RÉPUBLIQUE DU BÉNIN
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR
ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITÉ D'ABOMEY-CALAVI

INSTITUT DE FORMATION ET DE RECHERCHE EN INFORMATIQUE

BP 526 Cotonou Tel : +229 21 14 19 88
http://www.ifri-uac.bj Courriel : contact@ifri.uac.bj

MÉMOIRE

pour l'obtention du

Diplôme de Licence en Informatique

Option : Génie Logiciel

Présenté par :
Aquilas Ouwatobi Elisée **KIKISSAGBE**

Conception d'une application web pour la gestion des réservations de transport et l'expédition de marchandises par voie routière

Sous la supervision :
M. Fidèle **HOUNSOUNON**

Membres du jury :

KONNON Abel	Docteur	UNSTIM	Président
AFFOGNON Lionel	Ingénieur	IMSP	Examinateur
HOUNSOUNON Fidèle	Ingénieur	IFRI	Rapporteur

Année Académique : **2023-2024**

Sommaire

Dédicace ii

Remerciements iii

Résumé iv

Abstract v

Liste des Figures vi

Liste des acronymes vii

Glossaire 1

Introduction 2

1 Revue de littérature 4

2 Analyse, conception et choix techniques 10

3 Résultats & discussion 20

Conclusion 29

Bibliographie 30

Webographie 31

Table des matières 33

Dédicace

Je dédie ce travail à :

- l'Éternel Dieu Tout-Puissant, source de toute inspiration et de sagesse infinie, qui m'a guidé et protégé tout au long de ce parcours académique ;
- mes chers parents, pour leur soutien et leurs encouragements ;
- mes frères et sœurs, ainsi qu'à tous ceux qui, de près ou de loin, ont contribué à mon évolution personnelle et intellectuelle.

Remerciements

Je tiens à exprimer ma profonde gratitude à toutes les personnes qui ont contribué, d'une manière ou d'une autre, à la réalisation de ce mémoire.

En premier lieu, je remercie sincèrement le Professeur C. Eugène EZIN, Directeur de l'Institut de Formation et de Recherche en Informatique (IFRI), ainsi que tous les membres de l'administration d'IFRI pour leur disponibilité et leur encadrement durant ce parcours universitaire.

Je remercie également mon encadreur, M. Fidèle HOUNSOUNON, pour le temps qu'il a su me consacrer pour l'élaboration de ce mémoire.

Je n'oublie pas mes amis et collègues, avec qui j'ai partagé des moments de doute, de joie et de soutien mutuel. Vos encouragements et votre camaraderie ont rendu ce parcours plus enrichissant.

Enfin, je tiens à remercier chaleureusement ma famille pour leur soutien constant, sans lequel ce mémoire n'aurait pas été possible.

Que Dieu vous bénisse tous !

Résumé

Le transport est un pilier fondamental de l'économie mondiale, permettant de relier les différents points de production, de distribution et de consommation, facilitant ainsi les échanges commerciaux et la mobilité des personnes. Plus spécifiquement, le transport routier occupe une place centrale dans l'économie moderne. Il assure la circulation des biens et des personnes à travers des réseaux de routes et d'autoroutes, et est particulièrement essentiel dans la gestion des chaînes logistiques, en reliant les zones de production aux zones de consommation.

C'est dans ce contexte que nous avons décidé de concevoir une application web dédiée à la gestion des réservations et des opérations liées au transport routier et à l'expédition de marchandises. Le système permet une gestion complète des véhicules, des expéditions et des réservations de transport. Les administrateurs peuvent réserver des créneaux pour l'expédition de marchandises, initier des transports, et suivre les status des véhicules et des livraisons. Les utilisateurs, quant à eux, peuvent réserver des transports, enregistrer et suivre leurs marchandises expédiées. Développée avec les technologies PHP, TypeScript, JavaScript, ainsi que les frameworks Laravel et Angular, cette solution repose sur une base de données MySQL, garantissant ainsi la performance et la scalabilité des opérations. En utilisant le langage de modélisation UML, l'application a été conçue pour être modulaire et facilement extensible, permettant l'ajout de nouvelles fonctionnalités à l'avenir.

Cette solution vise à optimiser l'ensemble des opérations logistiques liées au transport routier, en améliorant l'efficacité des processus de réservation et d'expédition.

Mots-clés : Application web, transport routier, gestion logistique, réservation de transport, expédition de marchandises

Abstract

Transport is a fundamental pillar of the global economy, connecting various points of production, distribution, and consumption, thus facilitating commercial exchanges and the mobility of people. More specifically, road transport plays a central role in the modern economy. It ensures the movement of goods and people through networks of roads and highways, and is particularly essential in managing supply chains by connecting production areas to consumption areas.
In this context, we decided to develop a web application dedicated to managing bookings and operations related to road transport and freight shipping. The system allows for comprehensive management of vehicles, shipments, and transport reservations.
Administrators can book slots for shipping goods, initiate transports, and monitor the status of vehicles and deliveries. Users, on the other hand, can book transports, register, and track their shipped goods. Developed using PHP, TypeScript, JavaScript technologies, and the Laravel and Angular frameworks, this solution is based on a MySQL database, ensuring the performance and scalability of operations. By using UML modeling language, the application is designed to be modular and easily extensible, allowing for the addition of new features in the future.
This solution aims to optimize all logistics operations related to road transport by improving the efficiency of booking and shipping processes.
Keywords:Web application, road transport, logistics management, transport booking, merchandises shipping

Liste des Figures

2.1 Diagramme de cas d'utilisation 14
2.2 Diagramme de classe 15
2.3 Diagramme de séquence d'ajout des transports 16
2.4 Diagramme de séquence de réservation de transports 17
2.5 Diagramme de séquence des expéditions 18
2.6 Diagramme de séquence des marchandises 19

3.1 LogiTrack 20
3.2 Inscription 21
3.3 Connexion 22
3.4 Liste des véhicules 22
3.5 Liste des transports 23
3.6 Liste des expéditions 24
3.7 Listes des transports 25
3.8 Listes des marchandises 26
3.9 Enregistrer une marchandise 27

Liste des acronymes

API :

Application Programming Interface *:* API

ASP :

Active Server Pages *:* ASP

CMS :

Content Management System *:* CMS

ERP :

Enterprise Resource Planning *:* ERP

HTTP :

Hypertext Transfer Protocol *:* HTTP

MVC :

Model-View-Controller *:* MVC

SVG :

Scalable Vector Graphics *:* SVG

XML :

Extensible Markup Language *:* XML

Glossaire

API : Une interface de programmation (API) est un ensemble normalisé de classes, de méthodes, de fonctions et de constantes qui sert d'interface de programmation pour un certain service, bibliothèque ou programme informatique. vii

ASP : est une technologie de Microsoft permettant de créer des pages web dynamiques. vii

CMS : est un logiciel permettant de créer et de gérer facilement du contenu web. vii

ERP : est un système d'information qui intègre l'ensemble des processus clés d'une organisation dans un système unique. vii

HTTP : est un protocole de communication client-serveur développé pour le World Wide Web. C'est un langage permettant d'écrire de l'hypertexte, d'où son nom. vii

MVC : est un motif d'architecture logicielle qui sépare les données d'une application, l'interface utilisateur et la logique de contrôle en trois composants distincts. vii

SVG : est un format ouvert et vectoriel d'images destiné à être utilisé sur le web. Il permet de créer des graphiques évolutifs, c'est-à-dire pouvant s'agrandir sans perte de qualité. vii

XML : est un langage de balisage générique qui sert à encoder des documents de manière lisible par l'homme et lisible par la machine. vii

Introduction Générale

La gestion des transports joue un rôle crucial dans le secteur logistique. Elle est indispensable pour la mobilité des personnes, facilitant l'accès aux opportunités d'emploi, à l'éducation, aux services de santé et aux loisirs. Elle permet d'optimiser les flux de marchandises, de réduire les coûts et de garantir une livraison efficace des produits. Une accessibilité optimisée aux services de transport améliore significativement la qualité de vie des citoyens en leur offrant des moyens de déplacement efficients et économiques.

Contexte de l'étude

Les échanges commerciaux, tant au niveau local qu'international, reposent largement sur des systèmes de transport fiables et efficaces. Qu'ils soient terrestres, maritimes, aériens ou ferroviaires, les transports jouent un rôle crucial dans le rapprochement des producteurs et des consommateurs, réduisant ainsi les distances géographiques et les barrières à l'échange. Ils permettent aux individus de se déplacer pour le travail, l'éducation, les loisirs et les activités sociales. Ils favorisent la mobilité sociale et économique en facilitant l'accès aux opportunités et aux ressources, contribuant ainsi non seulement au développement global des régions, mais également à la diversification des marchés, à l'amélioration de l'accès aux biens et services essentiels et à l'augmentation des échanges commerciaux.

Problématique

Le secteur des transports joue un rôle central dans le développement économique et social d'un pays. Cependant, ce secteur fait face à de nombreux défis qui entravent son efficacité et sa durabilité. Parmi ces défis figurent, entre autres, la gestion des ressources et les coûts élevés, limitant leur performance et leur rentabilité. Dans ce contexte, concevoir un système de gestion des opérations de transport devient crucial, notamment pour la digitalisation des processus. Comment concevoir un système performant et flexible, capable de répondre aux besoins diversifiés des acteurs de la chaîne de transport, tout en assurant une meilleure rentabilité et une gestion optimisée des ressources ?

Objectifs de l'étude

L'objectif principal de ce projet est de concevoir une application web performante et flexible dédiée à la gestion des transports de biens et de personnes. Plus précisément, cette application permettra de :

- Gérer les réservations de transport routier en ligne de manière simplifiée ;
- Optimiser la gestion des expéditions de marchandises en garantissant leur sécurité et efficacité;
- Améliorer la visibilité des stocks de marchandises à expédier en temps réel ;
- Gérer les véhicules de manière optimisée, avec une superposition d'informations contextuelles pour une prise de décision rapide ;

Organisation du mémoire

Le présent mémoire, qui rend compte de nos travaux, est structuré en trois chapitres. Le premier chapitre propose une revue de la littérature relative à notre problématique. Dans le deuxième chapitre, nous détaillons les choix techniques réalisés pour la conception et la mise en œuvre de la solution proposée. Enfin, le troisième chapitre présente les résultats obtenus, la description des interfaces, ainsi que les limitations identifiées dans notre solution.

Chapitre 1

Revue de littérature

Introduction

Ce chapitre pose les fondements de notre étude en définissant quelques concepts clés et en analysant les solutions existantes dans le secteur de la gestion des transports.

1.1 Clarification Conceptuelle

1.1.1 Application Web

1.1.1.1 Définition

Une **application web** désigne un logiciel applicatif hébergé dans le cloud ou sur des serveurs dédiés, et accessible via un navigateur web. Contrairement à un logiciel traditionnel, l'utilisateur d'une application web n'a pas besoin de l'installer sur son ordinateur. Il lui suffit de se connecter à l'application à l'aide de son navigateur, ce qui permet d'y accéder depuis n'importe quel type d'appareil (mobile, ordinateur, etc.).[13]

1.1.1.2 Fonctionnement d'une application web

Les applications web récupèrent et stockent des informations en utilisant des scripts côté serveur (dans des langages de script tels que PHP et ASP), tandis que les scripts côté client (en JavaScript et HTML5) présentent les informations pertinentes sur l'interface utilisateur. Ces informations peuvent prendre de nombreuses formes.

