

CAHIER DES CHARGES

My Planning une application web simple de gestion de tâches (To-Do List). Ce projet est un tableau de bord minimal pour aider les utilisateurs à organiser leur journée et gérer leurs tâches de manière plus efficace.

Fonctionnalités de base :

1. Création des tâches

- L'utilisateur peut créer une nouvelle tâche en fournissant un titre et une description.
- Chaque tâche doit avoir un statut par défaut (Non complétée).

2. Consultation des tâches

- L'utilisateur peut consulter la liste complète des tâches créées.
- Il peut également consulter une tâche spécifique par son identifiant.
-

3. Mise à jour des tâches

- L'utilisateur peut modifier les informations d'une tâche existante (titre, description, statut).

4. Suppression des tâches

- L'utilisateur peut supprimer une tâche de la liste.
-

5. Statut des tâches

- Chaque tâche peut être marquée comme Complétée ou Non complétée.
- Les tâches Complétées peuvent être filtrées ou consultées séparément.

6. Filtrage et recherche

- L'utilisateur peut filtrer les tâches par statut (Complétée, Non complétée).
- Il peut également rechercher des tâches par titre ou contenu dans la description.

Fonctionnalités supplémentaires (suggestions pour contributions)

- **Pagination** : Si la liste des tâches devient trop longue, ajouter la pagination pour éviter une surcharge de données.
- **Auth** : Mettre en place une authentification pour que chaque utilisateur gère ses propres tâches.
- **Priorités** : Ajouter un système de priorités pour chaque tâche (basse, moyenne, haute).
- **Catégories** : Permettre de classer les tâches par catégories (ex. travail, personnel).
- **Notifications** : Implémenter un système de notifications pour avertir l'utilisateur lorsqu'une tâche est proche de sa date limite.
- **Autres propositions des contributeurs.**

Technologie de base du projet :

-**Frontend** : NuxtJS (OpenSource) pour coder le client Side

-**Backend** : NestJs (OpenSource aussi) pour coder l'api

-**Base de données** : Mongo DB (Cluster pour le projet)

-**Sécurité** : JWT

Le projet ci utilise l'architecture MVC

Comment contribué

- Avoir des connaissances en **TypeScript**, **NestJS**, et **REST APIs**.
- Installer **Node.js** et **npm**.
- Connaître **Git** pour la gestion des branches et des pull requests.
- Forker le dépôt.
- Créer une branche avec une description claire (ex : feature_add_[nom de la fonctionnalité]).
- Faire des commit réguliers et explicites.
- Proposer un pull **request** pour que les changements soient revus.

Déroulement du projet

Le projet sera ouvert à la contribution pour toute la durée du Hacktoberfest. Les contributeurs peuvent proposer de nouvelles fonctionnalités, corriger des bugs ou améliorer la documentation.

Les contributions les plus pertinentes et bien documentées seront intégrées dans le projet final.