

**Projet de fin semestre**

**Conception et réalisation d'une application de Système de gestion de location voiture**

**SPECIALITE : Informatique Appliquée de gestion**

encadre par :

Mr Mohamed Saidi

elabore par :

Bouraoui Mohamed

### ANNEE DE FORMATION 2022-2024

Sommaire

[ANNEE DE FORMATION 2022-2024 0](#_Toc167134623)

[Introduction générale 3](#_Toc167134624)

[Chapitre 1 : Analyse et spécification des besoins 4](#_Toc167134625)

[I : Introduction : 4](#_Toc167134626)

[1. La problématique : 4](#_Toc167134627)

[2. Critique de l’existant : 4](#_Toc167134628)

[3. Solution proposées : 4](#_Toc167134629)

[4. Les besoins fonctionnels : 4](#_Toc167134630)

[5. Les besoins non fonctionnels : 5](#_Toc167134631)

[Conclusion : 5](#_Toc167134632)

[Chapitre 2 : Conception 6](#_Toc167134633)

[Chapitre 2 : Conception 6](#_Toc167134634)

[I. Introduction 6](#_Toc167134635)

[1. Présentation du langage UML : 6](#_Toc167134636)

[a. Présentation : 6](#_Toc167134637)

[B.Les diagrammes d’UML : 7](#_Toc167134638)

[c. Critiques : 7](#_Toc167134639)

[2. Les diagrammes : 7](#_Toc167134640)

[a. Présentation et rôles des acteurs : 7](#_Toc167134641)

[b. Vue fonctionnel : 7](#_Toc167134642)

[c. Les diagrammes des cas d’utilisations : 8](#_Toc167134643)

[3. Cas d’utilisation globale : 8](#_Toc167134644)

[a. Vue statique : 9](#_Toc167134645)

[b. Vue dynamique : 10](#_Toc167134646)

[Conclusion : 12](#_Toc167134647)

[Chapitre 3 : 13](#_Toc167134648)

[Choix des outils matériels et logiciels de développement 13](#_Toc167134649)

[Chapitre 3 : Choix des outils matériels et logiciels de développement : 13](#_Toc167134650)

[**I : Introduction :** 13](#_Toc167134651)

[**II : Choix de standard de développement :** 13](#_Toc167134652)

[Chapitre 4 : Réalisation 15](#_Toc167134653)

[I -Introduction 15](#_Toc167134654)

[II Présentation du portail : 15](#_Toc167134655)

[1. Introduction : 15](#_Toc167134656)

[**2 .Présentation des principales pages** 15](#_Toc167134657)

[**3. Module demande et gestion voiture :** 15](#_Toc167134658)

[**II. Quelques interfaces du site** 15](#_Toc167134659)

[3 Interface création Compte : 16](#_Toc167134660)

[4 Interface menu: 17](#_Toc167134661)

[5 Interface gestion des voitures : 17](#_Toc167134662)

[Conclusion générale 19](#_Toc167134663)

# Introduction générale

Aujourd'hui, dans chaque agence de location des voitures utilise une application desktop, pour gérer les taches de location(gérer les voitures, stocker tout ancienne opération ) , et pour faciliter les relations avec les clients, mais avec cette solution il y à toujours des manques le client doit déplacer toujours vers l'agence pour visiter et choisir les voitures, le client ne sait pas les nouvelles promotions des agences à l'exception des supports traditionnels tels que les journaux ce qui rend la relation entre les clients et l'agence désagréable, Dans ces contexte l'agence de location de voiture doit avoir un espace de travail sur l'Internet pour publier les annonces et pour en savoir plus sur les nouveaux développements dans le monde de réservation en ligne pour les voitures.

Notre projet consiste à réaliser un portail web de réservation des voiture en ligne pour la société tunisienne, et les touristes.

La réalisation de projet comporte trois chapitres. Le premier chapitre présente l'étude de faisabilité du projet, Le deuxième chapitre est consacré à la conception et la modélisation des données de portail, Le troisième chapitre présentera les outils de développement et la technologie utilisée pour implémenter le projet.

Enfin, on présentera la mise en œuvre (hébergement) et l'implémentation de portail et les différents Résultats obtenus.

# Chapitre 1 : Analyse et spécification des besoins

## I : Introduction :

L’analyse consiste à comprendre le contexte du système, à produire et de déterminer la faisabilité du projet. Il s’agit de déterminer les acteurs les plus pertinents et les fonctionnalités, de préciser les risques les plus critiques la gestion de location voiture.

