

# PROJET

## Carte grise



DOSSIER REDIGER PAR:  
[KOPOIN B  linda]

Date: 05/12/2024

Email: belkopoin@gmail.com

lien Github:  
<https://github.com/kokobelly29/Carte-grise-kopoin-Belinda>

# SOMMAIRE

1. Contexte
  - 1.1. Les cartes grises: présentation
  - 1.2. Objectifs
  - 1.3. Date de rendu du projet
2. Besoins fonctionnels
3. Ressources nécessaires à la réalisation du projet
  - 3.1. Ressources logiciels
  - 3.2. Ressources matérielles
  - 3.3. Languages nécessaires
4. Gestion du projet
5. Conception du projet
  - 5.1. Le front-end
    - 5.1.1. Wireframes
    - 5.1.2. Maquettes
    - 5.1.3. Arborescence (sitemap)
  - 5.2. Le back-end
    - 5.2.1. Diagramme de cas d'utilisation
    - 5.2.2. Diagramme d'activités
    - 5.2.3. Modèles Conceptuel de Données (MCD)
    - 5.2.4. Modèle Logique de Données (MLD)
    - 5.2.5. Modèle Physique de Données (MPD)
6. Technologies utilisées
  - 6.1. Languages de développement Web
  - 6.2. Base de données
7. Sécurité
  - 7.1. Protection contre les attaques XSS (Cross-Site Scripting)
  - 7.2. Protection contre les injections SQL

# 1. Contexte

## 1.1. Les cartes grises : présentation

Une carte grise est un document permettant d'identifier un véhicule et également de donner différentes informations concernant son propriétaire, mais aussi les caractéristiques du véhicule. La carte grise est en définition une carte d'identité de véhicule. (source: [autodemarches.fr](https://www.autodemarches.fr))

## 1.2. Objectifs

Vous occupez actuellement le poste de concepteur et développeur au sein de la Direction des systèmes d'information de la préfecture de votre département. La responsable du service des cartes grises souhaiterait faire évoluer leur application métiers. Cependant, aucun document de conception n'est disponible. Votre travail consiste donc à travailler sur l'élaboration de documents de conception de l'application actuelle en vue de faciliter la réflexion autour de son évolution.

## 1.3. Date de rendu du projet

Le projet sera rendu au plus tard le 05/12/2024.

# 2. 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

# 3. Ressources nécessaires à la réalisation du projet

### 3.1. Ressources logicielles

Les plateformes et logiciels nécessaires au développement de l'application métier sont:

- > l'IDE (environnement de développement ) Visual Studio Code
- > GitHub (plateforme de développement collaboratif)
- > Apache ( serveur web contenu dans MAMP )
- > MySQL (base de donnée relationnelle / Système de gestion de la base de données relationnelle contenue dans MAMP).
- > Trello (outils de gestion de projet )
- > Visual paradigm online (outils de conception UML et arborescence)
- > Figma (plateforme de maquettage )
- > Mocodo ( Conception bdd )

j'utiliserais des sources documentaires tels que, <https://tinyurl.com/bdf8kasb>

### 3.2. Ressources matérielles

Les matériaux nécessaires à la réalisation de la mission sont un ordinateur fixe et portable (écran, souris, unité centrale). Ces ordinateurs seront reliés à Internet par câble ou Wi-Fi.

### 3.3. Langues nécessaires

Pour la réalisation de l'application métier, nous emploierons différents langages tels que: Java et sql

Java est un langage de programmation polyvalent et hautement transférable, utilisé pour développer une large gamme d'applications sur différentes plateformes et types d'appareils. Ainsi dans le cadre de l'évolution de l'application métiers du service des cartes grises, ce langage serait parfaitement adaptées.

