

DOCUMENT DE SPÉCIFICATION DU PROJET :

Plateforme de gestion d'événements communautaires

1. Aperçu du projet

1.1 Introduction

La plateforme de gestion d'événements communautaires est une application Web conçue pour rationaliser l'organisation, la gestion et la promotion des événements communautaires. Elle vise à simplifier la logistique pour les organisateurs d'événements et à améliorer l'expérience des participants. La plateforme prend en charge la création, l'inscription, la communication et l'événement. Analytics, fournissant un outil complet pour les organisateurs et les participants.

1.2 Objectifs

- Simplifiez le processus de création et de gestion d'événements pour les organisateurs communautaires.
- Améliorez l'expérience utilisateur en rendant la découverte et l'inscription des événements transparentes.
- Fournir des outils de communication robustes pour favoriser l'engagement entre.
- Offrez des analyses pertinentes aux organisateurs pour une prise de décision basée sur les données.

1.3 Public cible

- **Organisateurs d'événements** : personnes ou organisations responsables de la planification et de la gestion d'événements communautaires.
- **Participants** : membres de la communauté intéressés à assister et à s'impliquer dans des événements locaux.

2.Exigences fonctionnelles

2.1 Enregistrement et authentification de l'utilisateur

2.1.1 S'inscrire et se connecter

- Les utilisateurs peuvent créer un compte en utilisant une adresse e-mail et un mot de passe.
- * Récupération de mot de passe par e-mail.

2.1.2 Gestion des profils

- Les utilisateurs peuvent afficher et modifier les informations de leur profil, y compris leur nom, leurs coordonnées et leurs préférences.
- Les photos de profil peuvent être téléchargées et mises à jour.

2.1.3 Rôles des utilisateurs

- **Créateur** : Accès complet pour créer, gérer et supprimer les événements qui leur sont liés. Peut afficher toutes les analyses concernant leurs événements.
- **Participants** : peuvent parcourir, s'inscrire et participer à des événements. Accès limité aux outils de communication et aux analyses personnelles.

2.2 Création et gestion d'événements

2.2.1 Créer des événements

- **Détails de l'événement** : nom, description, date, heure, lieu et capacité de l'événement.
- **Billetterie** : Possibilité de créer des événements gratuits ou payants. Pour les événements payants, intégrez Stripe ou PayPal pour le traitement des paiements.
- **Formulaires personnalisables** : permettez aux organisateurs de créer des formulaires d'inscription personnalisés avec des questions spécifiques pour les participants.

2.2.2 Gérer les événements

- **Modifier les événements** : les organisateurs peuvent mettre à jour les détails de l'événement, gérer les listes d'inscription et ajuster la capacité.
- **Annulation d'événement** : possibilité d'annuler des événements, en informant automatiquement tous les participants inscrits.

2.3 Inscription à l'événement

2.3.1 Parcourir et s'inscrire

- **Liste des événements** : affichez tous les événements à venir avec des options de filtrage basées sur la date, le lieu, le type et les préférences de l'utilisateur.
- **Page de détails de l'événement** : informations détaillées sur l'événement, y compris les coordonnées de l'organisateur, le calendrier et la carte de localisation.
- **Inscription** : Processus d'inscription simple avec e-mails de confirmation automatisés.

2.3.2 Gestion des sièges

- **Gestion de la capacité** : fermez automatiquement l'inscription lorsque l'événement atteint sa capacité.

* 2.4 Intégration du calendrier

2.4.1 Vue du calendrier

- **Vue mensuelle/hebdomadaire** : calendrier visuel sur la plateforme affichant tous les événements qui intéressent ou auxquels l'utilisateur est inscrit.

2.5 Notifications et rappels

2.5.1 Notifications d'événements à venir

- **Notifications push** : envoyez des rappels aux utilisateurs sur les événements à venir par e-mail, SMS ou notifications dans l'application.
- **Mises à jour d'événements** : informez les utilisateurs de tout changement (par exemple, heure, lieu) via leur canal de communication préféré.

2.6 Communauté et communication

2.6.1 Forums de discussion

- **Tableaux spécifiques à l'événement** : chaque événement dispose d'un forum de discussion dédié sur lequel les participants peuvent publier et participer à des discussions.
- **Outils de modération** : les organisateurs peuvent modérer les discussions, épingler des messages importants et gérer la participation des utilisateurs.

2.6.2 Messagerie directe

- **Messagerie privée** : Permet une communication directe entre organisateurs et participants ou entre participants.
- **Discussions de groupe** : activez les discussions de groupe pour des événements spécifiques ou des groupes d'intérêt au sein de la communauté.

2.7 Tableau de bord d'analyse

2.7.1 Informations sur les événements

- **Données d'inscription** : suivez le nombre d'inscriptions, les données démographiques et les niveaux d'engagement.
- **Suivi des revenus** : pour les événements payants, suivez les revenus totaux, les ventes de billets et l'état du paiement.

