

PFE

PROJET DE FIN D'ETUDES

Thème : Conception et développement d'une application de GMAO avec Laravel

Entreprise

Weblinking

Filière

Génie Informatique

Etudiant

Ziad Bouzid

Encadrant

Mme. Sanaa Mouhim

Superviseur

Mr. Aboubaker Chajia

Dédicace

Dédié à tous ceux qui m'ont soutenu tout au long de mon parcours, cette dédicace témoigne de ma gratitude et de ma reconnaissance.

A mes amis et ma famille, pour leur soutien indéfectible et leurs encouragements tout au long de cette expérience. Vos mots d'encouragement, vos oreilles attentives et votre amour inconditionnel ont été des piliers essentiels qui m'ont aidé à persévérer et à donner le meilleur de moi-même.

Enfin, à mes enseignants et formateurs, qui ont contribué à forger ma passion et mes connaissances dans mon domaine d'études. Vos enseignements, conseils et corrections avisées ont été essentiels pour ma croissance personnelle et professionnelle

Remerciements

Je tiens à exprimer mes sincères remerciements à toutes les personnes qui ont contribué à la réussite de mon stage et à mon développement professionnel.

Tout d'abord, nous aimerions exprimer notre gratitude envers notre formateur Mr chajia pour sa guidance précieuse et ses conseils avisés,. Sa vision, son expertise et sa disponibilité ont été des atouts précieux pour mener à bien ce projet dans les meilleures conditions.

Nous souhaitons également remercier notre professeure Mme. Sanaa Mouhim pour leurs conseils éclairés. Leur connaissances approfondies et leur dévouement à notre réussite ont grandement contribué à notre développement professionnel et académique.

Abréviations

- **GMAO** : Gestion de Maintenance Assistée par Ordinateur
- **SEO** : Search engine optimization
- **CMMS** : Computerized maintenance management system
- **API** : application programming interface
- **HTTP** : Hypertext Transfer Protocol
- **MVC** : Modèle Vue Contrôleur
- **CSRF** : Cross-Site Request Forgery
- **MySQL** : My Structured Query Language
- **UML** : Unified Modeling Language
- **URL** : Uniform Resource Locator
- **SQL** : Structured Query Language
- **XSS** : Cross Site Scripting
- **CSS** : Cascading Style Sheets
- **HTML** : HyperText Markup Language
- **PHP** : Hypertext Preprocessor
- **ORM** : Object-Relational Mapping
- **NoSQL** : not only SQL
- **SGBD** : Système de Gestion de Base de Données

Table des matières

RESUME :	11
ABSTRACT :	13
CHAPITRE 1 CONTEXTE DU STAGE	16
1 INTRODUCTION	17
2 PRESENTATION DE L'ENTREPRISE	17
3 Problématique	20
4 Gestion de Projet	21
5 Diagramme de gantt	22
6 Conclusion	23
CHAPITRE 2 ANALYSE	24
1 Introduction	25
2 Analyse des besoins	25
2.1 Les besoins fonctionnels	25
2.2 Les besoins non fonctionnels	26
3 Identification des acteurs	27
4 Diagramme des cas d'utilisation	27
4.1 Cas d'utilisation global	28
4.2 Cas d'utilisation gestion des Utilisateurs	29
4.3 Cas d'utilisation gestion des Equipement	30
4.4 Cas d'utilisation gestion des Piece de rechange	31
4.5 Cas d'utilisation gestion des Ordres de travail	32
5 Diagramme de classe	33

6 MCD.....	34
7 MPD	35
8 Conclusion	36
CHAPITRE3 LITTERATURE/BIBLIOGRAPHIQUE	37
1 Introduction.....	38
2 Les technologies utilisées :.....	38
2.1 Qu'est-ce qu'un Framework ?.....	38
2.2 Front-End Framework :.....	39
2.3 Back-End Framework (PHP) :.....	39
2.4 Pourquoi un Framework ?.....	40
2.5 Analyse des différents Framework :	41
2.6 Choix du Framework et explication de ce choix	43
3 Editeur de code Visual Studio Code	46
3.1 Intégration avec Laravel Extensions :	46
3.2 Prise en charge native de PHP :	46
4 Outils et Technologies de Base de données	47
4.1 PowerAMC	47
4.2 phpMyAdmin	47
5 Définition de la maintenance :	48
6 Conclusion	48
CHAPITRE 4 REALISATION.....	49
1 Introduction	50
2 Architecture globale de l'application.....	50

3 Sécurité côté back-end.....	52
4 Sécurité côté front-end	53
5 Installation et organisation	53
5.1 Installation de Laravel.....	54
5.2 Organisation de Laravel.....	55
6 Artisan	57
7 Les migrations et la configuration de la base	58
7.1 Les migrations	58
7.2 La configuration de la base	58
8 Les interfaces de l'application	65
9 Interface de Tableau de bord	66
10 Interface de gestion des utilisateurs.....	66
11 Interface de gestion d'équipe.....	67
12 Interface de gestion de pièce de Rechange	67
12.1 formulaire pour Ajouter une Pièce de Rechange.....	68
13 Interface de gestion d'équipements.....	68
13.1 le formulaire d'Ajouter un Equipement.....	69
13.2 Supprimer un équipement	69
13.3 Historique des ordre de travail associes à un équipement	70
14 Le formulaire pour ajouter un Ordre de travail	70
15 Interface de gestion des ordres de travail	71
16 Interface de gestion des ordres de travail terminé.....	71
17 Conclusion	72

CONCLUSION GENERAL.....	73
--------------------------------	-----------

BIBLIOGRAPHIE :	74
------------------------------	-----------

Listes des tableaux et des figures

Figure 1 : logo de la société	17
Figure 2 : organigramme de l'organisme	20
Figure 3:Method en cascade.....	21
Figure 4: Diagramme de gantt.....	22
Figure 5: Diagramme de cas d'utilisation global	28
Figure 6: Diagramme de cas d'utilisation gestion des utilisateurs.....	29
Figure 7: Diagramme de cas d'utilisation gestion des équipement.....	30
Figure 8: Diagramme de cas d'utilisation gestion des Piece de rechange ..	31
Figure 9: Diagramme de cas d'utilisation gestion des ordres de travail	32
Figure 10:diagramme de classe.....	33
Figure 11: MCD	34
Figure 12: MPD.....	35
Figure 13: LOGO BOOTSTRAP	43
Figure 14: LOGO Laravel.....	44
Figure 15: LOGO Visual Studio Code.....	46
Figure 16 : Logo PowerAMC	47
Figure 17: LOGO phpMyAdmin	47
Figure 18: Architecture MVC	50
Figure 19 : logo composer.....	54
Figure 20 : Dossier app	55
Figure 21 : fichier .env	59
Figure 22 : Dossier migrations	59
Figure 23: Interface de Tableau de bord	66

Figure 24: Interface gestion des utilisateurs.....	66
Figure 25 : interface gestion d'équipe	67
Figure 26: Interface gestion des Pièces de rechange.....	67
Figure 27: Interface Ajouter une pièce de rechange	68
Figure 28: Interface gestion des équipements.....	68
Figure 29: Interface Ajouter un équipement	69
Figure 30 : interface supprimer un équipement	69
Figure 31 : Interface historique des ordres de travail associes à Cette Equipement	70
Figure 32: Interface Ajouter un Ordre de travail	70
Figure 33: Interface gestion des Ordre de travail.....	71
Figure 34: Interface gestion des Ordre de travail terminé.....	71
 Tableau 1: Fiche technique	 18

Résumé :

Ce rapport présente le projet de conception et développement d'une application de GMAO basée sur le framework Laravel, en utilisant la méthode en cascade. L'objectif principal de ce projet était de créer une application robuste et conviviale pour la gestion efficace des demandes de travail, des équipements, des pièces de rechange et des utilisateurs au sein d'une entreprise.

Dans le cadre de ce projet, nous avons suivi une approche de développement en cascade, qui comprend des étapes distinctes et séquentielles. Ces étapes comprennent l'analyse des besoins, la conception, le développement, les tests et la mise en production.

La fonctionnalité de gestion des demandes de travail permet aux utilisateurs d'afficher, d'ajouter, de modifier et de supprimer des demandes de travail. De plus, le technicien est notifié par email lorsqu'un ordre de travail lui est attribué. Cette fonctionnalité de gestion des demandes de travail permet aussi d'afficher la liste des ordres de travail terminés. Cela permet une traçabilité complète des demandes de travail et une meilleure coordination entre les techniciens et les demandeurs.

La gestion des équipements offre la possibilité d'afficher, d'ajouter, de modifier et de supprimer des équipements. Les détails techniques des équipements sont enregistrés, et la liste des équipements est disponible pour une visualisation facile. Cela facilite la gestion des équipements.

La gestion des pièces de rechange permet d'afficher, d'ajouter, de modifier et de supprimer des pièces de rechange. De plus, la quantité d'une pièce de rechange est décrétementée après la réalisation d'un ordre de travail, et l'administrateur est notifié lorsque la quantité d'une pièce de rechange est inférieure au stock minimum défini. Cela garantit un suivi précis des pièces de rechange et évite les pénuries ou les retards de maintenance.

La gestion des utilisateurs comprend des fonctionnalités d'affichage, d'ajout, de modification et de suppression d'utilisateurs. De plus, il est possible d'attribuer un rôle à un utilisateur, ce qui détermine ses autorisations et ses accès au sein de l'application. Cela permet une gestion centralisée des utilisateurs et une sécurité renforcée.

En conclusion, ce projet de conception et développement d'une application de GMAO avec Laravel, en utilisant la méthode en cascade, a abouti à la création d'une application complète et fonctionnelle pour la gestion des demandes de travail, des équipements, des pièces de rechange et des utilisateurs. Cette application offre des avantages significatifs en termes d'efficacité, de traçabilité, de coordination et de prise de décisions pour les processus de maintenance au sein de l'entreprise.

Abstract :

This report presents the design and development project of a CMMS application based on the Laravel framework, using the waterfall method. The main objective of this project was to create a robust and user-friendly application for the efficient management of work requests, equipment, spare parts and users within a company.

As part of this project, we followed a waterfall development approach, which includes distinct and sequential steps. These stages include needs analysis, design, development, testing and production.

The work request management feature allows users to view, add, edit, and delete work requests. In addition, the technician is notified by email when a work order is assigned to him. This work request management feature also allows you to display the list of completed work orders. This allows full traceability of work requests and better coordination between technicians and requesters.

Equipment management offers the ability to view, add, modify and delete equipment. Technical details of equipment are recorded, and the list of equipment is available for easy viewing. This facilitates the management of equipment.

Spare parts management allows you to view, add, modify and delete spare part. In addition, the quantity of a spare part is decremented after the completion of a work order, and the administrator is notified when the quantity of a spare part is less than the defined minimum stock. This ensures accurate tracking of spare parts and avoids shortages or maintenance delays.

User management includes features for viewing, adding, editing, and deleting users. In addition, it is possible to assign a role to a user, which determines his permissions and access within the application. This allows for centralized user management and enhanced security.

In conclusion, this design and development project of a CMMS application with Laravel, using the waterfall method, resulted in the creation of a complete and functional application for the management of work requests, equipment, parts spares and users. This

application offers significant advantages in terms of efficiency, traceability, coordination and decision-making for maintenance processes within the company.

Introduction Générale

La maintenance des équipements et des installations est un élément crucial pour assurer le bon fonctionnement et la pérennité des entreprises. Dans ce contexte, la Gestion de la Maintenance Assistée par Ordinateur(GMAO)joue un rôle essentiel en facilitant la planification, le suivi et la gestion des activités de maintenance. Afin de répondre aux besoins croissants de gestion de la maintenance, nous avons entrepris le développement d'une application web GMAO basée sur le framework Laravel.

Notre projet de fin d'études vise à concevoir et à mettre en œuvre une application web robuste qui permettra aux entreprises de gérer efficacement leurs équipements, leurs

pièces de rechange et leurs demandes d'intervention. En utilisant Laravel, un framework PHP populaire et puissant, nous avons choisi une approche moderne pour développer cette application, en tirant partie de ses fonctionnalités avancées et de sa flexibilité.

Le plan du rapport de stage comprendra le chapitre 1 dans lequel on traitera la partie du contexte de stage , le chapitre 2 dans lequel on traitera la partie de l'Analyse. Le chapitre 3 traitera la partie littérature et bibliographique puis le chapitre 4 qui va

couvrir la partie réalisation .

Chapitre 1 Contexte du stage

1 INTRODUCTION

La gestion efficace de la maintenance des équipements et des installations est un défi majeur pour de nombreuses entreprises de divers secteurs. La Gestion de la Maintenance Assistée par Ordinateur (GMAO) Offre une solution technologique permettant d'optimiser la gestion des activités de maintenance, en facilitant la planification, le suivi et la gestion des équipements, des pièces de rechange et des demandes d'intervention. Dans ce contexte, notre projet de fin d'études vise à concevoir et à mettre en œuvre une application web de GMAO basée sur le framework Laravel .

2 PRESENTATION DE L'ENTREPRISE



Figure 1 : logo de la société

Weblinking est une société créée en 2019 spécialisée dans le domaine de création des sites web et SEO ainsi que le e-marketing , son directeur Aboubaker CHAJIA a créé ses propres sites dans le but de monétiser le trafic mais aussi pour les faire tester sur le moteur de recherche Google et voir comment les algorithmes de Google fonctionnent.

Ces tests lui ont permis de mieux connaître le poids de chaque variable dans les algorithmes de Google et de maîtriser les outils du référencement naturel.

RAISON SOCIALE	Weblinking Sarl
SIEGE SOCIAL	Avenue Al Marinienne rue 3 n°28 Hay Al Massira 4 – 73000 Dakhla
FORME JURIDIQUE	S.A.R.L
DATE DE CREATION	01 Avril 2019
SECTEUR / ACTIVITES	Secteur primaire Activités : <ul style="list-style-type: none"> • Développement web • Gestion et maintenance informatique
EMAIL	contact@weblinking.ma
DIRECTEUR	M. Aboubaker Chajia
EFFECTIF DU PERSONNEL	10

Tableau 1: Fiche technique

Pour de nombreuses agences web, le référencement naturel est un service parmi bien d'autres. Mais pour l'agence Weblinking, le référencement naturel est une spécialité maîtrisée et une expertise reconnue depuis plusieurs années. Le SEO est au cœur de leurs métiers et ils assurent à leurs clients, petits et grands comptes, d'excellents résultats et un retour sur investissement.

C'est ce qui fait de Weblinking l'agence idéale pour sous-traiter les clients !

Le développement web est l'activité principale de la société Weblinking. L'équipe crée, développe et référence des sites des clients selon les cahiers des charges exigés.

L'équipe intervient notamment dans :

- Développement des logiciels.
- Référencement naturel (positionnement sur Google).
- Création des sites web vitrine.
- Création des sites e-commerce.
- Conception graphique (Logo, brochure, bannière...).

