# SOMMAIRE

[SOMMAIRE i](file:///D:\Doc1.docx#_Toc204618268)

[DEDICACE ii](file:///D:\Doc1.docx#_Toc204618269)

[REMERCIEMENTS iii](file:///D:\Doc1.docx#_Toc204618270)

[AVANT-PROPOS iv](file:///D:\Doc1.docx#_Toc204618271)

[Abréviations v](file:///D:\Doc1.docx#_Toc204618272)

[Liste des tableaux et des figures vii](file:///D:\Doc1.docx#_Toc204618273)

[Résume viii](file:///D:\Doc1.docx#_Toc204618274)

[ABSTRACT ix](file:///D:\Doc1.docx#_Toc204618275)

[Introduction 1](file:///D:\Doc1.docx#_Toc204618276)

[PARTIE I : PRÉSENTATION DE L’ÉCOLE SUPÉRIEURE LA CANADIENNE (ESCa) 2](file:///D:\Doc1.docx#_Toc204618277)

[PARTIE II : DEVELOPPEMENT DU STAGE 6](file:///D:\Doc1.docx#_Toc204618278)

[9](#_Toc204618279)

[PARTIE III : système de gestion des ressources humaines 9](file:///D:\Doc1.docx#_Toc204618280)

[Chapitre II : CONCEPTION DE LA BASE DE DONNÉES & ARCHITECTURE TECHNIQUE DU PROJET 14](#_Toc204618281)

[I- CONCEPTION DE LA BASE DE DONNÉES 14](#_Toc204618282)

[II - ARCHITECTURE TECHNIQUE DU PROJET 16](#_Toc204618283)

[Chapitre III - DÉVELOPPEMENT DU PROJET 20](#_Toc204618284)

[PARTIE IV : RECOMMANDATIONS ET SUGGESTIONS 33](file:///D:\Doc1.docx#_Toc204618285)

[CONCLUSION GENERAL 35](file:///D:\Doc1.docx#_Toc204618286)

[Bibliographie x](file:///D:\Doc1.docx#_Toc204618287)

[Annexe xi](file:///D:\Doc1.docx#_Toc204618288)

# DEDICACE

A

MA FAMILLE

# REMERCIEMENTS

Je tiens à exprimer ma profonde gratitude à l’ensemble des personnes qui, de près ou de loin, ont apporté leur contribution précieuse à la réalisation de ce travail. Leurs soutiens, conseils et encouragements ont été déterminants tout au long de cette expérience. Je souhaite ainsi adresser mes remerciements les plus sincères notamment à :

* **M. TETCHA TAKOU THIERRY**, promoteur du Centre de Formation Professionnel La Canadienne.
* **MADAME LILI MARIAME Laure**, Directrice du Centre de Formation Professionnel La Canadienne, pour ses conseils avisés et son dévouement exemplaire.
* **M. WAFFO LELE Rostand**, Mon encadreur professionnel pour son accompagnement constant, ses précieux conseils et son encadrement bienveillant.
* **L’équipe pédagogique et administrative de l’établissement** pour m’avoir offert cette précieuse opportunité d’apprentissage et de perfectionnement.
* **Mes parents**, pour leur soutien indéfectible, tant sur le plan moral que financier, tout au long de mon parcours.
* **Mes camarades de stage**, pour leur engagement constant, leur esprit de collaboration et leur soutien tout au long de cette expérience formatrice.
* **L’ensemble du personnel du CFPC**, pour l’accueil chaleureux, le professionnalisme et l’environnement de travail convivial qui ont largement contribué à la qualité de mon stage.

# AVANT-PROPOS

Le présent document constitue le fruit d’une année d’apprentissage au **Centre de Formation Professionnelle La Canadienne**. Le CFP La Canadienne, agréé par le **MINEFOP**, a pour vocation de préparer les jeunes à mobiliser leurs compétences dans divers domaines professionnels. Implanté au marché B de Bafoussam, cet établissement propose des formations qualifiantes conduisant à l’obtention du **Diplôme de Qualification Professionnelle** dans plusieurs filières, notamment :

* Webmaster
* Secrétariat bureautique
* Comptabilité informatisée
* Maintenance des systèmes informatiques
* Développement web
* Graphisme de production
* Secrétariat de direction
* Maintenance des réseaux informatiques

Les formations dispensées au sein du CFP La Canadienne s’étendent sur une durée d’une année. Durant ce parcours, chaque apprenant est amené à réaliser un **rapport de stage**, constituant une étape déterminante pour la validation des acquis et l’obtention du diplôme. Ce dispositif pédagogique revêt une importance majeure en ce qu’il permet aux apprenants de consolider leurs connaissances théoriques tout en développant des compétences pratiques essentielles. L’obtention du diplôme facilite par la suite l’accès à de nombreuses opportunités professionnelles.

# Abréviations

| **Abréviation** | **Signification** | **Description / Rôle dans le projet** |
| --- | --- | --- |
| **API REST** | Application Programming Interface (REST) | Interface permettant la communication entre applications via le protocole HTTP. |
| **CMS** | Content Management System | Système de gestion de contenu (non utilisé directement, mais mentionné dans le contexte web). |
| **CRUD** | Create, Read, Update, Delete | Opérations fondamentales sur les données dans le projet Laravel. |
| **CSRF** | Cross-Site Request Forgery | Protection Laravel contre les attaques via formulaire (@csrf). |
| **CSS** | Cascading Style Sheets | Feuilles de style utilisées avec Tailwind CSS pour la mise en forme. |
| **DQP** | Diplôme de Qualification de Professionnelle | Nom du projet ou livrable principal à remettre. |
| **DOM** | Document Object Model | Représentation de la structure HTML manipulée via JavaScript ou Alpine.js. |
| **ESCA** | École Supérieure la canadienne | Établissement ou contexte d’exécution du projet. |
| **CFPC** | Centre de Formation Professionnelle la canadienne | Mentionnée dans certains contextes RH. |
| **Git** | Git (système de versionnage) | Utilisé pour gérer le code source localement. |
| **GitHub** | Plateforme GitHub | Hébergement du code source et collaboration à distance. |
| **GRH** / **RH** | (Gestion des) Ressources Humaines | Domaine principal du projet Laravel (application RH). |
| **HTTP** | HyperText Transfer Protocol | Protocole utilisé pour les requêtes web. |
| **MCD** | Modèle Conceptuel de Données | Schéma de base de données au niveau conceptuel (UML ou Merise). |
| **MLD** | Modèle Logique de Données | Schéma logique de la base (tables, clés étrangères). |
| **MVC** | Modèle-Vue-Contrôleur | Architecture utilisée par Laravel. |
| **NPM** | Node Package Manager | Utilisé pour installer Tailwind CSS, Alpine.js, etc. |
| **ORM** | Object Relational Mapping | Technique de mapping entre objets PHP et tables SQL (Eloquent dans Laravel). |
| **PATCH / PUT** | Méthodes HTTP | Utilisées pour mettre à jour des données via formulaire Laravel. |
| **PC** | Poste Client | Ordinateur de développement ou test du projet. |
| **PDF** | Portable Document Format | Format d’export des données généré avec DomPDF. |
| **PHP** | Hypertext Preprocessor | Langage de programmation utilisé avec Laravel. |
| **POO** | Programmation Orientée Objet | Paradigme utilisé dans Laravel (modèles, contrôleurs…). |
| **SGRH** | Système de Gestion des Ressources Humaines | Objet du projet Laravel. |
| **UML** | Unified Modeling Language | Langage de modélisation utilisé pour les diagrammes (cas d’utilisation, classes…). |
| **UI/UX** | User Interface / User Experience | Désigne le design et l’expérience utilisateur (réalisés avec Tailwind CSS). |

# Liste des tableaux et des figures

# Résume

Le stage professionnel, intégré au cursus de formation au **DQP,** a pour objectif principal de mettre en pratique les connaissances théoriques acquises et de familiariser les étudiants avec le monde professionnel. Notre stage, effectué à **l’ESCA (**École Supérieure La Canadienne) du 2 juin au 31 juillet 2025, nous a permis de réaliser diverses tâches. La principale mission consistait à concevoir et développer une application web de gestion des ressources humaines, visant à informatiser les processus manuels de suivi du personnel. Ce rapport retrace les étapes clés de notre projet, s’inspirant du cas spécifique de l’École Supérieure La Canadienne.

# ****ABSTRACT****

The professional internship, integrated into the DQP training curriculum, primarily aims to apply the theoretical knowledge acquired and to familiarize students with the professional environment. Our internship, carried out at ESCA (École Supérieure La Canadienne) from June 2 to July 31, 2025, allowed us to undertake various tasks. The main mission was to design and develop a web application for human resources management, intended to computerize the manual processes involved in employee tracking. This report outlines the key stages of our project, based on the specific case of École Supérieure La Canadienne.

# Introduction

Dans un contexte économique marqué par la compétitivité et les mutations constantes du marché du travail, la gestion optimale des ressources humaines constitue un enjeu stratégique majeur pour toute organisation. Les ressources humaines, considérées comme le capital le plus précieux de l’entreprise, requièrent une gestion rigoureuse et efficace afin d’assurer le développement des compétences, la motivation du personnel et l’atteinte des objectifs organisationnels.

Dans le cadre de ma formation au **Centre de Formation Professionnelle Canadienne (CFPC)**, j’ai eu l’opportunité d’effectuer un stage pratique au sein de l’**ESCA**, établissement d’accueil, où j’ai été intégré au service chargé de la gestion des ressources humaines. Ce stage m’a permis de mettre en application les connaissances théoriques acquises au CFPC, tout en découvrant les réalités et les exigences du milieu professionnel.

Le présent rapport vise à présenter les travaux réalisés durant cette période de stage, ainsi qu’à analyser le fonctionnement du système de gestion des ressources humaines de l’ESCA. Il exposera dans un premier temps le cadre général du stage, avant de détailler les différentes missions effectuées, les compétences développées, les difficultés rencontrées ainsi que les solutions proposées.

L'objectif principal de ce travail est de démontrer l’importance d’un système de gestion des ressources humaines performant dans l’amélioration continue des performances de l’entreprise.

# ****PARTIE I : PRÉSENTATION DE L’ÉCOLE SUPÉRIEURE LA CANADIENNE** (ESCa)**

### ****I. SECTEUR D’ACTIVITÉ****

L’École Supérieure la Canadienne (ESCa) est un établissement d’enseignement supérieur situé à Bafoussam, entre la pharmacie Marché B et la CCA Bank. Elle propose une gamme variée de programmes de formation professionnelle visant à former des étudiants qualifiés, prêts à répondre aux exigences du monde du travail.

