



**ECOLE MAROCAINE DES
SCIENCES DE L'INGENIEUR**
Membre de
HONORIS UNITED UNIVERSITIES



Cahier De Charge

Projet :

Une Application Web de Gestion de Projet en Équipe

VERSION	AUTEUR	DATE D'ÉDITION	VALIDATEUR	DATE VALIDATION
1.0	Marwa El-assri Rim Nacerddine	06-11-2024	Prof Mehdi Belasla	
2.0	Marwa El-assri Rim Nacerddine	06-12-2024	Prof Mehdi Belasla	

Sommaire

1.	Périmètre.....	3
1.1.	But.....	3
1.2.	Missions	3
1.3.	Contraintes.....	3
2.	Etude de l'existant.....	3
	→ Fonctionnalités principales	4
	→ Architecture Technique	4
	→ Forces & Faiblesses	4
3.1	Objectifs fonctionnels	5
	→ Planification et Structuration des Projets	5
	→ Gestion des Ressources	5
	→ Suivi et Contrôle des Projets	5
	→ Amélioration de la Communication et de la Collaboration	6
3.2	Besoins fonctionnels :	8
3.3	Acteurs	9
4.	Besoins non fonctionnels	11

1. Périmètre

1.1. But

Le but de la création d'une application web de gestion de projet en équipe est de faciliter la collaboration, la planification et la coordination des tâches au sein d'une équipe. Cette application permet de :

- **Centraliser les informations** : Accès facile aux documents et données du projet en un seul endroit.
- **Améliorer la communication** : Outils intégrés (messagerie, commentaires) pour faciliter les échanges entre membres.
- **Suivi des tâches et des responsabilités** : Définition, attribution et suivi des tâches et deadlines pour chaque membre.
- **Gestion des ressources** : Optimisation des ressources (temps, personnel, budget) nécessaires au projet.
- **Suivi de l'avancement et des performances** : Indicateurs pour évaluer l'avancement et identifier les obstacles.
- **Faciliter la prise de décisions** : Rapports et données centralisées pour des décisions éclairées.

1.2. Missions

- Faciliter la planification et l'organisation.
- Optimisation de l'utilisation des ressources.
- Améliorer la collaboration et la communication.
- Suivre et contrôler les projets.
- Fournir des rapports et des analyses.
- Assurer la satisfaction des parties prenantes.

1.3. Contraintes

- **Délais de réalisation** :
 - Respecter les délais fixés pour chaque phase du projet (conception, développement, tests et déploiement).
 - Livraison de la première version de l'application dans un délai de 3 mois à compter du début du projet.
- **Respect de la charte graphique** :
 - Utiliser les couleurs et les polices définies par l'entreprise.
 - Respecter le logo et son emplacement dans l'interface.
 - Maintenir une cohérence visuelle entre les différentes pages de l'application.

2. Etude de l'existant

❖ Présentation de l'application Zoho Projects :

Zoho Projects est une application web de gestion de projets développée par Zoho Corporation. C'est une solution populaire pour la gestion des tâches, la collaboration en équipe

et le suivi des projets en temps réel. Elle propose une variété d'outils pour les entreprises de toutes tailles, facilitant la planification, l'exécution et la gestion des ressources.

→ **Fonctionnalités principales**

a) Gestion des tâches et des projets

- **Création et hiérarchisation des tâches** : Zoho Projects permet de créer des tâches avec des sous-tâches, de les assigner à des membres, et de définir des priorités et des délais.
- **Diagramme de Gantt** : Zoho Projects visualise l'avancement du projet et montre les dépendances entre les tâches.
- **Suivi du temps** : Intègre un outil de suivi du temps pour enregistrer les heures travaillées sur chaque tâche.