Pour stocker, gérer ou récupérer des informations en rapport avec les cartes grises; ils nous sera impératif d'avoir une base de données. Elle contiendra :

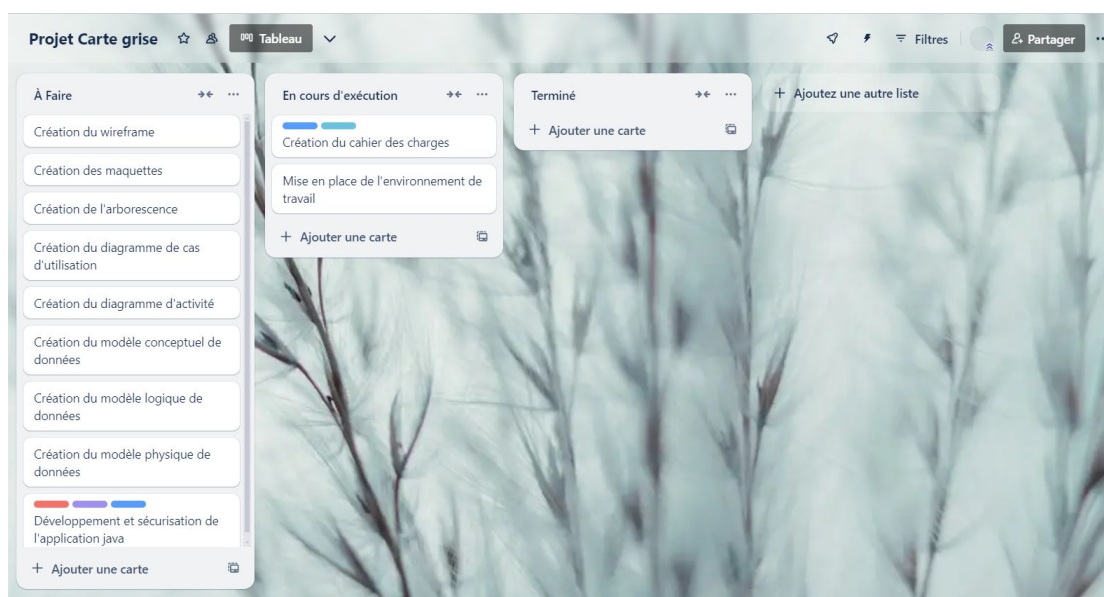
- > les données des propriétaires
- > les données des véhicules
- > les données sur les propriétés
- > les données sur les marques
- > les données sur les modèles

### 3.4. Sources documentaires

Pour la réalisation de la mission, nous utiliserons des sources documentaires tels que le site de java.

## 4. Gestion du projet

Pour réaliser le projet, nous utiliserons la méthode Agile Kanban. Afin de pouvoir se répartir le volume de travail plus facilement. Nous utiliserons comme outil de gestion de projet en ligne Trello.



Nous travaillerons également sur GitHub, une plateforme de développement collaboratif.

## 5. Conception du projet

Pour mener à bout ce projet, nous sommes passées par ces différentes étapes:

### 5.1. Le front-end

Vous trouverez les aspects visuels de l'application java avec lesquels les salariés interagiront ci-contre:

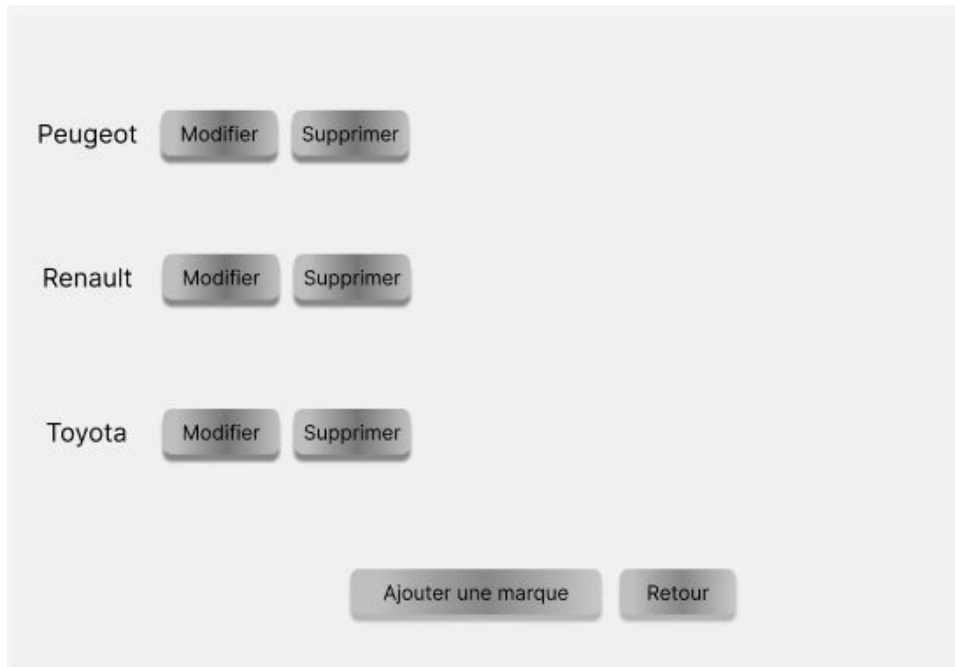
#### 5.1.1. Wireframes

Vous pouvez consulter quelques wireframes ci-dessous:

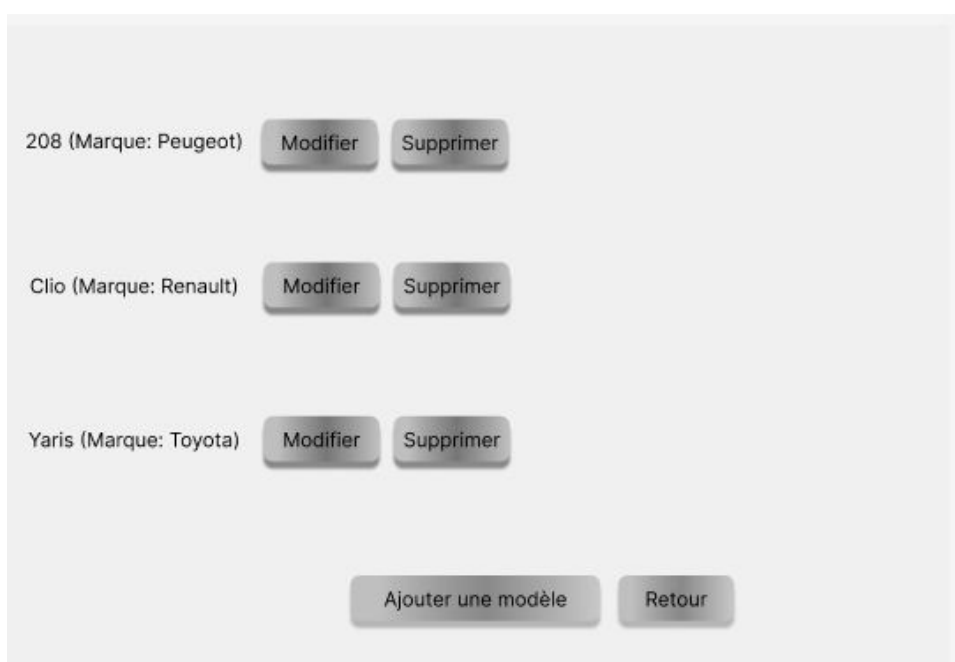
##### Wireframe carte grise – page dédié à l'accueil



## Wireframe carte grise – page dédié à la gestion des marques



## Wireframe carte grise – page dédié à la gestion des modèles de marques



## Wireframe carte grise – page dédié à la gestion des propriétaires

Doe John (123 Rue de Paris, 75000 Paris)	Modifier	Supprimer
Doe Jane (456 Avenue des Champs, 75008 Paris)	Modifier	Supprimer
Smith Alice (789 Boulevard Haussmann, 75009 Paris)	Modifier	Supprimer
Ajouter un propriétaire		Retour

