

CAHIER DES CHARGES DU PROJET CARTES GRISES



Réalisé par KANOUNI Sofiane

Sommaire

Sommaire	2
I. Contexte du projet	3
1. Carte grises : présentation	3
2. Objectifs	3
3. Date de rendu du projet	3
II. Besoins fonctionnels	3
III. Ressources nécessaires à la réalisation du projet	4
1. Besoins matériels	4
2. Besoins logiciels	4
IV. Gestion du projet	4
V. Conceptions du projet	5
1. Le front-end	5
2. Le back-end	10
VI. Technologies utilisées	13
3. Langages de programmation applicatif	13
4. Base de données	13
VII. Sécurité	13
6. Protection contre les injections SQL	13

I. Contexte du projet

1. Carte grises : présentation

Le certificat d'immatriculation, communément appelé carte grise, est un document officiel indispensable pour tout véhicule en circulation en France. Émis par l'État, il regroupe des informations clés comme l'identité du propriétaire, les spécificités du véhicule (marque, modèle, puissance...) ainsi que son numéro d'immatriculation. Obligatoire pour rouler en toute légalité, il sert aussi à retracer l'historique du véhicule lors d'une vente ou d'un contrôle routier.

2. Objectifs

Actuellement en poste en tant que concepteur et développeur à la Direction des systèmes d'information de la préfecture de votre département, vous êtes sollicité par la responsable du service des cartes grises pour faire évoluer leur application métier. Toutefois, aucun document de conception n'est disponible. Votre mission consiste donc à établir une documentation détaillée de l'application existante afin de faciliter l'analyse et la réflexion sur son évolution future.

3. Date de rendu du projet

Le projet doit être rendu au plus tard le 06/03/2025

II. Besoins fonctionnels

L'application métier de gestion des cartes grises devra présenter :

- Des marques de voitures
- Des modèles de voitures
- Les propriétaires
- Les véhicules
- Les propriétés
- Device compatible

Les informations seront conservées dans une base de données relationnelle afin de simplifier leur gestion et leur mise à jour. Elles pourront être modifiées et administrées directement.

III. Ressources nécessaires à la réalisation du projet

1. Besoins matériels

- Ordinateur portable (Connecter à internet)
- Fixe (Connecter à internet)

2. Besoins logiciels

Environnement de développement (IDE) : Visual Studio Code

Plateforme de développement collaboratif : Github

Outils de gestion de projet : Trello

Conception UML et arborescence : Visual Paradigm online

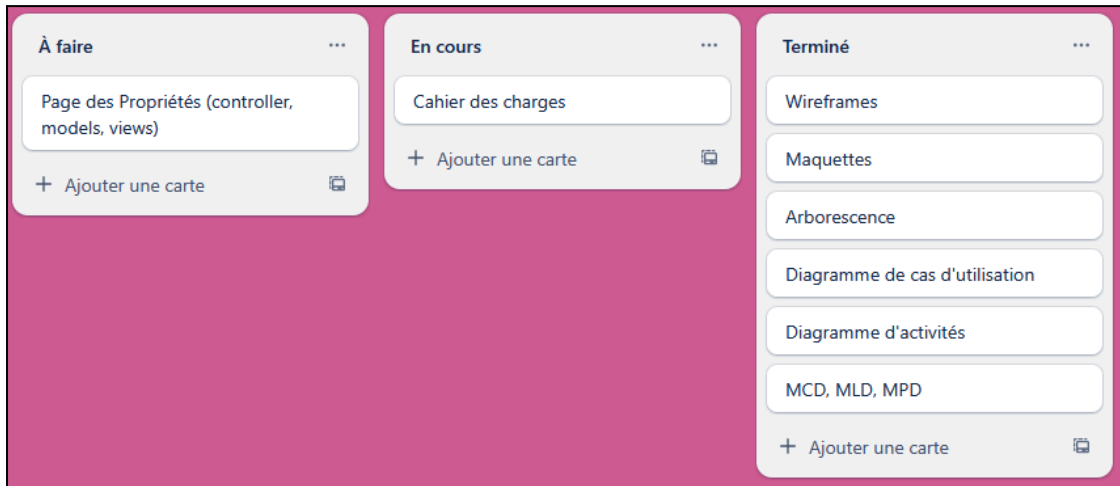
Maquettage : Figma

Conception de base de données : Mocodo

Sources documentaire possible : StackOverFlow

IV. Gestion du projet

Pour la gestion du projet, nous avons créé un tableau avec la plateforme Trello qui est visible ci-dessous :



