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.net Courriel : contact@ifri.uac.bj

MÉMOIRE

pour l'obtention du

Diplôme de Licence en Informatique

Option : Système d'Information et Réseaux Informatique

Présenté par :
Delkaël Mankponsè **DJOGBEDE**

Création d'une Application de Géolocalisation d'Atelier de Mécanique à Cotonou

Sous la supervision :
Mr Jean-Eudes **AZOMBAKIN**

Membres du jury :

HOUNDJI Vinansetan Ratheil	Docteur	Président
GOHOUE Rose	Ingénieur	Examinatrice
AZOMBAKIN Jean-Eudes	Ingénieur	Rapporteur

Année Académique : **2023-2024**

Sommaire

Dédicace ii

Remerciements iii

Résumé iv

Abstract v

List of Figures vi

Liste des acronymes vii

Introduction 1

1 Revue de littérature 3

2 Modélisation et conception 7

3 Résultats et Discussion 18

Conclusion 29

Bibliographie 30

Bibliographie 30

Table des matières 31

Dédicace

À la mémoire de ma mère, **Marie ASSONGBA**, dont l'amour, les sacrifices et les valeurs continuent de m'inspirer chaque jour.

À mon père, **Anatole DJOGBEDE**, pour son soutien indéfectible, ses encouragements constants et sa foi en mon avenir.

À ma famille, pour leur patience, leur compréhension et leur bienveillance tout au long de ce parcours.

Remerciements

Bien consciente que ces quelques mots sont loin de refléter la grandeur du rôle précieux que vous avez joué dans la réalisation de ce travail, mes remerciements les plus sincères vont à l'endroit de :

- **Professeur Eugène C. EZIN**, Directeur de l'Institut de Formation et de Recherche en Informatique (IFRI) ;
- **Professeur EDAH Gaston**, Directeur adjoint de l'IFRI.
- **Monsieur Jean-eudes AZOMBAKIN**, mon encadreur, pour sa disponibilité, sa patience, son suivi et la confiance qu'il a placée en nous ;
- **Mme Anychat ABDALLAH**, Directrice de Tics Master Sarl
- **Tous les enseignants** de notre institut, pour avoir accepté de partager avec dévouement une partie de leurs connaissances avec nous ;
- **Toute ma famille, mes tantes et oncles, mes frères et sœurs**, ainsi que tous mes amis qui m'ont toujours soutenue et encouragée dans mes choix d'étude ;
- **Tous mes amis et camarades de promotion**, avec qui j'ai collaboré tout le long de ce parcours.

Résumé

Ce projet consiste en la conception et la réalisation de MecaZone, une application web de géolocalisation [1] d'ateliers mécaniques. Son objectif principal est de faciliter pour les utilisateurs la recherche d'ateliers de réparation automobile situés à proximité, tout en tenant compte de critères tels que la localisation actuelle, les services spécifiques offerts et les avis des clients. L'application propose une interface intuitive permettant aux utilisateurs de consulter les profils des ateliers, de prendre des rendez-vous en ligne, et de bénéficier d'une carte interactive pour une expérience utilisateur optimale.Dans le cadre de ce projet, nous avons utilisé des technologies modernes telles que Nuxt.js pour le front-end et Node.js pour le back-end, garantissant ainsi une application performante et réactive. Une attention particulière a été portée à la sécurité, notamment en implémentant une authentification sécurisée, des contrôles d'accès selon le rôle de l'utilisateur, et des mesures de protection contre les attaques courantes (XSS, CSRF, injections).La plateforme est pensée pour être évolutive, avec des perspectives d'ajout de nouvelles fonctionnalités telles que l'évaluation plus détaillée des services, une gestion améliorée des ateliers et la possibilité pour les ateliers d'offrir des promotions. MecaZone se positionne ainsi comme un outil innovant, contribuant à une meilleure connectivité entre les automobilistes et les professionnels de la réparation automobile. Resume en francais

Mots clés : géolocalisation, application web, ateliers mécaniques, Nuxt.js, Node.js, sécurité, authentification, carte interactive, expérience utilisateur.

Abstract

This project presents the design and development of MecaZone, a web application for locating nearby mechanical workshops, providing users with a streamlined and practical tool to access automotive services based on various search criteria. By integrating features like search by location, specific services offered, and user reviews, the application enables a highly personalized experience. Users can view workshop profiles, make appointments online, and interact with a dynamic map for easy navigation. MecaZone enhances accessibility for car owners by helping them quickly find reliable and competent mechanical workshops. With a strong focus on usability, security, and service quality, this project offers a comprehensive solution that simplifies the maintenance process for car users while fostering trust and convenience.

Key words: MecaZone, mechanical workshop localization, web application, service accessibility, user-centered design, secure access, interactive map, automotive services

List of Figures

2.1 Diagramme de cas d'utilisation 9
2.2 Diagramme de cas classe 10
2.3 Diagramme de séquence du cas «Inscription» 12
2.4 Diagramme de séquence du cas «Connexion» 13

3.1 Page d'acceuil 19
3.2 Inscription utilisateur 20
3.3 Inscription Atelier 21
3.4 Page de connexion 22
3.5 Interface profil Atelier 23
3.6 Interface de recherche 24
3.7 Interface de résultat de recherche d'Atelier 25
3.8 Interface d'offre de service 26
3.9 Interface administration(gestion des Ateliers) 27

Liste des acronymes

BLE :

Bluetooth Low Energy 3

CSRF :

Cross-Site Request Forgery iv, 17

GPS :

Global Positioning System 3

IHM :

Interface Homme-Machine 11

MERISE :

Méthode d'Étude et de Réalisation Informatique dans les Systèmes d'Entreprise 7

SQL :

Structured Query Language 17

UML :

Unified Modeling Language 7

XSS :

Cross-Site Scripting iv, 17

Introduction Générale

Contexte

La ville de Cotonou, capitale économique du Bénin, est une métropole en pleine expansion où les motos constituent un moyen de transport essentiel. Cependant, les pannes de moto en cours de route sont fréquentes et il est souvent difficile pour les motocyclistes de trouver rapidement un atelier de mécanique à proximité.

Problématique

Lorsqu'un motocycliste tombe en panne en plein trajet, il peut se retrouver dans une situation stressante et potentiellement dangereuse, surtout s'il ne connaît pas la localisation des ateliers de mécanique proches. Actuellement, il n'existe pas de solution numérique efficace pour localiser rapidement un mécanicien disponible à proximité.

Justificatif

L'application proposée vise à combler ce vide en fournissant une solution de géolocalisation des ateliers de mécanique à Cotonou. Elle permettra aux motocyclistes de gagner du temps, de réduire le stress lié aux pannes et d'améliorer la sécurité en facilitant la localisation rapide d'un mécanicien.

Objectif

L'objectif principal de ce projet est de développer une application web et mobile qui permettra aux utilisateurs de :

- Localiser en temps réel les ateliers de mécanique les plus proches.
- Obtenir des informations détaillées sur chaque atelier (heures d'ouverture, services offerts, avis des clients, etc.).

- Évaluer et laisser des commentaires sur les services des mécaniciens.

Organisation du document

Le présent document est structuré en trois chapitres. Le premier chapitre présente les solutions existantes et l'intérêt de notre solution par rapport à celles-ci. Le deuxième chapitre expose notre solution à travers sa modélisation, son fonctionnement et les choix techniques liés à sa réalisation.. Enfin, le troisième chapitre détaillera la mise en œuvre du projet.

Chapitre 1

Revue de littérature

Introduction

Il est primordial de réaliser une phase de l'état de l'art, car elle permet d'appréhender les différents concepts fondamentaux du thème d'étude grâce aux travaux menés par d'autres acteurs sur le sujet, et de prendre une position critique par rapport à leurs conclusions. Dans ce chapitre, nous définirons d'abord les mots clés de notre thème de recherche. Ensuite, nous présenterons les recherches connexes réalisées par d'autres acteurs. Enfin, nous conclurons par la présentation de la valeur ajoutée que nous apportons aux solutions existantes.

1.1 Technologies de géolocalisation

GPS : Le GPS (Global Positioning System) reste la technologie de base pour la géolocalisation. Bien que très précis en milieu extérieur, il peut devenir moins efficace dans les environnements urbains denses ou à l'intérieur des bâtiments. Par exemple, dans une ville comme Cotonou, où de nombreux ateliers sont situés dans des zones urbaines complexes, l'intégration du GPS peut être affectée par des bâtiments élevés, des murs épais, et d'autres structures qui bloquent le signal. Cependant, les technologies GPS modernes, notamment les GPS à double fréquence, offrent de meilleures performances dans ces conditions.

Géolocalisation par Wi-Fi et Bluetooth : Ces technologies permettent de compléter le GPS dans les zones où la réception du signal est faible. Le Wi-Fi est couramment utilisé dans les environnements urbains pour identifier la localisation en fonction des points d'accès disponibles. Il est plus efficace pour des positions dans des bâtiments ou des zones ouvertes où le GPS échoue. Bluetooth Low Energy (BLE), via des balises ou des "beacons", peut offrir une localisation très précise dans des environnements intérieurs complexes, permettant par exemple de localiser un atelier mécanique à l'intérieur d'un complexe commercial.

Méthodes hybrides : Une approche combinée utilisant GPS, Wi-Fi et Bluetooth est souvent utilisée dans les applications modernes pour maximiser la précision, surtout dans des zones urbaines. Par exemple, un utilisateur en panne dans une rue de Cotonou pourrait utiliser son smartphone pour se

localiser via GPS, puis la géolocalisation Wi-Fi pourrait affiner cette position lorsqu'il entre dans un bâtiment ou une zone plus dense.

1.2 Services et Applications de Géolocalisation

[2]Google Maps API :L'une des solutions les plus populaires pour la géolocalisation et l'affichage de cartes interactives. Elle permet aux développeurs d'intégrer des cartes interactives dans leurs applications et de fournir des résultats de recherche locaux. Grâce à son API, Google permet la géolocalisation en temps réel et la recherche d'ateliers mécaniques à proximité.

Foursquare :Une autre plateforme de géolocalisation qui a évolué pour devenir un outil très utilisé dans le secteur des services locaux. Foursquare permet aux utilisateurs de rechercher des lieux et de laisser des avis. Cela peut être particulièrement utile dans un contexte comme celui de votre projet, car cela permet de centraliser des informations sur les ateliers mécaniques en plus de proposer des avis et des évaluations par les utilisateurs.

OpenStreetMap (OSM) :Une alternative à Google Maps qui permet une cartographie collaborative. Utilisant les données fournies par les utilisateurs, OSM permet une couverture géographique détaillée, notamment dans des zones moins bien desservies par les services commerciaux traditionnels.