Les techniciens et ingénieurs de Weblinking, hautement qualifiés, interviennent sur l'ensemble de la chaîne numérique. Ainsi, ils se chargent de l'installation et de la configuration de tous éléments numériques (ordinateur, imprimante, appareil photo, scanner, webcam, dvd, tous éléments haute- définition, bureautique, etc.), de tous éléments de communication (téléphone fixe et portable, ADSL, Wifi, etc.) ou encore d'applications dédiées (anti-virus, logiciels métier, logiciels de PAO – Publication Assistée par Ordinateur, etc.).

De plus, les techniciens aident le client à faire ses premiers pas et à découvrir les principales fonctionnalités des outils installés et configurés en accompagnant le client lors d'une première utilisation.

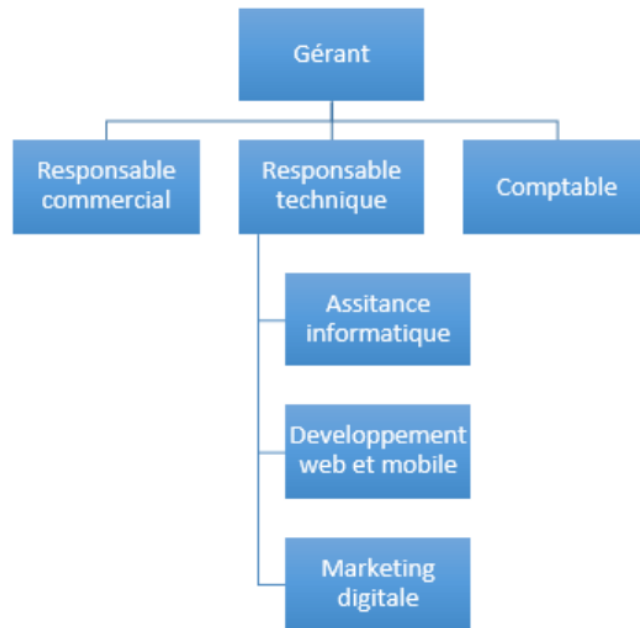


Figure 2 : organigramme de l'organisme

Des raisons pour lesquelles Weblinking est l'une des agences SEO les plus efficaces : **La sécurité & la maintenance :** Protection les business contre le piratage et les pénalités algorithmiques

Support : L'équipe propose un support 24/24 & 7/7

Qualité : La qualité est leurs premières priorités

Optimisation : Grâce au techniques d'optimisation adoptés par Weblinking les sites seront séduit par les moteurs de recherche comme Google, Bing et Yahoo.

3 Problématique

La maintenance des équipements et des installations est un aspect essentiel de toute entreprise pour garantir la continuité des opérations et la satisfaction des clients. Cependant, de nombreuses organisations se retrouvent confrontées à des défis lorsqu'il s'agit de gérer efficacement la maintenance de leurs équipements.

Des problèmes tels que la difficulté à suivre les informations sur les équipements, à gérer les pièces de rechange et à traiter les demandes d'intervention de manière efficace peuvent entrainer des temps d'arrêt prolongés.

C'est dans ce contexte que notre projet de fin d'études intervient. Nous avons

identifié la nécessité de développer une application web de GMAO qui permettra aux entreprises de surmonter ces défis en fournissant une solution centralisée pour la gestion de la maintenance des équipement. L'objectif est de développer une application conviviale et puissante qui permettra aux utilisateurs de gérer facilement leurs équipements, leurs pièces derechange et leurs demandes d'intervention, tout en améliorant l'efficacité globale de la maintenance.

4 Gestion de Projet

Nous avons utilisé la méthodologie de gestion de projet en cascade, qui consiste en une série de phases distinctes, chacune étant réalisée séquentiellement. Ces phases comprennent la définition des besoins, l'analyse des exigences, La conception, le développement, les tests, le déploiement et la maintenance.

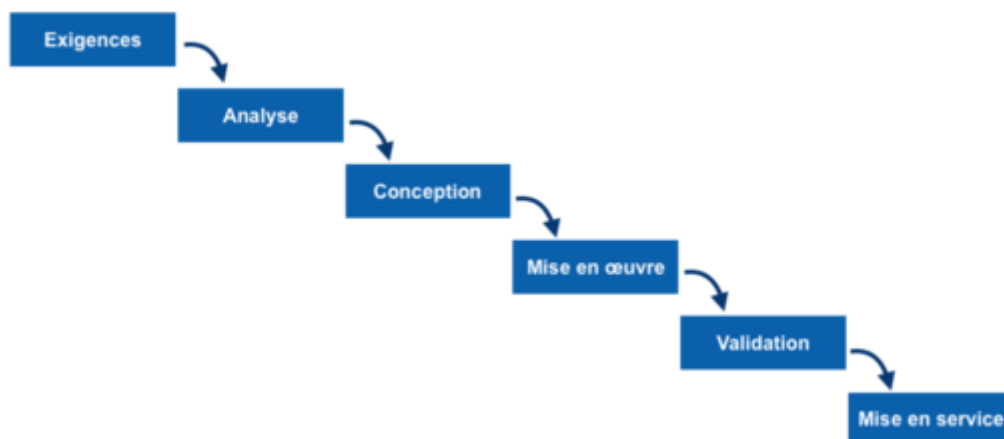


Figure 3:Method en cascade

5 Diagramme de gantt

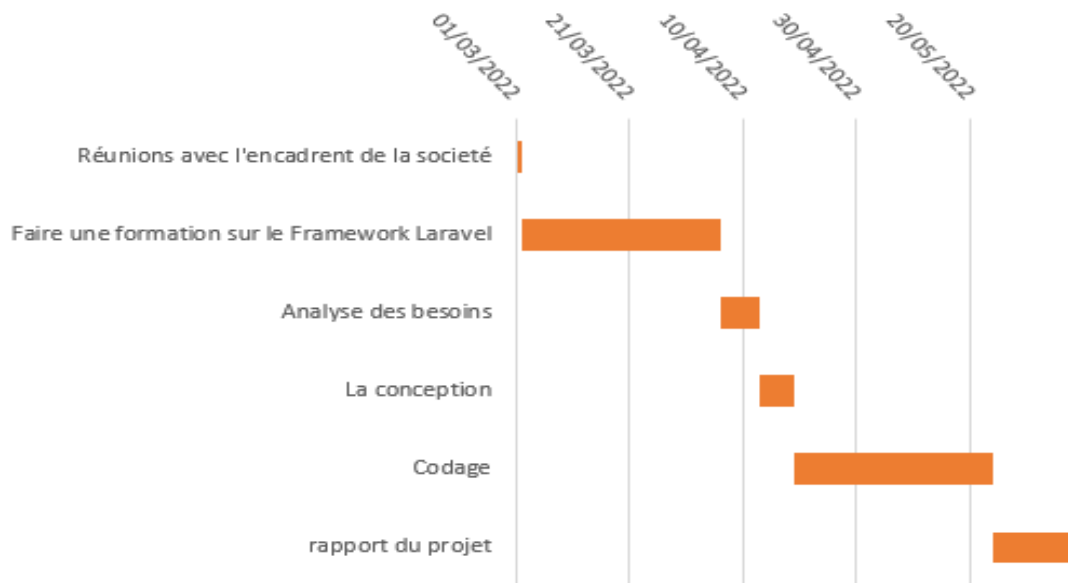


Figure 4: Diagramme de gantt

Le diagramme de Gant est un outil de gestion de projet largement utilisé pour planifier, organiser et visualiser les différentes tâches et activités d'un projet sur une période donnée. Il tire son nom de son créateur, Henry Gantt, un Ingénieur américain du début du XXe siècle. En utilisant un diagramme de Gantt, les chefs de projet et les équipes peuvent mieux comprendre l'ordonnancement des tâches, identifier les éventuels problèmes de délai ou de ressources, et prendre des mesures correctives en conséquence. Il est également utile pour communiquer efficacement l'avancement du projet à toute les parties prenantes concernées.

6 Conclusion

En conclusion, ce chapitre introductif a présenté la problématique de la gestion de maintenance des équipements et a mis en évidence la nécessité de développer une application web de GMAO. De plus, nous avons abordé la gestion de projet adoptée pour assurer la réalisation efficace de ce projet de fin d'études. Dans le chapitre suivant, nous explorerons plus en détail l'étapes de l'analyse.

Chapitre 2 Analyse

1 Introduction

Dans ce chapitre on va voir la phase d'analyse qui est essentielle dans le développement des logiciels. Cette phase permet de clarifier les besoins et les objectifs du projet, de définir les besoins fonctionnels et non fonctionnels du logiciel, de concevoir l'architecture du système, d'identifier les risques et les contraintes, et de planifier la mise en œuvre.

Premièrement nous commencerons par définir les besoins fonctionnels et non fonctionnels, et ensuite présenter le diagramme général de cas d'utilisation, Le diagramme de classe.

2 Analyse des besoins

2.1 Les besoins fonctionnels

Il s'agit des exigences requises pour qu'un système puisse remplir les fonctions pour lesquelles il a été conçu. Les besoins fonctionnels sont généralement exprimés en termes de tâches à accomplir, d'actions à effectuer ou de fonctionnalités à fournir.

L'application qu'on va concevoir doit permettre à l'utilisateur d'effectuer les actions suivantes :

➤ **S'authentifier :**

- A travers un nom d'utilisateur et un mot de passe.

➤ **Gérer les Utilisateurs :**

- Afficher, Ajouter, Modifier et supprimer les Utilisateurs
- Attribuer un Rôle à un Utilisateur

➤ **Gérer les Equipe :**

- Afficher, Ajouter, Modifier et supprimer une Equipe
- Attribuer un technicien a une Equipe

➤ **Gérer les Equipement :**

- Afficher, Ajouter, Modifier et supprimer les équipements
- Afficher la liste des équipements

➤ Gérer les pièces de rechange :

- Afficher, Ajouter, Modifier et supprimer les pièces de rechange
- Décrémenter la quantité d'une pièce de rechange après la réalisation d'un ordre de travail
- Notifier l'administrateur lorsque la quantité d'une pièce de rechange est inférieure au stock min .

➤ Gérer les ordres de travail :

- Afficher, Ajouter, Modifier et supprimer les ordres de travail
- Notifier le technicien lorsqu'un ordre de travail lui a été attribué
- Afficher la liste des ordres de travail terminés

2.2 Les besoins non fonctionnels

Les Besoins non fonctionnels, sont des besoins liés aux caractéristiques et aux propriétés d'un système qui ne sont pas directement liées à son fonctionnement ou à sa capacité à effectuer une fonction spécifique.

Les besoins non fonctionnels sont souvent des besoins de qualité qui définissent les caractéristiques du système. En termes de sécurité, de performance, de disponibilité, de scalabilité etc.

- **La sécurité** : L'application doit être sécurisée. C'est-à-dire que l'application doit être capable de garantir que les données sensibles ne sont accessibles que par les utilisateurs autorisés et qu'elles sont protégées contre les attaques malveillantes.
- **La performance** : Les besoins de performance sont liés à la rapidité et à la réactivité du système. C'est-à-dire que l'application doit réagir dans un délai précis, quel que soit l'action de l'utilisateur.
- **La disponibilité** : Les besoins de disponibilité concernent la capacité du système à être disponible et opérationnel pendant des périodes prolongées.

- **La scalabilité** : L'application doit être capable de s'étendre pour supporter de plus en plus d'utilisateurs sans affecter les performances.

3 Identification des acteurs

- L'application va avoir trois acteurs :

- **Administrateur** : l'administrateur aura le droit de créer ou supprimer un Utilisateur et de définir les rôles des utilisateurs du système. L'administrateur va aussi avoir un accès complet, y compris la modification l'ajout et la suppression des Demande d'intervention. L'administrateur pourra aussi gérer les pièce de rechange et les équipement.
- **Technicien** : Le technicien aura le droit de créer, modifier et supprimer les équipement et les Demandes d'intervention et pourra aussi changer le statut des demande d'intervention.
- **employé** : l'employé aura le droit uniquement de soumettre des ordres D'intervention et voir leur statut.

4 Diagramme des cas d'utilisation

Le diagramme de cas d'utilisation est un type de diagramme UML (Unified Modeling Language) qui permet de décrire les interactions entre les utilisateurs (ou acteurs) et un système informatique. Il représente graphiquement les différents cas d'utilisation d'un système en identifiant les acteurs Impliqués, les actions ou tâches effectuées et les interactions entre eux.

4.1 Cas d'utilisation global

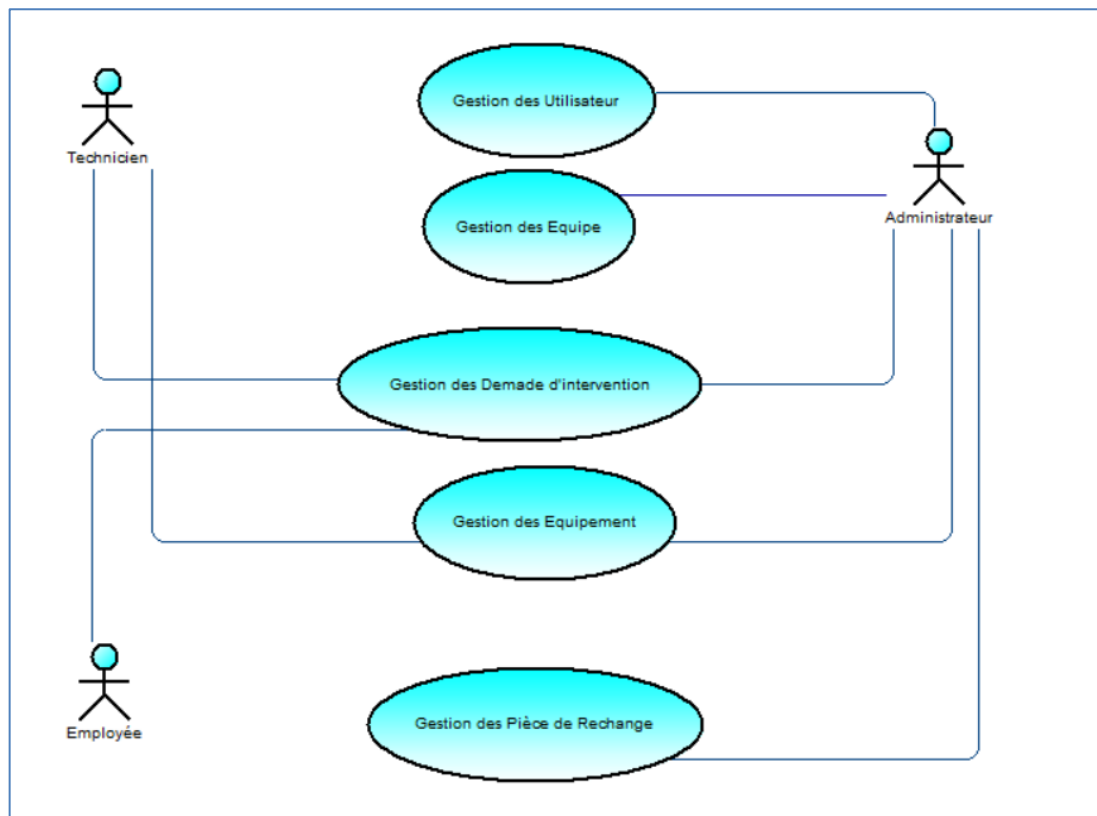


Figure 5: Diagramme de cas d'utilisation global

La figure 5 présente le diagramme des cas d'utilisation global de l'application étudié

