



# CodeIguanas

## Cahier des charges

### Application HomeSkolar

## Spécifications fonctionnelles

1. *Introduction* Le projet HomeSkolar a pour objectif de créer une plateforme web permettant de mettre en relation des enfants en difficulté scolaire avec des tuteurs bénévoles. Le site web doit être convivial, intuitif et répondre aux besoins spécifiques de HomeSkolar pour faciliter le tutorat scolaire en ligne.
2. *Fonctionnalités principales :*

### **INSCRIPTION**

Objectif : Permettre aux utilisateurs (élèves et tuteurs) de s'inscrire à la plateforme.

Fonctionnalité :

- Page d'inscription avec les champs suivants : nom, prénom, adresse email, mot de passe, rôle (élève ou tuteur).

### **AUTHENTIFICATION**

Objectif : Permettre aux utilisateurs (élèves et tuteurs) de s'authentifier sur la plateforme.

Fonctionnalité :

- Page de connexion avec les champs "adresse email" et "mot de passe".

### **MODIFICATION DE MOT DE PASSE**

Objectif : Permettre aux utilisateurs de modifier leur mot de passe.

Fonctionnalités :

- L'utilisateur peut demander la réinitialisation de son mot de passe en entrant son adresse email.

- Un lien de réinitialisation est envoyé à l'utilisateur par e-mail.
- L'utilisateur clique sur le lien, accède à une page de réinitialisation et entre un nouveau mot de passe valide selon les règles syntaxiques mises en place.

## **GESTION DE PROFIL**

Objectif : Permettre aux utilisateurs de gérer leur compte et de mettre à jour leurs informations personnelles.

Fonctionnalités :

- Possibilité de se déconnecter de son compte.
- Possibilité d'afficher les informations personnelles de l'utilisateur.
- Possibilité de modifier les informations personnelles (adresse email, mot de passe, description personnelle).

## **COMMUNICATION**

Objectif : Permettre les échanges entre les élèves et les tuteurs depuis la plateforme.

Fonctionnalités :

- Système de messagerie intégré permettant l'envoi et la réception de messages.
- Possibilité d'épingler des messages importants pour un accès facile ultérieur.
- Notifications pour les messages non lus.

## **PLANIFICATION DES RENCONTRES ÉLÈVE-TUTEUR**

Objectif : Permettre la planification et la gestion des rendez-vous entre élèves et tuteurs.

Fonctionnalités :

- Page de calendrier affichant les événements et rendez-vous de l'utilisateur.
- Possibilité pour les tuteurs de créer et de planifier des sessions de tutorat avec date, heure et description.
- Possibilité pour les élèves de visualiser leur emploi du temps sur le calendrier.
- Possibilité pour les tuteurs uniquement de modifier un rendez-vous (annulation ou modification de date/heure).

## **GESTION DES TÂCHES**

Objectif : Création et suivi des tâches à réaliser pour les élèves et tuteurs.

Fonctionnalités :

- Les tuteurs peuvent créer des tâches et les attribuer à leurs élèves.
- Les utilisateurs peuvent se créer des tâches personnelles.
- Notifications pour les nouvelles tâches assignées par les tuteurs aux élèves.

# Veille technologique

## 1. Simplicité et Synergie des Technologies

### a. JavaScript Universel (Node.js)

Le fait que Node.js utilise JavaScript, tout comme React pour le front-end, offre une grande cohérence dans le développement. Les développeurs peuvent ainsi utiliser un seul langage de programmation sur toute la pile technologique, ce qui simplifie l'apprentissage et réduit les contextes de switch entre différents langages.

### b. Express.js - Simplification du Serveur

Express.js est un framework minimaliste pour Node.js qui facilite la création de serveurs web. Sa simplicité et flexibilité permettent de développer des applications robustes avec moins de code et plus d'efficacité, grâce à un riche écosystème de middleware pour répondre à divers besoins de l'application.

### c. MongoDB - Flexibilité des Données

MongoDB est une base de données NoSQL qui offre une grande flexibilité dans la gestion des données. Elle permet une scalabilité horizontale et est conçue pour gérer de grands volumes de données de manière efficace, avec un schéma dynamique qui est idéal pour les applications modernes qui nécessitent une évolution rapide.