1.3 Limites et Défis de la Géolocalisation

Problèmes de précision :La précision des systèmes de géolocalisation est un problème majeur. Bien que le GPS soit précis à 3-5 mètres dans de bonnes conditions, sa précision peut être réduite dans des environnements urbains denses, des zones couvertes ou à l'intérieur des bâtiments. Le Wi-Fi et les systèmes Bluetooth peuvent améliorer cette précision dans certaines situations.

Consommation d'énergie : Les systèmes de géolocalisation, notamment le GPS, consomment beaucoup d'énergie, ce qui peut être un inconvénient majeur pour les utilisateurs. Des solutions comme les modes de faible consommation ou l'optimisation des services de localisation sont souvent nécessaires.

Protection de la vie privée et sécurité des données : La géolocalisation soulève des préoccupations liées à la confidentialité. Les utilisateurs sont souvent réticents à partager leur localisation en temps réel, ce qui nécessite des garanties concernant la protection des données personnelles et la sécurité des informations sensibles.

1.4 Présentation des solutions existantes

Les automobilistes et personnes en quête de services mécaniques ont traditionnellement recours à plusieurs méthodes pour localiser des ateliers. Le bouche-à-oreille, toujours très répandu, reste une source essentielle de recommandations basées sur les expériences personnelles de l'entourage (amis, famille, collègues). De plus, les annuaires téléphoniques, tels que les Pages Jaunes, sont également utilisés pour trouver des ateliers mécaniques dans les zones locales, bien que cette méthode soit de plus en plus remplacée par des solutions numériques.
Avec l'essor d'Internet, la recherche d'ateliers mécaniques en ligne est devenue courante. Les utilisateurs recherchent des services sur des moteurs de recherche en utilisant des termes comme "atelier mécanique à Cotonou" ou "réparation automobile Cotonou", et consultent souvent des sites d'avis et de comparaison pour évaluer la qualité des services offerts. Cependant, ces solutions en ligne restent limitées par l'absence d'informations centralisées sur la localisation précise des ateliers et les services spécifiques proposés.
À l'international, des applications comme [8]Fixd (États-Unis) et [7]CarAdvise (États-Unis) se sont imposées comme des solutions efficaces dans ce domaine. Fixd utilise la géolocalisation pour localiser les ateliers mécaniques proches, tout en offrant des services additionnels, comme le diagnostic automobile et l'estimation des coûts de réparation. Cette application permet aux utilisateurs d'obtenir des informations détaillées sur les ateliers et de prendre des décisions éclairées. CarAdvise, quant à elle, se distingue par la transparence des prix et la possibilité de réserver des rendez-vous en ligne, offrant une expérience fluide et centralisée aux utilisateurs. Ces applications sont principalement utilisées aux États-Unis.
Ces solutions numériques, bien que populaires en dehors du Bénin, ne sont pas encore largement déployées dans le pays. En effet, les services mécaniques au Bénin restent largement informels, et les technologies de géolocalisation et de gestion des services sont encore peu développées. Cela crée un vide qui pourrait être comblé par une solution numérique similaire à celles existant dans d'autres pays, et qui offrirait une meilleure accessibilité, transparence et efficacité pour les utilisateurs béninois à la recherche de services mécaniques.

1.5 Intérêt de la solution par rapport aux existantes

L'application de géolocalisation des ateliers mécaniques répond à plusieurs des limitations des méthodes traditionnelles de recherche et présente des avantages considérables. Contrairement au bouche-à-oreille et aux annuaires téléphoniques, qui reposent souvent sur des informations fragmentées et nécessitent beaucoup de temps pour obtenir des recommandations et des coordonnées, cette application permet d'accéder rapidement à des données centralisées, fiables et actualisées sur les ateliers mécaniques. Elle élimine ainsi la nécessité de consulter plusieurs sources, simplifiant et accélérant le processus de recherche, ce qui permet aux utilisateurs de gagner un temps précieux.
De plus, l'application résout le problème du manque de détails souvent associé aux avis en ligne ou aux annuaires classiques. Elle fournit des informations exhaustives sur les services proposés, les horaires d'ouverture, ainsi que des évaluations de clients précédents, ce qui aide les utilisateurs à faire des choix plus éclairés et à éviter les mauvaises surprises lorsqu'ils sélectionnent un atelier. Cette

transparence renforce la confiance des utilisateurs, améliorant ainsi leur expérience.
Pour les utilisateurs non familiers avec une région ou en cas de panne dans une zone inconnue, l'application offre une solution rapide pour localiser les ateliers mécaniques les plus proches. Cela minimise non seulement le temps de recherche mais également les désagréments liés aux pannes, en garantissant une prise en charge plus rapide des réparations.

Enfin, l'application propose des fonctionnalités pratiques telles que la prise de rendez-vous en ligne et la gestion des réparations. Cela optimise l'efficacité tant pour les utilisateurs que pour les ateliers mécaniques. En réduisant les déplacements inutiles et en facilitant les interactions entre clients et professionnels, cette solution innovante améliore globalement l'expérience des utilisateurs dans la recherche et la gestion des services mécaniques. Ainsi, l'application se positionne comme une réponse pertinente et moderne aux défis actuels du secteur, en combinant rapidité, précision et facilité d'utilisation.

Conclusion

L'analyse des méthodes traditionnelles de localisation des ateliers mécaniques montre qu'elles offrent des recommandations basées sur l'expérience, mais souffrent de limites en rapidité et en précision, surtout pour les utilisateurs non familiers avec la région. Les solutions numériques, bien qu'utiles, manquent souvent de fiabilité et de détails.
planLe développement d'une application de géolocalisation constitue une réponse efficace à ces défis. Elle vise à simplifier la recherche d'ateliers mécaniques à Cotonou et dans ses environs en fournissant des informations précises et accessibles. Cette approche améliorera l'expérience utilisateur et facilitera le choix des services mécaniques.

Chapitre 2

Modélisation et conception

Introduction

La modélisation des données est une étape clé pour représenter et comprendre les processus métiers d'un logiciel ou d'un système d'information. Elle permet de simplifier et de clarifier le fonctionnement de ces processus, rendant l'information accessible même à un public non technique. En décomposant le système en plusieurs sous-systèmes, on réduit la complexité du projet, transformant ainsi une problématique globale en plusieurs éléments plus simples à gérer. En informatique, UML et MERISE sont deux langages couramment utilisés pour cette modélisation, offrant des outils graphiques efficaces pour illustrer les interactions entre les acteurs et les fonctionnalités du système.

2.1 Comparaison entre UML et MERISE

Dans la modélisation des systèmes d'information, deux méthodes courantes sont MERISE et UML. MERISE est une méthode française utilisée pour modéliser les flux d'information et les bases de données dans les systèmes d'information. Elle repose sur une structuration en plusieurs niveaux (conceptuel, logique, organisationnel) qui est efficace pour la gestion des données et des processus, mais elle est moins flexible pour les applications orientées objets. UML, en revanche, est un langage de modélisation moderne, conçu spécifiquement pour représenter les systèmes orientés objets. Il propose divers diagrammes, tels que les diagrammes de cas d'utilisation et les diagrammes de classes, qui permettent de modéliser les interactions entre les utilisateurs et les composants du système de manière claire et intuitive.

2.2 Choix de la méthode de modélisation

En nous basant sur le comparatif précédent, nous avons choisi d'utiliser UML pour la modélisation de notre projet en raison de ses nombreux avantages. UML est un langage de modélisation standardisé qui permet de représenter efficacement les différentes étapes de la conception orientée objet d'un logiciel. Il prend en charge tout le cycle de vie d'un logiciel, depuis la définition des besoins jusqu'à l'étape de développement. UML propose plusieurs méthodes pour décrire et modéliser les acteurs

du système, le comportement des objets, les flux internes, les opérations, les composants d'implémentation et leurs interactions, ainsi que la structure matérielle et la répartition des objets et des composants. De plus, UML est flexible et peut être modifié en fonction des évolutions du projet.

2.3 Modélisation UML

UML, qui signifie Unified Modeling Language, est un langage de modélisation visuel utilisé pour illustrer des systèmes logiciels ainsi que des processus métiers. Il permet de transmettre de manière claire et précise les idées et les conceptions des systèmes, facilitant ainsi la communication entre les développeurs, les architectes, les clients et les autres parties prenantes impliquées.

2.3.1 Identification des acteurs du système

Un cas d'utilisation représente une série d'actions réalisées par le système pour fournir un résultat observable à un acteur particulier. Ces cas d'utilisation sont identifiés en analysant les diverses interactions entre les acteurs et le système.

Dans notre contexte d'étude, nous pouvons classer les cas d'utilisation de la manière suivante :

- **Client :**
  - Se connecte à l'application.
  - Effectue des recherches d'ateliers mécaniques en fonction des services offerts.
  - Consulte les profils d'ateliers.
  - Contacte un atelier.
  - Donne un avis sur les ateliers.
  - Prend des rendez-vous avec un atelier.
- **Atelier mécanique :**
  - Se connecte à l'application.
  - Offre des services mécaniques.
  - Modifie ses services.
  - Supprime ses services.
  - Consulte la liste des profils d'autres ateliers.
  - Contacte d'autres ateliers.
  - Valide les rendez-vous avec les clients.
- **Administrateur :**
  - Accède à la liste des offres de services proposées par les ateliers.
  - Modifie les offres de services.
  - Supprime les offres de services.
  - Valide les offres de services.
  - Peut visualiser les rendez-vous pris par les clients avec les ateliers.

2.3.2 Diagramme de cas d'utilisation

Avant de commencer tout développement, il est important de se poser les questions suivantes : À quoi servira-t-il ? Quels besoins devra-t-il satisfaire ? Le diagramme de cas d'utilisation en UML permet de répondre à ces interrogations. Il illustre un service complet, incluant un déclenchement, un déroulement et une conclusion pour l'acteur qui initie l'action. Ainsi, notre diagramme de cas d'utilisation apportera des réponses à ces questions. Présentons maintenant le diagramme de cas d'utilisation de notre plateforme.

