



جامعة عبد المالك السعدي  
 UNIVERSITÉ ABDELMALEK ESSAADI

LICENCE FONDAMENTALE SCIENCES  
MATHÉMATIQUES ET INFORMATIQUE  
PROJET DE FIN D'ÉTUDES

---

Conception et réalisation d'un site web  
pour la réservation des taxis

---

*Jury:*

Président : BEN MAÂTI MOHAMED  
LARBI

Examinateur : RAFIK TAOUIL

Encadrant : ATTARIUAS HICHAM

*Étudiants:*

EL QADDOUH MARYAM

GAZI SALMA

RGUILA SOUIANE

# Dédicace

Nous souhaitons dédier ce travail à nos chers parents, dont les sacrifices quotidiennes ont contribué à faire de nous ce que nous sommes aujourd’hui. Nous voulons qu’ils sachent que nous les associons pleinement à la réalisation de ce projet.

Nous exprimons également notre gratitude à ceux qui n’ont jamais cessé de nous encourager et de nous conseiller. Leur soutien constant a été précieux. Nous tenons à remercier chaleureusement nos éducateurs bienveillants pour tous les efforts qu’ils ont déployés afin de nous aider. Leur engagement et leurs conseils ont grandement contribué à notre réussite.

Nous adressons une mention spéciale à notre cher encadrant pédagogique, Monsieur **AT-TARIUAS Hicham**. Sa guidance et son expertise ont été d’une importance capitale.

Enfin, nous voulons exprimer notre gratitude envers tous nos amis et nos proches qui ont été présents à nos côtés. Leur soutien inconditionnel a été une source d’encouragement et de motivation tout au long de ce parcours.

# Remerciements

Nous remercions d'abord et avant tout Allah qui nous a donné le courage, la santé, la possibilité et la patience pour réaliser ce travail.

Un remerciement particulier à notre encadreur Monsieur **ATTARIUAS Hicham** pour son soutien, son sérieux, sa disponibilité, ses précieux conseils et son aide tout au long de l'élaboration de ce travail.

Nous remercions également les membres du jury d'avoir accepté d'examiner et d'évaluer notre travail. Sans oublier tous les enseignants du département d'informatique pour la qualité de l'enseignement qu'ils ont bien voulu nous prodiguer durant nos études afin de nous fournir une formation efficiente.

Enfin, nos remerciements s'adressent à tous ceux qui ont participé de près ou de loin à la réalisation de ce travail.

*Mariam, Salma et Soufiane*

# Résumé

Le site web de réservation de taxis est un projet visant à faciliter la réservation de taxis pour les membres de votre association. Le site permettra aux utilisateurs de réserver des taxis en ligne avec des informations telles que l'adresse de départ et d'arrivée, le nombre de passagers et type de paiement. L'objectif principal est de simplifier le processus de réservation et de fournir un moyen pratique pour les membres de votre association de réserver des taxis.

Il permettra aux utilisateurs de visualiser les disponibilités des taxis en temps réel et de réserver en fonction de leurs besoins.

Le site affichera également une carte graphique . Il fournira également des informations telles que le nombre total de réservations etc. Cela permettra aux utilisateurs de suivre l'évolution de leur réservation et de planifier en conséquence.

L'objectif principal de ce site web est de faciliter la réservation de taxis pour les membres de votre association et de rendre le processus plus efficace. Il contribuera à améliorer l'expérience des utilisateurs et à éviter les complications liées aux réservations de taxis.

**Mots-clés:** Site web, HTML, PHP, Komodo IDE, XAMPP, Visual Paradigm, MySQL, SGBD, réservation de taxis, itinéraires, disponibilité des taxis, carte géographique.

# Abstract

The taxi booking website is a project aimed at facilitating taxi reservations for members of your association. The website will allow users to book taxis online with information such as the departure and arrival addresses, the number of passengers, and the payment method. The main objective is to simplify the reservation process and provide a convenient way for association members to book taxis.

It will enable users to view real-time taxi availability and make bookings based on their needs.

The website will also display a graphical map. It will provide information such as the total number of bookings, etc. This will allow users to track the progress of their booking and plan accordingly.

The primary goal of this website is to streamline taxi reservations for members of your association and make the process more efficient. It will contribute to improving the user experience and avoiding complications associated with taxi bookings.

**Key words:** Website, HTML, PHP, Komodo IDE, XAMPP, Visual Paradigm, MySQL, SGBD, taxi reservation, routes, taxi availability, geographical map.

## Liste des Abréviations

Abréviation	Signification
UML	Unified Modeling Language
MLD	Modèle Logique des Données
SGBD	Système de Gestion de Base de données
MCD	Modèle Conceptuel des Données
BD	Base de Données
PHP	Hypertext Preprocessor
MySQL	My Structure Query Language

# Table des matières

<b>1 Présentation et contexte général du projet</b>	<b>14</b>
1.1 Introduction : . . . . .	15
1.2 Conduite du projet : . . . . .	16
1.2.1 Cycle de vie : . . . . .	16
1.2.2 Planification d'un projet : . . . . .	17
<b>2 Analyse fonctionnelle</b>	<b>19</b>
2.1 Étude de l'existant : . . . . .	20
2.1.1 Problématique : . . . . .	20
2.1.2 Solution proposée : . . . . .	20
2.2 Expression des besoins : . . . . .	20
2.2.1 Besoins fonctionnels : . . . . .	20
2.2.2 Besoins non fonctionnels : . . . . .	21
<b>3 Conception</b>	<b>22</b>
3.1 Méthodologie de conception : . . . . .	23

3.2 Langage de modélisation UML : . . . . .	23
3.2.1 Définitions : . . . . .	23
3.2.2 Les objectifs de l'UML : . . . . .	23
3.2.3 Les différents types de diagrammes de UML : . . . . .	24
3.3 Diagramme des cas d'utilisation : . . . . .	24
3.4 Diagramme de classes : . . . . .	25
3.5 Diagramme de séquence : . . . . .	26
3.5.1 Diagramme de séquence d'inscription : . . . . .	27
3.5.2 Diagramme de séquence de rechercher taxi : . . . . .	27
3.5.3 Diagramme de séquence de réservation de taxi : . . . . .	28
3.6 Conception de base de données : . . . . .	29
3.6.1 Modèle rationnel de données : . . . . .	29
3.7 Base de données : . . . . .	31
3.7.1 Développement de la base de données : . . . . .	31
3.7.2 Les tables de la base de données : . . . . .	34
<b>4 Réalisation</b>	<b>36</b>
4.1 Environnement logiciels : . . . . .	37
4.1.1 Serveur de la base de données MySQL : . . . . .	37
4.1.2 PHP MyAdmin : . . . . .	37
4.1.3 XAMPP : . . . . .	38

4.1.4	Team Gantt : . . . . .	38
4.1.5	Visual paradigm : . . . . .	39
4.1.6	Komodo IDE 12 : . . . . .	39
4.2	Langages de programmation : . . . . .	40
4.2.1	Langage PHP : . . . . .	40
4.2.2	Laravel Framework de PHP : . . . . .	41
4.2.3	Langage HTML : . . . . .	41
4.2.4	Langage CSS : . . . . .	42
4.2.5	Langage Javascript : . . . . .	42
4.2.6	Langage Bootstrap : . . . . .	43
4.3	Les interfaces : . . . . .	43
4.3.1	Interface d'authentification : . . . . .	43
4.3.2	Page de sélection . . . . .	44
4.3.3	Page d'inscription d'un client : . . . . .	44
4.3.4	Page d'accueil : . . . . .	45
4.3.5	Footer du site . . . . .	46
4.3.6	Page de profil : . . . . .	47
4.4	La page principale : . . . . .	47
4.4.1	Page principale de passager : . . . . .	47
4.4.2	Page principale de chauffeur : . . . . .	48

4.4.3	La page de commande de chauffeur . . . . .	49
4.4.4	Page d'accueil d'admin : . . . . .	49
4.4.5	Page de demandes de passager : . . . . .	50

# Table des figures

1.1	Cycle de vie en cascade . . . . .	17
1.2	Planification par GANTT . . . . .	17
3.1	diagramme des cas d'utilisation . . . . .	25
3.2	diagramme de classes . . . . .	26
3.3	diagramme de séquence d'inscription . . . . .	27
3.4	diagramme de séquence de recherche taxi . . . . .	28
3.5	diagramme de séquence de réservation de taxi . . . . .	29
3.6	Connexion au serveur . . . . .	32
3.7	Création de la table users . . . . .	33
3.8	Création de la table chauffeurs . . . . .	33
3.9	Création de la table réservations . . . . .	33
3.10	Table de données de réservation . . . . .	34
3.11	Table de données de passagers . . . . .	34
3.12	Table de données de chauffeurs . . . . .	34
4.1	Logo MySQL . . . . .	37

4.2 Logo de phpMyAdmin . . . . .	38
4.3 logo de XAMPP . . . . .	38
4.4 Logo de teamgantt . . . . .	39
4.5 Logo de Visual Paradigm . . . . .	39
4.6 Logo de Komodo IDE 12 . . . . .	40
4.7 Logo de php . . . . .	41
4.8 Logo de Laravel . . . . .	41
4.9 Logo de HTML . . . . .	42
4.10 Logo de CSS . . . . .	42
4.11 Logo de Javascript . . . . .	43
4.12 Logo de boorstrap . . . . .	43
4.13 Interface d'authentification . . . . .	44
4.14 Interface de sélection . . . . .	44
4.15 Interface d'inscription de passager . . . . .	45
4.16 Interface d'inscription de chauffeur . . . . .	45
4.17 Interface d'accueil . . . . .	46
4.18 Footer de site . . . . .	46
4.19 Page de profil . . . . .	47
4.20 Page principale de passager . . . . .	48
4.21 Page principale de chauffeur . . . . .	48

4.22	Page de commande de chauffeur	49
4.23	Page d'accueil d'admin	49
4.24	Page de demande de passager	50

# Chapitre 1

## Présentation et contexte général du projet

Dans le cadre de cette chapitre, nous allons commencer par présenter le projet que nous avons réalisé. Ensuite, nous allons abordé la spécification des besoins, qui consiste à qualifier les besoins fonctionnels attendus par le site afin de mieux comprendre le projet.