Un flux de travail typique d'une application web se déroule de la manière suivante :

1. Un utilisateur envoie une requête à un serveur web via Internet, en utilisant soit un navigateur web, soit l'interface utilisateur d'une application.
2. La requête est transmise par le serveur web au serveur d'application web correspondant.
3. Le serveur d'application web exécute la tâche demandée (par exemple, le traitement de nouvelles données), puis génère les résultats appropriés.

4. Les résultats, ainsi que les informations demandées, sont ensuite envoyés du serveur d'application web au serveur web.

5. Le serveur web répond alors au client et présente les informations demandées à l'utilisateur.

Les applications web intègrent généralement des langages de programmation tels que JavaScript et HTML. Cela s'explique par le fait que ces langages sont supportés par les navigateurs : autrement dit, ils s'appuient sur les navigateurs web pour rendre l'application exécutable. Un serveur web est nécessaire pour gérer les requêtes côté client, ainsi qu'un serveur d'application pour exécuter les tâches correspondantes. Une base de données peut également être nécessaire pour stocker les informations.[14]

1.1.1.3 Type d'application web

Nous distinguons deux types d'applications web :

1. **Application Web Statique**

   Il s'agit de l'application web de base qui possède peu de contenu et offre peu de possibilités de mouvement. Ces applications web sont souvent créées en utilisant du CSS et du HTML et peuvent supporter du contenu animé comme les GIFS et les vidéos.

   À titre d'exemples, les applications web statiques incluent les portfolios en ligne ou les curriculums digitaux. Dans la même veine, vous pourriez considérer la page de service (landing page) comme une application web statique dont les informations de contacts sont affichées.

   Lorsqu'un serveur Web reçoit une requête de page statique, il lit la requête, localise la page et la transmet au navigateur qui l'a sollicitée.

2. **Application Web Dynamique**

   Ce style d'application est techniquement plus complexe que l'application statique précédemment mentionnée. En ce qui concerne les applications dynamiques, celles-ci sont en place pour faire en sorte que les bases de données ou les forums soient constamment capables de mettre à jour ou de changer l'information disponible. Cela est habituellement possible grâce à l'utilisation d'un CMS ou d'un système de gestion de contenu.

   Un CMS accompagne généralement une application web dynamique afin que les administrateurs ou les utilisateurs puissent facilement mettre à jour ou éditer le contenu. Plusieurs langages web différents peuvent être utilisés avec des applications web dynamiques. Néanmoins, PHP et ASP sont les plus communs, étant donné qu'ils sont les plus faciles à comprendre pour structure le contenu. [14]

1.1.2 TMS

1.1.2.1 Définition

Un Transport Management System (TMS), ou Système de Gestion des Transports en français, est une plateforme logicielle conçue pour planifier, exécuter, et optimiser le transport physique. Le TMS est très souvent utilisé par les entreprises pour gérer les opérations de transport, qu'il s'agisse de l'expédition de produits aux clients, de la gestion de la chaîne d'approvisionnement ou du déplacement de matières premières entre les sites de production.

1.1.2.2 Fonctionnalités d'un TMS

Les logiciels TMS modernes comportent des fonctionnalités qui peuvent considérablement réduire la complexité et améliorer l'efficacité. Les fonctionnalités clés sont les suivantes :

- Planification et exécution du transport : rationalisez les achats et l'expédition du fret grâce à la comparaison et à la réservation automatisées des tarifs des transporteurs. Sélectionnez le mode de transport (fret aérien, maritime, routier ou ferroviaire) et planifiez l'itinéraire le plus efficace pour le transport de marchandises. Optimisez les charges et tirez parti des fonctionnalités de suivi et de surveillance en temps réel pour suivre l'avancement. [11]
- Gestion du fret : rationalisez le processus du devis au contrat. Gérez efficacement le calcul du coût de revient du fret, la gestion des commandes, la détermination des tarifs, ainsi que la facturation et le règlement du fret pour le transport multimodal et intermodal.
- Tableaux de bord, reporting et analytique TMS : anticipez la demande de transport, analysez les tarifs et la rentabilité et adaptez-vous rapidement pour répondre aux circonstances imprévues. Grâce à une visibilité en temps réel sur tous les aspects du processus de transport, vous pouvez prendre des décisions immédiates et fondées sur les données. [17]

1.2 Application Web et TMS

Dans le contexte de la logistique moderne, où la digitalisation joue un rôle crucial, les applications web s'imposent comme des outils indispensables pour une gestion efficace. Leur disponibilité multiplateforme permet aux utilisateurs d'accéder aux services en tout lieu et à tout moment.
Les applications web de gestion des transports peuvent facilement s'intégrer avec des systèmes tiers tels que les systèmes de gestion des entrepôts (WMS), les ERP (Enterprise Resource Planning), ou les CRM (Customer Relationship Management), en utilisant des API et des services web. Cela permet un flux d'informations fluide en temps réel et une coordination optimale entre différents départements, renforçant ainsi leur interopérabilité.
Ces applications peuvent également s'adapter aisément à la croissance de la demande et aux évolutions technologiques, en ajoutant de nouvelles fonctionnalités ou en augmentant la capacité du système sans nécessiter de refonte majeure, ce qui en fait des solutions évolutives.
En résumé, les applications web de gestion des transports offrent une solution complète, combinant la puissance de l'internet avec des fonctionnalités robustes pour optimiser la logistique. Ces systèmes permettent une gestion plus fluide, une meilleure coordination et une amélioration continue des performances.

1.3 Les solutions existantes

Nous présenterons ici une analyse comparative de quelques solutions de transport et logistique. Cette analyse s'appuie sur les principes de gestion des transports modernes, les meilleures pratiques en logistique [1] ainsi que des avis utilisateurs collectés sur la **plateforme Trustpilot** [16].

1.3.1 Gozem

1.3.1.1 Présentation

Gozem est une application de services de mobilité et de logistique en Afrique de l'Ouest, notamment au Togo, Bénin et Gabon. Lancée en 2018, elle a démarré comme plateforme de réservation de taxi-moto (zémidjan) avant d'étendre son offre.

- **Réservation de taxis-motos, tricycles et voitures** pour des déplacements urbains
- **Services de livraison** de colis et de marchandises
- **Paiement électronique** sécurisé, incluant Mobile Money, facilitant les transactions

1.3.1.2 Limites de Gozem

- Concentration davantage sur le transport de personnes que sur l'expédition de marchandises
- Accessibilité limitée aux appareils mobiles.

1.3.2 Benin Ticket Express (BTX)

1.3.2.1 Présentation

BTX est une plateforme de réservation de bus au Bénin, lancée le 1er août 2018. Elle permet de réserver des billets de bus pour des trajets au Bénin et dans d'autres pays de l'Afrique de l'Ouest, avec des fonctionnalités de comparaison des tarifs, consultation des horaires et paiement en ligne. [24]

1.3.2.2 Limites de BTX :

- Absence de services d'envoi de colis.
- Réponses automatiques parfois non pertinentes aux requêtes des utilisateurs.

1.3.3 Flash Réservation

1.3.3.1 Présentation

Flash Réservation est une plateforme béninoise de réservation en ligne offrant divers services liés aux voyages et événements.

- **Réservation de transport** : billets de bus, taxis et tricycles.
- **Hébergement** : réservation d'hôtels, appartements et salles de fête.
- **Sites touristiques** : visites organisées à des sites populaires.
- **Paiement en ligne** sécurisé (Mobile Money, cartes bancaires, etc.).

1.3.3.2 Limites de Flash Réservation

- Couverture limitée pour certaines destinations.
- Disponibilité des services pouvant varier en temps réel.

1.3.4 Happy Post

1.3.4.1 Présentation

Happy Post propose des services de livraison de colis et courriers au Bénin, avec un suivi en temps réel. [23]

- Livraison rapide et sécurisée de colis.
- Suivi en ligne des envois.
- Tarification compétitive et couverture nationale.

1.3.4.2 Limites de Happy Post

- Difficulté à joindre le service client dans certains cas.
- Retards possibles dans les livraisons.
- Problèmes techniques occasionnels liés au suivi.

1.3.5 Kobo360

1.3.5.1 Présentation

Kobo360 est une plateforme logistique nigériane spécialisée dans le transport de fret en Afrique.

- **Services** : interconnexion entre expéditeurs et transporteurs, suivi des expéditions en temps réel, gestion documentaire numérisée et options de financement via KoboCare
- **Impact** : optimisation des chaînes d'approvisionnement et amélioration de la transparence logistique

1.3.5.2 Limites de Kobo360

- Focalisation exclusive sur le transport de marchandises, excluant le transport de personnes

1.4 Avantages différenciateurs de notre solution

Notre solution se distingue par plusieurs avantages clés qui répondent aux limitations identifiées dans les solutions existantes :

- **Accessibilité optimisée** :
  - Interface intuitive et ergonomique conforme aux standards d'ergonomie [4]
  - Disponibilité sur plateformes web et mobile
- **Optimisation des coûts** :
  - Réduction significative des coûts grâce à une meilleure planification
  - Mutualisation des ressources de transport

- **Solution intégrée** :
  - Gestion unifiée du transport de personnes et de marchandises
  - Système de suivi des marchandises par les utilisateurs

Conclusion

Ce chapitre nous a permis de mieux clarifier la thématique des systèmes de gestion des transports et de prendre connaissance des diverses solutions existantes ainsi que leurs forces et leurs limites, ce qui nous permettra de mieux adapter notre solution. Le chapitre suivant fera l'objet d'une présentation de notre projet à travers l'analyse, la conception et les choix techniques utilisés pour sa réalisation.

Chapitre 2

Analyse, conception et choix techniques

Introduction

Ce projet propose une application web de gestion étendue des transports de biens et de passagers, permettant de gérer et de suivre les transports, les réservations, les expéditions et les marchandises. Ce chapitre présente une analyse des besoins (fonctionnels et non fonctionnels) du système et les méthodes techniques envisagées pour atteindre les objectifs fixés.

2.1 Analyse des besoins

L'analyse des besoins est une étape essentielle dans tout projet de développement. Elle définit les fonctionnalités et les exigences du système pour répondre aux besoins des utilisateurs et des parties prenantes.

2.1.1 Besoins fonctionnels

1. **Gestion des utilisateurs**
    - **Gestion des rôles :** Implémentation d'une logique de gestion des accès par rôles :
        - **Administrateur :**
            - Ajouter et gérer les transports.
            - Ajouter et gérer les expéditions.
            - Gérer et suivre les véhicules.
            - Gérer les marchandises enregistrées par les utilisateurs.
        - **Utilisateur :**
            - Consulter et visualiser les transports et les expéditions.
            - Gérer ses propres marchandises.
            - Effectuer des réservations sur les transports disponibles.
2. **Gestion des réservations**

- **Réservation en ligne :** Permettre aux clients de réserver des transports via une interface web.
- **Gestion des capacités :** Assurer le suivi des capacités des véhicules pour éviter les surcharges ou sous-utilisations en fonction des réservations.