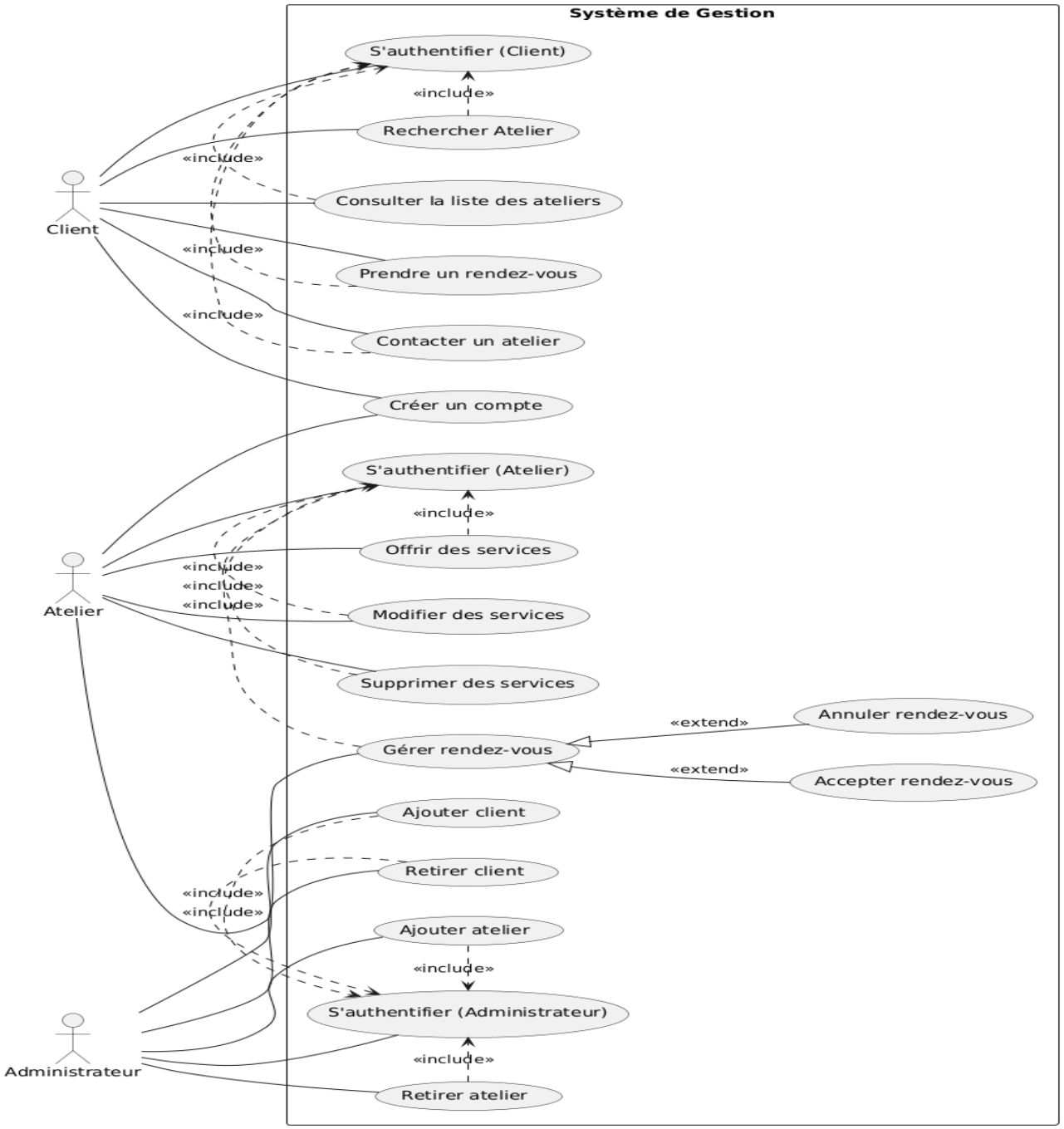


FIGURE 2.1 : Diagramme de cas d'utilisation

2.3.3 Diagramme des classes

Avant de commencer le développement de la structure interne du système, il est essentiel de comprendre comment les différentes entités interagissent entre elles. Le diagramme de classes en UML permet de modéliser la structure statique du système en représentant les classes, leurs attributs, leurs méthodes ainsi que les relations qui les lient. Il constitue une étape cruciale pour définir clairement les composants du système et leurs interactions, facilitant ainsi le développement et la maintenance du projet.
Nous allons maintenant présenter le diagramme de classes de notre plateforme.

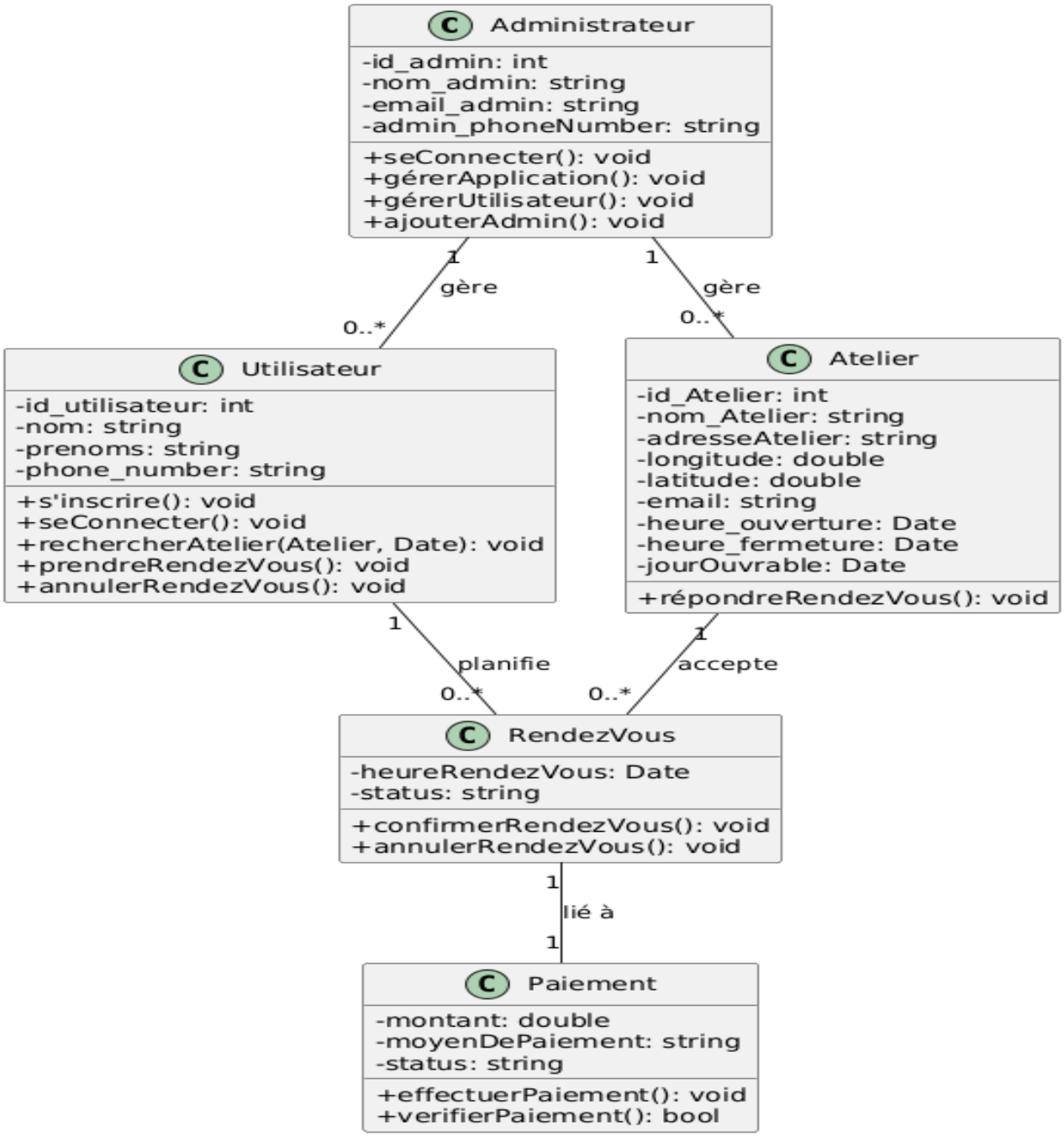


FIGURE 2.2 : Diagramme de cas classe

2.3.4 Diagramme de séquences

Les diagrammes de séquences illustrent les échanges entre les différents composants du système ainsi que leurs interactions avec les acteurs externes. Ils montrent comment les objets au sein du système échangent des messages pour accomplir certaines actions. Quant aux acteurs externes, ils interagissent avec le système à travers des interfaces utilisateur, appelées (IHM).

2.3.4.1 Cas d'Inscription

Le diagramme de séquence de l'inscription illustre le processus permettant à un utilisateur ou un atelier de créer un compte sur le système. L'utilisateur commence par saisir ses informations d'inscription, telles que son nom, son email, et un mot de passe, via l'interface web. Ces informations sont ensuite transmises au serveur pour traitement. Le serveur effectue une vérification en consultant la base de données pour s'assurer que l'utilisateur n'existe pas déjà. Si l'utilisateur est déjà enregistré, un message d'erreur est renvoyé à l'interface pour informer l'utilisateur que le compte ne peut pas être créé. Dans le cas contraire, les informations fournies sont enregistrées dans la base de données, et un message de confirmation est retourné à l'utilisateur pour lui signaler que son inscription a été réussie. Ce diagramme met en évidence les différentes interactions entre les composants du système (utilisateur, interface web, serveur, base de données) tout au long de ce processus.