b) Collaboration et communication

- **Commentaires et discussions** : Les utilisateurs peuvent commenter directement sur les tâches et initier des discussions en équipe.
- **Chat en temps réel** : Fonctionnalité de messagerie pour faciliter la communication immédiate.
- **Partage de documents** : Permet de partager, stocker et organiser des documents liés au projet.

c) Reporting et analyses

- **Tableaux de bord personnalisés** : Affiche les données de performance du projet, telles que les tâches en cours, les tâches terminées, et les délais.
- **Rapports analytiques** : Génération de rapports pour suivre les performances et identifier les blocages.

d) Intégrations

Zoho Projects propose des intégrations avec d'autres applications comme Slack, Google Drive, Dropbox, Zoho CRM, et des applications de bureautique. Cela facilite la synchronisation des données et la collaboration multi-plateforme.

→ **Architecture Technique**

Zoho Projects repose sur une architecture en SaaS (Software as a Service), permettant aux utilisateurs de se connecter depuis n'importe quel navigateur sans installation locale. La plateforme est construite pour une accessibilité multi-appareil (ordinateurs, smartphones, tablettes) et repose sur des technologies web modernes telles que React et Angular pour l'interface, ainsi qu'une infrastructure cloud pour l'hébergement.

→ **Forces & Faiblesses**

✓ **Forces** :

- ★ **Accessibilité et mobilité** : Accessible partout et offre une application mobile performante.
- ★ **Interface conviviale** : L'interface est intuitive et bien organisée, rendant la navigation facile.
- ★ **Grande flexibilité** : Permet une personnalisation selon les besoins de l'équipe.
- ★ **Intégrations étendues** : Compatible avec de nombreux autres outils, ce qui évite de basculer entre les applications.

✓ **Faiblesses :**

- ★ **Courbe d'apprentissage** : Certaines fonctionnalités avancées nécessitent une période d'adaptation.
- ★ **Limites des versions gratuites** : La version gratuite a des limitations importantes, ce qui peut contraindre certaines équipes.
- ★ **Manque de certaines fonctionnalités spécifiques** : Bien qu'elle soit complète, elle peut manquer de certaines fonctionnalités spécialisées que d'autres applications comme Asana ou Trello proposent.

3. Etude fonctionnelle

3.1 Objectifs fonctionnels

Les principaux objectifs fonctionnels d'une application de gestion de projets d'équipes comme Zoho Projects

→ Planification et Structuration des Projets

- **Définir les étapes clés** : Créer des jalons et structurer le projet en étapes pour une meilleure organisation et une vision claire des objectifs.
- **Planification des tâches** : Assigner des tâches à chaque membre de l'équipe, définir des délais, établir des dépendances, et identifier les priorités pour chaque tâche.
- **Établir les calendriers** : Utiliser des outils comme les diagrammes de Gantt pour visualiser les plannings, planifier les ressources, et éviter les retards.

→ Gestion des Ressources

- **Allocation des ressources** : Gérer les affectations en fonction de la disponibilité, des compétences, et des charges de travail de chaque membre de l'équipe.
- **Suivi des heures de travail** : Suivre le temps passé sur chaque tâche et calculer le coût de la main-d'œuvre, ce qui aide à mieux contrôler les budgets.
- **Optimisation de la charge de travail** : S'assurer que les tâches sont réparties équitablement pour éviter les surcharges et augmenter la productivité.

→ Suivi et Contrôle des Projets

- **Suivi de l'avancement** : Surveiller l'évolution du projet en temps réel et repérer rapidement les éventuels retards ou obstacles.
- **Suivi des performances** : Analyser les performances de chaque membre et de l'équipe globale, avec des rapports et des tableaux de bord.
- **Gestion des risques** : Identifier et anticiper les problèmes potentiels en suivant les progrès et en ajustant les ressources ou les délais si nécessaire.

