

# **RAPPORT DE PROJET SUPERCAR**

## **Sommaire :**

- **Introduction :**
  - Présentation (entreprise supercar et multisys)
  - Démarche du projet (objectif, besoins (parties admin et client), missions)
  - Calendrier de réalisation (Trello et excel)
- **Analyse du projet :**
  - Analyse des besoins
  - Arborescence
- **Conception :**
  - MCD
  - Fonctionnel
- **Développement :**
  - Outils (langue, Framework et autres...)
  - Programmation (capture d'écran)
  - Test
- **Déploiement et maintenance :**
  - Déploiement : en local
  - Maintenance : Utilité du module admin
- **Conclusion**
- **Annexe**

## Introduction

- **Présentation**

Supercar est une société spécialisée dans la vente de voitures neuves de plusieurs marques depuis 2009. Elles importent des voitures de différents pays tel que le Japon, la Singapour, l'Afrique du Sud, la Chine, les États-Unis, l'Allemagne et autre. Ainsi, mon équipe et moi nous sommes multisys, prestataires en informatique, qui a pour objectifs de créer un site web de voiture dans lequel on gère la partie client et admins grâce avec le serveur Apache et MySQL, avec comme langage du HTML, du PHP.

- **Démarche du projet**

La section client de notre site web est dédiée à fournir aux visiteurs des informations détaillées sur les véhicules.

Lorsque vous accédez au site, vous êtes accueilli par une page d'accueil comprenant deux textes explicatifs sur notre entreprise et nos services, accompagnés de deux images illustratives. Nous proposons des descriptions approfondies des véhicules, incluant leur tarification, leur vitesse, leur année de fabrication, ainsi que des détails sur leur extérieur et leur intérieur, accompagnés d'images pour les visualiser. En outre, nous présentons les services complémentaires offerts par Supercar, tels que le lavage, la maintenance et les réparations. De plus, nous avons mis en place un système permettant aux clients de demander des essais de conduite pour les véhicules qui les intéressent. Si le client est intéressé pour une demande d'essai, le client doit s'inscrire obligatoirement de même pour le service de l'entreprise. De plus, si le client nécessite plus d'informations par rapport aux véhicules, il est libre de nous contacter en mettant son, nom, prénom, le courriel, l'adresse, le numéro.

La partie administrative du site est conçue pour faciliter la gestion de la base de données. Les administrateurs ont la possibilité de modifier, supprimer ou ajouter des données en fonction des besoins de l'entreprise. Et la gestion de base de données est accessible à la page d'accueil, les pages voitures, les services, les demandes d'essais, l'inscription et contact. Cette section garantit une gestion efficace et précise de toutes les informations liées aux véhicules et aux interactions avec les clients.

Le but de projet est de vendre des voitures, en proposant des informations concrètes, mais aussi gérer la base de données. De prodiguer des conseils aux clients, de faire connaître les voitures de notre entreprise, et de pouvoir administrer les différents contenus du site qui peuvent mettre à jour les contenus dynamiques des pages web.

- **Calendrier de réalisation**

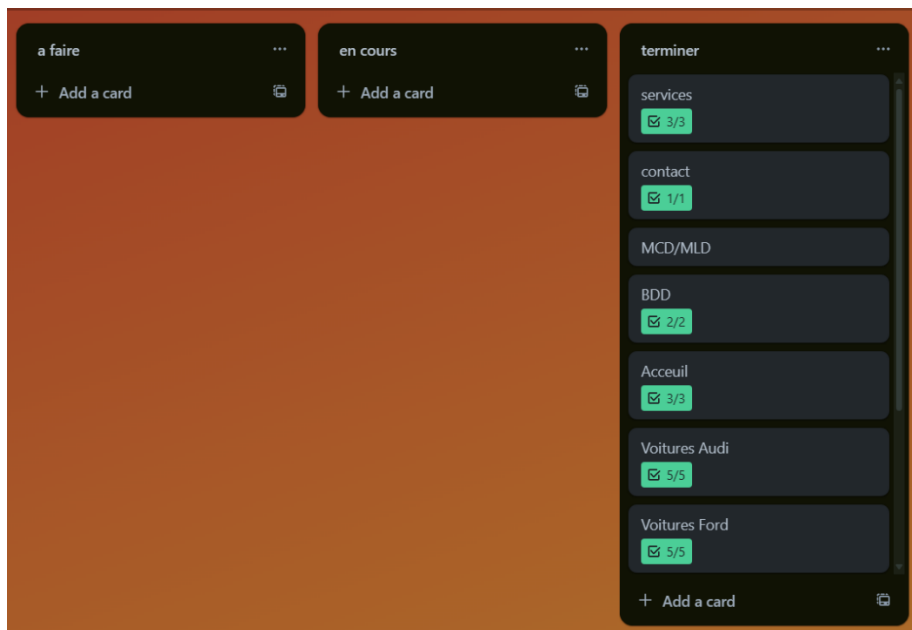
Pour mener à bien ce projet, nous avons utilisé des outils de gestion de projet tels que Trello et Excel, ce qui nous a permis de répartir efficacement les tâches entre les membres de l'équipe et de suivre l'avancement du projet. Étant chef de projet à partir de février, nous avons débuté la partie admin, une partie complètement nouvelle pour nous, mais qui a été assez facile à le