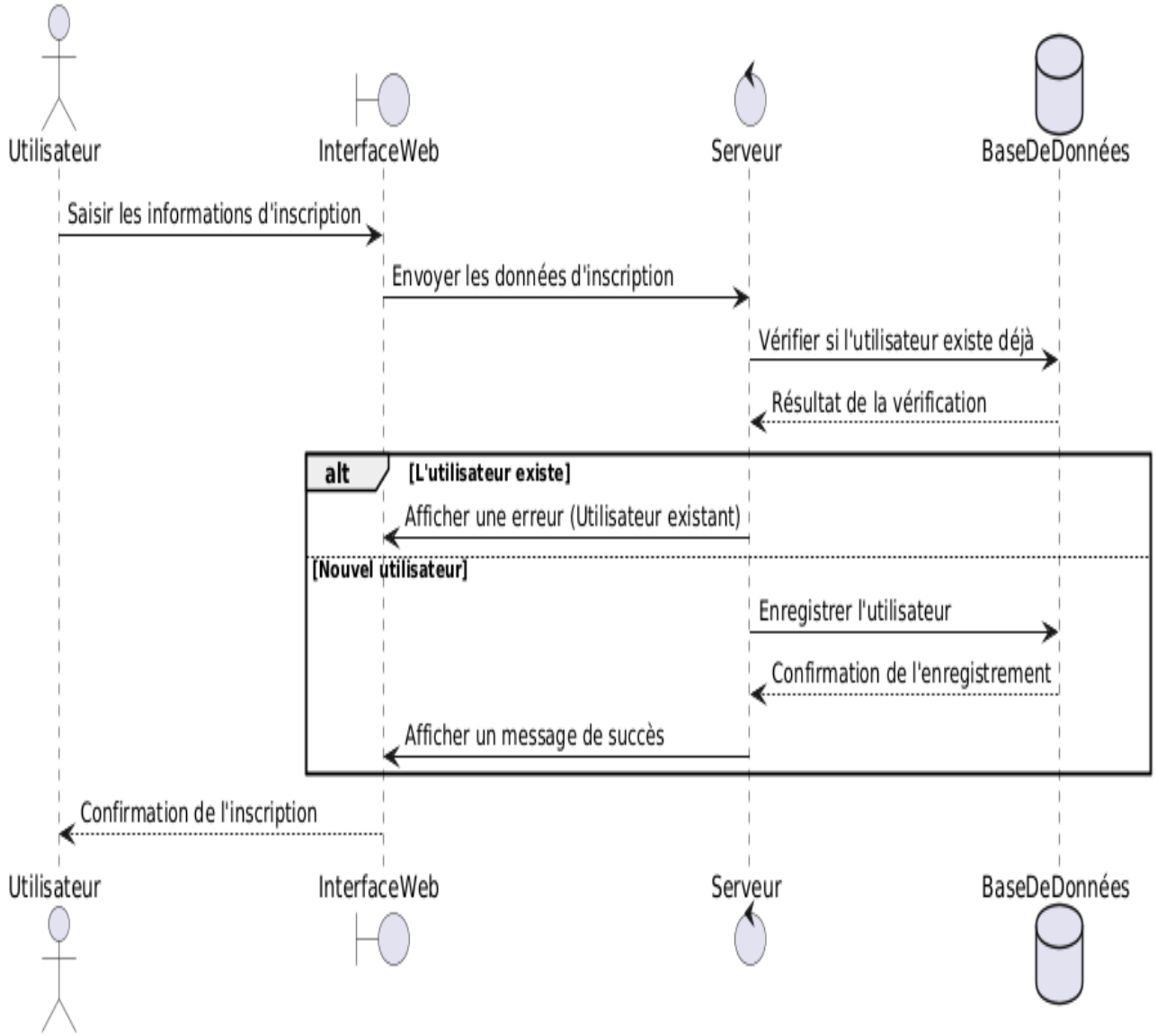


FIGURE 2.3 : Diagramme de séquence du cas «Inscription»

2.3.4.2 Cas de Connexion

Le diagramme de séquence de la connexion montre le processus par lequel un utilisateur ou un atelier peut accéder à son compte. L'utilisateur commence par fournir ses identifiants de connexion (email et mot de passe) via l'interface web. Ces données sont envoyées au serveur pour vérification.Le serveur interroge la base de données pour confirmer si les identifiants fournis correspondent à un utilisateur enregistré. Si les informations sont correctes, l'utilisateur est authentifié, et une redirection est effectuée vers son tableau de bord ou sa page d'accueil. En revanche, si les informations sont invalides, un message d'erreur est retourné à l'utilisateur pour l'informer de l'échec de la connexion. Ce diagramme met en lumière les étapes nécessaires pour garantir que seules les personnes disposant d'informations valides peuvent accéder au système, renforçant ainsi la sécurité.

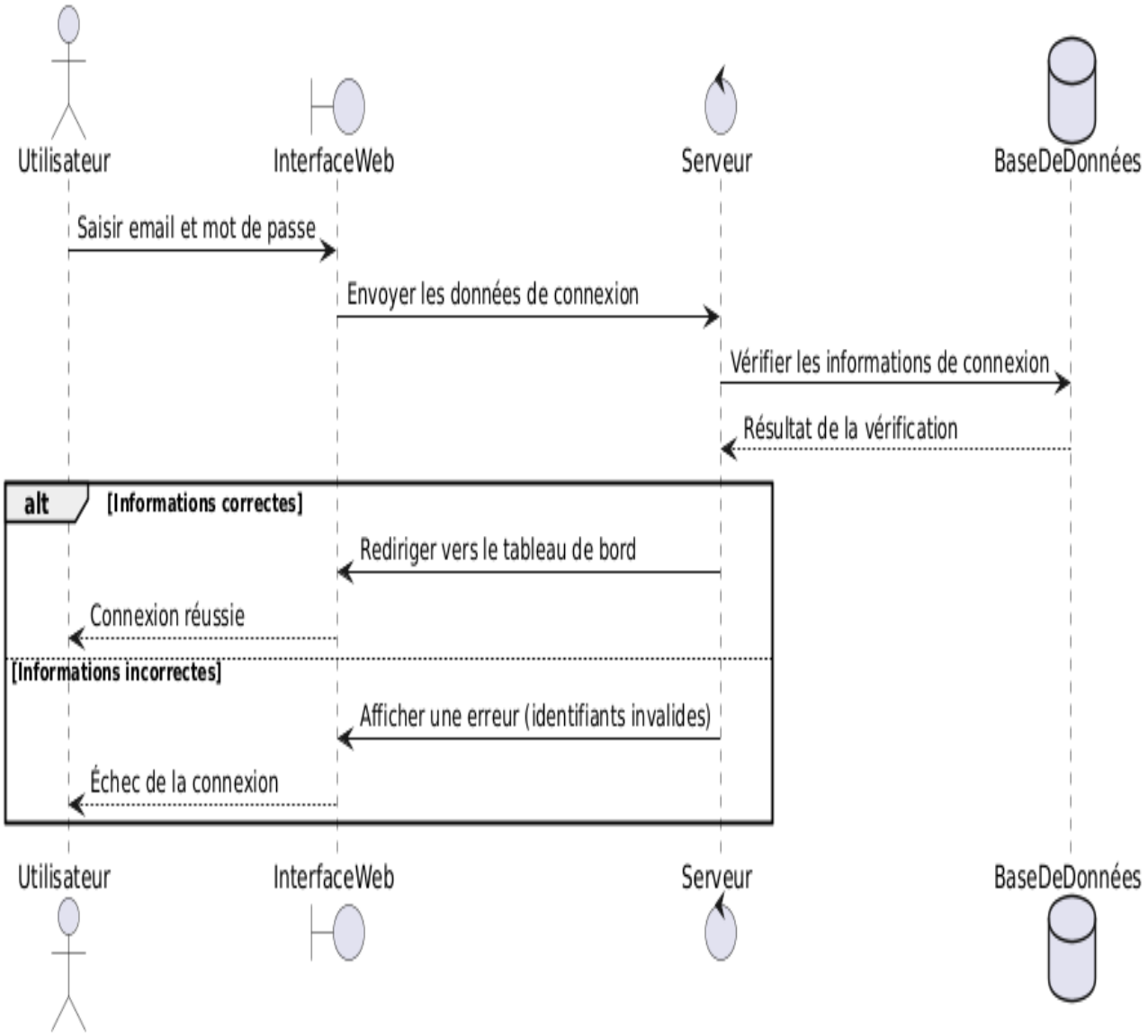


FIGURE 2.4 : Diagramme de séquence du cas «Connexion»

2.4 Choix Techniques

2.4.1 Choix de l'outil PlantText pour la modélisation UML

2.4.1.1 Présentation de l'outil PlantText

[5]PlantText est un outil en ligne basé sur le langage [6]PlantUML. Il permet de créer différents types de diagrammes UML (diagrammes de séquence, de cas d'utilisation, etc.) à l'aide d'un langage textuel simple et structuré. Ce choix technique s'est imposé pour plusieurs raisons :

- **Simplicité** : La création de diagrammes avec une syntaxe textuelle est intuitive et rapide, particulièrement utile pour des projets évolutifs.
- **Accessibilité** : PlantText est utilisable directement dans un navigateur, ce qui évite les installations complexes et facilite le travail collaboratif.

- **Modifications faciles** : Les diagrammes peuvent être modifiés rapidement en ajustant quelques lignes de code.
- **Exportation pratique** : Les diagrammes générés peuvent être exportés dans des formats adaptés (PNG, SVG) pour leur intégration dans des rapports ou documents.

2.4.1.2 Exemple d'application dans le projet

Dans le cadre de ce projet, l'outil a été utilisé pour modéliser plusieurs processus clés. Par exemple, le cas d'inscription d'un utilisateur sur la plateforme a été documenté grâce à PlantText. Le diagramme correspondant, déjà inclus dans ce document, illustre comment un utilisateur interagit avec l'interface web, le serveur et la base de données pour s'inscrire.

Voici un extrait du code PlantUML utilisé pour générer ce diagramme :

Listing 2.1 : Extrait du code pour le diagramme de séquence d'inscription

```
@startuml
actor Utilisateur
participant "InterfaceWeb" as IW
participant "Serveur" as S
participant "BaseDeDonnées" as BD

Utilisateur -> IW : Saisir les informations d'inscription
IW -> S : Envoyer les données d'inscription
S -> BD : Vérifier si l'utilisateur existe déjà[U+FFFD]
BD --> S : Résultat de la vérification

alt [L'utilisateur existe]
    S --> IW : Afficher une erreur (Utilisateur existant)
    IW --> Utilisateur : Afficher une erreur
else [Nouvel utilisateur]
    S -> BD : Enregistrer l'utilisateur
    BD --> S : Confirmation de l'enregistrement
    S --> IW : Afficher un message de succ[U+FFFD]s
    IW --> Utilisateur : Confirmation de l'inscription
end
@enduml
```

PlantText s'est révélé particulièrement efficace pour modéliser et documenter les interactions système de manière professionnelle et conforme aux standards UML. En centralisant la création et les modifications sur un code textuel, l'outil a permis d'accélérer la production de diagrammes clairs et fonctionnels, tels que le diagramme d'inscription intégré dans ce document.

2.4.2 Choix du Système de Gestion de Base de Données (SGBD)

Un Système de Gestion de Base de Données (SGBD) est un ensemble de programmes qui permet à un utilisateur de créer, gérer et contrôler une base de données. Il existe plusieurs types de SGBD, chacun ayant ses propres caractéristiques et étant adapté à des situations spécifiques.

2.4.2.1 PostgreSQL

Le choix de **PostgreSQL** pour le projet de géolocalisation d'ateliers mécaniques s'explique par plusieurs raisons. Tout d'abord, PostgreSQL est un SGBD open source, ce qui signifie qu'il n'y a aucun coût de licence, idéal pour un projet à budget limité. Cette base de données est largement utilisée

pour les applications nécessitant une forte intégrité des données, notamment dans les environnements transactionnels. Grâce à son respect strict des normes SQL et sa prise en charge des transactions ACID (Atomicité, Cohérence, Isolation, Durabilité), PostgreSQL est parfaitement adapté à une application où la fiabilité des données est cruciale.
De plus, PostgreSQL dispose de puissantes fonctionnalités géospatiales, grâce à l'extension `PostGIS`, ce qui est particulièrement utile pour des projets de géolocalisation. L'extension `PostGIS` permet de manipuler et d'interroger des données géographiques (points, polygones, lignes) de manière efficace, ce qui est essentiel pour la gestion des emplacements des ateliers mécaniques. Cela facilite la gestion des requêtes de géolocalisation, permettant des recherches basées sur des critères géographiques tels que la distance entre deux points.

Enfin, PostgreSQL bénéficie d'une large communauté active et d'une documentation exhaustive. Cela garantit un support solide pour le développement et la maintenance du projet.