→ **Amélioration de la Communication et de la Collaboration**

- **Communication centralisée** : Faciliter les échanges en centralisant les discussions, commentaires et notifications dans l'application, ce qui réduit les emails dispersés.
- **Partage de fichiers et de documents** : Permettre à tous les membres d'accéder aux documents nécessaires, de partager des fichiers, et de les mettre à jour en temps réel.
- **Collaboration en temps réel** : Encourager la collaboration en permettant aux membres de travailler ensemble sur des tâches, de commenter et de suivre les progrès simultanément.

Règle de gestion	Description	Type
RG01	Les projets doivent être créés avec un titre, une description, une date de début, une date de fin, et un chef de projet.	Métier
RG02	Chaque tâche doit être assignée à un membre de l'équipe, avec une priorité et une échéance définie.	Métier
RG03	Seuls les membres authentifiés peuvent accéder aux projets et tâches de l'équipe.	Sécurité
RG04	Les chefs de projet sont les seuls à pouvoir modifier ou supprimer un projet.	Sécurité
RG05	Chaque membre ne peut consulter que les tâches qui lui sont assignées ou celles de son équipe.	Sécurité
RG06	Une notification doit être envoyée lorsqu'une tâche est en retard ou proche de son échéance.	Fonctionnelle
RG07	Les modifications des projets et tâches doivent être enregistrées dans un journal d'activités.	Métier

RG08	Les données de l'application doivent être sauvegardées quotidiennement et accessibles en cas de panne.	Technique
RG09	L'accès aux fonctionnalités est défini par le rôle : administrateur, chef de projet ou membre de l'équipe.	Sécurité
RG10	Les rapports d'avancement peuvent être générés uniquement par les chefs de projet.	Métier