Nous travaillons également sur Github, plateforme de développement collaboratif :

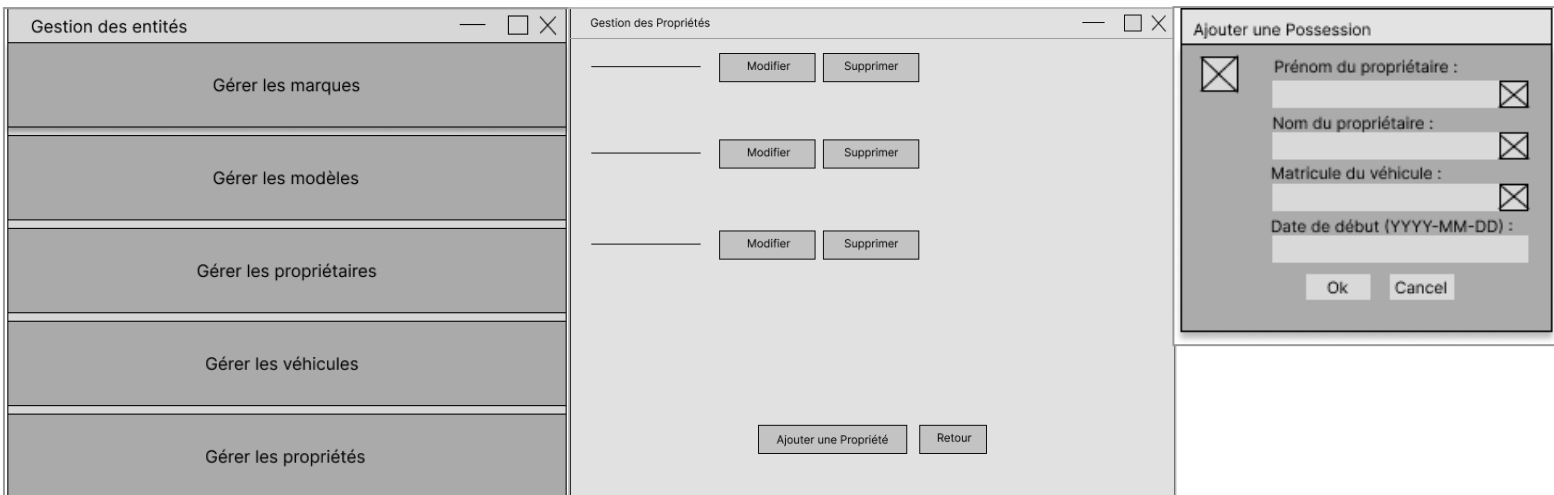


V. Conceptions du projet