## Wireframe carte grise – page dédié à la gestion des véhicules

AB-123-CD (Modèle: 208)	Modifier	Supprimer
EF-456-GH (Modèle: Clio)	Modifier	Supprimer
IJ-789-KL (Modèle: Yaris)	Modifier	Supprimer
Ajouter un véhicule		Retour



## Wireframe carte grise – page dédié à la gestion des propriétés

Wireframe of a page for managing vehicle properties. The page displays a list of properties with the following details:

- Doe John (Véhicule: B-123-CD) Date début propriété: 01/01/2023 Date fin propriété: 01/01/2023 [Modifier] [Supprimer]
- Doe Jane (Véhicule: EF-456-GH) Date début propriété: 01/01/2023 Date fin propriété: ..... [Modifier] [Supprimer]
- Smith Alice (Véhicule: IJ-789-KL) Date début propriété: 01/06/2023 Date fin propriété: ..... [Modifier] [Supprimer]

At the bottom of the page, there are two buttons: [Ajouter une propriété] and [Retour].

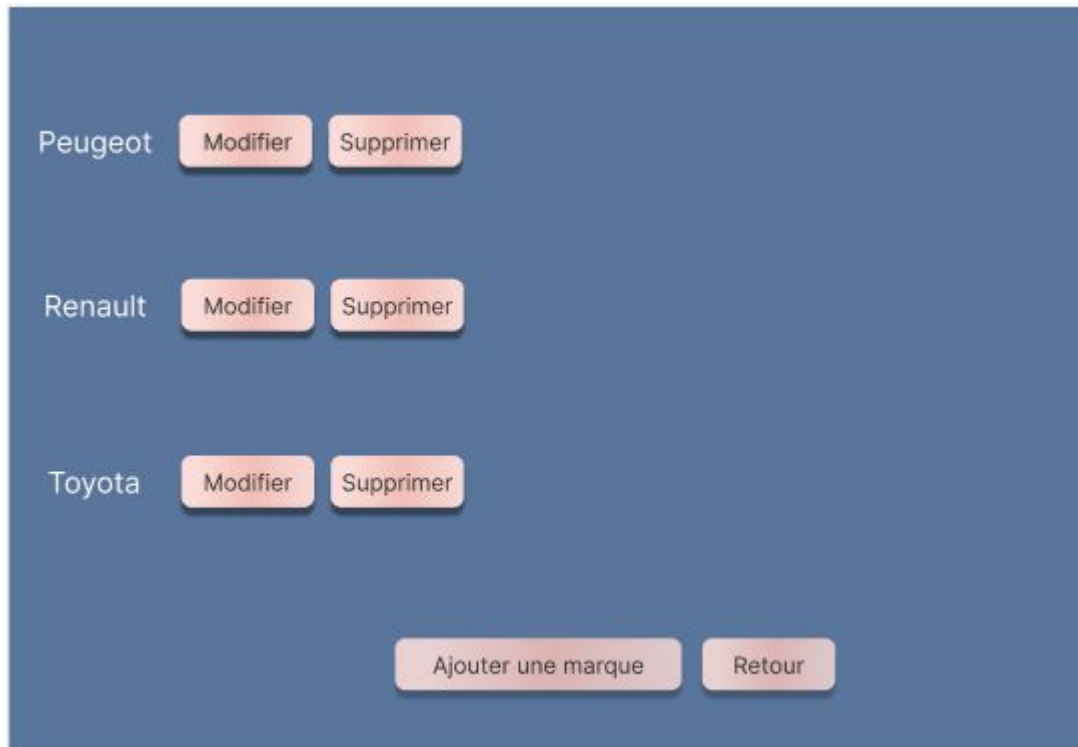
### 5.1.2. Maquettes

## Maquette carte grise – page dédié à l'accueil

Maquette of a home page with five menu items:

- Gérer les marques
- Gérer les modèles
- Gérer les propriétaires
- Gérer les véhicules
- Gérer les propriétés

### Maquette carte grise – page dédié à la gestion des marques



### Maquette carte grise – page dédié à la gestion des modèles de marques



## Maquette carte grise – page dédié à la gestion des propriétaires




Maquette de la page de gestion des propriétaires. L'interface est présentée sur un fond bleu foncé. Elle contient trois entrées de propriétaires, chacune avec un nom et une adresse, suivies de boutons 'Modifier' et 'Supprimer'. En bas, il y a deux boutons : 'Ajouter un propriétaire' et 'Retour'.

Doe John (123 Rue de Paris, 75000 Paris)	Modifier	Supprimer
Doe Jane (456 Avenue des Champs, 75008 Paris)	Modifier	Supprimer
Smith Alice (789 Boulevard Haussmann, 75009 Paris)	Modifier	Supprimer

Ajouter un propriétaire   Retour

## Maquette carte grise – page dédié à la gestion des propriétaires



Maquette de la page de gestion des véhicules. L'interface est présentée sur un fond bleu foncé. Elle contient trois entrées de véhicules, chacune avec un numéro d'immatriculation et un modèle, suivies de boutons 'Modifier' et 'Supprimer'. En bas, il y a deux boutons : 'Ajouter un véhicule' et 'Retour'.

AB-123-CD (Modèle: 208)	Modifier	Supprimer
EF-456-GH (Modèle: Clio)	Modifier	Supprimer
IJ-789-KL (Modèle: Yaris)	Modifier	Supprimer

Ajouter un véhicule   Retour

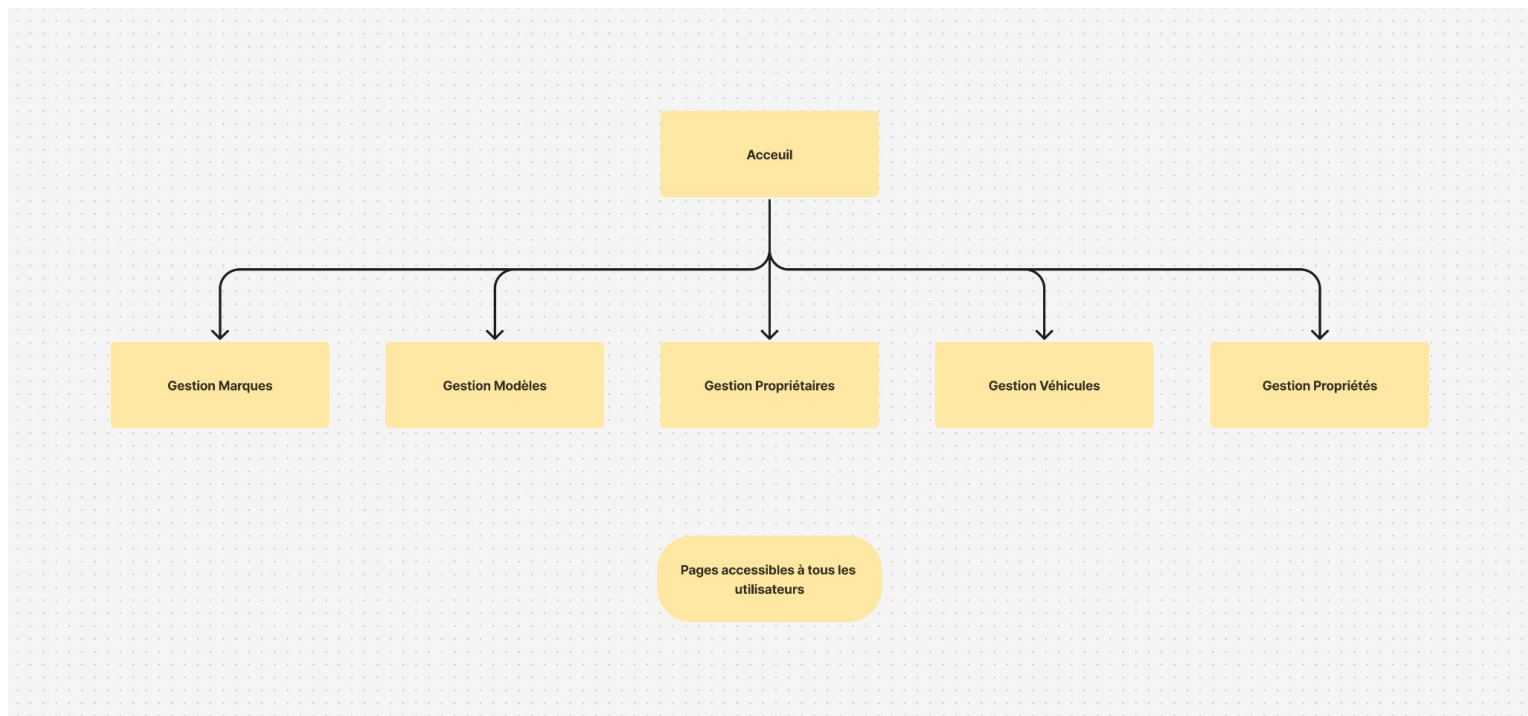
## Maquette carte grise – page dédié à la gestion des propriétaires

