

JUNBI

Cahier des Charges

Atanassov Julien

FSD48A



Sommaire

1. Présentation	p3
2. Public cible	p5
3. Besoins et objectifs à atteindre	p7
4. Les avantages attendus	p11
5. Fonctionnalités attendues	p12
6. Exigences techniques	p15
7. Contraintes et limites du projet	p21
8. Référencement SEO	p24
9. Notre équipe	p26
10. Compétences requises	p28
11. Planning	p29
12. Budget	p30
13. Accessibilité	p31
14. Adaptabilité de l'application	p32
15. Eco conception	p33
16. RGPD	p34
17. Langages et technologies utilisés	p35
18. Veilles techniques	p37
19. Outils collaboratifs	p38
20. Sécurité	p39
21. MCD	p40
22. Navigation de l'application	p41
23. Améliorations futures	p42
24. Les livrables	p43
25. Annexes	p44



1. Présentation

1.1 Contexte général et Enjeux

Dans un monde où les interactions sociales évoluent constamment vers le numérique, l'organisation d'événements nécessite des outils adaptés et performants. L'évolution des interactions sociales en ligne a créé un besoin croissant pour des plateformes capables de gérer des événements de manière efficace et sécurisée.

JUNBI émerge comme une solution innovante pour répondre à ces besoins, en proposant une plateforme complète de gestion d'événements qui simplifie et optimise chaque étape du processus d'organisation.

1.2 Objet de l'Application

JUNBI est une plateforme numérique destinée à faciliter la gestion d'événements en permettant aux utilisateurs de :

- Créer et gérer des événements privés ou publics
- Faciliter la communication entre les participants
- Intégrer des fonctionnalités avancées de gestion (invitations, dépenses partagées, messagerie instantannée)
- Offrir une expérience utilisateur fluide et intuitive

1.3 Vision du Projet

Notre vision s'articule autour de deux axes principaux :

1. Simplicité

- Interface intuitive et accessible
- Expérience utilisateur optimisée

2. Communauté

- Facilitation des interactions sociales
- Création de liens entre les participants

1.4 Objectifs Principaux

Les objectifs fondamentaux de JUNBI sont :

1. **Efficacité Organisationnelle**

- Centralisation des informations événementielles
- Gestion simplifiée des participants

2. **Communication Optimisée**

- Système de messagerie intégré
- Partage facile des informations

3. **Sécurité et Fiabilité**

- Protection des données personnelles
- Gestion sécurisée des accès
- Stabilité de la plateforme



2. Public Cible

L'analyse détaillée du public cible est essentielle pour garantir que JUNBI réponde précisément aux besoins et aux attentes de ses utilisateurs. Notre application s'adresse à un large éventail d'utilisateurs, chacun ayant des besoins spécifiques en matière de gestion d'événements.

2.1 Segmentation des Utilisateurs

La diversité des utilisateurs potentiels de JUNBI peut être catégorisée en plusieurs segments principaux, chacun avec ses propres caractéristiques et besoins.

2.1.1 Particuliers

Le segment des particuliers représente une part importante de notre public cible :

Organisateurs Occasionnels

- Personnes organisant des événements familiaux (anniversaires, réunions de famille)
- Individus planifiant des rencontres entre amis
- Utilisateurs organisant des activités de loisirs ponctuelles

Utilisateurs Réguliers

- Membres actifs d'associations
- Organisateurs de clubs de sport ou de loisirs
- Responsables de groupes communautaires
- Planificateurs d'activités récurrentes

2.1.2 Professionnels

Le segment professionnel inclut différents types d'organisations :

Entreprises

- Départements des ressources humaines
- Équipes événementielles
- Services marketing et communication
- Gestionnaires de projets

Organisations

- Associations à but non lucratif
- Établissements d'enseignement
- Organismes culturels
- Structures sportives

2.2 Analyse des Besoins par Segment

Chaque segment du public cible présente des besoins spécifiques qui doivent être adressés par notre solution.

2.2.1 Besoins des Particuliers

Les particuliers recherchent principalement :

Simplicité et Accessibilité

- Accès rapide aux fonctionnalités essentielles
- Support mobile

Fonctionnalités Sociales

- Interaction avec les participants
- Gestion des confirmations de présence

2.2.2 Besoins des Professionnels

Les utilisateurs professionnels nécessitent :

Outils Avancés

- Fonctionnalités de reporting
- Gestion des droits d'accès

Sécurité et Conformité

- Respect des normes RGPD
- Gestion des autorisations



3. Besoins et Objectifs à Atteindre

La réussite de JUNBI repose sur une compréhension approfondie des besoins des utilisateurs et sur la définition d'objectifs clairs et mesurables. Cette section détaille l'ensemble des exigences fonctionnelles et techniques qui guideront le développement de la plateforme.

