# Cahier des Charges - Transformation Digitale de l’Emploi du Temps

**Projet** : Refonte de l’outil de gestion des emplois du temps  
**Version** : 1.0  
**Date** : Octobre 2025  
**Responsable** : Équipe Transformation Digitale - Workshop Poudlard EPSI/WIS

## Table des matières

1. [Contexte et enjeux](#Xe7de20a3ac287dc6579f4f60a7589140d9eb26d)
2. [Analyse de l’existant](#X0530698e96057ea608904195f1ef8fdde54f15e)
3. [Étude des besoins](#X263f7cee9b5068547d649a6d96f7b265f42625f)
4. [Solution proposée](#X4913c978ceacfff5740852b776109e06d73ccde)
5. [Spécifications fonctionnelles](#Xa48b79a3a27cb2257baaf756c7ad1beaed46bca)
6. [Spécifications techniques](#X7678c6596d44a25f7758087f33fde31b5d69b1c)
7. [Contraintes et exigences](#X9222eeee67fbfdef9707dc9c92c95b5f80c7574)
8. [Planning et jalons](#X984872ed352de11b958b82334b7fa45be3da194)
9. [Risques et mitigation](#X8fe9c479498db1f515059105d237611b13febb8)
10. [Budget prévisionnel](#X73104d1c7b56de46b70805f7dc5be6c1168a873)

## 1. Contexte et enjeux

### 1.1 Présentation du contexte

L’école utilise actuellement un outil de gestion des emplois du temps développé il y a plus de 10 ans. Cet outil, bien que fonctionnel à l’origine, ne répond plus aux standards actuels en termes d’ergonomie, de performance et de fiabilité.

**Situation actuelle** : - Plus de 2 500 étudiants consultent leur emploi du temps quotidiennement - 150+ enseignants gèrent leurs disponibilités et cours - 15 personnes de l’administration planifient et maintiennent les emplois du temps - Environ 200 modifications par semaine en moyenne

### 1.2 Problématiques identifiées

#### Problèmes critiques

1. **Performance** : Temps de chargement de 5 à 15 secondes
2. **Fiabilité** : 30+ incidents signalés par mois
3. **Ergonomie** : Interface non intuitive, nombreux clics nécessaires
4. **Compatibilité** : Non responsive, inutilisable sur mobile
5. **Notifications** : Système de notification défaillant

#### Impact business

* Perte de temps estimée : **2h/semaine** par administrateur (soit 150h/an)
* Satisfaction utilisateur actuelle : **35%** (enquête interne 2024)
* Coût des incidents : Support technique surchargé
* Image de l’école : Décalage avec les standards modernes

### 1.3 Objectifs de la transformation

#### Objectifs stratégiques

* Moderniser l’expérience utilisateur
* Améliorer l’efficacité opérationnelle
* Renforcer l’image digitale de l’école
* Faciliter l’intégration avec l’écosystème numérique existant

#### Objectifs mesurables

* Temps de chargement : **< 1 seconde**
* Satisfaction utilisateur : **> 85%**
* Réduction des incidents : **-70%**
* Gain de temps administratif : **-30%**
* Adoption mobile : **> 60%** des consultations

## 2. Analyse de l’existant

### 2.1 Fonctionnalités actuelles

#### Pour les étudiants

* ✅ Consultation de l’emploi du temps personnel
* ✅ Vue par jour/semaine/mois
* ✅ Export iCal (fonctionnalité instable)
* ❌ Notifications de changements
* ❌ Vue responsive mobile
* ❌ Filtres avancés

#### Pour les enseignants

* ✅ Consultation des cours assignés
* ✅ Déclaration des disponibilités
* ✅ Demande de modification de salle
* ❌ Gestion des absences en temps réel
* ❌ Accès aux ressources pédagogiques liées
* ❌ Statistiques personnelles

#### Pour l’administration

* ✅ Planification des cours
* ✅ Gestion des salles et ressources
* ✅ Attribution des enseignants
* ✅ Gestion des groupes et promotions
* ❌ Détection automatique des conflits
* ❌ Tableaux de bord et analytics
* ❌ Import/Export CSV/Excel fiable
* ❌ Historique des modifications

### 2.2 Architecture technique actuelle

#### Stack technique

* **Backend** : PHP 5.6 (obsolète)
* **Base de données** : MySQL 5.5
* **Frontend** : jQuery + HTML/CSS legacy
* **Serveur** : Apache 2.2
* **Déploiement** : Manuel via FTP

