

Ecole Supérieure de Technologie

Département d’informatique

**Rapport de Projet Laravel**

## **Application Web – DevProfile**

Réalisé par : Souhayla Ouchen

Encadré par : : RABHI Ouzayr

Année universitaire : 2024/2025

TABLE DES MATIÈRE

1. **Introduction** ................................................................................................. 4

1.1 Présentation du projet

1.2 Objectifs pédagogiques

1. **Description de l’application DevProfile** .................................................... 5

2.1 Fonctionnalités principales

2.2 Utilisateurs ciblés

1. **Architecture de l’application** ...................................................................... 6

  3.1 Structure MVC

  3.2 Schéma des relations (Eloquent ORM)

1. **Technologies et outils utilisés** ................................................................... 8

  4.1 Environnement de développement

  4.2 Frameworks et bibliothèques

  4.3 Base de données

1. **Développement du projet** ...........................................................................10

  5.1 Étapes de réalisation

  5.2 Modèles, contrôleurs et vues

  5.3 Génération du CV en PDF

1. **Sécurité et authentification** ....................................................................... 13

  6.1 Système d’authentification (Breeze)

  6.2 Sécurisation des routes et des données

1. **Tests, validation et déploiement** .............................................................. 15

  7.1 Tests fonctionnels

  7.2 Validation du design

  7.3 Dépôt GitHub et démonstration

1. **Conclusion et perspectives** ...................................................................... 19

  8.1 Bilan du projet

  8.2 Améliorations futures

# **Introduction**

## **1.1 Présentation du projet**

Dans un monde numérique en constante évolution, les développeurs ont besoin de plateformes leur permettant de présenter efficacement leurs compétences et projets. Le projet **DevProfile** s’inscrit dans cette logique. Il s’agit d’une application web développée avec le framework **Laravel**, qui permet à tout développeur de créer un profil personnalisé, gérer ses projets, ses compétences, et générer automatiquement un **CV au format PDF**.

Ce projet a été conçu dans un cadre académique afin d’initier les étudiants aux fondamentaux du développement web moderne, en mettant l'accent sur les bonnes pratiques de structuration et de conception d'applications Laravel.

## **1.2 Objectifs pédagogiques**

Le projet vise plusieurs objectifs pédagogiques essentiels pour les étudiants en première année du parcours **Génie Logiciel et Cybersécurité** :

* Maîtriser l’installation et la configuration d’un projet Laravel avec **Breeze**.
* Comprendre et appliquer le modèle **MVC** (Modèle – Vue – Contrôleur).
* Concevoir et manipuler une base de données relationnelle via **Eloquent ORM**.
* Mettre en place les opérations **CRUD** pour la gestion des projets et des compétences.
* Implémenter un système d’authentification sécurisé.
* Générer dynamiquement un fichier **CV PDF** à partir des données utilisateur.
* Déployer une application fonctionnelle répondant aux standards du web moderne.

# **Description de l’application DevProfile**

## **2.1 Fonctionnalités principales**

L’application **DevProfile** a été conçue pour permettre à un développeur de :

* **Créer un profil public personnalisé**, avec ses informations de base (nom, email, description, localisation).
* **Gérer ses projets** : ajouter, modifier, supprimer, et afficher une liste de projets réalisés avec description, date, technologies utilisées.
* **Gérer ses compétences** : indiquer des compétences techniques sous forme de mots-clés.
* **Afficher un profil public** via une URL personnalisée : /profile/username.
* **Télécharger un CV en PDF**, généré automatiquement à partir des données saisies.

Aperçu des fonctionnalités sous forme de tableau :

|  |  |  |
| --- | --- | --- |
| **Fonctionnalité** | **Description** | **Accès utilisateur** |
| Authentification | Inscription, connexion, déconnexion via Laravel Breeze |  |
| Édition du profil | Modifier nom, bio, photo, adresse, etc. |  |
| Gestion des projets | CRUD complet sur les projets liés à l’utilisateur |  |
| Gestion des compétences | Ajouter/Supprimer des compétences techniques |  |
| Génération du CV (PDF) | Télécharger un fichier PDF structuré depuis les données saisies |  |
| Profil public | Accessible via une route dynamique : /profile/nom\_utilisateur |  |

## **2.2 Utilisateurs ciblés**

Le projet cible principalement les profils suivants :

* **Étudiants développeurs** : souhaitant créer un portfolio de projets dans le cadre de leurs études.
* **Développeurs juniors** : en quête d’un espace pour organiser et diffuser leurs compétences techniques.
* **Recruteurs / Visiteurs** : pouvant consulter les profils publics pour évaluer les candidats.