3.1 Besoins Fonctionnels Fondamentaux

Les besoins fonctionnels constituent le cœur de l'application et définissent les capacités essentielles que JUNBI doit offrir à ses utilisateurs.

3.1.1 Gestion des Utilisateurs et des Profils

Système d'Authentification

- Inscription simplifiée
- Gestion des sessions utilisateur

Gestion des Profils

- Création et personnalisation des profils utilisateurs
- Upload et gestion des photos de profil
- Paramètres de confidentialité configurables
- Historique des activités et des événements

Système de Rôles et Permissions

- Hiérarchie claire des rôles (administrateur, organisateur, participant)
- Permissions personnalisables par événement
- Système de délégation des droits
- Audit trail des modifications de permissions

3.1.2 Gestion des Événements

Création d'Événements

- Interface intuitive de création d'événements
- Système de duplication d'événements
- Options de personnalisation

Administration des Événements

- Tableau de bord de gestion centralisé
- Outils de planification et de scheduling
- Gestion des inscriptions et des listes d'attente

Gestion des Participants

- Import/export des listes de participants
- Statistiques de participation en temps réel

3.2 Objectifs Techniques

Les objectifs techniques définissent les standards de performance et de qualité que JUNBI doit atteindre.

3.2.1 Performance

Objectifs de Performance

- Temps de chargement initial < 2 secondes
- Temps de réponse des actions < 500ms
- Disponibilité de 99.9%

3.2.2 Sécurité et Protection des Données

Sécurité Applicative

- Chiffrement des données en transit (TLS 1.3)
- Protection contre les injections SQL
- Défense contre les attaques XSS et CSRF
- Validation stricte des entrées utilisateur

Protection des Données

- Conformité RGPD complète
- Chiffrement des données sensibles
- Politique de rétention des données

3.3 Objectifs d'Expérience Utilisateur

L'expérience utilisateur est au cœur de la conception de JUNBI, avec des objectifs précis à atteindre.

3.3.1 Interface Utilisateur

Design et Ergonomie

- Interface responsive sur tous les appareils
- Navigation intuitive et cohérente

Accessibilité

- Conformité WCAG 2.1 niveau AA
- Support des lecteurs d'écran
- Contraste et lisibilité optimisés

3.3.2 Performances Perçues

Réactivité

- Feedback immédiat des actions utilisateur
- Animations fluides et optimisées
- Chargement progressif des données
- État de chargement explicite

Fiabilité

- Gestion des erreurs
- Restauration de session

3.4 Évolution et Maintenance

La pérennité de JUNBI repose sur sa capacité à évoluer et à s'améliorer continuellement.

3.4.1 Maintenance Prédictive

Monitoring

- Surveillance des performances
- Maintenance préventive régulière

Mises à Jour

- Déploiement continu (CD/CI)
- Tests automatisés
- Rollback automatique en cas d'incident



4. Les Avantages Attendus

Cette section présente les principaux bénéfices que JUNBI apportera à ses utilisateurs.

4.1 Avantages Pratiques

Pour les Organisateurs

- Interface unique et intuitive pour gérer les événements
- Moins d'erreurs dans la planification

Pour les Participants

- Inscription simplifiée aux événements
- Vue claire des événements à venir
- Accès facile sur mobile et ordinateur

4.2 Bénéfices pour la Communication

- Messagerie intégrée pour échanger facilement
- Centralisation des informations importantes
- Meilleur suivi des participations



5. Fonctionnalités Attendues

Cette section détaille l'ensemble des fonctionnalités que JUNBI devra implémenter pour répondre aux besoins identifiés.

5.1 Gestion des Utilisateurs

Le système de gestion des utilisateurs constitue la base de la plateforme et détermine l'accès aux différentes fonctionnalités.

5.1.1 Système d'Authentication

Inscription et Connexion

- Inscription avec email et mot de passe
- Connexion sécurisée via email/mot de passe

Gestion des Sessions

- Sessions sécurisées avec tokens JWT
- Déconnexion automatique après période d'inactivité

5.1.2 Gestion des Profils

Informations Utilisateur

- Profil personnel personnalisable
- Upload et gestion de photo de profil
- Informations de contact (email, téléphone)
- Historique des événements

Paramètres de Confidentialité

- Gestion des informations partagées
- Options de blocage d'utilisateurs

5.2 Gestion des Événements

Le cœur fonctionnel de JUNBI réside dans sa capacité à gérer efficacement les événements.