2.4.2.2 Autres SGBD

Bien que des alternatives comme **Oracle** et **SQL Server** puissent offrir des fonctionnalités avancées en termes de sécurité et de gestion des données, ces SGBD sont généralement plus coûteux et plus complexes à mettre en place, ce qui ne justifie pas leur utilisation dans un projet de géolocalisation d'ateliers mécaniques de taille modeste. **MySQL**, bien qu'il soit populaire et largement utilisé, n'offre pas la même robustesse que PostgreSQL pour les requêtes géospatiales et ne bénéficie pas d'une intégration aussi poussée avec des extensions comme `PostGIS`. De plus, PostgreSQL est souvent mieux optimisé pour les applications complexes et offre une plus grande flexibilité dans la gestion des données volumineuses.

En résumé, PostgreSQL a été choisi pour sa robustesse, ses fonctionnalités géospatiales, et son absence de coût de licence.

2.4.3 Choix des outils de développement

2.4.3.1 Frontend (Nuxt.js avec Vuetify)

Le choix de **Nuxt.js** pour le frontend est motivé par plusieurs raisons. Tout d'abord, Nuxt.js est basé sur **Vue.js**, un framework léger et performant qui facilite le développement d'applications web interactives et réactives. En utilisant Nuxt.js, le projet bénéficie d'un rendu côté serveur (SSR - Server-Side Rendering), ce qui améliore le SEO (référencement naturel) et la rapidité du chargement des pages. Nuxt.js permet également de générer des pages statiques, ce qui est particulièrement adapté pour des applications nécessitant des performances optimisées et une expérience utilisateur fluide.
Vuetify, intégré à Nuxt.js, est une bibliothèque de composants basée sur Material Design qui permet de créer des interfaces élégantes et modernes de manière rapide et efficace. Vuetify propose une grande variété de composants prêts à l'emploi, facilitant ainsi la création d'une interface utilisateur esthétique et fonctionnelle, tout en réduisant les délais de développement. Ce framework permet aussi de garantir que l'application sera responsive, s'adaptant ainsi parfaitement aux différents types d'appareils (ordinateurs, tablettes, smartphones).

2.4.3.2 Backend (Node.js)

Le choix de **Node.js** pour le backend repose sur plusieurs facteurs. Tout d'abord, Node.js est conçu pour gérer un grand nombre de connexions simultanées, ce qui est idéal pour des applications en temps réel comme la géolocalisation. Il permet une gestion efficace des requêtes utilisateurs et des connexions API. De plus, Node.js est basé sur JavaScript, ce qui permet de maintenir une cohérence dans l'écriture du code à la fois côté frontend (Nuxt.js) et backend (Node.js), facilitant la gestion du projet et la collaboration entre les développeurs.

En utilisant `npm` (Node Package Manager), Node.js offre une vaste bibliothèque de modules tiers, permettant d'ajouter facilement des fonctionnalités supplémentaires, comme l'intégration de services externes (cartes, géolocalisation, paiements en ligne, etc.).

2.4.3.3 SQL pour la communication avec la base de données

SQL est un choix logique pour interagir avec la base de données, car il est le standard de facto pour la gestion des bases de données relationnelles. SQL permet de manipuler les données de manière claire et efficace, que ce soit pour la création de tables, l'insertion de nouvelles données ou l'exécution de requêtes complexes. Il est largement utilisé, bien compris par les développeurs, et bénéficie d'une documentation solide. Grâce à son adoption universelle, SQL permet d'assurer une intégration simple et flexible avec PostgreSQL et d'autres SGBD si nécessaire.

2.5 Sécurité

La sécurité [4] de notre application est une priorité essentielle pour protéger les données des utilisateurs et garantir une utilisation fiable et sûre du service. Étant donné que notre application de géolocalisation d'ateliers mécaniques manipule des informations sensibles (données personnelles des utilisateurs, coordonnées des ateliers, etc.), nous avons mis en place plusieurs mesures pour garantir un environnement sécurisé.

Authentification sécurisée :

L'authentification permet de contrôler l'accès aux fonctionnalités de l'application. Seuls les utilisateurs enregistrés et authentifiés peuvent accéder à certaines fonctionnalités sensibles, comme la modification des informations des ateliers ou l'accès aux données des utilisateurs. Nous avons mis en place un mécanisme d'authentification basé sur des mots de passe forts pour protéger l'accès.

Protection des données sensibles :

Les données sensibles, comme les informations personnelles et les coordonnées des ateliers, sont stockées de manière sécurisée. Nous avons mis en place des techniques de chiffrement pour protéger ces données en transit et au repos, garantissant ainsi leur confidentialité.

Contrôle d'accès basé sur les rôles :

Un système de gestion des rôles permet de restreindre l'accès à certaines fonctionnalités en fonction des droits de l'utilisateur. Par exemple, un administrateur d'atelier a plus de privilèges qu'un utilisateur classique, comme la possibilité de modifier les informations de son atelier. Ce système permet de limiter l'accès aux fonctionnalités critiques et de protéger les informations sensibles.

Prévention des attaques XSS et CSRF :

Afin de prévenir les attaques de type Cross-Site Scripting XSS et Cross-Site Request Forgery CSRF, nous avons mis en place des techniques de filtrage des entrées et utilisé des jetons anti-CSRF pour vérifier la légitimité des requêtes et éviter les attaques malveillantes.

Mises à jour et correctifs de sécurité :

Nous utilisons régulièrement les mises à jour des bibliothèques et des frameworks utilisés dans l'application, comme Nuxt.js et Node.js [3] , pour appliquer les derniers correctifs de sécurité et réduire les risques de vulnérabilités connues. Cela inclut également le maintien à jour des dépendances de sécurité.

Protection contre les injections de code :

Pour prévenir les attaques par injection de code, comme les injections SQL ou JavaScript malveillantes, nous utilisons des requêtes préparées et encodons systématiquement les données entrantes. Cela permet de s'assurer que seules les données valides et filtrées sont traitées par l'application.

Conclusion

Ce travail de modélisation et de conception a permis de structurer notre application de géolocalisation d'ateliers mécaniques de manière efficace. Nous avons utilisé des diagrammes UML pour clarifier les interactions et les relations entre les différentes entités de l'application. Les choix technologiques, notamment l'utilisation de PostgreSQL pour la gestion des données et de Nuxt.js associé à Node.js pour le développement, ont été soigneusement évalués pour répondre aux besoins du projet. De plus, des mesures de sécurité robustes ont été mises en place afin de protéger les données sensibles des utilisateurs, assurant ainsi une expérience sécurisée et fiable. Ces éléments constituent les fondations essentielles pour le développement réussi de notre application.

Chapitre 3

Résultats et Discussion

Introduction

La présentation du projet de géolocalisation d'ateliers mécaniques met en lumière ses fonctionnalités principales et sa structure globale. Ce projet vise à fournir une solution intuitive pour permettre aux utilisateurs de localiser facilement les ateliers à proximité. L'organisation de l'application, les services proposés aux utilisateurs, ainsi que les éléments garantissant son bon fonctionnement y seront explorés.

3.1 Présentation de l'application

MecaZone est une application web de géolocalisation d'ateliers mécaniques, offrant aux utilisateurs un moyen simple et rapide de localiser des ateliers proches en fonction de divers critères. Grâce à des options de recherche par emplacement, services offerts et avis des utilisateurs, elle propose une interface intuitive pour consulter les profils des ateliers, prendre des rendez-vous en ligne et interagir avec une carte dynamique. MecaZone offre une solution efficace aux automobilistes en facilitant l'accès à des ateliers de confiance à proximité. Sa géolocalisation avancée permet d'afficher une liste d'ateliers qualifiés, évalués selon des critères détaillés. L'application simplifie ainsi la recherche et la réservation de services d'entretien et de réparation, garantissant aux utilisateurs des prestations de qualité en toute facilité. Grâce à MecaZone, trouver un atelier mécanique fiable et compétent devient rapide et accessible.

3.1.1 Page d'acceuil

La page d'accueil de MecaZone présente une interface conviviale où les utilisateurs peuvent découvrir les fonctionnalités de l'application. Avant d'accéder aux ateliers mécaniques et de profiter des services, il est nécessaire de créer un compte. Cette étape garantit une expérience personnalisée et sécurisée, facilitant la recherche et la prise de rendez-vous.

