



Université Mohammed V de Rabat - Rabat  
École Nationale d'Informatique et d'Analyse des  
Systèmes



# RAPPORT DE STAGE DE FIN D'ANNÉE 2024/2025

FILIÈRE

GÉNIE LOGICIEL

SUJET:

---

Conception et développement d'une application web  
pour la gestion des files d'attente.

---

*Réalisé par :*

M.Ismail AHAKAY

*Encadré par :*

M. Ahmed EL HAMDI

*Jury :*

Prof. M. Azeddine ELHASSOUNY

Prof. Mme. Widad ETTAZI

Année Universitaire 2024-2025

# Remerciements

Avant tout développement sur cette expérience professionnelle, il apparaît opportun de commencer par des remerciements à ceux qui m'ont beaucoup appris au cours de cette année.

Je tiens ensuite à exprimer ma profonde gratitude envers mon encadrant de stage, **M. Ahmed El Hamdi**, pour avoir accepté de m'encadrer durant mon stage chez REDAL et pour avoir apporté toute l'aide nécessaire à l'aboutissement de ce projet.

Je remercie également toute l'équipe pédagogique de l'École Nationale Supérieure d'Informatique et d'Analyse des Systèmes (ENSIAS) pour avoir assuré une formation de haute qualité et pour m'avoir soutenu tout au long de cette année. Veuillez trouver ici le témoignage de mon respect le plus profond.

Que tous ceux qui ont contribué de près ou de loin à l'aboutissement de ce projet trouvent l'expression de mes remerciements les plus sincères.

J'ai eu l'immense plaisir de passer ma première année au sein de l'École Nationale Supérieure d'Informatique et d'Analyse des Systèmes. Cette expérience de stage restera gravée dans ma mémoire en grande partie grâce à l'encadrement attentif de **M. Ahmed El Hamdi** et au soutien continu de l'équipe pédagogique de l'ENSIAS.

# Résumé

Le présent document constitue un résumé détaillé de mon engagement dans le projet de création d'un système de gestion de fils d'attente, une initiative majeure visant à concevoir et développer une solution innovante pour l'optimisation de la gestion des files d'attente. Mon implication dans ce projet a couvert divers aspects essentiels, allant de l'analyse des besoins initiaux à la conception détaillée et à la définition des objectifs.

L'objectif principal de ce projet a été de créer un système qui simplifie et rationalise la gestion des fils d'attente, offrant une expérience fluide et efficace pour toutes les parties prenantes, des gestionnaires aux utilisateurs finaux. Le système est conçu pour répondre aux défis liés à la gestion des files d'attente en automatisant les processus, en améliorant la communication entre les gestionnaires et les usagers, et en optimisant l'utilisation des ressources disponibles.

Le projet de gestion des fils d'attente représente une fusion de l'innovation technologique et de la gestion opérationnelle, visant à offrir une solution plus efficace et équitable pour la gestion des flux de personnes. Mon engagement dans ce projet a été motivé par la conviction que la technologie peut jouer un rôle crucial dans l'amélioration de l'efficacité et du confort dans des environnements à forte affluence.

En résumé, ce document reflète mon implication dans le développement du système de gestion de fils d'attente, depuis l'analyse des besoins jusqu'à la conception détaillée et la réalisation de la solution. J'ai travaillé avec détermination pour contribuer à la mise en place de cet outil innovant, visant à simplifier et améliorer la gestion des files d'attente et à offrir de nouvelles perspectives pour l'organisation des flux de personnes.

---

**Mots clés :** Développement Web, Laravel, Vue.js, MySQL, Gestion des files d'attente

---

# Abstract

This document provides a detailed summary of my involvement in the development of a queue management system, a major initiative aimed at designing and creating an innovative solution for optimizing queue management. My involvement in this project covered various essential aspects, from initial needs analysis to detailed design and goal setting.

The primary objective of this project was to create a system that simplifies and streamlines queue management, offering a smooth and efficient experience for all stakeholders, from managers to end-users. The system is designed to address the challenges associated with queue management by automating processes, enhancing communication between managers and users, and optimizing the use of available resources.

The queue management project represents a fusion of technological innovation and operational management, aiming to provide a more effective and equitable solution for managing the flow of people. My commitment to this project was driven by the belief that technology can play a crucial role in improving efficiency and comfort in high-traffic environments.

In summary, this document reflects my involvement in the development of the queue management system, from needs analysis through detailed design and implementation. I have worked with dedication to contribute to the creation of this innovative tool, aiming to simplify and enhance queue management and to offer new perspectives on managing the flow of people.

---

**Keywords :** Web Development, Laravel, Vue.js, MySQL, Queue Management.

---

# Table des matières

|  |           |
|--|-----------|
| <b>Remerciements</b>                           | <b>1</b>  |
| <b>Résumé</b>                                  | <b>2</b>  |
| <b>Abstract</b>                                | <b>3</b>  |
| <b>Introduction générale</b>                   | <b>7</b>  |
| <b>1 Présentation générale de l'organisme</b>  | <b>8</b>  |
| 1.1 Historique de Redal . . . . .              | 8         |
| 1.2 Mission et Vision . . . . .                | 8         |
| 1.3 Organisation et Structure . . . . .        | 8         |
| 1.4 Valeurs de l'entreprise . . . . .          | 9         |
| 1.5 Impact Social et Environnemental . . . . . | 9         |
| <b>2 Contexte général du projet</b>            | <b>10</b> |
| 2.1 Le sujet et son contexte . . . . .         | 10        |
| 2.1.1 Contexte du sujet . . . . .              | 10        |
| 2.1.2 Sujet . . . . .                          | 10        |
| 2.2 Problématique . . . . .                    | 11        |
| 2.3 Solution . . . . .                         | 11        |
| 2.4 Gestion de projet . . . . .                | 11        |
| 2.5 Besoins fonctionnels . . . . .             | 11        |
| 2.6 Besoins non fonctionnels . . . . .         | 12        |
| 2.7 Objectifs . . . . .                        | 12        |
| <b>3 Conception Globale du Projet</b>          | <b>14</b> |
| 3.1 Analyse théorique et Conception . . . . .  | 14        |
| 3.1.1 Analyse Théorique . . . . .              | 14        |
| 3.2 Conception Globale . . . . .               | 15        |

|          |   |           |
|----------|---|-----------|
| 3.2.1    | Conception Globale . . . . .  | 15        |
| 3.3      | Conception Détaillée . . . . .  | 15        |
| 3.3.1    | Diagramme de Classes . . . . .  | 16        |
| 3.3.2    | Diagramme de Cas d'Utilisation Global . . . . .                       | 17        |
| 3.3.3    | Diagramme de Gantt . . . . .  | 18        |
| <b>4</b> | <b>Développement de l'Application et Performances obtenues</b>        | <b>19</b> |
| 4.1      | Langages de Programmation et Technologies Utilisées . . . . .         | 19        |
| 4.1.1    | Architecture de l'Application : . . . . .                             | 19        |
| 4.1.2    | Laravel (v11.9.1) : . . . . .   | 20        |
| 4.1.2.1  | Introduction à Laravel : . . . . .                                    | 20        |
| 4.1.2.2  | Explication du choix de Laravel : . . . . .                           | 20        |
| 4.1.2.3  | Utilisation des fonctionnalités de Laravel : . . . . .                | 20        |
| 4.1.2.4  | Avantages et bénéfices de l'utilisation de Laravel : . . . . .        | 20        |
| 4.1.3    | Vue.js (v3.2.37) : . . . . .  | 21        |
| 4.1.3.1  | Introduction à Vue.js : . . . . .                                     | 21        |
| 4.1.3.2  | Explication du choix de Vue.js : . . . . .                            | 21        |
| 4.1.3.3  | Utilisation des fonctionnalités de Vue.js : . . . . .                 | 21        |
| 4.1.3.4  | Avantages et bénéfices de l'utilisation de Vue.js : . . . . .         | 21        |
| 4.1.4    | MySQL (v8.0.37) : . . . . .   | 22        |
| 4.1.4.1  | Introduction à MySQL : . . . . .                                      | 22        |
| 4.1.4.2  | Explication du choix de MySQL : . . . . .                             | 22        |
| 4.1.4.3  | Utilisation de MySQL : . . . . .                                      | 22        |
| 4.1.4.4  | Avantages et bénéfices de l'utilisation de MySQL : . . . . .          | 22        |
| 4.1.5    | Environnement de Travail : . . . . .                                  | 22        |
| 4.2      | Accomplissement du Projet - Version Finale . . . . .                  | 23        |
| 4.2.1    | Page d'Accueil - Application de Gestion des Files d'Attente . . . . . | 23        |
| 4.2.2    | Espace Visiteur - Gestion des Files d'Attente . . . . .               | 24        |
| 4.2.2.1  | Distributeur de Tickets . . . . .                                     | 25        |
| 4.2.2.2  | Gestion des Files d'Attente . . . . .                                 | 27        |
| 4.2.3    | Espace Administrateur - Gestion des Files d'Attente . . . . .         | 27        |
| 4.2.3.1  | Tableau de Bord Administrateur . . . . .                              | 28        |
| 4.2.3.2  | Statistiques . . . . .  | 29        |
| 4.2.3.3  | Gestion du Profil . . . . .   | 30        |
| 4.2.3.4  | Paramètres . . . . .  | 31        |
| 4.2.4    | Espace Agent - Gestion des Files d'Attente . . . . .                  | 34        |
| 4.2.4.1  | Tableau de Bord Agent . . . . .                                       | 35        |

|          |  |           |
|----------|--|-----------|
| 4.2.4.2  | Statistiques . . . . .   | 36        |
| 4.2.4.3  | Profil . . . . .   | 36        |
| 4.2.4.4  | Validation des Tickets . . . . .   | 37        |
| 4.2.4.5  | Accès Restreint aux Services . . . . .                                       | 38        |
| 4.2.5    | Espace Gestionnaire des Comptes . . . . .                                    | 38        |
| 4.2.5.1  | Gestion du Profil . . . . .  | 38        |
| 4.2.5.2  | Paramètres . . . . .   | 39        |
| 4.3      | Caractéristiques de Notre Application . . . . .                              | 39        |
| 4.3.1    | Laravel pour la Gestion des Données et des Fonctionnalités Backend . . . . . | 40        |
| 4.3.1.1  | Gestion des Données avec Laravel . . . . .                                   | 40        |
| 4.3.1.2  | Authentification et Autorisation avec Laravel . . . . .                      | 40        |
| 4.3.2    | Vue.js pour une Interface Utilisateur Dynamique et Réactive . . . . .        | 40        |
| 4.3.2.1  | Composants Réactifs avec Vue.js . . . . .                                    | 40        |
| 4.3.2.2  | Vue Router pour la Navigation . . . . .                                      | 41        |
| <b>5</b> | <b>Conclusion</b>  | <b>42</b> |
| 5.1      | Bénéfices du projet de stage . . . . .                                       | 42        |
| 5.2      | Futures améliorations . . . . .  | 42        |

# Introduction générale

Dans un monde en perpétuelle évolution où la technologie façonne notre manière de vivre, d'apprendre et de travailler, l'innovation est devenue essentielle dans tous les domaines de la société, y compris la gestion des files d'attente dans divers services. Nous sommes fiers de présenter notre application web, une solution innovante conçue pour améliorer et optimiser la gestion des files d'attente.