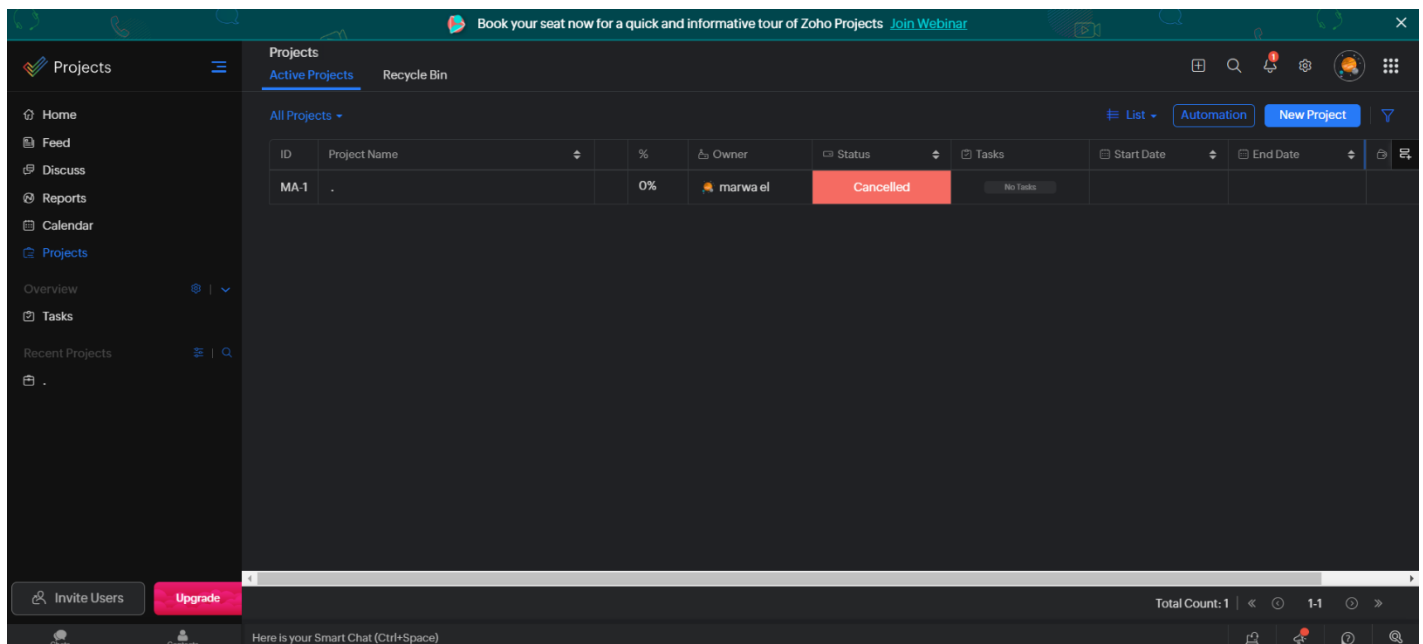


Figure : La page d'accueil de l'application ZOHO Project.

3.2 Besoins fonctionnels :

Les besoins fonctionnels représentent les fonctionnalités essentielles que l'application doit offrir pour atteindre les objectifs du projet.

- **Gestion de Projet et des Tâches**

- Création de projets* : Permettre aux utilisateurs de créer des projets en saisissant un titre, une description, une date de début, une date de fin, et en désignant un chef de projet.

- Gestion des tâches* : Les utilisateurs doivent pouvoir créer des tâches et sous-tâches, les affecter à des membres, définir des priorités et des statuts.

- Suivi de l'avancement* : L'application doit fournir une vue d'ensemble de l'avancement des tâches et projets, via des vues de type tableau de bord et diagrammes de Gantt.

- **Collaboration et Communication**

- Messagerie instantanée* : Intégration d'une messagerie pour permettre la communication en temps réel entre les membres d'un projet.

- Commentaires et discussions* : Chaque tâche doit disposer d'un espace de commentaires pour permettre les discussions spécifiques aux tâches.

- Notifications* : L'application doit envoyer des notifications pour les mises à jour importantes (nouvelle tâche assignée, tâche à échéance, modifications de projet).

- **Gestion des Documents**

- Téléversement et partage de fichiers* : Les utilisateurs doivent pouvoir télécharger et partager des fichiers dans le cadre des projets.

- Gestion des versions* : Historique des versions des documents pour suivre les modifications et revenir aux versions antérieures si nécessaire.

- **Suivi du Temps et des Ressources**

- Suivi du temps* : Un chronomètre doit être intégré pour permettre aux membres de suivre le temps passé sur chaque tâche, avec possibilité de saisie manuelle.

- Tableau de bord des ressources* : Vue centralisée des heures travaillées et des ressources allouées pour chaque projet.

- **Génération de Rapports**

- Rapports d'avancement* : Génération de rapports sur l'avancement des projets et des tâches, avec filtres personnalisables par projet, équipe, ou période.

- Rapports de performance* : Génération de rapports sur les performances des équipes, taux de complétion des tâches, et respect des délais.

- **Gestion des Utilisateurs**

- Authentification et gestion des rôles* : L'application doit permettre une gestion des utilisateurs avec des rôles (administrateur, chef de projet, membre d'équipe) et des droits d'accès spécifiques pour chaque rôle.

- Profil utilisateur* : Chaque utilisateur doit disposer d'un profil où sont affichées ses informations de contact, son rôle, et son historique de tâches.

3.3 Acteurs

→ En général :

- Authentification
- Déconnexion

→ Chef de projet :

- Création de projet.
- Planification et répartition des tâches entre les membres de l'équipe.
- Suivi de l'avancement des projets et gestion des délais.
- Gestion des ressources et des charges de travail.
- Suivi des délais et des budgets
- Communication avec l'équipe