Le système a été pensé en **mon-utilisateur**, sans rôles hiérarchiques ou d’administration complexe, dans le but de se concentrer sur l’essentiel.

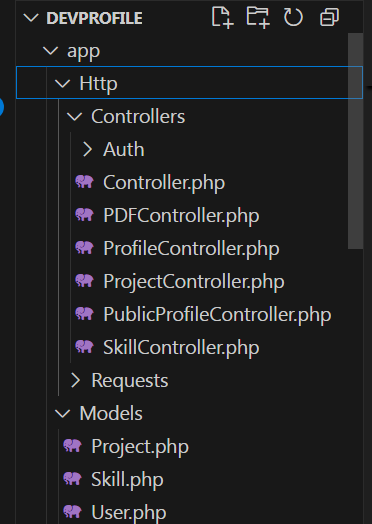
# **Architecture de l’application**

## **3.1 Structure MVC**

L’application suit strictement l’architecture **MVC (Modèle - Vue - Contrôleur)**, standardisée dans Laravel. Cette séparation permet une maintenance facilitée et un développement modulaire.

* **Modèles (Models)** : représentent les entités User, Project, Skill. Ils contiennent la logique métier et les relations Eloquent.
* **Vues (Views)** : fichiers Blade utilisés pour afficher les interfaces (ex. : formulaire de projet, profil public, PDF).
* **Contrôleurs (Controllers)** : gèrent les requêtes HTTP et orchestrent la logique entre modèles et vues.

Exemple de structure des fichiers:

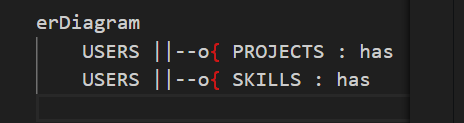


## **3.2 Schéma des relations (Eloquent ORM)**

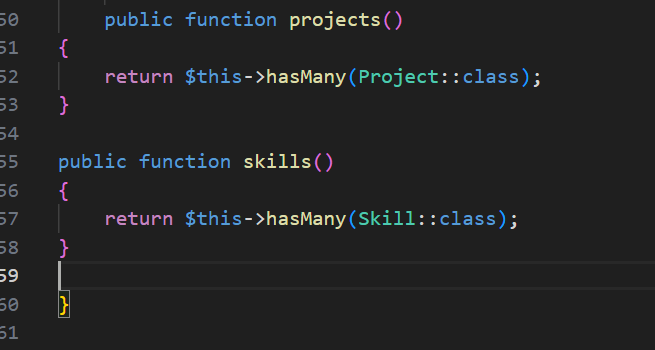
L’application repose sur le système **Eloquent ORM** de Laravel pour gérer les relations entre les entités :

* **Un utilisateur peut avoir plusieurs projets**
* **Un utilisateur peut avoir plusieurs compétences**

### **Schéma relationnel :**



### **Exemple de relation Eloquent dans le modèle User.php :**



Cette approche simplifie la récupération des données liées et rend l’application plus fluide.

# **Technologies et outils utilisés**

## **4.1 Environnement de développement**

Le projet a été développé sur un environnement local, avec les outils suivants :

|  |  |  |
| --- | --- | --- |
| **Outil** | **Version utilisée** | **Rôle dans le projet** |
| Laravel | v11.x | Framework principal |
| PHP | >= 8.2 | Langage backend |
| MySQL | 8.x | Système de gestion de base de données |
| Composer | 2.x | Gestionnaire de dépendances PHP |
| Node.js + NPM | LTS | Compilation des assets (CSS, JS) |