Notre application va bien au-delà de la simple gestion des files d'attente ; elle propose une solution complète qui vise à rationaliser l'ensemble du processus de gestion des files d'attente pour différentes agences et services. Cette application offre une expérience transparente et efficace aux diverses parties prenantes, y compris les administrateurs, les gestionnaires du système, les agents et, bien sûr, les visiteurs. Depuis la prise de ticket jusqu'à la validation par les agents, notre application simplifie chaque étape du processus, permettant ainsi un déroulement fluide et une gestion précise.

Grâce à cette application novatrice, les administrateurs peuvent suivre en temps réel l'évolution des files d'attente, les gestionnaires du système disposent d'outils puissants pour gérer les profils et les paramètres, les agents peuvent valider les tickets de manière efficace et les visiteurs ont un accès convivial pour suivre leur progression et leur état dans la file d'attente. Notre application s'engage à garantir une gestion transparente, équitable et sécurisée des files d'attente, tout en offrant une expérience utilisateur de premier ordre.

En résumé, cette application représente une avancée majeure dans le domaine de la gestion des files d'attente, en plaçant la technologie au service de l'efficacité et de la satisfaction des utilisateurs. Nous sommes convaincus que cette application révolutionnaire jouera un rôle crucial dans l'amélioration de la gestion des files d'attente, permettant ainsi aux différents utilisateurs de se concentrer davantage sur leurs besoins et objectifs. Bienvenue dans une nouvelle ère de gestion intelligente des files d'attente grâce à notre application.



# Chapitre 1

## Présentation générale de l'organisme



### 1.1 Historique de Redal

Redal a été fondée en 1998 dans le but de fournir des services d'eau potable, d'assainissement et d'électricité à la région de Rabat-Salé au Maroc. Depuis sa création, l'entreprise s'est engagée à garantir une qualité de service exemplaire pour ses clients tout en respectant l'environnement.

### 1.2 Mission et Vision

La mission de Redal est de gérer les services publics essentiels de manière durable, en assurant un approvisionnement continu et sécurisé en eau potable, en électricité, ainsi que l'évacuation et le traitement des eaux usées. Redal aspire à devenir un leader dans le secteur des services publics au Maroc, en adoptant des technologies innovantes et en favorisant la satisfaction de ses clients.

### 1.3 Organisation et Structure

Redal est structurée en plusieurs départements clés, chacun ayant des responsabilités spécifiques :

- **Direction Générale** : Supervise l'ensemble des opérations et définit les orientations stratégiques de l'entreprise.
- **Département Eau Potable** : Responsable de la production, du traitement et de la distribution de l'eau potable.
- **Département Assainissement** : Gère le réseau d'assainissement, y compris la collecte et le traitement des eaux usées.
- **Département Électricité** : S'occupe de la distribution de l'électricité et de la maintenance des infrastructures électriques.
- **Département Commercial et Relation Client** : Assure le service client, la facturation, et le suivi des abonnements.

## 1.4 Valeurs de l'entreprise

Redal s'engage à respecter un ensemble de valeurs qui guident toutes ses actions :

- **Responsabilité** : Assurer un service public de qualité avec une grande rigueur professionnelle.
- **Innovation** : Adopter les dernières technologies pour améliorer les services offerts.
- **Transparence** : Maintenir une communication ouverte et honnête avec les parties prenantes.
- **Respect de l'environnement** : Mettre en œuvre des pratiques durables pour minimiser l'impact environnemental.

## 1.5 Impact Social et Environnemental

Redal joue un rôle crucial dans le développement social et économique de la région de Rabat-Salé. En plus de fournir des services essentiels, l'entreprise participe activement à des initiatives environnementales, telles que la gestion des ressources en eau et la promotion de l'efficacité énergétique.

## Chapitre 2

# Contexte général du projet

Dans ce chapitre, nous présentons le contexte général du projet, en abordant le sujet et son contexte, la problématique à résoudre, la solution envisagée, la gestion des files d'attente mise en place, ainsi que les besoins fonctionnels, les besoins non fonctionnels et les objectifs du projet.

### 2.1 Le sujet et son contexte

#### 2.1.1 Contexte du sujet

Dans le contexte actuel où la gestion efficace du temps et des ressources est cruciale, la gestion des files d'attente représente un défi majeur pour de nombreux services publics et privés. La lenteur et l'inefficacité des files d'attente peuvent entraîner une frustration accrue des clients, une perte de productivité, et une mauvaise image pour l'organisation. Face à ces défis, notre projet vise à développer une solution innovante pour la gestion des files d'attente, permettant une amélioration significative de l'expérience utilisateur et de l'efficacité opérationnelle.

Notre application de gestion des files d'attente s'adresse à divers secteurs, tels que les services administratifs, les agences de santé, les banques, et bien d'autres. Elle permet de réduire les temps d'attente, d'optimiser la répartition des ressources, et de garantir une transparence et une équité dans le traitement des utilisateurs.

#### 2.1.2 Sujet

Le projet consiste en la conception et le développement d'une application web pour la gestion des files d'attente. Grâce à cette application, les administrateurs peuvent suivre en temps réel l'évolution des files d'attente, les gestionnaires du système disposent d'outils puissants pour gérer les profils et les paramètres, les agents peuvent valider les tickets de manière efficace, et les visiteurs ont un accès convivial pour suivre leur progression dans la file d'attente. L'application s'engage à garantir une gestion transparente, équitable et sécurisée des files d'attente, tout en offrant une expérience utilisateur de premier ordre.

## 2.2 Problématique

Comment améliorer la gestion des files d'attente pour rendre un processus administratif complexe plus simple et moderne, tout en garantissant une gestion claire, juste et sécurisée pour tous ?

## 2.3 Solution

Le projet propose une solution novatrice qui simplifie et optimise la gestion des files d'attente à toutes les étapes. Grâce à cette application, les administrateurs ont un suivi en temps réel des files d'attente, les gestionnaires du système disposent d'outils puissants pour gérer efficacement les données et les processus, les agents peuvent valider les tickets de manière efficiente, et les visiteurs bénéficient d'un accès convivial pour suivre leur progression dans la file d'attente. L'application s'engage à garantir une gestion transparente, équitable et sécurisée des files d'attente, offrant ainsi une expérience utilisateur de premier ordre pour toutes les parties prenantes.

## 2.4 Gestion de projet

La réalisation se compose essentiellement de 3 parties :

- Analyse des besoins
- Conception du projet
- Développement (codage)

## 2.5 Besoins fonctionnels

Voici les principaux besoins fonctionnels identifiés pour l'application de gestion des files d'attente :

1. **Consultation des informations générales** : Les utilisateurs doivent pouvoir accéder à des informations détaillées sur la gestion des files d'attente via l'application.
2. **Prise de ticket** : Les visiteurs doivent avoir la possibilité de prendre un ticket via l'application pour accéder aux services.
3. **Suivi des files d'attente** : Les visiteurs doivent être en mesure de suivre l'évolution de la file d'attente correspondant au service, à travers une page dédiée de l'application.
4. **Validation des tickets** : Les agents doivent pouvoir valider les tickets des visiteurs, garantissant ainsi un service efficace et rapide.
5. **Gestion du système** : Les administrateurs doivent pouvoir gérer tout le système, ce qui comprend la création des agences et services, la mise à jour des informations générales liées aux services et agences, et la gestion des utilisateurs.

6. **Administration des comptes utilisateurs** : Les gestionnaires de comptes doivent pouvoir gérer les comptes des utilisateurs, y compris la création, la modification et la suppression de comptes.

## 2.6 Besoins non fonctionnels

Voici les principaux besoins non fonctionnels identifiés pour l'application de gestion des files d'attente :

- **Sécurité** : L'application doit garantir la confidentialité et l'intégrité des données. Des mécanismes d'authentification solides et des contrôles d'accès appropriés doivent être mis en place pour assurer la sécurité des informations.
- **Performance** : L'application doit être réactive et offrir une performance optimale. Les requêtes doivent être traitées rapidement, et les technologies utilisées doivent être efficaces pour assurer une utilisation fluide de l'application.
- **Convivialité** : L'interface utilisateur doit être conviviale et intuitive. La navigation doit être fluide, et les fonctionnalités de l'application doivent être clairement présentées pour faciliter son utilisation par les utilisateurs.
- **Extensibilité** : Il est important de concevoir l'application de manière à ce qu'elle puisse être étendue avec de nouvelles fonctionnalités à l'avenir. Cela permettra de s'adapter aux évolutions et aux besoins changeants des utilisateurs.
- **Portabilité** : L'application doit être compatible avec différentes plateformes et navigateurs web. Elle doit fonctionner de manière optimale sur différents dispositifs, tels que les ordinateurs de bureau, les smartphones et les tablettes.
- **Maintenance** : Il est essentiel de faciliter la maintenance de l'application. Pour cela, il convient d'utiliser des technologies robustes, de suivre les meilleures pratiques de développement et de documenter soigneusement le code source afin de permettre une maintenance efficace et rapide de l'application.

## 2.7 Objectifs

Les objectifs de notre projet sont les suivants :

1. Simplifier et automatiser la gestion des files d'attente.
2. Offrir une plateforme centralisée pour la coordination et la gestion des files d'attente.
3. Faciliter la consultation des informations générales sur la gestion des files d'attente au sein de l'espace public de l'application.
4. Permettre aux visiteurs de prendre un ticket et de suivre leur progression dans la file d'attente.

5. Mettre en place un espace privé avec des niveaux d'accès différents pour les administrateurs, les agents et les visiteurs.
6. Simplifier les tâches administratives en permettant aux administrateurs de gérer les comptes utilisateurs.
7. Favoriser une meilleure coordination entre les parties prenantes, en facilitant la gestion des tickets et des files d'attente.
8. Assurer la sécurité, la performance et la convivialité de l'application pour une expérience utilisateur optimale.

Ces objectifs visent à simplifier la gestion des files d'attente, à améliorer la communication et la coordination entre les acteurs, tout en offrant une expérience utilisateur fluide et efficace.

## Chapitre 3

# Conception Globale du Projet

Dans cette section, nous allons présenter la conception globale de notre projet, en mettant en évidence les différentes étapes et les choix techniques que nous avons effectués.