### ****II. PROGRAMMES DE FORMATION****

L’ESCa dispense une diversité de formations supérieures permettant aux étudiants de se spécialiser dans plusieurs domaines. Les filières disponibles sont les suivantes:

* Agriculture et élevage
* Arts et métiers de la culture
* Commerce et vente
* Économie sociale et familiale
* Gestion
* Génie civil
* Génie informatique
* Études médico-sanitaires
* Sciences et techniques biomédicales

Ces programmes sont conçus pour offrir une formation approfondie, pratique et adaptée aux réalités du marché de l’emploi, en intégrant les outils technologiques et les pratiques pédagogiques actuelles.

### ****III. ACCOMPAGNEMENT PERSONNALISÉ****

L’établissement met un point d’honneur à offrir un accompagnement individualisé à ses étudiants, en les encadrant dans leurs projets académiques et professionnels. Cet encadrement vise à maximiser leurs chances de réussite en leur garantissant l’accès à des stages de qualité et à des opportunités professionnelles concrètes, en collaboration avec des entreprises partenaires réparties sur l’ensemble du territoire national.

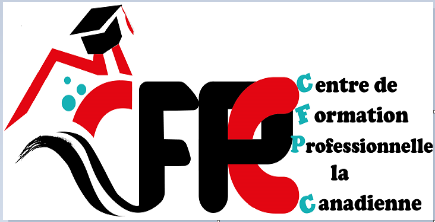
### ****IV. PARTENAIRES DE L’ESCA****

L’ESCa développe des relations de partenariat basées sur la confiance avec des institutions académiques et des entreprises locales et internationales. Ces partenariats visent à favoriser l’insertion professionnelle des étudiants en facilitant leur immersion dans le monde du travail.

Les partenaires de l’ESCa incluent :















### ****V. FICHE D’IDENTIFICATION DE L’ÉTABLISSEMENT****

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Table 1:fiche d’identification de l'ESCa   |  |  | | --- | --- | | **Élément** | **Information** | | **Nom de l’établissement** | École Supérieure la Canadienne | | **Année de création** | 2022 | | **Pays** | Cameroun | | **Adresse** | 837, Bafoussam | | **Boîte postale** | 837 Bafoussam | | **Courriel** | contact@escanadienne.com | | **Domaine d’activité** | Formation professionnelle | | **Téléphone** | +237 695 829 230 / 671 337 829 | | **Site web** | <https://escanadienne.com> | |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

|  |  |
| --- | --- |
|  | VI. ORGANIGRAMME |

**PROMOTEUR**

DIRECTEUR GENERAL

CONSULTANT

SECRETARIAT DE DIRECTION 1

SECRETARIAT DE DIRECTION 2

# PARTIE II : DEVELOPPEMENT DU STAGE

## **I. Objectifs et domaine d’activités de l’entreprise**

L’École Supérieure La Canadienne (ESCA) est un établissement d’enseignement supérieur privé situé à Bafoussam, spécialisé dans les formations techniques et professionnelles. Son domaine d’activités couvre principalement :

* La formation dans les filières : Agriculture et élevage, Arts et métiers de la culture, Commerce et vente, Economie sociale et familiale, Génie civil, Génie informatique, Études médico-sanitaires, Sciences et techniques biomédicales.
* L’accompagnement des étudiants dans leur insertion professionnelle
* La mise en place de projets pédagogiques innovants

**Objectifs du stage :**

Le stage a pour objectif principal de permettre à l'étudiant de :

* Mettre en application les connaissances acquises durant la formation
* Découvrir l’environnement professionnel et s’y adapter
* Participer activement à la conception et au développement de projets web en entreprise
* Acquérir une autonomie technique et fonctionnelle

## **II. Introduction au déroulement du stage**

Le stage a été effectué au sein du département informatique de l’ESCA sur une période de **huit semaines** (du 2 juin au 31 juillet 2025). Il a permis d’aborder plusieurs technologies backend et frontend à travers des projets concrets, en particulier la réalisation d’une plateforme de gestion des ressources humaines (SGRH) propre à l’école.

Le tableau ci-dessous présente de façon synthétique les différentes phases du stage, réparties par semaine, en indiquant les thèmes abordés, les activités réalisées et les compétences développées.

### III. Tableau récapitulatif du déroulement hebdomadaire du stage

**Table 2:fiche du déroulement du stage**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Semaine | Période | Thème étudié | Activités réalisées | **Compétences développées** |
| Semaine 1 | 2 - 9 Juin | PHP POO / WordPress | Finalisation de projets orientés objets en PHP, introduction à WordPress | Maîtrise de la POO, bases de CMS WordPress |
| Semaine 2 | 9 - 15 Juin | Laravel (CRUD) | Création de projets Laravel avec opérations CRUD (Create, Read, Update, Delete) | Compréhension du modèle MVC, manipulation de base de données |
| Semaine 3 | 16 - 22 Juin | Laravel (Gestion des rôles) | Intégration de Spatie Laravel Permission pour gérer les rôles (Directeur, Étudiant, etc.) | Gestion des utilisateurs par rôle |
| Semaine 4 | 23 - 1 Juillet | Laravel (Permissions fines) | Assignation de permissions spécifiques aux rôles, sécurisation des routes | Sécurité, access control, middleware Laravel |
| Semaine 5 | 2 - 8 Juillet | Laravel + Flutter (connexion API) | Création d’API Laravel pour future intégration avec Flutter | Création et test d’API REST avec Laravel Sanctum |
| Semaine 6 | 9 - 15 Juillet | Début du projet RH | Analyse, maquettes, structure du projet ESCA RH | Méthodologie projet, maquettage, architecture Laravel |
| Semaine 7 | 16 - 22 Juillet | Frontend (React) | Initiation à React pour la partie interface utilisateur | Création de composants, gestion d’état (state), JSX |
| Semaine 8 | 23 - 31 Juillet | Laravel + Flutter avancé | Communication Laravel ↔ Flutter, tests, début du frontend mobile | Intégration d’API, construction d’une interface mobile avec Flutter |

## **III. Problèmes rencontrés et solutions apportées**

Durant la période de stage à l’ESCA, bien que le projet ait globalement évolué de manière satisfaisante, deux difficultés majeures ont été rencontrées et ont eu un impact sur le déroulement normal du travail :

**1. Manque de personnel encadrant**

L’un des obstacles les plus marquants a été **le manque de personnel de formation disponible** pour encadrer les stagiaires. Cela a parfois conduit à:

* Des retards dans la validation de certaines étapes (maquettes, choix techniques…)
* Des temps morts ou d’attente avant de pouvoir passer à une phase suivante
* Un accompagnement technique limité sur certaines technologies complexes

**Solution apportée:**

* Développement d’une autonomie plus forte : j’ai pris l’initiative de m’auto-guider grâce à des ressources en ligne fiables (Laravel documentation, YouTube).
* Mise en place d’un système de gestion personnelle des tâches avec un Trello ou Notion pour suivre mes progrès.
* Demande d’aide ponctuelle aux développeurs seniors ou aux collègues disponibles pour débloquer les points bloquants.

**2. Problèmes de connexion Internet**

La seconde difficulté a été **l’instabilité de la connexion Internet** dans les locaux, ce qui a affecté :

* L’accès à la documentation et aux bibliothèques en ligne
* La mise à jour de packages via Composer ou NPM

**Solution apportée:**

* Téléchargement en local des documentations Laravel, Tailwind et Flutter pour consultation hors-ligne.
* Utilisation de hotspots mobiles personnels lorsque cela était nécessaire pour lancer une commande importante.
* Planification du travail de développement de manière à effectuer les tâches ne nécessitant pas de connexion (maquettage, tests unitaires, code Blade) durant les coupures.

# 

# PARTIE III : système de gestion des ressources humaines

**Chapitre 1** : **phase d’analyse**

I - introduction au projet

Dans un monde professionnel de plus en plus compétitif et numérique, la **gestion des ressources humaines (GRH)** joue un rôle stratégique au sein des entreprises. Elle ne se limite plus à l'administration du personnel, mais s’étend à des enjeux majeurs comme l’optimisation des talents, le suivi des carrières, la gestion des compétences ou encore l’analyse des performances. Pour répondre à ces exigences, les entreprises doivent s’appuyer sur des outils technologiques performants et adaptés à leurs réalités.

## 1.Contexte général

La gestion des ressources humaines dans les organisations nécessite aujourd’hui un traitement fluide et structuré de l'information relative aux employés. Cette gestion comprend la tenue des dossiers du personnel, l'affectation des emplois, le suivi des salaires, des grades, des affectations régionales, ainsi que des historiques professionnels.

Cependant, dans de nombreuses petites et moyennes structures, cette gestion se fait encore de manière manuelle ou semi-automatisée, avec des outils comme Excel ou des fichiers papier. Cette situation crée des limites en termes de fiabilité, de temps de traitement, de sécurité des données et de difficulté d’accès rapide à l’information.

## 2.Problématique de la gestion manuelle des employés

La gestion manuelle présente plusieurs inconvénients majeurs :

* **Perte de temps** pour retrouver ou mettre à jour des informations sur un employé.
* **Erreurs humaines** fréquentes dans la saisie ou le traitement des données.
* **Manque de traçabilité** et d’historique sur les évolutions de carrière, les affectations ou les changements de poste.
* **Difficultés de reporting**, rendant complexe la production de rapports, de statistiques RH, ou d’exports professionnels.

Face à ces limites, il devient indispensable de mettre en place un **système informatique** capable d’automatiser et de structurer l’ensemble des processus RH.

## 3.Objectif principal du projet

Le projet intitulé **SGRH – Système de Gestion des Ressources Humaines** a pour but de **concevoir une application web intuitive, sécurisée et performante**, destinée à gérer les employés d'une organisation fictive nommée ESCA. Ce système permettra:

* La gestion des informations du personnel (nom, prénom, email, téléphone…)
* L’affectation des emplois, grades et régions
* Le suivi automatisé des historiques professionnels
* L’export de données au format **PDF** et **Excel**
* La recherche et le filtrage multi-critères (nom, région, emploi, grade, etc.)

L’objectif est donc de **remplacer les pratiques manuelles** par une plateforme dynamique et évolutive, répondant aux standards du développement moderne.