4.2 Cas d'utilisation gestion des Utilisateurs

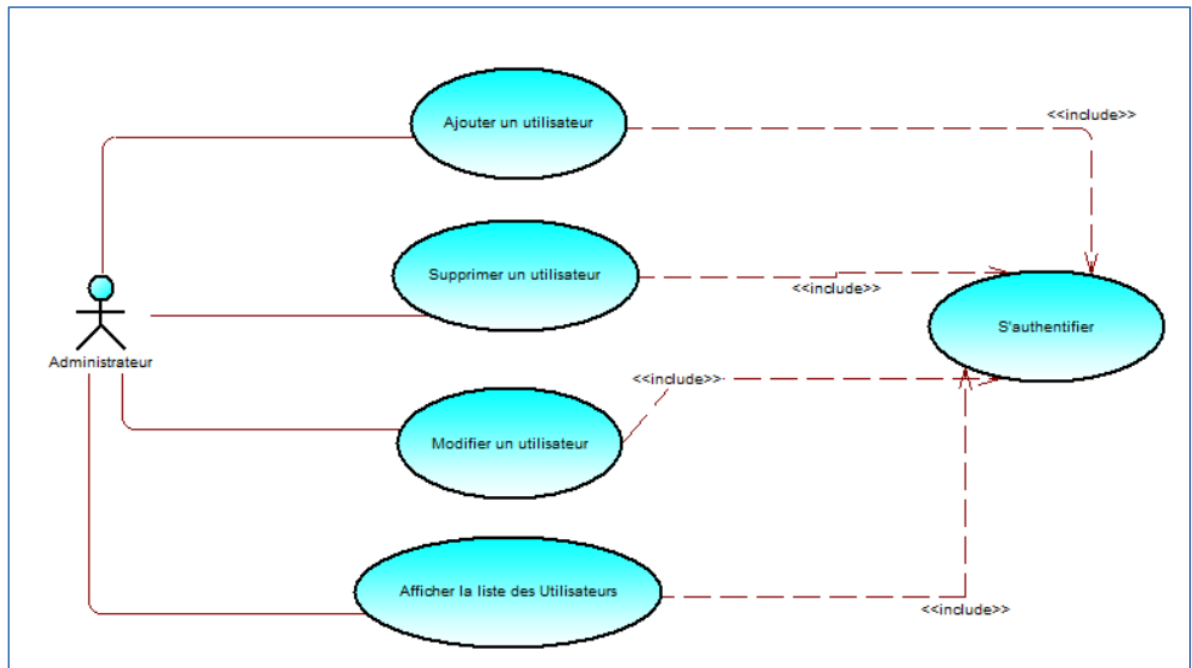


Figure 6: Diagramme de cas d'utilisation gestion des utilisateurs

La figure 6 présente le diagramme des cas d'utilisation de gestion des utilisateurs

- S'authentifier : Permet à l'administrateur de s'authentifier avant d'accéder à l'application.
- Gérer les utilisateur : Permet à l'administrateur d'ajouter , Supprimer ,Modifier les utilisateurs et Afficher la liste des utilisateurs

4.3 Cas d'utilisation gestion des Equipement

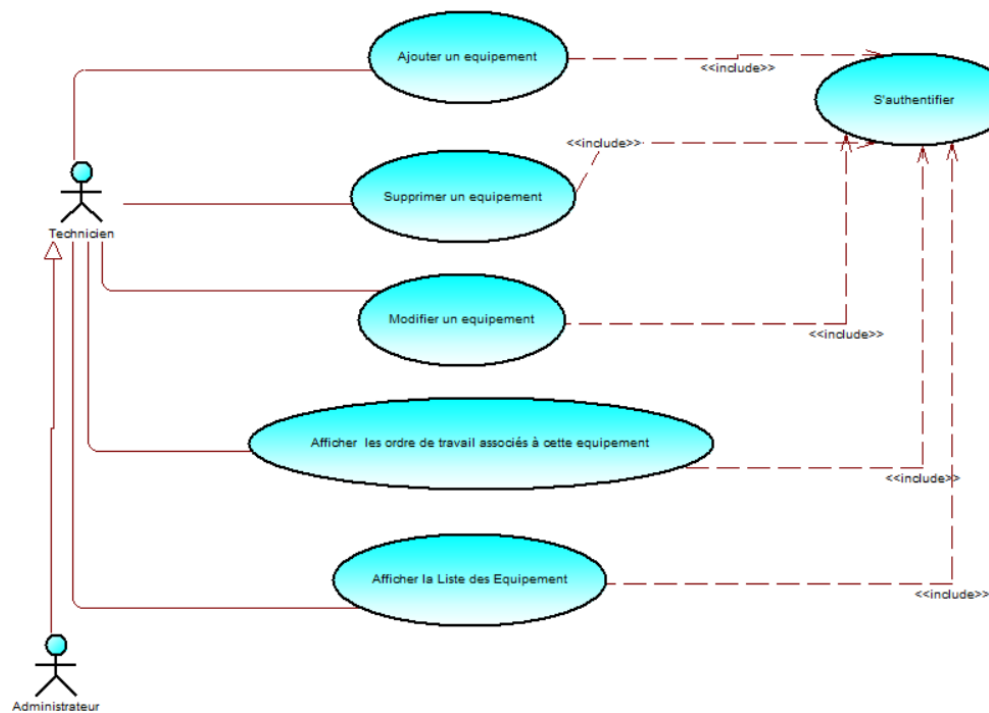


Figure 7: Diagramme de cas d'utilisation gestion des équipement

La figure 7 présente le diagramme des cas d'utilisation de gestion des équipements

- Gérer les équipement : Permet à l'utilisateur avec le rôle administrateur et technicien d'ajouter , Supprimer ,Modifier les équipement et Afficher la liste des équipements et aussi afficher les ordre de travail associées à cette équipement

4.4 Cas d'utilisation gestion des Piece de rechange

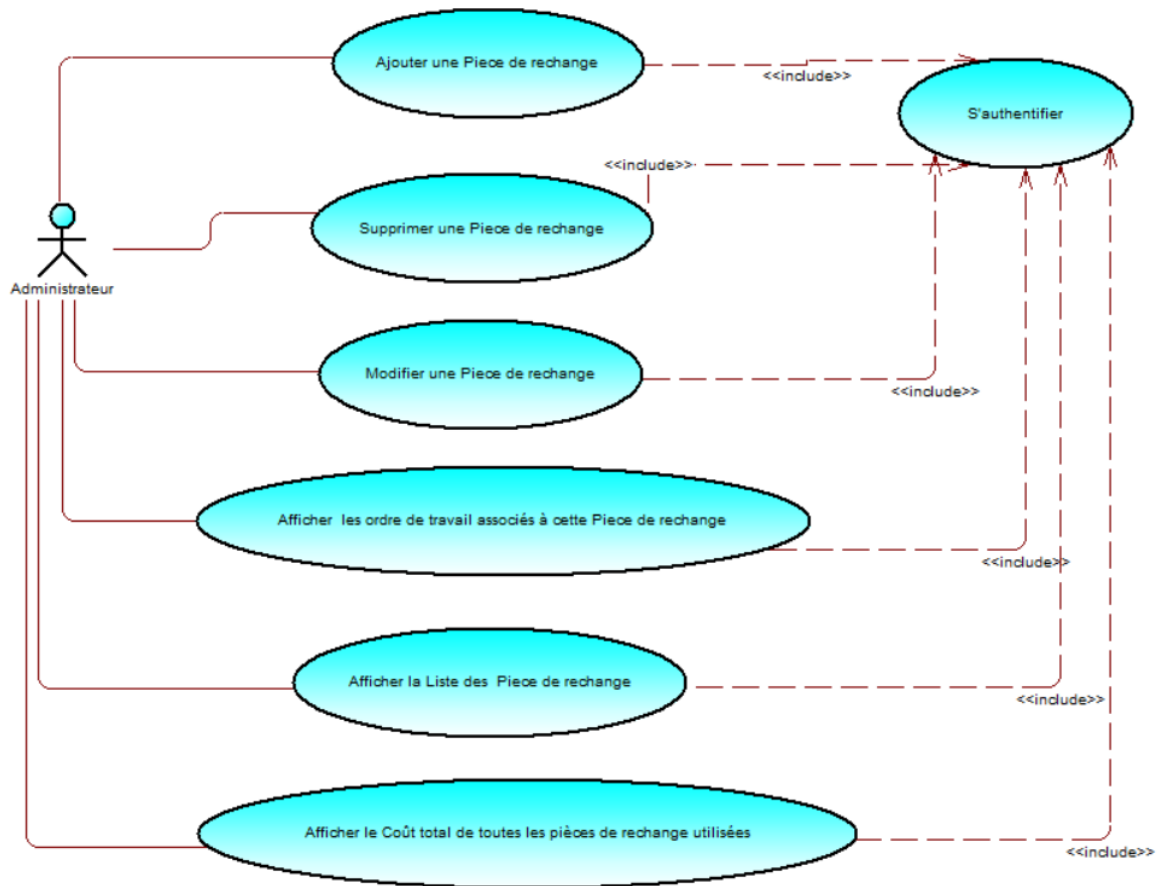


Figure 8: Diagramme de cas d'utilisation gestion des Piece de rechange

La figure 8 présente le diagramme des cas d'utilisation de gestion des pièces de rechange

- Gérer les pièce de rechange : Permet à l'administrateur d'ajouter , Supprimer Modifier les utilisateurs et Afficher la liste des pièces de rechange
- Afficher le coût total de toutes les pièces de rechange utilisées dans la maintenance d'un équipement.
- Afficher les ordres de travail associé à une pièce de rechange

4.5 Cas d'utilisation gestion des Ordres de travail

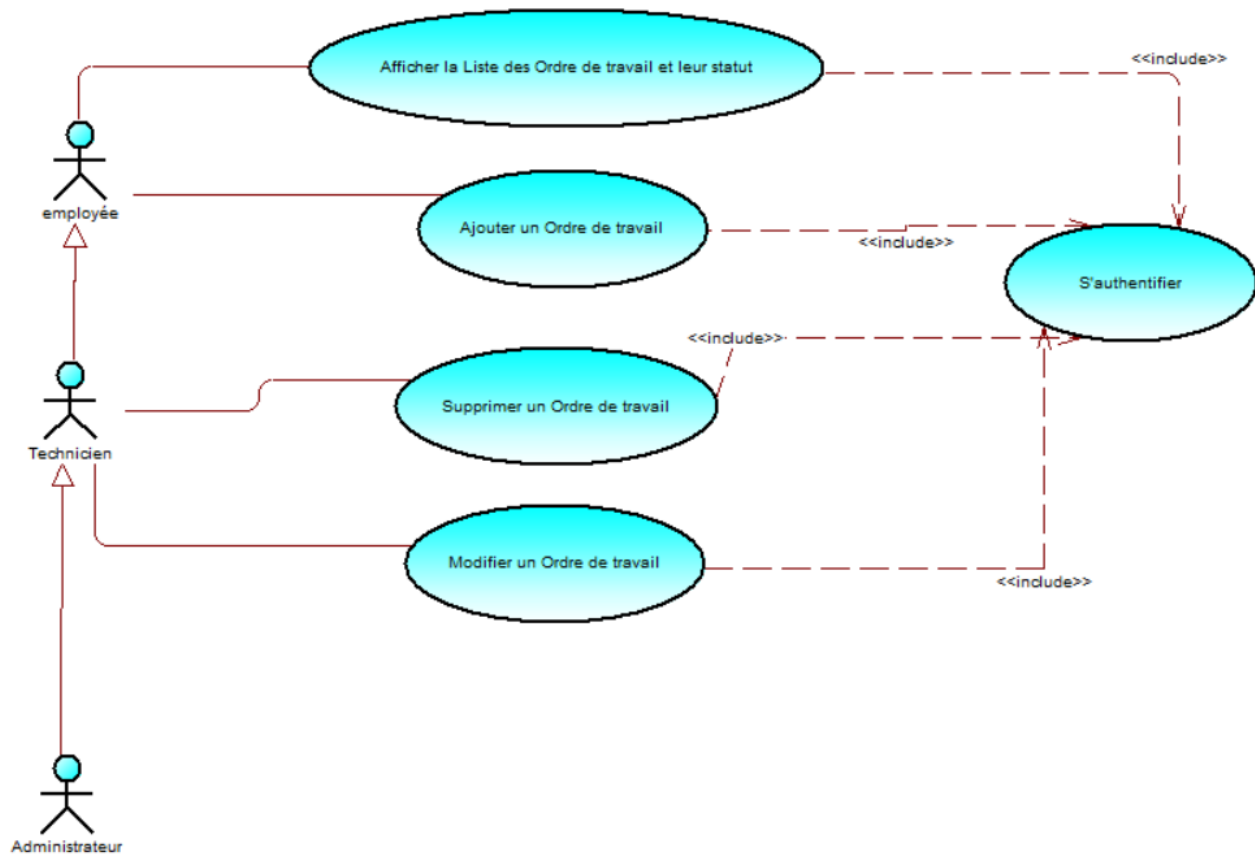


Figure 9: Diagramme de cas d'utilisation gestion des ordres de travail

La figure 9 présente le diagramme des cas d'utilisation de gestion des ordres de travail

- Gérer les pièce de rechange : Permet à l'administrateur d'ajouter , Supprimer ,Modifier les ordres de travail et Afficher la liste des ordre de travail et aussi changer leur statut (overt/terminer) .
- Les utilisateur avec le rôle technicien ne peuvent voir que les odres de travail qui leur sont assignés et aussi ajouter Supprimer ,Modifier et changer les statut des ordres de travail qu'ils ont créer au leur ont assignés.
- Les utilisateur avec le rôle employé ne peuvent que ajouter les ordre de travail et voir leur statut.