## 1.1 Introduction :

Au cours de la dernière décennie, les applications de réservation de taxi ont révolutionné l'industrie du transport en offrant une solution pratique, efficace et personnalisée pour répondre aux besoins de déplacement des individus. Ces applications, telles que Uber, Lyft, Didi, Grab et bien d'autres, ont transformé la manière dont les gens se déplacent en leur offrant une alternative conviviale aux méthodes de réservation traditionnelles.

Les chiffres parlent d'eux-mêmes. Selon les dernières statistiques, le marché mondial des applications de réservation de taxi a connu une croissance exponentielle. En 2021, il était estimé à plus de 108 milliards de dollars, avec une prévision de croissance annuelle de près de 15. [4]

L'un des facteurs clés de cette popularité croissante est la commodité offerte par ces applications. Grâce à une interface utilisateur conviviale et intuitive, les utilisateurs peuvent facilement réserver un taxi en quelques clics sur leur smartphone. Plus besoin de faire des appels téléphoniques fastidieux ou de chercher désespérément un taxi disponible dans la rue. Il suffit d'entrer sa destination, de choisir le type de véhicule souhaité et de confirmer la réservation. Les chauffeurs affiliés aux plateformes reçoivent ensuite la demande et se rendent rapidement sur le lieu de prise en charge.

Outre la commodité, les applications de réservation de taxi offrent également une transparence accrue. Les utilisateurs ont accès à des informations détaillées sur le chauffeur, telles que son nom, sa photo, les évaluations et les commentaires des autres utilisateurs. Cette transparence renforce la confiance des utilisateurs et leur permet de choisir un chauffeur en fonction de leurs préférences et de leurs besoins.

Enfin, ces applications ont également contribué à améliorer l'efficacité et la productivité dans le secteur des transports. En optimisant les itinéraires et en permettant une meilleure gestion des ressources, les plateformes de réservation de taxi contribuent à réduire les temps d'attente, à augmenter le taux d'utilisation des véhicules et à réduire les émissions de carbone.

## 1.2 Conduite du projet :

### 1.2.1 Cycle de vie :

#### Modèle en cascade : [6]

Le modèle en cascade, ou « waterfall » en anglais, est une organisation des activités d'un projet sous forme de phases linéaires et séquentielles, où chaque phase correspond à une spécialisation des tâches et dépend des résultats de la phase précédente. Il comprend les phases d'exigences, de conception, de mise en œuvre et de mise en service.

Le modèle en cascade est un cycle de vie de projet issu des industries manufacturières et du secteur de la construction, où une conception préalable est nécessaire, compte tenu des fortes contraintes matérielles et des coûts élevés afférents aux changements de la conception en cours de réalisation. Il est utilisé notamment dans les domaines de l'ingénierie et du développement de logiciels.

#### Le cycle en cascade est divisé en cinq phases :

1. **Définition des besoins** : elle consiste à définir la finalité du projet et son intégration dans une stratégie globale.
2. **Conception détaillée** : dans cette activité, il s'agit de la préparation de l'architecture générale de l'application et modélisation.
3. **Développement** : il s'agit d'une traduction des fonctionnalités définies dans la phase de conception en langage de programmation.
4. **Intégration** : elle consiste à vérifier que le site web correspond exactement au cahier des charges du projet en obtenant enfin un manuel d'utilisation bien détaillé aux utilisateurs.
5. **Mise en production** : mettre le site web en production.

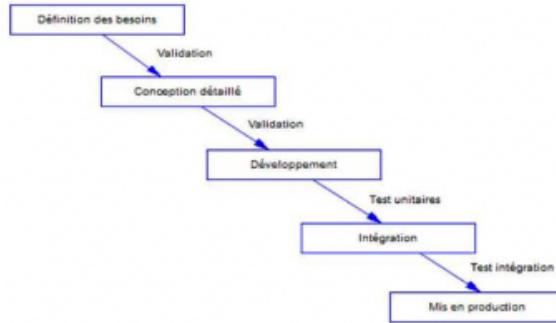


FIGURE 1.1 – Cycle de vie en cascade

### 1.2.2 Planification d'un projet :

La planification de projet correspond à l'organisation des tâches à réaliser sur une période donnée. L'objectif de la planification est de déterminer le coût, les ressources mobilisées et la meilleure manière d'ordonnancer toutes les tâches à effectuer, afin de conduire avec succès le projet et réussir son organisation.

Le diagramme de GANTT [3] aide à mieux présenter les différentes tâches durant les mois de travail pour terminer notre projet.

Le diagramme ci-dessous est réalisé à l'aide de Gantt Project :

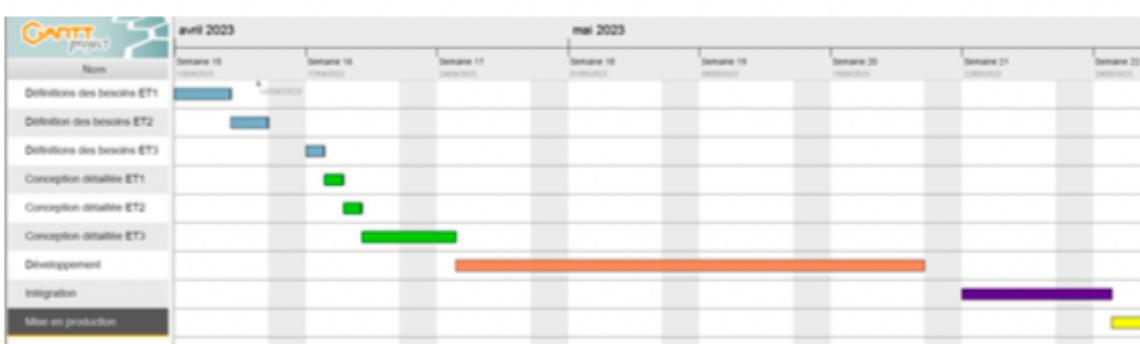


FIGURE 1.2 – Planification par GANTT

Dans ce chapitre, nous avons présenté le cahier des charges, la planification, le diagramme de Gantt et le cycle de vie que nous avons utilisé pour conduire notre projet.

Dans le chapitre suivant nous allons entamer l'analyse fonctionnelle.

# **Chapitre 2**

## **Analyse fonctionnelle**

Dans ce chapitre, nous nous intéresserons à expliquer la problématique et à proposer une solution possible, tout en déterminant les besoins des utilisateurs. Nous spécifierons les besoins fonctionnels et non fonctionnels nécessaires pour assurer la réalisation d'une notre site web.

## 2.1 Étude de l'existant :

### 2.1.1 Problématique :

De nombreuses personnes sont confrontées à la difficulté de trouver un taxi pour se rendre au travail ou aux études. Au lieu de pouvoir satisfaire rapidement leurs besoins de transport, elles se retrouvent à passer de longues heures à attendre qu'une voiture se présente. De plus, le problème de l'augmentation des tarifs ajoute à leurs préoccupations.

### 2.1.2 Solution proposée :

Après une étude approfondie, la solution proposée consiste à développer un site web qui améliorera considérablement le service de réservation de taxis, permettant ainsi de résoudre plusieurs problèmes liés à sa gestion. Cette solution permettra aux utilisateurs de gagner du temps en simplifiant le processus de réservation et en réduisant les délais d'attente.

## 2.2 Expression des besoins :

### 2.2.1 Besoins fonctionnels :

Les besoins fonctionnels représentent les principales fonctionnalités du système. Ces besoins proviennent généralement des utilisateurs du système. Ce site web devra permettre :

- **Économiser de l'argent** : proposer un prix à payer par les passagers et approuvé par les chauffeurs.
- **Économiser du temps** : en réservant un taxi, vous évitez de perdre du temps à attendre un taxi disponible dans la rue.
- **Sécurité** : en réservant un taxi, vous pouvez être sûr que le chauffeur est un professionnel agréé et expérimenté, qui connaît bien la ville et les règles de circulation.
- **Garantir la disponibilité** : en réservant un taxi à l'avance, vous pouvez être sûr qu'un chauffeur sera disponible pour venir à l'heure et à l'endroit convenus.
- **Éviter les retards** : en réservant un taxi, vous pouvez assurer d'arriver à votre desti-

nation à l'heure.

### 2.2.2 Besoins non fonctionnels :

En plus des besoins fonctionnels, il est également essentiel de répondre à des besoins non fonctionnels pour garantir un site web de réservation de taxis optimal. Ces besoins comprennent :

- **La convivialité** : le site web doit être facile à utiliser. Il doit présenter enchainement logique entre les interfaces, c'est-à-dire simples, ergonomiques et adaptées à l'utilisateur.
- **La performance** : Temps de réponse – le chargement du site web, ouverture d'onglet et des délais de rafraîchissement, etc.
- **La compatibilité** : La compatibilité avec des sites partagées. La compatibilité sur des systèmes d'exploitation différents. La compatibilité sur des plateformes différentes.
- **La stabilité** : le site doit être stable dans son environnement d'exécution et ne doit pas contenir des bogues bloquants.
- **Le site responsive** : Un site responsive est un site qui est conçu et développé de façon à pouvoir s'adapter à toutes les résolutions d'écran. C'est donc un seul et même site qui peut être consulté sur ordinateur, sur smartphone ou sur tablette.