### 3.1 Analyse théorique et Conception

#### 3.1.1 Analyse Théorique

L'analyse théorique constitue la première étape de notre projet de gestion de fils d'attente, où nous approfondissons notre compréhension du sujet. Cette phase est cruciale car elle nous permet de définir les objectifs, les besoins et les contraintes de notre système de gestion des files d'attente.

Au cours de cette analyse, nous avons examiné les différents aspects liés à la gestion des flux de personnes. Nous avons étudié les processus et les flux de travail existants, identifié les principales lacunes et les problèmes rencontrés, et analysé les attentes des utilisateurs finaux, y compris les gestionnaires, les employés de service et les usagers.

De plus, nous avons réalisé une étude approfondie des fonctionnalités nécessaires pour répondre aux besoins spécifiques de notre système de gestion des fils d'attente. Nous avons également pris en compte les aspects techniques tels que la compatibilité avec les différentes plateformes, la sécurité des données et l'évolutivité du système.

Cette analyse théorique nous a permis d'établir une base solide pour la conception et le développement de notre solution. Nous avons pu définir les principales fonctionnalités, les interfaces utilisateur, les flux de travail et les cas d'utilisation clés. En comprenant profondément notre sujet, nous sommes en mesure de concevoir un système qui répondra de manière efficace et efficiente aux besoins et aux attentes des utilisateurs finaux dans la gestion des files d'attente.

## 3.2 Conception Globale

Après avoir minutieusement analysé les besoins et les attentes de notre projet de gestion de fils d'attente, nous abordons dans ce chapitre une phase cruciale : la conception. Cette étape est déterminante pour le succès du projet, car elle vise à établir une structure solide pour notre système. Notre objectif principal est de définir avec précision les tâches à accomplir et de préparer le terrain pour la phase de réalisation.

Dans cette première partie, nous nous concentrons sur la conception globale de notre système. Nous définissons les grandes lignes de son architecture, en mettant l'accent sur les éléments clés qui la composeront. Cette vue d'ensemble nous permettra de garantir une cohérence globale dans notre approche.

Par la suite, dans la deuxième partie, nous approfondirons en utilisant les diagrammes UML appropriés. Ces diagrammes nous permettront de représenter de manière visuelle et précise la structure et les interactions au sein de notre système. Ils serviront de guide précieux pour les développeurs lors de la phase de mise en œuvre.

La phase de conception marque un pas important vers la concrétisation de notre projet de gestion de fils d'attente. Elle nous permet de transformer les besoins en une structure claire et cohérente, posant ainsi les bases d'un système performant et fonctionnel.

### 3.2.1 Conception Globale

Notre projet de gestion de fils d'attente vise à développer un système capable d'optimiser et de gérer efficacement les files d'attente dans divers environnements. Ce système adopte une architecture à trois niveaux, comprenant :

- Un utilisateur, où l'ordinateur demande des ressources et est équipé d'une interface utilisateur, généralement sous la forme d'une application web ou d'une interface mobile.
- Un serveur, chargé de fournir les ressources et de traiter les demandes des clients. Ce serveur peut également faire appel à d'autres services si nécessaire.
- Une base de données, qui stocke et gère les informations relatives aux files d'attente, aux utilisateurs et aux transactions.

Cette architecture spécifique guide notre conception pour garantir un fonctionnement fluide et une expérience utilisateur optimale dans le contexte de notre projet de gestion de fils d'attente.

## 3.3 Conception Détaillée

Dans cette phase, nous abordons la conception approfondie de notre projet de gestion des files d'attente. Nous présentons des diagrammes de classes, des cas d'utilisation et un diagramme de Gantt. Ces éléments illustreront avec précision la structure et le fonctionnement de notre application.



### 3.3.1 Diagramme de Classes

Le diagramme de classes fournit une vue d'ensemble du système en montrant ses classes, interfaces, collaborations et leurs relations. Il constitue le cœur de l'architecture et illustre les principales entités de notre projet. La figure 3.1 présente le diagramme de classes de notre projet.

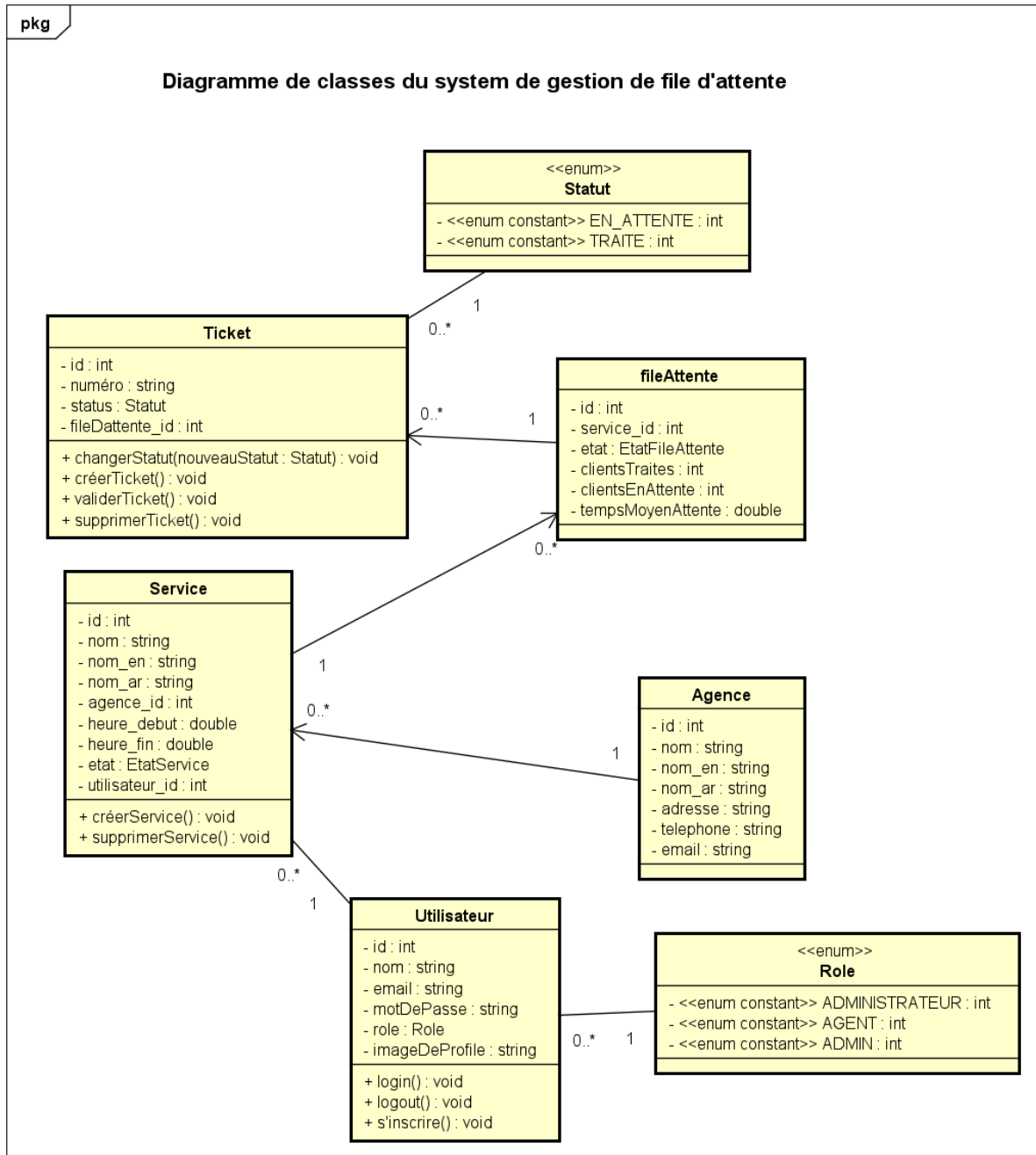


FIGURE 3.1 – Diagramme de classes

### 3.3.2 Diagramme de Cas d'Utilisation Global

Après avoir défini la structure générale du système, nous présentons les cas d'utilisation, qui montrent les interactions possibles entre les utilisateurs et l'application. Chaque cas d'utilisation identifie un scénario d'utilisation correspondant aux objectifs des utilisateurs et aux besoins du système. Les figures ci-dessous offrent une vue d'ensemble des cas d'utilisation internes et externes.

Le diagramme est divisé en deux parties principales : les cas d'utilisation internes, qui concernent les processus internes de l'entreprise, et les cas d'utilisation externes, qui traitent des interactions avec les clients.

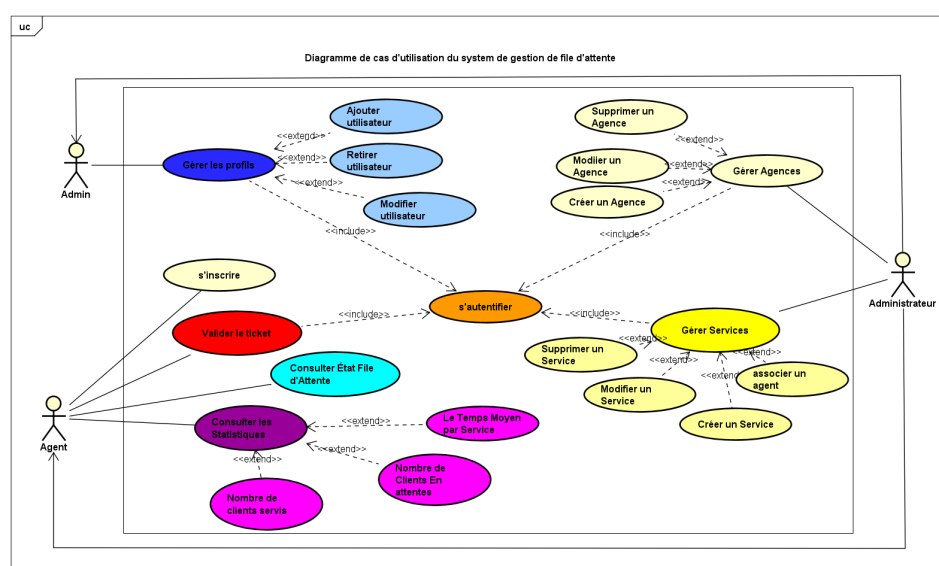


FIGURE 3.2 – Diagramme des cas d'utilisation internes

La figure 3.2 englobe les opérations liées à la gestion des services, à la supervision des agents et à l'administration générale. Les utilisateurs internes peuvent accéder aux fonctionnalités de gestion, de suivi des files d'attente et de visualisation des statistiques.

