste et scalable pour le développement d'un site de bibliothèque en ligne. Avec une séparation claire des responsabilités, un choix technologique moderne et des mécanismes de sécurité appropriés, ce projet garantit une expérience utilisateur fluide et une gestion efficace des ressources par les administrateurs.