5.2.1 Création d'Événements

Interface de Création

- Options de personnalisation

Configuration des Événements

- Paramétrage des dates et horaires
- Gestion des fuseaux horaires
- Configuration de la capacité d'accueil

5.2.2 Gestion des Participants

Système d'Invitation

- Envoi d'invitations groupées
- Suivi des statuts des invitations
- Gestion des listes d'attente

Suivi des Inscriptions

- Tableau de bord des participants
- Export des listes de participants
- Statistiques en temps réel

5.3 Communication et Interaction

Les outils de communication sont essentiels pour assurer le succès des événements.

5.3.1 Messagerie Intégrée

Chat en Temps Réel

- Conversations groupées

5.3.2 Outils Collaboratifs

Planification Collaborative

- Votes et enquêtes
- To-do lists partagées
- Assignment des tâches
- Suivi des responsabilités

5.4 Fonctionnalités Avancées

Des fonctionnalités supplémentaires enrichissent l'expérience utilisateur.

5.4.1 Gestion Financière

Suivi Budgétaire

- Suivi des dépenses en temps réel
- Partage des coûts entre participants

5.5 Administration et Sécurité

Les fonctionnalités d'administration assurent le contrôle et la sécurité de la plateforme.

5.5.1 Console d'Administration

Gestion des Utilisateurs

- Gestion des rôles et permissions
- Modération des contenus
- Surveillance des activités
- Actions en masse

5.5.2 Sécurité et Conformité

Protection des Données

- Chiffrement bout en bout
- Contrôle des accès
- Conformité RGPD



6. Exigences Techniques

Cette section détaille les spécifications techniques nécessaires pour garantir le bon fonctionnement et la sécurité de JUNBI.

6.1 Architecture Technique

L'architecture de JUNBI est conçue pour être simple, maintenable et performante, tout en suivant les bonnes pratiques de développement modernes.

6.1.1 Architecture Générale

Architecture MVC (Model-View-Controller)

- **Models** : Représentation des données et logique métier
 - Validation des données
 - Interactions avec la base de données
- **Views** : Interface utilisateur
 - Pas de logique métier
- **Controllers** : Gestion des requêtes
 - Routes Express.js
 - Middleware d'authentification
 - Validation des entrées
 - Gestion des erreurs

Structure des Dossiers

```
src/  
├─ models/      # Modèles de données  
├─ views/       # Composants React et templates  
├─ controllers/ # Logique de contrôle  
├─ middleware/  # Middleware Express  
├─ config/      # Configuration  
├─ utils/       # Utilitaires  
└─ tests/       # Tests unitaires et d'intégration
```

Stack Technologique Frontend

Next.js (React)

- Utilisation de React 18+ pour ses performances optimales
- Hooks personnalisés pour la gestion d'état
- Architecture basée sur les composants réutilisables
- Rendu côté serveur (SSR) pour optimiser le SEO

Styling

- Sass pour une meilleure organisation du CSS
- Tailwind CSS pour le développement rapide et cohérent
- Convention BEM pour le nommage des classes CSS
- Variables CSS pour la gestion des thèmes

Tests

- Jest pour les tests unitaires
- Tests manuels pour les cas d'utilisation

Stack Technologique Backend

Node.js avec Express.js

- Architecture REST suivant la structure MVC
- Middleware de validation
- Gestion des erreurs centralisée

Base de Données

- PostgreSQL via Supabase
- Migrations manuelles

6.1.2 Infrastructure et Déploiement

Configuration Serveur

Hébergement

- Déploiement principal sur Vercel
- Configuration multi-environnements :
 - Production
 - Développement
- CDN pour les assets statiques

CI/CD

- GitHub Actions pour l'automatisation
- Tests automatiques avant chaque déploiement
- Déploiements automatiques sur les branches principales
- Monitoring des déploiements

Conventions de Développement

Nommage

- camelCase pour les variables et fonctions JavaScript
- PascalCase pour les composants React
- snake_case pour les champs de base de données
- kebab-case pour les fichiers et URLs

Gestion du Code

- ESLint pour le linting JavaScript/TypeScript
- Prettier pour le formatage du code

6.2 Performance et Optimisation

6.2.1 Objectifs de Performance

Métriques Clés

- Core Web Vitals :
 - LCP (Largest Contentful Paint) < 2.5s
 - FID (First Input Delay) < 100ms
 - CLS (Cumulative Layout Shift) < 0.1
- Temps de chargement initial < 3s
- Score Lighthouse minimum de 90

Capacité et Scalabilité

- Support de 100 utilisateurs simultanés
- Temps de réponse API < 300ms
- Taux de disponibilité de 99.9%

6.2.2 Stratégies d'Optimisation

Frontend

- Lazy loading des images et composants
- Prefetching des routes principales
- Mise en cache des assets statiques