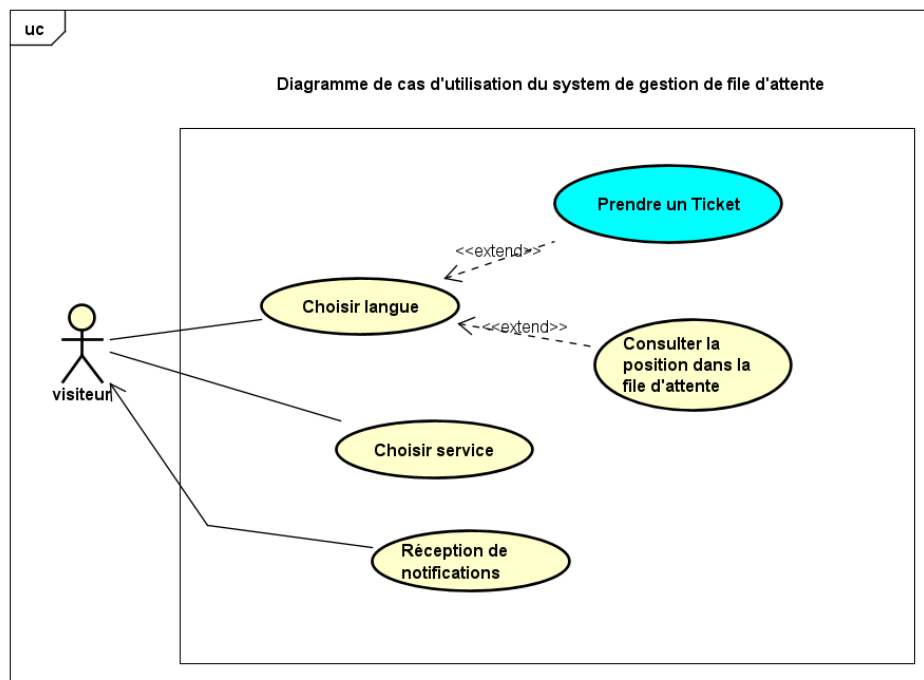


FIGURE 3.3 – Diagramme des cas d'utilisation externes

La figure 3.3 se concentre sur les interactions avec les clients, incluant la prise de tickets, la consultation de l'état des files d'attente et la réception de notifications.

### 3.3.3 Diagramme de Gantt

Enfin, le diagramme de Gantt ci-dessous présente le calendrier de notre projet de gestion des files d'attente. Il illustre les phases du projet, les tâches associées ainsi que les dates de début et de fin prévues, facilitant ainsi le suivi des étapes du projet.

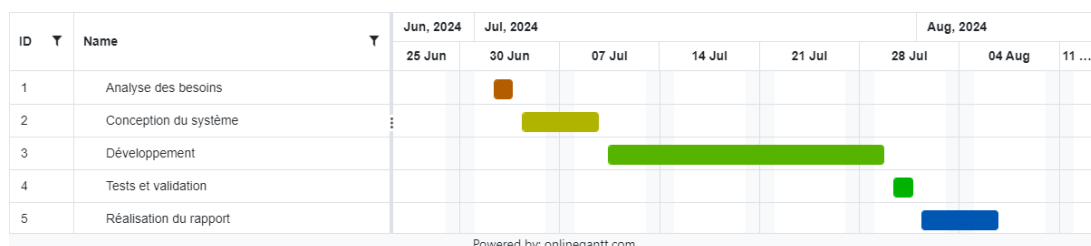


FIGURE 3.4 – Diagramme de Gantt du projet

La figure 3.4 montre la répartition des tâches et leur calendrier, permettant de suivre les progrès du projet et de s'assurer que les délais sont respectés.

## Chapitre 4

# Développement de l'Application et Performances obtenues

Dans cette section, nous allons discuter de l'implémentation de notre projet de gestion des files d'attente et présenter le résultat final à l'aide de captures d'écran.

### 4.1 Langages de Programmation et Technologies Utilisées

Pour mon projet de gestion des files d'attente, j'ai adopté une architecture monolithique avec Laravel pour le backend, Vue.js pour le frontend, et MySQL pour la base de données. Cette décision stratégique vise à intégrer efficacement la logique métier, l'interface utilisateur et la gestion des données au sein d'un environnement de développement cohérent et performant.

#### 4.1.1 Architecture de l'Application :

L'architecture de notre application se compose des parties suivantes :

- Backend (Laravel) : Gère la logique métier, les API et la gestion des files d'attente.
- Frontend (Vue.js) : Fournit une interface utilisateur interactive et réactive, intégrant les fonctionnalités du backend.
- Base de Données (MySQL) : Stocke les données des files d'attente, les informations utilisateur, ainsi que les configurations de l'application.

Cette structure monolithique assure une intégration fluide entre les différentes composantes de l'application, facilitant le développement et la maintenance.

### 4.1.2 Laravel (v11.9.1) :

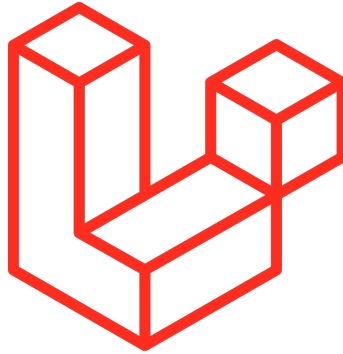


FIGURE 4.1 – Logo de Laravel.

#### 4.1.2.1 Introduction à Laravel :

Laravel est un framework PHP moderne et puissant qui simplifie le développement rapide d'applications web sécurisées et évolutives. J'ai choisi Laravel (version 11.9.1) pour sa robustesse, sa facilité d'utilisation et sa large communauté de soutien, en faisant ainsi un choix optimal pour mon projet de gestion des files d'attente.

#### 4.1.2.2 Explication du choix de Laravel :

J'ai opté pour Laravel en raison de sa capacité à simplifier le développement backend tout en offrant des fonctionnalités avancées telles que la gestion des routes, l'ORM Eloquent pour la manipulation des bases de données, la gestion des tâches avec Laravel Queues, et la sécurité intégrée.

#### 4.1.2.3 Utilisation des fonctionnalités de Laravel :

Dans mon projet, j'ai tiré parti des capacités de Laravel pour créer une API robuste pour la gestion des files d'attente, intégrer des bibliothèques tierces via Composer, et gérer efficacement les sessions utilisateur ainsi que l'authentification.

#### 4.1.2.4 Avantages et bénéfices de l'utilisation de Laravel :

L'utilisation de Laravel m'a permis de réduire significativement le temps de développement, d'améliorer la sécurité de mon application, et de maintenir une structure de code claire et maintenable, facilitant ainsi mon travail en solo.

### 4.1.3 Vue.js (v3.2.37) :



FIGURE 4.2 – Logo de Vue.js.

#### 4.1.3.1 Introduction à Vue.js :

Vue.js est un framework JavaScript progressif conçu pour construire des interfaces utilisateur modernes et réactives. J'ai choisi Vue.js (version 3.2.37) en raison de sa flexibilité, de sa facilité d'intégration avec d'autres technologies, et de sa capacité à gérer efficacement les composants réutilisables.

#### 4.1.3.2 Explication du choix de Vue.js :

Mon choix de Vue.js s'explique par sa simplicité de mise en œuvre, sa documentation exhaustive, son approche progressive pour la construction d'applications, ainsi que sa capacité à optimiser les performances frontend.

#### 4.1.3.3 Utilisation des fonctionnalités de Vue.js :

Dans mon projet, j'ai exploité les fonctionnalités avancées de Vue.js pour créer une interface utilisateur interactive et réactive. J'ai intégré des composants Vue pour modulariser mon application, gérer l'état global avec Vuex, et optimiser la navigation avec Vue Router.

#### 4.1.3.4 Avantages et bénéfices de l'utilisation de Vue.js :

L'utilisation de Vue.js a simplifié le développement frontend, amélioré l'expérience utilisateur, et facilité la maintenance de mon application en permettant une séparation claire entre le frontend et le backend.

#### 4.1.4 MySQL (v8.0.37) :



FIGURE 4.3 – Logo de MySQL.

##### 4.1.4.1 Introduction à MySQL :

MySQL est un système de gestion de base de données relationnelle largement utilisé pour le stockage et la récupération de données. J'ai choisi MySQL (version 8.0.37) pour sa fiabilité, sa performance et sa compatibilité avec les applications web modernes.

##### 4.1.4.2 Explication du choix de MySQL :

Le choix de MySQL pour mon projet de gestion des files d'attente repose sur sa capacité à gérer efficacement les transactions, à fournir des fonctionnalités avancées telles que les index et les procédures stockées, ainsi que sa compatibilité avec les outils de développement PHP comme Laravel.

##### 4.1.4.3 Utilisation de MySQL :

Dans mon projet, MySQL est utilisé pour stocker les données des files d'attente, les informations utilisateur, ainsi que les configurations de l'application. J'ai conçu le schéma de base de données pour optimiser les requêtes et assurer la cohérence des données.

##### 4.1.4.4 Avantages et bénéfices de l'utilisation de MySQL :

L'utilisation de MySQL m'a permis de bénéficier d'une gestion efficace des données, d'une performance optimale pour les requêtes complexes, et d'une intégration transparente avec ma stack technologique existante.

#### 4.1.5 Environnement de Travail :

Pour le développement de mon application de gestion des files d'attente, j'ai utilisé un ensemble d'outils et de pratiques pour assurer un environnement de travail efficace :

- Environnement de Développement : J'ai utilisé Visual Studio Code comme principal éditeur de code, bénéficiant de ses extensions pour Laravel et Vue.js qui facilitent le développement et le débogage.
- Gestion de Version : Git a été utilisé pour le contrôle de version de mon code source, permettant une gestion efficace des modifications et des collaborations avec moi-même.

Cette approche intégrée m'a permis de créer une application web moderne, robuste et évolutive pour la gestion des files d'attente, répondant ainsi à mes exigences tout en offrant une expérience utilisateur optimale.

## 4.2 Accomplissement du Projet - Version Finale

### 4.2.1 Page d'Accueil - Application de Gestion des Files d'Attente

Dans cette section, nous présentons la page d'accueil de notre application dédiée à la gestion des files d'attente, avec un focus particulier sur les utilisateurs.

**Introduction à la Page d'Accueil** La page d'accueil de notre application est conçue pour offrir une vue d'ensemble claire et intuitive des fonctionnalités disponibles. Bien que l'application soit principalement dédiée aux agents et administrateurs, certaines fonctionnalités importantes sont destinées aux visiteurs, mais leur accès est conditionné par une configuration préalable par les utilisateurs.

**Page d'Accueil Dédiée aux Utilisateurs** La page d'accueil est spécifiquement conçue pour les utilisateurs de l'application, leur offrant un accès facile aux fonctionnalités principales.