réaliser. Avant de se lancer dans la programmation, nous avons séparé les tâches, vu que la partie de admin nécessite de quatre tâches, de modification, de suppression et d'ajout, visualisation qui appartient à chaque rubrique, chaque personne ont la responsabilité de faire une rubrique.

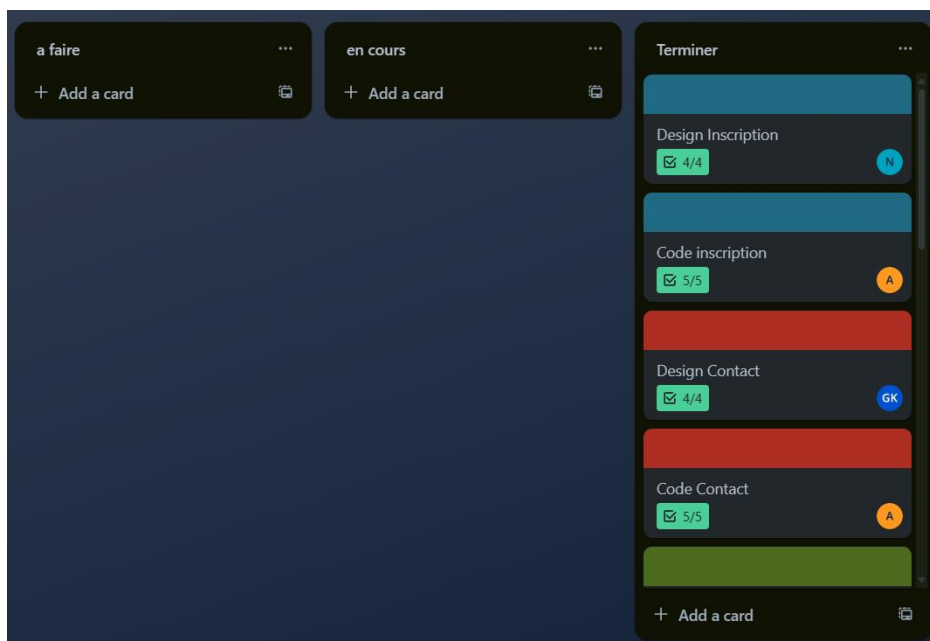
De plus avec l'aide de Trello est Excel, qu'on mettait à jour régulièrement, permet d'avoir une idée de l'avancement.

Voici un exemple de notre Trello :

- Trello de la partie client



- Trello de la partie admin



Comme vous pouvez le voir Trello organiser en trois groupes, ‘à faire’ ‘en cours’ et ‘terminer’, pour connaître l’avancement des tâches de chaque membre du groupe.

De part Trello nous avons aussi utiliser Exel

Voici notre Excel :

