

# Plan de Projet pour le Système de Gestion des Absences Scolaires

## Semaine 1 : Configuration et Fondation

### Jours 1-2 : Configuration du Projet

- Mettre en place l'environnement de développement
- Initialiser le projet backend Laravel
- Initialiser le projet frontend React
- Configurer les connexions à la base de données
- Configurer Laravel Sanctum pour l'authentification
- Créer un dépôt GitHub pour le contrôle de version

### Jours 3-5 : Conception de la Base de Données & Modèles Principaux

- Concevoir le schéma de la base de données (tables pour élèves, enseignants, classes, absences)
- Créer les migrations pour toutes les tables
- Construire les modèles principaux avec leurs relations
- Créer des seeders pour les données de test
- Mettre en place le processus de réinitialisation quotidienne automatique des absences (tâche CRON)

## Semaine 2 : Développement Backend

### Jours 6-8 : Système d'Authentification

- Implémenter les endpoints d'inscription et de connexion
- Créer les permissions basées sur les rôles (enseignants vs administrateurs)
- Configurer l'authentification par token API
- Créer des middlewares pour la protection des routes

### Jours 9-12 : Endpoints API Principaux

- Développer les endpoints de gestion des élèves
- Développer les endpoints de gestion des classes
- Créer les endpoints d'enregistrement/mise à jour des absences
- Implémenter les endpoints de filtrage et de rapport des absences
- Écrire des tests unitaires pour les fonctions API critiques

## Semaine 3 : Développement Frontend

### Jours 13-15 : Authentification & Structure

- Créer l'interface de connexion/déconnexion
- Construire la structure de l'application et la navigation
- Implémenter les composants UI basés sur les rôles
- Configurer le routage React

### Jours 16-19 : Implémentation des Fonctionnalités Principales

- Construire l'interface du tableau de bord enseignant
- Créer l'interface de marquage des absences
- Développer le tableau de bord administratif
- Implémenter les écrans de gestion des élèves/classes
- Créer l'interface de rapport
- Ajouter des capacités de filtrage

## Semaine 4 : Intégration, Tests et Déploiement

### Jours 20-22 : Intégration & Raffinement

- Connecter tous les composants frontend à l'API
- Implémenter la gestion des erreurs et la validation des formulaires
- Ajouter des ajustements de design responsive
- Optimiser les appels API et la gestion d'état

### Jours 23-25 : Tests

- Effectuer des tests de bout en bout
- Corriger les bugs et traiter les cas limites
- Réaliser des tests d'utilisabilité
- Optimiser les performances

### Jours 26-28 : Déploiement & Documentation

- Préparer l'environnement de déploiement
- Configurer les tâches CRON pour la production
- Déployer l'application sur le serveur
- Créer la documentation utilisateur
- Documenter les endpoints API pour la maintenance future

### Jours 29-30 : Marge et Ajustements Finaux

- Traiter les problèmes restants
- Effectuer les tests finaux
- Préparer le transfert
- Réviser la documentation

## Composants Techniques Clés à Prioriser

1. **Mécanisme de Réinitialisation Quotidienne :**
  - Créer une commande qui peut être exécutée par CRON
  - La concevoir pour archiver/réinitialiser la table des absences chaque minuit
2. **Système d'Enregistrement des Absences :**
  - Rendre cela aussi efficace que possible avec un minimum de clics
  - Envisager d'utiliser des cases à cocher ou des boutons à bascule pour un marquage rapide
3. **Vues du Tableau de Bord :**
  - Vue enseignant : filtrée pour ne voir que leurs classes
  - Vue administrateur : complète avec options de filtrage
4. **Authentification & Sécurité :**
  - Implémenter des vérifications de rôle appropriées pour chaque route
  - Assurer la séparation des données entre les enseignants