*planification du projet*

*1. Liste des tâches à réaliser*

***=> Phase 1 – Analyse du projet***

* Définir les rôles : Administrateur, Professeur, Élève
* Identifier les fonctionnalités principales (ex : gestion des notes, absences, emploi du temps)
* Écrire des user stories simples pour chaque utilisateur
* Créer les premiers schémas UML (cas d’utilisation, classes)

***=> Phase 2 – Conception technique***

* Concevoir la base de données (tables + relations)
* Définir les modèles Laravel à créer (User, Classe, Note, etc.)
* Préparer la structure du projet React (pages, composants, navigation)
* Planifier les routes API (backend) et les routes React (frontend)

***=> Phase 3– Développement Back-end (Laravel 12)***

* *Mettre en place l’authentification (Sanctum, JWT ou Laravel Breeze)*
* *Créer les modèles + migrations (avec relations entre les tables)*
* *Développer les contrôleurs API pour gérer :*
* *les utilisateurs*
* *les classes*
* *les matières*
* *les notes*
* *Ajouter les permissions avec des policies ou middleware*
* *Testez routes(API) avec Postman*

***=> Phase 4– Développement front-end (REACT.Js)***

* Créer la page de connexion
* Afficher un tableau de bord différent selon le rôle (élève, prof, admin)
* Faire des appels API (avec axios ou fetch) pour afficher les données
* Mettre en place les routes sécurisées selon le rôle
* Ajouter du style avec Bootstrap ou css, pour une interface claire

***=> Phase 5– Tests***

* Tester la création, la lecture, la modification et la suppression de données
* Vérifier que chaque utilisateur ne voit que ce qu’il a droit de voir
* Corriger les bugs s’il y en a

***=> Phase 6– Déploiement***

* Configurer les variables d’environnement (.env)
* Tester la version en ligne avec des comptes test

*2. Estimation de la durée*

|  |  |
| --- | --- |
| *Taches* | *Durée estimée* |
| *- Définir rôles, user stories, fonctionnalités - Schémas UML* | *2 jours* |
| *- Base de données - Modèles Laravel - Structure React + routes* | *3 jours* |
| *Authentification - Modèles, Migrations - API (CRUD : notes, absences...) - Policies & Middleware* | *10 jours* |
| *Pages : Login, Dashboard, Notes, etc. - Intégration API - Routage + Rôles* | *8 jours* |
| *Stylisation avec Bootstrap - Amélioration responsive & UX* | *2 jours* |
| *Vérification des rôles - Tests fonctionnels API & UI* | *2 jours* |
| *Déploiement Laravel & React - Tests finaux en ligne*   * *- Préparation de la soutenance - Documentation et livrables* | *2 jours*  *1 jours* |

*4. Outils nécessaires au développement*

***Langages principaux****: PHP (Laravel 12) ,JavaScript (React), HTML5 , CSS3, SQL*

***Frameworks / Libs****: LARAVEL 12 (bak-end), React (front-end), Bootsrap (UI / Responsiveness)*

|  |
| --- |
| ***Database*** *:* |

*MySQL (via XAMPP / phpMyAdmin)  
- Modélisation avec MySQL Workbench*

|  |
| --- |
| ***IDE*** *:* |

*- Visual Studio Code (principal)  
- MySQL Workbench (modèle BDD)*

|  |
| --- |
| ***Versioning*** *:* |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
| *- Git (en local) - GitHub (pour héberger le projet et suivre l’évolution du code)* |

|  |
| --- |
| ***Serveur local*** *:* |

*-XAMPP (Apache + Mysql pour executer laravel en local )*

|  |
| --- |
| ***Outils de test*** *:*  *- Postman (pour tester les api Laravel )* |

*-Navigateur web (test des interfaces React )*