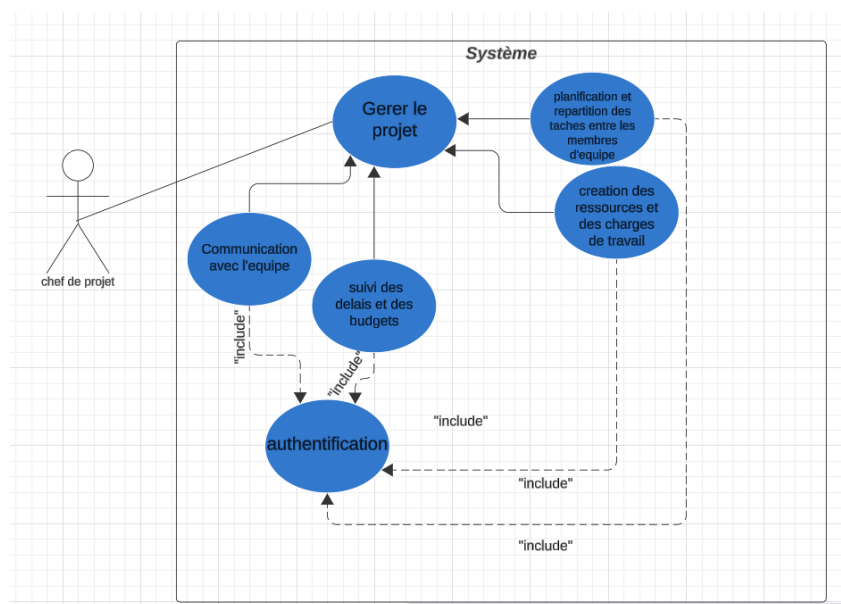


Figure : Diagramme de cas d'utilisation de chef de projet

→ Membres de l'équipe :

- Exécution des tâches assignées.
- Utilisation du suivi du temps pour enregistrer les heures passées sur chaque tâche (ce qui aide le chef de projet à comprendre les charges de travail).

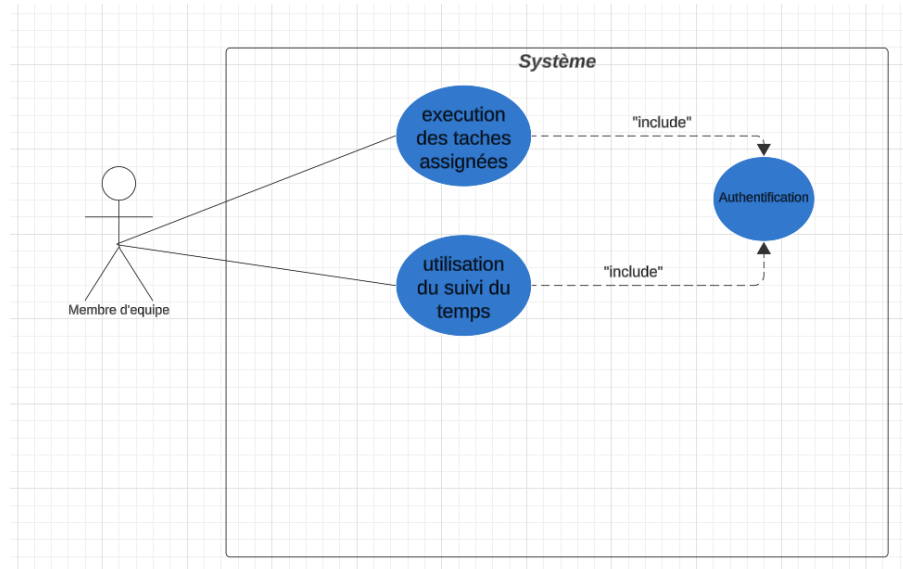


Figure : Diagramme de cas d'utilisation des membres de l'équipe

→ **Clients (parties prenantes externes) :**

- Suivi de l'avancement.
- Validation des livrables.
- Commentaires et retours.