3. **Gestion des expéditions**
   - **Suivi des expéditions :** Permettre aux expéditeurs de suivre leurs envois en temps réel.
   - **Gestion des inventaires :** Offrir une visibilité en temps réel sur les marchandises en transit.
4. **Gestion des véhicules**
   - **Suivi des véhicules :** Assurer le suivi de la disponibilité des véhicules.
   - **Allocation des ressources :** Affecter les véhicules en fonction des besoins d'expédition et du type de transport (routier, maritime, aérien).
5. **Gestion des marchandises**
   - **Enregistrement des marchandises :** Permettre aux utilisateurs d'enregistrer les marchandises.
   - **Suivi des marchandises :** Permettre aux utilisateurs de modifier, supprimer et suivre l'état de leurs marchandises.
6. **Facturation et paiements**
   - **Facturation automatisée :** Générer automatiquement les tickets de réservation.
   - **Paiements en ligne :** Intégrer des options de paiement en ligne pour faciliter les transactions.
7. **Communication client**
   - **Notifications et alertes :** Envoyer des notifications aux clients sur le statut de leurs transports ou expéditions.

2.1.2 Besoins non fonctionnels

1. **Performance**
   - **Temps de réponse :** Le système doit offrir des temps de réponse rapides.
   - **Scalabilité :** Le système doit pouvoir gérer une augmentation du nombre d'utilisateurs, de véhicules et de marchandises sans perte de performance.
2. **Sécurité**
   - **Confidentialité des données :** Protéger les informations des clients et des expéditions par des mécanismes de sécurité.
   - **Authentification sécurisée :** Mettre en œuvre des protocoles d'authentification robustes pour sécuriser l'accès au système.
3. **Facilité d'utilisation**

- **Interface utilisateur :** Concevoir une interface intuitive et facile à utiliser.
- **Accessibilité :** Le système doit être accessible sur différents types d'appareils (ordinateurs, tablettes, smartphones) et respecter les normes d'accessibilité.

4. **Maintenabilité**

   - **Modularité :** Construire le système de manière modulaire pour faciliter la maintenance, les mises à jour, et l'ajout de nouvelles fonctionnalités.
   - **Documentation :** Fournir une documentation complète du système.

2.2 Méthode de développement

Pour le développement de l'application, nous avons adopté la méthode **Scrum**. Cette méthode, une des méthodes de gestion de projet Agile, tire son nom du monde du rugby (scrum = mêlée), car les équipes agiles qui utilisent Scrum se réunissent le plus souvent possible afin de vérifier que le projet avance correctement, toujours prêtes à le réorienter au fil de son avancement. C'est donc une approche dynamique de la conduite de projet, qui garantit que les objectifs sont réellement atteints. [18]

2.3 Choix des outils techniques

2.3.1 Frameworks et languages de programmation

- **Backend** :

  Le backend de l'application a été réalisé avec le framework **Laravel (version 11)**. Ce choix a été motivé par les nombreux avantages qu'offre Laravel, tels que son écosystème robuste et sa facilité de développement rapide [7]. Laravel permet de mettre en place des fonctionnalités essentielles telles que l'authentification, la gestion des migrations de bases de données et la validation des données de manière simple et efficace. Il est particulièrement apprécié pour sa structure MVC, qui facilite la séparation des responsabilités [8], et pour sa vaste communauté, garantissant un support et des ressources abondantes pour résoudre divers défis du développement backend.

- **Frontend** :

  Le frontend a été développé avec **Angular (version 16)** pour sa capacité à créer des applications web dynamiques et réactives [10]. Angular permet de construire des applications web monopages (SPA) qui offrent une expérience utilisateur fluide sans rechargement de page. Sa structure basée sur TypeScript améliore la maintenabilité du code et la détection des erreurs, tandis que l'architecture modulaire d'Angular facilite la gestion des composants [8]. Ce framework offre également des outils performants pour le routage, la gestion des formulaires et l'intégration de services, optimisant le développement et la scalabilité de l'application.

- **Languages de programmations et frameworks additionnels**

Les langages de programmation et outils additionnels qu'on a eu à utiliser pour le développement de l'appliction sont :

- **HTML :**
  HTML signifie « HyperText Markup Language » qu'on peut traduire par « langage de balises pour l'hypertexte ». Il a été utilisé afin de créer et de représenter le contenu des pages web et de leur structure. [25]
- **CSS :**
  CSS (pour Cascading Style Sheets en anglais), soit feuilles de style en cascade, est un langage de feuille de style utilisé pour décrire la présentation d'un document écrit en HTML ou XML (y compris les dialects XML que sont SVG, MathML, ou XHTML). CSS décrit la façon dont les éléments doivent être affichés à l'écran, sur papier, à l'oral ou sur d'autres médias. [26]
- **Bootstrap :**
  Bootstrap est une collection d'outils utiles à la création du design (graphisme, animation et interactions avec la page dans le navigateur, etc.) de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option. [27]
- **Ng-zorro :**
  NG-ZORRO est une bibliothèque de composants Angular de haute qualité qui implémente les spécifications d'Ant Design. Autrement dit, c'est une Bibliothèque de composants complète , incluant une large gamme de composants tels que des tableaux, des formulaires, des modaux, des notifications, et bien plus encore, tous conçus selon les normes d'Ant Design ; qui ont été très utiles lors de la réalisation de l'application.
- **Markdown :**
  Markdown est un langage de balisage léger et facile à utiliser pour formater du texte. Il a est utilisé pour rédiger la documentation de l'application.

2.3.2 Choix du SGBD

Nous avons opté pour le système de gestion de bases de données **MySQL** en raison de sa complémentarité avec le langage de programmation PHP, et donc avec le framework Laravel [2]. Cette combinaison assure une intégration fluide, facilitant le développement et la gestion des applications. MySQL est particulièrement efficace pour la gestion des données intensives et leur traitement [5].

2.3.3 Logiciels utilisés

- **Environnement de développement :** Pour le développement de notre application, nous avons utilisé **Visual Studio Code (version 1.92)** de Microsoft pour sa flexibilité, son large éventail d'extensions, et sa capacité à offrir une expérience de codage fluide et efficace [9].
- **Gestion des versions : Git (version 2.47.1)** a été utilisé pour gérer les versions du code source, permettant un suivi précis des modifications [6].
- **LAMP :** Nous avons utilisé la pile LAMP (Linux, Apache, MySQL, PHP) pour mettre en place un serveur de développement performant et adapté aux besoins de notre application.

- **UML version 2.5.1 :**

  Le **Langage de Modélisation Unifié (UML)** a été utilisé pour concevoir et documenter les différents diagrammes nécessaires à la modélisation du projet [3].

2.4 Modélisation & Conception

La modélisation et la conception visent à représenter les interactions et les fonctionnalités du système à travers des schémas clairs. Cette étape permet de définir la structure de notre système et d'anticiper son comportement en fonction des besoins des utilisateurs.

2.4.1 Diagramme de cas d'utilisation

Un diagramme de cas d'utilisation permet de représenter les fonctions d'un système du point de vue de l'utilisateur (appelé « acteur » en UML).

Ce diagramme montre les interactions entre les acteurs et le système : C'est à dire les actions que peuvent mener les acteurs dans le système.

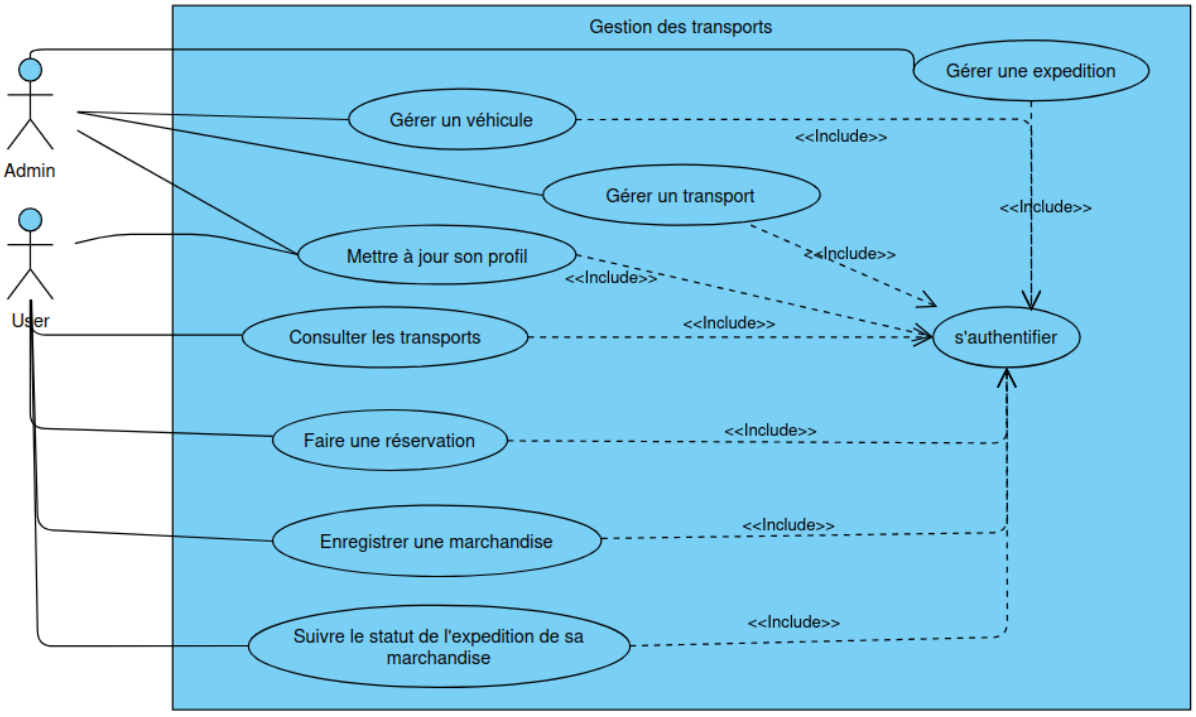


FIGURE 2.1 : Diagramme de cas d'utilisation

2.4.2 Diagramme de classes

Le diagramme de classes est un schéma utilisé en génie logiciel pour représenter les classes, les interfaces et leurs relations au sein du système. Il fait partie de la partie statique de UML, se concentrant sur la structure du système sans prendre en compte les aspects temporels et dynamiques.

Une classe regroupe des fonctions et des données (attributs) liées par un même domaine sémantique. Dans la programmation orientée objet, les classes servent à modéliser un programme en divisant une tâche complexe en plusieurs sous-tâches plus simples.

La figure ci après présente le diagramme de classe de notre système.