Dans cette partie, nous avons présenté les points principaux de notre projet et la solution adoptée pour résoudre la problématique qui va répondre aux besoins fonctionnels et non fonctionnelle du site web.

Dans la section suivante, nous allons présenter la conception qui a été mise en œuvre tout au long de la réalisation de ce projet.

# Chapitre 3

## Conception

Dans cette partie, nous allons établir la conception générale du système de réservation de taxi en utilisant des diagrammes UML [2] et des modèles de base de données qui seront créés à l'aide du logiciel Visual Paradigm. L'objectif de cette étape préliminaire est de définir les étapes clés du développement du site web, en mettant en place une structure solide et en déterminant les différentes composantes du système.

Dans le cycle de vie de notre projet, la phase de conception revêt une importance capitale pour produire une application de haute qualité. C'est à ce stade que nous clarifions tout d'abord l'étude technique réalisée, ce qui nous permet d'initier la projection des besoins à l'aide du langage de modélisation UML à travers différents diagrammes tels que le diagramme de cas d'utilisation, le diagramme de séquence et le diagramme de classe. Ensuite, nous présentons la base de données sur laquelle nous avons travaillé. Cette étape constitue une partie essentielle du processus de conception pour assurer la cohérence et l'efficacité de notre application.

### **3.1 Méthodologie de conception :**

Pour faciliter notre travail, nous avons recours au langage de modélisation unifié (UML : Unified Modelling Language). Il s'agit d'une notation standard permettant de modéliser un problème de manière cohérente. Ce langage est le résultat de la fusion de plusieurs méthodes existantes et est aujourd'hui la référence en matière de modélisation orientée objet. En fait, la maîtrise de l'UML est souvent indispensable pour obtenir un poste de développeur orienté objet.

### **3.2 Langage de modélisation UML :**

#### **3.2.1 Définitions :**

UML est l'acronyme anglais de « Unified Modeling Language ». On le traduit en français par « Langage de modélisation unifié ». Il s'agit d'un langage visuel composé d'un ensemble de schémas appelés diagrammes, qui fournissent différentes perspectives sur le projet à traiter. UML nous offre ainsi des diagrammes permettant de représenter le logiciel à développer : son fonctionnement, sa mise en route, les actions pouvant être effectuées par le logiciel, etc. Ces diagrammes offrent une compréhension visuelle claire du système et facilitent la communication entre les différents acteurs du projet.

#### **3.2.2 Les objectifs de l'UML :**

Voici les étapes clés de la phase de conception qui permettront d'élaborer efficacement le système de réservation de taxi :

- Découvrez les erreurs et les lacunes au début du cycle de vie.
- Comprenez les objectifs du projet.
- Présentez les conceptions proposées.
- Raisonnez sur le comportement attendu du système.

### **3.2.3 Les différents types de diagrammes de UML :**

Voici les différents types de diagrammes UML utilisés dans notre projet de réservation de taxi :

- Diagramme de cas d'utilisation UML.
- Diagramme de séquence UML.
- Diagramme de composants UML.
- Diagramme de classe UML.
- Diagramme d'activité UML.
- Diagramme de collaboration UML.
- Diagramme de déploiement UML.
- Diagramme d'état UML.
- Diagramme de paquet UML.

### **3.3 Diagramme des cas d'utilisation :**

Les diagrammes de cas d'utilisation (DCU) sont des outils de modélisation UML utilisés pour fournir une vision d'ensemble du comportement fonctionnel d'un système logiciel. Ils sont particulièrement utiles pour des présentations auprès de la direction ou des parties prenantes d'un projet. Cependant, pour le développement lui-même, les cas d'utilisation sont plus appropriés. Un cas d'utilisation représente une interaction spécifique entre un utilisateur (qu'il s'agisse d'une personne ou d'une machine) et le système. Il constitue une unité de travail significative. Dans un diagramme de cas d'utilisation, les utilisateurs sont appelés acteurs et ils interagissent avec les cas d'utilisation.

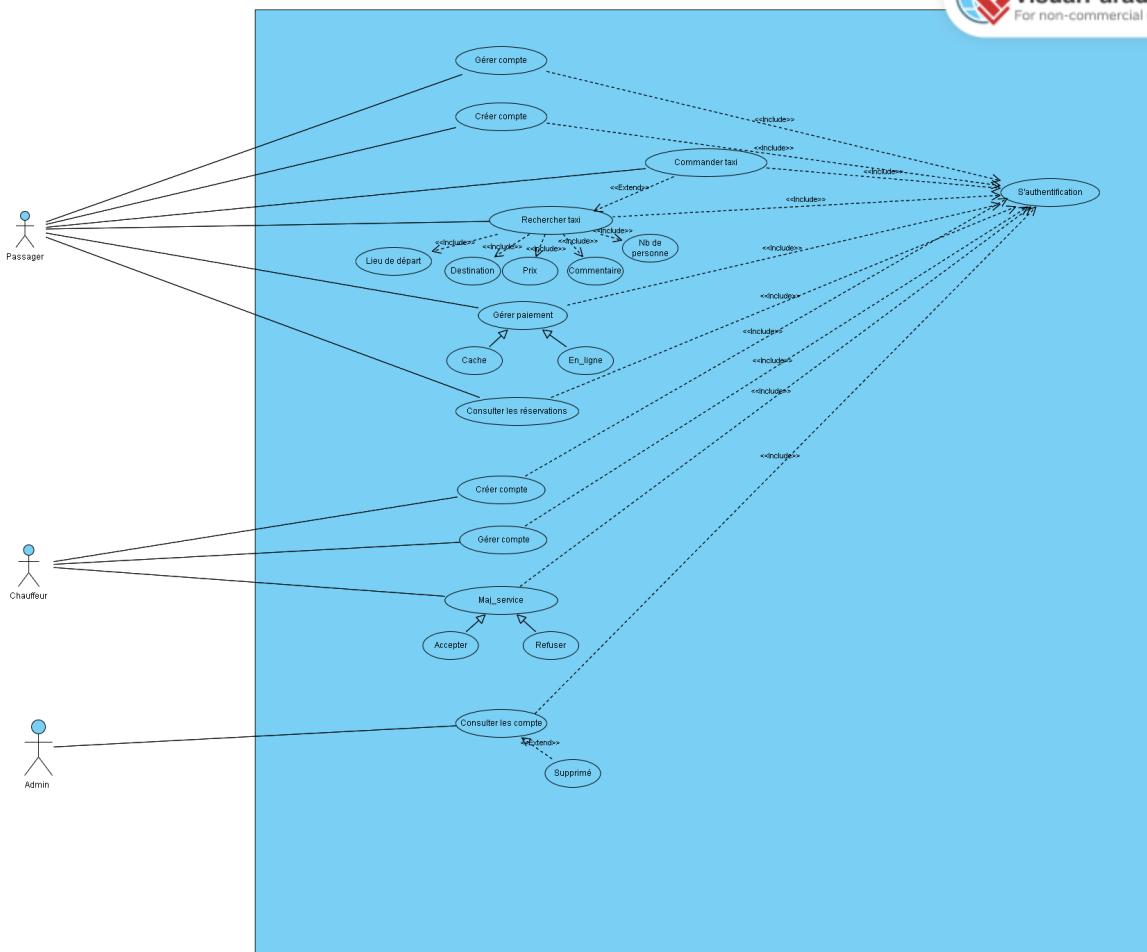


FIGURE 3.1 – diagramme des cas d'utilisation

### 3.4 Diagramme de classes :

Le diagramme de classes est un schéma utilisé en génie logiciel pour représenter les classes et les interfaces d'un système, ainsi que les relations entre elles. Ce diagramme fait partie de la partie statique d'UML, se concentrant sur la structure et les liens entre les éléments, sans considérer les aspects temporels et dynamiques.

Une classe est un regroupement de fonctions et de données (attributs) liées les unes aux autres par un domaine sémantique commun. Les classes sont utilisées dans la programmation orientée objet pour modéliser un programme et décomposer une tâche complexe en plusieurs tâches plus simples.

Les classes peuvent être liées entre elles grâce au mécanisme d'héritage, qui permet de mettre en évidence des relations de parenté. D'autres types de relations sont également possibles entre les classes, chacune étant représentée par un lien spécifique dans le diagramme de classes.

Finalement, les classes sont instanciées pour créer des objets. Une classe agit comme un modèle pour les objets, en décrivant leurs caractéristiques. Lorsqu'ils sont instanciés, les objets prennent des valeurs spécifiques pour chacune de ces caractéristiques.