## 4.Méthodologie adoptée pour le développement

Pour la réalisation du projet, une démarche agile a été adoptée, articulée autour de plusieurs phases :

1. **Analyse des besoins** – Identification des fonctionnalités nécessaires pour le système RH.
2. **Conception de la base de données** – Définition des entités (utilisateur, emploi, grade, région, historique des employés) et de leurs relations.
3. **Développement technique**:

* **Backend :** Le framework PHP **Laravel** a été utilisé pour sa structure MVC, sa sécurité intégrée, son ORM Eloquent, et ses puissantes fonctionnalités (routes, requêtes, migrations, etc.)
* **Frontend :** La bibliothèque CSS **Tailwind CSS** a permis de créer une interface moderne, responsive, en **mode clair et sombre**, avec des badges, des icônes et des effets dynamiques.
* **Exportation :** Les packages **Laravel DomPDF** et **Laravel Excel** ont été intégrés pour permettre l'export des données au format **PDF** et **Excel**, avec une mise en page professionnelle (logo, en-tête, pied de page, couleurs).

1. **Tests** – Des tests manuels ont été réalisés à chaque étape pour valider le bon fonctionnement des fonctionnalités.

Ce projet s’inscrit dans une logique de **transformation digitale** de la gestion RH, et constitue une solution solide pour toute structure souhaitant améliorer l'efficacité et la traçabilité de ses opérations internes.

**II - ANALYSE DES BESOINS**

La phase d’analyse des besoins constitue une étape primordiale dans tout projet de développement logiciel. Elle permet de cerner les attentes des utilisateurs, de formaliser les objectifs fonctionnels et de définir clairement les contours de l’application à réaliser. Dans le cadre de ce projet, il s’agit de concevoir une application web dédiée à la gestion efficace des ressources humaines au sein d'une structure fictive nommée **ESCA**.

## 1. Objectifs fonctionnels du système

L’objectif principal est de développer un outil permettant une **gestion automatisée, fluide et sécurisée des employés**, de leurs postes, grades et historiques. Le système doit également permettre des fonctionnalités de **consultation, modification et exportation des données RH** de manière ergonomique et professionnelle.

Voici les fonctionnalités principales que devra couvrir le système :

### 1.1 Gestion des employés

Le module "Employés" est le cœur du système. Il permet d’enregistrer les informations personnelles et professionnelles des utilisateurs. L’administrateur pourra:

* Créer un nouvel employé via un formulaire sécurisé ;
* Modifier les données existantes (email, téléphone, région, emploi…) ;
* Supprimer un profil si nécessaire ;
* Lister tous les employés de façon paginée, avec des options de recherche et de tri.

### 1.2 Gestion des emplois, grades et régions

Chaque employé appartient à une région, un emploi et un grade. Ces éléments sont eux-mêmes **gérés indépendamment** dans des tables séparées, pour assurer la normalisation de la base de données.

* Le module "Emplois" permettra de gérer les intitulés de poste.
* Le module "Grades" regroupe les niveaux hiérarchiques de l’organisation.
* Le module "Régions" permet de localiser géographiquement les employés.

### 1.3 Suivi des historiques d’employés

Ce module permet de **garder une trace des changements** :

* Date de début et de fin d’un poste
* Salaire minimal et maximal
* Grade et région associés à la période concernée
* Possibilité d'ajouter manuellement des dates de création/modification (utile pour des imports ou rétroactions)

Ce suivi permet une **traçabilité complète** du parcours d’un salarié au sein de l’organisation.

### 1.4 Espace personnel de l’employé

Chaque employé doit pouvoir accéder à un espace personnel où il peut :

* Se connecter avec ses identifiants
* Consulter ses informations (nom, prénom, emploi, grade…)
* Mettre à jour son mot de passe ou certaines informations personnelles

Cela renforce l’autonomie des utilisateurs et limite les interventions RH sur les tâches simples.

### 1.5 Exportation des données (PDF et Excel)

Pour des besoins administratifs ou d’archivage, les données doivent pouvoir être **exportées** :

* En format PDF : bien présenté, avec logo, entête, pied de page et couleurs de l’organisation
* En format Excel : avec les bons intitulés de colonnes, une structure claire et éventuellement des totaux

### 1.6 Recherche et filtrage avancé

Les données doivent être facilement accessibles. Pour cela, un **système de recherche multi-critères** sera mis en place :

* Par nom, prénom, email
* Par emploi, grade, région
* Par période (date de début / fin)
* En combinant plusieurs champs

Ces filtres seront disponibles aussi bien dans les tableaux d’affichage que dans les exports.

## 2. Acteurs du système

Le système distinguera plusieurs **rôles utilisateurs** avec des **droits d'accès spécifiques**. Chaque acteur aura un espace dédié et des fonctionnalités ciblées.

**Table 3:acteur du SGRH**

| **Acteur** | **Rôle & Fonctions principales** |
| --- | --- |
| **Administrateur** | Accès complet : gestion des employés, emplois, grades, régions, historiques, exports |
|  |  |
| **Employé** | Accès à ses propres données, lecture seule + possibilité de mettre à jour certaines informations (mot de passe, contact) |
|  | |

Le contrôle d'accès sera assuré par **middleware Laravel** et les politiques d’autorisation intégrées (via Gate ou Policy).

## 3. Diagrammes de cas d’utilisation (UML)

Pour modéliser les différentes interactions entre les acteurs et le système, des **diagrammes de cas d’utilisation UML** seront intégrés au rapport.

### Exemple de cas d’utilisation:

* **Créer un employé** : L’administrateur saisit les informations du nouveau collaborateur via un formulaire sécurisé.
* **Consulter son profil** : L’employé se connecte, accède à sa fiche et à l’historique de ses affectations.
* **Exporter les données** : Le RH génère un fichier PDF/Excel contenant les historiques ou la liste des employés.

Administrateur Employé

*Figure 1 : Diagramme d’UML du système de gestion du personnel*

# ****Chapitre II : CONCEPTION DE LA BASE DE DONNÉES & ARCHITECTURE TECHNIQUE DU PROJET****

# I- ****CONCEPTION DE LA BASE DE DONNÉES****

### **1. Présentation générale**

La base de données constitue le socle central de toute application dynamique. Dans le cadre de ce projet de gestion des ressources humaines (SGRH), la conception de la base de données repose sur une analyse rigoureuse des besoins fonctionnels. Elle vise à modéliser les différentes entités du domaine RH tout en assurant l’intégrité, la cohérence et la performance de l'application.

### **2. Choix du modèle relationnel**

L’approche relationnelle a été adoptée avec l’outil Laravel Eloquent ORM, ce qui permet de définir des relations fortes entre les entités. Chaque table correspond à un modèle Laravel, facilitant les opérations CRUD tout en maintenant la clarté de l’architecture.

### **3. Entités principales**

Les entités retenues dans ce projet sont les suivantes :

* **Users** : contient les informations de base des employés (nom, prénom, email, téléphone, etc.), en plus de leur emploi, grade, région, salaire minimal et salaire maximal.
* **Emplois** : contient les différents types d’emplois occupés par les employés.
* **Grades** : définit les grades associés aux emplois.
* **Régions** : regroupe les différentes zones géographiques d’affectation.
* **Historique employés** : stocke les données d’évolution de carrière d’un employé (emploi précédent, date de début/fin, grade, région…).

**4. Schéma conceptuel de données (MCD)**

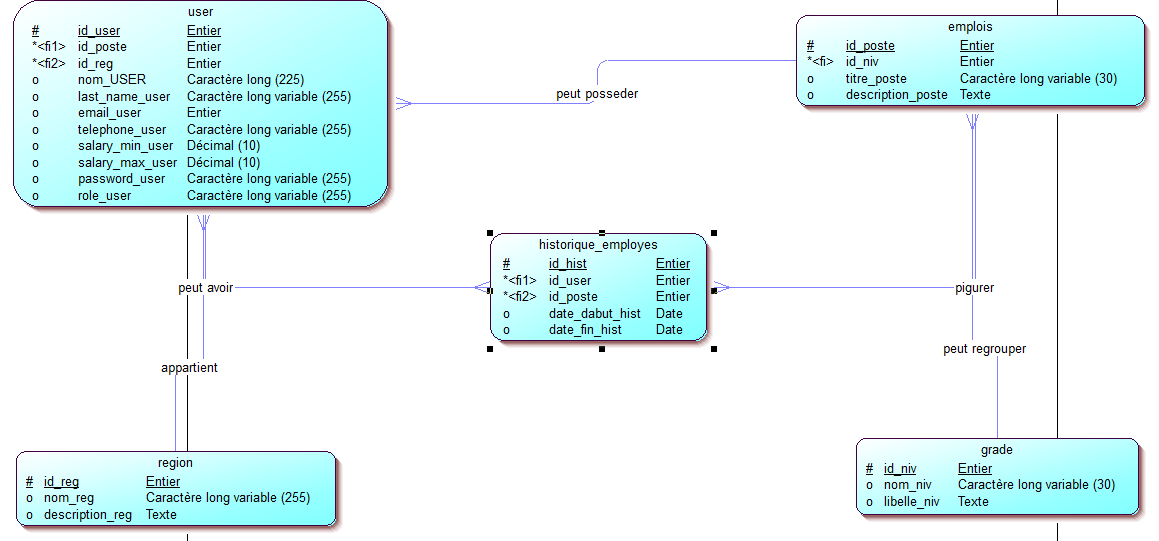
### 

*Figure2: Model Conceptuel de Donnees*

**Exemples de relations:**

* Un employé (**user**) appartient à un **emploi**, un **grade** et une **région**.
* Un **emploi** peut être associé à plusieurs employés.
* Un employé peut avoir plusieurs **entrées d'historique** dans historique\_employés.

### **5. Modèle logique de données (MLD)**

Le MLD est la version relationnelle du MCD. Il définit les clés primaires, étrangères et les types de données pour chaque attribut.

*Figure3:* ***Modèle* *Logique de Données***

### ****6**. Contraintes d'intégrité**

* **Clés primaires** pour chaque table (id).
* **Clés étrangères** pour assurer la cohérence des relations.
* **Contraintes NOT NULL** sur les champs essentiels (nom, emploi, grade…).
* **Contrainte UNIQUE** sur l’email dans users.
* **Contrainte de date** : date\_fin ≥ date\_debut dans l’historique.

### **7. Avantages de cette structure**

* Elle est **scalable** (facile à étendre).
* Elle respecte la **normalisation** jusqu’à la 3ème forme normale (3FN).
* Elle garantit l’**intégrité référentielle** via les clés étrangères.
* Elle permet une **automatisation** des opérations via Eloquent et ses relations (hasMany, belongsTo, etc.).

