



Neo Start Technology

**PROJET STAGE : REALISATION ET CONCEPTION
D'UNE PLATEFORME DE GESTION DE SUIVI DE
REPETITION EN PRESENTIEL**

CAHIER DE CHARGE DU PROJET

Membres du groupe :

MATCHATOM Balakibawi

TAGBA Tchédre Saturnin

APEDOH Kodjo René

EGBOHOU fridos

ALORISSA fatima

Sommaire

1. Introduction
 - 1.1 Contexte
 - 1.2 Problématique
 - 1.3 Objectif principal
 - 1.4 Objectifs spécifiques
2. Acteurs et rôles
3. Fonctionnalités détaillées
 - 3.1 Gestion des utilisateurs
 - 3.2 Planning des répétitions
 - 3.3 Rapports de séance
 - 3.4 Gestion des incidents
 - 3.5 Rôle du témoin
 - 3.6 Analyse et statistiques
4. Architecture technique
 - 4.1 Frontend
 - 4.2 Backend
 - 4.3 Base de données
5. Contraintes et exigences
 - 5.1 Sécurité
 - 5.2 Performance
 - 5.3 Ergonomie
 - 5.4 Maintenance
6. Livrables
7. Diagrammes UML
8. Planning détaillé – 25 jours

1. Introduction

1.1 Contexte

La gestion des cours particuliers ou répétitions est souvent manuelle, entraînant des difficultés dans la planification des séances, le suivi des élèves et l'évaluation des enseignants. La plateforme web centralise ces informations pour une meilleure organisation et suivi.

1.2 Problématique

Comment créer une solution centralisée qui permette :

- Une planification fiable des séances
- Le suivi détaillé de l'évolution des élèves
- L'évaluation et classement des enseignants
- La communication et validation par tous les acteurs - La gestion des incidents

1.3 Objectif principal

Créer une plateforme web complète permettant planification, suivi, évaluation et analyse des séances pour tous les acteurs.

1.4 Objectifs spécifiques

- Gestion des comptes utilisateurs
- Planning intelligent et flexible
- Rapports détaillés de séances
- Validation et observations par témoin
- Tableau de bord et statistiques
- Sécurité et confidentialité

2. Acteurs et rôles

Acteur	Rôle principal	Détail fonctionnel
Parent	Crée comptes, consulte planning et rapports	Visualisation planning, historique paiements, rapports détaillés
Élève	Accède au planning, rôle de témoin, suit évolution	Notifications, confirmation présence, remarques
Enseignant	Gère séances, remplit rapports	Reprogrammation, assiduité, rapport détaillé
Témoin	Valide présence, note observations	Accès limité aux séances assignées
Administrateur	Supervise utilisateurs et plateforme	Tableau de bord complet : statistiques, incidents, droits d'accès

3. Fonctionnalités détaillées

3.1 Gestion des utilisateurs

- Inscription et connexion sécurisée (JWT)
- Profils modifiables
- Association élèves-parents-enseignants
- Gestion des droits

3.2 Planning des répétitions

- Création, modification, suppression de séances
- Reprogrammation automatique
- Visualisation calendrier par rôle
- Notifications et rappels automatiques

3.3 Rapports de séance

- Évaluation progression/régression
- Mode de paiement
- Recommandations pédagogiques
- Fichiers annexes
- Consultation par parents et admins

3.4 Gestion des incidents

- Signalement par enseignant
- Historique consultable
- Association à séance concernée

3.5 Rôle du témoin

- Validation présence
- Ajout commentaires/observations

3.6 Analyse et statistiques

- Progression des élèves (graphique)
 - Performance des enseignants
 - Suivi financier
 - Tableaux de bord interactifs
-

4. Architecture technique

4.1 Frontend

- Nuxt.js + Tailwind CSS

- Pages clés : Dashboard, Planning, Rapports, Incidents, Statistiques, Profil
- Composants : Calendrier, Tableau, Graphiques, Formulaire

4.2 Backend

- PHP (Laravel) + MySQL
- Modules : Auth, Gestion utilisateurs, Séances, Rapports, Incidents, Stats
- API sécurisée compatible mobile

4.3 Base de données

- Tables : users, eleves, enseignants, temoins, seances, rapports_seance, paiements
- Relations clés : Parent → plusieurs élèves, séance → élève + enseignant + témoins, rapport → séance, incident → séance

5. Contraintes et exigences

5.1 Sécurité

- Hash mot de passe (bcrypt)
- Authentification JWT
- Validation côté serveur et client