Maquette d'interface pour la gestion des propriétaires. Elle présente une liste de trois propriétaires avec leurs informations personnelles et de leurs véhicules, accompagnées de boutons 'Modifier' et 'Supprimer'. En bas, il y a des boutons 'Ajouter une propriété' et 'Retour'.

Propriétaire	Véhicule	Date début propriété	Date fin propriété	Modifier	Supprimer
Doe John	B-123-CD	01/01/2023	01/01/2023		
Doe Jane	EF-456-GHI	01/01/2023	.....		
Smith Alice	IJ-789-KLM	01/06/2023	.....		

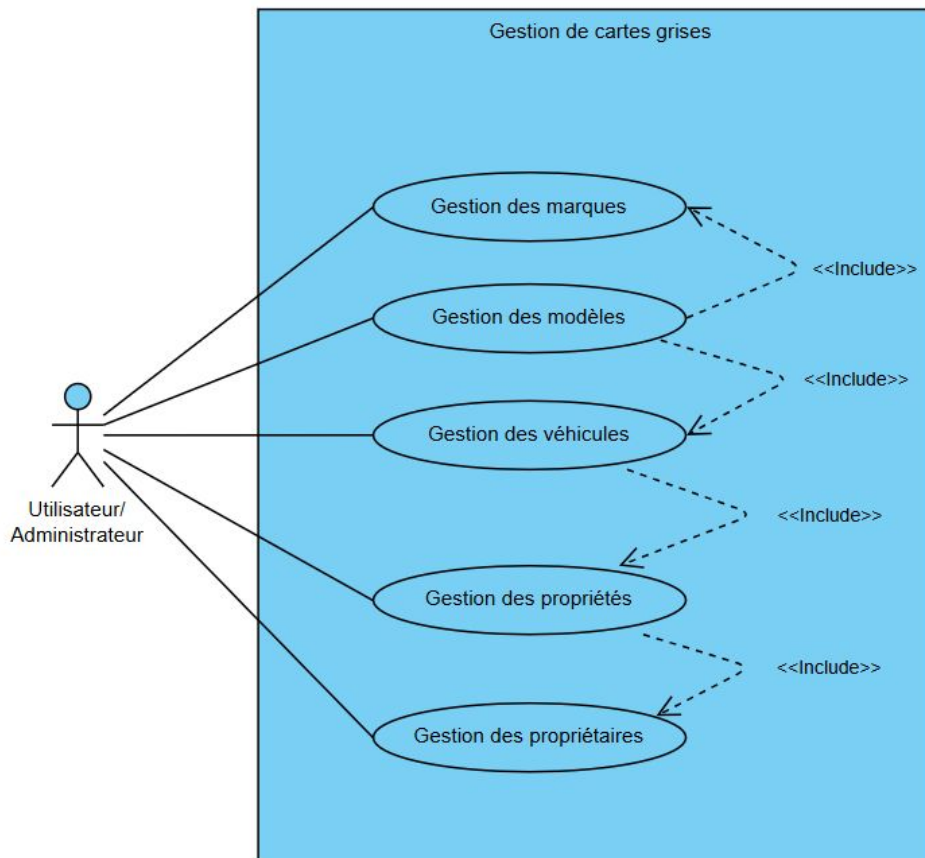
Ajouter une propriété    Retour

### 5.1.3. Arborescence (sitemap)



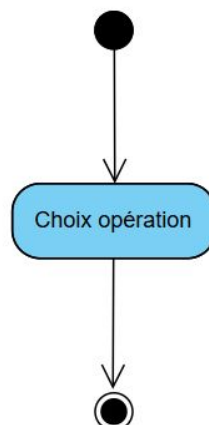
## 5.2. Le back-end

### 5.2.1. Diagramme de cas d'utilisation



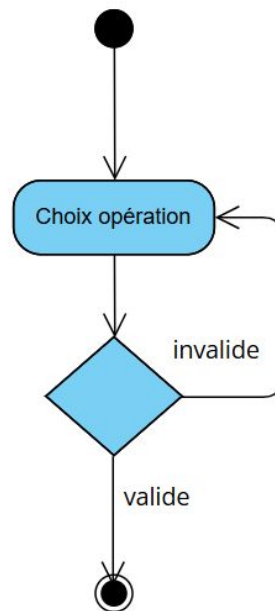
### 5.2.2. Diagramme d'activité

Diagramme d'activité pour la page d'accueil

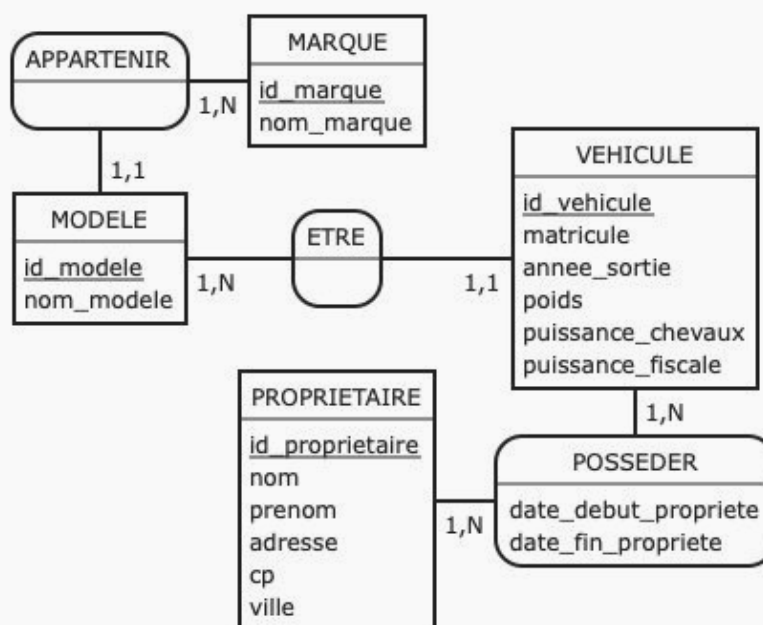


### 5.2.2. Diagramme d'activité

Diagramme d'activité pour la page gestion marques, modèles, véhicules, propriétaires, propriétés