Backend

- Pagination optimisée
- Rate limiting

6.3 Sécurité

6.3.1 Authentication et Autorisation

Systeme d'Authentication

- JWT avec refresh tokens
- Sessions sécurisées côté serveur
- Protection contre la force brute
- Politique de mot de passe fort

Gestion des Permissions

- RBAC (Role-Based Access Control)
- Permissions granulaires
- Validation côté serveur systématique

6.3.2 Protection des Données

Sécurité Applicative

- Protection contre les attaques XSS
- Protection CSRF
- Validation stricte des entrées

Chiffrement

- TLS 1.3
- Chiffrement des données sensibles
- Hachage des mots de passe avec Argon2

6.4 Compatibilité et Standards

6.4.1 Support Navigateur

Navigateurs Desktop

- Chrome (dernières 2 versions)
- Firefox (dernières 2 versions)
- Safari (dernières 2 versions)
- Edge (dernières 2 versions)

Support Mobile

- iOS Safari (2 dernières versions majeures)
- Chrome pour Android (dernières 2 versions)
- Responsive design

6.4.2 Accessibilité

Standards

- Conformité WCAG 2.1 niveau AA
- Support des lecteurs d'écran
- Navigation au clavier
- Tests d'accessibilité automatisés

Responsive Design

Breakpoints principaux (min-width):

- Mobile : 480px
- Tablette : 768px
- Desktop : 1200px



7. Contraintes et Limites du Projet

Cette section identifie les principales contraintes et limitations à prendre en compte dans le développement de JUNBI.

7.1 Contraintes Temporelles

7.1.1 Planning Global

Échéances Principales

- Phase de conception (4-22 mars 2024)
 - Analyse des besoins : 1 semaine
 - Design UX/UI : 1 semaine
 - Architecture technique : 1 semaine
- Phase de développement (25 mars - 31 octobre 2024)
 - Développement des features : 5 mois
 - Tests et corrections : 2 mois
- Phase de déploiement (1-19 novembre 2024)
 - Migration des données : 1 semaine
 - Mise en production : 1 semaine
 - Formation interne : 1 semaine

7.1.2 Jalons Critiques

Points de Contrôle

- Validation de la conception : 22 mars 2024
- Version bêta interne : 31 août 2024
- Release candidate : 31 octobre 2024
- Mise en production : 19 novembre 2024

7.2 Contraintes Techniques

7.2.1 Limitations Système

Contraintes de Performance

- Taille maximale des fichiers : 20MB
- Nombre max d'utilisateurs simultanés : 500
- Temps de réponse maximal : 800ms
- Volume de stockage initial : 100GB

Contraintes Réseau

- Compatibilité multi-navigateurs
- Support mobile complet
- Système de cache efficace

7.3 Contraintes Légales

7.3.1 Conformité

Réglementations

- RGPD/GDPR
- Lois françaises sur la protection des données
- Mentions légales professionnelles
- CGU et CGV conformes

7.3.2 Protection des Données

Exigences de Sécurité

- Chiffrement complet des données sensibles
- Politique de conservation des données
- Procédures de sauvegarde
- Plan de reprise d'activité

7.4 Contraintes Organisationnelles

7.4.1 Équipe Projet

Structure

- 1 Chef de projet
- 2 Développeurs full-stack
- 1 Designer UI/UX

Communication

- Daily meetings (15min)
- Sprint planning bi-mensuel
- Revue de code systématique

7.5 Limites Fonctionnelles

7.5.1 Fonctionnalités

Restrictions V1

- Fonctionnalités core uniquement
- Intégrations externes limitées
- Templates pré-définis
- Personnalisation standard

Évolutions Planifiées

- Roadmap 2025 définie
- Backlog priorisé
- Montée en charge progressive
- Plan d'amélioration continue



8. Référencement SEO

Cette section détaille la stratégie SEO de base pour JUNBI, en s'appuyant sur les fonctionnalités de Next.js.

8.1 Objectifs SEO

8.1.1 Objectifs de Référencement

Objectifs Principaux

- Visibilité sur les moteurs de recherche
- Acquisition de trafic organique
- Amélioration de l'expérience utilisateur

Mots-Clés Ciblés

- "gestion événements"
- "plateforme événementielle"
- "organiser événement"

8.2 Optimisations Techniques

8.2.1 Avantages Next.js

Fonctionnalités SEO

- Rendu côté serveur (SSR)
- Génération statique des pages
- Optimisation automatique des images

8.2.2 Structure de Base

Éléments Techniques

- Meta tags essentiels
- Sitemap automatique
- URLs propres

8.3 Contenu

8.3.1 Pages Principales

- Page d'accueil optimisée
- Page de recherche d'événements
- Page d'un événement

8.3.2 Optimisation On-Page

- Description des images
- Textes de qualité
- Liens internes cohérents

8.4 Suivi

8.4.1 Outils

- Lighthouse pour les performances

8.4.2 Améliorations

- Suivi mensuel des positions
- Mise à jour du contenu
- Optimisation des performances



9. Notre Équipe

L'équipe de JUNBI est composée de professionnels passionnés, chacun apportant ses compétences au projet.