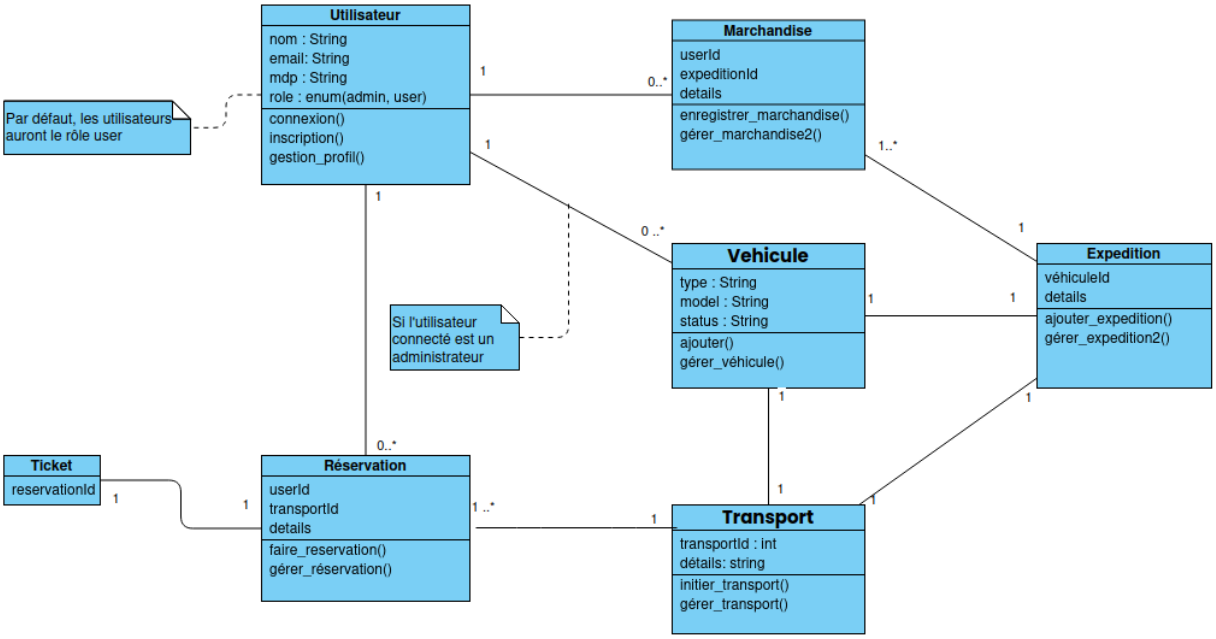


FIGURE 2.2 : Diagramme de classe

2.4.3 Diagramme de séquence

Le diagramme de séquence illustre les interactions entre les objets dans le cadre d'un scénario défini, tel que décrit dans un diagramme de cas d'utilisation.

La dimension verticale du diagramme représente l'écoulement du temps, permettant de visualiser la chronologie des actions et de spécifier la création et la destruction des objets. Les périodes d'activité des objets sont représentées par des rectangles, et les échanges entre ces objets sont illustrés par des messages.

La figure ci-après présente quelques diagrammes de séquence de notre système.