Tâches	Sous tâches	Détails	Personne affectée	Date de déb	Observations	Pourcentage d'avancem	Date de fin
MCD/MLD	Sur le logiciel Looping		Groupe	17-Feb	Révision de la structure du MCD (recommencer)	100%	18-Feb
Création de la BDD	Avec MySQL		Anne-Sophie	18-Feb	Création de la nouvelle BDD, facile à réaliser	100%	18-Feb
Création des tables de la BDD	Avec MySQL		Anne-Sophie		Facile à réaliser	100%	
Header	Logo		Anne-Sophie	02-Feb	Facile à créer	100%	05-Feb
	Barre de navigation		Anne-Sophie	02-Feb	Un peu de mal pour trouver le modèle et pour la	100%	05-Feb
Footer	Liens légales		Anne-Sophie	05-Feb	partie responsive mais terminer	100%	09-Feb
Maquette			Anne-Sophie	05-Feb	Facile à réaliser car le design était clair a	100%	05-Feb
Qui sommes-nous?	Texte + photo		Anne-Sophie	17-Feb	Facile à écrire	100%	17-Feb
Que faisons-nous?	Texte + photo		Anne-Sophie	17-Feb	Facile à écrire	100%	17-Feb
	Maquette	Informations sur la voiture + photos	Anne-Sophie	03-Feb		100%	03-Feb
	Audi R8	Informations sur la voiture + photos	Anne-Sophie	18-Feb		100%	
	Audi A1	Informations sur la voiture + photos	Anne-Sophie	18-Feb		100%	
	Audi Q3	Informations sur la voiture + photos	Anne-Sophie	18-Feb		100%	
Audi + maquette	Design		Anne-Sophie	21-Feb		100%	22-Feb
	Maquette		Nanjanee	05-Feb	facile à réaliser	100%	05-Feb
	Maquette	Informations sur la voiture + photos	Nanjanee	10-Feb	En cours de progression pages statique reliée à la	100%	
	Maquette	Informations sur la voiture + photos	Nanjanee	10-Feb	En cours de progression pages statiques	100%	
Ford + maquette	Maquette	Informations sur la voiture + photos	Nanjanee	09-Feb	En cours de progression Statiques reliée à la page	100%	
	Maquette		Nanjanee	19-Jan	facile à réaliser	100%	19-Jan
	Maquette	Informations sur la voiture + photos	Nanjanee			100%	
	Maquette	Informations sur la voiture + photos	Nanjanee	12-Feb	page commencé en html, mais avec le	100%	
	Maquette	Informations sur la voiture + photos	Nanjanee	12-Feb	changement vers php, j'ai pas encore débuté	100%	
Toyota + maquette	Maquette	Informations sur la voiture + photos	Nanjanee	19-Feb	page voiture presque fini , mais avec le	100%	
	Maquette		Nanjanee	19-Feb	changement ver le php, j'ai pas encore début	100%	
Maquette			Gloria	22-Jan		100%	
Formulaire	Connexion a la BDD		Gloria	05-Jan		100%	21-Feb
Numéro de téléphone			Gloria			100%	
Localisation			Gloria			100%	
Design			Gloria			100%	
Maquette			Anne-Sophie	03-Feb	Facile à réaliser	100%	03-Feb
Formulaire inscription (lien) (1 page)			Anne-Sophie	12-Feb		100%	18-Feb
Formulaire (Demande d'essai) (1 page)	Connexion a la BDD		Anne-Sophie	22-Feb		100%	22-Feb
Maquette			Annio			100%	
Formulaire inscription (lien) (1 page)			Annio			100%	
Formulaire (Services) (1 page)			Annio			100%	
Maquette			Anne-Sophie	03-Feb	Facile à réaliser	100%	03-Feb
Formulaire (1 page)	Connexion a la BDD		Anne-Sophie	12-Feb	Facile à réaliser	100%	18-Feb

## **Analyse du projet**

- **Analyse de besoins**

Mon équipe et moi avons développé un site de vente de voitures mettant en avant trois marques populaires, chacune proposant quatre modèles différents. Pour Toyota, nous offrons la Venza, la Camry et la Corolla Cross. Pour Audi, la R8, la A3 et la Q3 sont disponibles. Enfin, pour Ford, nos clients peuvent choisir parmi la Mustang, le Ford Explorer et la Focus. Nous avons sélectionné une variété de véhicules afin d'offrir à nos clients un large éventail de choix, tel que des SUV des Hatch back, des sedans et des 4x4 tout en veillant à ce que tous les modèles restent à un prix abordable.

En plus de notre sélection diversifiée, nous proposons également une gamme de services pour répondre aux besoins des clients. Nous mettons à leur disposition un formulaire d'inscription où ils peuvent choisir les services qui les intéressent tel que le lavage, la maintenance et la réparation, le client dispose de nos services s'ils ont acheté nos voitures. Notre objectif est d'être disponibles et attentifs à nos clients. Si jamais ils rencontrent des problèmes avec leur voiture, nous encourageons vivement nos clients à nous en informer afin que nous puissions prendre les mesures nécessaires pour les résoudre.