#### Problèmes techniques

* Technologies obsolètes et non maintenues
* Absence de tests automatisés
* Code non documenté
* Pas de versionning (Git)
* Pas de CI/CD
* Sécurité : vulnérabilités connues
* Pas d’API moderne

### 2.3 Points de douleur utilisateurs

#### Étudiants (enquête N=250)

1. **Lenteur** (92%) : “Ça prend trop de temps à charger”
2. **Mobile** (87%) : “Impossible à utiliser sur téléphone”
3. **Notifications** (78%) : “Je ne suis jamais prévenu des changements”
4. **Ergonomie** (68%) : “L’interface est trop compliquée”
5. **Export** (45%) : “Je n’arrive pas à exporter vers mon calendrier”

#### Enseignants (enquête N=80)

1. **Complexité** (85%) : “Trop d’étapes pour faire une simple modification”
2. **Visibilité** (72%) : “Je ne vois pas facilement mes conflits”
3. **Mobile** (70%) : “Je ne peux pas consulter en déplacement”
4. **Reporting** (58%) : “Pas de statistiques sur mes heures”
5. **Intégration** (50%) : “Pas de lien avec les autres outils”

#### Administration (enquête N=12)

1. **Conflits** (100%) : “Détection manuelle des conflits horaires”
2. **Import/Export** (92%) : “Processus d’import très chronophage”
3. **Performance** (83%) : “Plantages fréquents en période de rentrée”
4. **Historique** (75%) : “Impossible de retrouver qui a fait quoi”
5. **Analytics** (67%) : “Aucune vision d’ensemble”

## 3. Étude des besoins

### 3.1 Méthodologie d’enquête

#### Approche

1. **Questionnaires en ligne** (342 répondants)
   * Étudiants : 250 réponses
   * Enseignants : 80 réponses
   * Administration : 12 réponses
2. **Interviews approfondies** (25 personnes)
   * 15 étudiants (représentant différentes promotions)
   * 7 enseignants (temps plein et intervenants)
   * 3 administratifs (planification, direction, IT)
3. **Observation terrain** (2 semaines)
   * Shadowing des administrateurs
   * Analyse des tickets support
   * Mesures de performance réelles
4. **Benchmark concurrentiel**
   * Analyse de 5 solutions du marché
   * Visite d’écoles partenaires
   * Démonstrations d’éditeurs

### 3.2 Besoins par type d’utilisateur

#### Étudiants - Besoins essentiels (Must Have)

1. **Consultation rapide** : Accès en < 1 seconde
2. **Mobile first** : Application ou site responsive
3. **Notifications push** : Alertes de changements en temps réel
4. **Export calendrier** : Synchronisation iCal/Google Calendar
5. **Vue personnalisée** : Filtrage par type de cours

#### Étudiants - Besoins souhaités (Should Have)

1. **Mode hors ligne** : Consultation sans connexion
2. **Partage** : Partager son emploi du temps
3. **Recherche** : Recherche de cours/salles
4. **Favoris** : Marquer des cours importants
5. **Historique** : Consulter les anciennes versions

#### Enseignants - Besoins essentiels (Must Have)

1. **Gestion des disponibilités** : Interface simple et rapide
2. **Vue consolidée** : Tous mes cours en un coup d’œil
3. **Détection conflits** : Alertes automatiques
4. **Mobile** : Consultation et modification en mobilité
5. **Notifications** : Alertes de modifications

#### Enseignants - Besoins souhaités (Should Have)

1. **Statistiques** : Heures enseignées, répartition
2. **Planning annuel** : Vision long terme
3. **Ressources liées** : Accès aux supports de cours
4. **Absences** : Déclarer une absence facilement
5. **Export** : Export PDF de planning

#### Administration - Besoins essentiels (Must Have)

1. **Détection conflits auto** : Algorithme de validation
2. **Import/Export massif** : CSV, Excel, API
3. **Historique complet** : Traçabilité des modifications
4. **Tableaux de bord** : Métriques clés en temps réel
5. **Gestion des contraintes** : Règles métier paramétrables

#### Administration - Besoins souhaités (Should Have)

1. **Optimisation auto** : Suggestions de planification
2. **Reporting avancé** : Analytics et exports
3. **Multi-campus** : Gestion de plusieurs sites
4. **Workflows** : Validation multi-niveaux
5. **Intégrations** : ERP, LMS, CRM existants