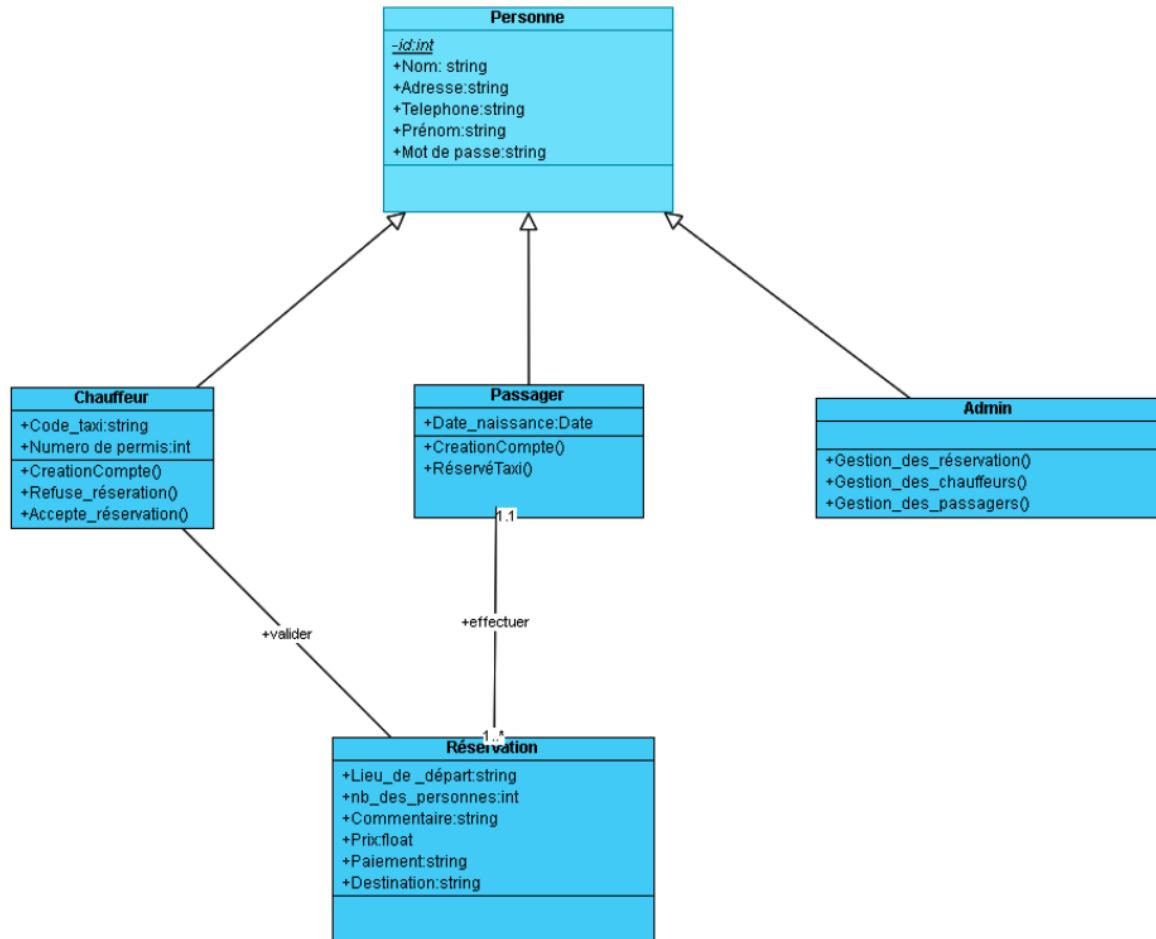


FIGURE 3.2 – diagramme de classes

### 3.5 Diagramme de séquence :

Les diagrammes de séquence sont largement utilisés dans UML pour représenter la dynamique d'un système. Ils se focalisent principalement sur les acteurs, les processus et les objets qui interagissent simultanément, ainsi que sur les messages échangés entre eux pour accomplir

une fonctionnalité avant la fin de leur cycle de vie.

### 3.5.1 Diagramme de séquence d'inscription :

Le diagramme de séquence d'inscription illustre le flux d'actions entre les utilisateurs et le système lors du processus d'inscription, offrant une visualisation claire des interactions clés pour une compréhension rapide.

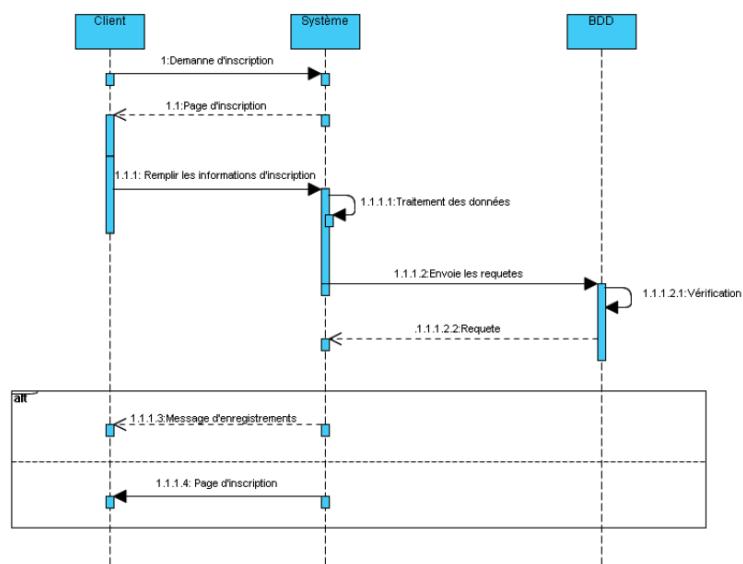


FIGURE 3.3 – diagramme de séquence d'inscription

### 3.5.2 Diagramme de séquence de rechercher taxi :

Le diagramme de séquence de recherche de taxi illustre les interactions entre l'utilisateur et le système lorsqu'il recherche un taxi. Il décompose le flux d'actions de manière visuelle et claire.

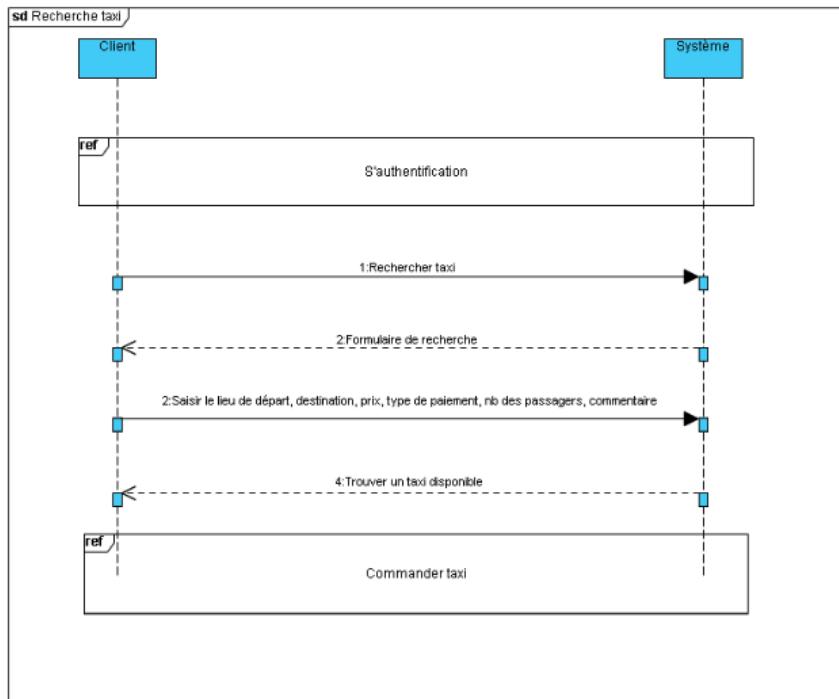


FIGURE 3.4 – diagramme de séquence de recherche taxi

### 3.5.3 Diagramme de séquence de réservation de taxi :

Le diagramme de séquence de réservation de taxi illustre le flux d’actions entre l’utilisateur et le système lors du processus de réservation de taxi, fournissant ainsi une représentation visuelle du fonctionnement de cette fonctionnalité clé.

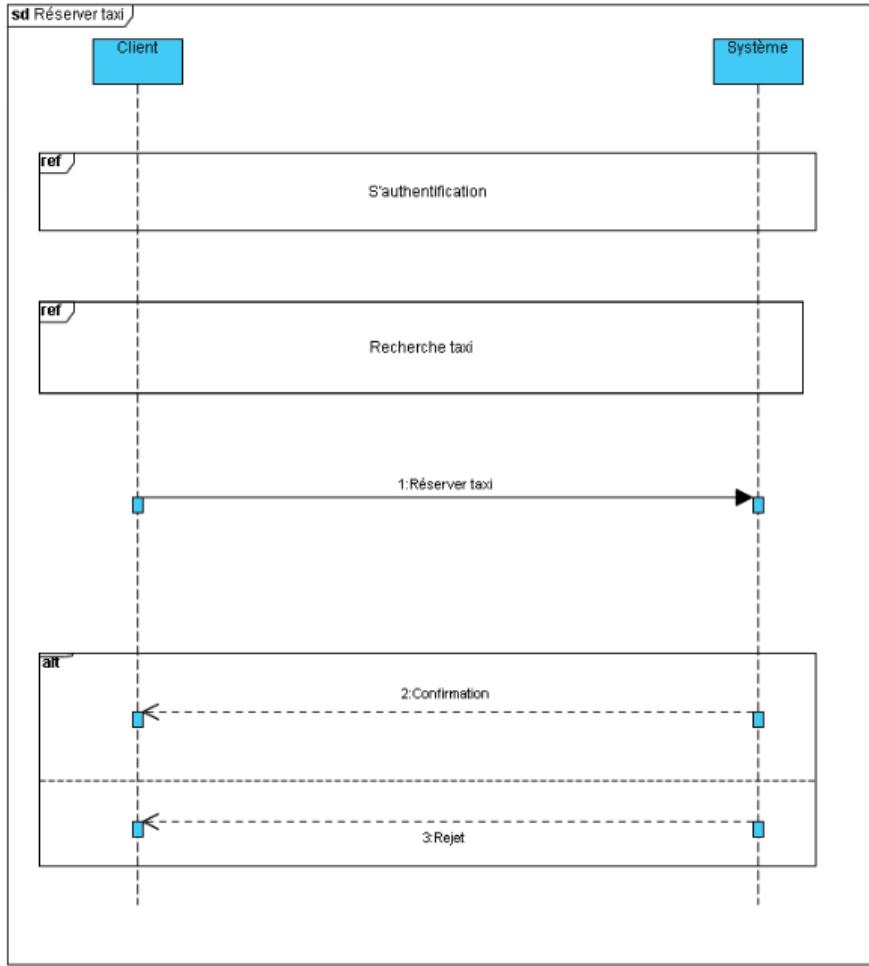


FIGURE 3.5 – diagramme de séquence de réservation de taxi

## 3.6 Conception de base de données :

Pour répondre aux besoins identifiés lors de la phase d’analyse, nous avons procédé à la création d’une base de données composée de cinq tables. Ce choix nous permet de garantir une couverture complète des besoins de notre site tout en respectant les contraintes spécifiées.

