

Enseignants : Salwa Mbarek et Samah Hadouej
Date de distribution : 20 Novembre 2024
Sujet : **Travail pratique 3 : Node.js, MongoDB**
Poids : 25%
Valeur : sur 100
Date de remise : Au plus tard lundi, 03 décembre 2024, 23h59
Remise : Tout votre projet sur Léa (fichier zip)

/100

Objectif

Cette partie vise à enrichir le site web développé lors du TP2 en intégrant une base de données **MongoDB** pour gérer les données associées aux différents formulaires. L'objectif est d'utiliser **Node.js** et **mongoose** pour interagir avec la base de données afin qu'elle reflète les informations soumises via ces formulaires.

Tâches

1. Configuration de MongoDB Atlas

- **Création de la base de données :**
 - Créer une base de données sur [MongoDB Atlas](#) pour y stocker les données pertinentes de votre site.
 - Créez des collections dans cette base pour organiser vos données. Par exemple :
 - Une collection pour les utilisateurs si vous avez un formulaire d'inscription.
 - Une collection pour les messages si vous avez un formulaire de contact.
 - Une collection pour d'autres types de données pertinentes selon votre projet.
- **Connexion à MongoDB avec Node.js**
 - Initialiser votre projet avec la commande `npm init -y`
 - Installez les modules nécessaires avec npm (exemple : `npm install mongoose`)
 - Configurez un fichier **.env** pour sécuriser vos informations de connexion (par exemple, la chaîne de connexion MongoDB).
 - Créer un script de connexion à la base de données à l'aide de Node.js et mongoose.

2. Modélisation des Données

- Analysez les formulaires existants sur votre site et identifiez les champs nécessaires pour chaque type de données. Par exemple :
 - Pour un formulaire d'inscription : prénom, nom, e-mail, mot de passe.

- Créez des schémas (mongoose) pour chaque collection, adaptées aux données des formulaires.
Par exemple:

```
const utilisateurSchema = {  
  prenom: "string",  
  nom: "string",  
  courriel: "string",  
  motDePasse : "string",  
};
```

Vous pouvez ajouter des contraintes sur les données (exemple : required, unique).

Les modèles associés aux schémas vous permettront d'effectuer les CRUD.

3. Gestion des données (CRUD)

Pour chaque collection, créer un script pour exécuter les CRUD !

- Insertion de données (Create)

Ajoutez une fonction dans votre script pour insérer des données fictives (documents) aux collections correspondant à vos formulaires.

- Lecture (Read)

Ajoutez une fonction dans votre script pour lire et afficher les données de la base dans la console.

- Modification (Update)

Ajoutez une fonction dans votre script pour mettre à jour une donnée existante dans la base.

- Suppression (Delete)

Ajoutez une fonction dans votre script pour supprimer des données existantes dans la base.

Grille de correction

Critères	Pondération
Configuration de MongoDB Atlas	/ 20
- Base créée et connectée avec succès	
- Collections adaptées aux formulaires	
Opérations CRUD	/ 50
- Création de données	
- Lecture de données	
- Mise à jour de données	
- Suppression de données	
Organisation et Qualité du Code	/ 30
- Code bien commenté et structuré	
- Respect des conventions	
Retard	-5% par jour de retard, jusqu'à 3 jours ouvrables (sinon 0)
Total	/ 100

Remarques

- Assurez-vous que toutes les nouvelles fonctionnalités respectent les normes de conception et de programmation.
- Testez chaque fonctionnalité pour garantir son bon fonctionnement sur différents navigateurs.
- Utilisez des commentaires dans votre code pour expliquer vos choix et la logique utilisée.
- La qualité du code et le fonctionnement sont deux critères différents. Toutefois, si vous n'avez pas tenté de faire l'ensemble du travail, vous ne pouvez pas avoir l'ensemble de vos points pour la qualité du code. Si des parties du travail sont absentes, vous serez pénalisé dans la qualité du code en proportion de la partie du travail qui n'a pas été faite.

Note sur le plagiat

Le travail est à faire **en équipes de 2. Tout le monde dans l'équipe doit participer.** Si vous avez un problème avec un coéquipier ou une coéquipière qui ne travaille pas, avertissez rapidement votre professeur pour trouver une solution.

Ne *partagez pas de code avec une autre équipe que la vôtre*, même pas "juste pour aider un ami". Ça serait un **plagiat**, et **tous les membres des équipes concernées auraient la note de zéro.**

Si jamais vous utilisez *chatGPT*, vous **devez** me citer quels bouts de code ont été générés avec quels prompts. Mieux vaut me donner plus de détails que pas assez.