## **4.2 Frameworks et bibliothèques**

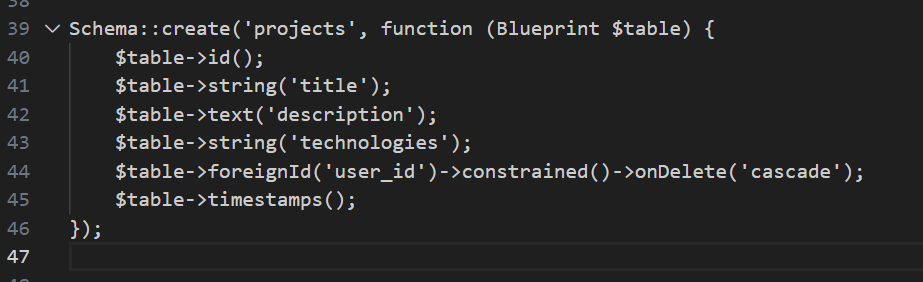
* **Laravel Breeze** : pour l’authentification rapide (stack Blade).
* **Tailwind CSS** : pour le design responsive et moderne.
* **DomPDF** : pour la génération du CV en fichier PDF.
* **Alpine.js** : pour des interactions front-end simples.

## **4.3 Base de données**

Le système utilise **MySQL** avec les tables suivantes :

|  |  |  |
| --- | --- | --- |
| **Table** | **Description** | **Relations** |
| users | Contient les informations personnelles | - |
| projects | Contient les projets liés à chaque utilisateur | user\_id (clé étrangère) |
| skills | Contient les compétences des utilisateurs | user\_id (clé étrangère) |

### **Exemple de migration projects :**



Ce schéma garantit une cohérence forte et une relation directe entre l’utilisateur et ses données.

# 

# **Développement du projet**

## **5.1 Étapes de réalisation**

Le développement de l’application **DevProfile** s’est déroulé selon une approche progressive. Voici les principales étapes suivies :

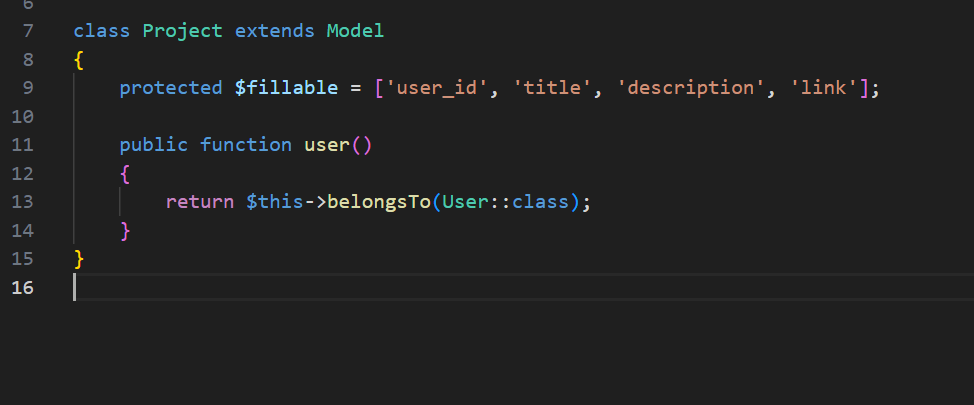
|  |  |
| --- | --- |
| **Étape** | **Description** |
| Initialisation du projet | Création du projet Laravel + installation de Breeze |
| Configuration de la base de données | Connexion à MySQL + création des tables via migrations |
| Définition des relations | Ajout des relations Eloquent entre User, Project, et Skill |
| Implémentation des fonctionnalités CRUD | Gestion des projets et compétences via formulaires Blade |
| Création du profil public | Affichage des données sur /profile/{username} |
| Génération du CV PDF | Intégration de DomPDF pour produire un fichier structuré à imprimer |
| Tests et validation | Vérification du bon fonctionnement des routes, formulaires et PDF |

## **5.2 Modèles, contrôleurs et vues**

L’architecture du projet repose sur les éléments fondamentaux du modèle MVC :