### 3.6.1 Modèle rationnel de données :

#### Définition :

Le MLD [5] définit les entités (tables) qui représentent les objets du monde réel, ainsi que les attributs (colonnes) associés à chaque entité. Il décrit également les relations entre les

entités, telles que les relations un-à-un, un-à-plusieurs ou plusieurs-à-plusieurs. Ces relations sont représentées à l'aide de clés primaires et étrangères.

Le MLD est généralement utilisé comme une étape intermédiaire dans le processus de conception d'une base de données, avant d'être traduit en un modèle physique de données (MPD) spécifique à un Système de Gestion de Base de Données (SGBD) particulier. Il sert de référence pour la création de schémas de bases de données, la définition des contraintes d'intégrité et la mise en place des opérations de manipulation des données.

### **MLD de projet :**

Le modèle relationnel de données du site web est structuré de la manière suivante :

#### **Table "Administrateur" :**

- id-Admin (identifiant de l'administrateur).
- Nom (nom de l'administrateur).
- Prénom (prénom de l'administrateur).
- Email (adresse e-mail de l'administrateur).
- Mot de passe (mot de passe de l'administrateur).

#### **Table "Administrateur" :**

- id-Admin (identifiant de l'administrateur).
- Nom (nom de l'administrateur).
- Prénom (prénom de l'administrateur).
- Email (adresse e-mail de l'administrateur).
- Mot de passe (mot de passe de l'administrateur).

#### **Table "Passager" :**

- id-Passager (identifiant du passager)
- Nom (nom du passager)
- Prénom (prénom du passager)
- Adresse (adresse du passager)
- Téléphone (numéro de téléphone du passager)
- Date de naissance (date de naissance du passager)
- Mode de passe (mot de passe du passager)

### **Table "Chauffeur" :**

- id-Chauffeur (identifiant du chauffeur)
- Nom (nom du chauffeur)
- Prénom (prénom du chauffeur)
- Adresse (adresse du chauffeur)
- Téléphone (numéro de téléphone du chauffeur)
- code (code spécifique au chauffeur)
- numéro de permis (numéro de permis du chauffeur)
- Mode de passe (mot de passe du chauffeur)

### **Table "Réservation" :**

- id-Réservation (identifiant de la réservation)
- Lieu (lieu de départ de la réservation)
- Destination (destination de la réservation)
- Prix (prix de la réservation)
- paiement (méthode de paiement utilisée)
- nombre de personne (nombre de personnes pour la réservation)
- commentaire (commentaire associé à la réservation)

## **3.7 Base de données :**

### **3.7.1 Développement de la base de données :**

Nous avons mis en place une base de données avec les caractéristiques suivantes :

- Nom de la base de données : "taxi"
- Nombre de tables : 3
- Nom d'utilisateur : "root"
- Mot de passe : [mot de passe non spécifié]

Avant de pouvoir accéder aux données de la base de données MySQL, il est nécessaire de se connecter au serveur en utilisant un nom d'utilisateur et un mot de passe choisis préalablement.

```
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8 LOG_DEPRECATIONS_CHANNEL=null
9 LOG_LEVEL=debug
10
11 DB_CONNECTION=mysql
12 DB_HOST=127.0.0.1
13 DB_PORT=3306
14 DB_DATABASE=laravel
15 DB_USERNAME=root
16 DB_PASSWORD=
17
18 BROADCAST_DRIVER=log
19 CACHE_DRIVER=file
20 FILESYSTEM_DISK=local
21 QUEUE_CONNECTION=sync
22 SESSION_DRIVER=file
23 SESSION_LIFETIME=120
24
25 MEMCACHED_HOST=127.0.0.1
26
27 REDIS_HOST=127.0.0.1
28 REDIS_PASSWORD=null
29 REDIS_PORT=6379
30
```

FIGURE 3.6 – Connexion au serveur

Une base de données est constituée d'une ou plusieurs tables. La création ou la suppression d'une base de données MySQL nécessite des priviléges CREATE spécifiques. Chaque table de la base de données possède son propre nom unique et est composée de colonnes et de lignes.

```

    /*
public function up(): void
{
    Schema::create('users', function (Blueprint $table) {
        $table->id();
        $table->string('name');
        $table->string('prenom');
        $table->integer('number');
        $table->boolean('is_admin')->default(false);
        $table->date('date');
        $table->string('adresse');
        $table->string('password');
        $table->timestamps();
    });
}

```

FIGURE 3.7 – Création de la table users

```

public function up(): void
{
    Schema::create('chauffeurs', function (Blueprint $table) {
        $table->id();
        $table->string('name');
        $table->string('prenom');
        $table->integer('number');
        $table->boolean('is_approved')->default(false);

        $table->string('code');
        $table->integer('numero');
        $table->string('adresse');
        $table->string('password');
        $table->timestamps();
    });
}

```

FIGURE 3.8 – Création de la table chauffeurs

```

    /*
public function up(): void
{
    Schema::create('reservations', function (Blueprint $table) {
        $table->id();
        $table->string('lieu');
        $table->string('destination');
        $table->integer('prix');
        $table->string('payment');
        $table->unsignedBigInteger('user_id');
        $table->foreign('user_id')->references('id')->on('users');
        // $table->foreignId('cheffaur_id')->nullable()->constrained('cheffieurs');
        $table->integer('nbrpersonne');
        $table->string('commentaire');
        $table->string('etat')->default(false);
        $table->timestamps();
    });
}

```

FIGURE 3.9 – Création de la table réservations

### 3.7.2 Les tables de la base de données :

#### Table de la base de donnée de réservation

<input type="checkbox"/>	<a href="#">Éditer</a> <a href="#">Copier</a> <a href="#">Supprimer</a>	160	en_especies	14	NULL	2	SALUT	1	2023-08-09 18:06:23		
<input type="checkbox"/>	<a href="#">Éditer</a> <a href="#">Copier</a> <a href="#">Supprimer</a>	48	T	D	120	en_ligne	14	NULL	1	SALUT	1
<input type="checkbox"/>	<a href="#">Éditer</a> <a href="#">Copier</a> <a href="#">Supprimer</a>	49	T	D	120	en_ligne	14	NULL	1	SALUT	1
<input type="checkbox"/>	<a href="#">Éditer</a> <a href="#">Copier</a> <a href="#">Supprimer</a>	50	cuzzzane tefwane		120	en_ligne	14	NULL	1	SALUT	1
<input type="checkbox"/>	<a href="#">Éditer</a> <a href="#">Copier</a> <a href="#">Supprimer</a>	51	cuzzzane tefwane		140	en_ligne	14	NULL	1	SALUT	1
<input type="checkbox"/>	<a href="#">Éditer</a> <a href="#">Copier</a> <a href="#">Supprimer</a>	52	sdf	sfg	120	en_ligne	14	NULL	1	SALUT	1
<input type="checkbox"/>	<a href="#">Éditer</a> <a href="#">Copier</a> <a href="#">Supprimer</a>	53	cuzzzane tefwane		120	en_ligne	14	NULL	1	SALUT	1
<input type="checkbox"/>	<a href="#">Éditer</a> <a href="#">Copier</a> <a href="#">Supprimer</a>	54	cuzzzane tefwane		120	en_ligne	14	NULL	2	SALUT	1
<input type="checkbox"/>	<a href="#">Éditer</a> <a href="#">Copier</a> <a href="#">Supprimer</a>	55	cuzzzane tefwane		120	en_ligne	14	NULL	2	SALUT	0

FIGURE 3.10 – Table de données de réservation

#### Table de la base de donnée passager

<input type="checkbox"/>	<a href="#">Éditer</a> <a href="#">Copier</a> <a href="#">Supprimer</a>	14	SCOUFIANE SOUFIANE	867326781	0	2002-03-12	CUEZZZANE	\$2y\$10\$V2TSP4krZcBy\$2t7SMW0KpzQN5w			
<input type="checkbox"/>	<a href="#">Éditer</a> <a href="#">Copier</a> <a href="#">Supprimer</a>	15	ad	ad	123456789	1	2000-03-12	TANGER	\$2y\$10\$YzKcwjYFKhR1cNUBtp0QZMxD6		
<input type="checkbox"/>	<a href="#">Éditer</a> <a href="#">Copier</a> <a href="#">Supprimer</a>	16	yassine	yassine	102030405	0	3333-03-12	CUEZZZANE	\$2y\$10\$Mn5QLBggzbTzMa70qZ2Ub8tw2D		

FIGURE 3.11 – Table de données de passagers

#### Table de la base de donnée de chauffeur :

<input type="checkbox"/>	<a href="#">Éditer</a> <a href="#">Copier</a> <a href="#">Supprimer</a>	11	nabil	ss	11111111	0	AB123	99900	CUEZZZANE	12345678	2023-05-09 18:18:29
<input type="checkbox"/>	<a href="#">Éditer</a> <a href="#">Copier</a> <a href="#">Supprimer</a>	12	ff	ss	22222222	0	AB123	99900	TANGER	12345678	2023-05-12 15:10:01
<input type="checkbox"/>	<a href="#">Éditer</a> <a href="#">Copier</a> <a href="#">Supprimer</a>	13	sd	as	634343434	0	AB123	99900	CUEZZZANE	12345678	2023-05-20 13:21:17
<input type="checkbox"/>	<a href="#">Éditer</a> <a href="#">Copier</a> <a href="#">Supprimer</a>	14	sd	as	634343434	0	AB123	99900	CUEZZZANE	12345678	2023-05-20 13:21:18

FIGURE 3.12 – Table de données de chauffeurs

Nous avons conclu cette phase de conception, qui a consisté à définir les méthodes de travail ainsi que les éléments visuels de notre application web, comprenant à la fois les parties statiques et dynamiques. Dans le prochain chapitre, nous aborderons la dernière étape de réalisation de notre application web, en nous basant sur les mécanismes et les solutions déterminés lors de la phase de conception.