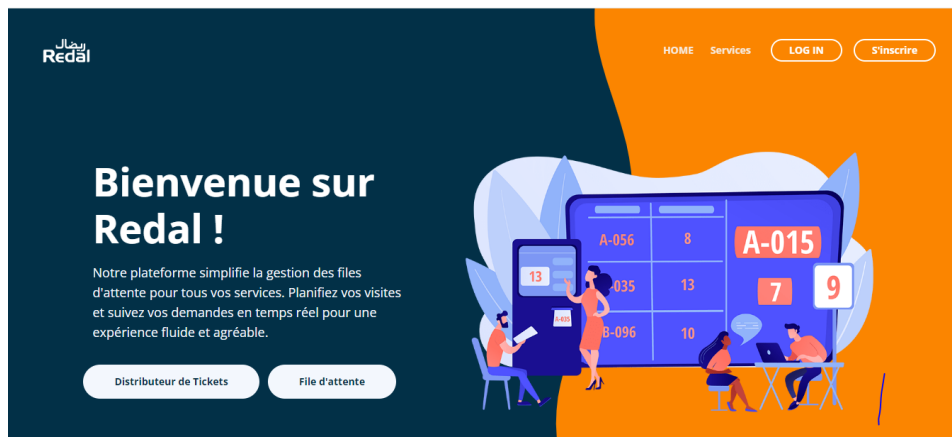


FIGURE 4.4 – Page d'accueil

**Description de l'Application** La section "Description" de notre page d'accueil offre une vue détaillée de notre système de gestion des files d'attente. Elle met en lumière les fonctionnalités clés qui simplifient la gestion des files d'attente pour améliorer l'expérience client.





FIGURE 4.5 – Section Description de l'application.

**Services Offerts** La section "Services" présente une gamme complète de services dédiés à la gestion des files d'attente. Elle est conçue pour répondre aux besoins des clients en améliorant l'efficacité et en optimisant l'organisation des files d'attente.

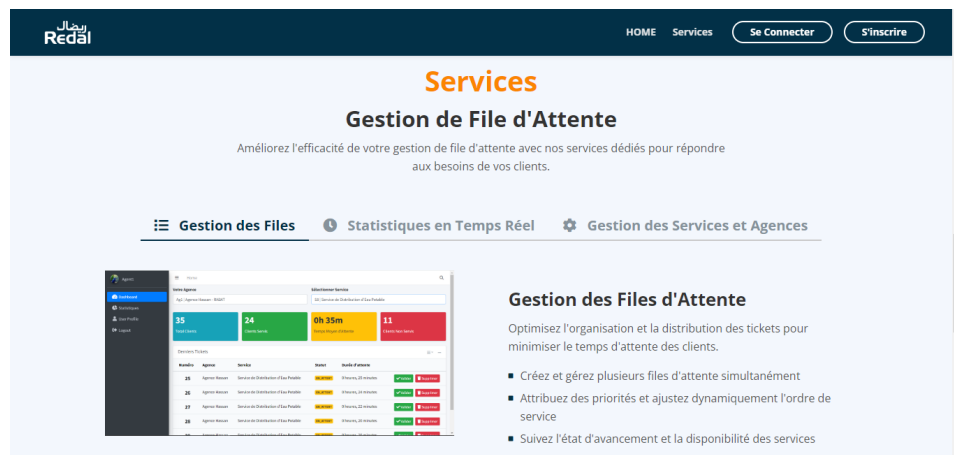


FIGURE 4.6 – Section Services présentant les solutions de gestion des files d'attente.

En résumé, la page d'accueil de l'application est conçue pour offrir une expérience fluide et accessible aux utilisateurs, tout en fournissant une description détaillée des fonctionnalités et services pour une gestion efficace des files d'attente.

#### 4.2.2 Espace Visiteur - Gestion des Files d'Attente

Dans cette section, nous mettons en avant l'espace réservé aux visiteurs dans notre application de gestion des files d'attente. Cet espace permet aux visiteurs de prendre leur ticket et de suivre l'état des files d'attente pour les différents services de l'agence.

#### 4.2.2.1 Distributeur de Tickets

Le distributeur de tickets permet aux visiteurs d'obtenir un ticket pour accéder aux services de l'agence. Les étapes pour utiliser le distributeur de tickets sont les suivantes :

Sélectionner la Langue :

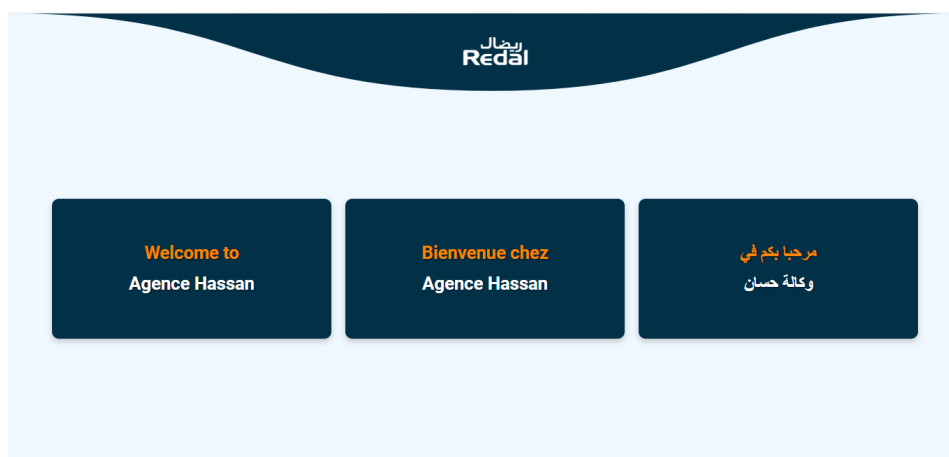


FIGURE 4.7 – Sélection de la langue dans le distributeur de tickets.

Sélectionner le Service :



FIGURE 4.8 – Sélection du service dans le distributeur de tickets.

Voir les Statistiques :

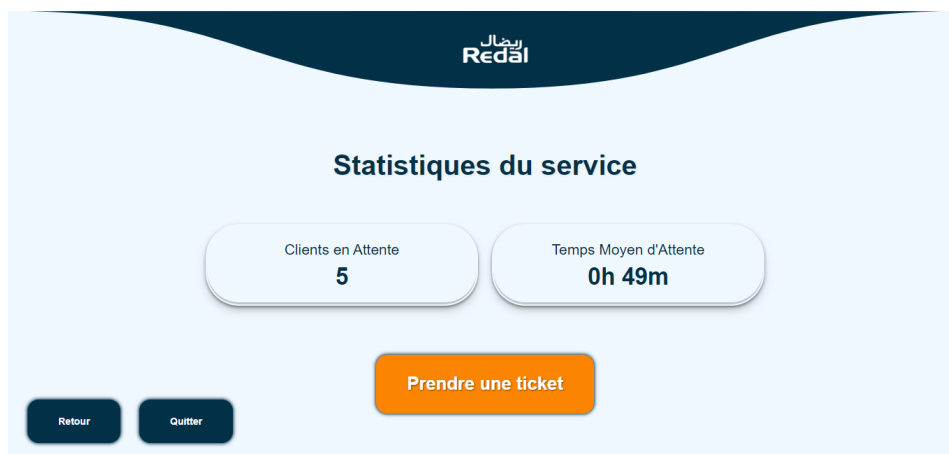


FIGURE 4.9 – Affichage des statistiques des clients en attente et du temps moyen d'attente.

Prendre un Ticket :



FIGURE 4.10 – Prise de ticket dans le distributeur de tickets.

Les visiteurs accèdent au distributeur de tickets après une configuration préalable par les utilisateurs. Une fois cette configuration terminée, les visiteurs peuvent obtenir un ticket en suivant ces étapes.

#### 4.2.2.2 Gestion des Files d'Attente



FIGURE 4.11 – Gestion des Files d'Attente pour les visiteurs.

La gestion des files d'attente permet aux visiteurs de visualiser les tickets actuels pour chaque service de l'agence. Cette fonctionnalité inclut :

**Visualisation des Tickets Actuels pour Chaque Service :** Les visiteurs peuvent consulter les tickets en cours pour chaque service disponible. Cette page leur permet de suivre le statut des files d'attente de chaque service en temps réel, facilitant ainsi leur planification et gestion du temps. Cette fonctionnalité est accessible après la configuration préalable par les utilisateurs, offrant aux visiteurs un aperçu clair de l'état des files d'attente pour les services disponibles.

#### 4.2.3 Espace Administrateur - Gestion des Files d'Attente

Dans cette section, nous présentons l'espace dédié aux administrateurs dans notre application de gestion des files d'attente. Cet espace permet aux administrateurs de gérer et de superviser les différents aspects du système, y compris la configuration des agences et services, la gestion des utilisateurs, et la visualisation des statistiques.

L'espace administrateur constitue un élément clé de notre application, offrant aux administrateurs un contrôle complet sur les différentes fonctionnalités et ressources disponibles pour la gestion des files d'attente. Les administrateurs disposent de privilèges étendus pour effectuer des tâches telles que la configuration des agences et services, la gestion des utilisateurs, et l'analyse des statistiques de performance.

Les sections suivantes détaillent les différentes fonctionnalités disponibles dans l'espace administrateur, en mettant en avant les étapes clés et les interactions possibles.

#### 4.2.3.1 Tableau de Bord Administrateur

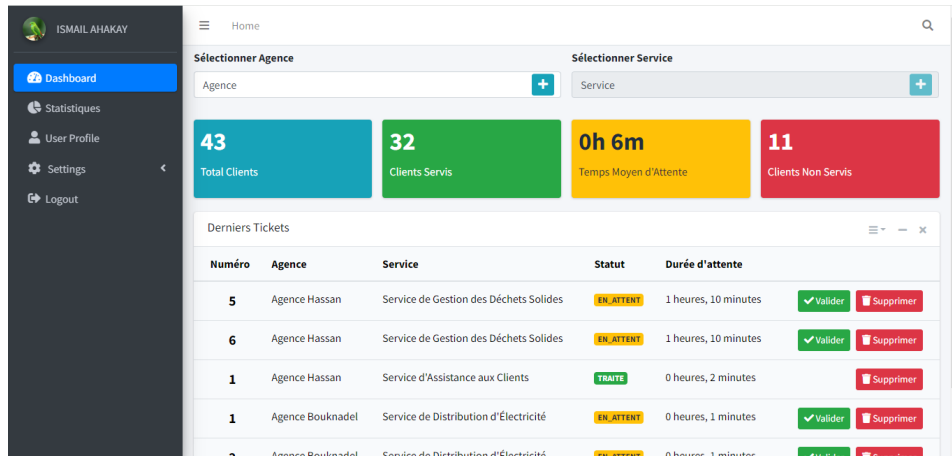


FIGURE 4.12 – Tableau de Bord Administrateur.

Le tableau de bord administrateur fournit une vue d'ensemble interactive pour la gestion des agences et des services. Il permet aux administrateurs de visualiser les statistiques et les tickets associés en fonction des sélections effectuées :

**Sélectionner une Agence :** Les administrateurs peuvent choisir une agence à partir d'un menu déroulant. La sélection d'une agence affiche les statistiques générales de cette agence ainsi que les tickets associés à tous les services de l'agence. Si aucune agence n'est sélectionnée, les statistiques et les tickets peuvent être affichés de manière globale.