5 Diagramme de classe

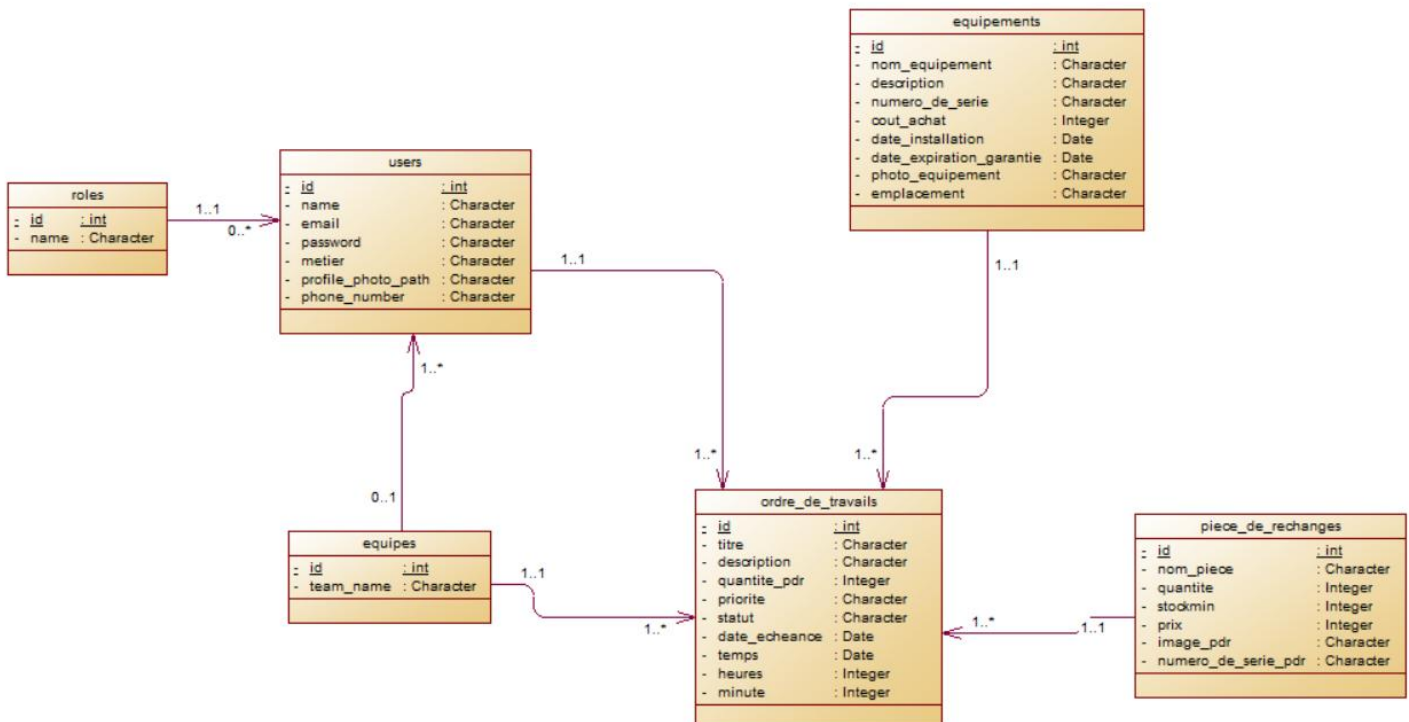


Figure 10:diagramme de classe

Le diagramme de classe est un outil de modélisation statique utilisé en génie logiciel pour représenter les différentes classes d'un système, ainsi que les relations et les interactions entre elles. Il offre une vue globale de la structure du système en mettant l'accent sur les classes, leurs attributs et leurs méthodes.

Le diagramme de classe est composé de plusieurs éléments clés, notamment :

1. **Classe** : Représente une entité du système et regroupe les attributs (variables) et les méthodes (fonctions) qui lui sont associées. Une classe est généralement représentée par un rectangle divisé en trois sections : le nom de la classe, les attributs et les méthodes.
2. **Attribut** : il décrit une caractéristique ou une propriété d'une classe.

Les attributs sont généralement affichés sous la forme de paires nom/valeur, indiquant le nom de l'attribut et son type de données.

3. **Méthode** : Représente le comportement ou les actions qu'une classe peut effectuer.
4. **Relation** : Les relations entre les classes sont représentées par les lignes et des symboles spécifiques. Par exemple, l'agrégation, la composition, l'héritage et l'association sont des type de relations couramment utilisées.

6 MCD

Le Modèle Conceptuel de Données (MCD) est une étape cruciale dans la conception de notre application web GMAO. Il permet de représenter de manière abstraite les concepts clés et les relations entre eux, en fournissant une vision claire de la structure des données du système .

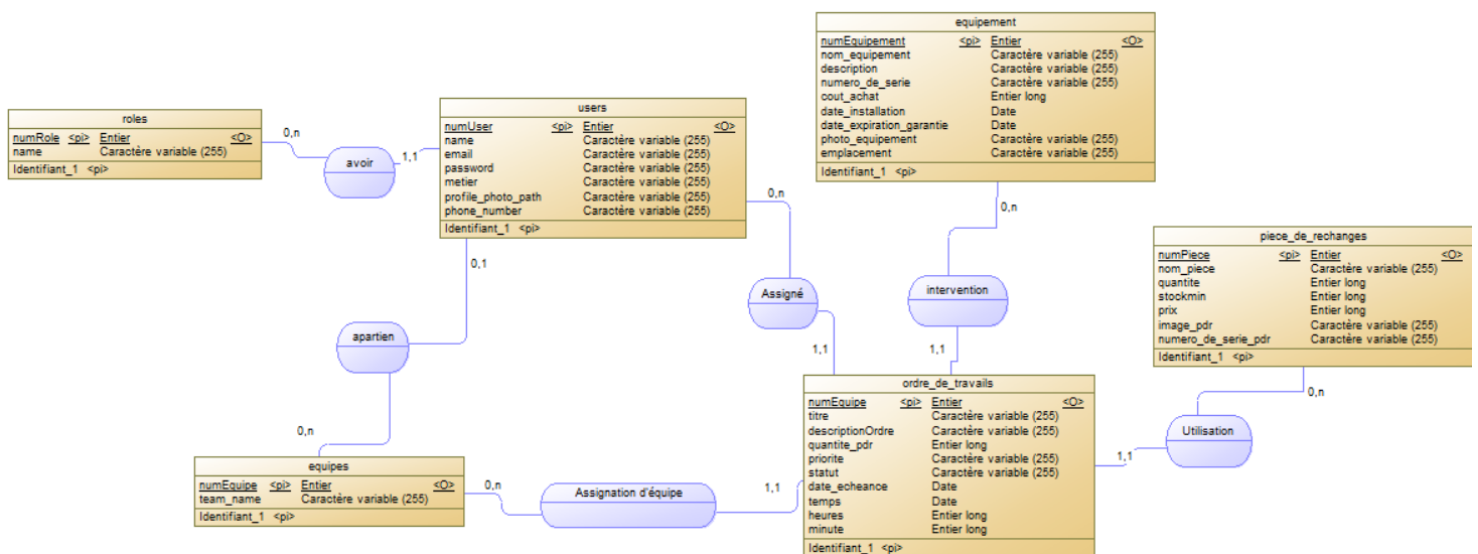


Figure 11: MCD

8 Conclusion

En résumé, ce chapitre d'analyse nous a fourni une compréhension approfondie des besoins du système, tant sur le plan fonctionnel que non fonctionnel. En combinant l'analyse des besoins, le diagramme de cas d'utilisation général et le diagramme de classe, nous avons pu établir une base solide pour la prochaine phase du projet, à savoir la conception. Ces éléments analytiques sont essentiels pour garantir la réussite du développement du système et la satisfaction des utilisateurs.

Chapitre3

Littérature/Bibliographique

1 Introduction

Dans ce chapitre nous examinerons plus en détail les frameworks, en nous concentrant sur les framework Front-end et back-end. Nous discuterons également des raisons d'utiliser un framework pour le développement d'applications web, notamment l'augmentation de la productivité, la réutilisabilité du code, la standardisation, la sécurité et l'évolutivité

2 Les technologies utilisées :

2.1 Qu'est-ce qu'un Framework ?

Un Framework est un ensemble de composants logiciels, de conventions de conception, de bibliothèques, de structures de données et de modèles de programmation qui fournissent une base pour le développement d'applications. Il s'agit essentiellement d'un cadre de travail ou d'une infrastructure qui permet aux développeurs de créer plus facilement des applications en fournissant des outils et des fonctionnalités réutilisables.

Les Framework sont souvent utilisés pour simplifier le développement d'applications en fournissant une structure de base et des fonctionnalités courantes, telles que la gestion des bases de données, la sécurité, l'authentification, la gestion des sessions, la validation des entrées utilisateur, la gestion des erreurs, la gestion des fichiers, la gestion des connexions réseau, etc.

Les Framework sont disponibles pour de nombreux langages de programmation populaires, tels que Python, Ruby, PHP, Java, JavaScript, etc. Les développeurs peuvent utiliser des Framework pour gagner du temps en évitant de devoir réinventer la roue pour chaque nouvelle application.

Ils peuvent également utiliser des Framework pour améliorer la qualité de leur code en adoptant les bonnes pratiques de conception et de développement proposées par le Framework

2.2 Front-End Framework :

Un Front End Framework est un ensemble d'outils, de bibliothèques et de technologies qui permettent aux développeurs web de concevoir et de créer des interfaces utilisateur pour des sites, des applications web et des applications mobiles.

Les Front End Framework offrent des fonctionnalités et des composants réutilisables tels que des boutons, des formulaires, des modales, des menus, des carrousels, des graphiques et bien plus encore, qui facilitent la création d'interfaces utilisateur attrayantes et efficaces.

Voici les Front End Framework les plus connus :

- **Bootstrap**
- **Materialize**
- **Foundation**
- **Semantic UI**

2.3 Back-End Framework (PHP) :

Un Framework PHP est un ensemble de bibliothèques, de composants et d'outils qui permettent de faciliter le développement de sites web et d'applications web côté serveur en utilisant le langage PHP.

Les Framework en PHP offre des fonctionnalités telles que la gestion des requêtes et des réponses HTTP, la gestion de base de données, la sécurité, la gestion de sessions utilisateur, et bien plus encore. Les développeurs peuvent ainsi se concentrer sur le développement des fonctionnalités de leur application, sans avoir à réinventer la roue à chaque fois.

Il existe plusieurs Back-End Framework populaires en PHP tels que :

- **Laravel**
- **Symfony**
- **CodeIgniter**
- **CakePHP**

Chacun de ces Framework a ses propres avantages et inconvénients, et il est important de choisir celui qui convient le mieux à vos besoins et à votre style de développement.

2.4 Pourquoi un Framework ?

Voici quelques raisons pour utiliser un Framework pour développer une application web :

- **Productivité accrue** : un Framework fournit une structure et une organisation prédéfinies qui vous permettent de vous concentrer sur la logique de votre application plutôt que sur les aspects techniques de l'infrastructure.
- **Réutilisabilité** : un Framework fournit des bibliothèques et des fonctionnalités communes qui peuvent être réutilisées pour plusieurs projets.
- **Standardisation** : en utilisant un Framework, vous utilisez des conventions communes qui facilitent la collaboration avec d'autres développeurs et l'entretien du code.
- **Sécurité** : les Framework ont souvent des fonctionnalités de sécurité intégrées, telles que la protection contre les attaques par injection SQL ou la protection contre les failles XSS.
- **Evolutivité** : un Framework peut faciliter la mise à l'échelle de votre application en fournissant une architecture flexible et modulaire.

En fin de compte, l'utilisation d'un Framework dépendra de vos besoins spécifiques en matière de développement. Si vous avez besoin d'une solution rapide et standardisée pour développer une application, un Framework peut être une bonne option.

2.5 Analyse des différents Framework :

2.5.1 *Front-End*

➤ **Bootstrap :**

Bootstrap est basé sur HTML , CSS et JavaScript, et il est compatible avec la plupart des navigateurs web modernes. Il est largement utilisé dans l'industrie du développement web pour créer des sites web professionnels et responsives rapidement et efficacement.

➤ **Materialize :**

Materialize est un Framework de conception d'interface utilisateur qui offre une bibliothèque complète de composants de design et de fonctionnalités pour créer des sites web modernes et réactifs, basés sur les principes de Google Material Design.

➤ **Foundation :**

Foundation est souvent comparé à Bootstrap, un autre Framework CSS populaire. Cependant, Foundation se concentre davantage sur la personnalisation et la flexibilité, tandis que Bootstrap a tendance à être plus strict et standardisé. Foundation est utilisé par de nombreuses entreprises et organisations pour créer des sites web professionnels et modernes

➤ **Semantic UI :**

Le Framework Semantic UI se distingue des autres Framework tels que Bootstrap ou Foundation en mettant l'accent sur la sémantique HTML et le langage naturel, plutôt que sur les classes abstraites.

Les classes CSS de Semantic UI sont basées sur des noms de concepts naturels tels que card , button , menu , form , modal ,etc.

En outre, Semantic UI est également compatible avec les dernières versions de jQuery et de React, ce qui en fait un choix populaire pour les développeurs web travaillant avec ces technologies.

2.5.2 Back-End

➤ **Laravel :**

Laravel est conçu pour simplifier le processus de développement web en fournissant une structure solide et des outils pour la gestion des tâches courantes telles que la gestion de base de données, l'authentification utilisateur, la validation de formulaire, la création de routes, la gestion des sessions et des cookies, la création de modèles de données ,la gestion des migrations de base de données, et plus encore.

Laravel utilise une architecture de modèle-vue-contrôleur (MVC) qui facilite la séparation des préoccupations entre les différentes parties de l'application web, ce qui permet une maintenance et une évolutivité plus facile à long terme.

Laravel est également livré avec de nombreuses fonctionnalités avancées, telles que l'envoi d'e-mails, la planification de tâches cron, l'intégration de services tiers via des API, la gestion de la file d'attente de tâches, et plus encore. En résumé, Laravel est un framework puissant et polyvalent qui facilite grandement le développement d'applications web en PHP.

➤ **Symfony :**

Symfony est un Framework PHP open source conçu pour faciliter le développement d'applications web et d'API. Il a été créé en 2005 par Fabien Potencier et il est maintenant l'un des Framework les plus populaires pour le développement web en PHP.

Symfony fournit une gamme d'outils pour faciliter le développement, notamment une bibliothèque de composants réutilisables qui peuvent être intégrés à d'autres projets, ainsi que des outils de ligne de commande pour la gestion de projets.

➤ **CodeIgniter :**

CodeIgniter est conçu pour être facile à apprendre, à utiliser et à déployer, tout en offrant une grande flexibilité et un haut niveau de personnalisation. Il est également connu pour sa légèreté et sa rapidité d'exécution, ce qui en fait un choix populaire pour les développeurs d'applications web de petite et moyenne envergure .

➤ **CakePHP :**

Le Framework CakePHP suit les conventions sur la configuration, plutôt que de nécessiter des configurations supplémentaires. Il offre également une gamme de fonctionnalités intégrées, telles que l'authentification, la validation de formulaires et la gestion des cookies, ce qui permet aux développeurs de se concentrer sur la création de leur application plutôt que sur la mise en place d'éléments de base.

2.6 Choix du Framework et explication de ce choix

2.6.1 Front-End



Figure 13: LOGO BOOTSTRAP

Bootstrap est l'un des Framework les plus populaires pour le développement de sites web responsives et mobiles.

