Cahier des charges techniques

Sommaire

[1. Contexte du projet](#_heading=h.1d6mma2ci7xl)

[1.1. Présentation du projet](#_heading=h.wek20lqal41w)

[1.2. Date de rendu du projet](#_heading=h.hsbf2s9v0gpe)

[2. Besoins fonctionnels](#_heading=h.72adhlswac0n)

[3. Ressources nécessaires à la réalisation du projet](#_heading=h.lr47i5krzqwp)

[3.1. Ressources matérielles](#_heading=h.moe5ly27tmhr)

[3.2. Ressources logicielles](#_heading=h.cvyk6p742phf)

[4. Gestion du projet](#_heading=h.wj6e27o1xw4s)

[5. Conception du projet](#_heading=h.4jjvlmxvhcof)

[5.1. Le front-end](#_heading=h.b3pf1ig00kap)

[5.1.1. Wireframes](#_heading=h.xbdoysf41ddh)

[5.1.2. Maquettes](#_heading=h.taj1xh3jo3sr)

[5.1.3. Arborescences](#_heading=h.7cpah6l43vkf)

[5.2. Le back-end](#_heading=h.ri4n2nlk4x5y)

[5.2.1. Diagramme de cas d’utilisation](#_heading=h.4hfnlk8rv9if)

[5.2.2. Diagramme d’activités](#_heading=h.d1ay6n1m7wn9)

5.2.3 Dictionnaire de données

[5.2.4 Modèles Conceptuel de Données (MCD)](#_heading=h.mhbyfyrbmn9)

[5.2.5. Modèle Logique de Données (MLD)](#_heading=h.rr2kq3n12ss)

[5.2.6. Modèle Physique de Données (MPD)](#_heading=h.5jowy0y259aj)

[6. Technologies utilisées](#_heading=h.6j2nbpnxpppw)

[6.1. Langages de développement Web](#_heading=h.uhwlwqptkq2s)

[6.2. Base de données](#_heading=h.q8i476rr9oys)

[7. Sécurité](#_heading=h.15p9a6rgigad)

[7.1. Login et protection des pages administrateurs](#_heading=h.q04sq9cnrw82)

[7.2. Cryptage des mots de passe avec Bcrypt](#_heading=h.ct1wz6q0jyd1)

[7.3. Protection contre les attaques XSS (Cross-Site Scripting)](#_heading=h.cimahn54kest)

[7.4. Protection contre les injections SQL](#_heading=h.cg22ajiezi4v)

# 

# Contexte du projet

## 1.1. Présentation du projet

Travail en classe et à la maison à partir d’un support étudiant reprenant un contexte professionnel.

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 souhaite faire évoluer l’application métier utilisée par son service. Toutefois, aucun document de conception n’est disponible. Votre mission consiste donc à élaborer la documentation technique de l’application existante afin de faciliter la réflexion sur son évolution.

## L’objectif est de concevoir et développer un logiciel de gestion des cartes grises (ajout, modifications et suppressions d’informations). Le logiciel est développé en Java.

## 1.2. Date de rendu du projet

Le projet doit être rendu au plus tard le 6 mars 2025.

# Besoins fonctionnels

Les données seront stockées dans une base de données relationnelle pour faciliter la gestion et la mise à jour des informations.

# Ressources nécessaires à la réalisation du projet

## 3.1. Ressources matérielles

Pc Portable ou fixe (unité centrale, souris, clavier). Ils doivent être reliés à internet.

## 3.2. Ressources logicielles

Environnement de développement : Visual code

Plateforme développement collaborative : GitHub

Serveur web Apache, Système de Base de Donnée Relationnelle MySQL (contenu dans MAMP)

Outil de Gestion de Projet : Trello

Conception UML Arborescence : Visual Paradigme

Maquette : Figma

Conception Base de données : MoCoDo online.

# Gestion du projet

Pour réaliser le projet, nous utiliserons la méthode Agile Kanban. Nous utiliserons également l’outil de gestion de projet en ligne Trello.

**Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Le contenu généré par l’IA peut être incorrect.**

Nous travaillons également sur GitHub, plateforme de développement collaboratif.

# Conception du projet

## 5.1. Le front-end

### 5.1.1. Wireframes

Wireframe page d’accueil

Une image contenant texte, capture d’écran, logiciel

Le contenu généré par l’IA peut être incorrect.

Wireframe liste des marques

Une image contenant texte, capture d’écran, affichage, logiciel

Le contenu généré par l’IA peut être incorrect.

Wireframe gérer les propriétés

Une image contenant capture d’écran, texte, ligne, nombre

Le contenu généré par l’IA peut être incorrect.

### 5.1.2. Maquettes

Une image contenant texte, capture d’écran, affichage, logiciel

Le contenu généré par l’IA peut être incorrect.

Une image contenant texte, capture d’écran, affichage, logiciel

Le contenu généré par l’IA peut être incorrect.

Une image contenant texte, capture d’écran, affichage, nombre

Le contenu généré par l’IA peut être incorrect.