2.7.2 Outils de commentaires

- **Enquêtes post-événement** : envoyez automatiquement des enquêtes aux participants pour recueillir des commentaires.
- **Analyse des sentiments** : analysez les réponses à l'enquête pour évaluer la satisfaction globale et identifier les domaines à améliorer.

2.8 Recherche et filtres

2.8.1 Recherche d'événement

- **Recherche par mot clé** : les utilisateurs peuvent rechercher des événements par mots clés liés au titre, à la description ou au lieu de l'événement.
- **Filtres avancés** : Filtres par date, catégorie, lieu, type de billet (gratuit/payant).

2.9 Conception réactive

2.9.1 Interface adaptée aux mobiles

- **Conception réactive** : assurez-vous que la plateforme est entièrement réactive, offrant une expérience optimale sur tous les types d'appareils (ordinateur de bureau, tablette, mobile).
- **Navigation tactile** : optimisez l'interface utilisateur pour les interactions tactiles sur les appareils mobiles.

2.9.2 Applications Web progressives (PWA)

- **Expérience de type application** : PWA fournira une expérience native de type application avec des transitions fluides et des temps de chargement rapides.

3. Exigences non fonctionnelles

3.1 Performances

- La plateforme doit gérer l'accès simultané de plusieurs utilisateurs sans dégradation des performances.
- Les temps de chargement des listes d'événements et des pages détaillées doivent être minimisés (objectif < 2 secondes).

3.2 Évolutivité

- L'application doit être conçue pour évoluer horizontalement afin de gérer un nombre croissant d'utilisateurs et d'événements.
- L'infrastructure basée sur le cloud (par exemple, AWS, Heroku) sera utilisée pour permettre une mise à l'échelle dynamique basée sur le trafic.

3.3 Sécurité

- *** Cryptage des données** : toutes les données utilisateur, en particulier les informations sensibles telles que les mots de passe et les détails de paiement, doivent être cryptées à la fois en transit et au repos.
- **Authentification** : utilisez JWT pour l'authentification sécurisée des utilisateurs et la gestion des sessions.
- **Contrôle d'accès basé sur les rôles (RBAC)** : assurez-vous que les utilisateurs n'ont accès qu'aux fonctionnalités et aux données appropriées à leurs rôles.

3.4 Utilisabilité

- **Interface intuitive** : La plateforme doit être conviviale, avec une navigation claire et un langage accessible.
- *** Accessibilité** : La plateforme doit répondre aux normes WCAG 2.1, garantissant qu'elle est accessible aux utilisateurs handicapés.

3.5 Fiabilité

- **Disponibilité** : La plateforme doit être disponible 99,9% du temps, avec des périodes de maintenance programmées clairement communiquées aux utilisateurs.
- **Sauvegarde et récupération** : mettez en œuvre des sauvegardes régulières et un plan de reprise après sinistre pour éviter la perte de données.

4. Pile technique

4.1 Front-end

- **React.js** : Pour créer l'interface utilisateur dynamique et réactive.
- **HTML/CSS** : Pour structurer et styliser les pages Web.
- **HookState** : Pour gérer l'état global de l'application.

4.2 Back-end

- **Node.js avec Express.js** : pour gérer la logique côté serveur et créer des API RESTful.
- **JWT** : Pour l'authentification et l'autorisation des utilisateurs.

4.3 Base de données

- **PostgreSQL** : pour stocker des données relationnelles telles que des profils d'utilisateurs, des détails d'événements et des enregistrements de transactions.
- **MongoDB** : utilisation facultative pour stocker des données non relationnelles, telles que des journaux ou de grands ensembles de données.

* 4.4 Intégration des paiements

- **Stripe/PayPal** : Pour gérer le traitement sécurisé des paiements pour les événements payants.

4.5 Hébergement

- **Heroku/AWS** : Pour l'hébergement de l'application, avec des options de mise à l'échelle et de déploiement.

5. Plan de développement

5.1 Jalon

1. **Jour 1** : Collecte des exigences, configuration initiale de l'environnement de développement et conception du schéma de base de données.
2. **Jour 2-3** : Enregistrement et authentification des utilisateurs, fonctionnalités de création et de gestion d'événements (API)
3. **Jour 4-6** : Inscription aux événements, intégration du calendrier et notifications (API)
4. **Jour 7-8** : Fonctionnalités communautaires, tableau de bord d'analyse (API).
5. **Jour 9-11** : Conception du frontend, mise en œuvre du design et fonctionnalité de recherche
6. **Jour 11-13** : Ajout d'événements recommandés basés sur l'IA, correction de bugs et déploiement.
7. **Jour 14** : Présentation finale

5.2 Déploiements

- **Environnement de staging** : déployez l'application dans un environnement de staging pour les tests finaux.
- **Déploiement en production** : lancez l'application dans l'environnement de production, avec une surveillance continue de tout problème.