## 1. La problématique :

La gestion de location voiture est une tâche difficile à réaliser qui demande beaucoup de temps et d’effort pour réaliser un tout système qualifier qui répond nos besoins de plus il faut garantir la sécurité de système

Alors que les archivistes sont moins nombreux, le désordre des catalogues d’archives et des feuilles de comptabilisation d’inventaire ont mène à plusieurs difficultés en terme administrative.

## 2. Critique de l’existant :

* On trouve une difficulté à manipuler les documents actuels.
* Manque des procédures automatisées de gestion de données.

## 3. Solution proposées :

* Réaliser une application de suivi de matériel informatique. Il s’agit de modéliser et implémenter les données.
* Sécuriser l’application même pour une simple consultation, ou un traitement qui modifie les données ainsi que sa base de données.
* Assurer une alimentation de la base de données en temps quasi-réel avec le maximum de fiabilité.

## 4. Les besoins fonctionnels :

La validation de ce système peut laisser entrevoir la possibilité d’intégrer d’autres systèmes irrigues dans la même région, à un niveau de décision plus élève.

* Gestion des données enregistrées :
* Ajout des nouvelles données.
* Modification de données.
* Suppression de données.

## 5. Les besoins non fonctionnels :

Les besoins non fonctionnels sont les besoins qui permettent au logiciel d’être plus convivial, plus simple à utiliser et qui rendent plus performent, certains d’entre eux s’appliquent à tous les types de logiciels, d’autres sont spécifiques à une catégorie de logiciel particulière. Nous présentant dans ce paragraphe quelques besoins contraintes pour que notre application soit dans les mesures de répondre aux divers besoins. Pour cette raison, nous avons décidé d’assurer les points suivants :

* **Ergonomie des interfaces** : Le système doit présenter une interface graphique homme machine ergonomique et facile à utiliser de façon que les différents éléments de l’application ainsi que les divers informations concernent les tâches doivent être manipulées d’une façon aisée à travers cette interface.
* **Sécurité** : Le système doit inclure une méthode d’authentification permettant de sécuriser les données de façon que le système ne puisse être utilisé que par la main tenancier qui possède le mot de passe.
* **Fiabilité**: Le système doit permettre de faire des recherches pour diminuer le temps d’accès à une information.

## Conclusion :

Dans ce chapitre nous avons fait une étude de l’existant avec une présentation des besoins du demandant de gestion de voiture et d’étudier l’architecture possible pour l’élaboration de notre système. Dans le prochain chapitre nous allons entamer la partie conceptuelle qui doit être adoptée pour l’application en question.

# Chapitre 2 : Conception

# Chapitre 2 : Conception

## I. Introduction

La phase de conception permet de traduire les besoins fonctionnels et les contraintes issues du cahier de charge dans un langage plus formalisé.

Le choix du langage UML (UnifiedModelingLanguage) comme langage de modélisation revient à ce qui il soit adapté à tous les phases du développement, compatible avec toutes les techniques de réalisation et aussi indépendante des langages de programmation.

Pour concevoir en langage unifié de modélisation, nous pouvons se trouver face à plusieurs AGL (Atelier de Génie Logiciel) parmi lesquels nous avons choisi d’utiliser Power AMC car il est performant et facile à manipuler.

## 1. Présentation du langage UML :

## a. Présentation :

UML (en anglais : UnifiedModelingLanguage, << langage de modélisation unifié>>) est un langage de modélisation des données et des traitements. C’est une formalisation non propriétaire de la modélisation objet utilisée en génie logiciel. Accompli de la fusion des précédents langages de modélisation objet Booch, OMT, OOSE, UML est un standard défini par l’OMG (Object Management Group).

Le formalisme UML est composé de 13 types de diagrammes. UML n’étant pas une méthode, leur utilisation est laissée à l’appréciation de chacun. De même, on peut se contenter de modéliser seulement partiellement un système, par exemple certaines parties critiques.

UML définit neuf types de diagramme devisés en trois catégories :

* **Diagramme fonctionnel :** diagramme de cas d’utilisation**.**
* **Diagrammes statiques :** diagramme de classe, diagramme d’objet, diagramme de composant, diagramme de déploiement**.**
* **Diagramme dynamique :** diagramme d’activité, diagramme de séquence

### B.Les diagrammes d’UML :

Les diagrammes sont des éléments graphiques. Ceux-ci décrivent le contenu des vues, qui sont des notions abstraites.

* **Diagramme de classe**: il représente les classes intervenant dans le système.
* **Diagramme d’objets :** il sert à représenter les instances des classes (objets) utilisées dans le système.
* **Diagramme de cas d’utilisation :** il décrit les possibilités d’interaction entre le système et les acteurs, c’est-à-dire toutes les fonctionnalités que doit fournir le système.