**Sélectionner un Service :** Après avoir choisi une agence, les administrateurs peuvent sélectionner un service spécifique. Les statistiques et les tickets sont alors filtrés pour ne montrer que ceux liés au service sélectionné au sein de l'agence choisie. En l'absence de sélection, les informations affichées resteront basées sur la sélection précédente ou globales.

**Affichage des Statistiques et des Tickets :** Les statistiques clés, telles que le nombre de clients en attente et les temps d'attente moyens, ainsi que les tickets associés, sont présentés en fonction de la sélection actuelle. Ces affichages permettent aux administrateurs de suivre et de gérer les performances des agences et des services de manière détaillée et en temps réel.

### 4.2.3.2 Statistiques

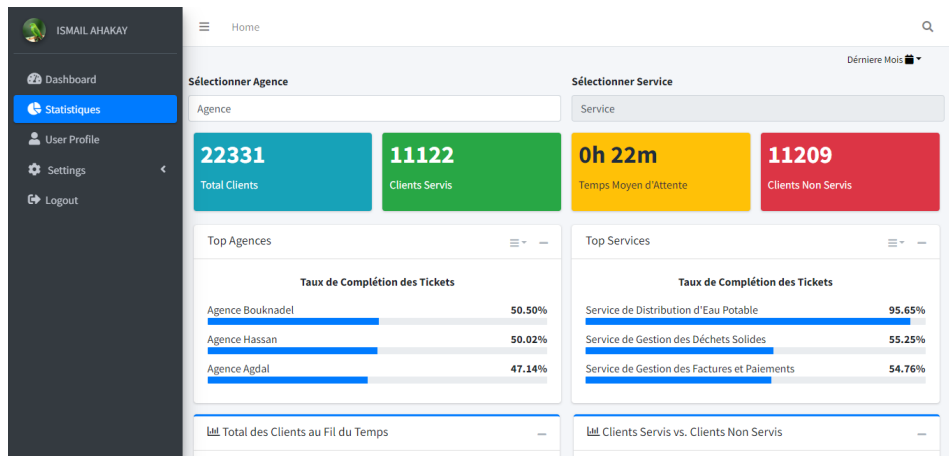


FIGURE 4.13 – Sélection des agences et services, Statbox et comparaison entre agences et services.

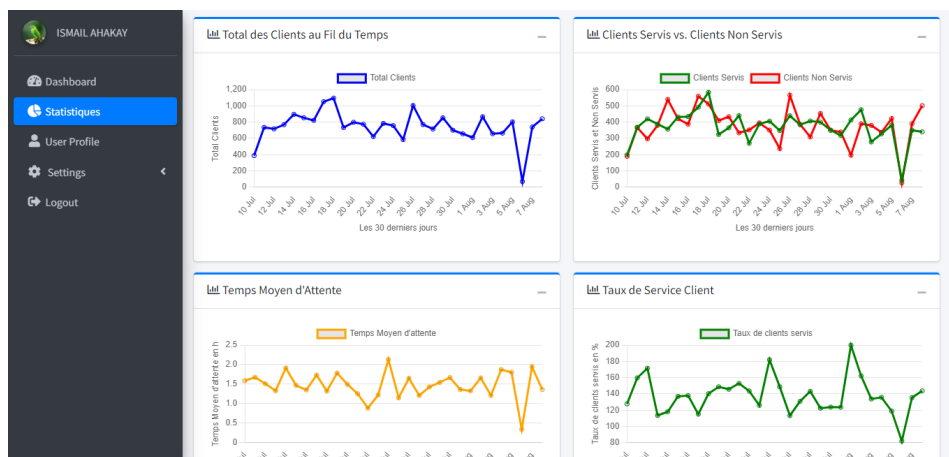


FIGURE 4.14 – Différentes courbes en graphique linéaire.

La section des statistiques permet aux administrateurs d'analyser en profondeur les données relatives aux agences et aux services. Cette section inclut :

**Analyse des Performances :** Visualiser des rapports détaillés sur les performances des agences et des services, y compris les temps d'attente moyens, le nombre de clients servis, et les tendances des visites.

- Total Clients : Affiche le nombre total de clients, incluant les clients servis et non servis.
- Clients Servis : Affiche le nombre de clients qui ont été servis.
- Temps Moyen d'Attente : Affiche le temps moyen d'attente pour les clients.
- Clients Non Servis : Affiche le nombre de clients qui n'ont pas été servis.

**Rapports Personnalisés :** Générer et consulter des rapports personnalisés en fonction des critères sélectionnés, tels que des périodes spécifiques ou des services particuliers.

- Filtrage par Période : Options de filtrage par jour, semaine, mois, ou année.
- Top Agences : Affiche les agences avec les meilleures performances.
- Top Services : Affiche les services les plus performants.

**Comparaison et Évaluation :** Comparer les performances entre différentes agences ou services pour identifier les points forts et les axes d'amélioration.

- Total des Clients au Fil du Temps : Graphique montrant l'évolution du nombre total de clients au fil du temps.
- Clients Servis vs. Clients Non Servis : Comparaison entre les clients servis et non servis.
- Temps Moyen d'Attente : Graphique montrant l'évolution du temps moyen d'attente.
- Taux de Service Client : Mesure de l'efficacité du service client.

Cette section offre une vue complète des performances du système, aidant les administrateurs à prendre des décisions informées pour optimiser le service. Elle permet de suivre les indicateurs clés en temps réel et d'analyser les données historiques pour améliorer l'efficacité et la qualité du service.

#### 4.2.3.3 Gestion du Profil

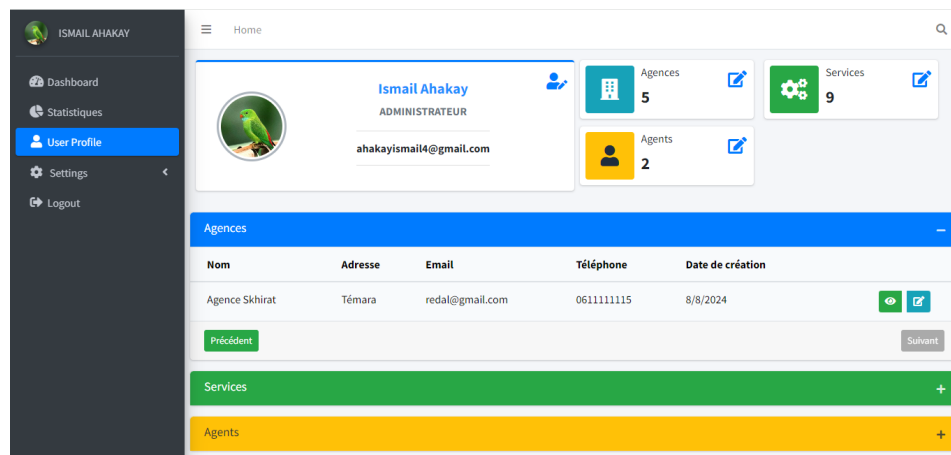
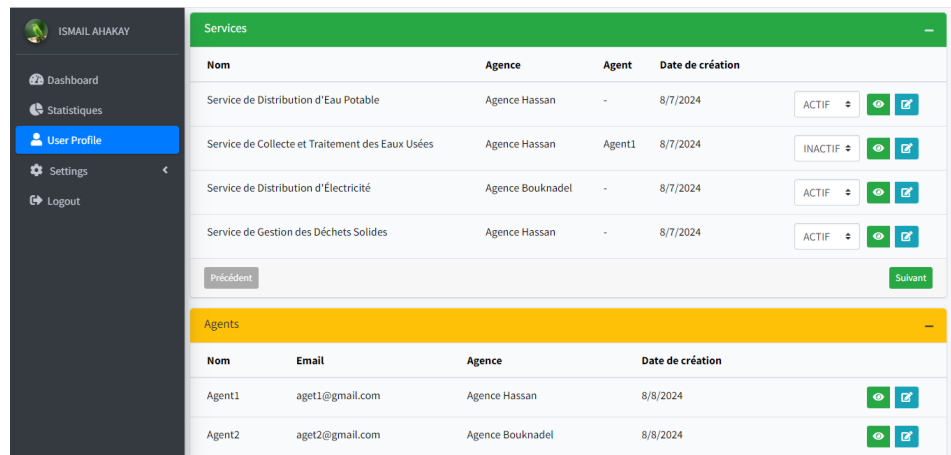


FIGURE 4.15 – Aperçu de la page de profil.



The screenshot shows a user profile page for 'ISMAIL AHAKAY'. The left sidebar contains navigation links: Dashboard, Statistiques, User Profile (highlighted), Settings, and Logout. The main content area is divided into two sections: 'Services' and 'Agents'.

**Services Table:**

| Nom  | Agence           | Agent  | Date de création | Statut  | Actions                             |
|--|------------------|--------|------------------|---------|-------------------------------------|
| Service de Distribution d'Eau Potable            | Agence Hassan    | -      | 8/7/2024         | ACTIF   | <a href="#">👁</a> <a href="#">✎</a> |
| Service de Collecte et Traitement des Eaux Usées | Agence Hassan    | Agent1 | 8/7/2024         | INACTIF | <a href="#">👁</a> <a href="#">✎</a> |
| Service de Distribution d'Électricité            | Agence Bouknadel | -      | 8/7/2024         | ACTIF   | <a href="#">👁</a> <a href="#">✎</a> |
| Service de Gestion des Déchets Solides           | Agence Hassan    | -      | 8/7/2024         | ACTIF   | <a href="#">👁</a> <a href="#">✎</a> |

Navigation: Précédent | Suivant

**Agents Table:**

| Nom    | Email           | Agence           | Date de création | Actions                             |
|--------|-----------------|------------------|------------------|-------------------------------------|
| Agent1 | aget1@gmail.com | Agence Hassan    | 8/8/2024         | <a href="#">👁</a> <a href="#">✎</a> |
| Agent2 | aget2@gmail.com | Agence Bouknadel | 8/8/2024         | <a href="#">👁</a> <a href="#">✎</a> |

FIGURE 4.16 – Aperçu de la page de profil.

La section de gestion du profil permet aux administrateurs de mettre à jour et de personnaliser leurs informations personnelles. Cette section inclut :

**Modification des Informations :** Mettre à jour les informations personnelles telles que le nom, l'adresse e-mail, et le mot de passe.

**Visualisation et Gestion des Agences, Services et Agents :** Dans la page de profil, l'administrateur peut visualiser toutes les agences, les services, et les agents. Il peut accéder à chaque élément pour effectuer des modifications ou les visualiser.

Cette section assure que les administrateurs peuvent gérer efficacement leurs informations et les différentes entités au sein du système.

#### 4.2.3.4 Paramètres

La section des paramètres permet aux administrateurs de configurer les éléments essentiels du système, y compris les agences, les services, et les utilisateurs. Cette section est subdivisée en trois parties principales :

##### Agences

La gestion des agences permet aux administrateurs de configurer et de superviser les différentes agences au sein du système. Les fonctionnalités incluent :

- Ajouter une Nouvelle Agence : Créer de nouvelles agences en fournissant les informations nécessaires telles que le nom et l'adresse.



