

Phases de projet:

exigences fonctionnelles

1. Collecte des exigences

Effectuer l'ingénierie des exigences pour identifier et documenter les besoins des utilisateurs potentiels :

1 -Élicitation :

1. Interviews et scénarios .
2. Un diagramme de **cas d'utilisation**(administrateur ,chef de projet ,membre de l'équipe) .
3. Prototypes .

Diagramme de cas d'utilisation

1:Authentification et autorisation des utilisateurs

-Le système doit permettre aux utilisateurs de s'inscrire avec un e-mail et un mot de passe sécurisé.

-Les utilisateurs doivent pouvoir se connecter avec leur identifiant et mot de passe.

- Un système de contrôle d'accès basé sur les rôles doit être mis en place, incluant au minimum les rôles suivants :(peut être pour la connexion il faut faire 3 boutons pour les 3 utilisateurs.

- **Administrateur** : Peut gérer tous les aspects de l'application, y compris les rôles des utilisateurs, les projets et les paramètres système.
- **Chef de projet** : Peut créer et gérer des projets, assigner des tâches, consulter les rapports.
- **Membre de l'équipe** : Peut voir les projets auxquels il est assigné, gérer les tâches qui lui sont assignées, et enregistrer le temps de travail.

-gérer les profils incluant les modifications des informations personnelles et les mots

de passe ,réinitialisation de mots de passe en cas d'oubli.

2:Tableau de bord:

Exigences Fonctionnelles :

Les tâches à venir et les échéances doivent être visibles pour l'utilisateur connecté.

Les indicateurs clés de performance (KPI) tels que le nombre de tâches en attente, les tâches en retard, et l'activité récente (ajout de tâches, modification d'état, etc.) doivent être affichés.

Affichage des tâches en attente, avec des indicateurs de priorités et d'échéances.

Barres de progression pour chaque projet indiquant le pourcentage d'avancement basé sur les tâches terminées.

Le tableau de bord doit être personnalisable en fonction des rôles :

- Administrateur : Vue d'ensemble de tous les projets et utilisateurs.
- Chef de projet : Vue détaillée des projets et tâches gérés par le chef de projet.
- Membre de l'équipe : Vue des tâches et des projets auxquels il est assigné.

3:Gestion de projets et de tâches :

Chef de projet:

- accéder à gestion des projets depuis le tableau de bord ,
- créer un projet , modifier et supprimer un projet .le projet doit inclure des informations :nom du projets la description ,les dates de début et de fin et les membres de l'équipe qui participent à ce projet
- Gérer les tâches notamment **créer ,affecter ,modifier ,supprimer** les tâches dans le cadre d'un projet ,afficher le priorité des tâches,les échéances et le suivi de l'état des tâches.
- Définir les dépendances entre les tâches par exemple une tâche x ne peut pas commencer avant que la tâche z soit terminé .

Membres de l'équipe:

- Classement des tâches par étiquette(tag) pour faciliter la recherche et mettre un système de connexion sécurisé avec un contrôle d'accès basé sur les rôles.
- Changement de l'état de la tâche .

Administrateur:

- Changer le rôle d'un utilisateur.

4 Collaboration équipes

- Les utilisateurs doivent pouvoir ajouter des commentaires dans les projets.
- Les autres utilisateurs doivent recevoir une notification pour les nouveaux commentaires ajoutés sur les projets auxquels ils sont assignés.

5:Suivi du temps

- Les membres de l'équipe doivent pouvoir enregistrer le temps passé sur chaque tâche, avec la possibilité d'ajouter des commentaires.
- Le système doit permettre d'afficher le temps total passé sur chaque projet et chaque tâche.
- Les chefs de projet doivent pouvoir générer des rapports de temps pour analyser la répartition des efforts sur les projets.

6 Notification et alertes :

- Le système doit envoyer des alertes aux utilisateurs en cas de dépassement d'échéance, d'assignation de nouvelles tâches, ou de modification d'état des tâches.
- Les notifications doivent être envoyées par e-mail et/ou via des notifications intégrées à l'application (push notifications).

exigences non fonctionnelles

1.Conception Réactive :

L'application est accessible sur divers appareils ,

Utilisations:

Media queries dans le css pour adapter les design à différentes tailles d'écran.

Grilles css .

Navigation facile sur mobile:menus déroulants et icônes simplifiées.

2.Sécurité des données et des utilisateurs :

-Garantir que les informations sensibles sont protégées contre les accès non autorisés et les cyberattaques.

Chiffrement des données : Utilisation de **HTTPS** pour sécuriser les échanges entre le client et le serveur.

Protection des mots de passe : Utilisation de **hachage** sécurisé (par exemple, bcrypt) pour les mots de passe des utilisateurs.

Contrôle d'accès strict : Mise en place de politiques d'accès basées sur les rôles, avec validation systématique des permissions sur chaque action critique.

3.Performances

Utilisation de chargement paresseux (lazy loading) pour recharger les sections de la page uniquement quand nécessaire.

Minification et compression des fichiers CSS,JS et HTML *pour réduire le temps de chargement* .

Mise en place une base de données optimisée en utilisant des d'index sur les colonnes fréquemment utilisées dans les requêtes(Projets, tâches, utilisateurs).

4.Evolutivité:

L'application doit pouvoir gérer une augmentation de nombre d'utilisateurs, de projets et des tâches