5.2 Performance

- Temps de réponse < 2s
- Supporte plusieurs centaines d'utilisateurs simultanés

5.3 Ergonomie

- Interface responsive
- Navigation intuitive
- Notifications claires

5.4 Maintenance

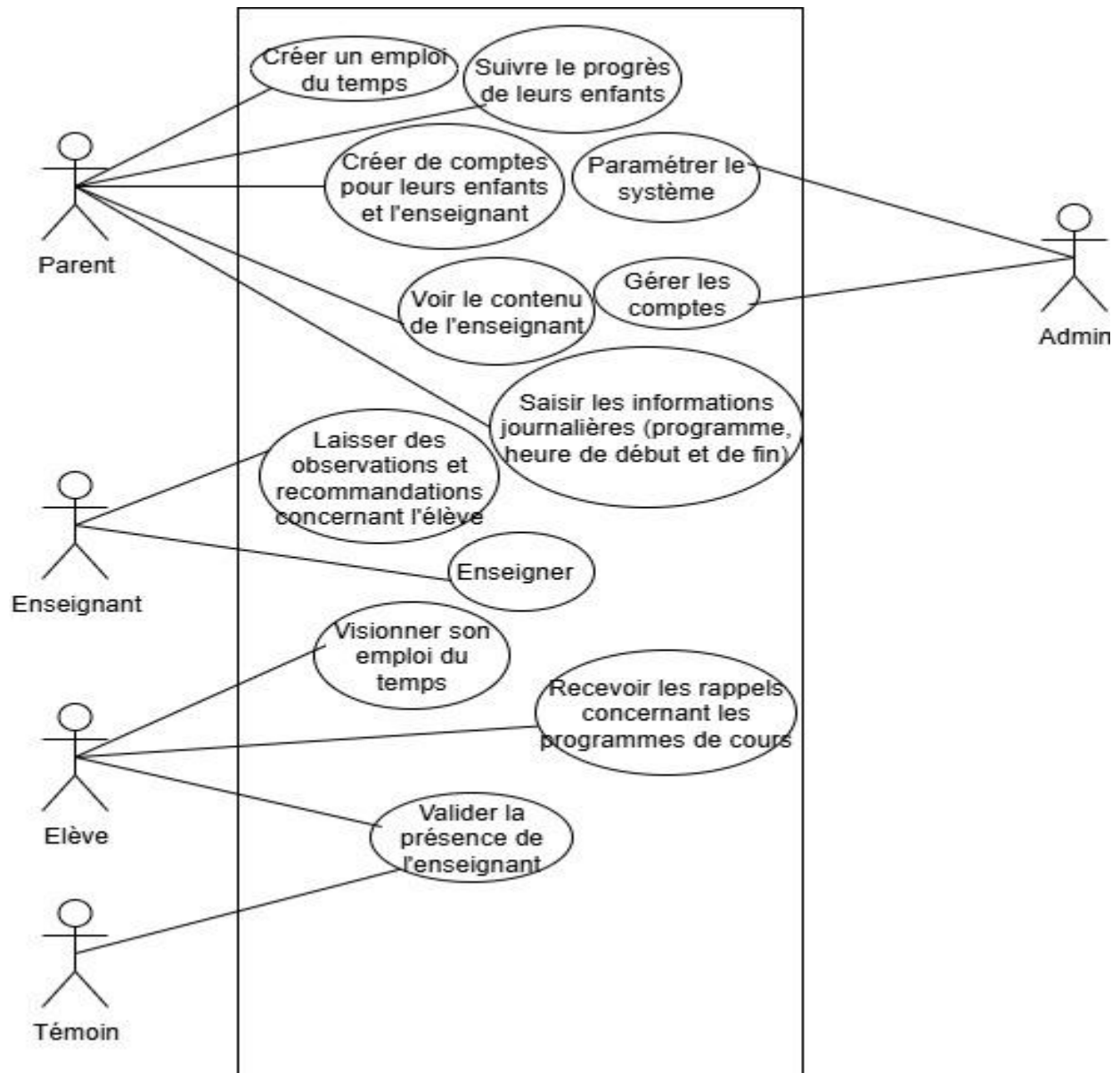
- Code structuré et documenté
- API modulable pour extensions futures