### c. Critiques :

Un des tout premiers avantages de l’UML est de faire se rencontrer et communiquer l’utilisateur et main-tenanciers. UML permet également outre le fait de se concentrer sur l’utilisateur de documenter très clairement les besoins exprimés par ces derniers, dans le cadre d’une gestion de projet de développement qui va de la conception jusqu’au déploiement de l’application, car il s’utilise sur l’ensemble du cycle de développement de logiciels (analyse, conception, réalisation)

## 2. Les diagrammes :

## a. Présentation et rôles des acteurs :

Un acteur est une abstraction d’une entité externe au système. En général, il permet d’interagir avec ce système. Ainsi, il peut lire ou modifier des informations de ce dernier et il ne peut être qu’un utilisateur humain ou un système connexe qui interagit avec l’application courante

* **Administrateur :** l’administration de parc va permettre à l’administrateur d’effectuer le suivi technique de ses matériels informatiques et d’administrer les ressources informatiques de son entreprise. Pour ceci, il va disposer de plusieurs types d’outils qui vont répondre aux différents besoins de l’entreprise.

## b. Vue fonctionnel :

Cette vus est capable de décrire le processus de transformation des informations à travers les digrammes de ces d’utilisation.

Les, use cases, permettent de structurer les besoins des utilisateurs et les objectifs correspondants d’un système. Ils centrent l’expression des exigences du système sur ses utilisateurs partant du principe que les objectifs sont touts motivés. Nous présentons dans cette partie, les interactions des différents acteurs avec notre application.

### c. Les diagrammes des cas d’utilisations :

Les cas d’utilisation permettant d’exprimer le besoin des utilisateurs d’un système, ils ont donc une vision orientée utilisateur, ils permettent donc de mettre en évidence les relations fonctionnelles entre les acteurs et le système étudié

L’objectif poursuivi par les ces d’utilisateurs est de permettre de décrire, dans des documents lisibles par tous, la finalité des interactions du système et des utilisateurs.

## 3. Cas d’utilisation globale :

Au niveau de cette partie on va présenter le cas d’utilisation générale de notre application.

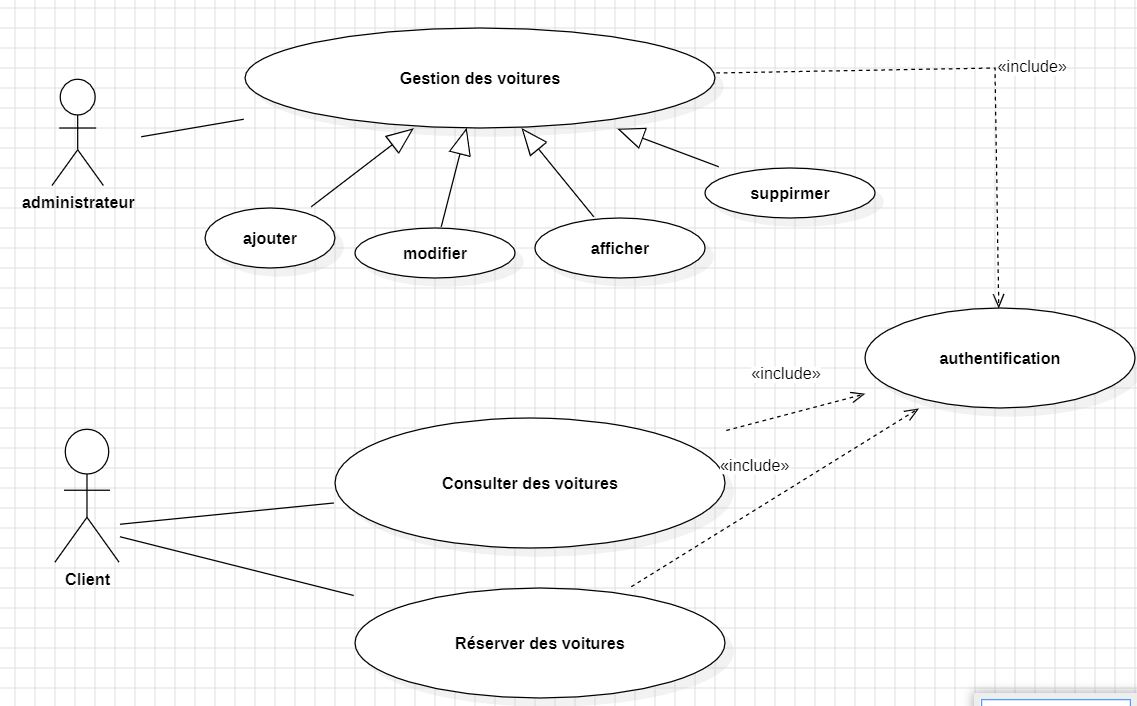


Figure 1 : cas d’utilisation global

Dans ce qui suite, nous détaillions ce cas d’utilisation :

