

---

# **Cahier des Charges – AutoLoc (Plateforme de Location de Voitures)**

**Nom: [Mustapha EL GHAZZAL]**

**Phone: [0675868916]**

**Email: [m.elghazzal7457@uca.ac.ma]**



---

## Cahier des Charges – AutoLoc

**Version :** 1.0

**Date :** Juillet 2025

**Auteur :** Mustapha El Ghazzal

---

### 1. Présentation du projet

AutoLoc est une plateforme web full stack de location de voitures à destination des clients particuliers et professionnels.

Elle permet la recherche, la réservation, la gestion des véhicules, le paiement, et l'administration du parc automobile.

Le projet intègre également des fonctionnalités d'intelligence artificielle (IA) pour suggérer des voitures selon les préférences des clients.

---

### 2. Objectifs du projet

Créer une interface utilisateur moderne, responsive et intuitive.

Permettre aux utilisateurs de s'inscrire, se connecter, et réserver un véhicule.

Offrir un tableau de bord administrateur pour gérer voitures, utilisateurs, documents et réservations.

Intégrer un module IA (via API GPT ou Gemini) pour :

Suggestions intelligentes de véhicules.

Chatbot pour aide à la réservation.

Gérer l'historique des statuts des voitures (entretien, disponible, panne...).

Gérer l'envoi et la vérification des documents client (CIN, permis).

Ajouter un système d'avis et de notation par les clients.

### 3. Fonctionnalités principales

#### **Frontend (interface utilisateur) :**

Page d'accueil avec les voitures à louer.

Fiches détaillées des voitures.

Filtres de recherche (marque, prix, type...).

Pages : inscription, connexion, espace client.

Tableau de bord utilisateur : réservations, documents.

Système de paiement (fictif ou réel).

Chatbot IA d'assistance.

Ajout d'avis/notes sur les véhicules.

#### **Backend (traitement des données) :**

Authentification sécurisée (avec mot de passe hashé).

API REST pour toutes les opérations CRUD.

Gestion des entités : voitures, utilisateurs, documents, réservations, avis, historiques.

Suivi des statuts des voitures.

Suggestion IA de voitures.

Upload et gestion des documents client.

### **4. Public cible**

Particuliers souhaitant louer une voiture simplement et rapidement.

Agences ou loueurs qui veulent un système moderne de gestion.

Administrateurs du système (agents, gérants).

### **5. Contraintes techniques**

Langages utilisés : HTML, CSS, JavaScript, PHP , MySQL.

Possibilité d'utiliser React pour le frontend .

Hébergement local (XAMPP).

API IA : Gemini.

Responsive design (mobile, tablette, desktop).

## 6. Structure de la base de données (entités principales)

UTILISATEUR (id\_user, nom, email, mot\_de\_passe, rôle).

VOITURE (id\_voiture, marque, modèle, statut, prix, image).

RESERVATION (id\_reservation, dates, user, voiture, montant).

DOCUMENT (id\_doc, type\_doc, utilisateur, date\_upload).

HISTORIQUE\_STATUT (id\_historique, statut, date, voiture).

AVIS (id\_avis, note, commentaire, utilisateur, voiture).

## 7. Sécurité

Hashage des mots de passe .

Authentification par sessions.

Validation des formulaires côté client et serveur.

Séparation des rôles (admin / client).

## 8. Livrables attendus

Code source complet (frontend + backend).

Fichier .sql de la base de données.

Manuel d'utilisation.

Ce cahier des charges.

Diagrammes (MCD, MLD, UML)

---

## 9. Planning prévisionnel

### Tâches principales

- 1 Analyse, conception MCD/MLD, Figma
  - 2 Développement backend : BDD, API, modèles PHP
  - 3 Développement frontend : pages HTML, JS, CSS
  - 4 Intégration IA, test complet, documentation
-