

Table des matières

Cahier des charges	2
Application sociale communautaire pour développeurs	2
1. Présentation générale du projet	2
1.1 Contexte et vision	2
1.2 Objectifs du projet.....	2
1.3 Public cible	2
2. Fonctionnalités détaillées.....	2
2.1 Authentification et gestion des comptes	2
2.2 Profil utilisateur	2
2.3 Système de publications (Posts)	3
2.4 Sondages (Polls)	3
2.5 Quiz techniques.....	3
2.6 Communautés de développeurs.....	4
2.7 Événements.....	4
2.8 Système de notifications.....	4
2.9 Recherche et découverte	4
3. Exigences techniques	5
3.1 Frontend	5
3.2 Backend	5
3.3 Base de données.....	5
3.4 Stockage et médias	5
3.5 Sécurité	5
4. Contraintes et qualité	5
5. Évolutions futures.....	5
6. Planning prévisionnel	5
7. Livrables.....	6
8. Conclusion	6

Cahier des charges

Application sociale communautaire pour développeurs

1. Présentation générale du projet

1.1 Contexte et vision

Le projet consiste à concevoir et développer une application sociale communautaire dédiée aux développeurs. Cette plateforme a pour vocation de devenir un espace central où les développeurs peuvent partager du contenu technique, interagir, apprendre ensemble, participer à des événements et valoriser leur profil professionnel.

L'application se positionne comme un réseau social spécialisé, combinant les fonctionnalités de partage (posts, médias), d'apprentissage (quiz, sondages), de collaboration (communautés) et de visibilité professionnelle (intégration GitHub).

1.2 Objectifs du projet

- Créer une communauté active et engagée de développeurs
- Faciliter le partage de connaissances techniques
- Encourager l'apprentissage continu via des contenus interactifs
- Mettre en valeur les profils techniques des utilisateurs
- Favoriser la collaboration à travers des communautés et des événements

1.3 Public cible

- Développeurs débutants, intermédiaires et expérimentés
- Étudiants en informatique et nouvelles technologies
- Créateurs de contenu technique
- Communautés tech (frontend, backend, mobile, DevOps, data, etc.)

2. Fonctionnalités détaillées

2.1 Authentification et gestion des comptes

- Inscription via email et mot de passe
- Connexion sécurisée
- Authentification via GitHub (OAuth)
- Gestion de session (JWT et cookies)
- Réinitialisation du mot de passe
- Gestion des rôles :

2.2 Profil utilisateur

- Photo de profil et image de couverture
- Nom d'utilisateur unique
- Bio / description personnelle

- Localisation (optionnelle)
- Compétences techniques (tags)
- Statistiques du profil :
 - Nombre de posts
 - Nombre de likes reçus
 - Communautés rejointes

Intégration GitHub

- Lien avec le compte GitHub
- Affichage du nom d'utilisateur GitHub
- Liste des repositories publics
- Informations affichées :
 - Nom du repository
 - Description
 - Langage principal
 - Nombre de stars et forks
- Option de visibilité (activer/désactiver l'affichage des repos)

2.3 Système de publications (Posts)

- Création de posts texte
- Publication d'images
- Publication de vidéos
- Ajout de tags technologiques
- Modification et suppression de posts
- Fil d'actualité personnalisé

Interactions

- Likes / réactions
- Commentaires
- Partage interne
- Mentions d'utilisateurs (@username)

2.4 Sondages (Polls)

- Création de sondages intégrés aux posts
- Choix simple ou multiple
- Durée limitée configurable
- Affichage des résultats en temps réel
- Restriction à un vote par utilisateur

2.5 Quiz techniques

- Création de quiz par les utilisateurs ou administrateurs
- Utiliser l'IA pour générer un quiz
- Questions à choix multiples

- Score automatique
- Affichage du résultat final
- Historique des quiz passés

2.6 Communautés de développeurs

- Création de communautés par thème ou technologie
- Description et règles de la communauté
- Rejoindre / quitter une communauté
- Fil de posts spécifique à chaque communauté
- Gestion des rôles dans la communauté :
 - Admin
 - Modérateur
 - Membre

2.7 Événements

- Création d'événements techniques.
- Informations d'un événement :
 - Titre
 - Description
 - Date et heure
 - Lieu (physique ou en ligne)
 - Technologies concernées
- Inscription / participation à un événement
- Liste des participants
- Historique des événements passés

2.8 Système de notifications

- Notifications en temps réel ou différées :
 - Likes
 - Commentaires
 - Mentions
 - Invitations à des communautés
 - Rappels d'événements

2.9 Recherche et découverte

- Recherche d'utilisateurs
- Recherche de posts
- Recherche de communautés
- Filtres par tags et technologies

3. Exigences techniques

3.1 Frontend

- Framework : React ou Next.js
- Style : tailwind css
- Design responsive (mobile / desktop)
- Interface moderne et minimaliste

3.2 Backend

- API REST
- Node.js avec Express
- Authentification sécurisée (JWT, OAuth)
- Intégration API GitHub

3.3 Base de données

- Mongo db

3.4 Stockage et médias

- Stockage cloud pour images et vidéos
- Optimisation des performances médias

3.5 Sécurité

- Hashage des mots de passe (bcrypt)
- Validation des données

4. Contraintes et qualité

- Application scalable
- Code maintenable et documenté
- Performances optimisées
- Respect de la confidentialité des données

5. Évolutions futures

- Messagerie privée
- Système de badges et gamification
- Monétisation (premium, sponsoring d'événements)
- API publique pour intégrations externes

6. Planning prévisionnel

1. Analyse et conception
2. Design UI/UX
3. Développement MVP
4. Tests et validation

5. Déploiement

7. Livrables

- Application web fonctionnelle
- Code source frontend et backend
- Documentation technique

8. Conclusion

Ce projet ambitionne de créer une plateforme communautaire moderne dédiée aux développeurs, combinant réseau social, apprentissage et collaboration, avec une forte orientation sur la valeur technique et la communauté.