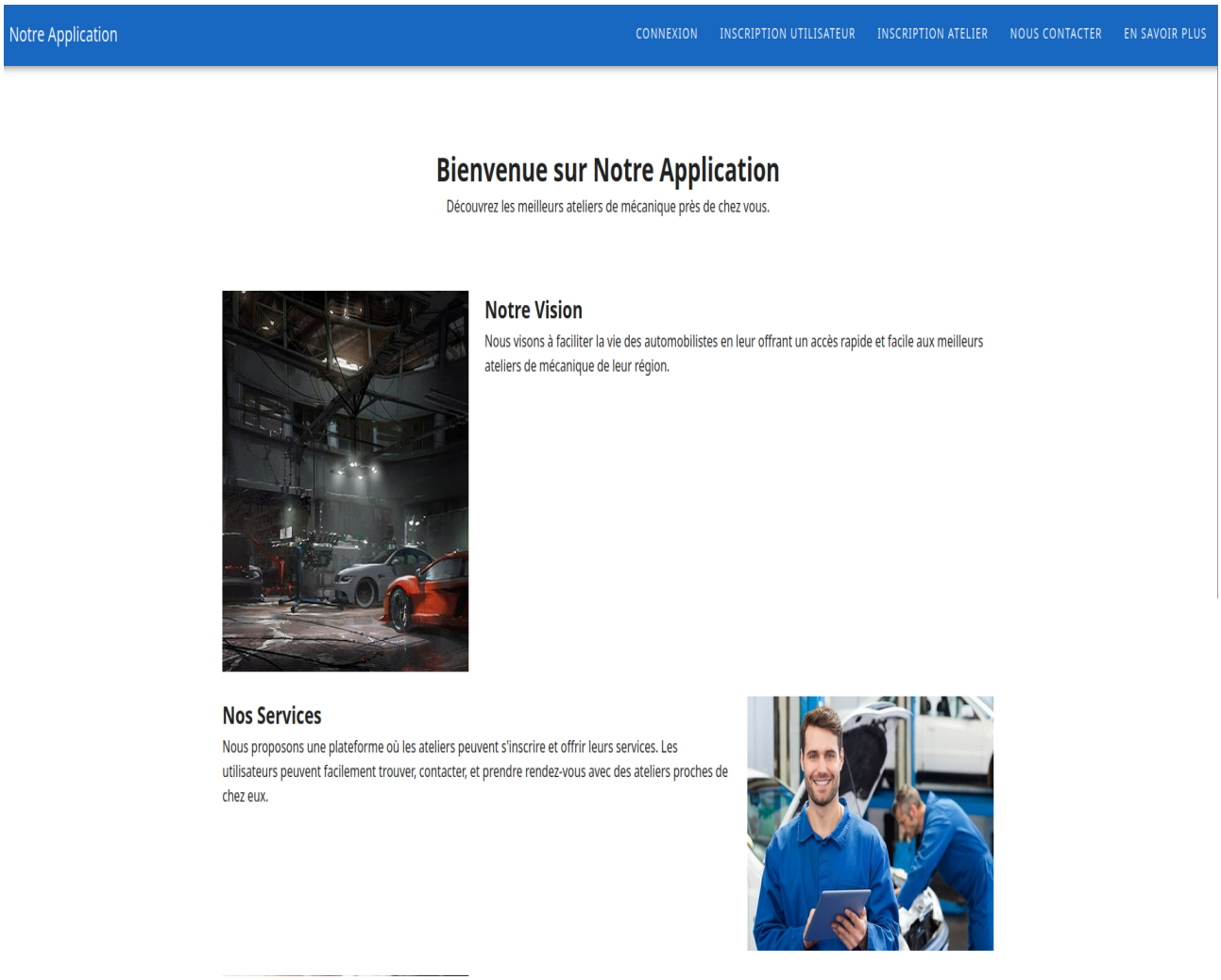


FIGURE 3.1 : Page d'acceuil

3.1.2 Pages d'inscription

MecaZone propose deux pages d'inscription distinctes pour répondre aux besoins variés de ses utilisateurs.
La première page d'inscription est destinée aux utilisateurs souhaitant accéder aux fonctionnalités de l'application. Elle leur permet de créer un compte personnel en fournissant des informations de base, telles que leur nom, leur adresse e-mail et un mot de passe. Une fois inscrits, les utilisateurs peuvent rechercher des ateliers mécaniques, consulter des évaluations et prendre des rendez-vous en ligne.
La seconde page d'inscription est conçue pour les ateliers mécaniques qui souhaitent rejoindre la plateforme. Ce formulaire d'inscription leur demande des informations spécifiques, telles que le nom de l'atelier, son adresse, les services offerts et les coordonnées du responsable. Une fois validée, cette inscription permet aux ateliers de gérer leur profil, d'interagir avec les utilisateurs et de recevoir des demandes de rendez-vous.



FIGURE 3.2 : Inscription utilisateur

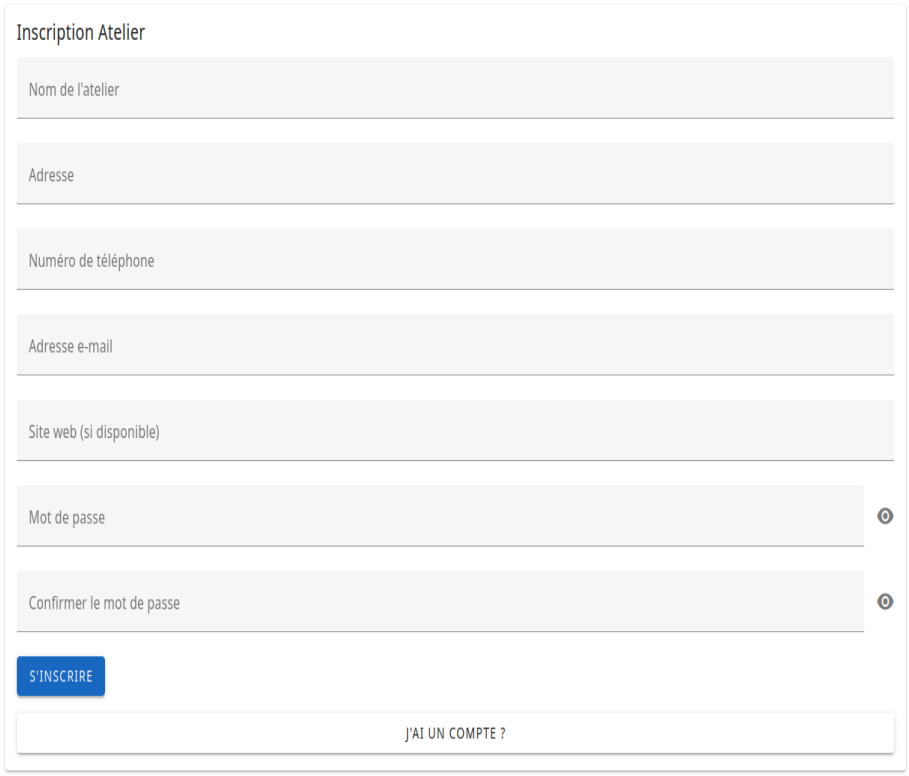


FIGURE 3.3 : Inscription Atelier

3.1.3 Page de connexion

La page de connexion de MecaZone est conçue pour offrir un accès sécurisé et rapide aux différents types d'utilisateurs de l'application. Elle regroupe trois sections distinctes : utilisateur, atelier, et administrateur.

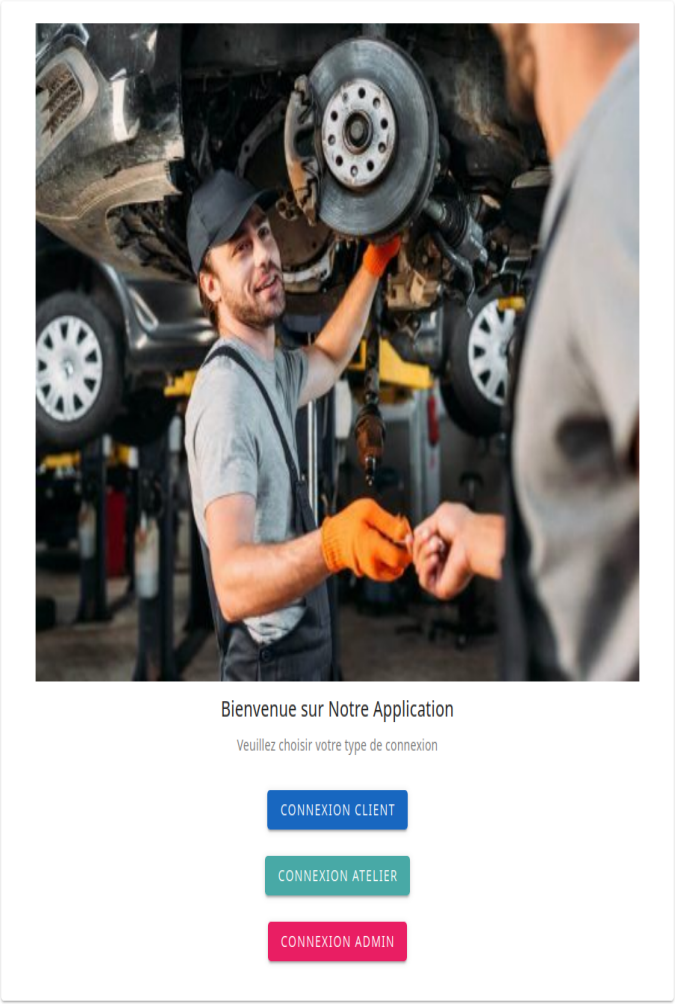


FIGURE 3.4 : Page de connexion

3.1.4 Interface profil Ateliers

L'interface de profil d'un atelier dans MecaZone est conçue pour permettre aux propriétaires d'ateliers de gérer facilement leurs activités et d'interagir avec leurs clients. Cette interface intuitive offre plusieurs fonctionnalités essentielles pour optimiser la gestion de l'atelier.

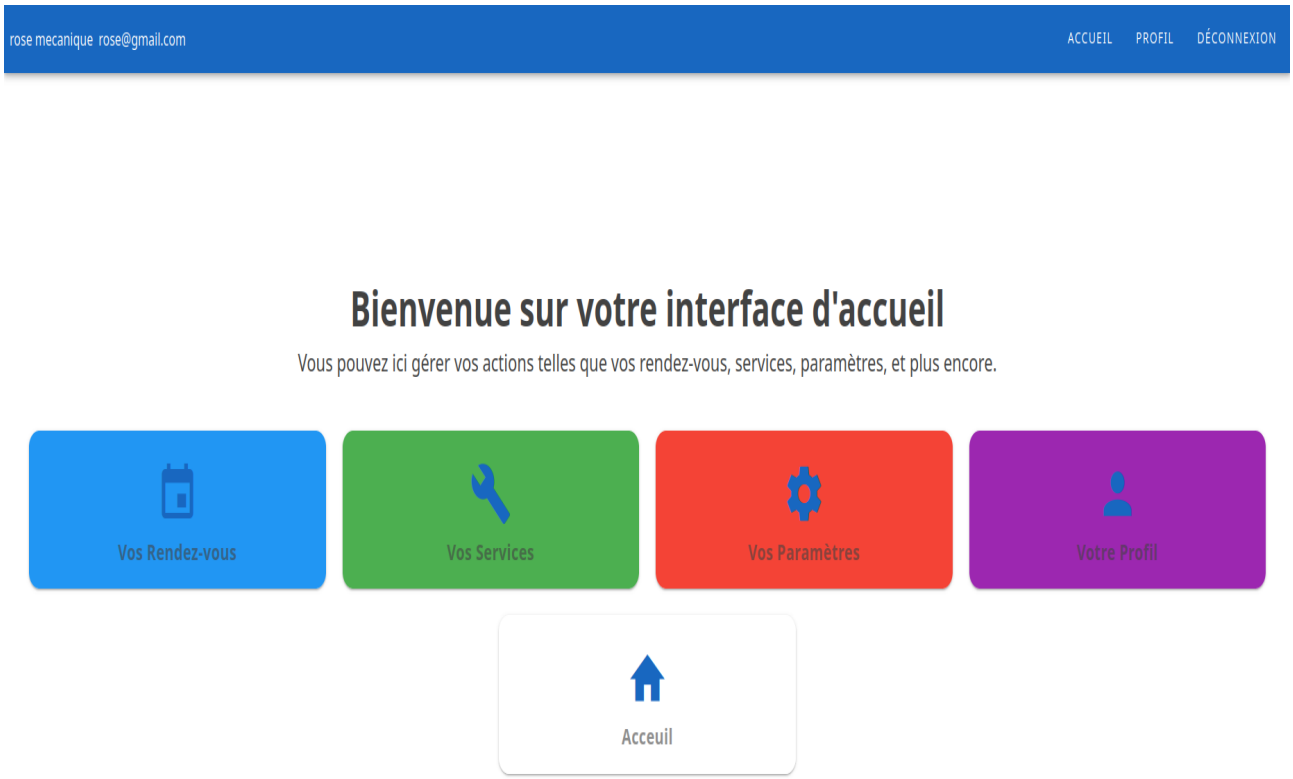


FIGURE 3.5 : Interface profil Atelier

3.1.5 Page recherche Ateliers

L'interface de recherche d'ateliers de MecaZone permet aux utilisateurs de localiser facilement des services mécaniques à proximité. Grâce à une barre de recherche intuitive, les utilisateurs peuvent rechercher des ateliers en fonction de leur position actuelle ou des services spécifiques dont ils ont besoin. Cette fonctionnalité est conçue pour simplifier le processus de recherche, offrant ainsi une expérience utilisateur optimale pour trouver rapidement des ateliers fiables et compétents.

