

Cahier des Charges pour le Système de Gestion de FabLab

Introduction

Ce cahier des charges détaille le développement d'un système de gestion pour un FabLab utilisant React, Node.js, MongoDB, avec intégration de l'API ChatGPT et des fonctionnalités de QR Code. Ce système comprendra des espaces utilisateurs et administrateurs, la gestion et l'inventaire des équipements, la gestion des abonnements et des prêts, des rapports d'exploitation, un historique personnel pour les utilisateurs, une base de données de démonstration, et d'autres fonctionnalités distinctives.

1. Technologies et Outils

- **Frontend:** React.js
- **Backend:** Node.js avec Express.js
- **Base de données:** MongoDB
- **Authentification:**
- **API ChatGPT:** Intégration pour le support utilisateur
- **QR Code:** Génération et lecture via une application mobile
- **Autres outils:** Git

2. Fonctionnalités

2.1 Espace Utilisateur et Administrateur

- **Technologie:** React pour le frontend, Node.js pour le backend
- **Description:** Interface distincte pour les utilisateurs et les administrateurs avec des droits et des fonctionnalités spécifiques.

2.2 Gestion et Inventaire des Équipements

- **Technologie:** React, Node.js, MongoDB
- **Description:** Système pour ajouter, modifier, supprimer et visualiser les équipements. Chaque équipement aura une photo, une description, un statut (disponible, en maintenance, etc.).

2.3 Gestion des Abonnements et des Prêts

- **Technologie:** Node.js, MongoDB
- **Description:** Gérer les abonnements des utilisateurs, les prêts d'équipements, les dates de retour prévues, et les historiques de prêts.

2.4 Gestion par QR Code

- **Technologie:** Bibliothèques QR Code pour React et Node.js
- **Description:** Utilisation de QR Codes pour faciliter l'emprunt et le retour des équipements. Les utilisateurs peuvent scanner les codes via une application mobile.

2.5 Rapports d'Exploitation

- **Technologie:** Node.js, MongoDB
- **Description:** Génération de rapports sur l'utilisation des équipements, les prêts, les retards, etc.

2.6 Historique Personnel pour les Utilisateurs

- **Technologie:** Node.js, MongoDB
- **Description:** Chaque utilisateur peut accéder à son historique personnel de prêts, abonnements, et interactions avec le FabLab.

2.7 Base de Données de Démonstration

- **Technologie:** MongoDB
- **Description:** Création d'une base de données de démonstration pour tester et présenter le système.

2.8 Autres Fonctionnalités

- **Intégration de l'API ChatGPT:** A DETERMINER
- **Ajout d'Articles:** Système de gestion de contenu pour publier des articles, des tutoriels, etc.

3. Développement et Déploiement

3.1 Conception

- **Wireframes et Maquettes:** Figma ?
- **Architecture du Système:** MVC

3.2 Développement

- **Environnement de Développement:** Configuration de l'environnement avec Node.js, React, et MongoDB.

3.3 Tests

- **Tests Unitaires:** Utilisation de Jest pour les tests unitaires.

- **Tests d'Intégration et de Performance:** Tests réguliers pour assurer la qualité et la performance.