### 3.3 Priorisation MoSCoW

#### Must Have (MVP - Release 1)

* Interface moderne et responsive
* Performance < 1s
* Notifications en temps réel
* Export calendrier
* Détection conflits automatique
* Historique et traçabilité
* API REST

#### Should Have (Release 2)

* Application mobile native
* Mode hors ligne
* Analytics avancés
* Import/Export avancé
* Optimisation automatique

#### Could Have (Release 3+)

* IA pour suggestions de planification
* Intégration chatbot
* Réalité augmentée (navigation campus)
* Blockchain pour certification présence

#### Won’t Have (hors scope)

* Gestion de la paie enseignants
* Système de notation
* Gestion des inscriptions

## 4. Solution proposée

### 4.1 Vision de la nouvelle solution

**Vision** : Un écosystème digital intégré qui facilite la gestion et la consultation des emplois du temps, offrant une expérience utilisateur moderne, rapide et fiable sur tous les supports.

**Principes directeurs** : 1. **Mobile First** : Conception prioritaire pour mobile 2. **Performance** : Temps de chargement minimal 3. **Simplicité** : Interfaces intuitives, peu de clics 4. **Fiabilité** : Tests automatisés, haute disponibilité 5. **Évolutivité** : Architecture modulaire et scalable

### 4.2 Fonctionnalités prioritaires

#### Phase 1 - MVP (3 mois)

1. **Interface utilisateur moderne**
   * Design System basé sur Material Design
   * Responsive (mobile, tablette, desktop)
   * Thème clair/sombre
2. **Consultation optimisée**
   * Temps de chargement < 1s
   * Vues multiples (jour/semaine/mois)
   * Filtres et recherche
3. **Notifications temps réel**
   * Push notifications (web + mobile)
   * Email de synthèse
   * Préférences personnalisables
4. **Export et synchronisation**
   * Export iCal/Google Calendar
   * Liens de synchronisation auto
   * Export PDF
5. **Administration simplifiée**
   * Détection automatique des conflits
   * Drag & drop pour planification
   * Historique des modifications

#### Phase 2 - Enrichissement (3 mois)

1. **Application mobile native**
   * iOS et Android
   * Mode hors ligne
   * Notifications natives
2. **Analytics et reporting**
   * Tableaux de bord
   * Exports personnalisés
   * Métriques de qualité
3. **Intégrations**
   * API REST publique
   * Webhooks
   * Connecteurs ERP/LMS
4. **Optimisation**
   * Suggestions automatiques
   * Détection de patterns
   * Alertes proactives

#### Phase 3 - Innovation (3 mois)

1. **IA et Machine Learning**
   * Prédiction de disponibilités
   * Optimisation automatique
   * Recommandations personnalisées
2. **Collaboration avancée**
   * Messagerie intégrée
   * Partage de documents
   * Visioconférence

### 4.3 Roadmap de déploiement

Mois 1-2 : Conception et Design  
 - Workshops utilisateurs  
 - Maquettes et prototypes  
 - Validation UX/UI  
  
Mois 3-5 : Développement MVP  
 - Setup infrastructure  
 - Backend API  
 - Frontend responsive  
 - Tests unitaires  
  
Mois 6 : Tests et Recette  
 - Tests utilisateurs  
 - Corrections bugs  
 - Performance tuning  
  
Mois 7 : Déploiement pilote  
 - 1 promotion test  
 - Feedback continu  
 - Ajustements  
  
Mois 8 : Déploiement général  
 - Tous les utilisateurs  
 - Formation  
 - Support renforcé  
  
Mois 9-11 : Phase 2  
 - App mobile  
 - Analytics  
 - Intégrations  
  
Mois 12+ : Phase 3  
 - Innovation  
 - IA  
 - Amélioration continue

## 5. Spécifications fonctionnelles

### 5.1 Module Consultation (Étudiants)

#### SF-01 : Authentification

* L’utilisateur doit pouvoir se connecter via SSO (Single Sign-On)
* Support OAuth2 avec Active Directory / LDAP
* Option “Se souvenir de moi” (30 jours)
* Connexion biométrique sur mobile

#### SF-02 : Tableau de bord personnel

* Affichage des cours du jour en page d’accueil
* Cours en cours mis en évidence
* Prochain cours avec compte à rebours
* Météo locale (optionnel)

#### SF-03 : Vues calendrier

* Vue jour : liste chronologique
* Vue semaine : grille 7 jours
* Vue mois : overview mensuel
* Vue agenda : liste filtrée
* Navigation rapide (aujourd’hui, dates)