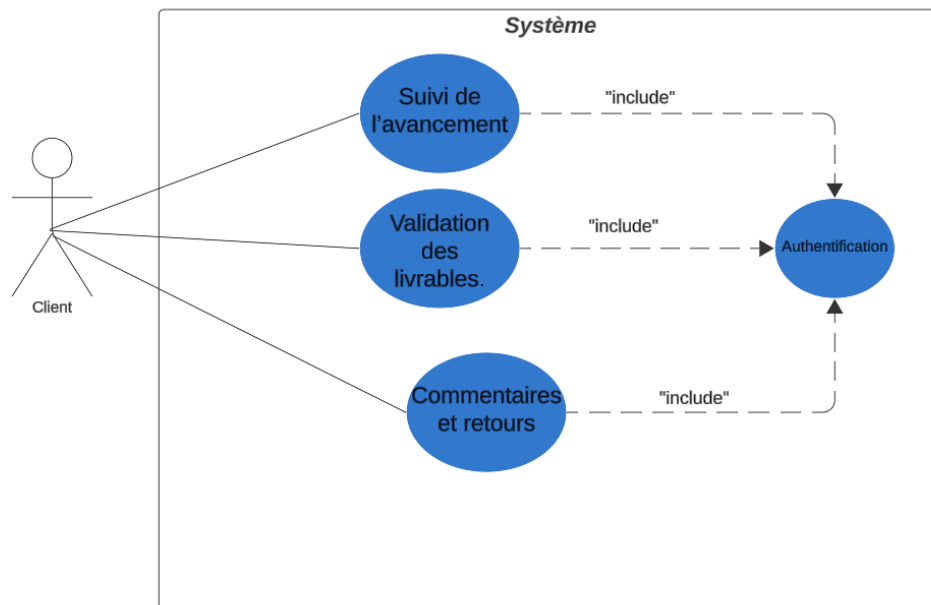


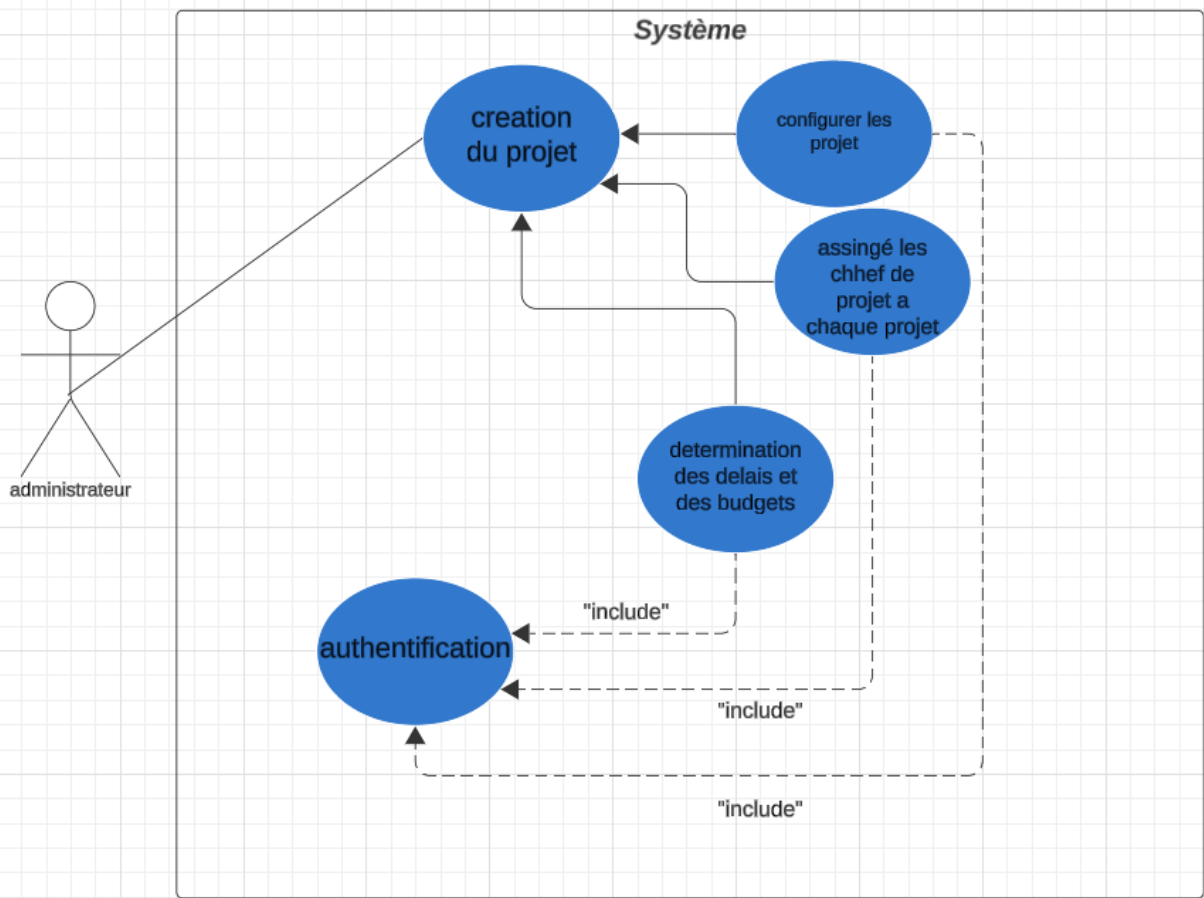
Figure : Diagramme de cas d'utilisation de client