* **Gestion des voitures:** cette fonction permet aux administrateurs peut ajouter, modifier, supprimer et recherche voiture dans la plateforme

### Vue statique :

* **Diagramme de classe :**

Le diagramme de classe décrit la structure du système à l’aide des classes, des paquetages et des relations. Il permet d’identifier les classes, les relations statiques et dynamiques entre eux.

Ce diagramme détermine également les cardinalités des relations, les attributs des classes ainsi que la méthode et leurs paramètres.

C’est le diagramme le plus important de la modélisation orienté objet, il représente les classes constituant le système et les associations entre elles, il regroupe les classes du monde réel et celles nécessaires à l’implémentation du système et permet de générer la base de données et les classes utilisées dans la programmation

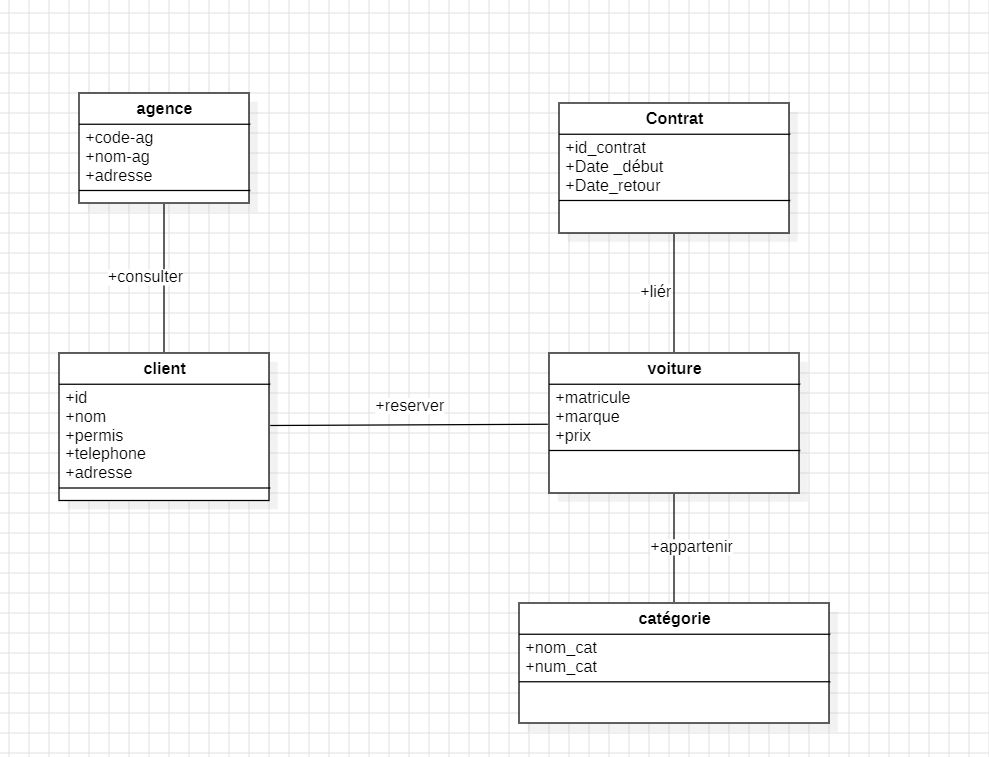


Figure 2 : Diagramme de classe

### Vue dynamique :

* **Diagramme des séquences :**

Le diagramme de séquence est un diagramme d’interaction UML. Il représente les échanges des messages entre les objets de point de vue temporel et permet de détailler un diagramme de cas d’utilisation en décrivant un scénario.

Il se base sur les concepts suivants :

* **Objet :** description d’un objet du monde réel (instance de classe). Il peut être une personne ou une chose.
* **Message :** c’est le véhicule de la communication entre les objets.