#### SF-04 : Détail d’un cours

* Nom du cours et code
* Enseignant(s)
* Salle et bâtiment avec plan interactif
* Horaires début/fin
* Type de cours (CM, TD, TP)
* Ressources liées (documents, liens)
* Historique des modifications

#### SF-05 : Filtres et recherche

* Filtre par type de cours
* Filtre par enseignant
* Filtre par salle/bâtiment
* Recherche textuelle
* Sauvegarde des filtres favoris

#### SF-06 : Export et synchronisation

* Export iCal (lien unique)
* Export Google Calendar (1 clic)
* Export PDF personnalisable
* Lien de partage temporaire
* QR Code pour partage rapide

#### SF-07 : Notifications

* Notification de changement de cours
* Notification de nouveau cours
* Notification d’annulation
* Notification de changement de salle
* Rappel avant cours (configurable)
* Résumé hebdomadaire par email

### 5.2 Module Gestion (Enseignants)

#### SF-10 : Mes cours

* Liste de tous mes cours
* Filtres par période
* Statistiques (heures totales, répartition)
* Export PDF/Excel

#### SF-11 : Disponibilités

* Calendrier de saisie des disponibilités
* Import de disponibilités (fichier)
* Récurrence (toutes les semaines)
* Exceptions ponctuelles
* Validation administrative

#### SF-12 : Demandes de modification

* Formulaire de demande simple
* Pièces justificatives
* Suivi du workflow
* Notifications de statut
* Historique des demandes

#### SF-13 : Absences

* Déclaration d’absence
* Proposition de remplacement
* Notification automatique administration
* Impact sur planning

### 5.3 Module Administration

#### SF-20 : Planification

* Interface drag & drop
* Vue multi-ressources (salles, enseignants)
* Détection de conflits en temps réel
* Suggestions d’alternatives
* Validation en masse
* Duplication de planning (semaine type)

#### SF-21 : Gestion des ressources

* CRUD Salles (capacité, équipements)
* CRUD Enseignants (disponibilités, matières)
* CRUD Groupes (étudiants, promotions)
* CRUD Matières (codes, crédits ECTS)

#### SF-22 : Import/Export

* Import CSV/Excel (template fourni)
* Validation des données
* Preview avant import
* Logs d’import détaillés
* Export CSV/Excel/PDF
* Export API (JSON)

#### SF-23 : Historique et audit

* Log de toutes les modifications
* Filtre par date, utilisateur, action
* Comparaison de versions
* Restauration d’une version
* Export des logs

#### SF-24 : Tableaux de bord

* Taux d’occupation des salles
* Charge enseignante
* Répartition CM/TD/TP
* Conflits détectés et résolus
* Statistiques de consultation
* Performances système

#### SF-25 : Notifications admin

* Alertes de conflits
* Alertes de validation requise
* Rapports quotidiens/hebdomadaires
* Anomalies détectées

### 5.4 Module API

#### SF-30 : API REST publique

* Authentification par token
* Rate limiting (100 req/min)
* Documentation OpenAPI/Swagger
* Endpoints :
  + GET /api/v1/schedule/:userId
  + GET /api/v1/courses
  + GET /api/v1/rooms
  + POST /api/v1/schedule/export
* Webhooks pour événements
* SDK JavaScript/Python

## 6. Spécifications techniques

### 6.1 Architecture système

#### Architecture cible : Microservices

┌─────────────────────────────────────────────────────────┐  
│ Load Balancer │  
│ (Nginx / HAProxy) │  
└────────────────┬──────────────────┬─────────────────────┘  
 │ │  
 ┌────────────▼─────────┐ ┌───▼──────────────┐  
 │ Frontend (SPA) │ │ Mobile Apps │  
 │ React/Vue/Angular │ │ iOS / Android │  
 └────────────┬─────────┘ └───┬──────────────┘  
 │ │  
 └──────────┬───────┘  
 │  
 ┌──────────▼──────────┐  
 │ API Gateway │  
 │ (Kong / Apigee) │  
 └──────────┬──────────┘  
 │  
 ┌───────────────────┼───────────────────┐  
 │ │ │  
 ┌────▼─────┐ ┌─────▼──────┐ ┌─────▼──────┐  
 │ Schedule │ │ User │ │ Notif │  
 │ Service │ │ Service │ │ Service │  
 └────┬─────┘ └─────┬──────┘ └─────┬──────┘  
 │ │ │  
 ┌────▼─────┐ ┌─────▼──────┐ ┌─────▼──────┐  
 │ Schedule │ │ User │ │ Message │  
 │ DB │ │ DB │ │ Queue │  
 └──────────┘ └────────────┘ │ (RabbitMQ)│  
 └────────────┘