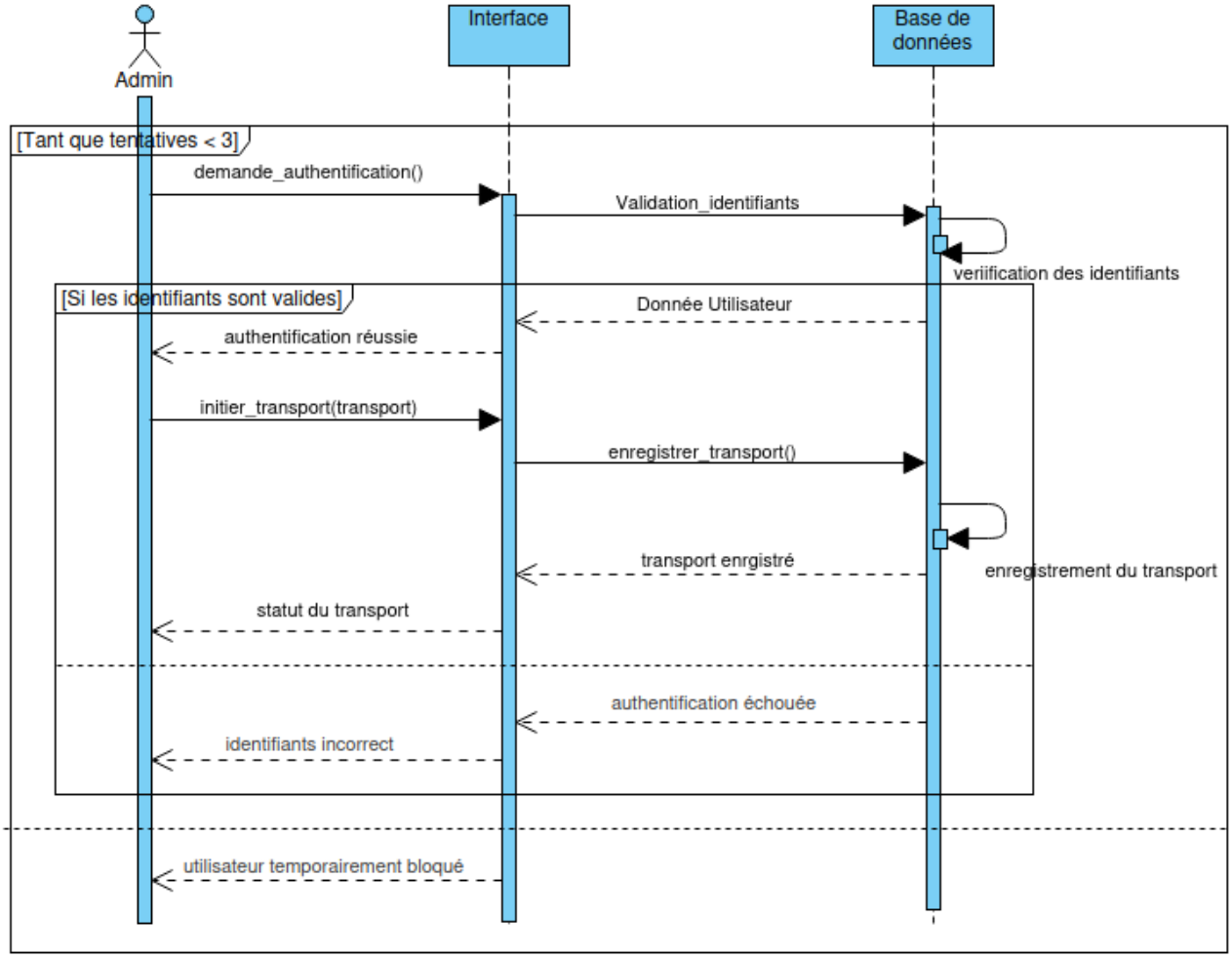


FIGURE 2.3 : Diagramme de séquence d'ajout des transports

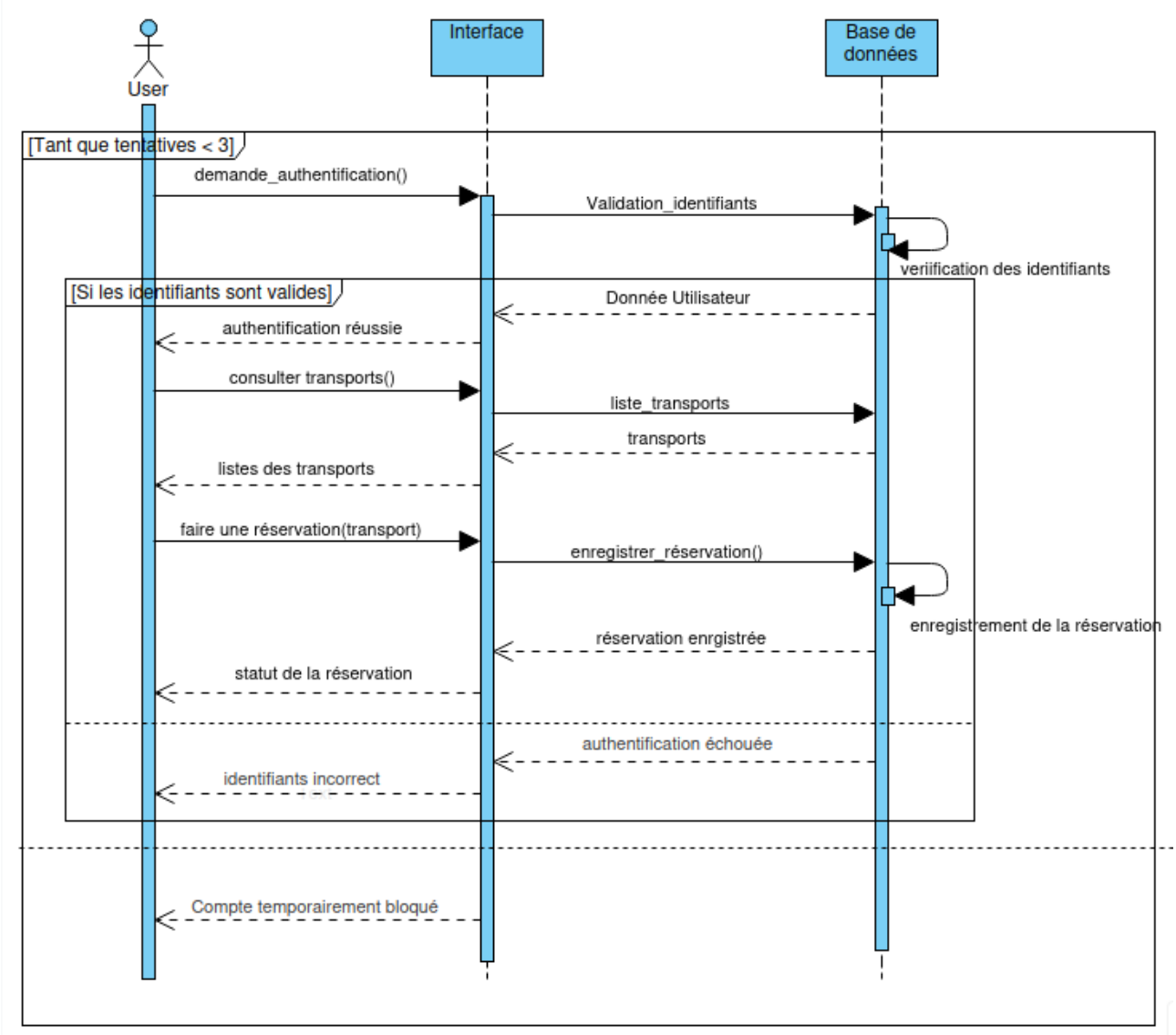


FIGURE 2.4 : Diagramme de séquence de réservation de transports

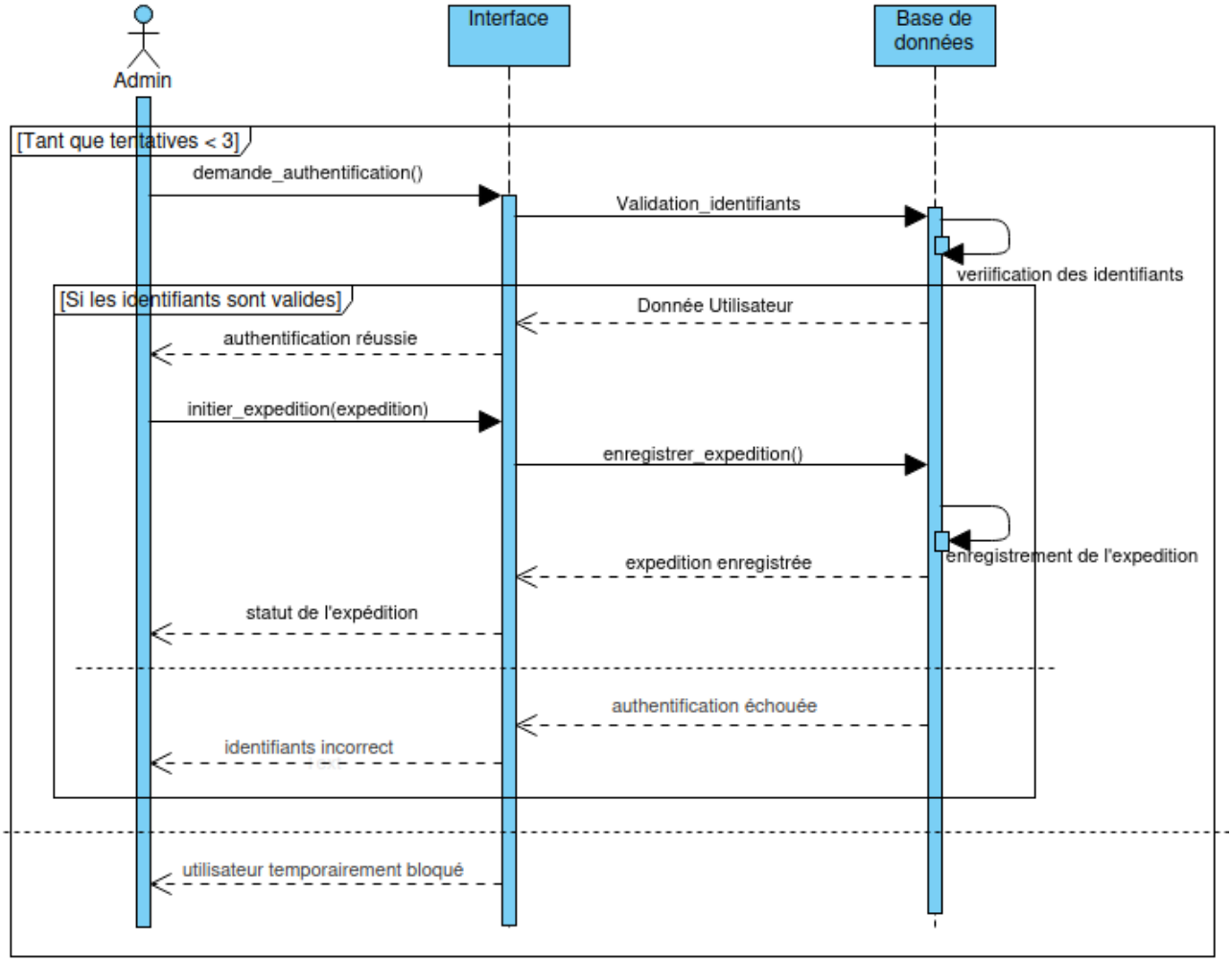


FIGURE 2.5 : Diagramme de séquence des expéditions

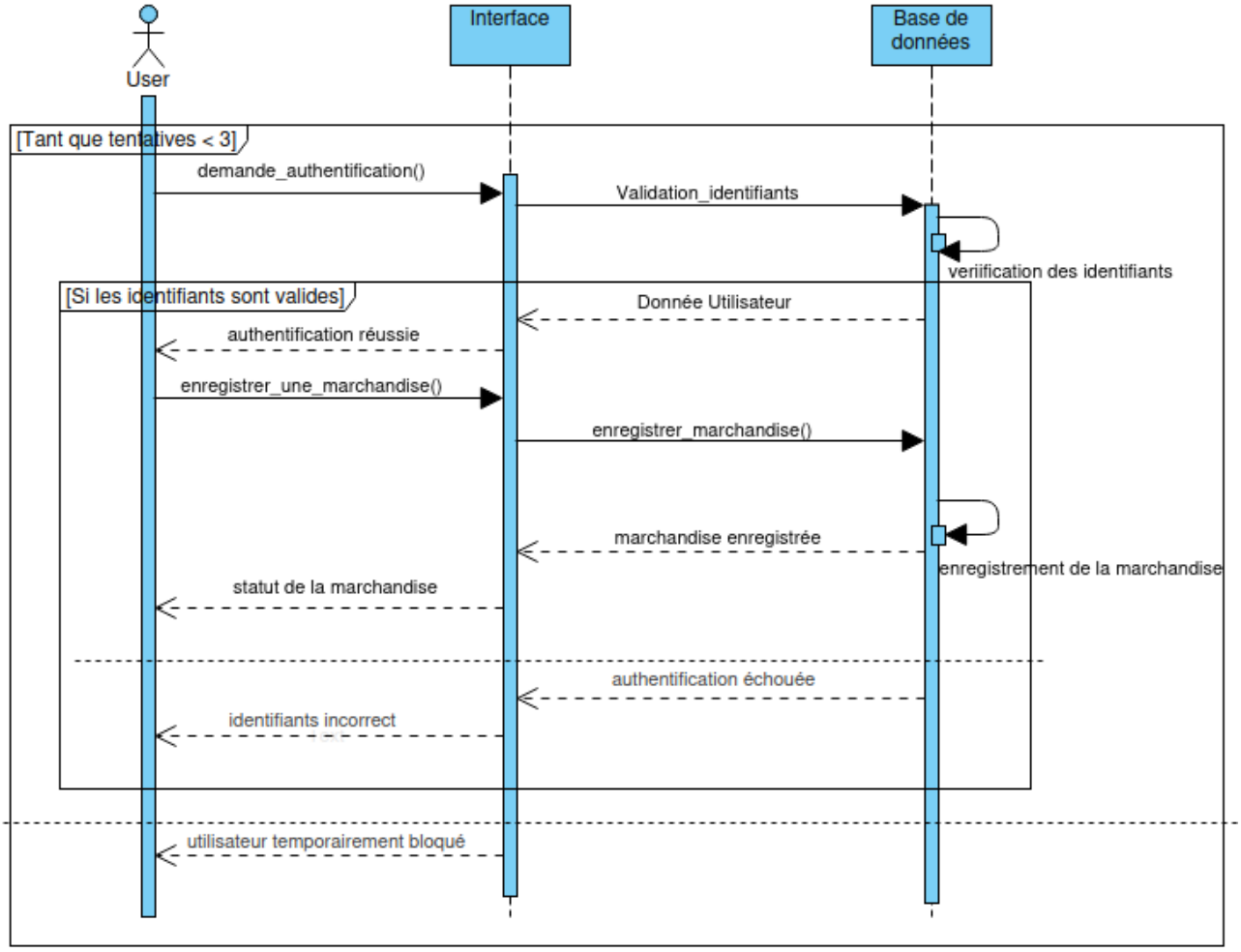


FIGURE 2.6 : Diagramme de séquence des marchandises

Conclusion

Dans ce chapitre, nous avons exposé les besoins du système, les différents outils et techniques utilisés pour la réalisation du système ainsi que la conception du système sous forme de diagrammes UML. Le chapitre suivant exposera la solution proprement dite ainsi que ses limites.

Chapitre 3

Résultats & discussion

Introduction

Eu égard aux outils et technologies, ainsi qu'à la modélisation décrite dans le chapitre précédent, nous avons procédé à la réalisation de l'application. Dans ce chapitre, nous détaillerons l'application en présentant ses différentes fonctionnalités, ainsi que ses limites et insuffisances.

3.1 Présentation de l'application

L'image ci-après représente le logo de notre application :



FIGURE 3.1 : LogiTrack

3.1.1 Ressources

Les ressources de l'application sont accessibles via les liens ci-après.

- **Documentation** : https://github.com/Aqui05/SGTE/blob/frontend/doc/documentation.md
- **Base de données** : https://github.com/Aqui05/SGTE/blob/frontend/doc/modélisation.pdf
- **Frontend** : https://github.com/Aqui05/SGTE
- **Backend** : https://github.com/Aqui05/SGTE/tree/backend

3.1.2 Gestion des utilisateurs

Avant d'effectuer toutes actions, l'utilisateur devra se connecter. Les images ci-après montrent les pages d'authentifiction.

NB : Rappelons que les utilisateurs du système sont classifiés par rôle : les **administrateurs** et les **simples utilisateurs**.

- Inscription

  L'utilisateur devra renseigner les informations le concernant pour créer son profil : tel que son nom, son adresse mail et son mot de passe.

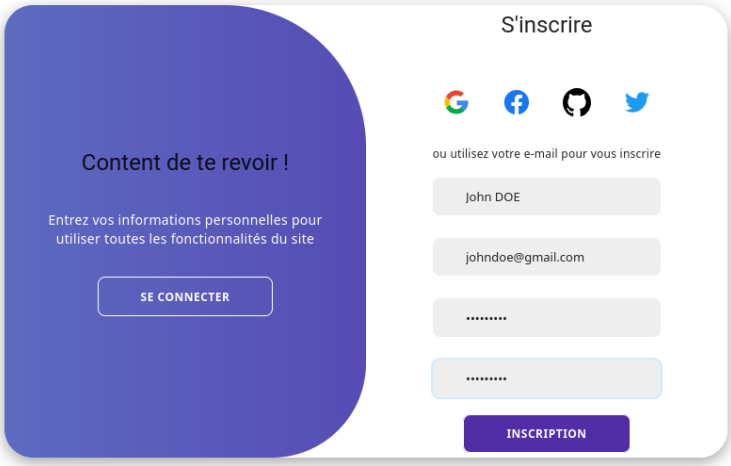


FIGURE 3.2 : Inscription

- Connexion

  L'utilisateur est invité à saisir son adresse mail et son mot de passe pour se connecter.

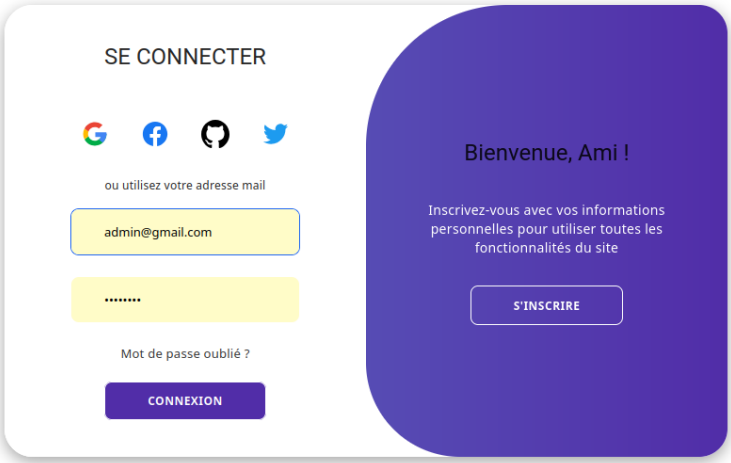


FIGURE 3.3 : Connexion

3.1.3 Quelques fonctionnalités pour l'administrateur

3.1.3.1 Gestion des véhicules

NB : Les véhicules sont considérés dans le système comme les moyens de transports. Que ce soit de type routier, maritime ou même aérien.