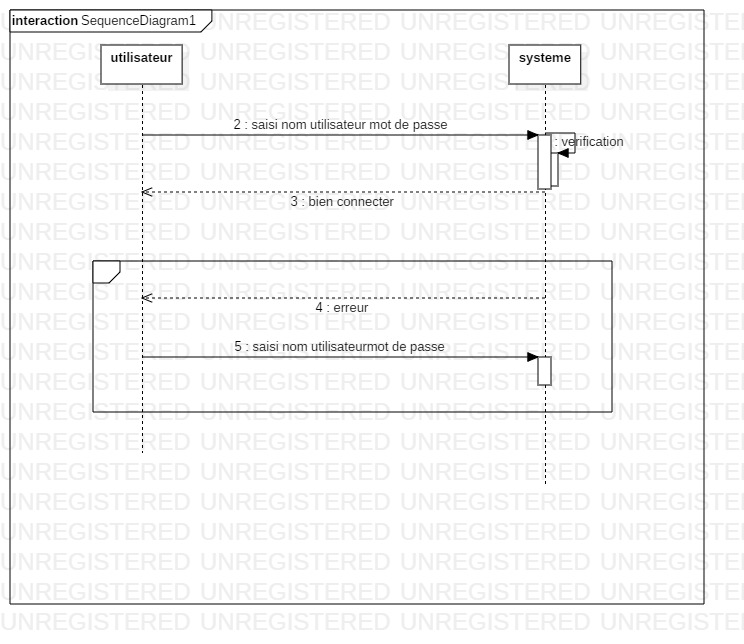
****

Figure 3 : diagramme de séquence du cas « authentification »

* L’administrateur saisie son login et son mot de passe.
* L’administrateur click sur le bouton OK.
* Le système accède à la table authentification pour vérifier les paramètres saisis.
* Si les paramètres existent le système affiche «  Accès autorisé »
* Si les paramètres n’existent pas le système affiche «  Accès refusé »

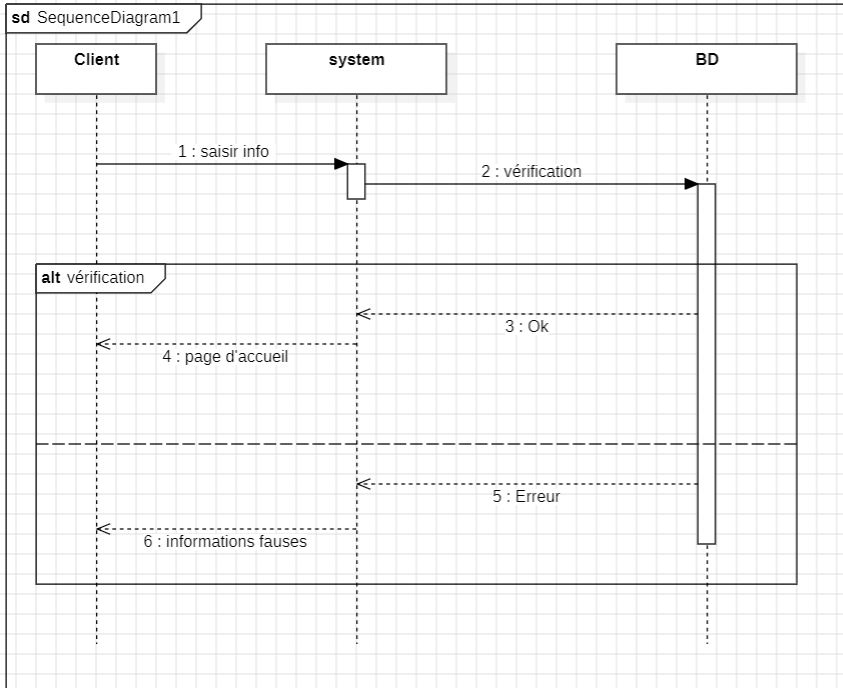


Figure 4:diagramme de séquence du cas « Ajouter un Client »

## 

## Conclusion :

A travers ce chapitre nous avons présenté notre conception de l’application dont nous avons fourni, dans une première étape, une conception globale à travers un schéma générale décrivant l’organisation de notre application. Ensuite nous avons présenté la conception détaillée de l’application à travers la description des différents diagrammes sur lesquels se base l’application de suivi de matériel informatique, le diagramme de cas d’utilisation ? Le diagramme de classes, et un ensemble de diagrammes de séquences.

La conception est terminée par le schéma entité association de la base de données pour passer enfin au volet de notre rapport qui est la partie choix des outils de développement

# Chapitre 3 :

# Choix des outils matériels et logiciels de développement

# 

# Chapitre 3 : Choix des outils matériels et logiciels de développement :

## **I : Introduction :**

Dans tout ce qui précède, nous avons essayé de suivre un enchainement logique qui nous permet de bâtir une application adéquate au suivi de la maintenance de matériel informatique. Ceci étant de la validité et notre approche ne peut être confirmée que par la présentation des résultats obtenus et la mise en valeur des gains réalisés. C’est pour ceci que nous consacrons ce chapitre à la concrétisation des idées antérieurement développées.