#### Composants principaux

1. **Frontend**
   * Framework : React 18 ou Vue 3
   * State management : Redux/Pinia
   * UI Library : Material-UI ou Ant Design
   * Build : Vite ou Webpack 5
   * Tests : Jest + React Testing Library
2. **API Gateway**
   * Kong ou AWS API Gateway
   * Rate limiting
   * Authentication/Authorization
   * Logging et monitoring
3. **Microservices**
   * **Schedule Service** : Gestion des emplois du temps
   * **User Service** : Authentification et profils
   * **Notification Service** : Envoi de notifications
   * **Export Service** : Génération PDF/iCal
   * **Analytics Service** : Métriques et statistiques
4. **Bases de données**
   * PostgreSQL 15+ (données relationnelles)
   * Redis (cache, sessions)
   * Elasticsearch (recherche full-text)
5. **Message Queue**
   * RabbitMQ ou Apache Kafka
   * Traitement asynchrone
   * Notifications temps réel
6. **Stockage**
   * S3 ou MinIO (fichiers, exports)
   * CDN (CloudFlare/CloudFront) pour assets

### 6.2 Stack technique recommandée

#### Backend

* **Langage** : Node.js (TypeScript) ou Python (FastAPI)
* **Framework** : NestJS ou Django REST Framework
* **ORM** : TypeORM ou SQLAlchemy
* **Validation** : Joi ou Pydantic
* **Documentation** : Swagger/OpenAPI
* **Tests** : Jest/Pytest, Supertest/httpx

#### Frontend

* **Framework** : React 18 + TypeScript
* **Router** : React Router 6
* **State** : Redux Toolkit + RTK Query
* **UI** : Material-UI v5
* **Forms** : React Hook Form
* **Tests** : Jest + Testing Library
* **E2E** : Playwright ou Cypress

#### Mobile

* **Framework** : React Native ou Flutter
* **Navigation** : React Navigation
* **State** : Redux ou Riverpod
* **Notifications** : Firebase Cloud Messaging
* **Offline** : SQLite + Sync

#### Infrastructure

* **Container** : Docker + Docker Compose
* **Orchestration** : Kubernetes (k8s)
* **CI/CD** : GitHub Actions ou GitLab CI
* **Monitoring** : Prometheus + Grafana
* **Logging** : ELK Stack (Elasticsearch, Logstash, Kibana)
* **APM** : New Relic ou Datadog

#### Sécurité

* **Authentication** : OAuth2 + OIDC
* **Authorization** : RBAC (Role-Based Access Control)
* **SSL/TLS** : Let’s Encrypt
* **WAF** : CloudFlare ou AWS WAF
* **Secrets** : HashiCorp Vault
* **Scanning** : SonarQube, Snyk

### 6.3 Exigences de performance

#### Temps de réponse

* Page d’accueil : **< 1 seconde**
* Chargement calendrier : **< 500ms**
* Recherche : **< 300ms**
* Export PDF : **< 3 secondes**
* API endpoints : **< 200ms** (p95)

#### Capacité

* Utilisateurs simultanés : **1 000+**
* Requêtes par seconde : **500+**
* Croissance annuelle : **20%**
* Peak hours : 8h-10h et 17h-19h

#### Disponibilité

* Uptime : **99.5%** (objectif 99.9%)
* RTO (Recovery Time Objective) : **1 heure**
* RPO (Recovery Point Objective) : **15 minutes**

#### Scalabilité

* Horizontal scaling sur tous les services
* Auto-scaling basé sur CPU/Memory
* Database read replicas
* Cache distribué (Redis Cluster)

### 6.4 Sécurité et conformité

#### Authentification et autorisation

* SSO avec Active Directory/LDAP
* OAuth2 + OpenID Connect
* MFA (Multi-Factor Authentication) optionnel
* Gestion des rôles : Étudiant, Enseignant, Admin

#### Protection des données

* Chiffrement en transit (TLS 1.3)
* Chiffrement au repos (AES-256)
* Anonymisation des logs
* Pseudonymisation RGPD