### 5.1.3. Arborescences

Une image contenant texte, capture d’écran, Police, conception

Le contenu généré par l’IA peut être incorrect.

## 5.2. Le back-end

### 5.2.1. Diagramme de cas d’utilisation

Une image contenant texte, capture d’écran, diagramme, Police

Le contenu généré par l’IA peut être incorrect.

### 5.2.2. Diagramme d’activités

Une image contenant texte, capture d’écran, diagramme, ligne

Le contenu généré par l’IA peut être incorrect.

DIAGRAMME DE CLASSE

Une image contenant texte, capture d’écran, diagramme, Police

Le contenu généré par l’IA peut être incorrect.

5.2.3. Dictionnaire de données

| **Nom de l'attribut** | **Type** | | **Contraintes** | **Description** | |
| --- | --- | --- | --- | --- | --- |
| id\_marque | INT | | PRIMARY KEY, AUTO\_INCREMENT | Identifiant unique de la marque | |
| nom\_marque | VARCHAR(100) | | NOT NULL | Nom de la marque | |
| id\_modele | | INT | PRIMARY KEY, AUTO\_INCREMENT | | Identifiant unique du modèle |
| nom\_modele | | VARCHAR(100) | NOT NULL | | Nom du modèle |
| id\_marque | | INT | FOREIGN KEY → Marque(id\_marque) | | Référence à la marque |
| id\_vehicule | | INT | PRIMARY KEY, AUTO\_INCREMENT | | Identifiant unique du véhicule |
| matricule | | VARCHAR(20) | NOT NULL, UNIQUE | | Numéro d'immatriculation |
| annee\_sortie | | YEAR | NOT NULL | | Année de mise en circulation |
| poids | | DECIMAL(6,2) | NOT NULL | | Poids du véhicule (en kg) |
| puissance\_chevaux | | INT | NOT NULL | | Puissance en chevaux |
| puissance\_fiscale | | INT | NOT NULL | | Puissance fiscale |
| id\_modele | | INT | FOREIGN KEY → Modèle(id\_modele) | | Référence au modèle du véhicule |
| id\_proprietaire | | INT | PRIMARY KEY, AUTO\_INCREMENT | | Identifiant unique du propriétaire |
| nom | | VARCHAR(100) | NOT NULL | | Nom du propriétaire |
| prenom | | VARCHAR(100) | NOT NULL | | Prénom du propriétaire |
| adresse | | TEXT | NOT NULL | | Adresse postale |
| cp | | VARCHAR(10) | NOT NULL | | Code postal |
| ville | | VARCHAR(100) | NOT NULL | | Ville |
| id\_vehicule | | INT | PRIMARY KEY (avec id\_proprietaire), FOREIGN KEY → Véhicule(id\_vehicule) | | Référence au véhicule |
| id\_proprietaire | | INT | PRIMARY KEY (avec id\_vehicule), FOREIGN KEY → Propriétaire(id\_proprietaire) | | Référence au propriétaire |
| date\_debut\_propriete | | DATE | NOT NULL | | Date de début de possession |
| date\_fin\_propriete | | DATE | NULL (si toujours propriétaire) | | Date de fin de possession |

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

Une image contenant texte, capture d’écran, diagramme, Police

Le contenu généré par l’IA peut être incorrect.

### 5.2.5. 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 reference 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 reference à id\_modele de MODELE

**Proprietaire** (id\_ proprietaire, nom, prenom, adresse, cp, ville)

Clé primaire : id\_proprietaire

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

Clé primaire : id\_proprietaire en reference à id\_ proprietaire de PROPRIETAIRE

id\_vehicule en reference à id\_vehicule de VEHICULE

Clé étrangère : id\_ proprietaire en reference à id\_ proprietaire de PROPRIETAIRE

id\_vehicule en reference à id\_vehicule de VEHICULE

### 

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

Une image contenant texte, capture d’écran, Police, diagramme

Le contenu généré par l’IA peut être incorrect.

### 

# Technologies utilisées

## 6.1. Langages de développement Web

JAVA

## 6.2. Base de données

Serveur Web : Apache contenu dans MAMP

Système de base de données : SQL

Langage de base de données : SQL

# Sécurité

## 7.1. Protections utilisées

## • Utilisation des requêtes préparées (PreparedStatement) afin d’empêcher l’exécution de commandes SQL malveillantes.

## • Vérification des doublons afin de garantir l’intégrité des données en évitant les doublons.

## 7.2. Protection contre les injections SQL

## Une injection SQL, parfois abrégée en SQLi, est un type de vulnérabilité dans lequel un pirate utilise un morceau de code SQL (« Structured Query Language », langage de requête structuré) pour manipuler une base de données et accéder à des informations potentiellement importantes. C'est l'un des types d'attaques les plus répandus et menaçants, car il peut potentiellement être utilisé pour nuire à n'importe quelle application Web ou n'importe quel site Web qui utilise une base de données SQL (soit la plupart).

## Pour se protéger des injections SQL, nous allons donc utiliser les requêtes préparées **PreparedStatement** , **try-catch**.