6. Livrables

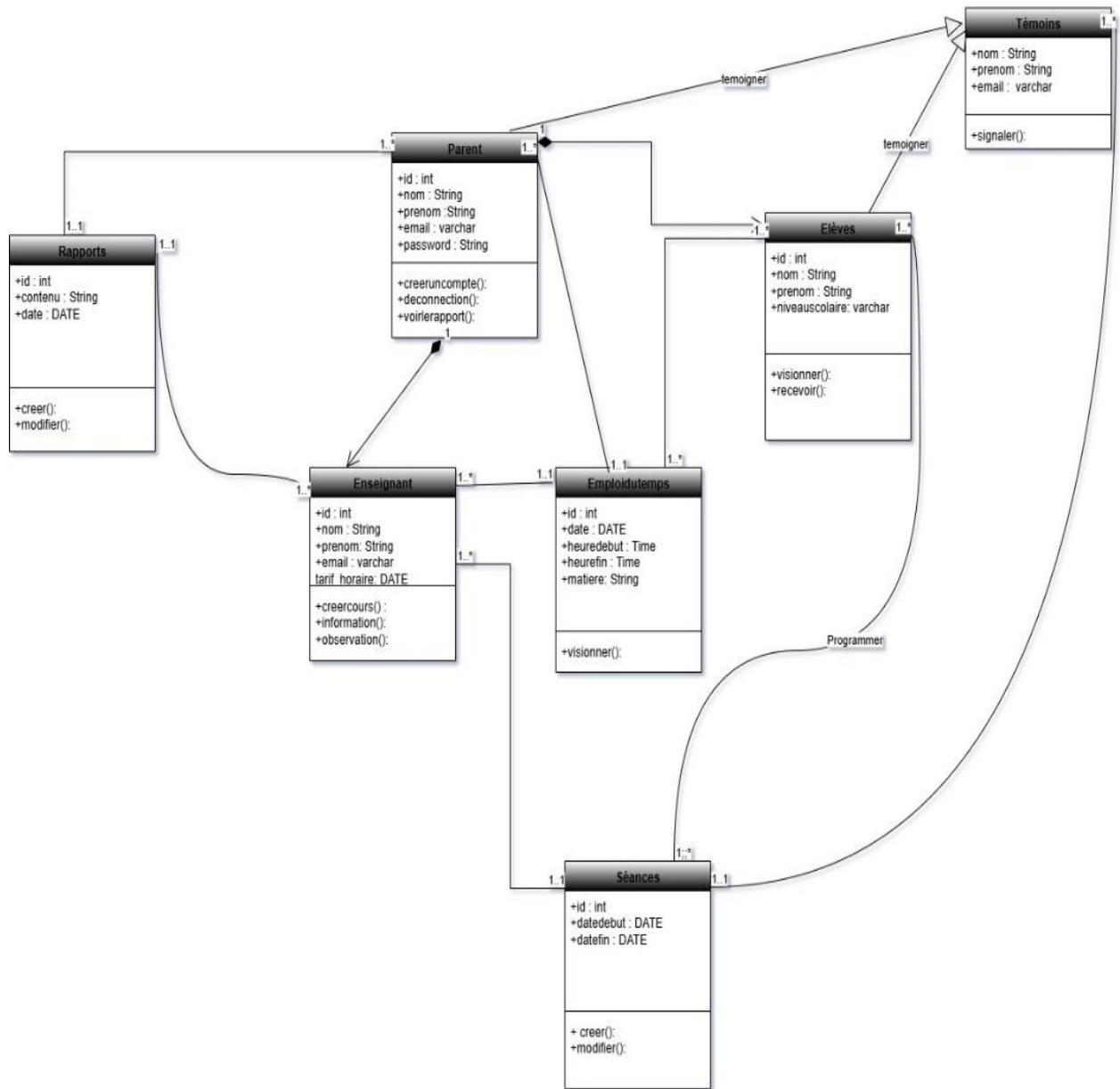
- Plateforme web complète
 - Base de données MySQL
 - Documentation technique et API
 - Guide utilisateur pour chaque rôle
-

7. Diagrammes UML

- Diagrammes Cas d'utilisation



- Diagrammes Classes



8. Planning détaillé – 25 jours

Phase	Durée	Description	Remarques
Conception & maquettage	3 j	Analyse besoins, maquettes, validation rôles	Inclure tous les acteurs
Développement Backend	8 j	API, DB, gestion utilisateurs, séances, rapports	Parallélisable avec frontend
Développement Frontend	8 j	Pages, intégration Tailwind, liaison API	Commence dès que backend prêt
Tests & débogage	4 j	Tests unitaires et intégration, vérification rôles	Inclure tous les acteurs
Déploiement & finalisation	2 j	Mise en production, vérification, documentation	Formation rapide aux utilisateurs