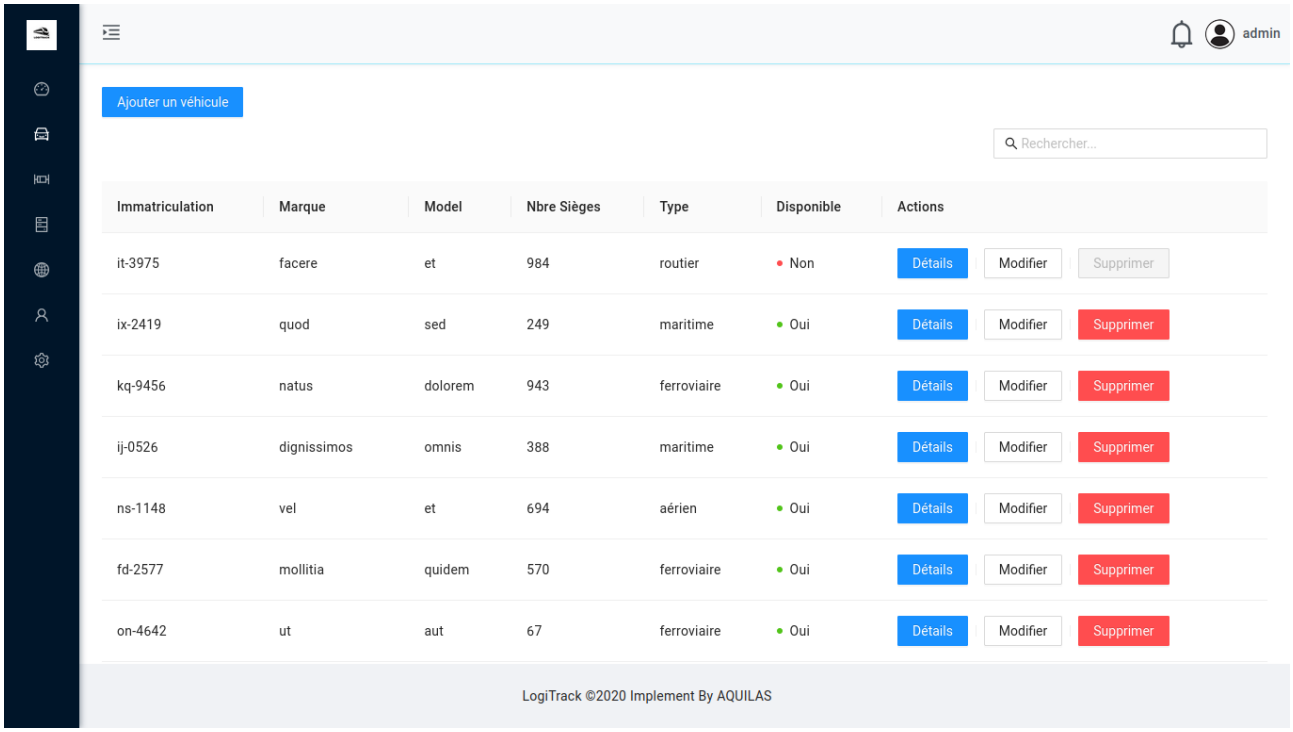


FIGURE 3.4 : Liste des véhicules

- L'administrateur sera en mesure d'ajouter un nouveau véhicule dans le système en appuyant sur le bouton **Ajouter un véhicule**.
- Il pourra modifier les informations du véhicule ou le supprimer en appuyant sur les boutons correspondants.
- Il pourra également obtenir plus de détails sur chaque véhicule en appuyant sur le bouton **Détails** (l'historique de l'utilisation du véhicule, par exemple).

3.1.3.2 Gestion des transports

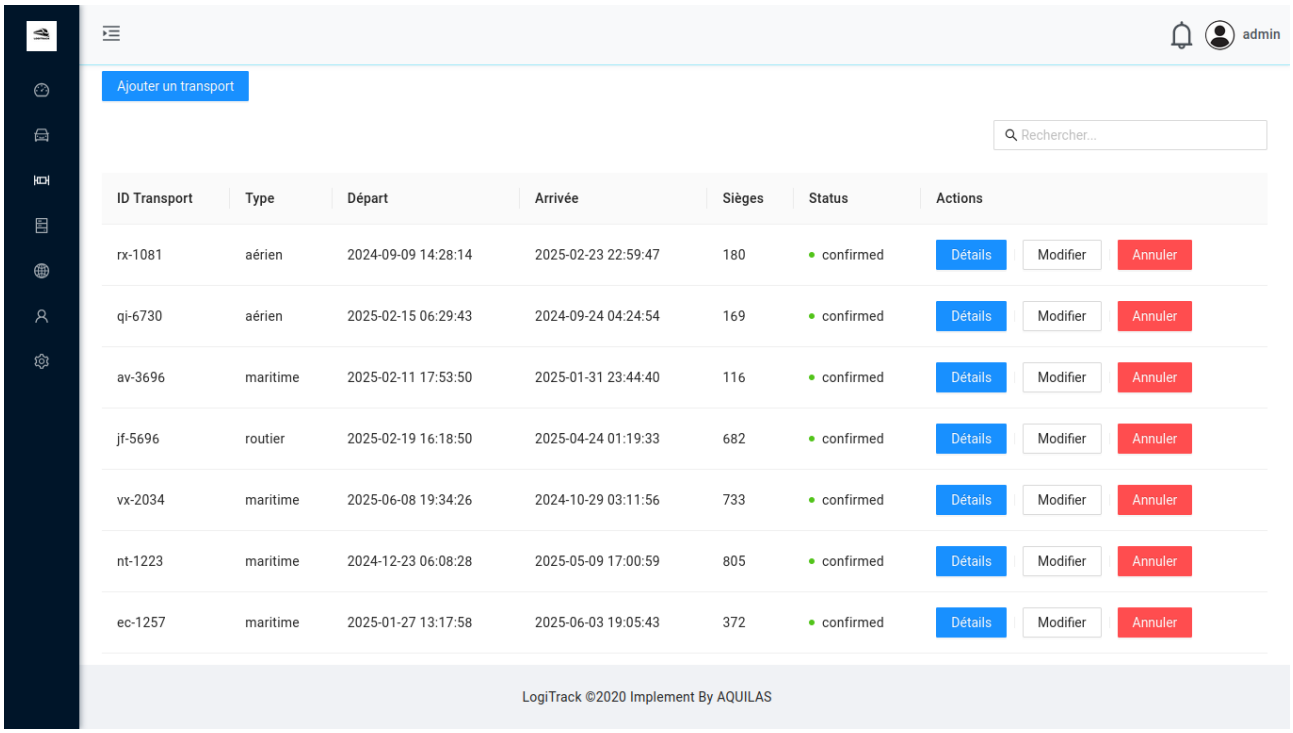


FIGURE 3.5 : Liste des transports

Ceci est l'une des fonctionnalités clés du système. A ce niveau , l'administrateur peut voir la liste de tous les transports initiés. IL peut également rechercher , ajouter un transport ou encore voir les détails et supprimer un transport spécifique.

3.1.3.3 Gestion des expéditions

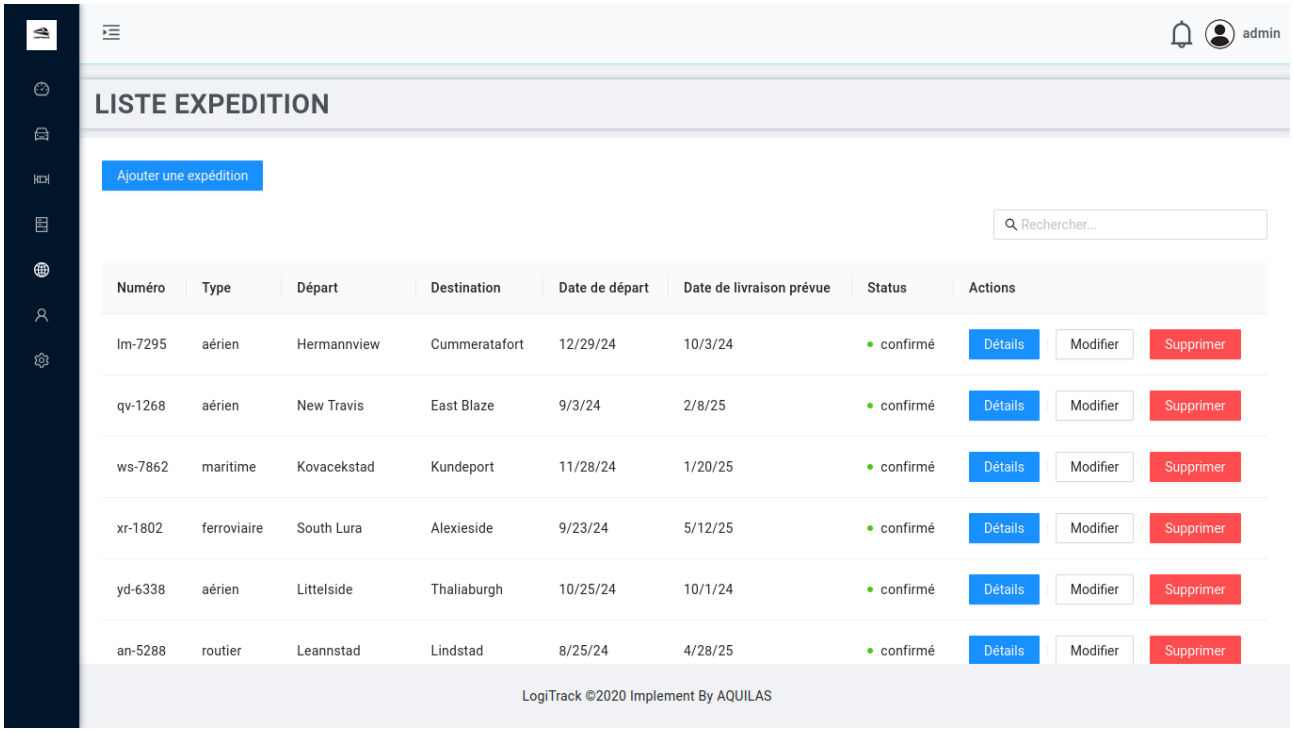


FIGURE 3.6 : Liste des expéditions

Cette page permet à l'administrateur de gérer entièrement les expéditions. Elle récapitule les expéditions et offre la possibilité d'ajouter de nouvelles expéditions ou d'annuler une expédition.