Nous consacrerons ce chapitre à la présentation du contexte matériel et logiciel de développement de l’application.

## **II : Choix de standard de développement :**

Pour le choix de la plateforme, nous étions confortés à la décision de développement d’une application En effet, notre choix a été choisir Visual Basic pour l’interface de l’application et Access pour notre base de données.



Figure 5: logo NetBeans IDE

Visual Basic est un ensemble de feuilles et de modules. Une fenêtre et le code qui lui est associé sont regroupés dans une feuille. Une feuille contient le dessin de la fenêtre, les objets qu’elle contient et le code de déclaration des fonctions, des objets, des procédures, des constantes, des variables globales à l’application.

* Outil pour la conception :

**StarUML :**



Figure 6 :StarUML

**StarUML** est un logiciel de modélisation [UML](http://fr.wikipedia.org/wiki/UML_(informatique)), cédé comme [open source](http://fr.wikipedia.org/wiki/Open_source) par son éditeur, à la fin de son exploitation commerciale, sous une licence modifiée de GNU GPL.

# Chapitre 4 : Réalisation

Chapitre 4 : Réalisation

## I -Introduction

Après l’analyse conceptuelle du projet, nous allons passer à l’étape de test et réalisation. En effet, cette étape permettra de vérifier si l’application en cours a atteint la satisfaction des utilisateurs.

Dans ce chapitre, nous allons présenter notre application

## II Présentation du portail :

## 1. Introduction :

Dans cette partie, nous présentons le travail réalisé à travers une présentation des pages tels que l’interface d’accueil, l’interface d’authentification, l’interface de la consultation de la base (gestion de voiture)…

## **2 .Présentation des principales pages**

Cette partie et une visualisation des relations entre les différentes pages du portail pour pouvoir comprendre le fonctionnement de l’application.

### **3. Module demande et gestion voiture :**

l’application identifie l’utilisateur ; s’il est personnel il se dirige vers l’interface de la demande où il peut déclarer et faire la description de la panne par rapport. Si l’utilisateur est un administrateur

**II. Quelques interfaces du site**Dans cette partie, nous allons présenter les interfaces utilisées dans notre application.