#### Conformité

* **RGPD** : Droit à l’oubli, portabilité, consentement
* **ISO 27001** : Sécurité de l’information
* **SOC 2** : Contrôles de sécurité

#### Audit et logs

* Logs centralisés (ELK)
* Retention : 1 an
* Logs d’accès et modifications
* Alertes de sécurité

### 6.5 Intégrations

#### Systèmes existants

1. **Active Directory** : Authentification SSO
2. **ERP étudiant** : Import des promotions
3. **LMS (Moodle/Blackboard)** : Synchronisation des cours
4. **CRM** : Export des statistiques
5. **Système de réservation de salles** : API bidirectionnelle

#### Services externes

1. **Email** : SendGrid ou AWS SES
2. **SMS** : Twilio ou Vonage
3. **Push** : Firebase Cloud Messaging
4. **Maps** : Google Maps API (plan campus)
5. **Storage** : AWS S3 ou Azure Blob

#### API REST

* Versioning : /api/v1, /api/v2
* Format : JSON
* Documentation : Swagger UI
* Rate limiting : 100 req/min par token
* Webhooks : notifications d’événements

## 7. Contraintes et exigences

### 7.1 Contraintes techniques

#### Infrastructure

* Hébergement : Cloud (AWS/Azure/GCP) ou On-premise
* Budget infrastructure : 15 000 € / an
* Scalabilité : Support de 5 000 utilisateurs
* Backup : Quotidien avec retention 30 jours

#### Compatibilité

* Navigateurs : Chrome 90+, Firefox 88+, Safari 14+, Edge 90+
* Mobile : iOS 14+, Android 10+
* Accessibilité : WCAG 2.1 niveau AA

#### Langues

* Français (par défaut)
* Anglais
* Support futur : Espagnol, Allemand

### 7.2 Contraintes organisationnelles

#### Équipe projet

* 1 Chef de projet
* 1 Product Owner
* 2 Développeurs Backend
* 2 Développeurs Frontend
* 1 Développeur Mobile
* 1 DevOps
* 1 UX/UI Designer
* 1 QA Engineer

#### Ressources externes

* 1 Expert sécurité (audit)
* 1 Expert RGPD (conformité)
* 1 Formateur (change management)

#### Disponibilité

* Développement : 9 mois
* Budget global : 250 000 €
* Mise en production : Rentrée 2026

### 7.3 Contraintes métier

#### Règles de gestion

1. Un étudiant ne peut pas avoir 2 cours simultanés
2. Un enseignant ne peut pas avoir 2 cours simultanés
3. Une salle ne peut pas être utilisée pour 2 cours simultanés
4. Un cours doit durer minimum 30 minutes
5. Un cours ne peut pas dépasser 4 heures consécutives
6. Un cours doit avoir au moins 1 enseignant
7. Un cours doit avoir au moins 1 étudiant
8. Modifications validées par 2 niveaux (chef de département + direction)

#### Périodes critiques

* **Rentrée** (septembre) : Forte charge
* **Examens** (janvier, juin) : Plannings spéciaux
* **Maintenance** : Nuits et weekends uniquement
* **Support** : Renforcé pendant rentrée

## 8. Planning et jalons

### 8.1 Phases du projet

#### Phase 0 : Cadrage (1 mois)

* Validation du cahier des charges
* Constitution de l’équipe
* Setup de l’infrastructure projet
* **Livrable** : Cahier des charges validé

#### Phase 1 : Conception (2 mois)

* Workshops utilisateurs
* Design System
* Maquettes UX/UI
* Architecture technique détaillée
* **Livrable** : Maquettes validées + Architecture

#### Phase 2 : Développement MVP (3 mois)

* Sprint 1 : Backend API + Auth
* Sprint 2 : Frontend consultation
* Sprint 3 : Frontend administration
* Sprint 4 : Notifications + Export
* Sprint 5 : Tests et corrections
* Sprint 6 : Performance tuning
* **Livrable** : MVP fonctionnel

#### Phase 3 : Tests et recette (1 mois)

* Tests utilisateurs (pilote)
* Tests de charge
* Audit sécurité
* Corrections bugs
* **Livrable** : Application en pré-production

#### Phase 4 : Déploiement (1 mois)

* Migration des données
* Déploiement progressif
* Formation utilisateurs
* Communication
* **Livrable** : Application en production

#### Phase 5 : Stabilisation (1 mois)

* Support renforcé
* Corrections post-prod
* Optimisations
* Retours utilisateurs
* **Livrable** : Application stable