Voici quelques avantages d'utiliser Bootstrap par rapport à d'autres Framework :

➤ **Grande communauté de développeurs :** Bootstrap est utilisé par de nombreux développeurs à travers le monde, ce qui signifie que l'on peut facilement trouver des solutions à des problèmes. Il y a également une grande quantité de documentation et de ressources en ligne pour nous aider à apprendre Bootstrap.

➤ **Facilité d'utilisation :** Bootstrap est conçu pour être facile à utiliser, avec une syntaxe claire et cohérente pour nous aider à créer rapidement des sites web. Il fournit également des modèles et des composants prêts à l'emploi pour nous aider à gagner du temps.

➤ **Design responsif** : Bootstrap est conçu pour être responsif, ce qui signifie que votre site web s'adaptera automatiquement à différents appareils et tailles d'écran, ce qui est de plus en plus important à l'ère des smartphones et des tablettes.

➤ **Compatibilité avec de nombreux navigateurs** : Bootstrap est conçu pour être compatible avec la plupart des navigateurs web modernes, ce qui signifie que notre application web fonctionnera correctement sur une grande variété d'appareils et de systèmes d'exploitation.

➤ **Mises à jour régulières** : Bootstrap est régulièrement mis à jour pour corriger les bugs, améliorer les performances et ajouter de nouvelles fonctionnalités. Cela implique que votre site web sera constamment actualisé et utilisera les dernières avancées technologiques.

En résumé, Bootstrap offre une grande facilité d'utilisation, une grande communauté de développeurs, un design responsif, une personnalisation facile, une compatibilité avec de nombreux navigateurs et des mises à jour régulières, ce qui en fait un choix populaire pour le développement de sites web modernes.

2.6.2 Back-End



Figure 14: LOGO Laravel

Il y a plusieurs avantages à utiliser le Framework Laravel par rapport à d'autres Framework :

➤ **Architecture élégante et expressive** : Laravel utilise une architecture MVC claire et facile à comprendre, qui permet aux développeurs de structurer leur code de manière logique et cohérente. De plus, Laravel est connu pour son code élégant et expressif, qui est facile à lire et à maintenir.

➤ **Documentation complète** : Laravel dispose d'une documentation complète et bien organisée, qui est constamment mise à jour. Cela facilite la tâche des développeurs, car ils peuvent facilement trouver les informations dont ils ont besoin pour résoudre les problèmes ou mettre en place des fonctionnalités.

➤ **Facilité de développement** : Laravel dispose d'un grand nombre d'outils intégrés qui permettent aux développeurs de créer des applications plus rapidement et plus facilement.

Par exemple, le système de migration intégré permet de gérer facilement les modifications de base de données, tandis que le système de routage permet de gérer efficacement les demande HTTP.

➤ **Sécurité intégrée** : Laravel est conçu avec la sécurité à l'esprit, ce qui signifie qu'il dispose de nombreuses fonctionnalités intégrées pour protéger les applications contre les attaques courantes, telles que les attaques par injection SQL, les attaques XSS et les attaques CSRF.

➤ **Intégration facile** : laravel est compatible avec de nombreux autres outils et bibliothèques, ce qui facilite l'intégration avec d'autres technologies. Par exemple, Laravel peut être utilisé avec des bases de données relationnelles et NoSQL, ainsi qu'avec des outils de Front-End tels que React et Vue.js.

➤ **Grande communauté** : Laravel dispose d'une large communauté de développeurs, qui contribuent constamment à l'amélioration du framework en ajoutant de nouvelles fonctionnalités et en corrigeant les bugs. Cette communauté est également très active sur les forums, les groupes de discussion et les réseaux sociaux, ce qui permet aux développeurs de poser des questions et de trouver des solutions rapidement.

En résumé, Laravel est un Framework populaire et puissant qui offre de nombreux avantages aux développeurs. Sa structure claire, sa documentation complète, sa grande communauté, ses outils intégrés, sa sécurité et son intégration facile avec d'autres technologies en font un choix populaire pour de nombreux projets web.

3 Editeur de code Visual Studio Code



Figure 15: LOGO Visual Studio Code

Pour le développement de notre application web utilisant le framework Laravel, nous avons choisi d'utiliser Visual Studio Code comme principal éditeur de code. Visual Studio Code est un éditeur de code gratuit et très populaire, offrant de nombreuses fonctionnalités et extensions spécifiquement adaptées au développement avec Laravel.

Voici comment Visual Studio Code a amélioré notre flux de travail de développement

3.1 Intégration avec Laravel Extensions :

Visual Studio Code dispose d'une large gamme d'extensions spécifiquement conçues pour le développement Laravel.

Par exemple , nous avons utilisé l'extension Laravel Extension Pack qui regroupe plusieurs extension essentielles pour Laravel, telles que Laravel Blade Snippets, Laravel Artisan , Laravel IntelliSense,etc .

Ces extensions ont grandement amélioré notre productivité en fournissant des fonctionnalités spécifiques à Laravel directement dans l'éditeur.

3.2 Prise en charge native de PHP :

Visual Studio Code offre une prise en charge native du langage de programmation PHP utilisé dans Laravel.

Il comprend une coloration syntaxique avancée pour PHP, ce qui facilite la lecture et la compréhension du code. De plus, il propose des suggestions de code intelligentes, une navigation rapide entre les fichiers et une mise en évidence des erreurs de syntaxe en temps réel, ce qui a rendu notre processus de développement plus fluide et efficace.

4 Outils et Technologies de Base de données

4.1 PowerAMC

PowerAMC est un logiciel de modélisation et de conception de base de données. Développé par Sybase, PowerAMC offre une plate-forme complète pour la modélisation conceptuelle, logique et physique des bases de données ainsi que pour la conception d'applications.



Figure 16 : Logo PowerAMC

PowerAMC permet aux développeurs, aux architectes logiciels et aux analystes de concevoir et de représenter graphiquement des structures de données, des relations entre les tables, des contraintes d'intégrité, des vues, des procédures stockées et d'autres éléments nécessaires à la création et à la gestion de bases de données. Il prend en charge diverses notations de modélisation, telles que le modèle entité-association (MEA), le modèle relationnel (MR) et le modèle objet (MO).

4.2 phpMyAdmin



Figure 17: LOGO phpMyAdmin

PhpMyAdmin, une application web open source développée en PHP, se destine à la gestion et à l'administration des bases de données MySQL. C'est un outil populaire et largement utilisé par les développeurs et les administrateurs de bases de données pour gérer facilement les opérations courantes sur une base de données, l'exécution de requêtes SQL, la gestion des utilisateurs et des privilèges, et bien plus encore. Grâce à son interface conviviale et intuitive, phpMyAdmin permet aux utilisateurs de visualiser et de manipuler les données de la base de données sans avoir à écrire de code SQL complexe. Il fournit également des fonctionnalités avancées telles que l'importation, la génération de requêtes SQL, la sauvegarde et la restauration de bases de données, et la gestion des paramètres du serveur MySQL.

5 Définition de la maintenance :

La maintenance peut être définie comme l'ensemble des activités entreprises pour préserver, rétablir ou améliorer le bon fonctionnement, la performance et la durée de vie d'un système, d'un équipement, d'un logiciel ou d'une infrastructure.

La maintenance englobe différentes tâches telles que l'inspection régulière, la réparation, le remplacement de pièces défectueuses, la prévention des pannes, la mise à jour des logiciels, la gestion des mises à niveau, l'optimisation des performances, et la résolution des problèmes rencontrés.

6 Conclusion

En résumé, l'utilisation de Framework dans le développement d'applications offre de nombreux avantages en terme de Productivité, de réutilisabilité, de standardisation, de sécurité et d'évolutivité. Le choix d'un Framework spécifique dépend des besoins et des préférences individuels, et dans ce chapitre, Bootstrap et Laravel ont été identifiés comme des choix populaires et puissants dans leurs domaines respectifs.

CHAPITRE 4 Réalisation

1 Introduction

Dans ce chapitre nous commencerons par présenter l'architecture globale de l'application, en décrivant les différents composants. Nous inclurons aussi des captures d'écran de l'interface de notre application pour donner une idée visuelle de son apparence et de son fonctionnement. Nous accorderons aussi une attention particulière à la sécurité, en expliquant les mesures prises pour garantir la protection des données et la prévention des attaques.

Cela inclura la sécurité côté back-end, où nous discuterons les techniques utilisées pour protéger les données sensibles, prévenir les attaques par injection SQL et gérer les autorisations des utilisateurs. Nous aborderons également la sécurité côté front-end, en mettant en évidence les pratiques mises en place pour éviter les attaques XSS.

2 Architecture globale de l'application

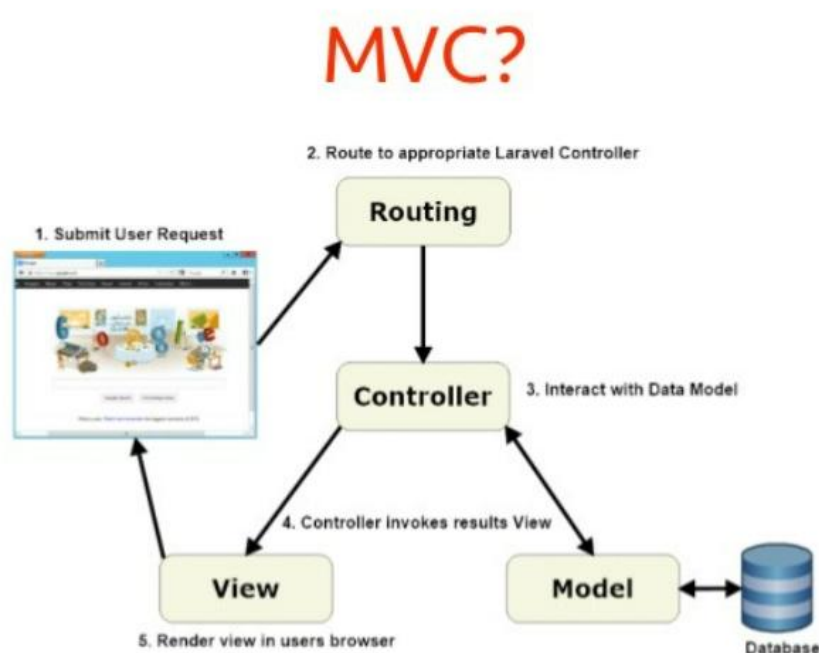


Figure 18: Architecture MVC

L'architecture d'une application Laravel repose sur le modèle de conception MVC (Modèle-Vue-Contrôleur), Qui favorise la séparation claire des responsabilités et facilite la maintenabilité du code. Voici une vue d'ensemble de l'architecture globale de notre application Laravel :

➤ **Modèle (Model)** : Le modèle représente la couche de données de l'application et est responsable de l'accès et la manipulation des données. Dans notre application, nous avons utilisé des modèles Eloquent, qui sont des classes PHP qui correspondent aux tables de la base de données. Ces modèles nous ont permis de réaliser des opérations de requête et de manipulation des données de manière simple et cohérente.

➤ **Vue (View)** : La vue est responsable de l'affichage des données à l'utilisateur. Dans Laravel, les vues sont généralement écrites en utilisant le langage de template Blade, qui offre une syntaxe claire et expressive pour générer le HTML. Nous avons utilisé des vues Blade pour définir la structure et le contenu des pages web de notre application.

➤ **Contrôleur (Controller)** : Le contrôleur gère les requêtes HTTP entrantes et interagit avec les modèles et les vues. Dans Laravel, les contrôleurs sont des classes qui regroupent des méthodes correspondantes à différentes actions de l'application. Nous avons utilisé des contrôleur pour traiter les demandes des utilisateurs ,effectuer des opérations sur les modèles et retourner les vues appropriées

➤ **Routes (Routes)** : Les routes définissent les point d'entrée de l'application et déterminent quelle action doit être exécutée en fonction de l'URL demandée. Dans Laravel, les routes sont définies dans des fichiers de configuration spécifiques. Nous avons utilisé les routes pour associer des URL spécifiques aux méthodes des contrôleurs, permettant ainsi d'acheminer les demandes vers les bonnes actions.

➤ **Middleware (Middleware)** : Les middleware sont des couches intermédiaires entre les requêtes entrantes et les réponses sortantes. Ils permettent d'effectuer des actions communes à plusieurs requêtes, comme la vérification de l'authentification de l'utilisateur, la gestion des autorisations ou la manipulation des données de la requête. Nous avons utilisé des middlewares pour mettre en place des fonctionnalités de sécurité supplémentaires, telles que la protection contre les attaques CSRF.

➤ **Base de données (Database)** : Laravel prend en charge plusieurs systèmes de gestion de bases de données, et nous avons utilisé MySQL comme SGBD pour notre application. Laravel facilite l'interrogation de la base de données en utilisant le langage SQL grâce à l'ORM (Object-Relational Mapping) Eloquent, qui offre une syntaxe expressive et permet de travailler avec les tables de la base de données en utilisant des objets PHP.

3 Sécurité côté back-end

La sécurité côté back-end joue un rôle essentiel dans le développement d'une application web en laravel .

Voici les mesures de sécurité que nous avons mises en place pour protéger notre application :

➤ **Protection contre les attaques par injection SQL** : Laravel intègre une couche de protection contre les attaques par injection SQL en utilisant des requêtes préparées et des bindings de paramètres. Nous avons utilisé activement ces fonctionnalités pour éviter toute exécution malveillante de code SQL.

➤ **Validation des données** : Laravel fournit des fonctionnalités de validation intégrées pour garantir que les données reçues de l'utilisateur sont conformes aux règles définies. Nous avons utilisé ces mécanismes de validation pour nous assurer que les données d'entrée étaient sûres et conformes aux attentes.

➤ **Protection contre les failles CSRF** : Laravel fournit une protection intégrée contre les attaques Cross-Site Request Forgery (CSRF). Nous avons activé cette protection en utilisant les jetons CSRF fournis par Laravel dans nos formulaires .

4 Sécurité côté front-end

La sécurité côté front-end est également importante pour garantir une expérience utilisateur sécurisée. voici les mesures que nous avons prises pour assurer la sécurité côté front-end :

➤ **Protection contre les attaques XSS** : Laravel fournit une protection par défaut contre les attaques de type Cross-Site Scripting (XSS) grâce à l'échappement automatique des données affichées dans les vies. Nous avons utilisé cette fonctionnalité pour empêcher l'exécution de code JavaScript malveillant.

5 Installation et organisation

Pour la réalisation de notre projet de développement d'une application web, nous avons choisi d'utiliser le Framework Laravel, qui offre une structure de projet robuste et facilite le développement rapide d'application web. Voici les étapes que nous avons suivies pour installer Laravel et organiser notre projet :

5.1 Installation de Laravel



Figure 19 : logo composer

Pour commencer nous avons installé composer un gestionnaire de dépendances pour PHP, en suivant les instructions fournies sur le site officiel de Composer. Une fois composer installé, nous avons créé un nouveau projet Laravel en utilisant la commande suivante dans notre terminal ou invite de commandes :

```
composer create-project laravel/laravel example-app
```

Cette commande a téléchargé les fichiers de base de Laravel et configuré automatiquement la structure initiale du projet.