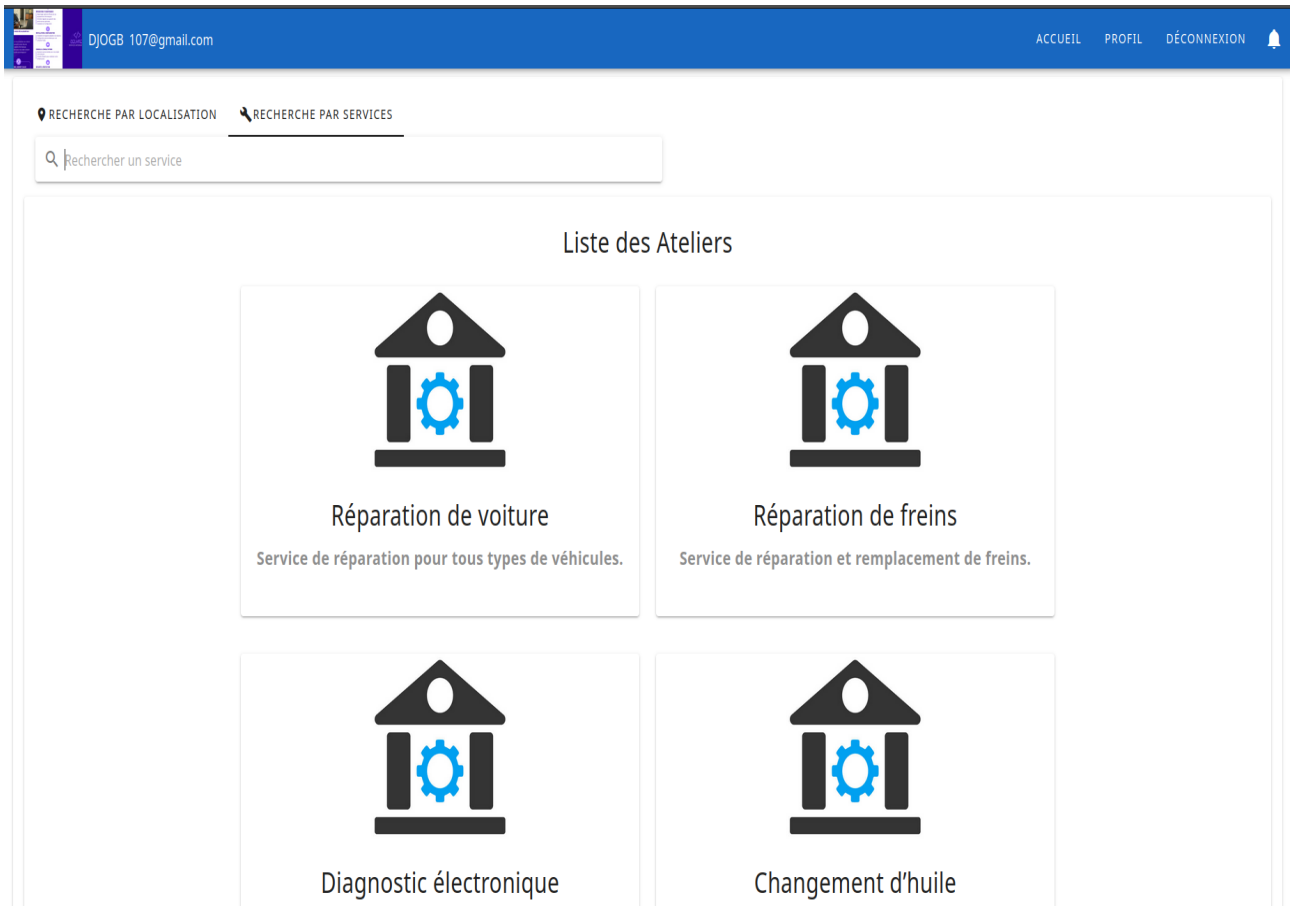


FIGURE 3.6 : Interface de recherche

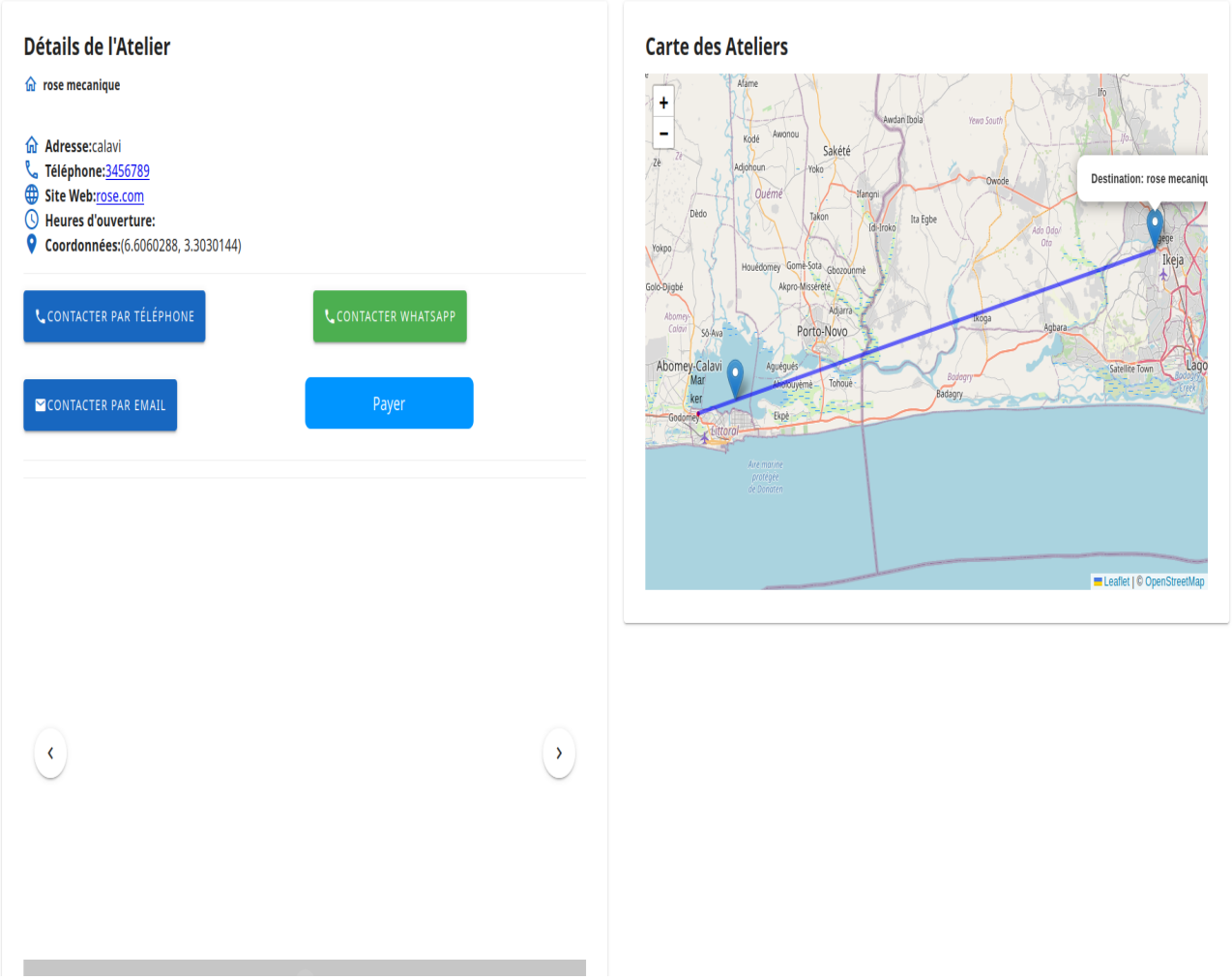


FIGURE 3.7 : Interface de résultat de recherche d'Atelier

3.1.6 Page offre de service(Atelier)

L'interface de gestion des services permet aux ateliers de gérer leur offre de services. Grâce à cette interface, les ateliers peuvent facilement ajouter de nouveaux services ou retirer ceux qui ne sont plus disponibles, offrant ainsi une flexibilité pour maintenir à jour leurs prestations pour les utilisateurs de MecaZone.