#### Phase 6 : Évolutions (3 mois)

* Application mobile
* Analytics
* Intégrations avancées
* **Livrable** : Fonctionnalités Phase 2

### 8.2 Jalons clés

| Jalon | Date | Critères de validation |
| --- | --- | --- |
| **J1** - Lancement | M0 | Équipe constituée, budget validé |
| **J2** - Conception validée | M2 | Maquettes approuvées, archi OK |
| **J3** - MVP prêt | M5 | Tests unitaires > 80%, démo OK |
| **J4** - Recette validée | M6 | UAT passés, audit sécurité OK |
| **J5** - Production | M7 | 100% utilisateurs migrés |
| **J6** - Stabilisation | M8 | < 5 bugs critiques, satisfaction > 80% |
| **J7** - Mobile déployé | M11 | Apps iOS/Android en stores |

### 8.3 Rétroplanning

M0 : Lancement projet  
M1 : Workshops + Design  
M2 : Validation maquettes  
M3 : Dev Sprint 1-2  
M4 : Dev Sprint 3-4  
M5 : Dev Sprint 5-6 + MVP  
M6 : Tests et recette  
M7 : Déploiement production  
M8 : Stabilisation  
M9 : Phase 2 - Sprint 1-2  
M10 : Phase 2 - Sprint 3-4  
M11 : Phase 2 - Déploiement mobile  
M12 : Amélioration continue

## 9. Risques et mitigation

### 9.1 Risques techniques

#### R-TECH-01 : Performance insuffisante

* **Probabilité** : Moyenne
* **Impact** : Élevé
* **Mitigation** : Tests de charge dès le sprint 2, architecture scalable, cache Redis
* **Plan B** : Optimisation DB, CDN, refactoring si nécessaire

#### R-TECH-02 : Intégration AD/LDAP complexe

* **Probabilité** : Moyenne
* **Impact** : Moyen
* **Mitigation** : POC dès le sprint 1, experts SSO
* **Plan B** : Authentication propriétaire temporaire

#### R-TECH-03 : Migration des données échoue

* **Probabilité** : Faible
* **Impact** : Critique
* **Mitigation** : Tests de migration dès M4, environnement de staging
* **Plan B** : Saisie manuelle assistée, import progressif

#### R-TECH-04 : Bugs en production

* **Probabilité** : Élevée
* **Impact** : Moyen
* **Mitigation** : Tests automatisés > 80%, peer review, staging
* **Plan B** : Hotfix process, rollback automatique

### 9.2 Risques organisationnels

#### R-ORG-01 : Résistance au changement

* **Probabilité** : Élevée
* **Impact** : Élevé
* **Mitigation** : Change management, formation, ambassadeurs
* **Plan B** : Support dédié, tutoriels vidéo, FAQ

#### R-ORG-02 : Équipe sous-dimensionnée

* **Probabilité** : Moyenne
* **Impact** : Élevé
* **Mitigation** : Recrutement anticipé, prestataires externes
* **Plan B** : Priorisation fonctionnalités, phases additionnelles

#### R-ORG-03 : Budget dépassé

* **Probabilité** : Moyenne
* **Impact** : Élevé
* **Mitigation** : Suivi budget mensuel, comité de pilotage
* **Plan B** : Réduction périmètre, phases additionnelles

#### R-ORG-04 : Planning non tenu

* **Probabilité** : Moyenne
* **Impact** : Moyen
* **Mitigation** : Sprints Agile, buffer 20%, suivi hebdo
* **Plan B** : Déploiement décalé, MVP réduit

### 9.3 Risques métier

#### R-BUS-01 : Faible adoption utilisateur

* **Probabilité** : Moyenne
* **Impact** : Critique
* **Mitigation** : UX testing, pilote utilisateurs, formation
* **Plan B** : Campagne communication renforcée, incentives

#### R-BUS-02 : Concurrence interne (ancien outil)

* **Probabilité** : Faible
* **Impact** : Élevé
* **Mitigation** : Décommissionnement ancien outil après migration
* **Plan B** : Période de transition, support double

#### R-BUS-03 : Non-conformité RGPD

* **Probabilité** : Faible
* **Impact** : Critique
* **Mitigation** : Expert RGPD, audit avant prod, DPO impliqué
* **Plan B** : Corrections urgentes, plan d’action conformité