### **🔹 Extraits des modèles (Models)**

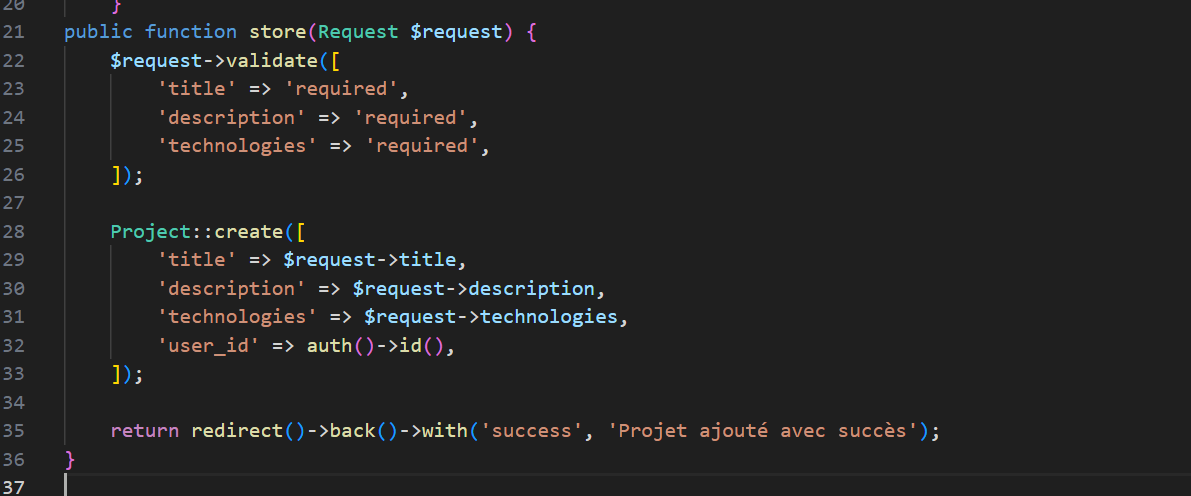
**Exemple : Modèle Project.php**



### **Contrôleurs (Controllers)**

* ProjectController : gère les opérations CRUD.
* SkillController : gère l’ajout et suppression des compétences.
* ProfileController : affiche le profil public et le PDF.

**Extrait typique du ProjectController:**



### **Vues Blade (Templates)**

Les vues sont organisées dans /resources/views/ :

|  |  |
| --- | --- |
| **Fichier** | **Rôle** |
| Profile.blade.php | Affiche le profil public |
| Projects/index.blade.php | Liste des projets |
| Skills/index.blade.php | Liste des compétences |
| Pdf.blade.php | Template du CV exporté en PDF |

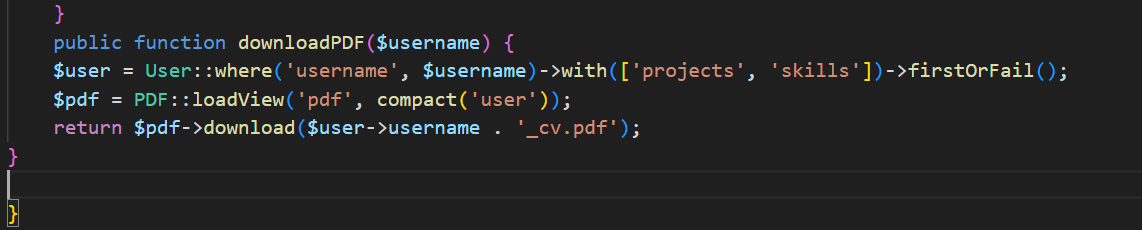
## **5.3 Génération du CV en PDF**

La génération du CV se fait via la bibliothèque **DomPDF**, intégrée dans Laravel.

### **Étapes :**

1. Création d’une route /profile/{username}/pdf
2. Récupération des données via Eloquent
3. Affichage dans une vue pdf.blade.php
4. Conversion via la façade PDF

**Exemple de code :**



# **Sécurité et authentification**

## **6.1 Système d’authentification (Breeze)**

Le système d’authentification est basé sur le package **Laravel Breeze** (stack Blade) :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  | | --- | | **Fonctionnalité** |  |  | | --- | |  | | |  | | --- | | **Fournie par Breeze ?** | |
| Inscription utilisateur |  |
| Connexion / Déconnexion |  |
| Protéger les routes | * via middleware auth |

Structure installée automatiquement :

php artisan breeze:install blade

npm install && npm run dev

php artisan migrate

Une fois installé, Breeze configure :

* Les routes : /login, /register, /logout
* Les vues : auth/login.blade.php, auth/register.blade.php
* Les contrôleurs d’authentification

## **6.2 Sécurisation des routes et des données**

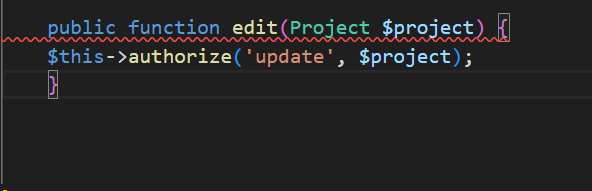
### **Middleware auth**

Toutes les routes sensibles (CRUD, dashboard, génération de CV) sont protégées :



### **Contrôle des accès**

Chaque ressource est filtrée selon l’utilisateur connecté :



### **Protection CSRF et validation**

Laravel intègre par défaut la protection CSRF (@csrf dans les formulaires) ainsi qu’une validation rigoureuse des champs ($request->validate()).