6.Maintenance et assistance

6.1 Assistance post-lancement

- **Correction de bugs** : résolvez tous les problèmes qui surviennent après le lancement.
- **Améliorations des fonctionnalités** : recueillez les commentaires des utilisateurs sur les futures améliorations des fonctionnalités.
- **Surveillance du système** : surveillance continue des performances, de la sécurité et de la disponibilité.

Glossaire

- **IA (Intelligence Artificielle)** : Technologie qui permet à un système d'effectuer des tâches qui nécessitent généralement l'intelligence humaine, comme recommander des événements en fonction du comportement de l'utilisateur.
- **Tableau de bord analytique** : un outil fournissant des représentations visuelles des données liées aux événements, notamment les numéros d'inscription, les revenus et les commentaires des participants.
- **Intégration du calendrier** : fonctionnalité qui synchronise les dates des événements avec une vue du calendrier, permettant aux utilisateurs de suivre et de gérer leurs événements.
- **Gestion de communauté** : processus de supervision et de facilitation des interactions au sein d'une plate-forme communautaire, y compris la gestion des discussions et des activités des utilisateurs.
- **Créateur** : un rôle d'utilisateur avec un accès complet pour créer, gérer et supprimer des événements, ainsi que pour afficher les analyses liées à leurs événements.
- **Messagerie directe** : fonctionnalité de communication permettant des conversations privées entre utilisateurs, par exemple entre organisateurs et participants.
- **Forums de discussion** : forums en ligne sur lesquels les utilisateurs peuvent publier et participer à des conversations liées à des événements spécifiques.
- **Création et gestion d'événements** : fonctionnalité permettant aux organisateurs de créer de nouveaux événements, de gérer ceux existants et de gérer les tâches liées à l'événement telles que l'inscription et les annulations.
- **Inscription à l'événement** : processus par lequel les participants s'inscrivent pour assister à des événements, notamment en parcourant les listes d'événements et en confirmant leur participation.
- **JWT (JSON Web Token)** : Une méthode sécurisée pour l'authentification des utilisateurs et la gestion des sessions dans les applications Web.
- **Notifications et rappels** : alertes envoyées aux utilisateurs concernant les événements à venir ou les modifications apportées aux détails de l'événement, envoyées par e-mail, SMS ou notifications dans l'application.
- **PWA (Progressive Web App)** : application Web conçue pour offrir une expérience native semblable à celle d'une application, notamment des temps de chargement rapides et des transitions fluides.
- **Conception réactive** : approche de conception garantissant que la plate-forme fonctionne correctement sur différentes tailles d'appareils, notamment les ordinateurs de bureau, les tablettes et les téléphones mobiles.
- **Contrôle d'accès basé sur les rôles (RBAC)** : modèle de sécurité qui restreint l'accès au système en fonction du rôle de l'utilisateur, garantissant un accès approprié aux fonctionnalités et aux données.
- **Évolutivité** : capacité de l'application à gérer un nombre croissant d'utilisateurs et d'événements sans dégradation des performances.

- **Stripe/PayPal** : Services de traitement des paiements intégrés à la plateforme pour gérer les transactions pour les événements payants.
- **Rôles des utilisateurs** : niveaux d'accès définis et autorisations attribuées aux utilisateurs, tels que les créateurs et les participants, déterminant leurs capacités au sein de la plateforme.
- **WCAG (Web Content Accessibility Guidelines) 2.1** : Normes garantissant que le contenu Web est accessible aux personnes handicapées.
- **Webhook** : un moyen permettant aux applications de fournir des informations en temps réel à d'autres applications via des rappels HTTP.
- **HookState** : une bibliothèque utilisée pour la gestion des états dans les applications React, fournissant une solution globale de gestion des états.
- **PostgreSQL** : une base de données relationnelle open source utilisée pour stocker des données structurées pour la plateforme.
- **MongoDB** : base de données NoSQL facultative utilisée pour stocker des données non relationnelles, telles que des journaux ou de grands ensembles de données.
- **Heroku/AWS** : plates-formes basées sur le cloud utilisées pour l'hébergement et la mise à l'échelle d'applications Web.

Table des matières

1. Aperçu du projet

- **Introduction**
- **Objectifs**
- **Public cible**

2. Exigences fonctionnelles

- **Enregistrement et authentification des utilisateurs**
- **Création et gestion d'événements**
- **Inscription à l'événement**
- **Intégration du calendrier**
- **Notifications et rappels**
- **Communauté et communication**
- **Tableau de bord d'analyse**
- **Recherche et filtres**
- **Conception réactive**

3. Exigences non fonctionnelles

4. Pile technique

5. Plan de développement

6. Entretien et assistance

7. Glossaire