### 5.2.3. Modèles Conceptuel de Données (MCD)



#### 5.2.4. Modèle Logique de Données (MLD)

**MARQUE** ( id\_marque, nom\_marque )

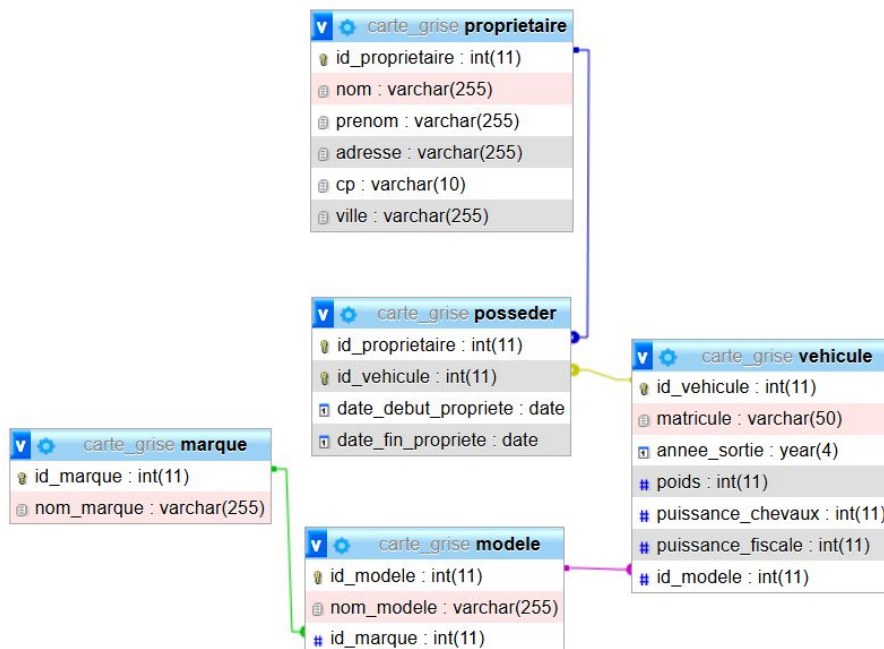
**MODELE** ( id\_modele, nom\_modele, #id\_marque )

**POSSEDER** ( #id\_vehicule, #id\_proprietaire, date\_debut\_propriete, date\_fin\_propriete )

**PROPRIETAIRE** ( id\_proprietaire, nom, prenom, adresse, cp, ville )

**VEHICULE** ( id\_vehicule, matricule, annee\_sortie, poids, puissance\_chevaux, puissance\_fiscale, #id\_modele )

#### 5.2.5. Modèle Physique de Données (MPD)



## 6. Sécurité

### 7.1. Protection contre les attaques XSS (Cross-Site Scripting)

Cross-site scripting (XSS) est une faille de sécurité qui permet à un attaquant d'injecter dans un site web un code client malveillant. Ce code est exécuté par les victimes et permet aux attaquants de contourner les contrôles d'accès et d'usurper l'identité des utilisateurs (source MDN Web Docs). La prévention XSS les plus courants et les plus utiles sont :

> Le filtrage :

Ceci implique de vérifier que les données d'entrée correspondent à ce qui est attendu (texte brut, chiffres, adresses e-mail, etc.).

> L'échappement :

Ceci consiste à transformer les caractères spéciaux en équivalents HTML ou JavaScript sûrs (exemple: < devient &lt; > devient &gt; & devient &amp et " devient &quot;)

### 7.2. Protection contre les injections SQL

Une attaque injection SQL consiste à récupérer des données sensibles ou à caractère personnel stockées sur le serveur SQL. Ainsi, il sera nécessaire de rechercher les paramètres vulnérables (données caractères personnelles) pour les protéger. Pour cela nous avons décidés d'utiliser:

> *des instructions préparées*

```
<?php
```

```
require_once('config.php');
```

```
// Utiliser les informations de configuration
```

```
$dsn = "mysql:host=" . $dbConfig['host'] . ";dbname=" . $dbConfig['dbname'];
```

```
$username = $dbConfig['username'];
```

```
$password = $dbConfig['password'];
```

```
?>
```

(source definitions-digital)