FIGURE 4.17 – Ajouter une Nouvelle Agence.

- Modifier une Agence Existante : Mettre à jour les informations d'une agence existante ou modifier ses paramètres, ou bien la supprimer.

FIGURE 4.18 – Modifier une Agence Existante.

## Services

La gestion des services permet aux administrateurs de configurer les services proposés au sein des agences. Les fonctionnalités incluent :

- Ajouter un Nouveau Service : Créer de nouveaux services avec des paramètres spécifiques tels que le nom de service, les horaires de disponibilité et les ressources nécessaires.

FIGURE 4.19 – Ajouter un Nouveau Service.

- **Modifier un Service Existant** : Ajuster les détails d'un service existant ou mettre à jour ses paramètres, ou bien le supprimer.

FIGURE 4.20 – Modifier un Service Existant.

## Utilisateurs

La gestion des utilisateurs permet aux administrateurs de superviser les agents et administrateurs au sein du système. Les fonctionnalités incluent :

- **Ajouter un Nouvel Utilisateur** : Créer de nouveaux comptes pour les agents ou administrateurs avec les informations nécessaires.

The screenshot shows a web application interface for adding a new user. On the left is a dark sidebar menu with the user's name 'ISMAIL AHAKAY' at the top. The menu items include Dashboard, Statistiques, User Profile, Settings, Agences, Services, and Utilisateurs (which is highlighted in blue). Below Utilisateurs are sub-options: Ajouter Utilisateur, Edit Utilisateur, and Logout. The main content area has a light blue header with a 'Home' link and a search icon. Below the header is a form titled 'Ajouter une nouvelle Utilisateur' in a blue box. The form contains the following fields: 'Nom' (text input), 'Email' (text input), 'Password' (text input), 'Role' (dropdown menu with 'AGENT' selected), 'Agence' (text input), and 'Services' (dropdown menu). The form is set against a light gray background.

FIGURE 4.21 – Ajouter un Nouvel Utilisateur.

- **Modifier un Utilisateur Existant** : Mettre à jour les informations ou les paramètres des agents ou administrateurs en fonction des besoins, ou bien les supprimer.

The screenshot shows the 'Modifier un Utilisateur Existant' form. The sidebar menu is the same as in Figure 4.21. The main content area has a light blue header with 'Home' and a search icon. Below the header is a form titled 'Modifier un Utilisateur Existant' in a yellow box. The form contains the following fields: 'Nom' (text input with 'Agent1' entered), 'Email' (text input with 'agent1@gmail.com' entered), 'Role' (dropdown menu with 'AGENT' selected), 'Agence' (text input with 'Agence' entered), and 'Services' (dropdown menu). There is a red button labeled 'Supprimer Utilisateur' next to the form title. The form is set against a light gray background.

FIGURE 4.22 – Modifier un Utilisateur Existant.

Cette section assure une gestion complète des ressources humaines associées au système de gestion des files d'attente.

#### 4.2.4 Espace Agent - Gestion des Files d'Attente

Dans cette section, nous présentons l'espace dédié aux agents dans notre application de gestion des files d'attente. Cet espace permet aux agents de gérer leurs tâches spécifiques, y compris la gestion des services dont ils sont responsables, la visualisation des statistiques pertinentes, et la validation des tickets. L'espace agent est conçu pour offrir une interface simplifiée tout en permettant une gestion efficace des services et des interactions avec les visiteurs.

Les sections suivantes détaillent les différentes fonctionnalités disponibles dans l'espace agent, en mettant en avant les étapes clés et les interactions possibles.

#### 4.2.4.1 Tableau de Bord Agent

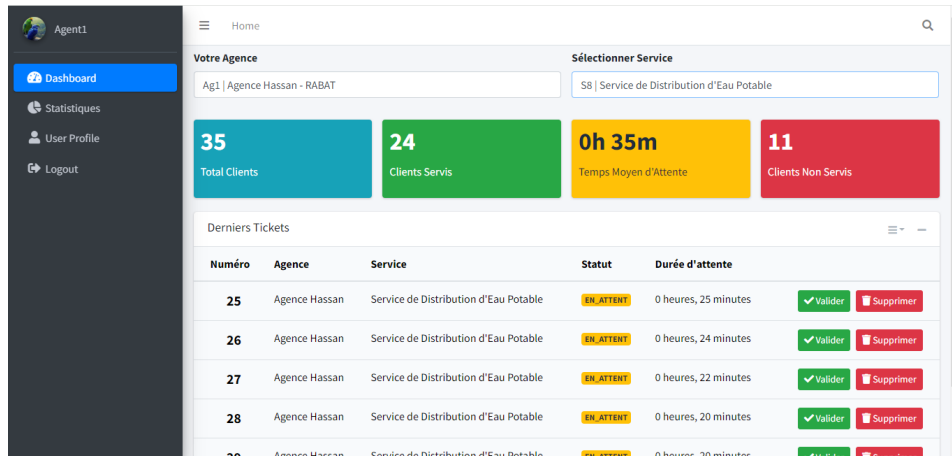


FIGURE 4.23 – Tableau de Bord Agent.

Le tableau de bord agent fournit une vue d'ensemble interactive pour la gestion des services dont l'agent est responsable. Il permet aux agents de visualiser les statistiques et les tickets associés en fonction des sélections effectuées :

- Sélectionner un Service : Les agents peuvent choisir un service à partir d'un menu déroulant. La sélection d'un service affiche les statistiques et les tickets associés à ce service. Si aucun service n'est sélectionné, les statistiques et les tickets peuvent être affichés de manière globale pour les services concernés.
- Affichage des Statistiques et des Tickets : Les statistiques clés, telles que le nombre de clients en attente et les temps d'attente moyens pour les services concernés, ainsi que les tickets associés, sont présentés en fonction de la sélection actuelle. Ces affichages permettent aux agents de suivre et de gérer les performances des services de manière détaillée et en temps réel.

#### 4.2.4.2 Statistiques

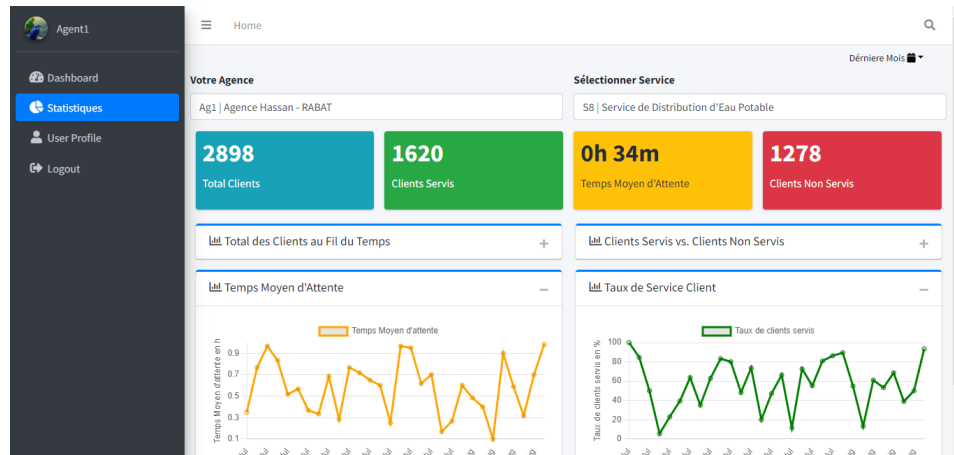


FIGURE 4.24 – Statistiques Agent.

La section des statistiques permet aux agents d'analyser les données relatives aux services dont ils sont responsables. Cette section inclut :

- Analyse des Performances : Visualiser des rapports détaillés sur les performances des services, y compris les temps d'attente moyens, le nombre de clients servis, et les tendances des visites.
- Total Clients : Affiche le nombre total de clients, incluant les clients servis et non servis pour les services concernés.
- Clients Servis : Affiche le nombre de clients qui ont été servis pour les services concernés.
- Temps Moyen d'Attente : Affiche le temps moyen d'attente pour les clients des services concernés.
- Clients Non Servis : Affiche le nombre de clients qui n'ont pas été servis pour les services concernés.

#### 4.2.4.3 Profil

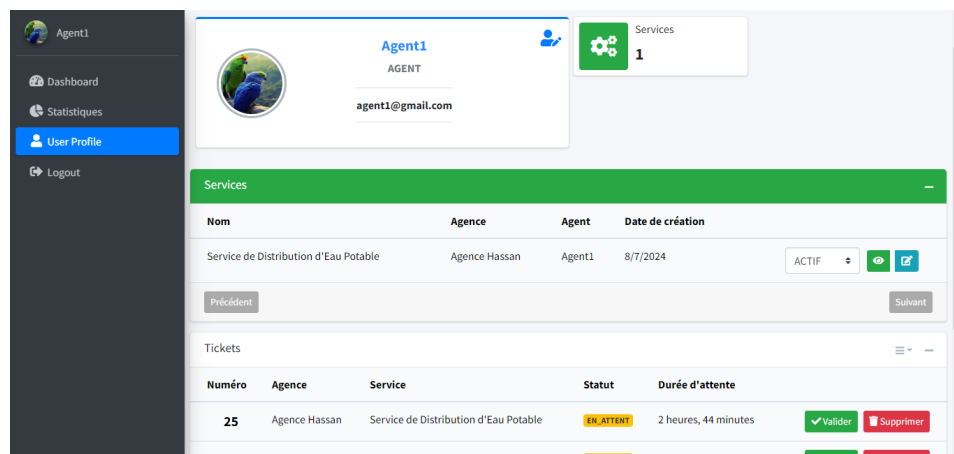


FIGURE 4.25 – Gestion du Profil de l'Agent.

La section de gestion du profil permet aux agents de mettre à jour et de personnaliser leurs informations personnelles. Cette section inclut :

- **Modification des Informations** : Mettre à jour les informations personnelles telles que le nom, l'adresse e-mail, et le mot de passe.
- **Visualisation des Services** : Dans la page de profil, l'agent peut visualiser les services dont il est responsable. Il peut accéder à chaque service pour consulter les détails.

Voici la mise à jour de la sous-section sur la validation des tickets, en incluant l'information sur la mise à jour quotidienne des files d'attente via les commandes de planification (schedules) :

#### 4.2.4.4 Validation des Tickets

| Numéro | Agence        | Service                               | Statut     | Durée d'attente      |
|--------|---------------|---------------------------------------|------------|----------------------|
| 25     | Agence Hassan | Service de Distribution d'Eau Potable | TRAITE     | 2 heures, 47 minutes |
| 26     | Agence Hassan | Service de Distribution d'Eau Potable | EN_ATTENTE | 2 heures, 47 minutes |
| 27     | Agence Hassan | Service de Distribution d'Eau Potable | EN_ATTENTE | 2 heures, 45 minutes |
| 28     | Agence Hassan | Service de Distribution d'Eau Potable | EN_ATTENTE | 2 heures, 42 minutes |
| 29     | Agence Hassan | Service de Distribution d'Eau Potable | EN_ATTENTE | 2 heures, 42 minutes |

FIGURE 4.26 – Validation des Tickets.