9.1 Composition de l'Équipe

Chef de Projet & Lead Developer

- Alexandre Durand
 - 3 ans d'expérience en développement web
 - Expert Next.js et Node.js
 - Responsable de l'architecture technique
 - Coordination de l'équipe

Développeurs Full-Stack

- Marie Chen
 - Spécialiste React/Next.js
 - Experte PostgreSQL
 - Développement des features principales
- Lucas Weber
 - Expert backend Node.js
 - Responsable sécurité et API
 - Intégration des services tiers

Design UI/UX

- Sofia Rodriguez
 - Expérience en design d'interfaces
 - Responsable de l'expérience utilisateur
 - Création des maquettes et prototypes

9.2 Organisation du Travail

- Méthodologie Agile simplifiée
- Sprints de 2 semaines
- Daily meetings rapides (15min)
- Revue de sprint mensuelle

9.3 Communication

- Discord pour la communication en temps réel
- Linear pour le suivi des tâches
- GitHub pour la gestion du code
- Réunions hebdomadaires le lundi matin sur Teams



10. Compétences Requises

10.1 Compétences Techniques

Développement Frontend

- Maîtrise de React et Next.js
- Développement responsive avec Tailwind et Saas CSS
- Intégration UI/UX
- Tests unitaires Jest

Développement Backend

- Node.js et Express
- PostgreSQL (Supabase)
- API REST

DevOps

- GitHub Actions
- Déploiement Vercel/Heroku
- Monitoring de base

10.2 Compétences Fonctionnelles

Gestion de Projet

- Méthodologie Agile simplifiée
- Planning et suivi
- Documentation technique

Design

- Design d'interfaces
- Prototypage Figma
- Tests utilisateurs basiques



11. Planning

11.1 Phases de Développement

Phase de Conception (4-22 mars 2024)

- Analyse des besoins
- Maquettes Figma
- Architecture technique
- Validation du scope

Phase de Développement (25 mars - 31 octobre 2024)

- Setup de l'infrastructure
- Développement des features core
- Intégration continue
- Tests et corrections

Phase de Déploiement (1-19 novembre 2024)

- Migration des données
- Tests finaux
- Mise en production
- Documentation

11.2 Jalons Critiques

- Validation conception : 22 mars 2024
- MVP fonctionnel : 30 juin 2024
- Beta interne : 31 août 2024
- Release candidate : 31 octobre 2024
- Production : 19 novembre 2024



12. Budget

12.1 Répartition

Développement (70%)

- Salaires équipe
- Environnement de développement

Infrastructure (20%)

- Hébergement (Vercel/Heroku)
- Base de données PostgreSQL (Supabase)
- Services cloud
- Domaine et SSL

Marketing et Documentation (10%)

- Communication digitale
- Documentation technique
- Support utilisateurs
- Matériel de formation



13. Accessibilité

13.1 Normes et Standards

Conformité WCAG 2.1

L'application respectera les directives d'accessibilité les plus récentes :

- Niveau AA des normes WCAG
- Compatibilité avec les lecteurs d'écran
- Contraste et lisibilité optimisés



14. Adaptabilité de l'Application

JUNBI est conçu pour s'adapter à tous les environnements d'utilisation, garantissant une expérience optimale sur l'ensemble des appareils.

14.1 Approche Responsive

Conception Mobile-First

Notre approche de développement priorise l'expérience mobile :

- Design adaptatif fluide
- Optimisation des performances sur mobile
- Interface tactile intuitive

Compatibilité Multi-Supports

L'application assurera une expérience cohérente sur :

- Smartphones et tablettes
- Ordinateurs de bureau
- Différentes tailles d'écran
- Différents systèmes d'exploitation



15. Eco-Conception

L'impact environnemental de JUNBI a été pris en compte dès sa conception, avec une approche éco-responsable du développement numérique.

15.1 Optimisations Techniques

Réduction de l'Impact

Les mesures suivantes seront mises en place :

- Compression optimisée des ressources
- Minimisation des requêtes serveur
- Gestion efficiente du cache
- Optimisation des images et médias

Bonnes Pratiques

Notre approche inclut :

- Code optimisé pour la performance
- Gestion raisonnée des données
- Archivage intelligent



16. RGPD et Protection des Données

La conformité aux réglementations sur la protection des données est une priorité absolue pour JUNBI.

