

Rapport de Projet Full Stack Laravel

1. Introduction

Dans le cadre de mon cursus universitaire, ce projet a pour objectif de concevoir et développer une application web **full stack** en utilisant le framework **Laravel**. Ce projet permet de mettre en pratique les compétences acquises en développement backend, frontend, bases de données, architecture logicielle et sécurité des applications web.

L'application développée est une **plateforme de gestion de projets et de tâches**, destinée à des utilisateurs souhaitant organiser leur travail, suivre l'avancement de leurs projets et collaborer efficacement.

2. Objectifs du projet

Objectifs pédagogiques

- Comprendre l'architecture MVC (Model–View–Controller)
- Développer une application complète avec Laravel
- Concevoir une base de données relationnelle
- Mettre en place une authentification sécurisée
- Créer une interface utilisateur fonctionnelle
- Gérer un projet de bout en bout

Objectifs fonctionnels

- Authentification des utilisateurs
- Création et gestion de projets
- Création et suivi des tâches
- Gestion des rôles utilisateurs
- Tableau de bord de suivi

3. Cahier des charges

Acteurs du système

- **Utilisateur** : crée et gère ses projets et tâches
- **Administrateur** : gère les utilisateurs et supervise l'application

Fonctionnalités principales

- Inscription et connexion

- Création / modification / suppression de projets
- Ajout de tâches à un projet
- Changement de statut des tâches
- Visualisation de l'avancement d'un projet

Contraintes techniques

- Framework : Laravel
- Base de données : MySQL
- Frontend : Blade, HTML, CSS, JavaScript
- Serveur local : Laravel Sail / XAMPP

4. Architecture technique

Architecture MVC

- **Model** : gestion des données et de la base
- **View** : affichage (Blade)
- **Controller** : logique métier

Organisation du projet

- app/Models
- app/Http/Controllers
- resources/views
- routes/web.php
- database/migrations

5. Backend (Laravel)

Routes

Les routes sont définies dans le fichier web.php et utilisent les contrôleurs Laravel.

Contrôleurs

- AuthController
- ProjectController
- TaskController

Modèles

- User
- Project

- Task

Chaque modèle utilise Eloquent ORM pour interagir avec la base de données.

6. Base de données

Modèle relationnel

- **users** (id, name, email, password)
- **projects** (id, user_id, title, description)
- **tasks** (id, project_id, title, status)

Relations

- Un utilisateur possède plusieurs projets
- Un projet possède plusieurs tâches

7. Frontend

Technologies utilisées

- Blade (templating Laravel)
- HTML / CSS
- JavaScript

Interfaces développées

- Page de connexion
- Tableau de bord
- Page projet
- Gestion des tâches

L'interface est pensée pour être simple, claire et responsive.

8. Sécurité

- Hashage des mots de passe avec Bcrypt
- Middleware d'authentification
- Protection CSRF
- Validation des formulaires

9. Tests et validation

Des tests manuels ont été réalisés afin de vérifier : - Le bon fonctionnement des routes - La création et modification des données - La sécurité de l'authentification

10. Déploiement

Le projet peut être déployé sur un serveur Apache ou via un hébergement compatible Laravel.

Étapes : - Configuration de l'environnement - Migration de la base de données - Configuration du fichier .env

11. Limites et améliorations

Limites

- Pas de notifications en temps réel
- Pas de gestion avancée des rôles

Améliorations possibles

- Ajout d'une API REST
- Notifications (email)
- Application mobile
- Tableau de bord analytique

12. Conclusion

Ce projet m'a permis de développer une application web complète en utilisant Laravel. Il m'a aidé à renforcer mes compétences techniques, ma compréhension de l'architecture logicielle et ma capacité à mener un projet de développement de bout en bout.

Ce travail constitue une base solide pour des projets futurs plus complexes.