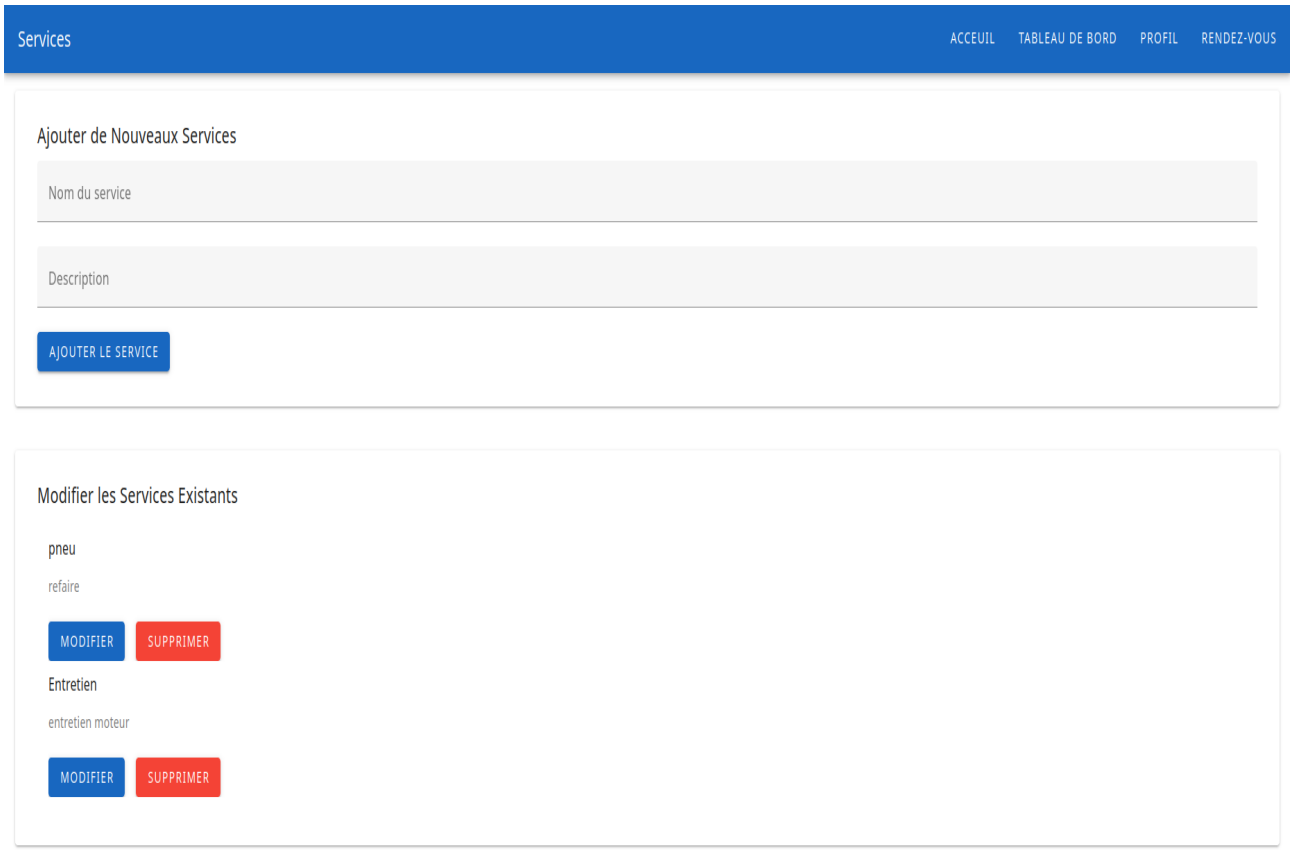


FIGURE 3.8 : Interface d'offre de service

3.1.7 Page d'administration

L'interface d'administration permet à l'administrateur de gérer les ateliers présents sur la plateforme. Grâce à cette interface, l'administrateur peut ajouter de nouveaux ateliers, modifier leurs informations, ainsi que bloquer ou retirer des ateliers ne respectant pas les conditions d'utilisation. Cette gestion centralisée assure la qualité et la fiabilité des ateliers disponibles pour les utilisateurs de MecaZone.

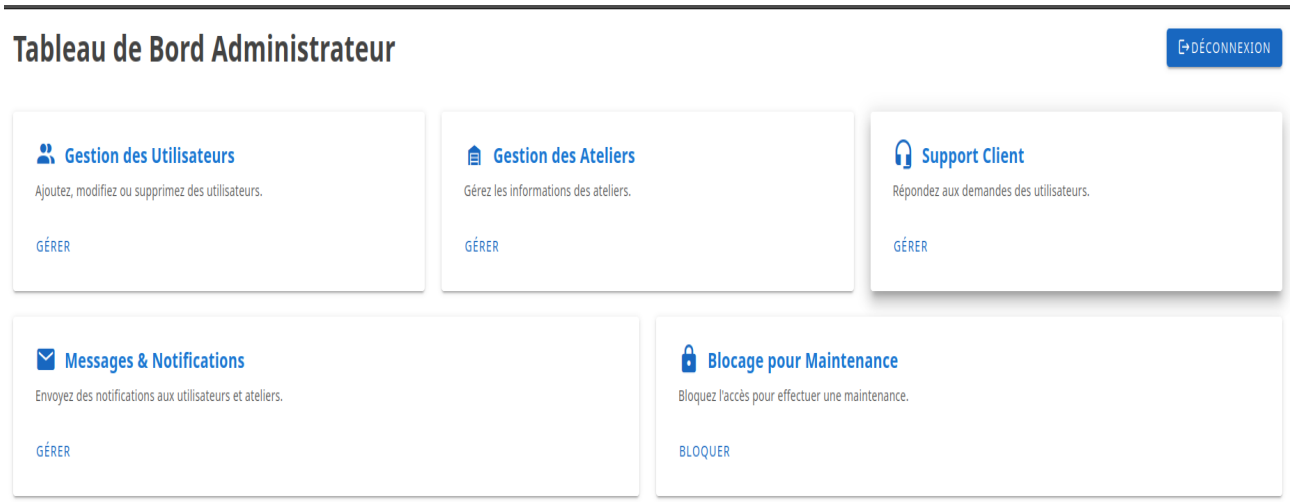


FIGURE 3.9 : Interface administration(gestion des Ateliers)

3.2 Discussion

Dans cette discussion, nous examinons les résultats obtenus avec MecaZone, une application qui aide les utilisateurs à trouver des ateliers mécaniques près de chez eux. MecaZone est efficace grâce à son interface facile à utiliser, qui permet aux utilisateurs de rechercher des ateliers selon leur emplacement ou les services qu'ils offrent

Cependant, nous avons rencontré quelques défis, comme la précision de la géolocalisation et la sécurité des données. Comparée à d'autres applications, MecaZone se distingue par son approche centrée sur les utilisateurs, permettant aux ateliers de gérer leurs services facilement.

Pour améliorer l'application à l'avenir, nous pensons qu'il serait utile d'intégrer des technologies plus avancées pour la géolocalisation, de renforcer la sécurité des données, d'ajouter un système de notation pour les utilisateurs, et d'explorer des partenariats avec des ateliers. En résumé, cette discussion souligne les réussites et les défis de MecaZone, tout en proposant des idées pour son développement futur.

Conclusion

Ce chapitre a permis de présenter les différentes interfaces de l'application MecaZone et de décrire les fonctionnalités principales de chaque page, depuis l'inscription et la connexion des utilisateurs jusqu'à la gestion des ateliers par les administrateurs. Ces interfaces offrent une navigation intuitive et structurée qui facilite l'expérience utilisateur, tout en assurant une gestion efficace des services proposés. MecaZone se distingue ainsi comme une plateforme pratique et accessible pour les utilisateurs à la recherche d'ateliers de confiance et pour les ateliers souhaitant optimiser leur visibilité et leur gestion.

Conclusion Générale

En conclusion, ce document a retracé le processus de conception, de développement et de mise en œuvre de l'application MecaZone. Ce projet de géolocalisation d'ateliers mécaniques vise à offrir une solution pratique et intuitive pour faciliter la recherche d'ateliers fiables. À travers une interface conviviale et des fonctionnalités telles que la recherche par emplacement, la gestion des services par les ateliers et le contrôle administratif, MecaZone propose une expérience utilisateur enrichie, répondant aux besoins tant des automobilistes que des professionnels de la mécanique.
Ce projet représente un pas vers la digitalisation du secteur, mettant en avant l'efficacité et la transparence, tout en garantissant la sécurité et la protection des données des utilisateurs. L'approche méthodique appliquée dans la conception, le choix des technologies, et la sécurisation des données a permis de répondre aux défis du projet, et place MecaZone comme un outil robuste et accessible pour le grand public.

Perspectives

Pour améliorer et enrichir les fonctionnalités de MecaZone, plusieurs pistes de développement futures sont envisagées :

- **Ajout de fonctionnalités d'évaluation des ateliers** : permettre aux utilisateurs de noter et de commenter les services des ateliers pour offrir plus de transparence et aider d'autres utilisateurs dans leur choix.
- **Mise en place de notifications et rappels** : proposer aux utilisateurs des alertes pour leurs rendez-vous ou des promotions des ateliers partenaires, facilitant ainsi la gestion de leurs services.
- **Développement de partenariats avec des acteurs du secteur automobile** : intégrer des collaborations avec des fournisseurs de pièces, des associations professionnelles, ou d'autres services pour étendre l'offre de MecaZone.
- **Optimisation de la recherche géolocalisée** : améliorer la précision de la géolocalisation pour offrir des suggestions encore plus adaptées en fonction de l'emplacement actuel et des préférences de l'utilisateur.

Ces perspectives visent à renforcer l'utilité et l'accessibilité de MecaZone, transformant l'application en un outil essentiel pour les automobilistes en quête de services de qualité. [**?**]

Bibliographie

[1] A. Alavi and M. Jahanbakhsh. Geolocation and its applications in automotive services. *Journal of Automotive Engineering*, 233(3) :450–460, 2019.

[2] Google. Google maps api, 2024.

[3] W. McDaniel. *Node.js Security : Secure Your Application with Confidence*. Apress, 2014.

[4] OWASP Foundation. Owasp top ten : The ten most critical web application security risks, 2021.

[5] PlantText. Planttext : A free online sequence diagram editor for plantuml, 2024.

[6] PlantUML. Plantuml official documentation, 2024.

[7] C. Team. Caradvise : Find auto repair shops and get service estimates, 2024.

[8] F. Team. Fixd : Car health monitor and diagnostic tool, 2024.

Table des matières

Dédicace ii

Remerciements iii

Résumé iv

. iv

Abstract v

. v

List of Figures vi

Liste des acronymes vii

Introduction 1

1 Revue de littérature 3

Introduction 3

1.1 Technologies de géolocalisation 3

1.2 Services et Applications de Géolocalisation 4

1.3 Limites et Défis de la Géolocalisation 4

1.4 Présentation des solutions existantes 5

1.5 Intérêt de la solution par rapport aux existantes 5

Conclusion 6

2 Modélisation et conception 7

Introduction 7

2.1 Comparaison entre UML et MERISE 7

2.2 Choix de la méthode de modélisation 7

2.3 Modélisation UML 8

2.3.1 Identification des acteurs du système 8

2.3.2 Diagramme de cas d'utilisation 9

2.3.3 Diagramme des classes 10

2.3.4 Diagramme de séquences 11

2.3.4.1 Cas d'Inscription 11

2.3.4.2 Cas de Connexion 12

2.4 Choix Techniques 13

2.4.1 Choix de l'outil PlantText pour la modélisation UML 13

2.4.1.1 Présentation de l'outil PlantText 13

2.4.1.2 Exemple d'application dans le projet . . . 14
2.4.2 Choix du Système de Gestion de Base de Données (SGBD) . . . 14
2.4.2.1 PostgreSQL . . . 14
2.4.2.2 Autres SGBD . . . 15
2.4.3 Choix des outils de développement . . . 15
2.4.3.1 Frontend (Nuxt.js avec Vuetify) . . . 15
2.4.3.2 Backend (Node.js) . . . 16
2.4.3.3 SQL pour la communication avec la base de données . . . 16
2.5 Sécurité . . . 16
Conclusion . . . 17

3 Résultats et Discussion **18**
Introduction . . . 18
3.1 Présentation de l'application . . . 18
3.1.1 Page d'acceuil . . . 18
3.1.2 Pages d'inscription . . . 19
3.1.3 Page de connexion . . . 21
3.1.4 Interface profil Ateliers . . . 22
3.1.5 Page recherche Ateliers . . . 23
3.1.6 Page offre de service(Atelier) . . . 25
3.1.7 Page d'administration . . . 26
3.2 Discussion . . . 27
Conclusion . . . 27

Conclusion **29**

Bibliographie **30**

Bibliographie **30**

Table des matières **31**