# **Chapitre 4**

## **Réalisation**

Ce chapitre expose les résultats obtenus tout au long de ce projet. Seront abordées les technologies utilisées, ainsi que l'environnement matériel, comprenant le langage de programmation et le système de gestion de base de données (SGBD). Enfin, des captures d'écran de notre site web seront présentées pour illustrer concrètement le travail accompli.

## 4.1 Environnement logiciels :

### 4.1.1 Serveur de la base de données MySQL :

MySQL est un système de gestion de base de données (SGBD) largement utilisé. Selon le type d'application, il présente les caractéristiques suivantes :

- Une performance élevée, offrant des temps de réponse rapides.
- Une facilité d'utilisation supérieure à la plupart des serveurs de bases de données commerciaux.
- Un système de sécurité robuste permettant de gérer les utilisateurs et les machines autorisés à accéder aux différentes bases de données.



FIGURE 4.1 – Logo MySQL

### 4.1.2 PHP MyAdmin :

PhpMyAdmin est une application Web conçue principalement en PHP et distribuée sous la licence GNU GPL. Elle offre une interface conviviale et largement utilisée pour gérer les bases de données MySQL sur un serveur PHP. De nombreux hébergeurs, qu'ils soient gratuits ou payants, la proposent à leurs utilisateurs, évitant ainsi la nécessité de l'installer manuellement. Cette interface pratique permet aux utilisateurs d'exécuter facilement des requêtes de base de données telles que la création, l'insertion, la mise à jour, la suppression et la modification de la structure des tables de la base de données. De plus, elle permet également de gérer les droits d'accès, d'importer/exporter des données et de sauvegarder facilement une base de données sous la forme d'un fichier .sql. Même sans connaissance approfondie du langage SQL, les utilisateurs peuvent utiliser cette application pour tester et interagir avec les requêtes SQL lors de la création d'un site, puis les exécuter en mode batch ultérieurement.



FIGURE 4.2 – Logo de phpMyAdmin

#### 4.1.3 XAMPP :

XAMPP est un pack de logiciels qui permet de créer facilement un serveur Web local, un serveur FTP et un serveur de messagerie électronique. Il s'agit d'une distribution de logiciels libres connue sous l'acronyme X (cross) Apache MariaDB Perl PHP. Ce pack offre une grande flexibilité d'utilisation et est réputé pour son installation simple et rapide. Ainsi, il est accessible à un large public, car aucune connaissance spécifique n'est requise, et il est compatible avec les systèmes d'exploitation les plus courants.



FIGURE 4.3 – logo de XAMPP

#### 4.1.4 Team Gantt :

Team GANTT est un outil qui facilite la planification de projets en utilisant des diagrammes de Gantt. Il offre la possibilité de créer des diagrammes de Gantt, des diagrammes de ressources et des réseaux PERT.



FIGURE 4.4 – Logo de teamgantt

#### 4.1.5 Visual paradigm :

Visual Paradigm (VP-UML) est un outil d'assistance à UML 2, SysML et à la Business Process Modeling Notation (BPMN) du Object Management Group (OMG). En plus de faciliter la modélisation, il propose des fonctionnalités de génération de rapports et d'ingénierie de code, y compris la génération de code. Il est capable d'effectuer une rétro-ingénierie des diagrammes à partir du code et de fournir une ingénierie à double sens pour différentes langues de programmation.

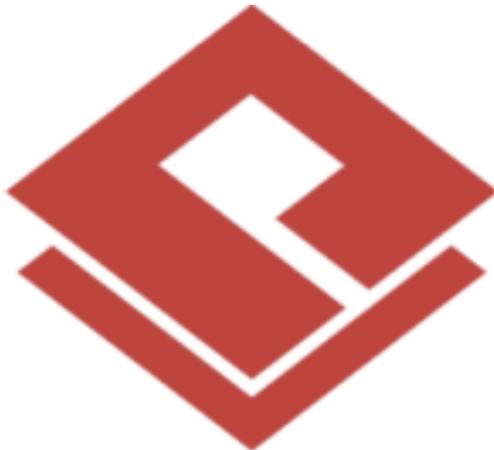


FIGURE 4.5 – Logo de Visual Paradigm

#### 4.1.6 Komodo IDE 12 :

Komodo Edit est un éditeur de texte et un environnement de développement open source. Il est basé sur Scintilla pour l'éditeur et Firefox pour le rendu. Le développement de Komodo Edit a débuté en janvier 2007 afin de compléter le produit commercial Komodo IDE, développé par ActiveState. Une grande partie des fonctionnalités de Komodo proviennent de son interpréteur Python intégré.

Cet éditeur prend en charge plusieurs langages de programmation et propose une colora-

tion syntaxique adaptée à chaque langage. Il dispose également d'un explorateur de fichiers intégré qui facilite la navigation dans les projets. De plus, Komodo Edit offre la possibilité de personnaliser son interface grâce à une sélection de thèmes.



FIGURE 4.6 – Logo de Komodo IDE 12

## 4.2 Langages de programmation :

### 4.2.1 Langage PHP :

PHP : Hypertext Preprocessor (PHP), abrégé PHP, est un langage de programmation libre largement utilisé pour générer des pages Web dynamiques via un serveur HTTP, mais il peut également être utilisé en tant que langage interprété en local. PHP est un langage impératif orienté objet. [1]

Grâce à PHP, de nombreux sites web célèbres tels que Facebook et Wikipédia ont été créés. Il est reconnu comme une pierre angulaire dans le développement de sites web dynamiques et d'applications web.



FIGURE 4.7 – Logo de php

#### 4.2.2 Laravel Framework de PHP :

Laravel est un Framework open-source pour le développement web qui repose sur PHP. Il propose une structure et des fonctionnalités avancées pour simplifier le processus de création d'applications web. Laravel suit le modèle MVC (Modèle-Vue-Contrôleur), ce qui permet de séparer clairement la logique métier, la présentation et le traitement des requêtes.



FIGURE 4.8 – Logo de Laravel

#### 4.2.3 Langage HTML :

HTML (Hypertext Markup Language) est un langage de balisage employé pour organiser le contenu d'une page web. Il utilise des balises pour délimiter et identifier divers éléments tels que les titres, les paragraphes, les liens et les images.



FIGURE 4.9 – Logo de HTML

#### 4.2.4 Langage CSS :

CSS (Cascading Style Sheets) est un langage qui permet d'appliquer des styles et de mettre en forme les pages web. Il offre un contrôle sur les couleurs, les polices, les marges, les tailles et d'autres attributs visuels des éléments présents sur une page. En d'autres termes, il permet de définir l'apparence et la présentation des éléments d'une page web.



FIGURE 4.10 – Logo de CSS

#### 4.2.5 Langage Javascript :

JavaScript est un langage de programmation de scripts largement utilisé, principalement dans les pages web interactives, mais aussi sur les serveurs avec l'utilisation de Node.js. Il s'agit d'un langage orienté objet basé sur le concept de prototypes, ce qui signifie que ses fonctionna-

lités fondamentales et ses principales interfaces sont fournies par des objets qui ne sont pas des instances de classes. En résumé, JavaScript offre une flexibilité unique dans la programmation et permet de créer des fonctionnalités dynamiques et interactives dans les applications web.



FIGURE 4.11 – Logo de Javascript

#### 4.2.6 Langage Bootstrap :

Bootstrap est un ensemble d'outils essentiels pour la conception de sites web et d'applications. Il regroupe des codes HTML et CSS qui facilitent la création du design. En utilisant Bootstrap, les développeurs bénéficient d'une base solide et de composants prêts à l'emploi pour accélérer le processus de développement et garantir un résultat visuellement attrayant.