# II - ARCHITECTURE TECHNIQUE DU PROJET

L’architecture du projet repose sur le framework Laravel, qui suit le paradigme MVC (Modèle-Vue-Contrôleur). Cette structure permet une séparation claire des responsabilités et une meilleure maintenabilité du code.

1. Structure générale du projet Laravel

Le projet est organisé en trois grandes couches :

* **Modèle (Model)** : représente les entités de la base de données (User, Emploi, Grade, Region, HistoriqueEmploye).
* **Vue (View)** : gérée par Blade, avec des composants visuels réalisés en Tailwind CSS.
* **Contrôleur (Controller)** : gère la logique métier et la réponse aux requêtes utilisateurs.

*Figure4 : Arborescence du projet Laravel SGRH dans Visual Studio Code (voir annexe)*

## 2. Technologies et logiciels utilisés

Le projet s’appuie sur plusieurs technologies modernes :

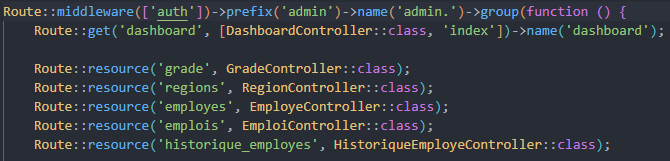
**Table 4:Technologies et logiciels utilisés**

| **Logo** | **Outil / Technologie** | **Rôle et lien officiel** |
| --- | --- | --- |
|  | **Visual Studio Code** | Éditeur de code utilisé pour développer le projet Laravel. 🔗 <https://code.visualstudio.com/> |
|  | **Laragon** | Environnement de développement local (Apache, MySQL, PHP). 🔗 <https://laragon.org/> |
|  | **phpMyAdmin** | Interface graphique pour gérer les bases de données MySQL. 🔗 <https://www.phpmyadmin.net/> |
|  | **Google Chrome** | Navigateur web pour tester l’interface utilisateur. 🔗 <https://www.google.com/chrome/> |
|  | **Laravel 12** | Framework PHP MVC utilisé pour la logique back-end du projet. 🔗 <https://laravel.com/> |
|  | **Tailwind CSS** | Framework CSS utilisé pour un design responsive et moderne. 🔗 <https://tailwindcss.com/> |
|  | **Laravel DomPDF** | Génération de fichiers PDF à partir des vues Blade. 🔗 <https://github.com/barryvdh/laravel-dompdf> |
|  | **Laravel Excel** | Export des données au format Excel (XLSX). 🔗 <https://laravel-excel.com/> |
|  | **GitHub** | Hébergement du code source et gestion des versions. 🔗 <https://github.com/> |

## 3. Composants clés du projet

### a. Les routes

Les routes sont définies dans les fichiers web.php sous forme groupée par middleware auth et préfixes (admin/).



*Figure4 : code web de l’admin*

### b. Les contrôleurs

Ils organisent toute la logique métier : envoi des données aux vues, traitement des formulaires, validation, redirection, etc.



*Figure5 : code contoller des emplois*

### c. Les vues Blade

Les interfaces sont conçues en Blade avec l’aide de Tailwind CSS, en mode clair/sombre. Les vues sont regroupées dans resources/views/admin/.



*Figure6 : code de gestion des employés – Vue index*

### d. Les requêtes personnalisées (Form Request)

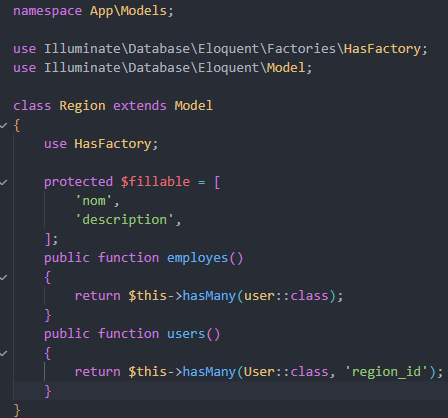
Pour chaque entité, des classes de validation spécifiques sont utilisées, héritées de FormRequest.

Figure : Architecture des Form Requests dans Laravel (voir annexe)

### e. Les modèles Eloquent

Ils contiennent les relations (belongsTo, hasMany), les casts et les scopes de filtrage.





*Figure7 : Capture du code d’un modèle Eloquent typique avec fonctionnalités intégrées*

# Chapitre III - DÉVELOPPEMENT DU PROJET

Cette phase est le cœur du travail réalisé. Elle comprend toutes les étapes allant de l’installation à l’implémentation des différentes fonctionnalités du système.

## 1. Initialisation du projet Laravel

Le projet a été initialisé dans **Laragon**, ouvert dans **VS Code**, avec Laravel installé via Composer. Tailwind CSS a été configuré manuellement.

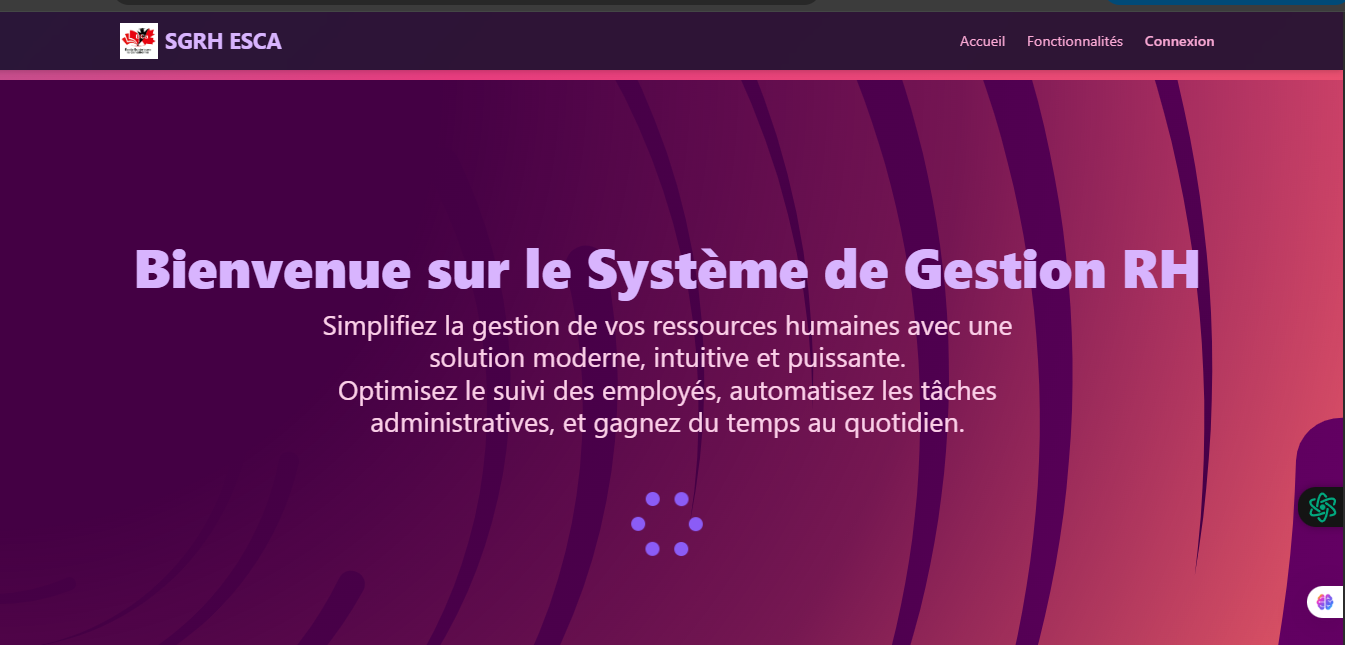
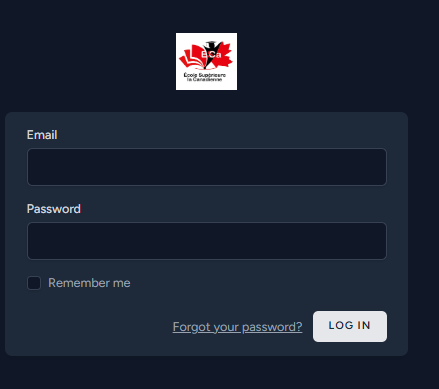


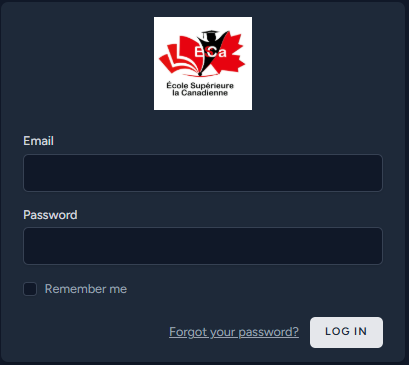
Figure7: page d’acueil du SGRH

## 2. Interface de connexion et tableau de bord

Dès l'installation, Laravel Breeze a été utilisé pour intégrer un système d'authentification.

* La page de connexion permet à l'administrateur ou à un employé de se connecter.
* Une fois connecté, l’utilisateur est redirigé vers un tableau de bord distinct selon son rôle (admin ou employé).

