

# Cahier des Charges

## Conception d'une Bibliothèque Numérique

Dr Riadh BOUSLIMI

**Matière : Base de données évoluée**

### 1 Introduction

Ce projet consiste à développer une bibliothèque numérique permettant la gestion complète des livres ainsi que la gestion des utilisateurs. Le système doit inclure des fonctionnalités de base pour ajouter, consulter, modifier et supprimer des livres, gérer les emprunts et les réservations, et permettre aux étudiants et enseignants de réserver ou emprunter des ouvrages. Le projet doit être réalisé en 20 jours, avec une possibilité de travail en binôme.

### 2 Objectifs du projet

L'objectif principal est de fournir un système simple mais efficace permettant la gestion complète des livres, des utilisateurs, des réservations et des emprunts dans une bibliothèque numérique. Ce système doit être intuitif, sécurisé et facilement maintenable.

Les objectifs spécifiques incluent :

- Une gestion complète des livres (ajout, suppression, modification, consultation).
- Une gestion des utilisateurs avec différents rôles (administrateur, étudiant, enseignant).
- La possibilité pour les utilisateurs de réserver et emprunter des livres.
- Un système de recherche et de filtrage des livres.
- La production d'un livrable sous forme de vidéo et d'un document PDF expliquant la manipulation de la base de données.

### 3 Fonctionnalités requises

#### 3.1 Gestion des livres

Le système doit permettre les actions suivantes pour les administrateurs :

- **Ajouter un livre** : Chaque livre doit inclure des informations comme le titre, l'auteur, le genre, l'année de publication, une description, l'ISBN, et le statut (disponible, réservé, emprunté).

- **Modifier un livre** : Les administrateurs doivent pouvoir mettre à jour les informations d'un livre.
- **Supprimer un livre** : Un livre peut être retiré de la base de données par les administrateurs.
- **Consulter la liste des livres** : Les utilisateurs doivent pouvoir afficher la liste complète des livres disponibles.

## 3.2 Gestion des utilisateurs

Le système doit gérer les différents types d'utilisateurs :

- **Administrateurs** : Gestion des livres, des utilisateurs, des réservations et des emprunts.
- **Étudiants/Enseignants** : Les étudiants et enseignants peuvent réserver et emprunter des livres.

## 3.3 Réservation et emprunt de livres

- **Réserver un livre** : Les étudiants et enseignants doivent pouvoir réserver des livres disponibles. Un livre réservé devient inaccessible à d'autres utilisateurs jusqu'à la fin de la période de réservation.
- **Emprunter un livre** : Si le livre est disponible, il peut être emprunté par un étudiant ou un enseignant. La durée de l'emprunt peut être définie.
- **Retourner un livre** : Une fois la période d'emprunt terminée, le livre doit être retourné pour devenir de nouveau disponible.

## 3.4 Recherche et filtrage des livres

Le système doit permettre aux utilisateurs de rechercher des livres selon différents critères :

- Titre
- Auteur
- Genre
- Année de publication
- ISBN
- Statut (disponible, emprunté, réservé)

## 3.5 Sécurité et authentification

- Les utilisateurs doivent s'authentifier avec un nom d'utilisateur et un mot de passe pour accéder au système.
- Les rôles (administrateur, étudiant, enseignant) doivent être définis pour restreindre ou accorder des permissions spécifiques.

## 4 Spécifications techniques

### 4.1 Technologies

- **Base de données** : Utilisation d'une base de données NoSQL orientée document (ex : MongoDB) pour stocker les informations sur les livres, utilisateurs, réservations et emprunts.
- **Langages** : Python pour le développement back-end, et HTML/CSS/JavaScript pour le front-end.
- **Interface utilisateur** : Application web avec une interface intuitive et responsive.
- **API REST ou GraphQL** : Pour gérer les interactions entre l'interface utilisateur et la base de données.

### 4.2 Architecture

- **Architecture client-serveur** : Le système doit être basé sur une architecture client-serveur, où les utilisateurs peuvent interagir avec la base de données via une API REST ou GraphQL.
- **Déploiement** : Le projet peut être déployé localement ou sur un serveur pour les tests.

## 5 Contraintes

- **Délai** : Le projet doit être réalisé en 20 jours.
- **Travail en binôme** : Le projet peut être réalisé par deux étudiants.
- **Sécurité** : Les données doivent être sécurisées, avec une authentification forte et des autorisations d'accès basées sur les rôles.
- **Livrables** :
  - **Vidéo démonstrative** : Chaque étudiant doit uploader une vidéo de démonstration expliquant le fonctionnement du projet, avec la photo et la voix de l'étudiant, sur Google Drive. Un lien de partage doit être fourni.
  - **Document PDF** : Un document PDF détaillant la manipulation de la base de données doit être fourni, incluant une description des opérations CRUD et des fonctionnalités de réservation et emprunt.

## 6 Planning

### 6.1 Semaine 1 (Jour 1 à 5)

- Jour 1-2 : Conception du modèle de données (livres, utilisateurs, réservations, emprunts) et mise en place de la base de données.
- Jour 3-5 : Implémentation des fonctionnalités de gestion des livres (ajout, modification, suppression, consultation).

## 6.2 Semaine 2 (Jour 6 à 10)

- Jour 6-7 : Mise en place du système d'authentification et de gestion des utilisateurs.
- Jour 8-10 : Intégration des fonctionnalités de réservation, emprunt et recherche de livres.

## 6.3 Semaine 3 (Jour 11 à 15)

- Jour 11-12 : Tests des fonctionnalités de base et correction des bugs éventuels.
- Jour 13-15 : Finalisation des fonctionnalités avancées (exportation, notifications, etc.).

## 6.4 Semaine 4 (Jour 16 à 20)

- Jour 16-17 : Tests finaux et validation du projet.
- Jour 18-19 : Enregistrement de la vidéo de démonstration expliquant les fonctionnalités du projet.
- Jour 20 : Livraison du projet (lien vers la vidéo et document PDF).

# 7 Livrables attendus

- **Base de données fonctionnelle** : La base de données doit être correctement configurée et opérationnelle, avec toutes les informations sur les livres, les utilisateurs, les réservations et les emprunts.
- **Interface utilisateur** : Une interface web intuitive et responsive.
- **Vidéo démonstrative** : Une vidéo uploadée sur Google Drive expliquant le fonctionnement du projet, avec la photo et la voix de l'étudiant.
- **Document PDF** : Un document expliquant en détail la manipulation de la base de données et les fonctionnalités (opérations CRUD, réservation, emprunt).

# 8 Équipe et rôles

- **Développeur principal** : Responsable du développement back-end et de la gestion de la base de données.
- **Développeur front-end** : Responsable de l'interface utilisateur (si nécessaire).
- **Testeur (optionnel)** : Chargé de la validation des fonctionnalités et des tests unitaires.