3.1.4 Quelques fonctionnalités pour l'utilisateur

3.1.4.1 Gestion des transports

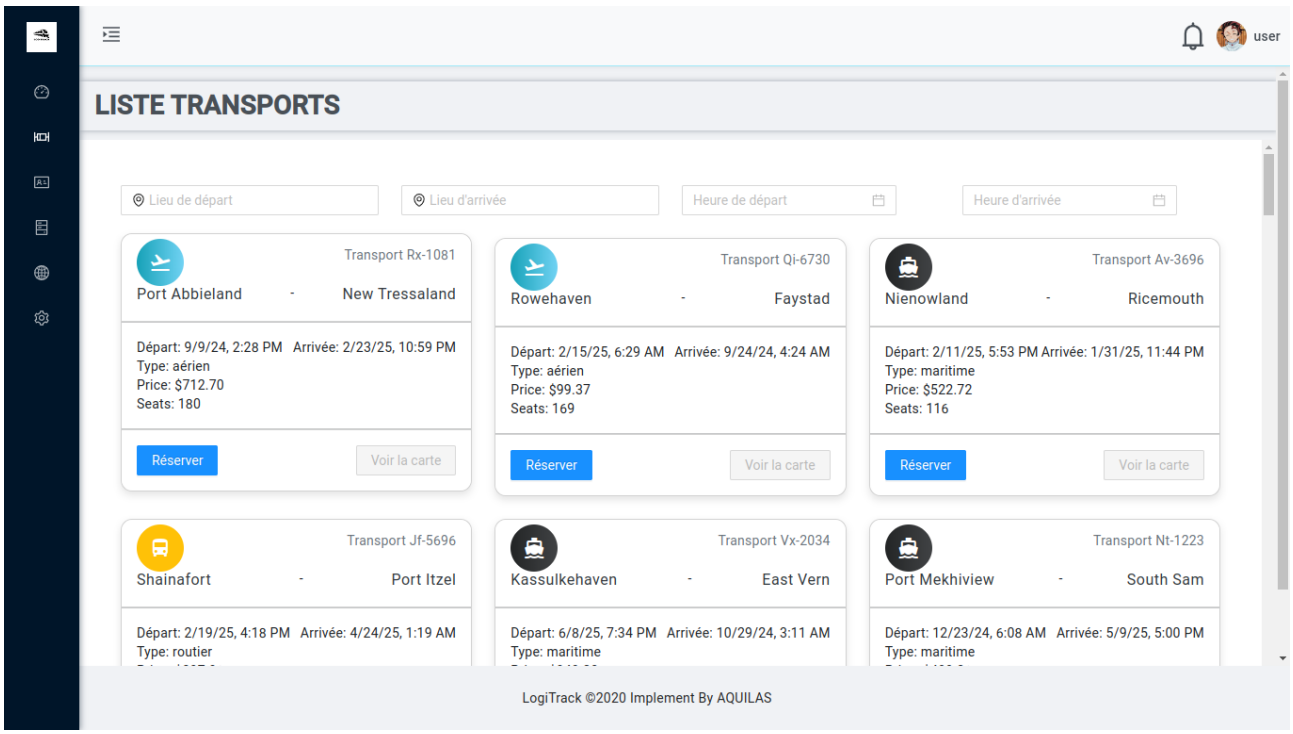


FIGURE 3.7 : Listes des transports

Cette page affiche les différents transports initiés par l'administrateur qui n'ont pas encore commencé. L'utilisateur peut rechercher un transport spécifique en saisissant le lieu de départ, le lieu d'arrivée et/ou la date.

En cliquant sur le bouton "Réserver", l'utilisateur sera redirigé vers la page de réservation, où il peut confirmer sa réservation.

Pour confirmer la réservation, l'utilisateur devra effectuer le paiement. Une fois le paiement validé, il pourra télécharger son ticket.

3.1.4.2 Gestion des marchandises

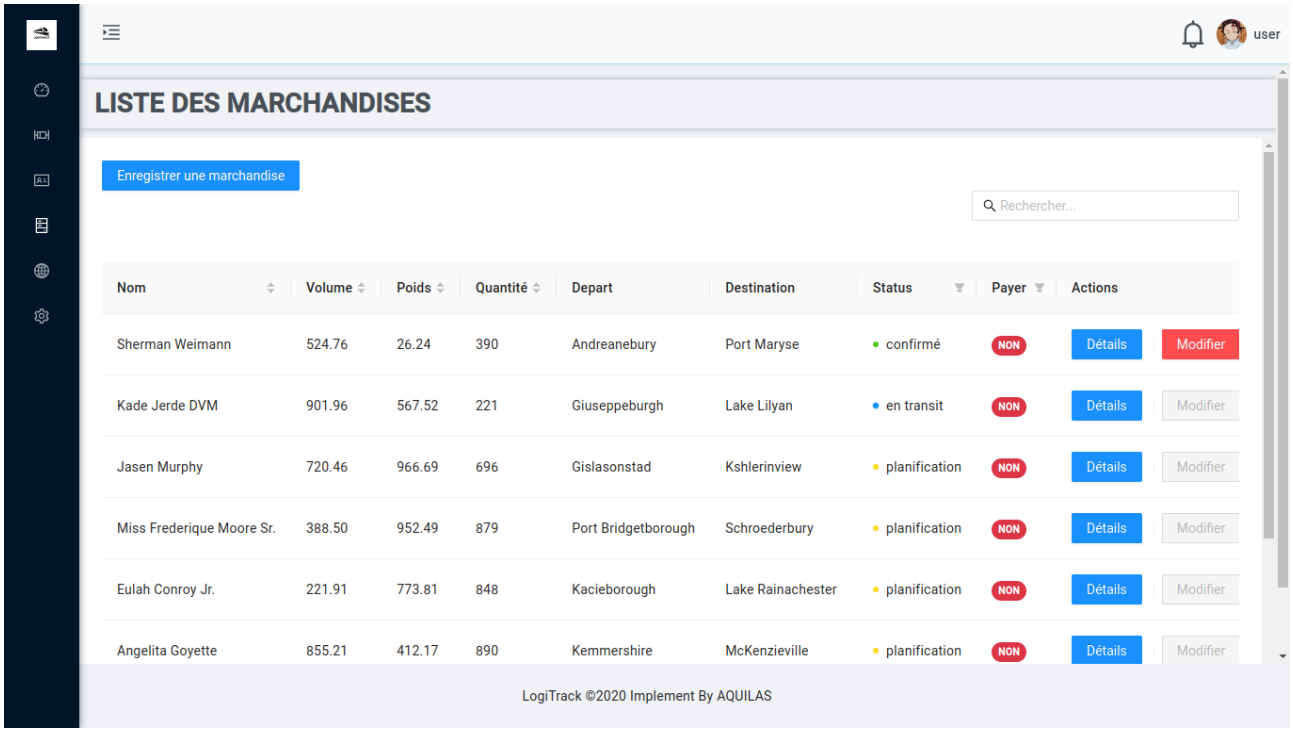


FIGURE 3.8 : Listes des marchandises

Cette page permet à l'utilisateur de visualiser, gérer et mettre à jour ses marchandises.

En cliquant sur le bouton 'Modifier', l'utilisateur accède à une page de modification des détails de la marchandise sélectionnée. Il peut également supprimer une marchandise via le bouton 'Supprimer'. À noter que la suppression n'est possible que si l'administrateur n'a pas encore lancé l'expédition de la marchandise.

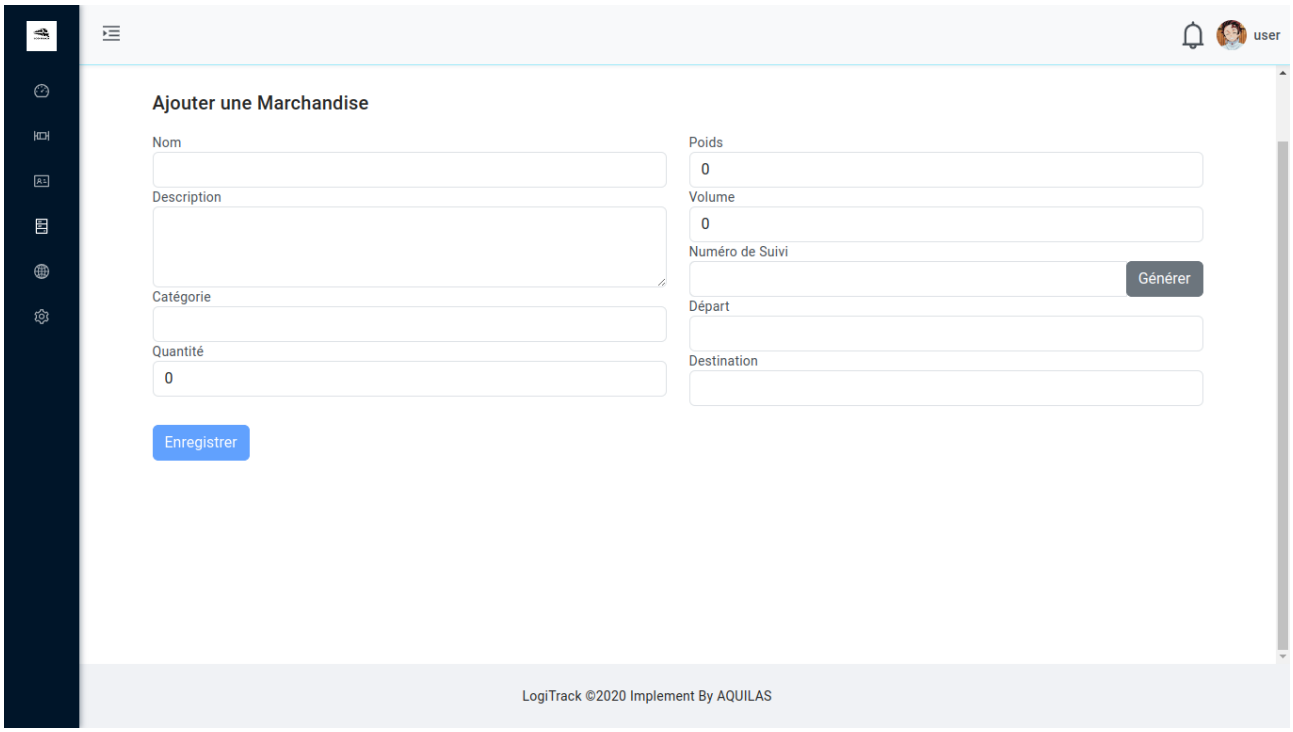


FIGURE 3.9 : Enregistrer une marchandise

La page d'enregistrement des marchandises permet à l'utilisateur d'ajouter de nouvelles marchandises au système en renseignant les informations nécessaires, telles que la description, la quantité, le poids, le volume, le point de départ et la destination. Une fois les informations saisies, l'utilisateur peut cliquer sur "Enregistrer" pour valider l'ajout de la marchandise. Un numéro de suivi peut également être généré automatiquement.

3.2 Discussion

Au regard des résultats obtenus, nous pouvons affirmer que les différents objectifs visés ont été atteints. À l'issue de notre travail, nous avons réalisé une plateforme web dédiée à la gestion des transports de biens et de personnes.

Elle est extensible et destinée à évoluer avec le temps. De nouvelles fonctionnalités pourront donc être ajoutées sans nécessiter de modifications majeures sur la modélisation en place. La plateforme est sécurisée grâce aux atouts du framework Laravel et dispose d'une interface interactive orientée utilisateur, grâce aux propriétés du framework Angular.