16.1 Mesures de Conformité

Protection des Données

Les mesures suivantes seront implémentées :

- Chiffrement des données sensibles
- Gestion sécurisée des consentements
- Procédures de droit à l'oubli
- Portabilité des données

Documentation et Procédures

Un cadre strict sera mis en place :

- Registre des traitements
- Politique de confidentialité claire
- Procédures de gestion des données
- Formation des équipes



17. Langages et Technologies Utilisés

L'architecture technique de JUNBI repose sur des technologies modernes et éprouvées, sélectionnées pour leur fiabilité et leur performance.

17.1 Technologies Frontend

Next.js (React) avec TypeScript

Le choix de Next.js et TypeScript pour le frontend permet :

- Un développement robuste et typé
- Une maintenance facilitée
- Une meilleure prévention des erreurs
- Une excellente performance utilisateur

Interface Utilisateur

L'expérience utilisateur est optimisée grâce à :

- Tailwind CSS et Saas CSS pour un design responsive et moderne
- Composants réutilisables
- Animations optimisées

17.2 Technologies Backend

Node.js et Express

L'architecture backend s'appuie sur :

- Node.js pour sa performance et sa scalabilité
- Express pour sa flexibilité et sa robustesse
- API RESTful pour la communication client-serveur
- WebSocket pour les fonctionnalités temps réel

Base de Données

Le stockage des données est assuré par :

- PostgreSQL pour les données structurées (Supabase)
- Optimisation des requêtes



18. Veilles Techniques

18.1 Domaines de Veille

Technologies Web

Une attention particulière est portée sur :

- Les évolutions des frameworks JavaScript
- Les nouvelles fonctionnalités des navigateurs
- Les améliorations de performance
- Les standards du web

Sécurité et Performance

Le suivi constant des aspects suivants :

- Nouvelles vulnérabilités
- Bonnes pratiques de sécurité
- Optimisations de performance
- Outils de monitoring



19. Outils Collaboratifs

La collaboration efficace de l'équipe est soutenue par une suite d'outils modernes et complémentaires.

19.1 Communication et Organisation

Outils de Communication

La communication est facilitée par :

- Discord pour les échanges quotidiens
- Teams pour les réunions d'équipe
- Email pour les communications formelles

Gestion de Projet

Le suivi du projet s'appuie sur :

- Linear pour le suivi des tâches
- GitHub pour la gestion du code
- Figma pour le design
- Notion pour la documentation



20. Sécurité

La sécurité est un aspect fondamental de JUNBI, intégrée à tous les niveaux de l'application.

20.1 Mesures de Sécurité

Protection des Données

Les données sont protégées par :

- Chiffrement de bout en bout
- Authentification multi-facteurs
- Sessions sécurisées
- Validation des entrées utilisateur

Surveillance et Maintenance

La sécurité est maintenue grâce à :

- Monitoring continu
- Mises à jour régulières
- Tests de pénétration
- Audits de sécurité



21. MCD (Modèle Conceptuel des Données)

L'architecture des données de JUNBI est conçue pour assurer performance et évolutivité.

21.1 Structure des Données

Entités Principales

Le modèle s'articule autour de :

- Gestion des utilisateurs et profils
- Organisation des événements
- Système de messagerie
- Gestion des ressources

Relations et Interactions

Les relations sont optimisées pour :

- Performance des requêtes
- Intégrité des données
- Évolutivité du système
- Maintenance simplifiée



22. Navigation de l'Application

L'expérience de navigation dans JUNBI est conçue pour être intuitive et efficace.

22.1 Structure de Navigation

Architecture de l'Information

L'organisation du contenu suit :

- Une hiérarchie claire
- Des parcours utilisateur optimisés
- Des accès rapides aux fonctions principales



23. Améliorations Futures

JUNBI est conçu pour évoluer et s'améliorer continuellement.

23.1 Perspectives d'Évolution

Développements Futurs

Les axes d'amélioration incluent :

- Nouvelles fonctionnalités basées sur les retours utilisateurs
- Optimisations de performance
- Intégrations supplémentaires
- Extension des capacités de personnalisation



24. Les Livrables

Les livrables du projet sont clairement définis pour assurer une réalisation conforme aux attentes.