5.2 Organisation de Laravel

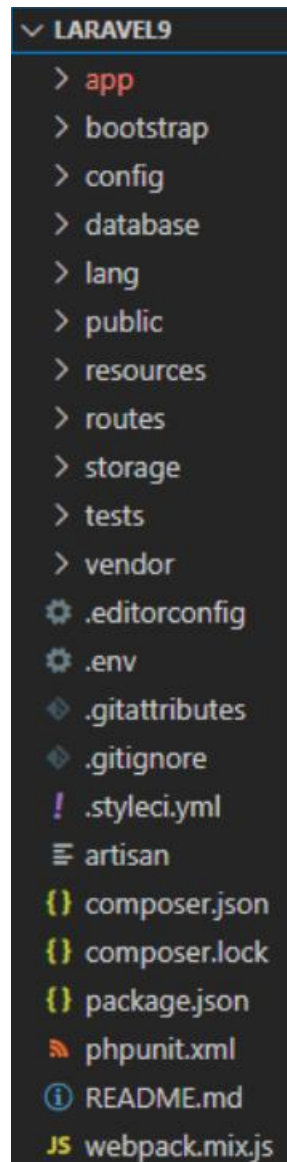


Figure 20 : Dossier app

Pour maintenir une structure de projet claire et cohérente, nous avons organisé notre code et nos fichiers Selon les bonnes pratiques de Laravel. Voici la structure de base que nous avons utilisée :

- **App** : Ce répertoire contient le code principal de notre application Laravel, y compris les modèles(models), les contrôleurs (controllers) et les vues (views).

• **Routes** : Ce répertoire contient les fichiers de définition des routes de notre application. Le fichier principal est 'web.php', où nous avons défini les routes HTTP pour nos différentes pages et actions.

• **Database** : Ce répertoire contient les migrations et les seeders. Les migrations sont utilisées pour définir et mettre à jour la structure de notre base de données, tandis que les seeders nous permettent de peupler la base de données avec des données de test.

• **Resources** : Ce répertoire contient les fichiers de ressources, tels que les fichiers CSS, JavaScript, les images, les vues Blade, etc. Nous avons organisé ces fichiers en sous-répertoires tels que 'css', 'js', 'img', 'views', etc, pour une meilleure gestion des ressources.

• **Public** : Ce répertoire contient les fichiers accessibles publiquement, tels que les fichiers CSS et JavaScript Compilés, les images, les fichiers de police, etc. Laravel gère automatiquement le routage des requêtes vers ce répertoire pour servir les fichiers statiques.

• **Config** : Ce répertoire contient les fichiers de configuration de notre application. Nous y avons défini divers paramètres tels que les informations de base de données, le service de messagerie, les services tiers, etc.

• **Storage** : Ce répertoire contient les fichiers générés par notre application, tels que les fichiers journaux, les fichiers de session, les fichiers de cache, etc.

• **Tests** : Ce répertoire contient les tests automatisés de notre application, écrits à l'aide du Framework de tests intégré de Laravel.

• **artisan** : est un outil en ligne de Laravel utilisé pour effectuer diverses tâches de gestion.

• **composer.json** : fichier de référence de composer

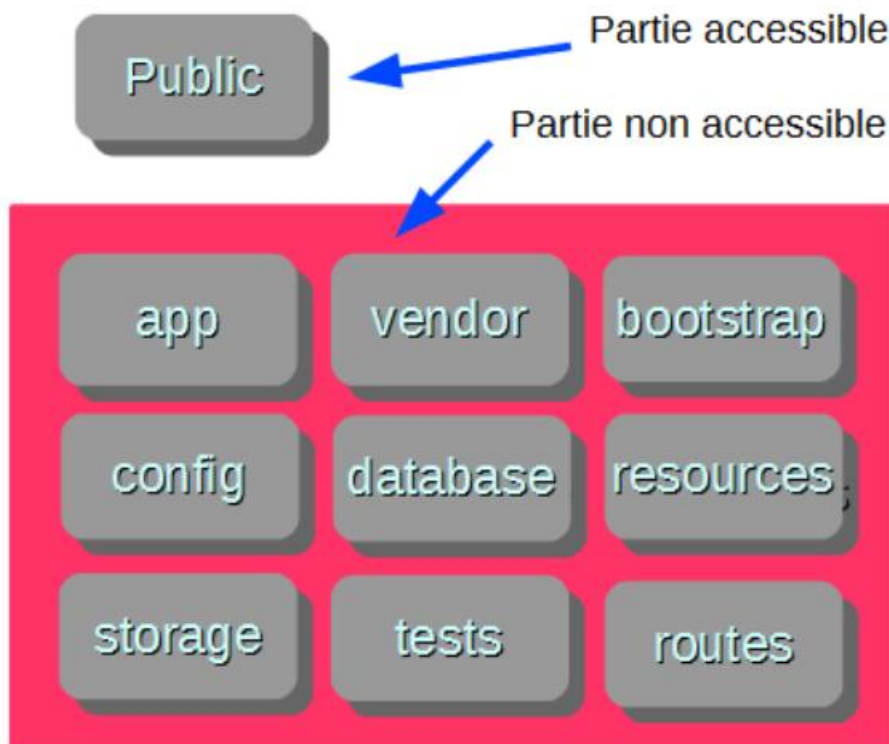
• **package.json** : est le fichier de référence de npm utilisé pour les assets.

• **phpunit.xml** : est le fichier de configuration de phpunit utilisé pour les tests unitaires.

- **.env** : fichier pour spécifier l'environnement d'exécution.

Accessibilité

Dans le but d'assurer la sécurité du serveur, il est impératif que seul le dossier **public** soit accessible :



6 Artisan

Lorsqu'on développe une application avec Laravel, diverses tâches doivent être accomplies, telles que la création de classes et la vérification des routes....

C'est à ce moment-là qu'entre en jeu Artisan, un compagnon indispensable pour le développement avec Laravel. Artisan fonctionne en ligne de commande, ce qui signifie qu'il s'utilise à partir de la console. Il vous suffit de vous placer dans le répertoire principal et d'exécuter la commande suivante :

```
php artisan
```

Pour obtenir la liste de ses possibilités, en voici un extrait :

```
Usage:
  command [options] [arguments]

Options:
  -h, --help            Display help for the given command. When no command is given display help for the list command
  -q, --quiet           Do not output any message
  -V, --version         Display this application version
  --ansi|--no-ansi     Force (or disable --no-ansi) ANSI output
  -n, --no-interaction  Do not ask any interactive question
  --env[=ENV]          The environment the command should run under
  -v|vv|vvv, --verbose Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for debug

Available commands:
  clear-compiled  Remove the compiled class file
  completion     Dump the shell completion script
  db             Start a new database CLI session
  down           Put the application into maintenance / demo mode
  env            Display the current framework environment
  help          Display help for a command
  inspire       Display an inspiring quote
  list          List commands
  migrate       Run the database migrations
  optimize      Cache the framework bootstrap files
  serve        Serve the application on the PHP development server
  test         Run the application tests
  tink         Interact with your application
  up           Bring the application out of maintenance mode
```

7 Les migrations et la configuration de la base

7.1 Les migrations

Une migration dans Laravel est un processus qui permet de créer et de mettre à jour la structure d'une base de données. En d'autres termes, vous pouvez créer de nouvelles tables, ajouter ou supprimer des colonnes, créer des index, et effectuer d'autres opérations de maintenance sur vos tables. Cet outil vous offre ainsi un suivi précis de toutes les modifications apportées à votre base de données.

7.2 La configuration de la base

Avant toute chose, vous devez disposer d'une base de données. Laravel offre la possibilité de gérer des bases de données de différents types tels que MySQL, Postgres, SQLite et SQL Server.

Dans le fichier de configuration **.env**, vous devez spécifier l'emplacement de votre base de données, son nom, le nom d'utilisateur et le mot de passe.

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=laravel
DB_USERNAME=root
DB_PASSWORD=
```

Figure 21 : fichier .env

Dans cet exemple, nous présentons les valeurs par défaut lors de l'installation de Laravel. Cependant, il est essentiel de modifier ces valeurs en fonction de votre environnement de développement. Plus spécifiquement, vous devrez définir le nom de la base de données, le nom d'utilisateur et le mot de passe. Pour une installation locale de MySQL, l'utilisateur par défaut est généralement "root" et il n'y a pas de mot de passe.

Installation

Si vous regardez dans le dossier **database/migrations** il y a déjà 4 migrations présentes :

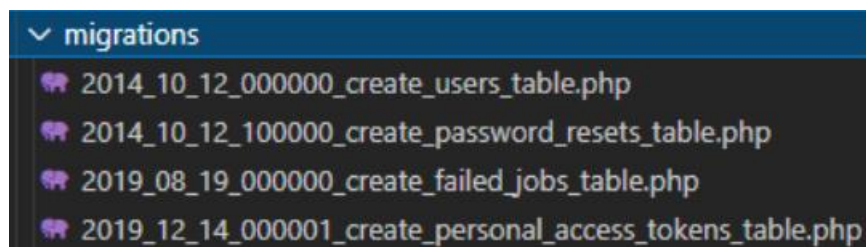


Figure 22 : Dossier migrations

- Table **users** : C'est une migration de base pour créer une table des utilisateurs,
- Table **password_resets** : c'est une migration liée à la précédente qui permet de gérer le renouvellement des mots de passe en toute sécurité,
- Table **failed_jobs** : une migration qui concerne les queues,
- Table **personal_access_tokens** concerne les api.

Constitution d'une migration

Si vous ouvrez le fichier `database/migrations/2014_10_12_000000_create_table.php`

Vous trouvez ce code :

```
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('users', function (Blueprint $table) {
            $table->id();
            $table->string('name');
            $table->string('email')->unique();
            $table->timestamp('email_verified_at')->nullable();
            $table->string('password');
            $table->rememberToken();
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('users');
    }
};
```

On dispose dans cette classe de deux fonctions :

- **up** : ici on a le code de création de la table et de ses colonnes
- **down** : ici on a le code de suppression de la table

Lancer les migrations

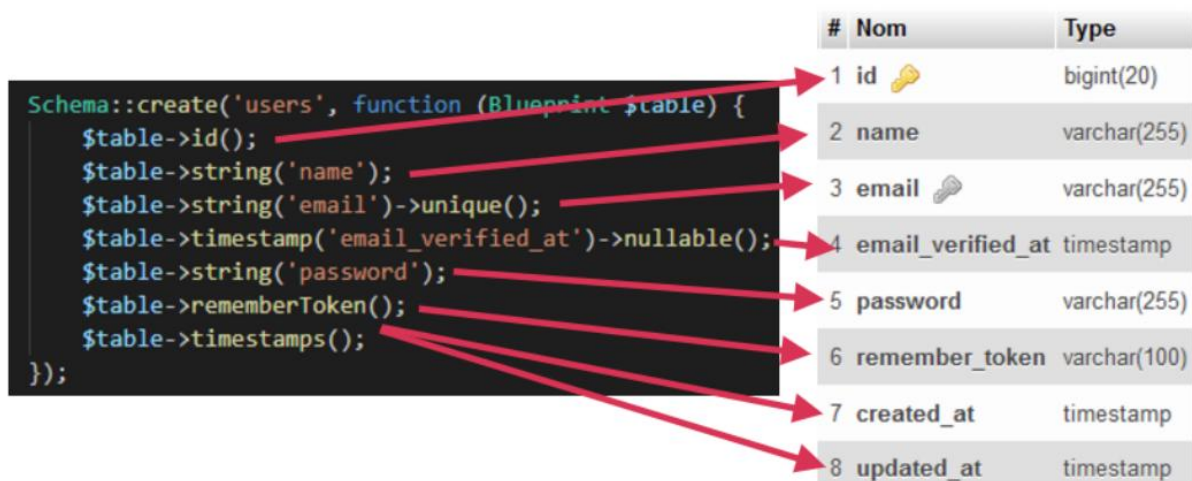
Pour lancer les migration on utilise la commande **migrate** :

```
E:\laragon\www\laravel9>php artisan migrate
Migrating: 2014_10_12_000000_create_users_table
Migrated: 2014_10_12_000000_create_users_table (12.66ms)
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated: 2014_10_12_100000_create_password_resets_table (11.33ms)
Migrating: 2019_08_19_000000_create_failed_jobs_table
Migrated: 2019_08_19_000000_create_failed_jobs_table (12.24ms)
Migrating: 2019_12_14_000001_create_personal_access_tokens_table
Migrated: 2019_12_14_000001_create_personal_access_tokens_table (16.56ms)
```

On voit que les 4 migrations présentes ont été exécutées et on trouve les 4 tables dans la base (en plus de celle de gestion des migration) :

Table ▲
failed_jobs
migrations
password_resets
personal_access_tokens
users
5 tables

Pour comprendre le lien entre migration et création de la table associée voici une illustration pour la table **users** :



La méthode `timestamps` permet la création des deux colonnes **created_at** et **updated_at**.

Annuler ou rafraichir une migration

La commande **rollback** est utilisée pour annuler une migration :

```
E:\laragon\www\laravel9>php artisan migrate:rollback  
Rolling back: 2019_12_14_000001_create_personal_access_tokens_table  
Rolled back: 2019_12_14_000001_create_personal_access_tokens_table (6.45ms)  
Rolling back: 2019_08_19_000000_create_failed_jobs_table  
Rolled back: 2019_08_19_000000_create_failed_jobs_table (2.33ms)  
Rolling back: 2014_10_12_100000_create_password_resets_table  
Rolled back: 2014_10_12_100000_create_password_resets_table (2.25ms)  
Rolling back: 2014_10_12_000000_create_users_table  
Rolled back: 2014_10_12_000000_create_users_table (2.39ms)
```

Cela exécute les méthodes **down** des migration et supprime les tables.

Pour effectuer une opération d'annulation et de relance en une seule fois, on peut utiliser la commande **refresh** :

```
E:\laragon\www\laravel9>php artisan migrate:refresh
Nothing to rollback.
Migrating: 2014_10_12_000000_create_users_table
Migrated: 2014_10_12_000000_create_users_table (13.22ms)
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated: 2014_10_12_100000_create_password_resets_table (10.32ms)
Migrating: 2019_08_19_000000_create_failed_jobs_table
Migrated: 2019_08_19_000000_create_failed_jobs_table (9.61ms)
Migrating: 2019_12_14_000001_create_personal_access_tokens_table
Migrated: 2019_12_14_000001_create_personal_access_tokens_table (13.89ms)
```

Afin d'éviter d'avoir à coder la méthode down, on dispose de la commande **fresh** qui supprime automatiquement les tables concernées :