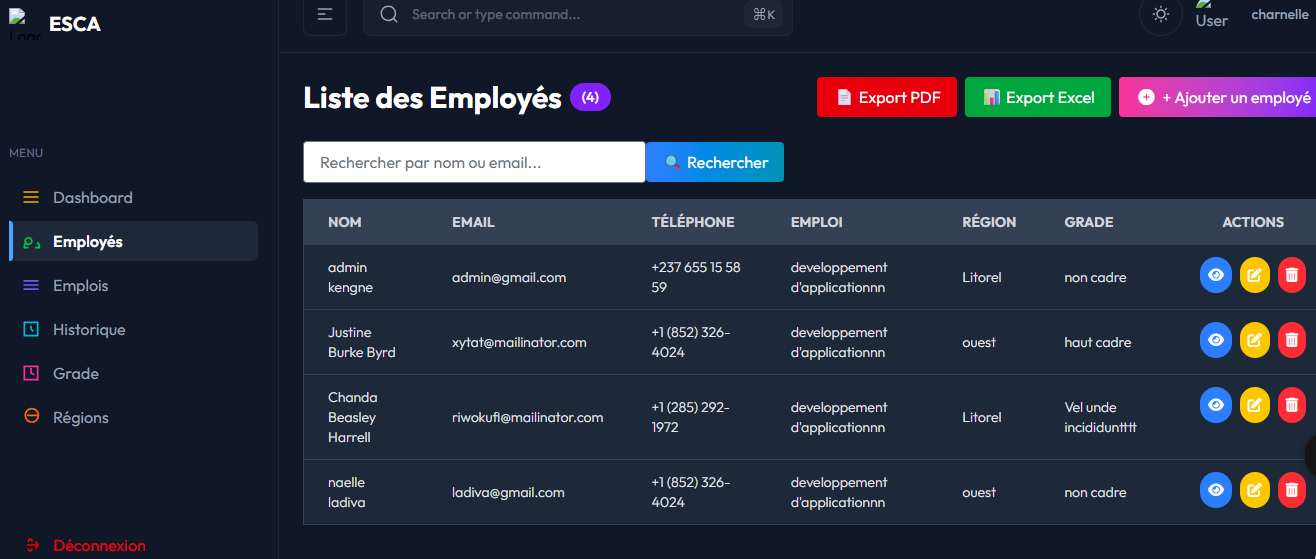




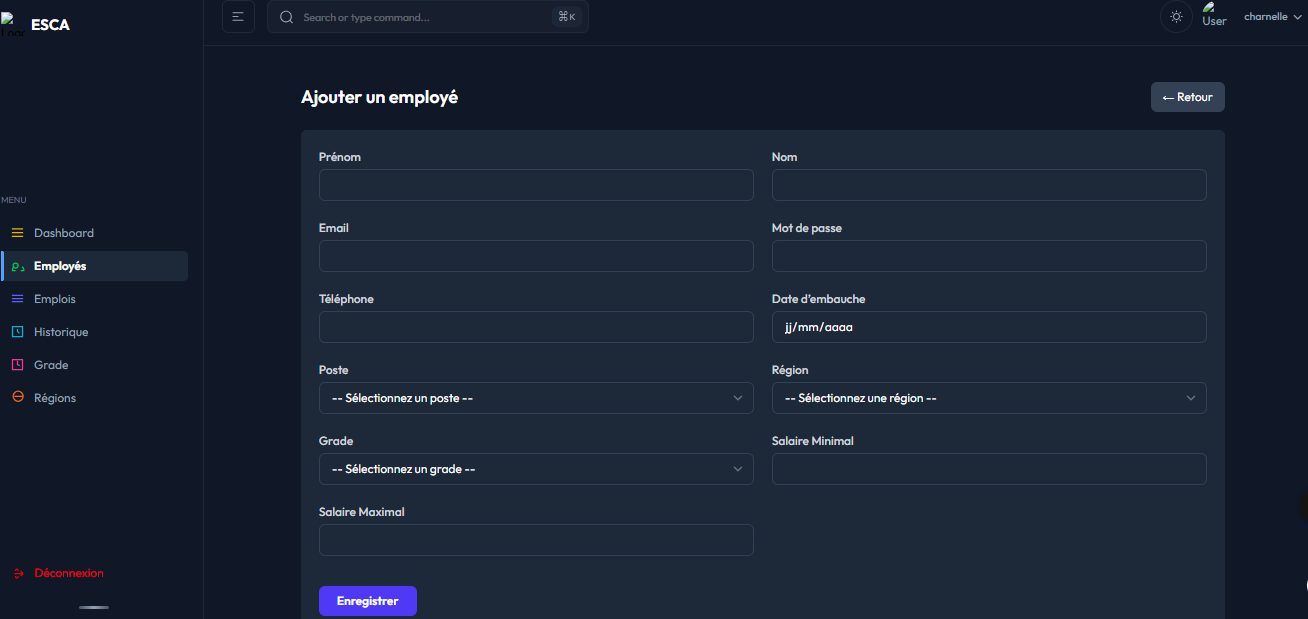
*Figure8 : Interface de connexion*

## 3. Gestion des employés (CRUD complet)

* **Page de liste (index)** : affiche tous les employés sous forme de tableau responsive avec options d’export PDF/Excel.
* **Page d’ajout** : formulaire de création avec champs dynamiques (emploi, région, grade).
* **Page d’édition** : permet la mise à jour des informations d’un employé.



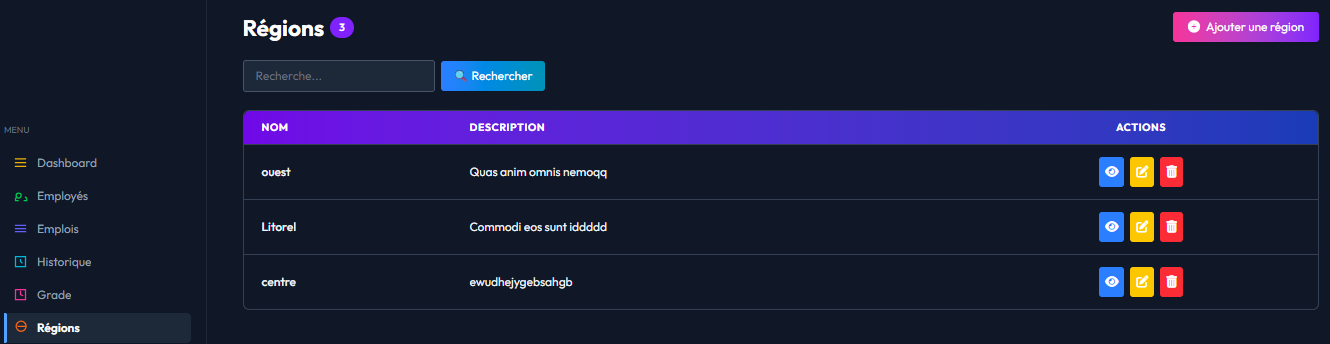
*Figure9 : Interface de gestion des employés (liste des employé)*



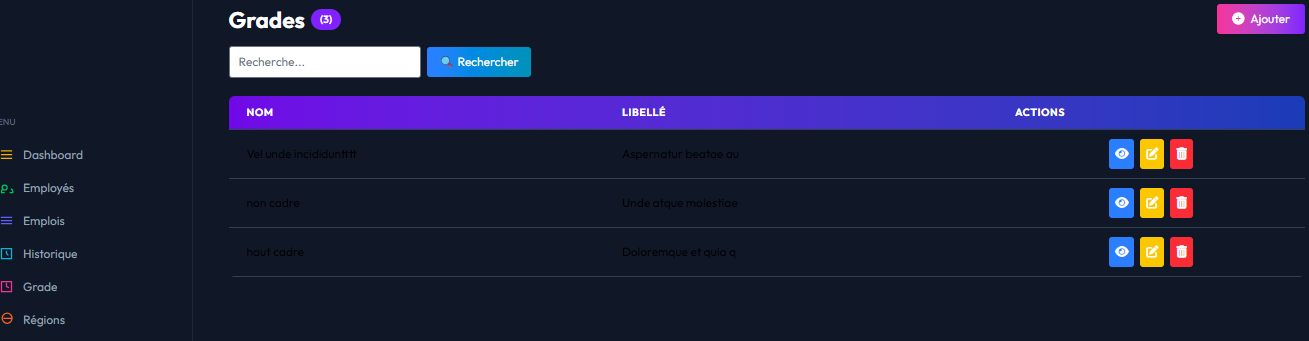
*Figure10 : Interface de gestion des employés (ajout et modification)*

## 4. Gestion des entités : Régions, Grades, Emplois

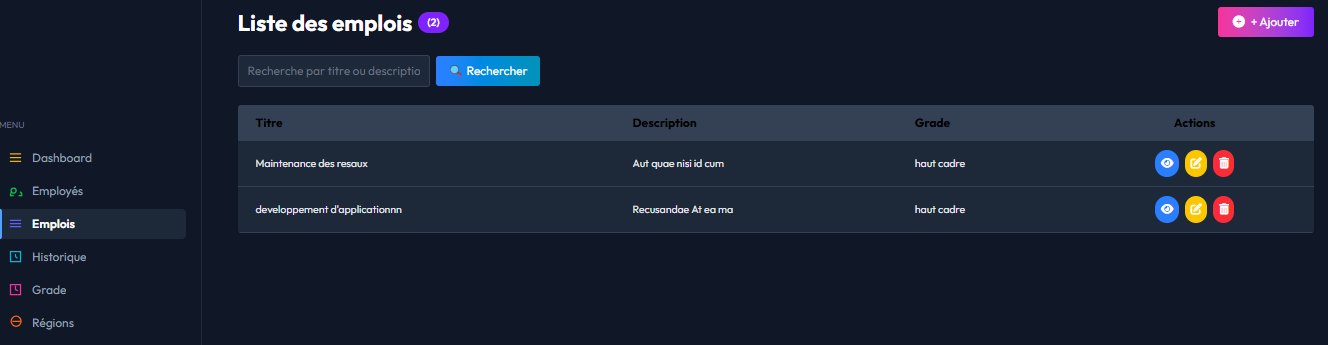
Chaque entité dispose d’un CRUD complet avec les mêmes principes : index, création, modification et suppression.



*Figure11 : Interfaces de gestion des régions (CRUD complet)*



*Figure12 : Interfaces de gestion des grades (CRUD complet)*



*Figure13 : Interfaces de gestion des emplois (CRUD complet)*

## 5. Gestion des historiques des employés

Permet d’enregistrer les changements de poste, de grade, de région d’un employé, tout en conservant une trace complète.