24.1 Documentation et Déploiement

Livrables Techniques

Le projet inclura :

- Code source documenté
- Documentation technique
- Guide d'utilisation
- Procédures de déploiement

Livrables Fonctionnels

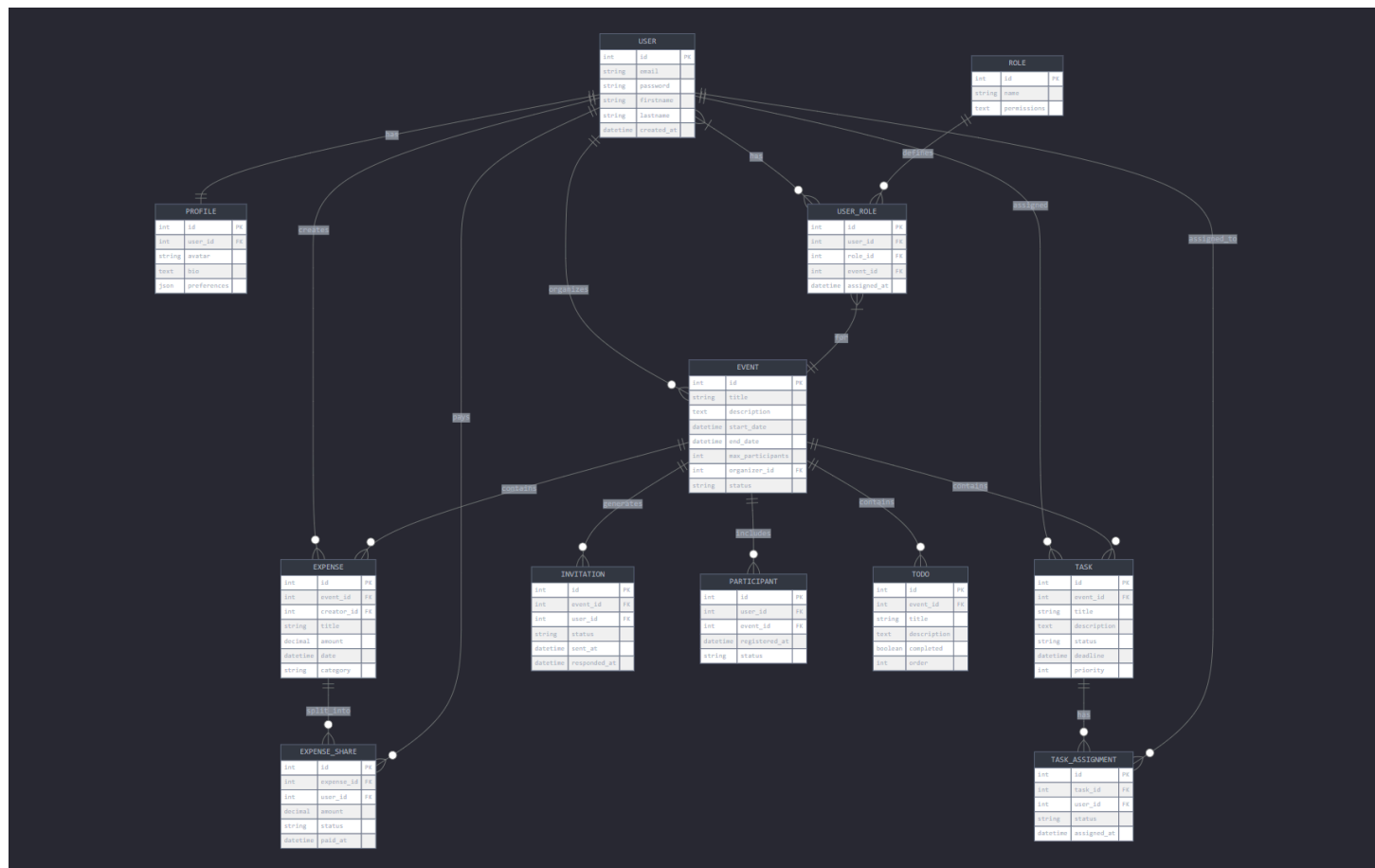
L'application sera livrée avec :