```
E:\laragon\www\laravel9>php artisan migrate:fresh
Dropped all tables successfully.
Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table
Migrated: 2014_10_12_000000_create_users_table (9.94ms)
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated: 2014_10_12_100000_create_password_resets_table (10.42ms)
Migrating: 2019_08_19_000000_create_failed_jobs_table
Migrated: 2019_08_19_000000_create_failed_jobs_table (9.61ms)
Migrating: 2019_12_14_000001_create_personal_access_tokens_table
Migrated: 2019_12_14_000001_create_personal_access_tokens_table (14.18ms)
```

Créer une migration

Il existe une commande d'artisan pour créer un squelette de migration :

```
E:\laragon\www\laravel9>php artisan make:migration test
Created Migration: 2022_01_21_213010_test
```

La migration est créée dans le dossier :

```
▼ migrations
  2014_10_12_000000_create_users_table.php
  2014_10_12_100000_create_password_resets_table.php
  2019_08_19_000000_create_failed_jobs_table.php
  2019_12_14_000001_create_personal_access_tokens_table.php
  2022_01_21_213010_test.php
```

Avec ce code de base :

```
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        //
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        //
    }
};
```

Ensuite, il est nécessaire de compléter ce code en fonction de vos besoins spécifiques !

Le nom du fichier de migration commence par la date de création, ce qui détermine son positionnement dans la liste des migrations. Il est crucial de noter que l'ordre des migrations revêt une grande importance lorsque des clés étrangères sont impliquées. Assurez-vous donc de respecter l'ordre logique des migrations pour éviter les erreurs liées aux dépendances entre les tables.

8 Les interfaces de l'application

L'utilisateur doit saisir son nom et son mot de passe via l'interface d'authentification pour accéder à la page d'accueil de l'application web. Si l'utilisateur n'a pas de nom utilisateur et mot de passe il peut s'inscrire en utilisant son email .

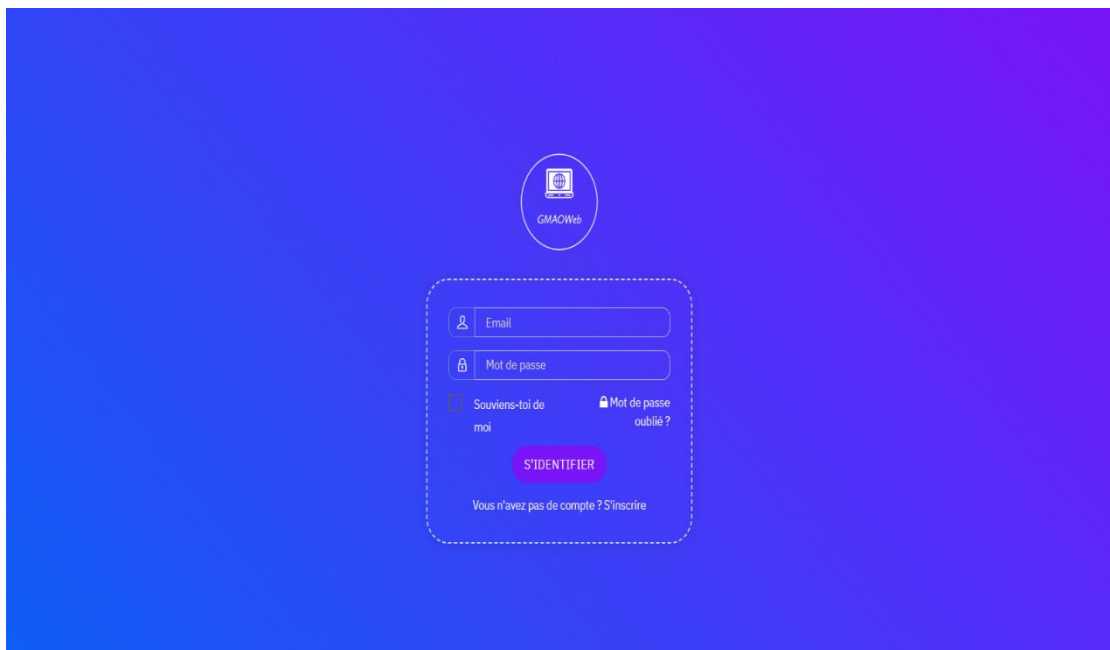


Figure 22: Interface d'authentification

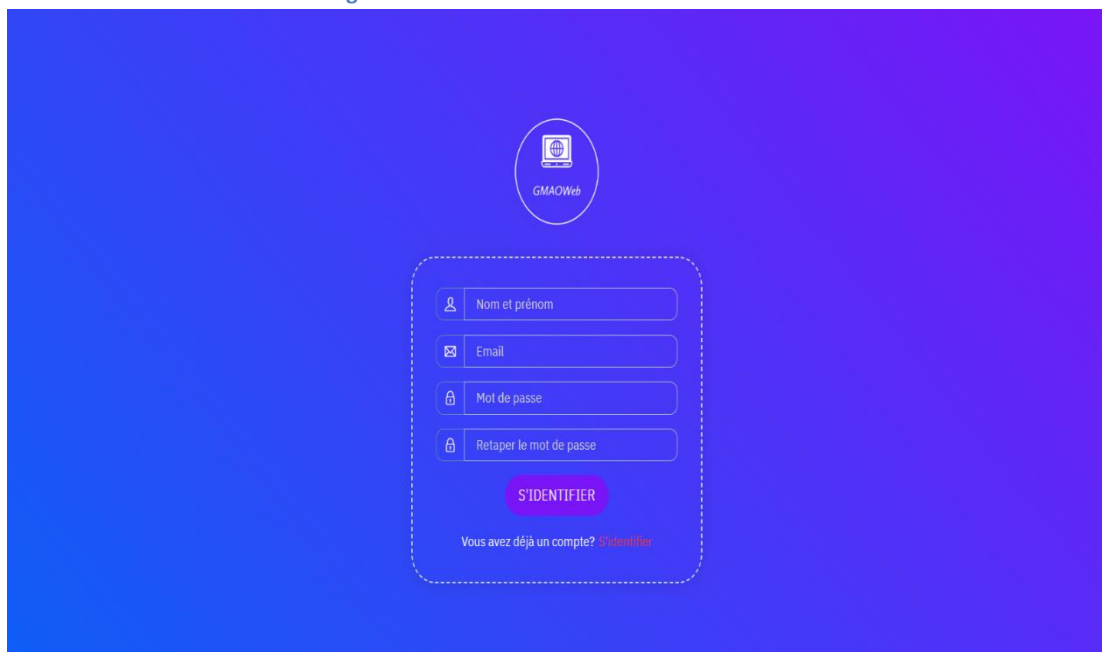


Figure 23 : interface pour s'inscrire

9 Interface de Tableau de bord

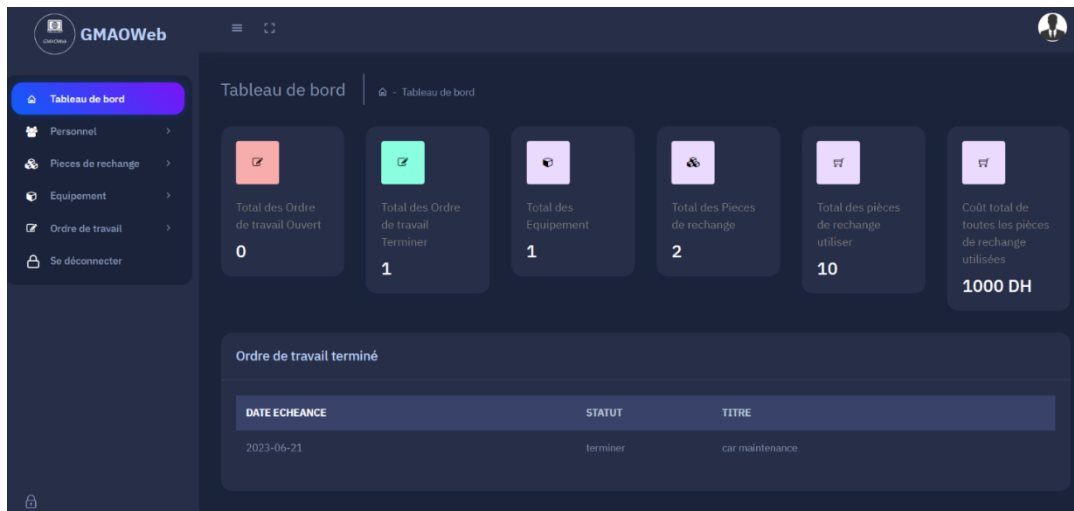


Figure 23: Interface de Tableau de bord

La première interface qui s’affiche après que l’utilisateur s’authentifie c’est la page du tableau de bord qui représente le total des ordre de travail qui sont ouvert et terminer aussi afficher le total des équipement et des pièces de rechange stocké et le total des pièce de rechange utiliser ,aussi calculer le coût total de toutes les pièces de rechange utilisées

10 Interface de gestion des utilisateurs

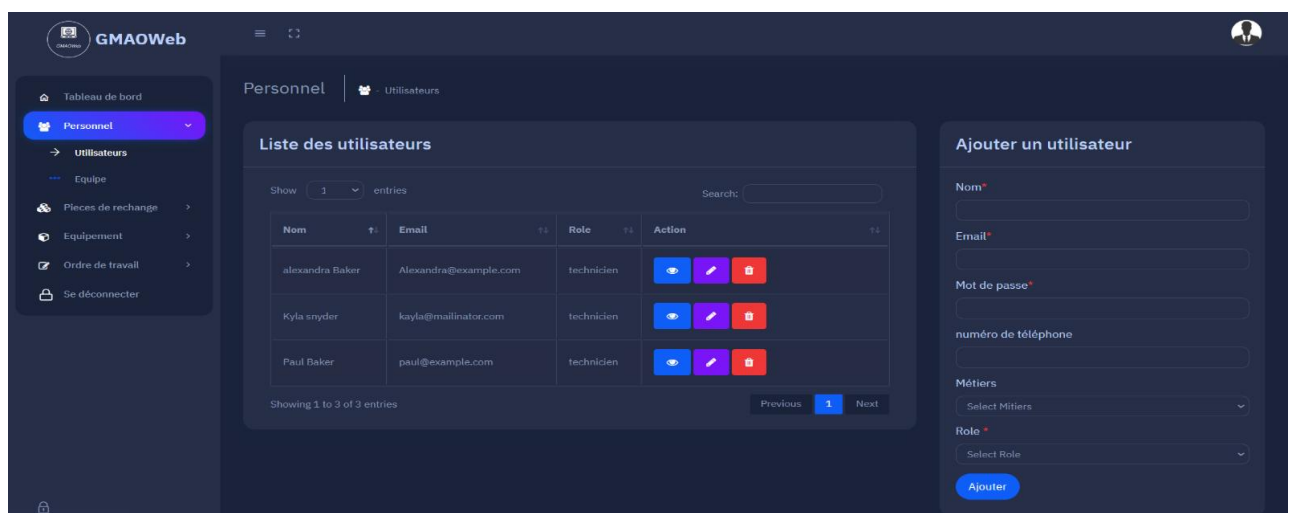


Figure 24: Interface gestion des utilisateurs

Chapitre 4 : Développement

Dans Cette page l'administrateur peut ajouter, modifier, supprimer les utilisateurs et aussi assigner un rôle a un utilisateur , il peut aussi chercher les utilisateur selon leur nom au adresse email.

11 Interface de gestion d'équipe

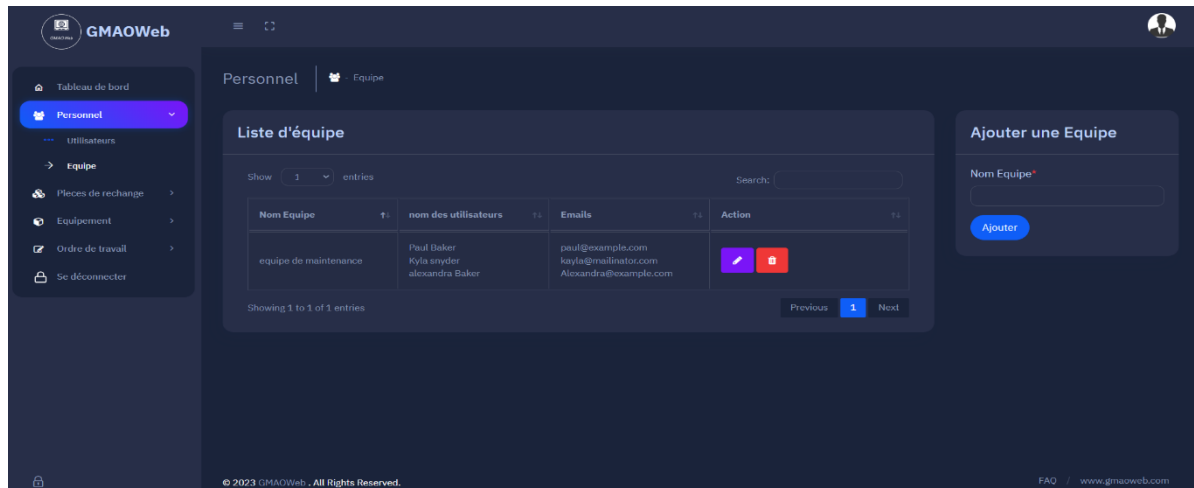


Figure 25 : interface gestion d'équipe

12 Interface de gestion de pièce de Rechange

Dans cette page l'utilisateur peut ajouter, modifier, supprimer les pièces de rechange et chercher une pièce selon son nom au son numéro de série.

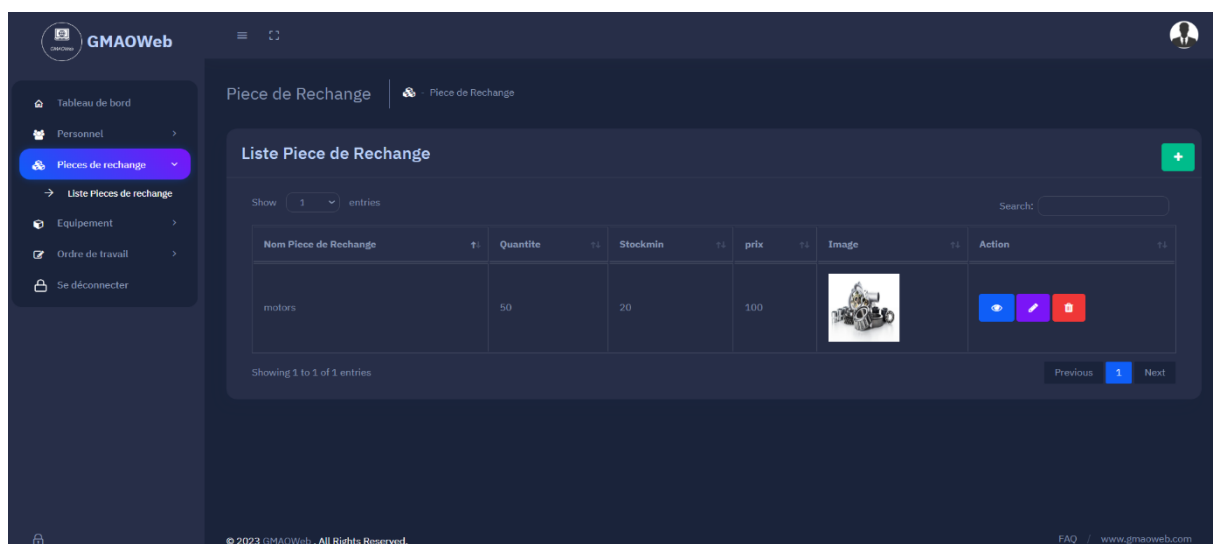


Figure 26: Interface gestion des Pièces de rechange

12.1 formulaire pour Ajouter une Pièce de Rechange

The screenshot shows the 'Ajouter Piece de Rechange' form in the GMAOWeb application. The form is located on the right side of the dashboard, with a sidebar on the left containing navigation links: 'Tableau de bord', 'Personnel', 'Pièces de rechange' (selected), 'Liste Pieces de rechange', 'Equipement', 'Ordre de travail', and 'Se déconnecter'. The form itself has the following fields:

- Nom Piece de rechange ***: A text input field containing 'seals'.
- Quantite ***: A text input field containing '100'.
- Image Piece de rechange ***: A file upload field with a preview image of a seal and a button labeled 'Choisir un fichier'.
- Numero de serie ***: A text input field containing '1523654'.
- prix ***: A text input field containing '200'.
- Stockmin ***: A text input field containing '20'.

At the bottom of the form is a blue button labeled 'Ajouter'. The footer of the application shows '© 2023 GMAOWeb . All Rights Reserved.' and 'FAQ / www.gmaoweb.com'.

Figure 27: Interface Ajouter une pièce de rechange

13 Interface de gestion d'équipements

Dans cette page l'utilisateur peut ajouter, modifier, supprimer les équipements et chercher un équipement selon leur nom au leur numéro de série.

The screenshot shows the 'Liste des équipements' page in the GMAOWeb application. The sidebar on the left has the following navigation links: 'Tableau de bord', 'Personnel', 'Pièces de rechange', 'Equipement' (selected), 'Liste Equipement', 'Ordre de travail', and 'Se déconnecter'. The main content area is titled 'Liste des équipements' and features a table with the following columns: 'Equipment Name', 'Numero de serie', 'Date d'installation', 'Photo equipment', and 'Action'. The table contains one entry for a 'car' with the serial number '123456' and the installation date '2023-06-20'. The 'Action' column for this entry contains three icons: a blue arrow, a purple pencil, and a red square. The table is paginated, showing 'Showing 1 to 1 of 1 entries' and 'Previous 1 Next'.

