CAHIER DES CHARGES TECHNIQUE

Application SaaS pour Avocats

Plateforme collaborative de gestion contractuelle

**Version :** 1.0

**Date :** Octobre 2025

**Statut :** Version synthétique

**SDEN - Sami Dzogang Engineering**

**TABLE DES MATIÈRES**

1. Architecture Technique 4
   1. Stack Technique Global 4
      1. [Frontend 4](#_TOC_250023)
      2. [Backend 4](#_TOC_250022)
      3. [Services Externes 4](#_TOC_250021)
   2. Architecture Base de Données 5
2. Fonctionnalités Principales 6
   1. Comparaison de Documents 6
   2. Éditeur Collaboratif Temps Réel 6
   3. Conversion PDF vers Format Éditable 6
   4. Suggestions Juridiques par IA 7
   5. Gestion des Rôles et Permissions (RBAC) 7
3. Sécurité et Conformité 8
   1. Chiffrement des Données 8
   2. Authentification et Autorisation 8
   3. Conformité RGPD 8
   4. Sauvegarde et Résilience 8
   5. Protection et Rate Limiting 9
4. API REST 9
   1. Structure Générale 9
   2. Endpoints Principaux 9
      1. [Authentification 9](#_TOC_250020)
      2. [Utilisateurs 10](#_TOC_250019)
      3. [Workspaces et Projets 10](#_TOC_250018)
      4. [Documents et Comparaison 10](#_TOC_250017)
      5. [Commentaires et Chat 11](#_TOC_250016)
      6. [Intelligence Artificielle 11](#_TOC_250015)
      7. [Notifications et Facturation 11](#_TOC_250014)
5. Roadmap de Développement - MVP 4 Semaines 12
   1. Approche Agile Accélérée 12
   2. Semaine 1 : Fondations & Authentification 12
      1. [Objectifs 12](#_TOC_250013)
      2. [Livrables 12](#_TOC_250012)
      3. [User Stories Validées 12](#_TOC_250011)
   3. Semaine 2 : Projets, Documents & Upload 13
      1. [Objectifs 13](#_TOC_250010)
      2. [Livrables 13](#_TOC_250009)
      3. [User Stories Validées 13](#_TOC_250008)
   4. Semaine 3 : Commentaires, Chat & Invitations 14
      1. [Objectifs 14](#_TOC_250007)
      2. [Livrables 14](#_TOC_250006)
      3. [User Stories Validées 14](#_TOC_250005)
   5. Semaine 4 : Comparaison, Paiement & Déploiement 15
      1. [Objectifs 15](#_TOC_250004)
      2. [Livrables 15](#_TOC_250003)
      3. [User Stories Validées 15](#_TOC_250002)
   6. Fonctionnalités Reportées (Post-MVP) 15
   7. Livrable Final MVP (Fin Semaine 4) 16
      1. [Application Fonctionnelle avec : 16](#_TOC_250001)
   8. Conclusion Roadmap MVP 16
6. Risques et Mitigation 17
   * 1. [Performance Technique 17](#_TOC_250000)

**ARCHITECTURE TECHNIQUE**

## Stack Technique Global

Le projet s’appuie sur une architecture moderne et éprouvée garantissant scalabilité, main- tenabilité et sécurité.

## Frontend

* + - * **Framework :** Next.js 14+ (App Router) avec React 18+
      * **Interface utilisateur :** Tailwind CSS + Shadcn/ui (composants)
      * **Animations :** Framer Motion
      * **Éditeur de texte :** Tiptap avec édition collaborative (Yjs)
      * **Gestion d’état :** Zustand (léger et performant)
      * **Validation :** Zod + React Hook Form

## Backend

* + - * **Runtime :** Node.js 20 LTS
      * **Framework API :** Express.js ou Fastify
      * **Base de données :** PostgreSQL 15+
      * **Cache :** Redis 7+ (sessions et performances)
      * **ORM :** Prisma (gestion schema et migrations)
      * **WebSocket :** Socket.io (chat et collaboration temps réel)
      * **File d’attente :** BullMQ (traitements asynchrones)

## Services Externes

* + - * **Authentification :** NextAuth.js v5 (Auth.js)
      * **OCR / Parsing PDF :** Google Document AI
      * **Intelligence Artificielle :** OpenAI GPT-4o (suggestions juridiques)
      * **Paiement :** Stripe (via Lemon Squeezy)
      * **Emails :** Resend
      * **Stockage documents :** AWS S3 ou compatible
      * **Monitoring :** Sentry (erreurs) + PostHog (analytics produit)

## Architecture Base de Données

Le schéma relationnel s’articule autour des entités suivantes :

users Utilisateurs de la plateforme (id, email, password\_hash, full\_name, role)

workspaces Espaces de travail (id, name, owner\_id, subscription\_tier)

projects Projets contractuels (id, workspace\_id, name, client\_name, contract\_type, status)

documents Versions de documents (id, project\_id, version\_number, file\_path, par- sed\_content)

comments Annotations collaboratives (id, document\_id, user\_id, content, posi- tion\_data)

chat\_messages notifications

Messagerie projet (id, project\_id, user\_id, content)

Système de notifications (id, user\_id, type, related\_entity\_id, is\_read)

Tables complémentaires : templates, document\_changes, project\_members, workspace\_members,

support\_tickets.