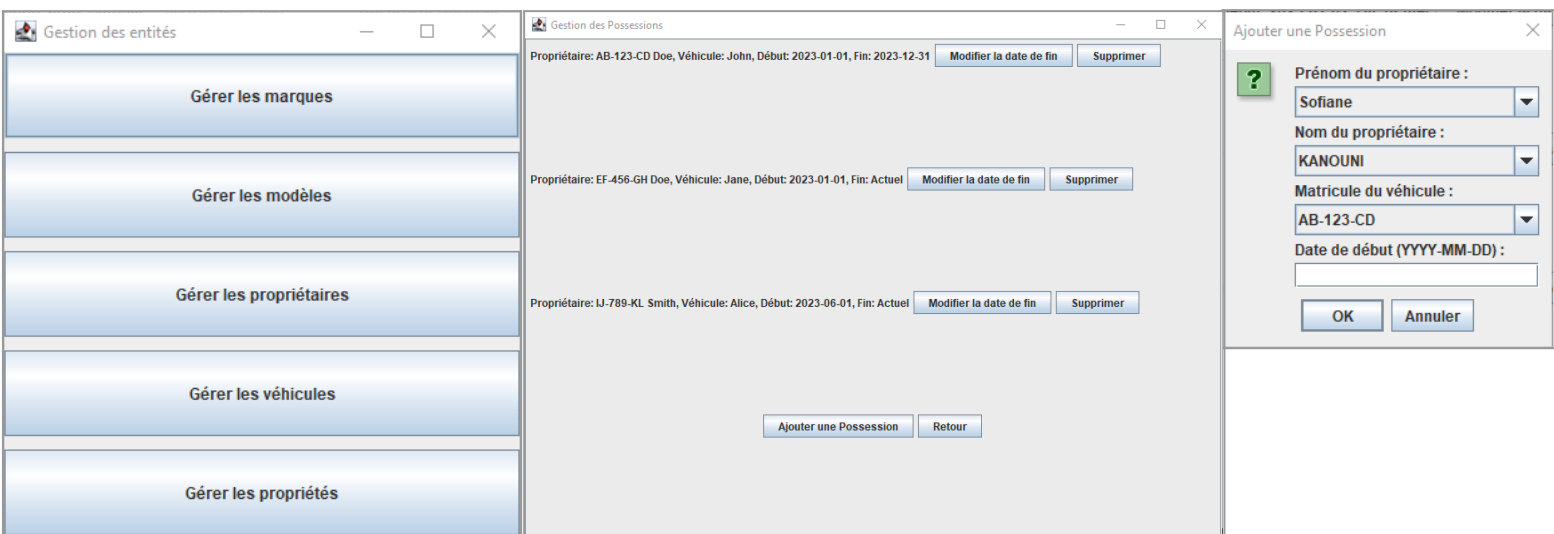
1. Le front-end

● Wireframes

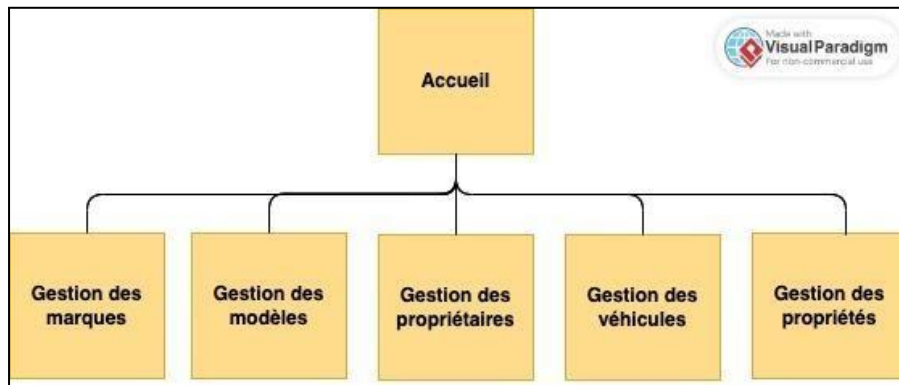
Les Wireframes réalisé pour ce projet avec Figma, sont les suivants :