- Application web complète
- Interfaces administrateur
- Outils de monitoring

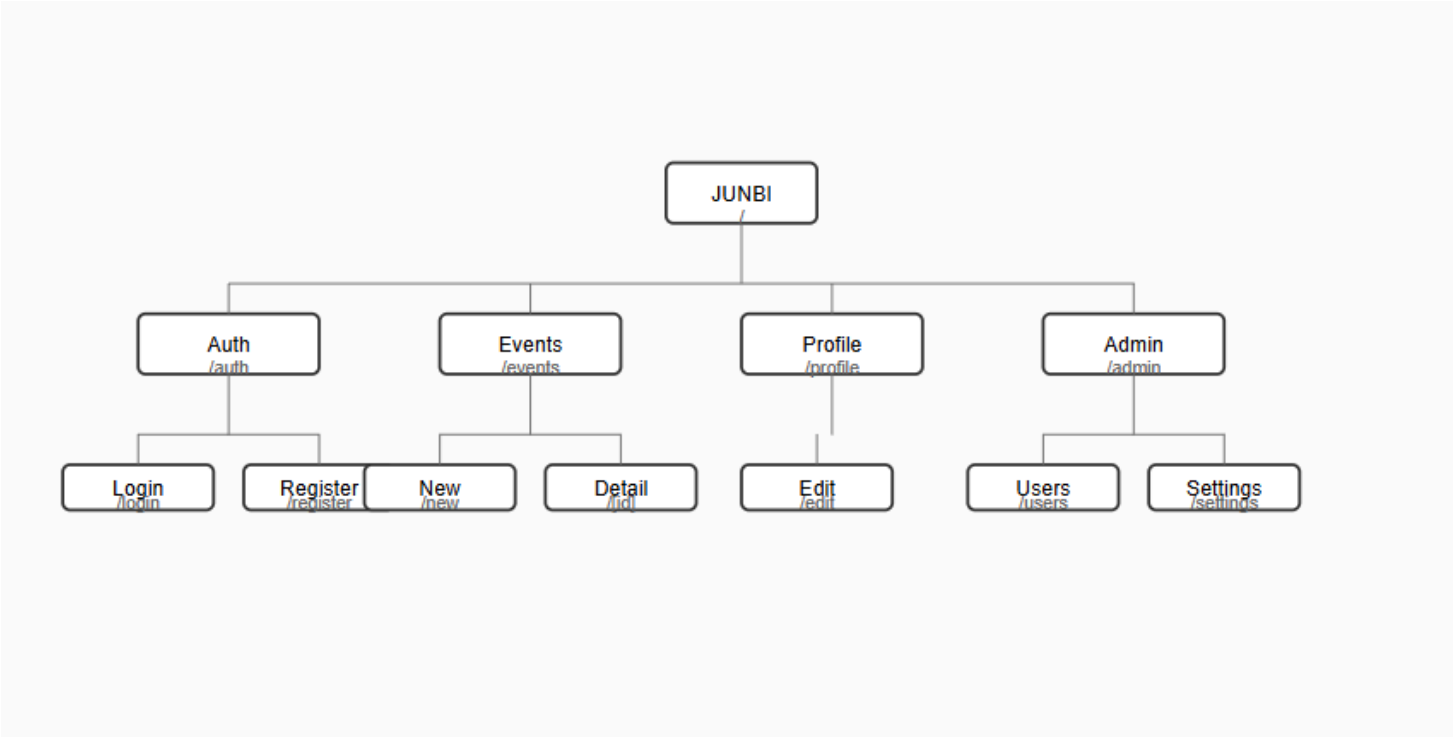
Les annexes fournissent des informations complémentaires au projet.

25.1 Documents Complémentaires

Modèle Conceptuel des Données



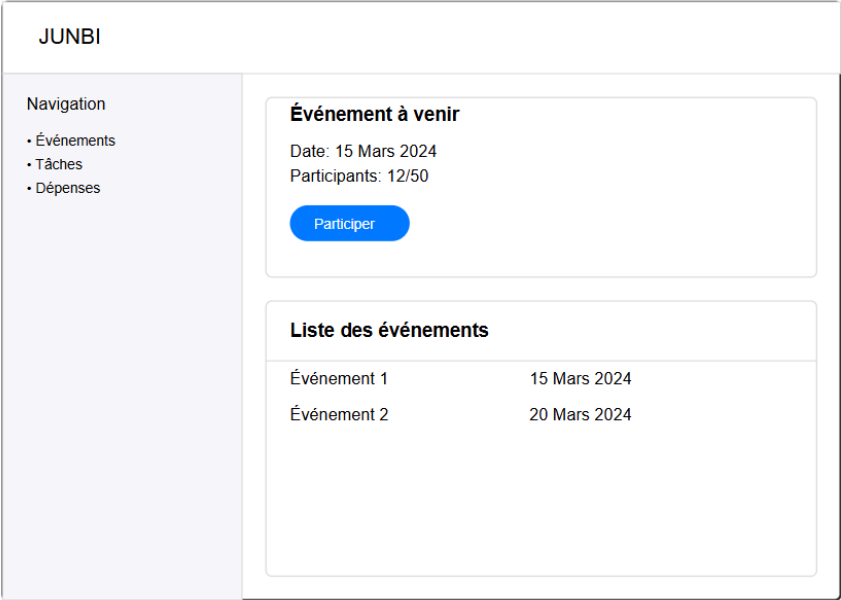
Plan du Site



Wireframes



Vue Mobile



Vue Desktop