

TP 02

M° 206 Création d'une application cloud native
Année 2023-2024

Filière : DDOWFS

Durée : 00 H 40

Niveau : Technicien spécialisé

Exercice: API RESTful pour gérer des livres avec Mongoose et Express.js

Objectif :

- Développer une API RESTful complète pour gérer des livres en utilisant Mongoose et Express.js.
- Mettre en œuvre les opérations CRUD (Create, Read, Update, Delete) pour les livres.
- Ajouter des routes supplémentaires pour filtrer les livres par genre et récupérer l'auteur d'un livre.
- Tester l'API avec des requêtes HTTP.

Tâches :

1. Créer un modèle Mongoose pour les livres:

- Définir un schéma Mongoose pour représenter un livre avec les champs *id*, *titre*, *auteur*, *genre* et *anneePublication*.
- Exporter le modèle pour l'utiliser dans les routes Express.

2. Implémenter les routes CRUD pour les livres:

- **GET /livres:** Récupérer tous les livres.
 - Afficher la liste de tous les livres dans la base de données.
- **GET /livres/:id:** Récupérer un livre spécifique par son ID.
 - Rechercher un livre par son ID unique et afficher ses détails.
 - Gérer le cas où aucun livre n'est trouvé avec l'ID donné.
- **POST /livres:** Créer un nouveau livre.
 - Créer un nouveau livre en utilisant les données fournies dans le corps de la requête.
 - Enregistrer le nouveau livre dans la base de données et renvoyer l'objet créé.
- **PUT /livres/:id:** Mettre à jour un livre existant.
 - Rechercher un livre par son ID et mettre à jour ses champs avec les données fournies dans le corps de la requête.
 - Gérer le cas où aucun livre n'est trouvé avec l'ID donné.
 - Renvoyer l'objet livre mis à jour.
- **DELETE /livres/:id:** Supprimer un livre.
 - Supprimer un livre de la base de données par son ID.
 - Gérer le cas où aucun livre n'est trouvé avec l'ID donné.
 - Renvoyer une réponse appropriée indiquant la réussite ou l'échec de la suppression.

3. Ajouter des routes supplémentaires:

- **GET /livres/genre/:genre:** Récupérer tous les livres d'un genre donné.
 - Filtrer les livres par le genre spécifié dans le paramètre :genre.
 - Afficher la liste des livres correspondant au genre donné.
 - Gérer le cas où aucun livre n'est trouvé pour le genre donné.
- **GET /livres/:id/auteur:** Récupérer l'auteur d'un livre par son ID.
 - Rechercher un livre par son ID et extraire le champ auteur.
 - Renvoyer le nom de l'auteur du livre.
 - Gérer le cas où aucun livre n'est trouvé avec l'ID donné.

4. **Tester l'API:**

- Utiliser un outil comme Postman ou un navigateur avec des extensions HTTP pour tester les différentes routes de l'API.
- Vérifier que les requêtes GET, POST, PUT et DELETE fonctionnent comme prévu pour les livres.