

Application web de gestion des demandes internes

1. Intitulé de la mission

Conception et développement d'une application web de gestion des demandes internes, inspirée du fonctionnement de l'outil Trello.

2. Contexte du projet

L'entreprise souhaite mettre en place un outil web simple, centralisé et visuel permettant de gérer efficacement ses demandes internes. Ces demandes peuvent concerter des tâches, des projets, des interventions, des tickets ou encore des demandes clients.

Actuellement, ces informations sont dispersées à travers différents supports (e-mails, fichiers, notes manuscrites), ce qui rend le suivi complexe, peu lisible et source d'erreurs. L'absence d'une vision globale empêche également une bonne organisation du travail et limite la collaboration entre les utilisateurs.

L'application proposée s'inspire du principe de Trello, basé sur des tableaux, des listes et des cartes, afin d'offrir une organisation claire, intuitive et collaborative.

3. Objectifs de la mission

L'objectif principal de ce projet est de concevoir et développer une application web fonctionnelle permettant de centraliser et suivre les demandes internes de l'entreprise.

Les objectifs secondaires sont :

- analyser un besoin de gestion interne collaboratif,
- concevoir une application web de type Kanban,
- mettre en place une base de données relationnelle,
- développer une interface utilisateur claire et intuitive,
- appliquer les bonnes pratiques de développement et de sécurité,
- documenter l'ensemble du projet pour une utilisation et une maintenance facilitées.

4. Périmètre de l'application

4.1 Utilisateurs concernés

L'application s'adresse à deux types d'utilisateurs.

L'administrateur dispose de droits étendus. Il peut gérer les utilisateurs, accéder à l'ensemble des tableaux, listes et demandes, et superviser globalement l'activité de l'application.

L'utilisateur peut créer et gérer ses propres demandes, participer à des tableaux partagés et suivre l'avancement des tâches qui lui sont attribuées.

5. Fonctionnalités attendues

5.1 Authentification et gestion des comptes

L'application doit proposer un système d'authentification sécurisé permettant aux utilisateurs de se connecter et de se déconnecter.

Les rôles (administrateur ou utilisateur) doivent être gérés afin de restreindre l'accès à certaines fonctionnalités.

Les mots de passe devront être stockés de manière sécurisée à l'aide d'un système de chiffrement (hash).

5.2 Gestion des tableaux

Les tableaux constituent l'élément principal de l'application.

Un utilisateur autorisé peut créer un tableau en renseignant un nom et une description.

Il peut consulter les tableaux auxquels il a accès, les modifier si nécessaire, et éventuellement les partager avec d'autres utilisateurs.

La suppression d'un tableau est réservée à l'administrateur.

5.3 Gestion des listes

Chaque tableau est composé de plusieurs listes représentant les différentes étapes d'avancement (par exemple : « À faire », « En cours », « Terminé »).

Les utilisateurs peuvent créer des listes, modifier leur nom, les supprimer et définir leur ordre d'affichage au sein du tableau.

5.4 Gestion des cartes (demandes)

Les demandes sont représentées sous forme de cartes placées dans les listes.

Lors de la création d'une carte, plusieurs informations peuvent être renseignées : un titre, une description, un niveau de priorité, un statut, une date de création, une date limite (optionnelle) ainsi que l'auteur de la demande.

Les cartes peuvent être consultées par tableau ou par liste.

Elles peuvent être modifiées à tout moment et déplacées d'une liste à une autre afin de refléter l'avancement du travail.

La suppression d'une carte est réservée à l'administrateur.

5.5 Fonctionnalités avancées inspirées de Trello

L'application intègre plusieurs fonctionnalités avancées afin d'améliorer la collaboration.

Il est possible d'assigner une ou plusieurs cartes à des utilisateurs afin de clarifier les responsabilités.

Chaque carte dispose d'un espace de commentaires permettant aux utilisateurs d'échanger et de conserver un historique des discussions.

Des étiquettes (labels) peuvent être créées et associées aux cartes afin de faciliter le classement et le filtrage.

L'ajout de pièces jointes (fichiers ou liens) peut être proposé de manière optionnelle.

Enfin, un historique des actions est conservé afin d'assurer une traçabilité des modifications effectuées sur les cartes.

5.6 Interface utilisateur

L'interface de l'application doit être claire, ergonomique et inspirée d'un tableau Kanban.

Elle peut intégrer, de manière optionnelle, une fonctionnalité de glisser-déposer en JavaScript pour déplacer les cartes entre les listes.

L'application doit afficher des messages de confirmation et d'erreur explicites, proposer une navigation simple et être responsive afin de s'adapter aux différents écrans.

5.7 Données et stockage

Les données sont stockées dans une base de données relationnelle MySQL.

Les principales tables concernent les utilisateurs, les tableaux, les listes, les cartes, les commentaires et les étiquettes.

Des relations doivent être établies entre ces tables afin d'assurer la cohérence des données.

6. Contraintes techniques

L'application doit être accessible via un navigateur web.

Le code doit être structuré, commenté et respecter une architecture claire, de préférence de type MVC.

Des règles de sécurité doivent être appliquées, notamment la validation des données, le contrôle des accès, la gestion des sessions et la protection des fonctionnalités sensibles.

7. Gestion des droits et sécurité

7.1 Droits de l'utilisateur

Un utilisateur peut créer des tableaux et des cartes, consulter uniquement les tableaux auxquels il a accès, modifier ses propres cartes et ajouter des commentaires.

Il ne peut pas supprimer de carte.

7.2 Droits de l'administrateur

L'administrateur dispose d'un accès complet à l'application.

Il peut modifier et supprimer toutes les cartes, accéder à tous les tableaux et gérer les comptes utilisateurs (création, modification des rôles, désactivation ou suppression).

7.3 Sécurité

Les droits doivent être contrôlés côté serveur.

Toute tentative d'accès non autorisé doit être bloquée et les routes sensibles protégées.

8. Technologies utilisées

- Frontend : HTML, CSS, JavaScript
- Backend : PHP
- Base de données : MySQL

9. Organisation du travail

Le projet est organisé en plusieurs phases :

- une phase d'analyse et de conception (analyse des besoins, schéma de base de données, maquettes),
- une phase de développement (authentification, gestion des tableaux, listes et cartes),
- une phase de tests et de sécurisation,
- une phase de documentation.

10. Livrables attendus

Les livrables attendus à la fin du projet sont :

- le code source complet de l'application,
- une base de données fonctionnelle,
- un schéma relationnel,
- une documentation technique,
- un guide utilisateur,
- une présentation du projet.