* **Liste des historiques** : avec filtres multicritères
* **Ajout automatique** : les champs sont remplis depuis la fiche employé
* **Affichage détaillé** : informations enrichies et datées

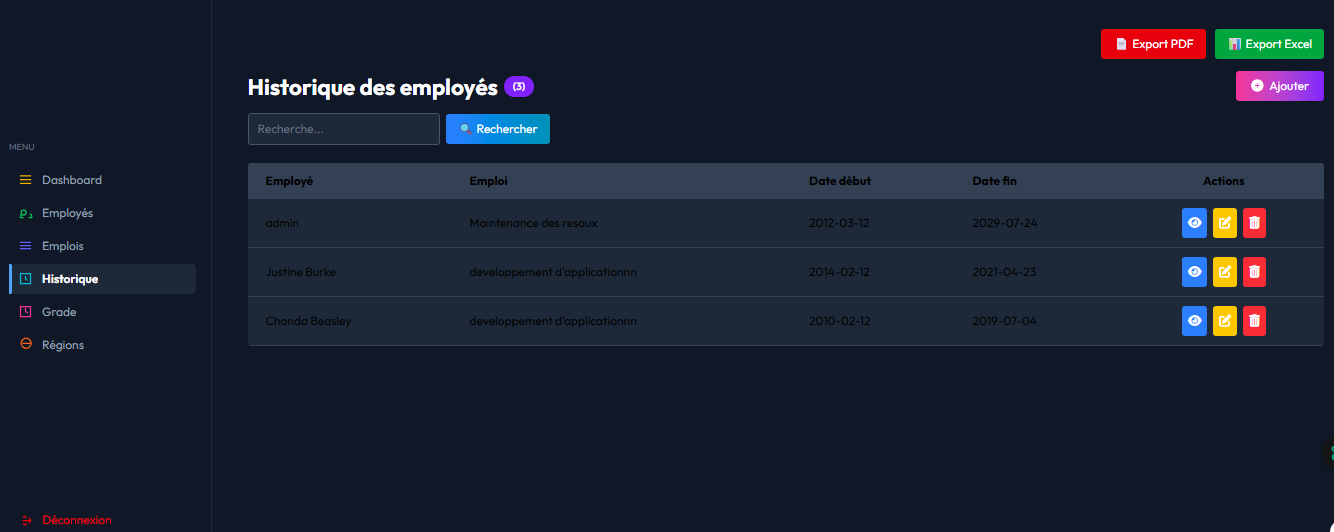
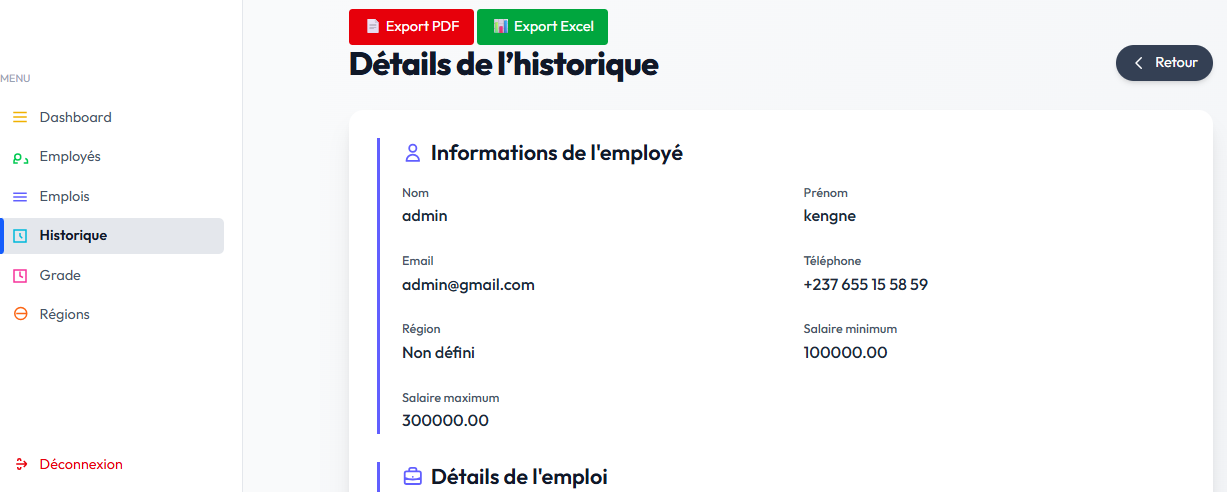
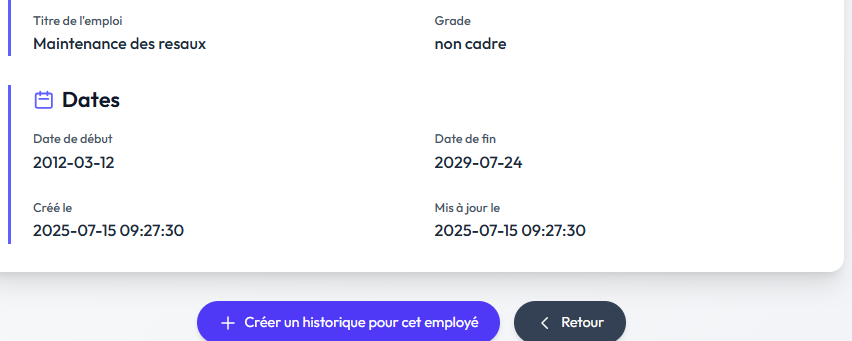


Figure14 : Page d’historique des employés



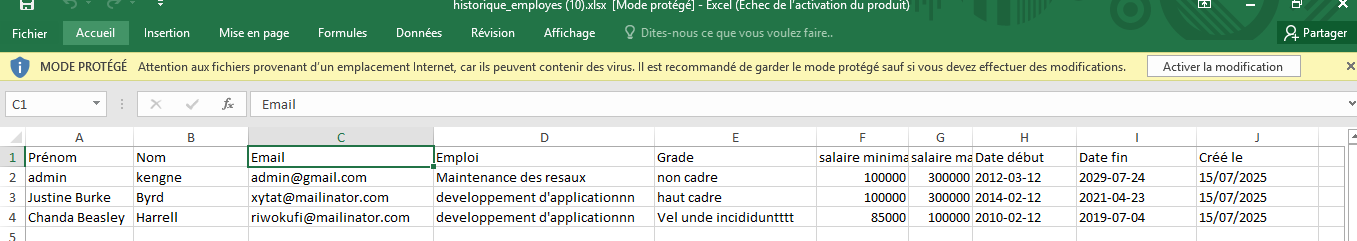


*Figure15 : Détail d’un historique d’employé*

## 6. Exports PDF / Excel

Des boutons d’export sont disponibles sur chaque entité, en particulier l’historique.

* PDF : design soigné avec logo, entête, pied de page, couleurs personnalisées
* Excel : colonnes dynamiques avec noms clairs



*Figure16 : Contenu du fichier Excel exporté des historiques employés*

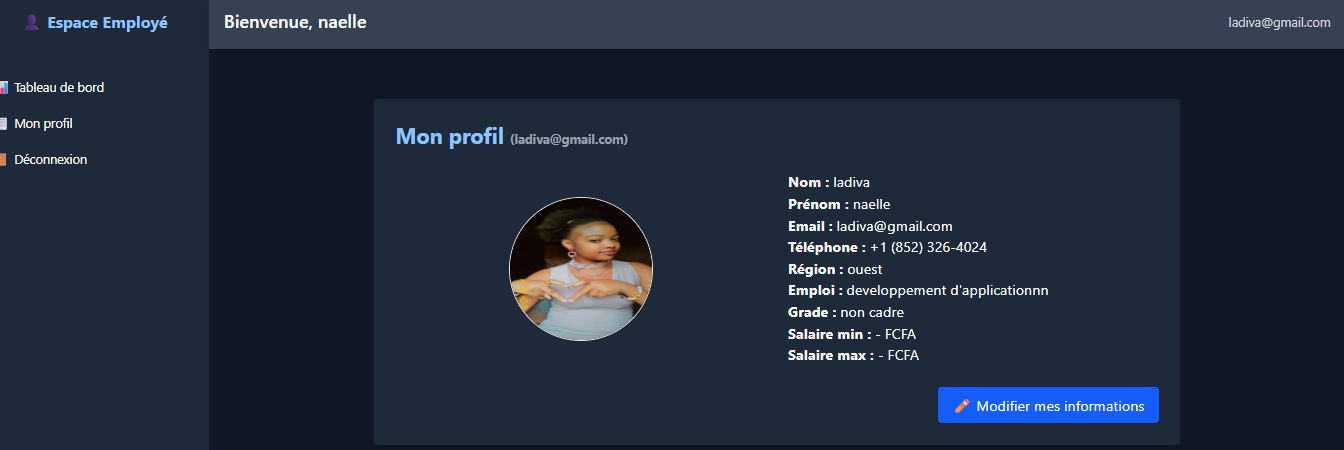


## *Figure17 : Aperçu du fichier PDF généré à partir des historiques employés*

## 7. Espace personnel de l’employé

Chaque employé connecté peut accéder à son profil :

* Informations personnelles (email, emploi, grade, région)
* Historique de ses postes



***Figure18: Interface du profil utilisateur connecté***

VI. DESIGN & EXPÉRIENCE UTILISATEUR (UI/UX)

L’interface utilisateur a été conçue pour être à la fois esthétique, moderne et fonctionnelle. Le but est d’offrir une expérience intuitive aussi bien pour l’administrateur que pour l’employé connecté.

## 1. Choix de la charte graphique

Le design repose sur une palette de couleurs frappantes, agréables à l’œil et adaptées à la lecture prolongée :

* **Mode clair** : équilibre entre bleu profond, violet lumineux, vert frais et rose doux
* **Mode sombre** : violet bleuté, vert doux et gris anthracite

Tailwind CSS a permis de créer un thème responsive adapté automatiquement aux deux modes.

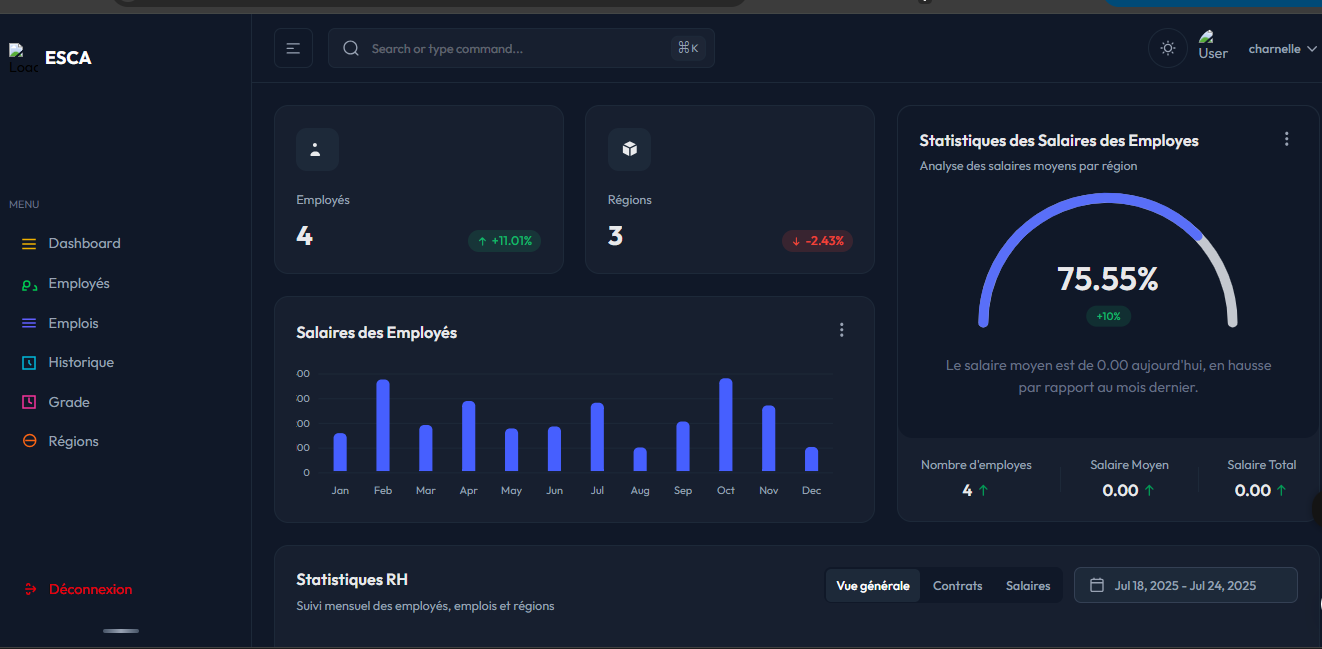
## 2. Typographie et composants visuels

* **Police** : sans-serif moderne pour une lecture claire
* **Boutons** : arrondis, animés au survol (hover), de couleurs différentes selon l’action (vert pour "Ajouter", rouge pour "Supprimer", etc.)
* **Badges** : utilisés pour les statuts, les emplois, ou les niveaux hiérarchiques
* **Icônes** : Font Awesome pour une meilleure reconnaissance visuelle

