

Cahier des Charges

1. Introduction

Une application web de gestion de projets collaboratifs. Elle permet aux utilisateurs de créer des projets, d'y ajouter des membres et des tâches, et de suivre l'avancement du travail dans un cadre structuré.

2. membres du groupe

Bibari Rayen, Ben Bachir Achref, Oussema Ghbara, Chebbi Jesser

3. Rôles et Fonctionnalités

1. Utilisateur normal

1.1. Accès au compte

- Peut s'inscrire
- Peut se connecter (log in)
- Peut modifier son compte (nom, prénom, courriel, image)

1.2. Accès aux projets

- Peut créer un projet
- Peut voir tous les projets :
 - S'il n'est pas membre : ne voit que la date de création et le statut
 - S'il est membre : voit toutes les informations

1.3. Accès aux tâches

- Peut voir toutes les tâches d'un projet où il est membre
- Peut modifier uniquement les tâches qui lui sont assignées ◦
Exemple : changer le statut, ajouter une note

2. Super administrateur

2.1. Accès aux utilisateurs

- Peut voir la liste de tous les utilisateurs
- Accès à toutes les informations sauf le mot de passe

- Ne peut pas modifier les comptes

2.2. Accès aux projets

- Peut voir la liste complète de tous les projets
 - Accès à toutes les informations en lecture seule
-

3. Créateur d'un projet (ou manager)

3.1. Permissions dans le projet

- Peut modifier toutes les infos du projet
- Peut supprimer le projet

3.2. Gestion des membres

- Peut ajouter ou retirer des membres
- Peut leur assigner un rôle (ex. : membre simple)

3.3. Gestion des tâches

- Peut créer des tâches
 - Peut assigner une tâche à un membre spécifique
-

4. Membre d'un projet

4.1. Accès aux données

- Peut voir toutes les informations du projet

4.2. Accès limité

- Ne peut rien modifier sauf :
 - Les tâches qui lui sont assignées

5. Technologies utilisées

Voici les technologies principales utilisées dans le développement de l'application

- JavaScript : utilisé pour le développement backend avec Node.js et Express.
- SQL : utilisé pour manipuler et interroger la base de données MySQL.
- Node.js : environnement d'exécution côté serveur.
- Express.js : Framework pour créer l'API REST.

- EJS : Le frontend de l'application sera développé à l'aide du moteur de Template **EJS**
- Sequelize : ORM pour interagir avec MySQL via des objets JavaScript.
- MySQL : système de gestion de base de données relationnelle.