Equipment Name	Numero de serie	Date d'installation	Photo equipment	Action
car	123456	2023-06-20		

Figure 28: Interface gestion des équipements

13.1 le formulaire d'Ajouter un Equipement

The screenshot shows the 'Ajouter Equipement' form in the GMAOWeb application. The form is titled 'Ajouter Equipement' and contains several input fields for adding a new piece of equipment. The left sidebar shows the navigation menu with 'Equipement' selected. The form fields include:

- Nom Equipement * (text input, value: bike)
- description (textarea, value: Textarea text)
- Equipement Image * (file upload, value: Choisir un fichier Bike.jpeg)
- Numero de serie * (text input, value: 95)
- Date installation * (date picker, value: 16/02/2023)
- Date d'expiration de la garantie * (date picker, value: 01/06/2023)
- Coût d'achat* (currency input, value: DH 1.000,00)
- emplacement * (text input, value: zone2)

A blue 'Ajouter' button is at the bottom left of the form. The footer shows '© 2023 GMAOWeb . All Rights Reserved.' and 'FAQ / www.gmaoweb.com'.

Figure 29: Interface Ajouter un équipement

13.2 Supprimer un équipement

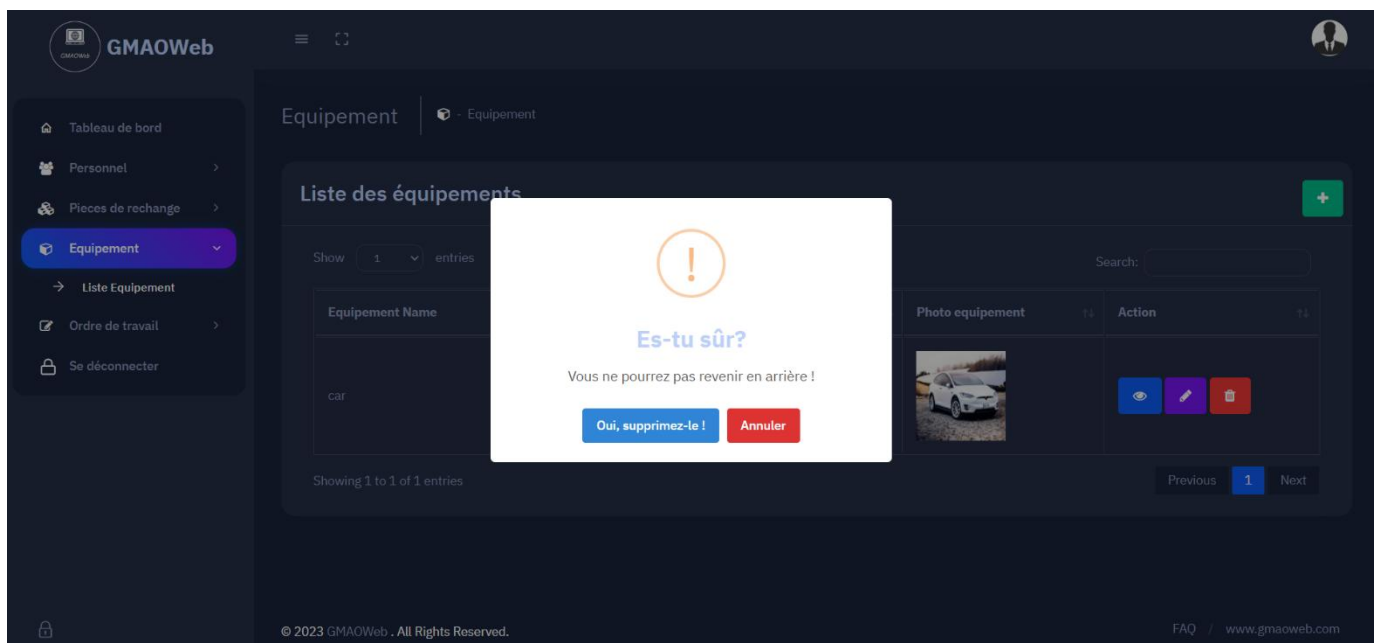


Figure 30 : interface supprimer un équipement

13.3 Historique des ordre de travail associes à un équipement

The screenshot displays the GMAOWeb application interface. On the left, a sidebar menu includes 'Tableau de bord', 'Personnel', 'Pieces de rechange', 'Equipement' (selected), 'Liste Equipement', 'Ordre de travail', and 'Se déconnecter'. The main content area shows details for a piece of equipment: 'Cout achat : 2000', 'Date d'installation : 2023-06-12', and 'Date d'expiration de garantie : 2023-09-07'. Below this, a section titled 'Historique des ordre de travail associes à Cette Equipement' features a table with one entry. The table has columns for 'Titre', 'Statut', 'Quantité piece de rechange', 'Priorite', and 'Action'. The entry shows 'car maintenance' with status 'terminer', quantity '10', and priority 'moyene'. A search bar and pagination controls are also present.

Titre	Statut	Quantité piece de rechange	Priorite	Action
car maintenance	terminer	10	moyene	

Figure 31 : Interface historique des ordres de travail associes à Cette Equipement

14 Le formulaire pour ajouter un Ordre de travail

The screenshot shows the 'Ajouter Ordre de travail' form in the GMAOWeb application. The sidebar menu is similar to the previous figure, with 'Ordre de travail' selected. The form contains several fields: 'Titre' (text input), 'Instructions' (text area), 'Equipement' (dropdown), 'Priorité' (dropdown), 'Equipe' (dropdown), 'date d'échéance' (date picker), 'Intervenant' (dropdown), and 'temps' (time picker). A blue 'Ajouter' button is at the bottom left of the form.

Figure 32: Interface Ajouter un Ordre de travail

15 Interface de gestion des ordres de travail

Dans cette page l'utilisateur peut ajouter, modifier, supprimer les ordres de travail .

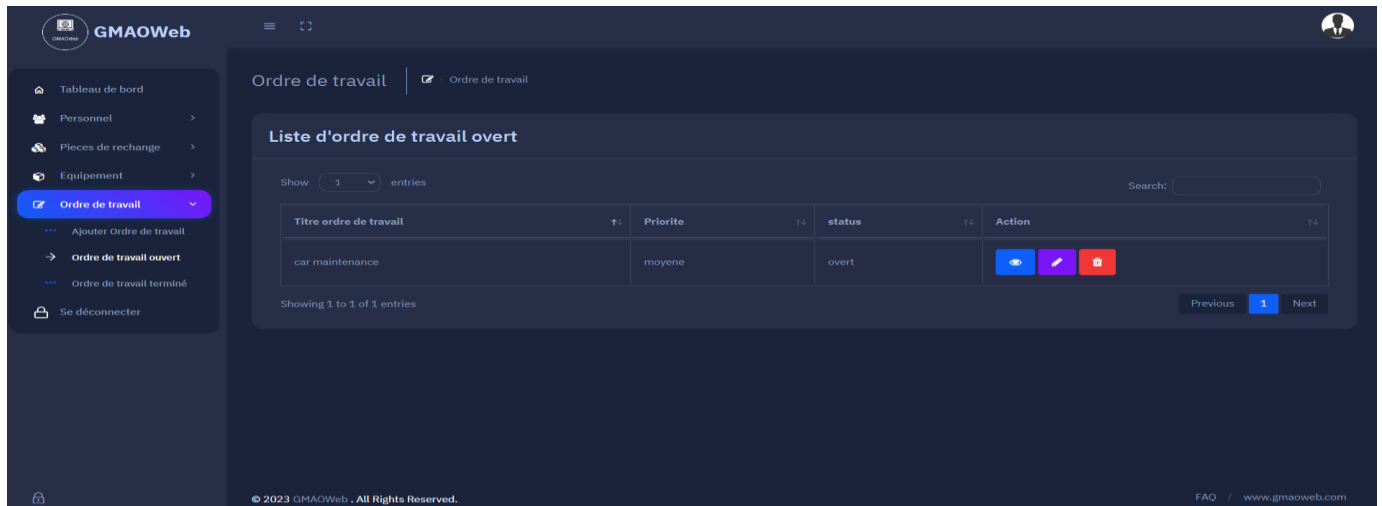


Figure 33: Interface gestion des Ordre de travail

16 Interface de gestion des ordres de travail terminé

Dans cette page l'utilisateur peut voir la liste des ordres de travail terminé et supprimer les ordre de travail terminé

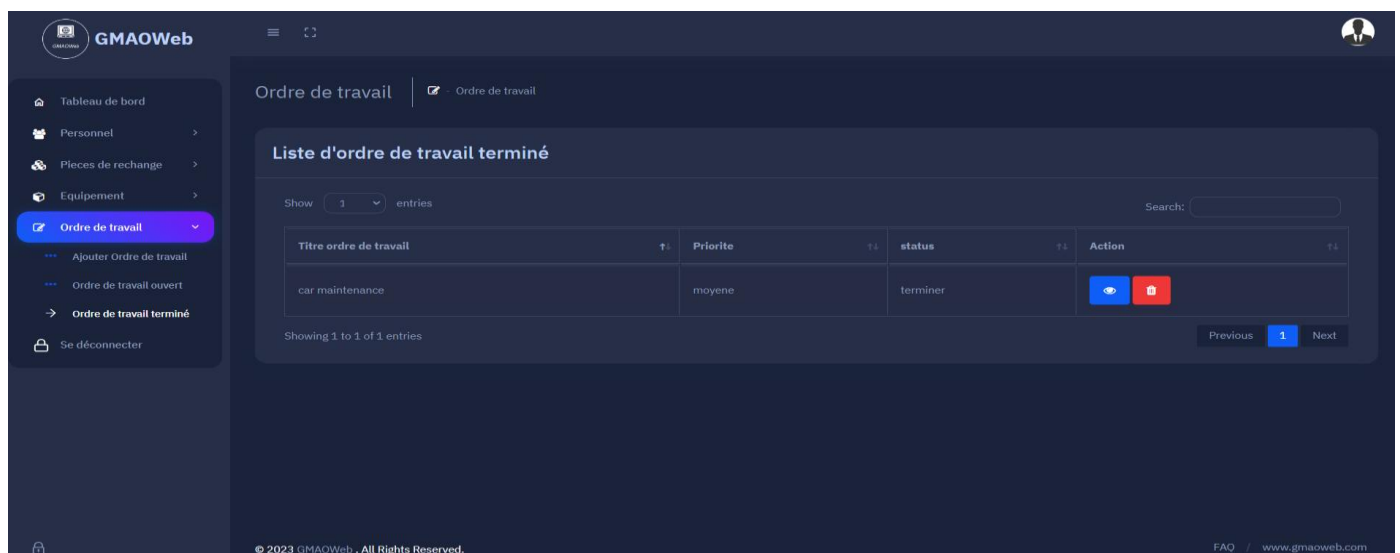


Figure 34: Interface gestion des Ordre de travail terminé

17 Conclusion

En résumé, notre application Laravel repose sur une architecture solide, avec des mesures de sécurité robustes à la fois côté back-end et côté front-end . Les interfaces de l'application offrent une expérience conviviale et permettent une gestion efficace des utilisateurs, des équipes, des équipements, des pièces de rechange et des ordres de travail.

Grâce à ces fonctionnalités, notre application est sécurisée, performante et facile à utiliser pour répondre aux besoins de nos utilisateurs.

Conclusion général

Au terme de ce stage de développement d'une application web de GMAO, nous avons pu acquérir une expérience précieuse dans la conception, la mise en œuvre et le déploiement d'un système complet de gestion des équipements, des pièces de rechange et des ordres de travail. Ce stage nous a permis de mettre en pratique nos connaissances en développement web et d'approfondir nos compétences dans le Framework Laravel.

L'objectif principal de ce projet était de fournir une solution efficace et conviviale pour la gestion des équipements et des ordres de travail au sein d'une entreprise . Grâce à l'utilisation Laravel, nous avons pu développer une application robuste, sécurisée et extensible, répondant aux besoins spécifiques du domaine de la maintenance industrielle

Bibliographie :

Les sites web

<https://laravel.sillo.org/cours-laravel-9-les-bases-installation-et-organisation/>

<https://laravel.sillo.org/cours-laravel-9-les-bases-artisan-et-les-controleurs/>

<https://laravel.sillo.org/cours-laravel-9-les-donnees-migrations-et-modeles/>

<https://getcomposer.org/doc/00-intro.md>