→ **Administration :**

-Création du projet :

- ❖ Configurer les projets
- ❖ Assigné les chefs de projet a chaque projet

❖ Détermination des délais et budgets



4. Besoins non fonctionnels

→ Performance :

Temps de réponse : L'application doit répondre rapidement aux actions des utilisateurs (chargement de pages, affichage des tableaux de bord, mise à jour des données) en moins de 2 secondes pour offrir une expérience fluide.

Capacité de montée en charge : L'application doit pouvoir gérer simultanément plusieurs utilisateurs actifs sans dégradation des performances, même pendant les pics d'utilisation.

Traitement des requêtes : La plateforme doit supporter un volume élevé de requêtes, surtout lors des tâches lourdes comme le chargement de rapports ou l'affichage des diagrammes de Gantt.

→ Fiabilité :

Fiabilité des données : Les données de projet doivent être sauvegardées régulièrement pour éviter toute perte en cas de panne.

Reprise après sinistre : Mettre en place des mécanismes de sauvegarde automatique et de récupération rapide en cas de défaillance du système.

→ Sécurité :

Gestion des accès et des rôles : Contrôler les autorisations d'accès pour chaque rôle (chef de projet, membre d'équipe, client, etc.) pour garantir la confidentialité des informations sensibles.

Chiffrement des données : Utiliser le chiffrement des données en transit et au repos pour protéger les informations sensibles (notamment les identifiants et les documents partagés).

Authentification renforcée : Implémenter des options de connexion sécurisées, telles que l'authentification à deux facteurs (2FA), pour limiter les risques d'accès non autorisé.

Journalisation des activités : Garder une trace des activités (consultations, modifications, créations) pour pouvoir effectuer des audits de sécurité.

→ **Interopérabilité**

Intégration avec d'autres outils : L'application doit pouvoir s'intégrer facilement avec des outils tiers comme Google Workspace, Microsoft Teams, Slack, ou d'autres CRM pour une meilleure productivité.

Support des API : Fournir des API REST ou GraphQL pour permettre aux développeurs d'intégrer l'application avec d'autres systèmes internes ou d'automatiser des flux de travail.

→ **Facilité d'utilisation (Ergonomie)**

Interface intuitive : L'interface doit être simple, intuitive et adaptée aux besoins des différents utilisateurs pour minimiser la courbe d'apprentissage.

Expérience utilisateur optimisée : Offrir une navigation fluide, des raccourcis pour les actions fréquentes, et une interface uniforme pour faciliter l'adoption.