## 

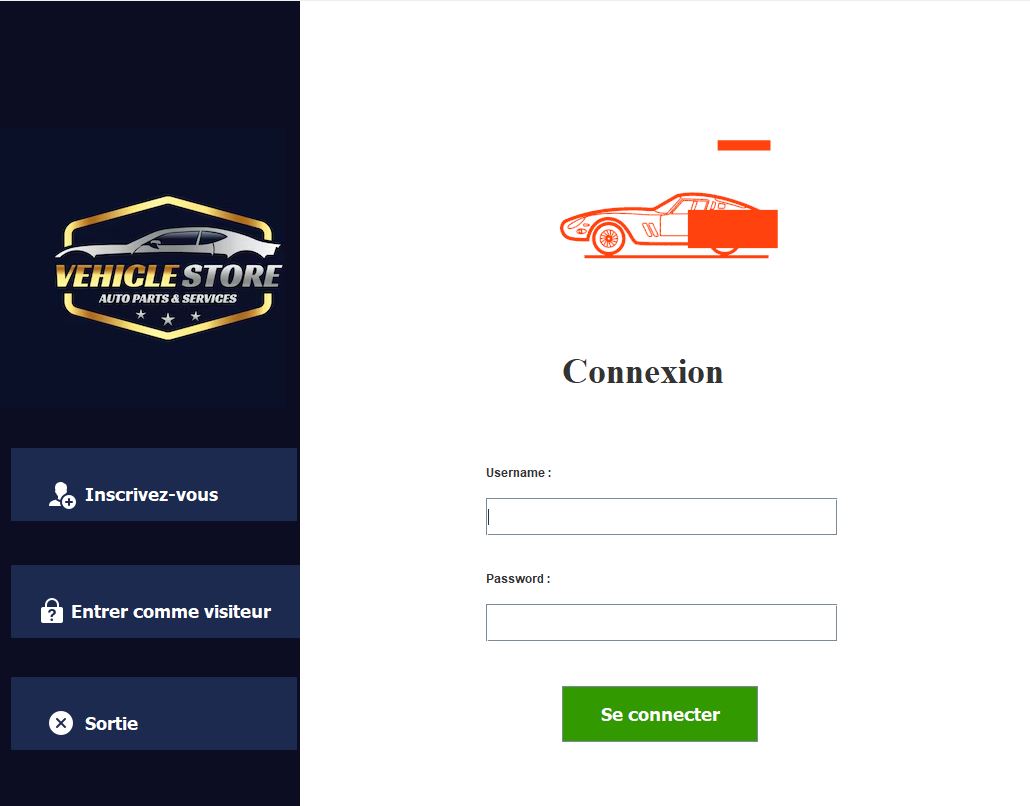


Figure 7 : interface Connexion

## 3 Interface création Compte :

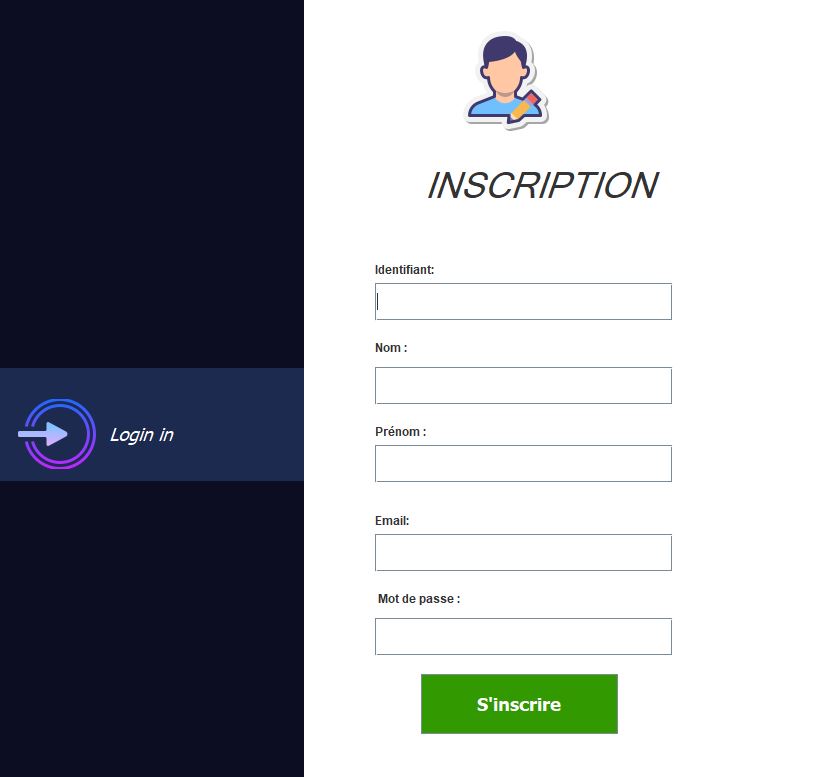


Figure 8 : Interface création compte

## 4 Interface menu:

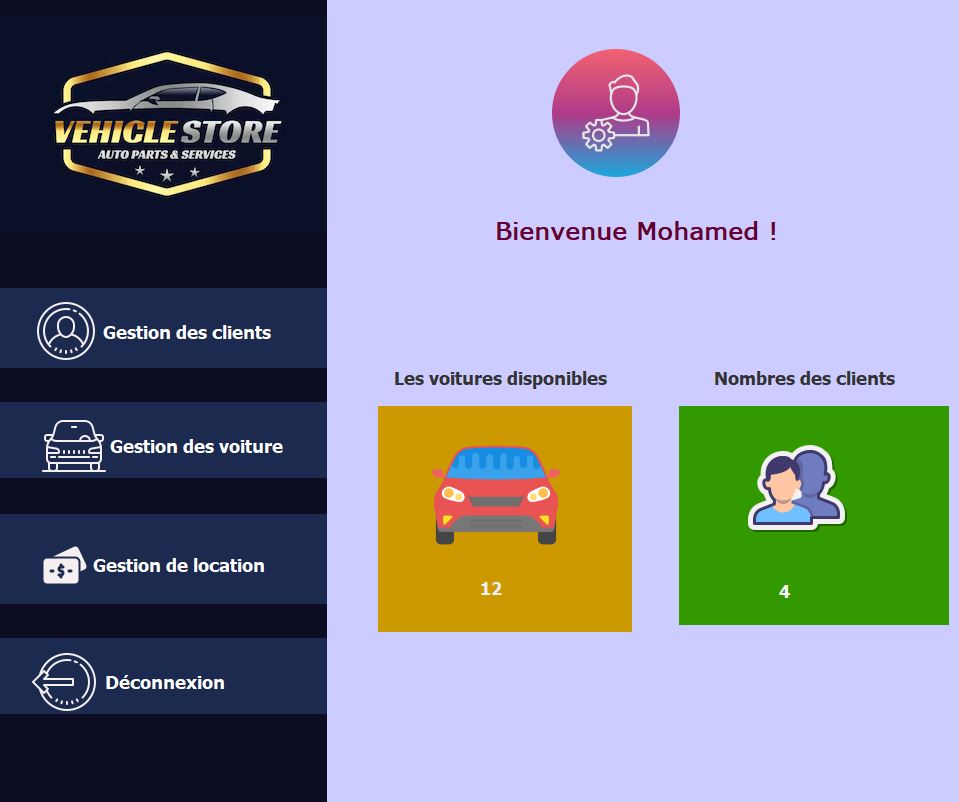
****

Figure 9 : interface Menu administrateur

## 5 Interface gestion des voitures :

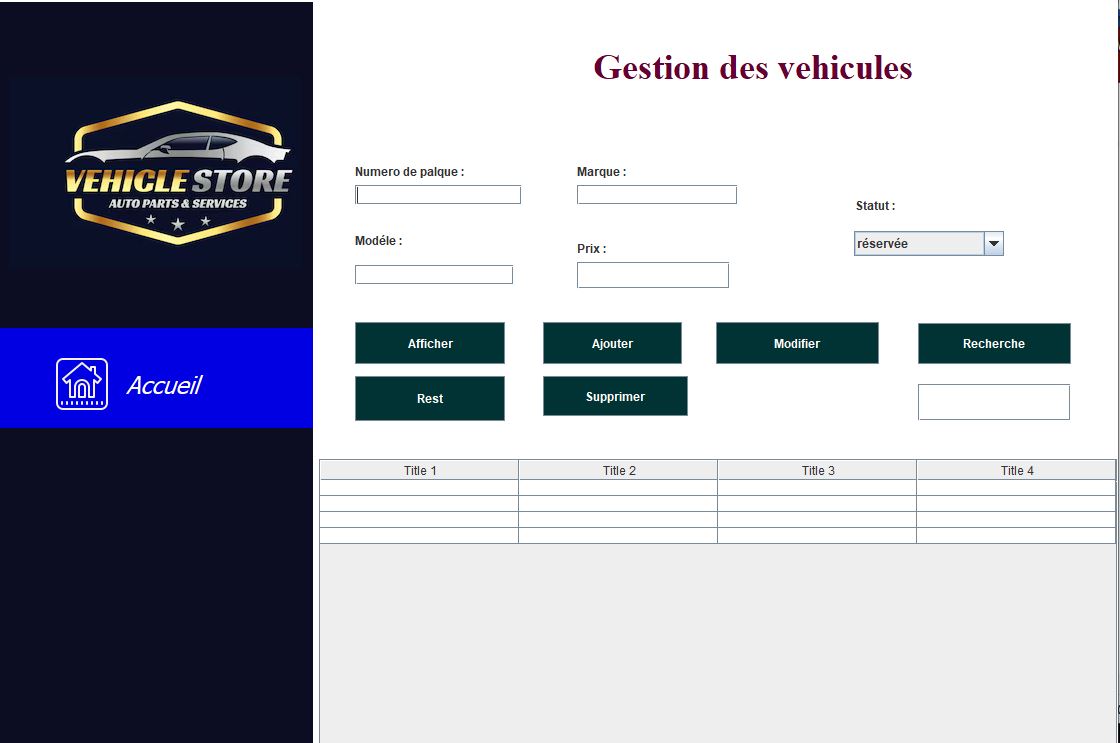
****

Figure 10 : interface gestion voiture



## 

Figure 11: interface gestion des clients



Figure 12 : interface gestion location

# 

# Conclusion générale

Au terme de rapport, nous pouvons conclure que ce projet de fin d’étude nous a donné une occasion opportune nous permettant de confronter l’acquis théorique à l’environnement pratique. En effet, le projet que nous a permis de prendre certaines responsabilités, par la suite de montrer de plus en plus nos connaissances théoriques et pratiques. C’est là que réside la valeur d’un tel projet de fin d’étude qui marie les exigences de la vie professionnelle aux côtés bénéfiques de l’enseignement pratique que nous eut à

Du point de technique, ce projet nous a permis de nous adapter avec l’environnement du développement informatique, la chose qui joue un rôle très important dans notre formation. Certes, ce travail pourra être enrichi et amélioré. Les points d’améliorations que nous constatons et l’ajout d’une application et liée à la condition personnalisée par la gestion de voiture.