FIGURE 4.12 – Logo de boorstrap

### 4.3 Les interfaces :

#### 4.3.1 Interface d'authentification :

Lorsque l'utilisateur ouvre le site, l'interface d'authentification s'affiche :

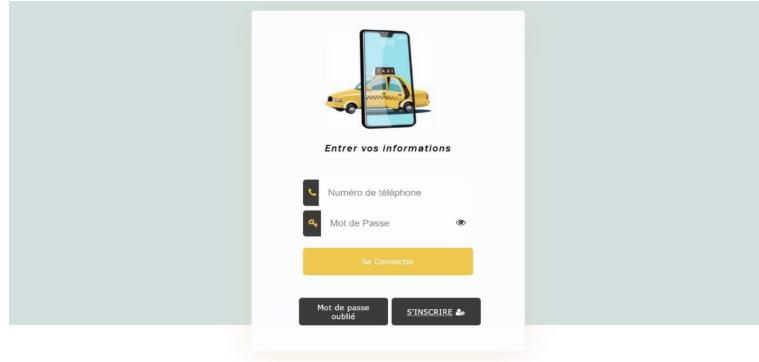


FIGURE 4.13 – Interface d'authentification

Si l'utilisateur veut accéder au site, il insère le numéro de téléphone et le mot de passe puis, il clique sur le bouton Se connecter pour se connecter.

#### 4.3.2 Page de sélection

Dans cette page le visiteur choisit son profil soit chauffeur ou passager.



FIGURE 4.14 – Interface de sélection

#### 4.3.3 Page d'inscription d'un client :

Cette interface offre la possibilité au client et au visiteur de procéder à l'authentification du système. Si le client n'a jamais effectué de réservation, il devra créer un compte. Lorsque les informations saisies sont exactes, vous serez redirigé vers la page principale, mais en cas d'erreur, un message s'affichera, invitant l'utilisateur à réessayer.

## Interface d'inscription de passager

The screenshot shows a registration form titled "Veuillez entrer vos informations". It includes fields for Name, First Name, Phone Number, Date of Birth (with a date picker icon), Address, and Password. A yellow "S'INSCRIRE" button is at the bottom.

FIGURE 4.15 – Interface d'inscription de passager

## Page d'inscription d'un chauffeur :

The screenshot shows a registration form titled "Veuillez entrer vos informations". It includes fields for Name, First Name, Phone Number, Taxi License (with a date picker icon), Driver's License Number, Address, and Password. A yellow "S'INSCRIRE" button is at the bottom.

FIGURE 4.16 – Interface d'inscription de chauffeur

### 4.3.4 Page d'accueil :

La page d'accueil joue le rôle d'une liste initiale qui permet à l'utilisateur de naviguer à travers les différents éléments de l'application Web. Elle constitue la vitrine du site et doit laisser une impression positive au visiteur afin de minimiser le taux de rebond et de l'inciter à explorer davantage le site. Cette page se compose d'une interface d'application Web affichant le nom de l'application et un bouton de connexion, suivi de la possibilité de réserver un taxi.

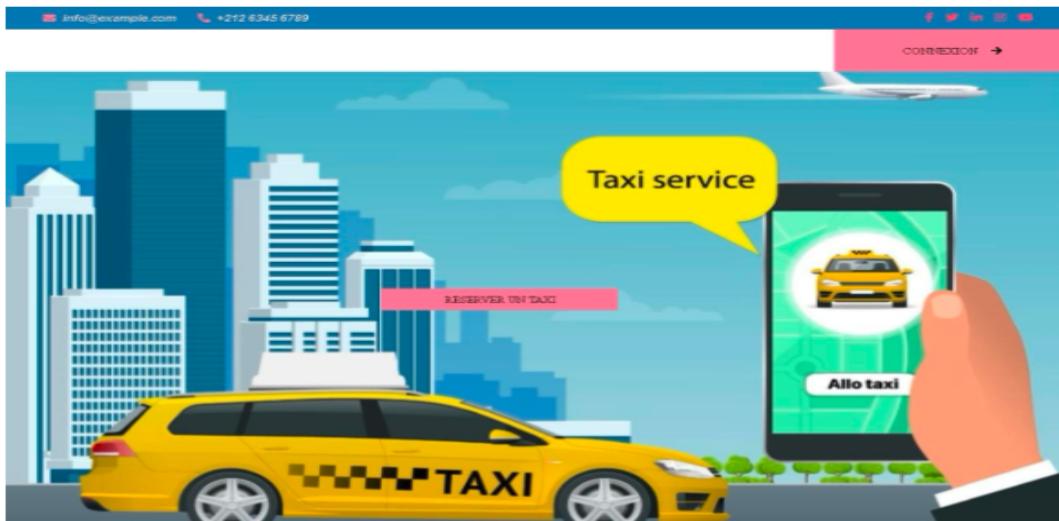


FIGURE 4.17 – Interface d'accueil

#### 4.3.5 Footer du site

Le footer, c'est la partie située en bas de notre page web. Il sert non seulement de conclusion visuelle mais offre également un grand nombre d'options pour exposer des informations approfondies et pertinentes. Il ajoute à notre site une apparence professionnelle dans la mesure de donner une idée générale sur notre site.



FIGURE 4.18 – Footer de site

#### **4.3.6 Page de profil :**

Une fois inscrit, le système redirige le client vers son profil qui comprend trois interfaces principales :

1. Une interface d'accueil qui affiche une photo et des informations spécifiques au client.
2. Une interface regroupant les demandes du client.
3. Une interface présentant l'historique des activités du client et les paramètres de sécurité.

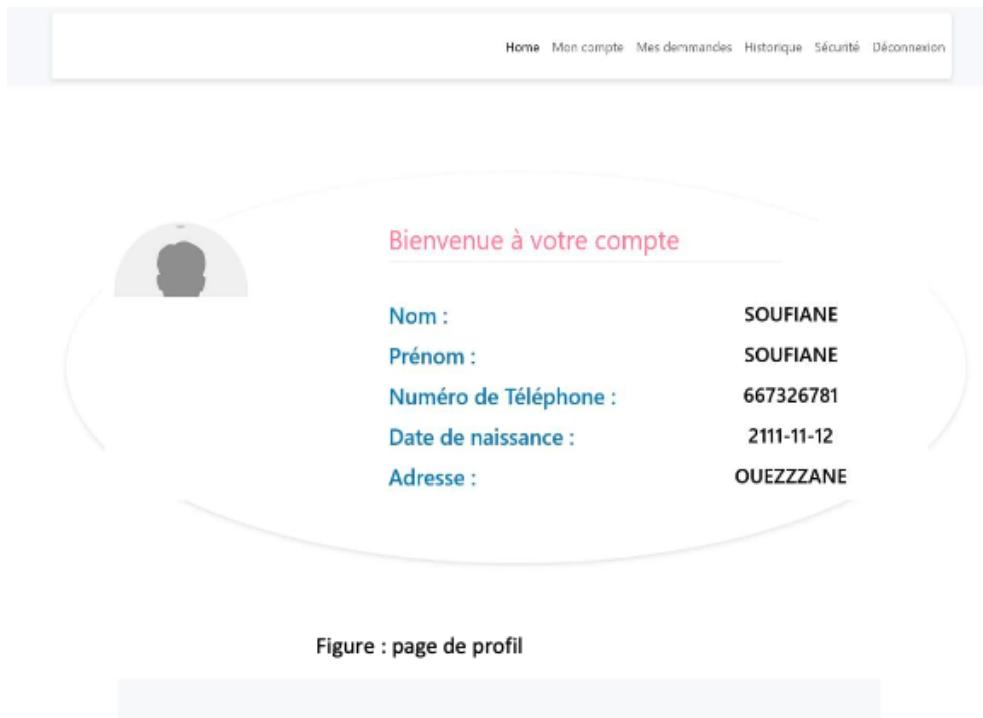


Figure : page de profil

FIGURE 4.19 – Page de profil

#### **4.4 La page principale :**

##### **4.4.1 Page principale de passager :**

Après l'inscription, le système renvoie le client directement à son page principale. Cette page contient une carte et un formulaire de réservation de taxi.

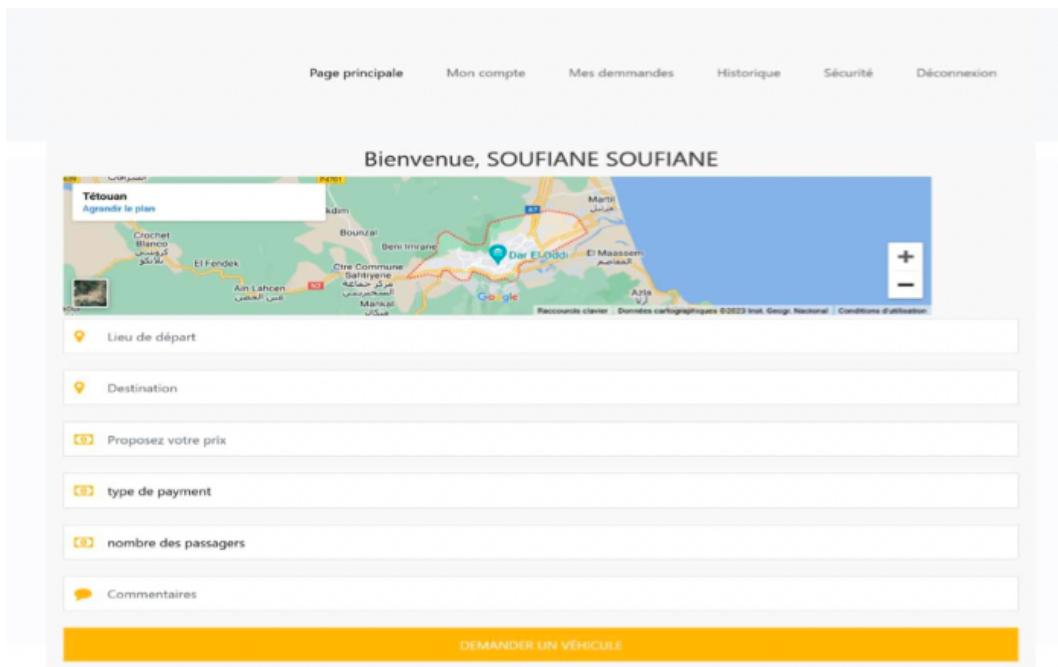


FIGURE 4.20 – Page principale de passager

#### 4.4.2 Page principale de chauffeur :

Après l'inscription, le système redirige le client directement vers sa page principale. Cette page comprend les réservations reçues des passagers ainsi que sa page de profil.