### d. ReactJS - Composants et Hooks

ReactJS favorise la maintenabilité et la scalabilité grâce à son système de composants réutilisables et son modèle de hooks. Ces derniers permettent de gérer l'état et le cycle de vie des composants de manière plus intuitive et fonctionnelle, rendant le code plus lisible et plus facile à maintenir.

## 2. Communauté Solide et Ressources Abondantes

Le stack MERN bénéficie d'une communauté très active et engagée. Que ce soit sur GitHub, Stack Overflow, ou d'autres forums spécialisés, il est facile de trouver de l'aide et de partager des connaissances.

- **Développeurs et contributeurs :** Les projets sous-jacents sont maintenus par des milliers de développeurs.
- **Ressources éducatives :** Il existe de nombreux cours, tutoriels, et guides pratiques

# Spécifications techniques

## **Introduction**

Le projet HomeSkolar a pour objectif de créer une plateforme web facilitant le tutorat scolaire en ligne. Les spécifications techniques détaillées ci-dessous permettent de guider le développement et la maintenance de cette plateforme.

## **2. Fonctionnalités Techniques**

### **INSCRIPTION**

Objectif : Permettre aux utilisateurs (élèves et tuteurs) de s'inscrire à la plateforme.

Backend :

- Endpoints API : Création de comptes utilisateur avec validation des données.
- Base de données : Stockage sécurisé des informations utilisateur.

Frontend :

- Formulaire d'inscription avec validation côté client.

### **AUTHENTIFICATION**

Objectif : Permettre aux utilisateurs de se connecter à la plateforme.

Backend :

- Endpoints API : Authentification des utilisateurs avec génération de JWT.
- Sécurité : Chiffrement des mots de passe.

Frontend :

- Formulaire de connexion avec gestion des tokens JWT pour maintenir la session utilisateur.

### **MODIFICATION DE MOT DE PASSE**

Objectif : Permettre aux utilisateurs de modifier leur mot de passe.

Backend :

- Endpoints API : Réinitialisation des mots de passe avec envoi de lien par email.

Frontend :

- Formulaire de réinitialisation de mot de passe avec validation des champs.

## **GESTION DE PROFIL**

Objectif : Permettre aux utilisateurs de gérer leur compte et de mettre à jour leurs informations personnelles.

Backend :

- Endpoints API : Récupération et mise à jour des informations de profil.

Frontend :

- Interface de gestion de profil pour afficher et modifier les informations personnelles.

## **COMMUNICATION**

Objectif : Permettre les échanges entre les élèves et les tuteurs depuis la plateforme.

Backend :

- Endpoints API : Gestion des messages.
- WebSockets : Communication en temps réel.

Frontend :

- Interface de messagerie avec support pour épingler des messages et notifications pour les nouveaux messages.

## **PLANIFICATION DES RENCONTRES ÉLÈVE-TUTEUR**

Objectif : Permettre la planification et la gestion des rendez-vous entre élèves et tuteurs.

Backend :

- Endpoints API : Création, récupération et modification des événements de calendrier.

Frontend :

- Interface de planification avec vue calendrier interactive.

## **GESTION DES TÂCHES**

Objectif : Création et suivi des tâches à réaliser pour les élèves et tuteurs.

Backend :

- Endpoints API : Gestion des tâches avec création et mise à jour des tâches.

Frontend :

- Interface de gestion des tâches pour afficher, créer et mettre à jour les tâches.
- qui couvrent chaque technologie du stack MERN.
- Documentation à jour : Chaque technologie du stack MERN dispose d'une documentation officielle complète, régulièrement mise à jour, avec des guides de démarrage rapide, des références API, et des exemples de code.

### 3. Documentation de Qualité

La qualité de la documentation est cruciale pour l'apprentissage et l'utilisation efficace de nouvelles technologies. MongoDB, Express, ReactJS, et NodeJS offrent toutes des documentations bien organisées, claires et détaillées. Elles incluent des tutoriels pour débutants, des descriptions approfondies des fonctionnalités, et des guides de meilleures pratiques qui sont essentiels pour les développeurs de tous niveaux.

**Conclusion** Le stack MERN se distingue par sa simplicité, une grande communauté, et d'excellentes documentations. Ces éléments contribuent à un cycle de développement efficace et à une expérience d'apprentissage enrichissante, ce qui fait de MERN un choix populaire pour le développement d'applications modernes, performantes et scalables.

## Diagramme de classes