## 3. Mise en page et navigation

L’interface s’appuie sur une structure en dashboard :

* **Sidebar à gauche** : navigation principale (employés, emplois, historiques, etc.)
* **Header** : logo ESCA RH, lien vers le profil, déconnexion
* **Contenu principal** : tableaux, formulaires, modales, etc.



*📸* ***Figure19: Aperçu de la structure hiérarchique du tableau de bord administrateur***

## 4. Tableaux stylisés et responsive

* Les tableaux utilisent des couleurs alternées, des ombres, et des coins arrondis
* Comportement responsive : s’adapte à l’écran (mobile, tablette, bureau)
* Intégration de la pagination Tailwind et de l’ordre des colonnes

## 5. Accessibilité et performance

* **Dark Mode** : activé automatiquement selon le système
* **Transitions douces** : pour les chargements et interactions
* **Messages clairs** : de succès, d’erreur ou de validation bien visibles

## 6. Valorisation de l’identité visuelle

* Logo ESCA visible en entête des vues PDF
* Couleurs de l’ESCA reprises dans les boutons et badges
* Uniformité graphique dans toutes les pages

Ce travail d’UI/UX permet une navigation fluide et une utilisation conviviale du système, autant pour l’admin que pour l’employé connecté.

## VII. SÉCURITÉ ET VALIDATION

La sécurité de l’application est un enjeu majeur dans la gestion de données sensibles liées aux employés. Le projet intègre plusieurs mécanismes pour assurer la confidentialité, l’intégrité et la fiabilité des informations traitées.

### 1. Authentification et gestion des accès

Le système utilise **Laravel Breeze** pour implémenter l’authentification classique (email/mot de passe) :

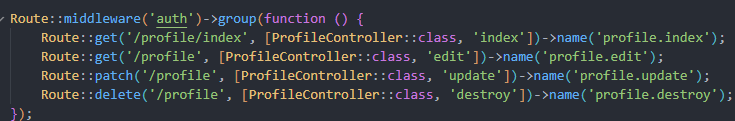
* Un utilisateur doit être connecté pour accéder au système.
* Une redirection est effectuée automatiquement vers la page de login si l’utilisateur n’est pas authentifié.

### 2. Rôles et accès différenciés

Le système gère deux grands types d’utilisateurs :

* **Administrateur (admin)** : a accès à toutes les fonctionnalités (employés, historiques, paramètres…)
* **Employé (user)** : a un accès limité à son propre profil et à son historique

Les contrôleurs sont protégés via des **middlewares** pour vérifier les rôles avant de permettre l’accès à certaines routes.



*Figure20 : Déclaration d’une route protégée par middleware dans web.php*

### 3. Validation des données

Toutes les données saisies via des formulaires sont vérifiées grâce aux **classes FormRequest** :

* Champs requis
* Types de données (email, date, entier…)
* Contraintes supplémentaires (unicité, existence dans une table, date de fin après la date de début…)

Cette validation centralisée permet de garder le contrôleur propre et sécurisé.

### 4. Protection CSRF

Laravel intègre automatiquement la **protection CSRF (Cross-Site Request Forgery)** :

* Chaque formulaire inclut un **token CSRF** généré par Laravel
* Toute tentative de soumission sans ce Toren est rejetée

Cela empêche qu’un utilisateur malveillant ne puisse envoyer des requêtes à l’insu d’un utilisateur authentifié.

### 5. Restrictions sur les requêtes HTTP

Les méthodes critiques (suppression, mise à jour…) sont protégées :

* Suppressions via la méthode HTTP DELETE
* Modifications via PUT ou PATCH
* Méthodes bien définies dans les routes pour éviter toute exécution involontaire

### 6. Sécurité du stockage des mots de passe

Les mots de passe sont **hachés** via l’algorithme bcrypt de Laravel, ce qui garantit qu’aucun mot de passe en clair ne transite ou ne soit stocké dans la base de données.

**7.Gestion du code source avec Git et GitHub**

Dans le cadre du développement de l'application SGRH-ESCA, le suivi du code source a été réalisé à l’aide de **Git**, un système de gestion de versions distribué, et de **GitHub**, une plateforme d’hébergement en ligne du code.

Chaque fonctionnalité du projet (authentification, gestion des employés, export PDF, tableau de bord, etc.) a été développée sur une **branche spécifique**. Cela a permis une meilleure organisation du code, la détection rapide d’erreurs, et une traçabilité complète des modifications grâce à des commits réguliers et explicites.

L’utilisation de GitHub a offert plusieurs avantages :

- 🔁 \*\*Historique des modifications\*\* du code

- 🔒 \*\*Sauvegarde en ligne\*\* sécurisée du projet

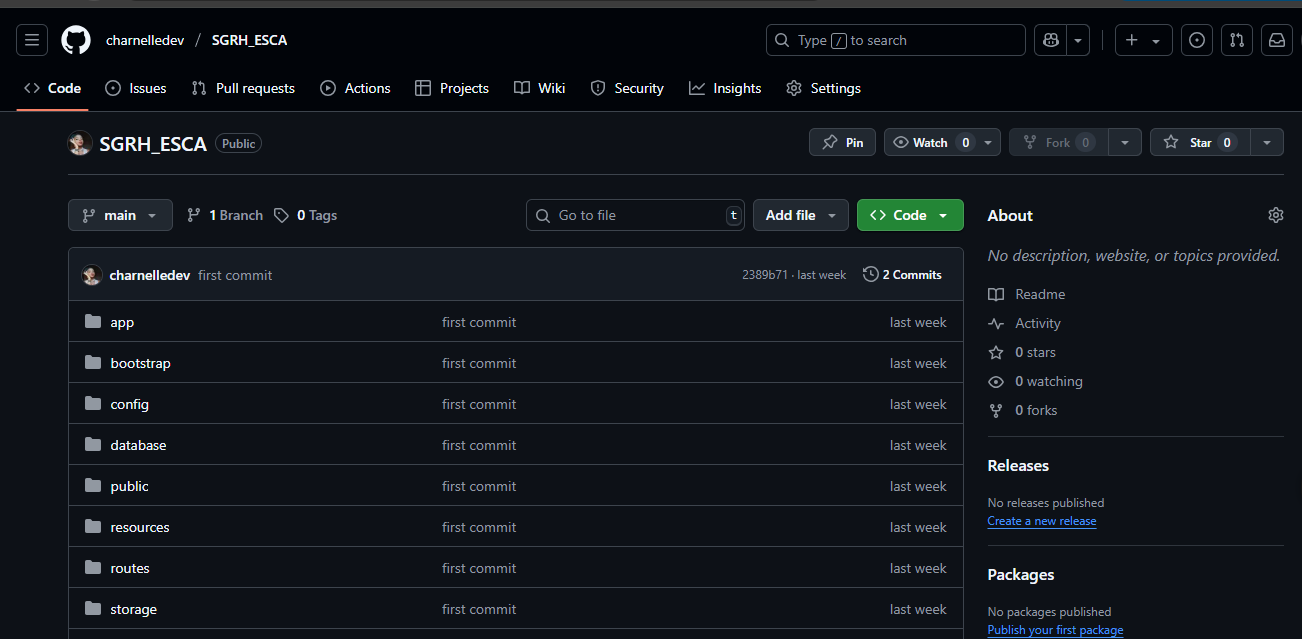
- 📁 \*\*Centralisation des fichiers\*\* pour un accès à tout moment

- 📝 Documentation via le fichier `README.md`

- 📌 Possibilité d’ouvrir des issues en cas de bugs ou suggestions

> 🔗 Lien du dépôt GitHub du projet :

> [https://github.com/charnelledev/SGRH\_ESCA]





*Figure21 : Présentation du dépôt GitHub hébergeant le projet Laravel*

Cette méthode de gestion de version est aujourd’hui indispensable dans tout projet professionnel, car elle facilite la maintenance, le travail en équipe, et garantit une meilleure qualité du produit final.

## VIII. TESTS ET VALIDATIONS

Après le développement des différentes fonctionnalités, des tests manuels ont été réalisés afin de vérifier le bon fonctionnement de l’application. Ces tests couvrent les actions principales de l’administrateur et de l’employé, et permettent de s’assurer de la conformité du système aux exigences définies.

### 1. Tests fonctionnels

Chaque module a été testé à travers différents scénarios :

* **Création d’un employé** : saisie complète des champs, enregistrement correct et affichage dans la liste
* **Modification et suppression** : modification des informations, vérification des validations, suppression confirmée avec message de succès
* **Affichage des historiques** : vérification que les bons historiques sont liés à chaque employé
* **Export PDF/Excel** : téléchargement des fichiers, vérification du contenu, du logo, de l’en-tête, des couleurs et du format
* **Recherche et filtrage** : test de la barre de recherche multi-critères avec ou sans résultat

### 2. Cas d'erreurs gérés

Le système affiche des messages explicites pour les erreurs courantes :

* Tentative d’enregistrement avec des champs manquants
* Dates incohérentes (date de fin avant la date de début)
* Email déjà existant
* Accès non autorisé à certaines pages

### 3. Tests sur différents navigateurs et résolutions

Les interfaces ont été testées sur le navigateur le plus utilisé : Google Chrome ✅

Et sur différents formats d’écran (mobile, tablette, PC) grâce à la structure responsive de Tailwind.