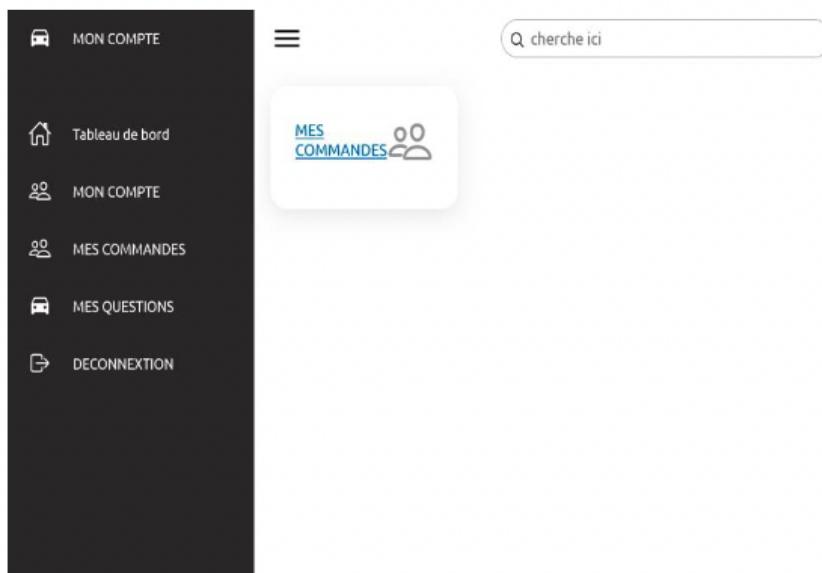


FIGURE 4.21 – Page principale de chauffeur

#### 4.4.3 La page de commande de chauffeur

Cette interface offre la possibilité de consulter et d'accepter les réservations disponibles.

The screenshot shows a web browser window titled "Liste des réservations" at the URL "127.0.0.1:8000/maccommode". The page displays a table of reservations with the following columns: number de utilisateur, Lieu, Destination, Prix, Type de paiement, Nombre de personnes, Commentaire, Date de demande, and Action. There are two rows of data:

number de utilisateur	Lieu	Destination	Prix	Type de paiement	Nombre de personnes	Commentaire	Date de demande	Action
667326781	Ain lahcen	Azla	15	en_especes	2	SALUT	2023-07-09 22:55:35	<button>Accepter</button>
667326781	ouezzane	tetwane	120	en_ligne	1	SALUT	2023-07-09 23:42:07	<button>Accepter</button>

At the bottom left are buttons for "Actualiser" and "Retour".

FIGURE 4.22 – Page de commande de chauffeur

#### 4.4.4 Page d'accueil d'admin :

L'interface d'administration comprend des statistiques et permet à l'administrateur de gérer les réservations, les chauffeurs et les passagers.

The screenshot shows the admin dashboard with a dark sidebar on the left and a main content area on the right.

**Sidebar (ADMINISTRATEUR):**

- Tableau de bord
- Gestion des passagers
- Gestion des chauffeurs
- Gestion des réservations
- Déconnexion

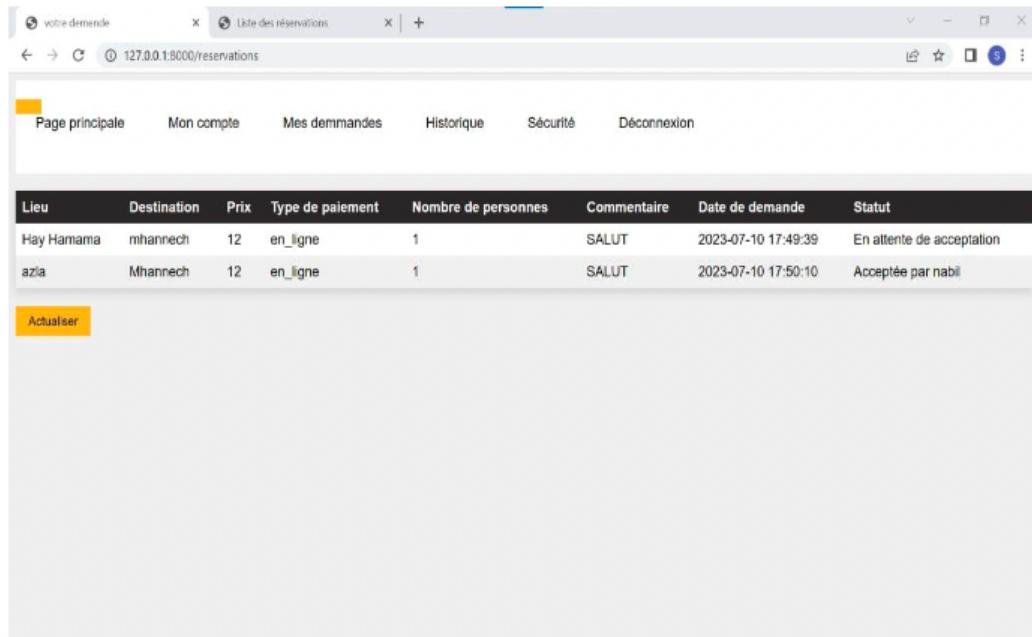
**Main Content Area:**

- A search bar labeled "cherche ici" with a magnifying glass icon.
- Three summary cards:
  - 2 TOTAL DES PASSAGERS** (Icon: two people)
  - 1 TOTAL DES CHAUFFEURS** (Icon: one person)
  - 3 TOTAL DES RÉSERVATIONS** (Icon: car)

FIGURE 4.23 – Page d'accueil d'admin

#### 4.4.5 Page de demandes de passager :

Sur cette page, le passager peut suivre sa réservation et vérifier si un taxi a été trouvé ou non.



The screenshot shows a web browser window with two tabs: "votre demande" and "Liste des réservations". The main content area is titled "Mes demandes". It contains a table with the following data:

Lieu	Destination	Prix	Type de paiement	Nombre de personnes	Commentaire	Date de demande	Statut
Hay Hamama azia	mhannech Mhannech	12	en_ligne en_ligne	1	SALUT SALUT	2023-07-10 17:49:39 2023-07-10 17:50:10	En attente de acceptation Acceptée par nabil

**Actualiser**

FIGURE 4.24 – Page de demande de passager

# Conclusion

Ce projet de fin d'études a été une opportunité enrichissante pour améliorer nos capacités d'abstraction, de conception, d'analyse et de gestion de projet. De plus, il a renforcé notre esprit d'équipe et nous a permis d'apprendre des méthodologies de travail efficaces. En outre, ce projet nous a permis d'approfondir nos connaissances techniques et de développer une rigueur analytique nécessaire pour résoudre les problèmes rencontrés.

L'objectif de ce travail était de créer une application web de réservation de taxi afin de faciliter la vie quotidienne en utilisant les dernières technologies. Notre principal but était de simplifier la réservation de taxis via Internet.

Pour mener à bien ce projet, nous avons commencé par une phase d'analyse des besoins, ce qui nous a permis de rédiger un document de spécifications détaillées contenant les exigences fonctionnelles, non fonctionnelles et techniques à respecter tout au long du cycle de développement. Ensuite, nous avons réalisé une modélisation conceptuelle en utilisant le langage UML. Enfin, nous avons procédé au développement de la solution en utilisant les langages HTML, CSS, Bootstrap, JavaScript, jQuery, PHP et SQL.

Ce projet a été un véritable défi pour nous, nous avons dû apprendre de nouveaux concepts et les appliquer pour mener à bien ce travail. Cependant, les contraintes de temps nous ont empêchés d'atteindre tous nos objectifs. Nous espérons pouvoir ajouter de nouvelles fonctionnalités dans les prochaines mises à jour, telles que la possibilité de paiement en ligne, le développement d'une application mobile connectée à la même base de données, l'ajout de la langue arabe pour faciliter l'utilisation de l'application pour les personnes ne maîtrisant pas le français, ainsi que l'hébergement et le marketing de l'application. Il est essentiel, dans le domaine de l'informatique, de rester vigilant face aux évolutions technologiques et de consacrer nos efforts à faciliter la vie quotidienne en intégrant ces avancées dans notre vie. Nous sommes convaincus que cette

révolution technologique peut être bénéfique pour tous.

# Bibliographie

- [1] Hicham ATTARIUAS. *Programmation web avancée*. 2022/2023.
- [2] Abderrahim EL MHOUTI. *Conception orientée objets (UML)*. 2022/2023.
- [3] Mohammed EL MOUHAJIR. *Gestion de projet*. 2022/2023.
- [4] INTOTHEMINDS. *Étude de marché sur les clients des applications VTC*. 2023. URL : <https://www.intotheminds.com/blog/etude-marche-clients-applications-vtc/> (visité le 12/07/2023).
- [5] Ismail JELLOULI. *Bases de données*. 2022/2023.
- [6] WIKIPÉDIA. *Modèle en cascade*. 2023. URL : [https://fr.wikipedia.org/wiki/Mod%C3%A8le\\_en\\_cascade](https://fr.wikipedia.org/wiki/Mod%C3%A8le_en_cascade) (visité le 12/07/2023).