De plus, si les clients sont intéressés par une voiture de notre entreprise, on leur propose des demandes d'essai pour qu'ils puissent tester la voiture, voir si elle répond aux critères voulues de l'acheteur, de ce fait le client va acheter la voiture car elle sait comment elles fonctionnent.

- **Arborescence :**

Avant de concevoir le site, nous avons tout d'abord construit l'arborescence pour comprendre comment nous allons naviguer sur le site.

Voici ci-dessus notre arborescence :

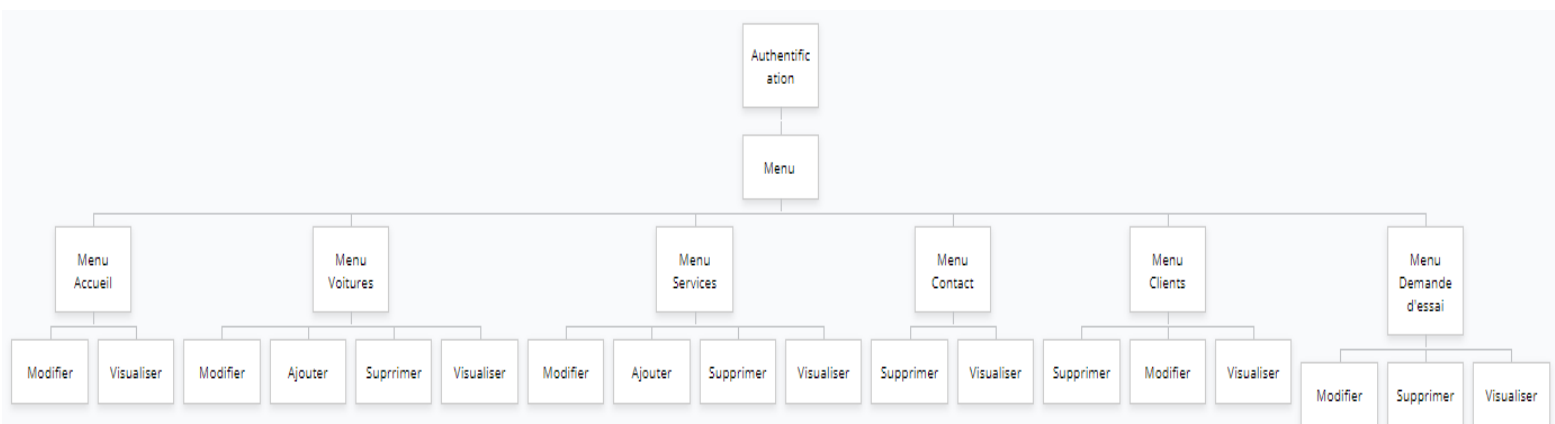
Il y a deux parties une partie cliente et une autre admin.

Voici la partie cliente :



Dans la partie client, le visiteur du site sera exposé tout d'abord par l'accueil, puis aura la possibilité de naviguer les différentes pages grâce à la barre de navigation, ainsi comme vous pouvez le voir, pour la rubrique voiture, elle donne accès à d'autre page, pour voiture il y a un dropdown qui permet de choisir parmi les modèle voulu, et lorsque vous choisissez la voiture de votre choix, vous êtes face à plusieurs modèle, et chaque modèle contient leur propre pages.

Et voici la part admin :



Pour la partie admin, l'administrateur doit obligatoirement s'authentifier avant d'accéder aux menu d'admin. Lorsque l'admin s'authentifie, il est emmené sur une page menu, qui propose plusieurs menues, tel qu'accueil, voiture, service, demande d'essai, contact, clients, demande d'essai. Et chaque menu vous avez la possibilité de choisir parmi visualiser, modifier, supprimer ou ajout.