● Maquettes

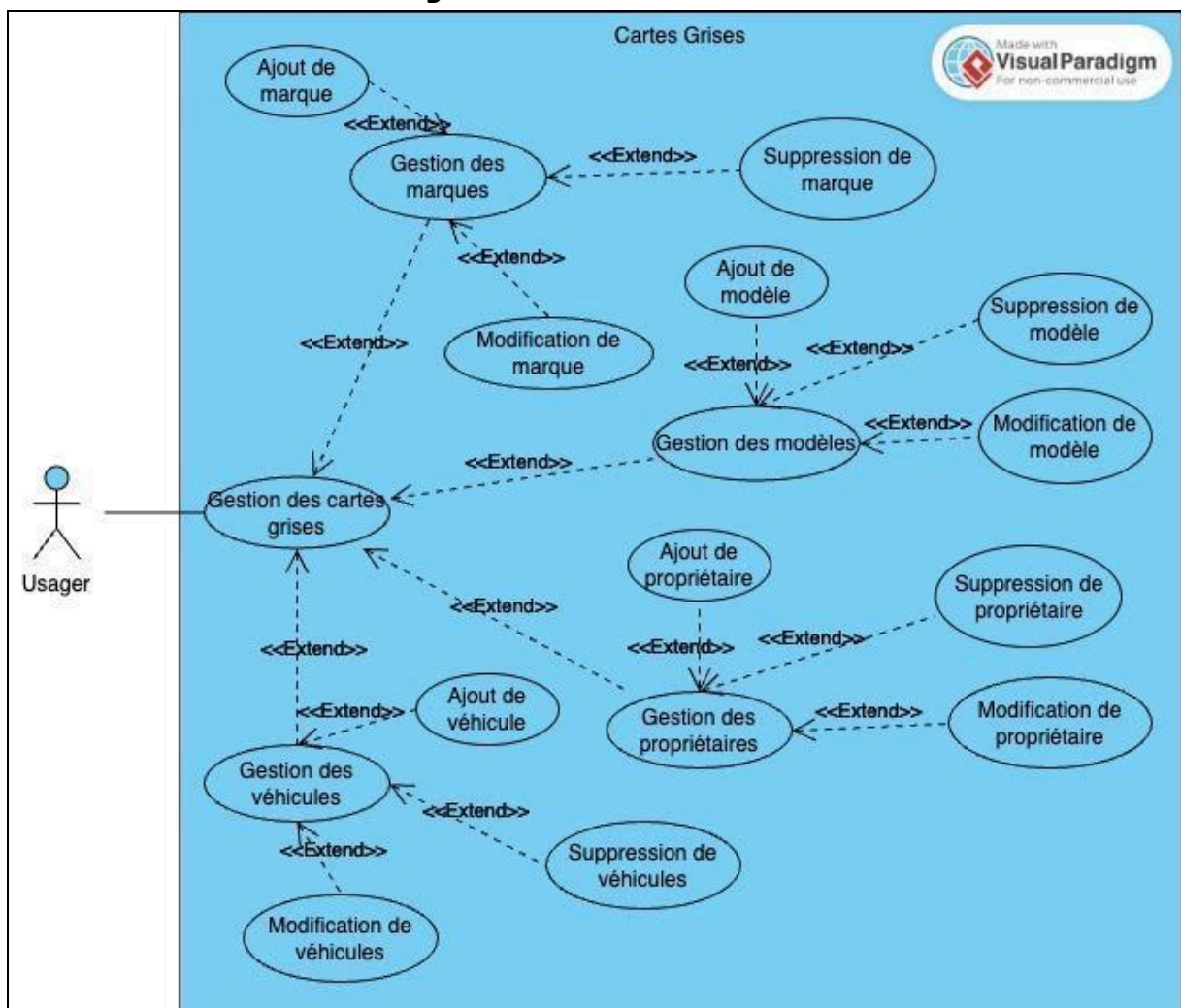


- Arborescences

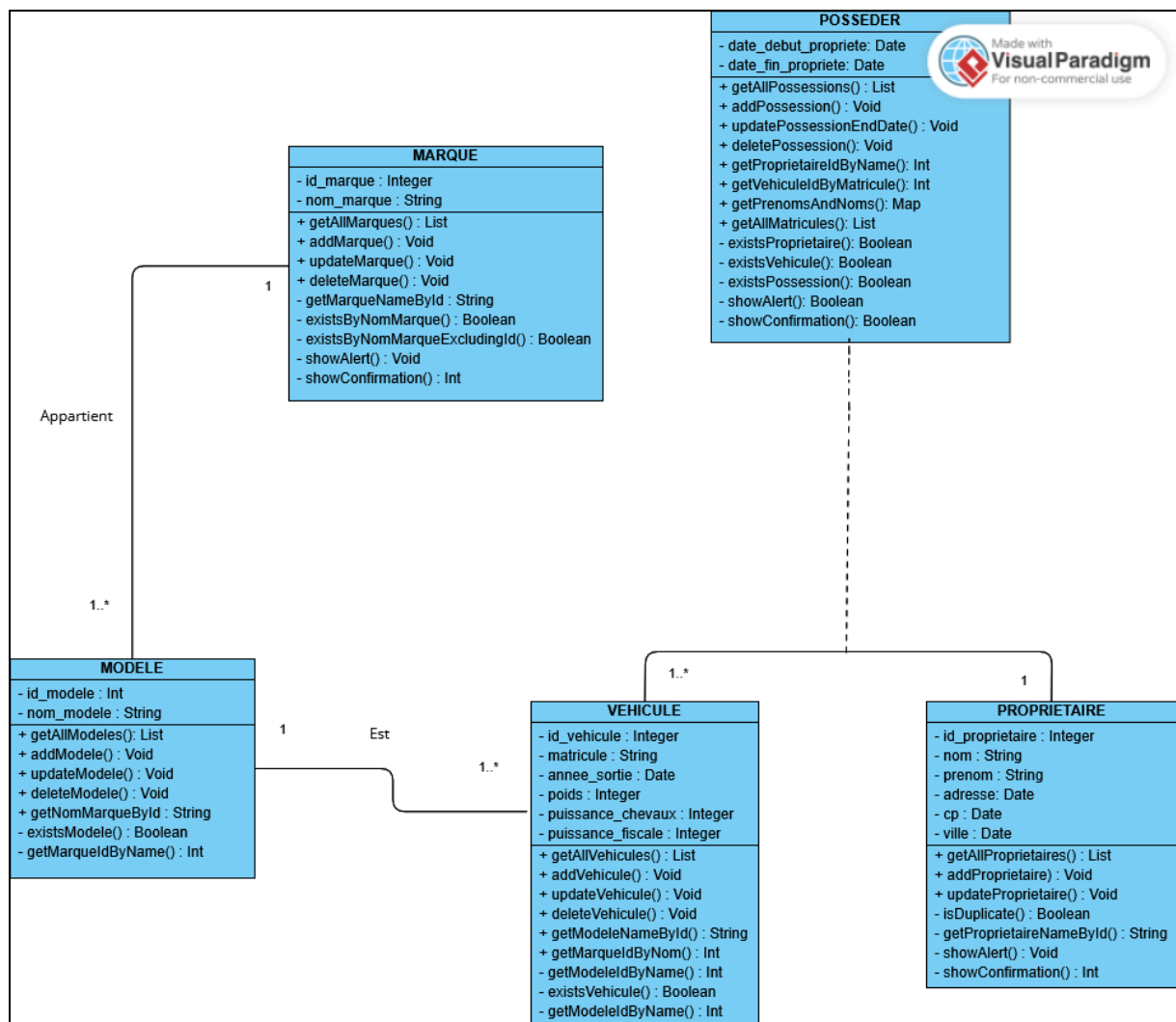


2. Le back-end

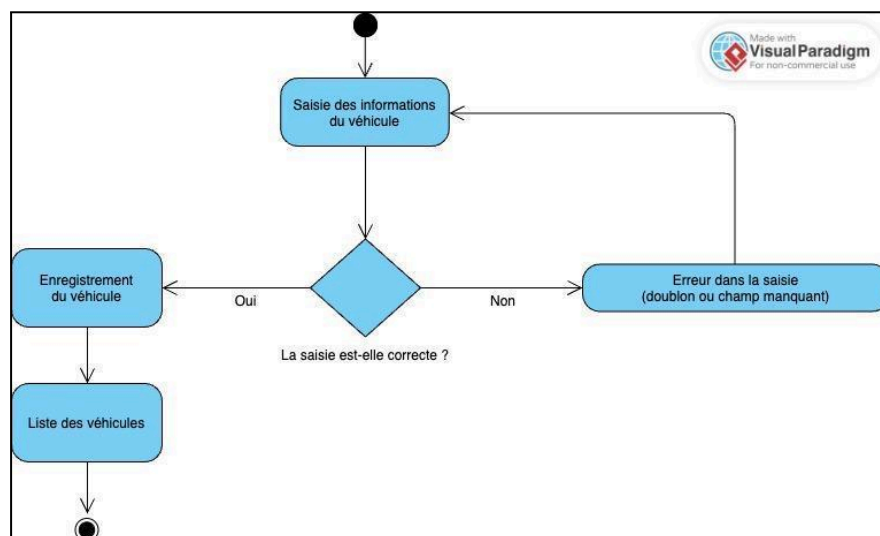
- Diagramme de cas d'utilisation



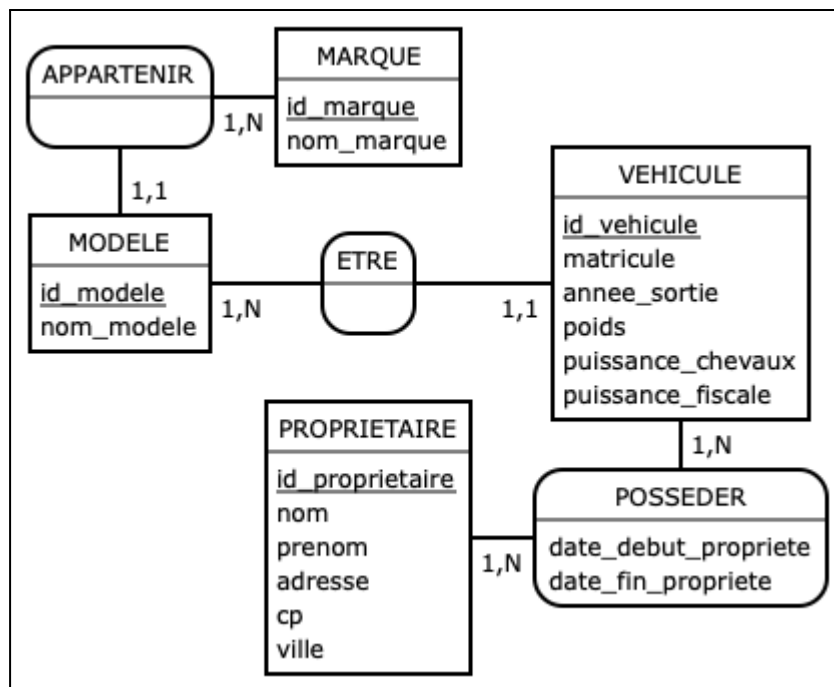
▪ Diagramme de classe carte grise



• Diagramme d'activité de l'ajout d'un véhicule



- **Modèles Conceptuel de Données (MCD)**



- **Modèle logique de Données (MLD)**

MARQUE (id_marque, nom_marque)

Clé primaire : id_marque

MODELE (id_modele, nom_modele, id_marque)

Clé primaire : id_modele

Clé étrangère : id_marque en référence à id_marque de MARQUE

VEHICULE (id_vehicule, matricule, annee_sortie, poids, puissance_chevaux, puissance_fiscale, id_modele)