La validation des tickets est une fonctionnalité clé pour les agents, leur permettant de gérer les demandes des visiteurs. Les fonctionnalités incluent :

- **Validation des Tickets** : Examiner et valider les tickets soumis par les visiteurs pour les services concernés.
- **Suppression des Tickets** : Si le client associé au ticket n'existe pas dans la file d'attente ou ne peut être localisé, l'agent peut supprimer le ticket correspondant.
- **Gestion des Tickets** : Suivre l'état des tickets et gérer les demandes des visiteurs pour les services concernés.
- **Mise à Jour Quotidienne des Files d'Attente** : Les files d'attente sont mises à jour chaque jour via l'utilisation de commandes de planification (schedules). Ces commandes permettent de supprimer tous les tickets existants et d'ajouter une nouvelle file d'attente pour la journée. Ce processus garantit que les tickets sont régulièrement réinitialisés et que chaque jour commence avec une nouvelle file d'attente.

#### 4.2.4.5 Accès Restreint aux Services

Les agents n'ont pas accès à la gestion des services et des agences dans le système, à l'exception des services dont ils sont responsables. Les fonctionnalités disponibles incluent :

- Accès aux Services : Visualiser et gérer uniquement les services spécifiques dont ils sont responsables.

Cette section assure que les agents peuvent gérer efficacement leurs tâches tout en ayant des accès restreints aux fonctionnalités spécifiques au système.

#### 4.2.5 Espace Gestionnaire des Comptes

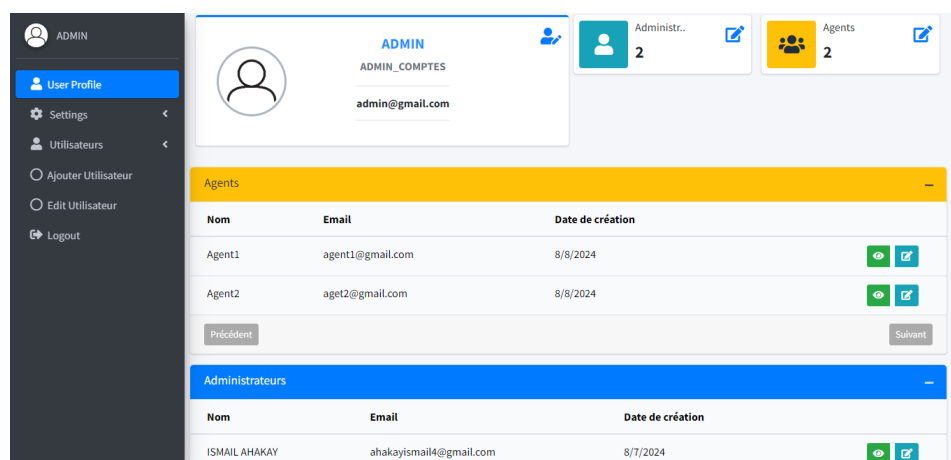


FIGURE 4.27 – Profil du Gestionnaire des Comptes.

Au sein de notre application, l'Espace Gestionnaire des Comptes permet aux gestionnaires d'accéder aux fonctionnalités essentielles liées au profil et aux paramètres du système.

##### 4.2.5.1 Gestion du Profil

Les gestionnaires peuvent mettre à jour leurs informations personnelles, telles que le nom, l'adresse e-mail et le mot de passe, à travers cette section.

#### 4.2.5.2 Paramètres

The screenshot shows a web application interface with a dark sidebar on the left. The sidebar contains a user profile section with 'ADMIN' and a menu with 'User Profile', 'Settings', 'Utilisateurs', 'Ajouter Utilisateur', 'Edit Utilisateur', and 'Logout'. The 'Utilisateurs' menu item is highlighted. The main content area is titled 'Home' and displays a form to add a new user. The form has a blue header 'Ajouter une nouvelle Utilisateur' and fields for 'Nom', 'Email', 'Password', and 'Role'. The 'Role' field is set to 'ADMINISTRATEUR'. A blue 'Submit' button is at the bottom.

FIGURE 4.28 – Ajouter Utilisateur.

The screenshot shows the same web application interface. The sidebar is identical. The main content area now displays a form to edit an existing user. The form has a yellow header 'Éditer Utilisateur' and a red button 'Supprimer Utilisateur'. It contains fields for 'Nom', 'Email', and 'Role'. The 'Nom' field is set to 'ISMAIL AHAKAY', the 'Email' field is set to 'ahakayismail4@gmail.com', and the 'Role' field is set to 'ADMINISTRATEUR'. A blue 'Submit' button is at the bottom.

FIGURE 4.29 – Éditer Utilisateur.

Cette section permet aux gestionnaires de configurer les éléments clés du système, notamment la gestion des utilisateurs, ce qui inclut l'ajout, la modification ou la suppression des comptes des agents et des administrateurs travaillant dans le système.

### 4.3 Caractéristiques de Notre Application

Dans cette section, nous allons discuter des caractéristiques clés de notre application qui contribuent à sa robustesse, sa sécurité et son efficacité. Deux de ces caractéristiques importantes sont l'utilisation de Laravel pour la gestion des données et des fonctionnalités backend, ainsi que l'intégration de Vue.js pour une interface utilisateur dynamique et réactive.



### 4.3.1 Laravel pour la Gestion des Données et des Fonctionnalités Backend

Nous avons intégré Laravel dans notre application pour bénéficier de ses fonctionnalités puissantes et de son architecture MVC robuste. Laravel facilite la gestion des données, l'authentification, l'autorisation, et bien d'autres aspects essentiels du développement backend.

#### 4.3.1.1 Gestion des Données avec Laravel

Dans notre application, Laravel est utilisé pour gérer les interactions avec la base de données de manière efficace. Grâce à Eloquent ORM, nous pouvons facilement définir des modèles de données, effectuer des requêtes complexes et assurer l'intégrité des données.

- **Eloquent ORM** : Permet de manipuler les données de la base de données de manière simple et intuitive.
- **Migrations** : Assurent la gestion des schémas de la base de données et permettent de versionner les modifications.
- **Seeders** : Facilitent le peuplement de la base de données avec des données de test.

#### 4.3.1.2 Authentification et Autorisation avec Laravel

Laravel simplifie l'implémentation des systèmes d'authentification et d'autorisation, garantissant ainsi la sécurité de l'application. Les middlewares de Laravel permettent de protéger les routes et de contrôler l'accès en fonction des rôles des utilisateurs.

- **Middleware** : Utilisé pour filtrer les requêtes HTTP et protéger les routes de l'application.
- **Gates et Politiques** : Définissent des règles d'autorisation basées sur les rôles et les permissions des utilisateurs.

### 4.3.2 Vue.js pour une Interface Utilisateur Dynamique et Réactive

Vue.js est utilisé pour créer une interface utilisateur dynamique et réactive. Grâce à son architecture flexible et à ses composants réutilisables, Vue.js permet de construire des interfaces utilisateur modernes et performantes.

#### 4.3.2.1 Composants Réactifs avec Vue.js

Vue.js permet de diviser l'interface utilisateur en composants réutilisables, facilitant ainsi le développement et la maintenance de l'application.

- **Composants** : Divisent l'interface utilisateur en éléments modulaires et réutilisables.
- **Props et Événements** : Facilitent la communication entre les composants.
- **Directives** : Ajoutent des comportements réactifs aux éléments du DOM.

#### 4.3.2.2 Vue Router pour la Navigation

Vue Router est utilisé pour gérer la navigation au sein de l'application, offrant une expérience utilisateur fluide et intuitive.

- **Routes Dynamiques** : Permettent de définir des chemins et de les associer à des composants spécifiques.
- **Navigation Protégée** : Utilisation de guards pour sécuriser l'accès aux différentes pages de l'application.

## Chapitre 5

# Conclusion

### 5.1 Bénéfices du projet de stage

Ce projet de gestion des files d'attente m'a offert une occasion précieuse de consolider et d'étendre mes compétences techniques et professionnelles. J'ai pu approfondir ma maîtrise des technologies web en travaillant avec divers frameworks pour le développement front-end et back-end, tout en m'immergeant dans les défis liés à l'optimisation des processus opérationnels et à l'amélioration de l'expérience utilisateur. En outre, j'ai acquis une meilleure compréhension des défis liés à la gestion des flux de personnes, à l'automatisation des processus, et à la communication efficace entre les usagers et les gestionnaires. Ce projet m'a également permis de développer des compétences en gestion de projet, en résolution de problèmes complexes et en collaboration au sein d'une équipe pluridisciplinaire, renforçant ainsi ma capacité à mener à bien des projets technologiques innovants.

### 5.2 Futures améliorations

Une amélioration clé pour le système de gestion des files d'attente serait de le rendre accessible en ligne, permettant ainsi aux clients de prendre leurs tickets à distance via une plateforme web. Cela offrirait une flexibilité accrue aux utilisateurs, qui pourraient réserver leur place à l'avance, réduisant ainsi les temps d'attente physiques et améliorant leur expérience globale.

De plus, l'ajout d'une fonctionnalité de feedback directement sur le site permettrait aux clients de partager leur expérience après leur visite. Ce retour d'information en temps réel fournirait des données précieuses pour les gestionnaires, favorisant une amélioration continue des services. Ces deux améliorations contribueraient à moderniser le système tout en répondant aux attentes croissantes des utilisateurs en matière de simplicité et d'efficacité.

# Bibliographie

- [1] Laravel official website. lien : <https://laravel.com/>
- [2] Vue.js official website. lien : <https://vuejs.org/>
- [3] Laravel + Vue.js : Fullstack Developer Guide, lien : <https://vueschool.io/articles/vuejs-tutorials/the-ultimate-guide-for-using-vue-js-with-laravel/>
- [4] Eloquent ORM documentation. lien : <https://laravel.com/docs/eloquent>
- [5] Vue Router documentation. lien : <https://router.vuejs.org/>
- [6] Laravel authentication documentation. lien : <https://laravel.com/docs/authentication>
- [7] Laravel Planification des tâches documentation. lien : <https://laravel.com/docs/11.x/scheduling>
- [8] Laravel Autorisation documentation.lien : <https://laravel.com/docs/11.x/authorization>
- [9] stackoverflow : <https://stackoverflow.com/>

Nous exprimons notre gratitude envers les auteurs et les chercheurs dont les travaux ont été une source précieuse d'information pour notre projet.

**© AHAKAY ISMAIL 2024 All rights reserved.**