# FONCTIONNALITÉS PRINCIPALES

## Comparaison de Documents

La plateforme permet la comparaison intelligente de multiples versions contractuelles :

* + - * Upload jusqu’à 6 versions simultanées et possibilité d’ajouté X versions supplémen- taires.
      * Algorithme de différenciation (bibliothèque jsdiff)
      * Affichage côte à côte ou fusionné
      * Coloration automatique des modifications (ajouts, suppressions, modifications)

## Éditeur Collaboratif Temps Réel

L’édition collaborative repose sur des technologies éprouvées :

* + - * Édition simultanée multi-utilisateurs sans conflit
      * CRDT (Conflict-free Replicated Data Type) via Yjs
      * Curseurs visibles en temps réel avec identification utilisateur
      * Historique complet avec restauration de versions antérieures
      * Synchronisation < 200ms entre clients

## Conversion PDF vers Format Éditable

Pipeline de traitement automatisé :

1. Upload du document PDF
2. Extraction par OCR (Google Document AI)
3. Parsing de la structure documentaire (sections, clauses, articles)
4. Conversion en JSON standardisé
5. Import dans l’éditeur Tiptap
6. Interface de correction manuelle des erreurs OCR

Objectif de précision : > 95% sur documents de qualité standard.

## Suggestions Juridiques par IA

Assistance intelligente à la rédaction :

* Sélection de texte + prompt utilisateur contextuel
* Exemples de requêtes : "rendre plus formel", "simplifier le langage", "traduire FREN"
* Génération de plusieurs suggestions par requête
* Workflow Accept / Reject / Edit manuel
* Rate limiting : 20 requêtes/minute par utilisateur

## Gestion des Rôles et Permissions (RBAC)

Système de permissions granulaires à deux niveaux :

## Niveau Workspace :

* **Owner :** Droits complets + gestion facturation
* **Admin :** Gestion membres et projets
* **Member :** Accès selon permissions projet

## Niveau Project :

* **Creator :** Administrateur du projet
* **Editor :** Modification des documents
* **Commenter :** Annotations uniquement (pas d’édition)
* **Reader :** Consultation seule

# SÉCURITÉ ET CONFORMITÉ

## Chiffrement des Données En transit :

* TLS 1.3 obligatoire sur toutes connexions
* HSTS (HTTP Strict Transport Security) activé

## Au repos :

* Chiffrement AES-256 des documents (S3)
* Chiffrement transparent base de données (PostgreSQL TDE)
* Secrets stockés dans vault sécurisé

## Messagerie :

* Chiffrement de bout en bout (E2EE)
* Implémentation Signal Protocol ou équivalent
* Clés générées côté client

## Authentification et Autorisation

* Tokens JWT avec mécanisme de refresh
* Hashing Argon2 ou bcrypt pour mots de passe
* MFA disponible (TOTP - applications d’authentification)
* Middleware de vérification des permissions sur chaque endpoint

## Conformité RGPD

* Export de données utilisateur (endpoint /api/users/:id/export)
* Droit à l’oubli (suppression complète des données)
* Consentement explicite pour analytics
* Conservation des logs d’audit : 1 an minimum

## Sauvegarde et Résilience

* Backup automatique quotidien de la base de données
* Backup incrémental toutes les 6 heures
* Rétention : 30 jours glissants
* Test de restauration mensuel obligatoire

## Protection et Rate Limiting

Configuration par endpoint pour prévenir les abus :

* /api/auth/login : 5 tentatives / 15 minutes
* /api/projects : 60 requêtes / minute
* /api/ai/suggest : 20 requêtes / minute (coûts API)
* Endpoints standards : 100 requêtes / minute

# API REST

## Structure Générale

* **Base URL :** https://companion.com/v1
* **Format :** JSON exclusivement
* **Authentification :** Bearer token (JWT dans header)
* **Pagination :** Paramètres ?page=1&limit=50
* **Filtrage :** Paramètres query string (?status=active&sort=-created\_at)