Clé primaire : id_vehicule

Clé étrangère : id_modele en référence à id_modele de MODELE

POSSEDER (id_proprietaire, id_vehicule, date_debut_propriete, date_fin_propriete)

Clé primaire : id_proprietaire, id_vehicule

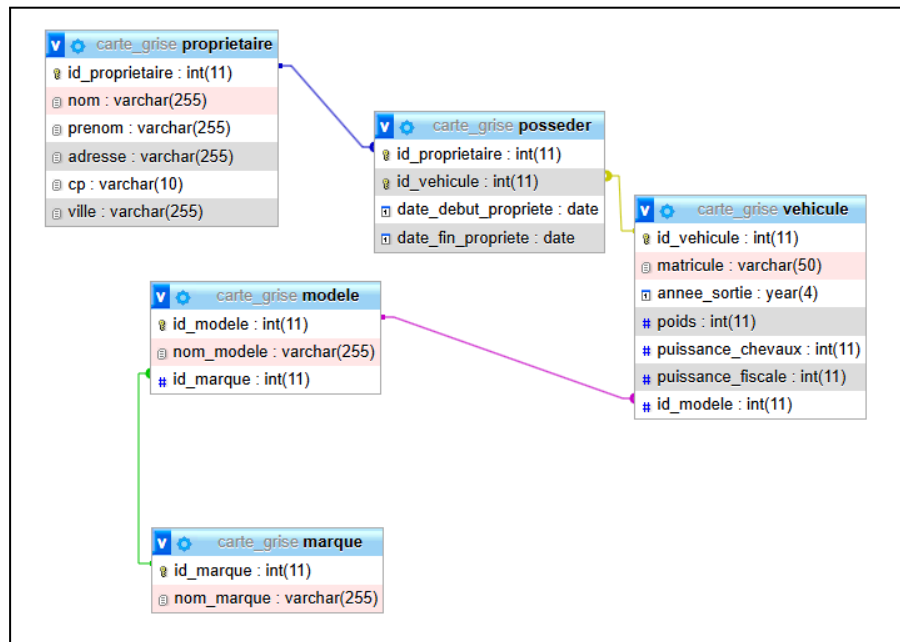
Clé étrangère : id_proprietaire en référence à id_proprietaire de PROPRIETAIRE

id_vehicule en référence à id_vehicule de VEHICULE

PROPRIETAIRE (id_proprietaire, nom, prenom, adresse, cp, ville)

Clé primaire : id_proprietaire

• Modèle Physique de Données (MPD)



VI. Technologies utilisées

3. Langages de programmation applicatif

- Java

4. Base de données

- Serveur web : Apache (contenu dans MAMP)
- Système de base de données relationnel : MySQL (contenu dans MAMP)
- Langage de base de donnée SQL

VII. Sécurité

6. Protection contre les injections SQL

```

try (Connection conn = DatabaseConnection.getConnection();
    PreparedStatement ps = conn.prepareStatement(
        sql:"INSERT INTO posseder (id_proprietaire, id_vehicule, date_debut_propriete) VALUES (?, ?, ?)")) {

    ps.setInt(parameterIndex:1, idProprietaire);
    ps.setInt(parameterIndex:2, idVehicule);
    ps.setDate(parameterIndex:3, dateDebut);
    ps.executeUpdate();
} catch (SQLException e) {
    e.printStackTrace();
    showAlert(title:"Erreur", message:"Une erreur s'est produite lors de l'ajout de la propriété.");
}
  
```