Cependant, dans la pratique, notre application présente certaines insuffisances et limites :

- Seuls les tests unitaires ont été effectués lors du développement.
- L'application n'a pas encore été déployée sur un serveur distant.
- Les paiements sont effectués de manière statique sur la plateforme : lorsque le prix est déterminé, l'utilisateur ne fait que confirmer pour procéder au paiement ; il n'y a pas de système efficace de gestion de paiements.

- Les informations concernant une marchandise sont saisies par l'utilisateur, ce qui pourrait poser problème si les données fournies ne sont pas exactes.
- Absence d'un système de gestion avancée des itinéraires et des transports.

Pour faire évoluer notre application, nous envisageons d'ajouter de nouvelles fonctionnalités telles que :

- Intégration des API Google Maps : pour une gestion et un suivi en temps réel des itinéraires.
- Une gestion avancée des véhicules incluant :
  * La géolocalisation en temps réel des véhicules.
  * Un diagnostic et une maintenance des véhicules pour prévenir d'éventuels accidents de circulation et garantir la disponibilité continue des véhicules.
  * Des rapports journaliers sur le fonctionnement et les opérations de chaque véhicule, générés automatiquement.
  * Des alertes automatiques pour les opérations de maintenance des véhicules.
- Une gestion efficace de la logistique des marchandises :
  * La gestion des modules de stockage des marchandises et des mesures (poids, quantité, volume) du côté administrateur.
  * La gestion électronique des documents liés aux marchandises, tels que les bons de livraison, les factures, etc.
- Génération de rapports détaillés sur les transports, les réservations et les expéditions.
- L'intégration d'un système dynamique de gestion des paiements (tels que l'utilisation des API de PayPal/Stripe ou Mobile Money).

Conclusion

Dans ce chapitre, nous avons exposé le fruit de notre travail. Les résultats obtenus à l'issue de la phase de développement ont été présentés sous forme de captures d'écran. Nous avions également mis l'accent sur les différents insuffisances et limites de l'application et notifier les fonctionnalités qui pourront y être intégrées dans un cadre plus large.

Conclusion Générale

Le développement de notre application web de gestion des transports de biens et de personnes représente une avancée significative dans la manière dont les opérations de transport sont gérées et optimisées. Grâce à l'intégration des fonctionnalités de gestion des réservations, de suivi des expéditions, de gestion des marchandises et de gestion des véhicules, l'application offre une solution complète pour répondre aux défis logistiques rencontrés par les entreprises de transport.

Au cours du projet, nous avons abordé plusieurs aspects critiques du transport, en tenant compte des exigences des utilisateurs finaux. L'optimisation des coûts, la sécurisation des expéditions et l'amélioration de la visibilité sur les stocks font partie des bénéfices directs apportés par cette solution. En outre, la gestion des rôles et des utilisateurs garantit une utilisation sécurisée et personnalisée de l'application, adaptée aux différents profils de ses utilisateurs [1].

Les défis techniques rencontrés, tels que l'intégration des différents modules, l'optimisation des performances et la sécurisation des données, ont été surmontés grâce à une méthodologie rigoureuse. Ce processus nous a permis d'acquérir une expertise précieuse dans le domaine du développement d'applications web et d'affiner nos compétences dans l'utilisation des technologies modernes de développement web [5].

En conclusion, cette application web n'est pas seulement un outil pratique pour la gestion des transports routiers, mais aussi une plateforme évolutive qui pourra s'adapter aux futurs besoins des utilisateurs. Ce projet marque ainsi une étape importante dans l'amélioration des services de transport, avec des perspectives prometteuses pour l'avenir.

Bibliographie

[1] Stuart Cole. *Transport Management and Policy : A Practical Introduction*. Kogan Page, London, UK, 2018.

[2] Carlos Coronel and Steven Morris. *Database Systems : Design, Implementation, and Management*. Cengage Learning, 12th edition, 2016.

[3] Martin Fowler. *UML Distilled : A Brief Guide to the Standard Object Modeling Language*. Addison-Wesley Professional, 3rd edition, 2003.

[4] International Organization for Standardization. Ergonomie de l'interaction homme-système - partie 210 : Conception centrée sur l'humain pour les systèmes interactifs, 2019.

[5] Martin Kleppmann. *Designing Data-Intensive Applications*. O'Reilly Media, Sebastopol, CA, 2017.

[6] Brent Laster. *Practical Git and GitHub*. O'Reilly Media, 2016.

[7] Francesco Malatesta. *Pro Laravel*. Apress, New York, NY, 2015.

[8] Robert C. Martin. *Clean Architecture : A Craftsman's Guide to Software Structure and Design*. Pearson Education, Upper Saddle River, NJ, 2017.

[9] Malcolm McDonald. *Web Security for Developers*. No Starch Press, 2020.

[10] Shyam Seshadri. *Angular : Up and Running*. O'Reilly Media, Sebastopol, CA, 2018.

Webographie

[11] Oracle (2024). “Les systèmes de gestion du transport (TMS)”. https://www.oracle.com/ch-fr/scm/logistics/transportation-management/what-is-transportation-management-system/. [Consulté le 30 juillet 2024].

[12] Fulfillment Hub USA (2024). “L’importance des systèmes de gestion des transports dans le secteur logistique”. https://fulfillmenthubusa.com/limportance-des-systemes-de-gestion-des-transports-dans-le-secteur-logistique/. [Consulté le 30 juillet 2024].

[13] Ideematic (2024). “Définition d’une application web”. https://www.ideematic.com/dictionnaire-digital/application-web/. [Consulté le 1 août 2024].

[14] RingCentral (2024). “Qu’est-ce qu’une application Web ? Définition et exemples”. https://www.ringcentral.com/fr/fr/blog/glossaire/application-web/. [Consulté le 1 août 2024].

[15] HelloDarwin (2024). “Les différents types d’applications web”. https://hellodarwin.com/fr/blogue/types-applications-web. [Consulté le 1 août 2024].

[16] Trustpilot (2024). “Avis utilisateurs sur les solutions de transport et logistique”. https://www.trustpilot.com. [Consulté le 28 Septembre 2024].

[17] SAP (2024). “Les fonctionnalités des TMS”. https://www.sap.com/france/products/scm/transportation-logistics/what-is-a-tms.html. [Consulté le 3 août 2024].

[18] Tuleap (2024). “Comprendre la méthode agile Scrum en 10 minutes”. https://www.tuleap.org/fr/agile/comprendre-methode-agile-scrum-10-minutes. [Consulté le 4 août 2024].

[19] Wikipédia (2024). “Laravel”. https://fr.wikipedia.org/wiki/Laravel. [Consulté le 5 août 2024].

[20] Lemon Interactive (2024). “Framework Laravel : avantages et inconvénients”. https://www.lemon-interactive.fr/actualites/developpement/framework-laravel-avantages-inconvenients/. [Consulté le 5 août 2024].

[21] Wikipédia (2024). “Angular”. https://fr.wikipedia.org/wiki/Angular. [Consulté le 5 août 2024].

[22] SQL.sh (2024). “MySQL”. https://sql.sh/sgbd/mysql. [Consulté le 6 août 2024].

[23] Truspilot : Happy Post (2024). “Avis des utilisateurs de Happy Post”. https://www.trustpilot.com/review/happy-post.com. [Consulté le 5 août 2024].

[24] Truspilot : BTX(2024). “Avis des utilisateurs de BTX”. https://www.trustpilot.com/review/benin ticketxpress.com. [Consulté le 5 août 2024].

[25] HTML (2024). “Le langage HTML”. https://developer.mozilla.org/fr/docs/Web/HTML. [Consulté le 5 août 2024].

[26] CSS (2024). “Le langage CSS”. https://developer.mozilla.org/fr/docs/Web/CSS. [Consulté le 5 août 2024].

[27] Bootstrap (2024). “Le Bootstrap”. https://fr.wikipedia.org/wiki/Bootstrap_(framework). [Consulté le 5 août 2024].

Table des matières

Dédicace ii

Remerciements iii

Résumé iv

Abstract v

Liste des Figures vi

Liste des acronymes vii

Glossaire 1

Introduction 2

1 Revue de littérature 4
Introduction . . . 4
1.1 Clarification Conceptuelle . . . 4
1.1.1 Application Web . . . 4
1.1.1.1 Définition . . . 4
1.1.1.2 Fonctionnement d'une application web . . . 4
1.1.1.3 Type d'application web . . . 5
1.1.2 TMS . . . 5
1.1.2.1 Définition . . . 5
1.1.2.2 Fonctionnalités d'un TMS . . . 6
1.2 Application Web et TMS . . . 6
1.3 Les solutions existantes . . . 6
1.3.1 Gozem . . . 7
1.3.1.1 Présentation . . . 7
1.3.1.2 Limites de Gozem . . . 7
1.3.2 Benin Ticket Express (BTX) . . . 7
1.3.2.1 Présentation . . . 7
1.3.2.2 Limites de BTX : . . . 7
1.3.3 Flash Réservation . . . 7
1.3.3.1 Présentation . . . 7
1.3.3.2 Limites de Flash Réservation . . . 7
1.3.4 Happy Post . . . 8
1.3.4.1 Présentation . . . 8

1.3.4.2 Limites de Happy Post . . . 8
1.3.5 Kobo360 . . . 8
1.3.5.1 Présentation . . . 8
1.3.5.2 Limites de Kobo360 . . . 8
1.4 Avantages différenciateurs de notre solution . . . 8
Conclusion . . . 9

2 Analyse, conception et choix techniques **10**
Introduction . . . 10
2.1 Analyse des besoins . . . 10
2.1.1 Besoins fonctionnels . . . 10
2.1.2 Besoins non fonctionnels . . . 11
2.2 Méthode de développement . . . 12
2.3 Choix des outils techniques . . . 12
2.3.1 Frameworks et languages de programmation . . . 12
2.3.2 Choix du SGBD . . . 13
2.3.3 Logiciels utilisés . . . 13
2.4 Modélisation & Conception . . . 14
2.4.1 Diagramme de cas d'utilisation . . . 14
2.4.2 Diagramme de classes . . . 14
2.4.3 Diagramme de séquence . . . 15
Conclusion . . . 19

3 Résultats & discussion **20**
Introduction . . . 20
3.1 Présentation de l'application . . . 20
3.1.1 Ressources . . . 20
3.1.2 Gestion des utilisateurs . . . 21
3.1.3 Quelques fonctionnalités pour l'administrateur . . . 22
3.1.3.1 Gestion des véhicules . . . 22
3.1.3.2 Gestion des transports . . . 23
3.1.3.3 Gestion des expéditions . . . 24
3.1.4 Quelques fonctionnalités pour l'utilisateur . . . 25
3.1.4.1 Gestion des transports . . . 25
3.1.4.2 Gestion des marchandises . . . 26
3.2 Discussion . . . 27
Conclusion . . . 28

Conclusion **29**

Bibliographie **30**

Webographie **31**

Table des matières **33**