### 4. Retour utilisateur

Des tests utilisateurs ont été réalisés avec quelques personnes en situation réelle :

* L’administrateur pouvait facilement gérer les employés et générer les exports
* Les employés accédaient à leur profil sans difficulté
* Les retours ont confirmé la clarté des messages, la lisibilité de l’interface, et la rapidité du système

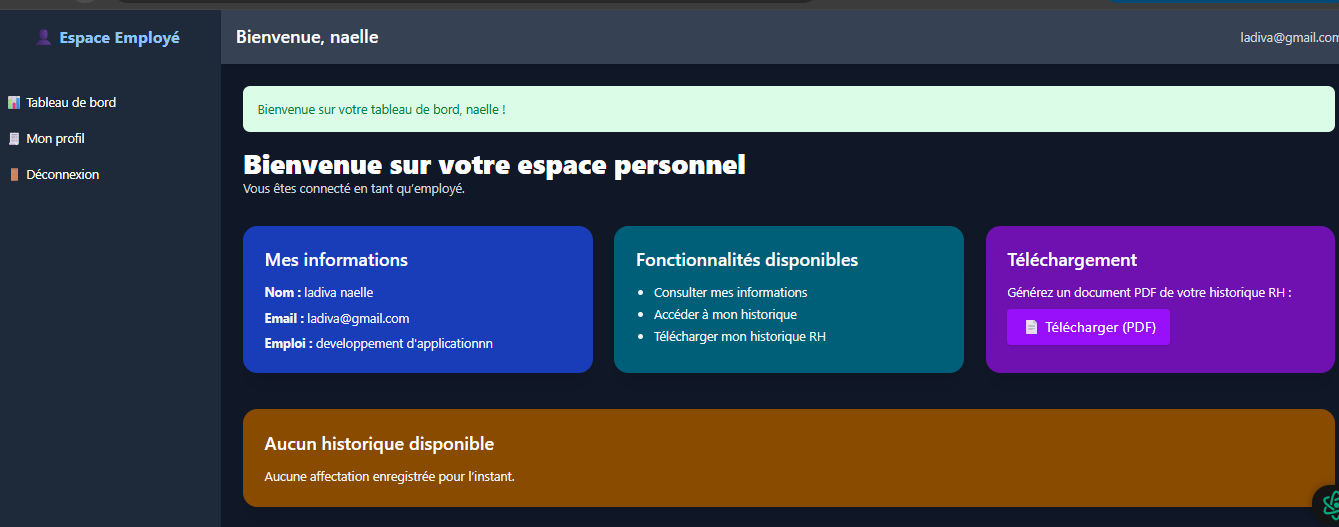


Figure : Tableau de bord de l’utilisateur connecté

# PARTIE IV : ****RECOMMANDATIONS ET SUGGESTIONS****

Le stage réalisé à l’ESCA (École Supérieure La Canadienne) a été une expérience enrichissante tant sur le plan professionnel que technique. Il a permis d’aborder un projet réel dans un contexte concret de digitalisation de la gestion des ressources humaines. Plusieurs aspects du stage ont été particulièrement positifs, tandis que d’autres ont révélé des axes d’amélioration.

### I. ****Remarques positives****

* **Encadrement et autonomie** : Le stagiaire a bénéficié d’une grande autonomie dans le développement du projet, tout en ayant accès aux conseils de l’équipe encadrante en cas de besoin.
* **Pertinence du projet** : Le projet de gestion RH est un cas pratique pertinent, qui permet d’appliquer de nombreuses notions apprises en cours (modélisation, développement web, architecture MVC, sécurité...).
* **Technologies modernes** : L’utilisation de Laravel, Tailwind CSS, Laravel Excel, Laravel DomPDF et Breeze a permis de se familiariser avec un environnement de développement professionnel.
* **Intégration progressive** : Les étapes de développement ont été bien planifiées : création des entités, élaboration des vues, mise en place des exports, tests utilisateurs, etc.

### II. ****Remarques négatives****

* **Manque de personnel encadrant** : À certains moments du stage, l'absence ou l’indisponibilité temporaire de tuteurs techniques a ralenti l’avancement du projet.
* **Connexion internet instable** : Des coupures régulières d’internet ont parfois compromis l’accès à la documentation en ligne ou au téléchargement de dépendances Laravel.
* **Temps limité** : Le délai imparti au projet a nécessité une bonne gestion des priorités, au détriment parfois de certaines fonctionnalités secondaires (comme les notifications ou la gestion multi rôle avancée).

### III. ****Recommendations****

* **Renforcer l’encadrement technique** : Il serait pertinent de prévoir un responsable technique de référence joignable à des horaires fixes, afin de mieux accompagner les stagiaires.
* **Optimiser les outils de travail** : L’accès à une connexion stable et rapide devrait être une priorité, surtout dans le cadre d’un stage impliquant des technologies en ligne (Laravel, npm, GitHub, etc.).
* **Prolonger la durée du stage** : Pour des projets complexes, une durée de 3 mois permettrait de mieux approfondir les fonctionnalités secondaires et d’aborder des notions avancées (tests automatisés, API REST, sécurité avancée…).
* **Intégrer un plan de test utilisateur** : Une phase de retour utilisateurs plus formalisée en fin de stage pourrait permettre de mieux valider l’ergonomie et l’efficacité du système développé.

# CONCLUSION GENERAL

**P**arvenue au terme de cette période de stage, il était question de contribuer activement au sein de l’école Supérieure La Canadienne (ESCA), à travers une immersion dans le monde professionnel et le développement d’un projet concret. Cette expérience a permis non seulement de découvrir l’organisation interne de l’établissement, mais aussi d’en comprendre le fonctionnement, les exigences du métier, ainsi que les défis liés à la gestion quotidienne du personnel. Le stage s’est inscrit dans une dynamique d’apprentissage et de mise en pratique des acquis théoriques, notamment à travers des missions à la fois formatrices et opérationnelles.

Par ailleurs, la réalisation du projet de Système de Gestion des Ressources Humaines (SGRH), développé avec Laravel, a constitué le fil conducteur de cette immersion professionnelle. Le stagiaire a pu mobiliser une pluralité de compétences techniques (modélisation, architecture MVC, développement backend et frontend, sécurisation d’applications web) pour proposer une solution pertinente face à une problématique réelle : l’automatisation des processus RH. Le déroulement du projet a respecté une logique structurée : analyse des besoins, conception des entités, développement, mise en œuvre des exports, tests, et validation fonctionnelle. Le tableau de bord personnalisé, les exports PDF/Excel, le filtrage intelligent et la gestion des utilisateurs (admin/employé) en témoignent.

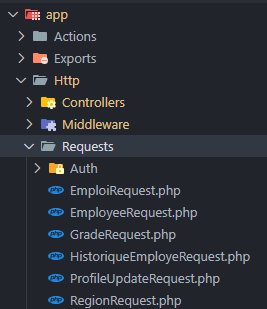
En outre, cette expérience a démontré que la digitalisation des ressources humaines n’est plus une option, mais une nécessité stratégique. L’entreprise, en s’appuyant sur des outils technologiques adaptés, peut gagner en efficacité, traçabilité et réactivité. Ainsi, il serait judicieux d’envisager des pistes d’amélioration comme l’intégration d’une API mobile, la gestion des notifications internes ou encore un système avancé de reporting RH. En somme, ce stage fut un véritable tremplin professionnel, tant sur le plan technique que personnel, et marque une étape clé dans la consolidation du parcours du stagiaire.

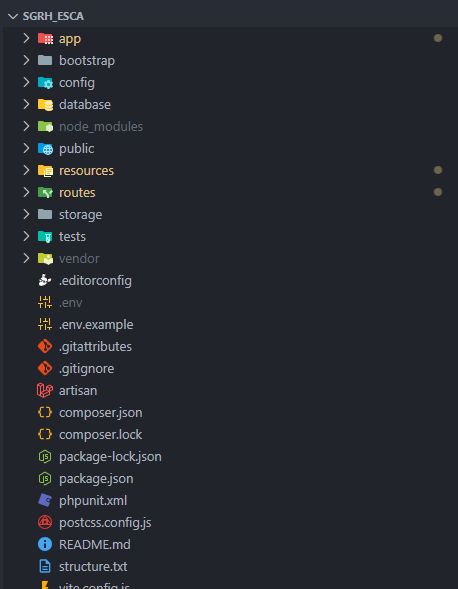
# Bibliographie

Table 5:Bibliographie des ressources externes utilisées

| **Logo** | **Ressource / Bibliothèque** | **Description et lien officiel** |
| --- | --- | --- |
|  | **Tailwind CSS** | Framework CSS utilitaire pour créer des interfaces responsive modernes. 🔗 <https://tailwindcss.com/> |
|  | **Alpine.js** | Librairie JavaScript légère pour la manipulation d’interactions dans le DOM. 🔗 <https://alpinejs.dev/> |
|  | **Font Awesome 6.5** | Bibliothèque d’icônes vectorielles utilisée avec fas fa-... pour embellir les interfaces. 🔗 https://cdnjs.com/libraries/font-awesome |
|  | **Bunny Fonts / Google Fonts** | Import de polices web modernes pour une typographie personnalisée. 🔗 [https://fonts.bunny.net](https://fonts.bunny.net/) |
|  | **Fichier CSS local** | Fichier de style personnalisé pour affiner le design du projet. 🗂️ public/assets/css/style.css |
|  | **YouTube** | Plateforme de tutoriels utilisée pour l’apprentissage et les démonstrations (ex: Laravel, Tailwind, PDF export…). 🔗 <https://www.youtube.com/> |
|  | **Intelligence Artificielle** | Assistance à la rédaction de code, génération d’idées et corrections via ChatGPT. 🔗 <https://chat.openai.com/> |

# Annexe





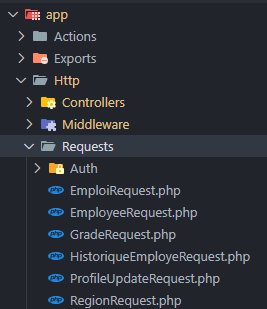


Figure : Architecture des Form Requests dans Laravel

Figure : Arborescence du projet Laravel SGRH dans Visual Studio Code (voir annexe)