

Sujet ECF

Sujet ECF : Application Web de Gestion de Projets

Objectif : Développer une application web de gestion de projets permettant aux utilisateurs de créer, gérer et suivre l'avancement de différents projets, ainsi que de gérer les tâches associées à ces projets. L'application devra fournir une interface utilisateur (front-end) conviviale et communiquer avec un backend via des API REST. Le backend stockera les données dans une base de données relationnelle ou non relationnelle selon votre préférence.

Fonctionnalités Requises

1. **Authentification des Utilisateurs** : Les utilisateurs doivent pouvoir s'inscrire, se connecter, et se déconnecter. Chaque utilisateur aura accès uniquement à ses projets et tâches.
2. **Gestion des Projets** :
 - **Création de Projet** : Un utilisateur peut créer un nouveau projet en spécifiant un nom et une description.
 - **Visualisation des Projets** : Les utilisateurs peuvent voir la liste de tous leurs projets.
 - **Modification et Suppression** : Les utilisateurs peuvent modifier les détails d'un projet ou supprimer un projet.
3. **Gestion des Tâches** :
 - **Création de Tâche** : Dans un projet, les utilisateurs peuvent ajouter des tâches avec des détails tels que le titre, la description, la priorité, et les échéances.
 - **Visualisation des Tâches** : Afficher les tâches associées à un projet.
 - **Modification et Suppression** : Les utilisateurs peuvent modifier ou supprimer les tâches.
4. **Suivi de l'Avancement** : Les utilisateurs peuvent marquer les tâches comme étant "En cours" ou "Complétées" et visualiser l'avancement du projet.

Réalisation

Les stagiaires doivent créer un dépôt Git pour héberger le code source de leur application. Ce dépôt devrait inclure :

- Front-end : Le code source de l'interface utilisateur, incluant HTML, CSS, et JavaScript (ou TypeScript), en utilisant le framework choisi.
- Le code source de l'API REST, incluant la logique métier, les modèles de données.
- Base de données : Les scripts de création de la base de données ou un modèle de données si une base de données NoSQL est utilisée.
- Modèle de données : Un diagramme de la base de données relationnelle (schéma ER) ou un modèle de données pour une base de données NoSQL. Cela inclut les tables/collections, les relations/clés, et une brève description de chaque élément.
- Diagrammes UML : Diagrammes de cas d'utilisation, diagrammes de séquence, et/ou diagrammes de classe pour décrire les interactions entre les utilisateurs et l'application, ainsi que la structure interne de l'application.
- Documentation : Un fichier README.md à la racine du dépôt qui explique comment installer, configurer, et exécuter l'application. Ce fichier devrait également documenter l'API REST (endpoints, méthodes HTTP, formats de requête/réponse).

