

Travail à rendre

Énoncé : Gestion d'une Bibliothèque

Objectif : Créer une application web pour gérer une bibliothèque en utilisant **Express**, **Mongoose**, **EJS**, **dotenv**. L'application doit permettre de réaliser des **opérations CRUD** pour gérer des livres. Ajoutez un peu de **CSS** pour rendre l'interface plus agréable.

Fonctionnalités demandées :

1. **Afficher la liste des livres** :
 - Chaque livre doit afficher : le titre, l'auteur, et l'année de publication.
 - Un bouton pour voir les détails du livre.
 - Un bouton pour supprimer un livre.
 2. **Créer un livre** :
 - Un formulaire pour ajouter un livre avec les champs suivants :
 - Titre (obligatoire, au moins 3 caractères)
 - Auteur (obligatoire, au moins 3 caractères)
 - Année de publication (obligatoire, ne doit pas dépasser l'année actuelle)
 - Résumé (facultatif, maximum 500 caractères)
 3. **Modifier un livre** :
 - Un formulaire pré-rempli pour modifier les informations d'un livre existant.
 4. **Afficher les détails d'un livre** :
 - Une page qui affiche toutes les informations du livre.
 5. **Supprimer un livre** :
 - Un bouton pour supprimer un livre directement depuis la liste.
-

Contraintes techniques :

1. **Backend** :
 - Utiliser **Express** pour gérer les routes de l'application.
 - Utiliser **Mongoose** comme ORM pour interagir avec une base de données MongoDB.
 - Utiliser **dotenv** pour gérer les variables d'environnement.
2. **Frontend** :

- Utiliser **EJS** pour gérer les vues côté serveur.
- Ajouter un peu de **CSS** pour rendre l'interface agréable :
 - Un design propre et simple.
 - Un header avec un titre pour l'application.
 - Une table pour afficher les livres dans la liste.
 - Des boutons stylisés pour les actions (Ajouter, Modifier, Supprimer).

3. Validation des données :

- Les champs obligatoires doivent être vérifiés (par exemple, titre, auteur, année).
- Afficher des messages d'erreur utilisateur en cas d'erreur de validation.

4. Organisation :

- Suivez une structure modulaire pour le projet (routes, modèles, vues).
- Séparez les fichiers CSS dans un fichier distinct pour les ajouter au projet.

Instructions supplémentaires :

1. Comment livrer le projet :

- Fournissez un lien vers le dépôt GitHub contenant tout le code source.
- Assurez-vous que le fichier .env est exclu via .gitignore.

2. Bonus :

- Ajoutez une barre de recherche pour filtrer les livres par titre ou auteur.
- Ou ajoutez une pagination à la liste des livres si elle devient trop longue.
- Ou utiliser le **jwt** en ajoutant une interface pour s'authentifier.