**Endpoints Principaux**

## Authentification

**Route Description**

POST

/auth/register POST

/auth/login POST

/auth/logout POST

/auth/refresh POST

/auth/forgot- password

Inscription nouveau compte Connexion utilisateur Déconnexion Renouvellement token Réinitialisation mot de passe

## Utilisateurs

**Route Description**

GET /users/me Profil utilisateur connecté

PATCH

/users/me DELETE

/users/me GET

/users/me/export

Modification profil Suppression compte (RGPD) Export données personnelles

## Workspaces et Projets

**Route Description**

GET /workspaces Liste workspaces POST /workspaces Création workspace GET /projects Liste projets (filtrable)

POST /projects Création projet GET /projects/ :id Détails projet

POST /pro- Invitation collaborateur jects/ :id/members

## Documents et Comparaison

**Route Description**

GET /pro-

jects/ :id/documents

POST /pro-

jects/ :id/documents

GET /docu-

ments/ :id/content

PATCH /docu- ments/ :id/content

POST /docu-

ments/compare

POST /docu-

ments/ :id/convert

Versions du document Upload nouvelle version Contenu structuré Modification contenu Comparaison versions Conversion PDF éditable

## Commentaires et Chat

**Route Description**

GET /docu-

ments/ :id/comments POST /docu-

ments/ :id/comments POST /com-

ments/ :id/resolve

Liste commentaires Ajout commentaire Résoudre commentaire

GET /projects/ :id/chat Historique messages POST /projects/ :id/chat Envoi message

## Intelligence Artificielle

**Route Description**

POST /ai/suggest Suggestions reformulation POST /ai/analyze Analyse juridique (future)

## Notifications et Facturation

**Route Description**

GET /notifications Liste notifications

PATCH /notifica- Marquer comme lu tions/ :id/read

GET /billing/subscription Infos abonnement POST /billing/checkout Session paiement Stripe

# ROADMAP DE DÉVELOPPEMENT - MVP 4 SEMAINES

## Approche Agile Accélérée

Cette roadmap concentre le développement sur un **MVP fonctionnel en 4 semaines** avec les fonctionnalités essentielles. L’objectif est de valider rapidement le produit auprès des utilisateurs avant d’investir dans des fonctionnalités avancées.

## Semaine 1 : Fondations & Authentification

## Objectifs

Mettre en place l’infrastructure technique et le système d’authentification complet.

## Livrables

**Infrastructure (Jours 1-2)**

* + - * Repository Git + CI/CD basique (GitHub Actions)
      * Next.js 14+ configuré avec App Router
      * PostgreSQL + Prisma (schema initial + migrations)
      * NextAuth.js configuré
      * Design system Shadcn/ui (composants de base)
      * Variables d’environnement (dev/staging)

## Authentification (Jours 3-5)

* + - * Pages : /login, /register, /forgot-password
      * API routes auth complètes (inscription, connexion, reset MDP)
      * Envoi emails (Resend) - templates basiques
      * Dashboard layout (navbar + sidebar responsive)
      * Protection routes authentifiées

## User Stories Validées

* + - * Je peux créer un compte avec email/mot de passe
      * Je peux me connecter et réinitialiser mon mot de passe
      * J’accède à un dashboard après connexion

## Semaine 2 : Projets, Documents & Upload

## Objectifs

Permettre la création de projets et l’upload de documents PDF avec visualisation.

## Livrables

**Projets (Jours 1-2)**

* + - * Modèle BDD : projects, project\_members
      * API CRUD projets (POST, GET, PATCH, DELETE)
      * Dashboard : liste projets avec cards (dernière activité en haut)
      * Modal création projet (nom, client, type contrat)
      * Page détail projet (layout avec tabs : Document, Chat)

## Documents (Jours 3-5)

* + - * Modèle BDD : documents (versioning simple)
      * Stockage S3 (ou équivalent) configuré
      * Zone upload drag & drop
      * API upload document (POST /projects/:id/documents)
      * Viewer PDF intégré (react-pdf)
      * Extraction texte basique (pdf-parse library - pas d’OCR avancé pour MVP)
      * Affichage historique versions (liste simple)

## User Stories Validées

* + - * Je peux créer un projet (nom + client + type)
      * Je peux uploader un document PDF dans un projet
      * Je vois mes projets sur le dashboard
      * Je peux consulter le PDF uploadé

## Semaine 3 : Commentaires, Chat & Invitations

## Objectifs

Activer la collaboration avec commentaires, chat projet et système d’invitations.