## 10. Budget prévisionnel

### 10.1 Coûts de développement

| Poste | Détail | Coût |
| --- | --- | --- |
| **Équipe interne** | 8 personnes × 9 mois | 180 000 € |
| **Prestataires** | Experts (sécurité, RGPD, UX) | 25 000 € |
| **Licences** | IDE, outils, services | 5 000 € |
| **Formation équipe** | Formation techniques | 3 000 € |
| **Total développement** |  | **213 000 €** |

### 10.2 Coûts d’infrastructure

| Poste | Détail | Coût |
| --- | --- | --- |
| **Hébergement** | Cloud (1 an) | 12 000 € |
| **Services tiers** | Email, SMS, push | 2 000 € |
| **Monitoring** | APM, logs | 1 500 € |
| **Sécurité** | WAF, certificats | 500 € |
| **Total infrastructure** |  | **16 000 €** |

### 10.3 Coûts de déploiement

| Poste | Détail | Coût |
| --- | --- | --- |
| **Formation utilisateurs** | 50 sessions | 10 000 € |
| **Communication** | Campagne interne | 3 000 € |
| **Support renforcé** | 3 mois supplémentaires | 5 000 € |
| **Documentation** | Manuels, vidéos | 2 000 € |
| **Total déploiement** |  | **20 000 €** |

### 10.4 Budget total

| Catégorie | Montant |
| --- | --- |
| Développement | 213 000 € |
| Infrastructure | 16 000 € |
| Déploiement | 20 000 € |
| **Sous-total** | **249 000 €** |
| Contingence (10%) | 25 000 € |
| **TOTAL** | **274 000 €** |

### 10.5 Coûts récurrents (annuels)

| Poste | Coût annuel |
| --- | --- |
| Hébergement cloud | 15 000 € |
| Services tiers | 3 000 € |
| Licences et outils | 2 000 € |
| Maintenance et support (TMA) | 30 000 € |
| **Total récurrent** | **50 000 € / an** |

## Annexes

### Annexe A : Glossaire

* **SSO** : Single Sign-On - Authentification unique
* **LDAP** : Lightweight Directory Access Protocol
* **OAuth2** : Protocole d’autorisation
* **API** : Application Programming Interface
* **RGPD** : Règlement Général sur la Protection des Données
* **MVP** : Minimum Viable Product
* **CI/CD** : Continuous Integration / Continuous Deployment
* **UX** : User Experience
* **UI** : User Interface
* **TMA** : Tierce Maintenance Applicative

### Annexe B : Références

1. **Benchmarks concurrentiels**
   * ADE (Adesoft) - Leader marché français
   * Hyperplanning - Alternative populaire
   * Celcat - Solution internationale
   * Primavera - Gestion de projets académiques
2. **Standards et normes**
   * WCAG 2.1 - Accessibilité web
   * ISO 27001 - Sécurité de l’information
   * OWASP Top 10 - Sécurité applicative
   * RGPD - Protection des données
3. **Documentation technique**
   * React Documentation : https://react.dev
   * NestJS Documentation : https://nestjs.com
   * PostgreSQL Documentation : https://postgresql.org

### Annexe C : Contacts

| Rôle | Nom | Email | Téléphone |
| --- | --- | --- | --- |
| Sponsor | Directeur EPSI | directeur@epsi.fr | 01 XX XX XX XX |
| Product Owner | Resp. Scolarité | scolarite@epsi.fr | 01 XX XX XX XX |
| Tech Lead | DSI | dsi@epsi.fr | 01 XX XX XX XX |
| Support | Helpdesk | support@epsi.fr | 01 XX XX XX XX |

### Annexe D : Documents liés

* Synthèse commerciale et ROI
* Enquête utilisateurs - Résultats détaillés
* Maquettes UX/UI (Figma)
* Architecture technique détaillée
* Plan de tests
* Plan de formation
* Plan de communication

## Validation et approbation

| Rôle | Nom | Signature | Date |
| --- | --- | --- | --- |
| **Chef de projet** |  |  |  |
| **Product Owner** |  |  |  |
| **Directeur technique** |  |  |  |
| **Directeur EPSI** |  |  |  |

**Document confidentiel - Usage interne uniquement**

*Version 1.0 - Octobre 2025* *Workshop Poudlard EPSI/WIS 2025*

✨ *“Le bonheur peut être trouvé même dans les moments les plus sombres, si l’on se souvient d’allumer la lumière.”* - Albus Dumbledore

*Un bon cahier des charges éclaire le chemin du projet.*