

# Carnet de Notes Évolué

## Contexte :

Vous avez décidé d'étendre votre projet de carnet de notes pour le rendre plus complet et polyvalent. L'objectif est de créer une application web qui permet aux utilisateurs d'ajouter, visualiser, et organiser leurs notes de manière plus sophistiquée.

## Fonctionnalités Requises :

### Ajout de Notes (POST) :

Permettre aux utilisateurs d'ajouter de nouvelles notes avec des détails tels qu'un titre, un contenu, une catégorie et une date de création en utilisant la méthode POST.

### Affichage des Notes (GET) :

Afficher la liste des notes sur la page principale en utilisant la méthode GET.

Chaque note doit afficher un titre, un résumé du contenu, la catégorie et la date de création.

### Catégorisation des Notes :

Permettre aux utilisateurs de spécifier une catégorie lors de l'ajout d'une note.

Afficher les notes groupées par catégorie sur une page dédiée.

### Modification de Notes (POST) :

Autoriser la modification du contenu, du titre ou de la catégorie d'une note existante en utilisant la méthode POST.

### Suppression de Notes (POST) :

Fournir une option pour supprimer une note en utilisant la méthode POST.

## Architecture Technique :

### Backend (Node.js + Express) :

Mettre en place un serveur Node.js à l'aide d'Express.

Implémentez des routes API RESTful pour les opérations d'ajout, de lecture et de suppression de notes.

Utilisez un système de stockage de données plus avancé, comme une base de données Mysql.

### Frontend (React) :

Créez une interface utilisateur avec React pour permettre aux utilisateurs d'ajouter, de visualiser, de modifier et de supprimer des notes.

Affichez les notes par catégorie et fournissez des options de filtrage.

### Points Supplémentaires (Optionnels) :

#### Authentification (Bonus) :

Mettez en place un système d'authentification pour permettre aux utilisateurs de s'inscrire, de se connecter et de gérer leurs propres notes.

#### Recherche (Bonus) :

Ajoutez une fonction de recherche permettant aux utilisateurs de rechercher des notes par titre, contenu ou catégorie.

#### Design Avancé :

Améliorez le design de l'application en utilisant des concepts avancés de CSS ou en intégrant une bibliothèque de composants React.

### Livrables Attendus :

#### Code Source :

Fournissez le code source complet de votre projet.

Organisez votre code de manière modulaire et commentez-le de manière appropriée.

GIT

#### Documentation :

Incluez une documentation expliquant comment configurer et exécuter votre application.

Ajoutez des captures d'écran de l'application en action.

### Conseils :

Utilisez des technologies modernes et des bonnes pratiques de développement.

Divisez votre projet en tâches plus petites et gérez-les avec un système de contrôle de version (par exemple, Git).

Testez votre application de manière approfondie pour vous assurer que toutes les fonctionnalités fonctionnent correctement.