## Livrables

**Commentaires (Jours 1-2)**

* + - * Modèle BDD : comments (avec position dans document)
      * API commentaires (GET, POST, DELETE)
      * Interface commentaires (sidebar latérale)
      * Ajout commentaire sur section de texte
      * Thread de réponses (1 niveau maximum pour MVP)

## Chat Projet (Jour 3)

* + - * Modèle BDD : chat\_messages
      * WebSocket setup basique (Socket.io)
      * Interface chat (onglet dans projet)
      * Envoi/réception messages temps réel

## Invitations & Permissions (Jours 4-5)

* + - * Système RBAC simplifié (2 rôles : Editor, Viewer)
      * API invitation collaborateur par email
      * Modal invitation dans projet
      * Notifications in-app basiques (dropdown navbar)
      * Email notification sur invitation

## User Stories Validées

* + - * Je peux ajouter un commentaire sur le document
      * Je peux discuter en temps réel dans le chat projet
      * Je peux inviter un collaborateur (Editor ou Viewer)
      * Je reçois une notification quand je suis invité

## Semaine 4 : Comparaison, Paiement & Déploiement

## Objectifs

Finaliser avec comparaison de versions, système de paiement et déploiement production.

## Livrables

**Comparaison de Versions (Jours 1-2)**

* + - * Upload multi-versions (historique dans BDD)
      * API comparaison : POST /documents/compare
      * Algorithme diff basique (bibliothèque jsdiff)
      * Interface split view (2 colonnes côte à côte)
      * Coloration différences (ajouts en vert, suppressions en rouge)

## Paiement (Jour 3)

* + - * Intégration Stripe Checkout (mode test)
      * Page /pricing simple
      * Modal checkout (redirection Stripe)
      * Webhook Stripe (activation abonnement)
      * Limitation freemium : 1 projet max, levée après paiement

## Déploiement & Tests (Jours 4-5)

* + - * Déploiement staging (Vercel ou équivalent)
      * Configuration environnement production
      * Tests E2E critiques (signup → création projet → upload → invitation)
      * Monitoring basique (Sentry pour erreurs)
      * Documentation technique (README complet)
      * Corrections bugs bloquants

## User Stories Validées

* + - * Je peux comparer 2 versions d’un document
      * Les différences sont clairement identifiées
      * Je peux souscrire à l’offre Standard (23€/mois)
      * Mon abonnement est activé automatiquement après paiement

## Fonctionnalités Reportées (Post-MVP)

Les éléments suivants sont **exclus du MVP 4 semaines** et seront développés en Phase 2 selon feedback utilisateurs :

* + - * Édition collaborative temps réel (Yjs/Tiptap Collaboration)
      * Suggestions IA (OpenAI GPT-4o)
      * Conversion PDF intelligente (OCR avancé)
      * Templates juridiques prédéfinis
      * Onboarding interactif
      * Mentions utilisateurs (@username)
      * Système notifications avancé
      * Analytics et exports avancés

**Justification :** Ces fonctionnalités nécessitent des intégrations complexes. Le MVP doit d’abord valider le besoin core : **collaboration autour de documents contractuels**.

## Livrable Final MVP (Fin Semaine 4)

## Application Fonctionnelle avec :

**Authentification sécurisée** (inscription, connexion, reset MDP)

**Gestion de projets** (création, consultation, archivage) **Upload & visualisation PDF** (multi-versions) **Comparaison de versions** (diff visuel côte à côte) **Collaboration basique** (commentaires + chat temps réel) **Invitations collaborateurs** (2 rôles : Editor, Viewer) **Paiement Stripe** (freemium → Standard 23€/mois) **Déployé en production** (accessible publiquement)

## Conclusion Roadmap MVP

Cette approche **Lean Startup** privilégie :

**Vitesse** : MVP fonctionnel en 4 semaines

**Validation** : Feedback terrain avant investissement massif **Flexibilité** : Pivots possibles selon retours utilisateurs **Qualité** : Focus sur features core, bien exécutées

# RISQUES ET MITIGATION

|  |  |  |  |
| --- | --- | --- | --- |
| **Risque** | **Impact** | **Prob.** | **Mitigation** |
| Précision OCR insuffi- sante | Élevé | Moyen | Interface correction + multi-API |
| Performance édition collaborative | Élevé | Moyen | Load testing précoce  + Yjs |
| Complexité parsing PDF | Moyen | Élevé | Structure simple MVP puis itération |
| Coûts API IA élevés | Moyen | Moyen | Rate limiting + cache agressif |
| Adoption freemium faible | Élevé | Moyen | A/B testing + inter- views utilisateurs |
| Sécurité données | Critique | Faible | Audit externe + bug bounty |

## Performance Technique

* + - * Temps chargement dashboard : < 2 secondes
      * Disponibilité (uptime) : > 99,5%
      * Zéro incident sécurité majeur

*Document établi par SDEN - Sami Dzogang Engineering Version 1.0 - Octobre 2025*