# **Tests, validation et déploiement**

## **7.1 Tests fonctionnels**

Des tests manuels ont été réalisés sur l’ensemble des fonctionnalités :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Fonction testée | Statut | Observations |  |  |
| Authentification |  | Redirection correcte post-login |  |  |
| |  | | --- | | CRUD projets | |  | Fonctionne avec validation |  |  |
| CRUD compétences |  | Ajout/suppression instantanés |  |  |
| |  | | --- | | Profil public (/profile/username) | |  | Affiche bien données publiques |  |  |
| |  | | --- | | Téléchargement PDF | |  | Mise en forme cohérente et propre |  |  |

## **7.2 Validation du design**

Le style visuel a été géré via **Tailwind CSS**, garantissant :

* Un design **responsive**
* Une hiérarchie visuelle claire
* Une interface moderne

Exemple : Affichage responsive du CV en PDF (portrait A4 structuré proprement).

## **7.3 Dépôt GitHub et démonstration**

Le projet a été versionné sur **GitHub**, incluant :

* Le code source complet
* Un fichier README.md détaillé
* Instructions pour l’installation locale

Exemple d’arborescence :

* app/
* database/
* resources/
* routes/
* README.md

Une démonstration directe est prévue le **22 mai**, durée : *10 minutes par groupe*, incluant :

* Présentation du fonctionnement
* Navigation sur l’interface
* Téléchargement d’un CV

# **Conclusion et perspectives**

## **8.1 Bilan du projet**

Le projet **DevProfile** a permis de mettre en œuvre un ensemble de compétences fondamentales en **développement web avec Laravel**, tout en respectant une méthodologie structurée de travail collaboratif. L'application développée est fonctionnelle, intuitive et répond parfaitement aux objectifs pédagogiques fixés :

**Compétences acquises** :

* Maîtrise de la structure MVC dans un framework PHP moderne (Laravel)
* Utilisation efficace d’**Eloquent ORM** pour gérer les relations complexes entre entités
* Intégration de **bibliothèques externes** (DomPDF, Breeze, Tailwind CSS)
* Développement d’interfaces **responsives** et ergonomiques avec Blade + Tailwind
* Gestion complète de l’**authentification sécurisée**
* Génération dynamique de documents PDF à partir des données utilisateurs

En résumé, ce projet a offert une expérience immersive dans le **cycle de vie complet** d'une application Laravel — depuis l'initialisation, la conception des modèles, jusqu’au déploiement final sur GitHub et la démonstration en direct.

## **8.2 Améliorations futures**

Bien que l’application soit stable et conforme aux exigences initiales, plusieurs pistes d’amélioration sont envisageables pour étendre ses fonctionnalités et sa robustesse :

|  |  |
| --- | --- |
| **Domaine** | **Amélioration proposée** |
| Expérience utilisateur | Ajouter la possibilité de **prévisualiser le CV** avant téléchargement |
| Fichiers multimédias | Permettre l’**upload de fichiers** (photo de profil, annexes, portfolio PDF, etc.) |
| Référencement | Améliorer le **SEO** du profil public pour une meilleure visibilité sur Google |
| Mobile | Développer une version mobile ou **Progressive Web App (PWA)** |
| ️ Sécurité avancée | Implémenter la **vérification en deux étapes** ou la gestion de rôles (admin, user) |
| Statistiques | Ajouter un tableau de bord avec **statistiques d’activité** ou de visualisations |
| Multilingue | Rendre l’interface **bilingue (Français / Anglais)** pour une plus large audience |

### **Perspective pédagogique**

Ce projet pourrait être réutilisé ou étendu comme **base de travail pour d’autres modules** (DevOps, cybersécurité, base de données, tests automatisés). Il peut également faire office de projet personnel dans un portfolio professionnel.

**Mot de la fin**  
 Ce travail collaboratif a renforcé non seulement nos compétences techniques, mais aussi nos capacités à travailler en équipe, à documenter proprement notre code, et à présenter un produit final avec professionnalisme.