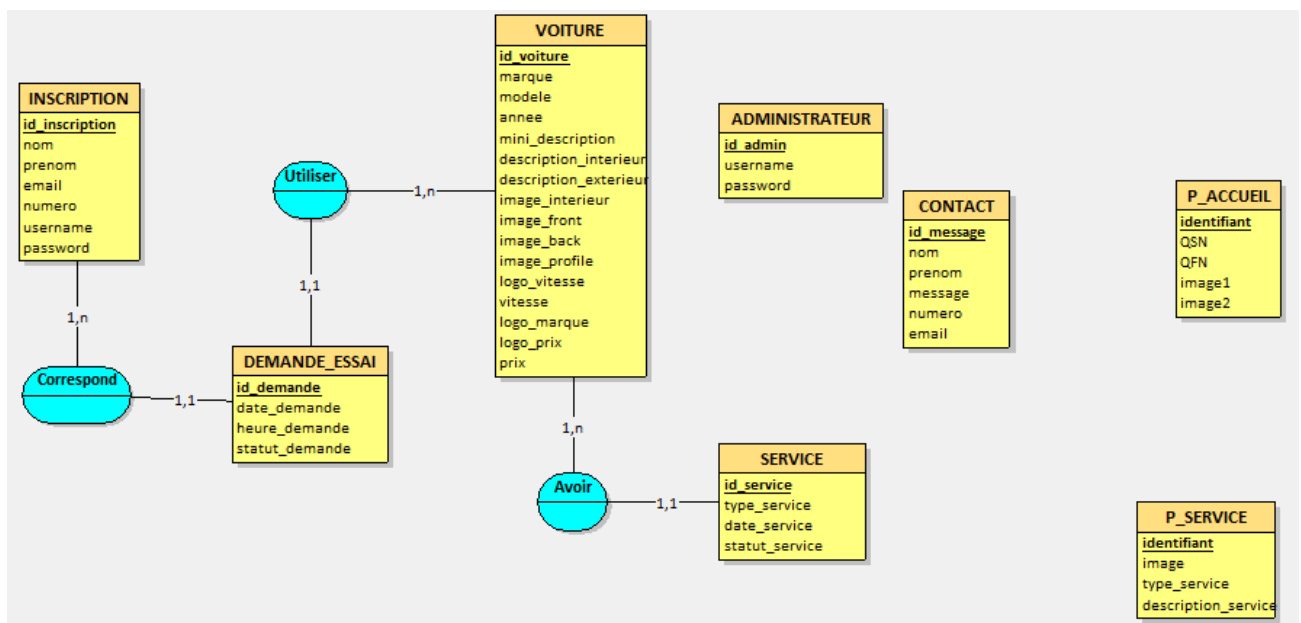
## Conception

- MCD

Avant de concevoir le site, nous avons élaboré un Modèle Conceptuel de Données (MCD) pour identifier les principales entités, leurs relations et leurs attributs. Cela nous a permis d'organiser, de structurer et de visualiser les données de manière claire. Nous avons identifié quatre entités principales : "voiture", "inscription", "demande d'essai", "service" et "contact". Chacune de ces entités possède ses propres attributs. Par exemple, pour l'entité "voiture", nous avons défini des attributs tels que la marque, le modèle, année, mini description, description intérieur, description extérieur, image intérieur, image front, image back, image profile, logo vitesse, vitesse, logo marque, logo prix, prix. Pour l'entité "inscription", nous avons attribué des attributs tels que nom, prénom, l'adresse électronique, le nom d'utilisateur et le mot de passe. Les entités "demande d'essai", date demande d'essai, heure demande et "service" ont également leurs propres attributs spécifiques tel que type de service et date de service.

Chacun des entités nommées ont tous des liens entre les autres, l'inscription correspond aux demandes d'essai, et la demande d'essai utilise un ou plusieurs voitures, et voiture a un et plusieurs services.

De plus nous avons des entités qui n'ont pas de lien avec les autres car elles ne dépendent pas des autres. Par exemple : l'entité contact qui a comme attribut, le courriel, le numéro, les messages, le nom et prénom, ainsi l'entité accueil a comme attributs le QSN (qui sommes-nous ?) et le QFN (que faisons-nous ?) et l'image 1 et l'image 2 qui sont les images associer chacun d'eux avec le texte. De plus nous avons comme entités page service avec comme attribut image, type de service et description de service. Et enfin nous avons l'entité administrateur qui a comme attributs le nom d'utilisateur et le password.



- Fonctionnel

Voici la composition de notre site web :

1. Barre de navigation :

- La barre de navigation comporte six rubriques principales : accueil, voiture, demande d'essai, service, contact et inscription.
- La rubrique voiture, il y a un drop down, qui permet aux clients de choisir et de naviguer parmi les modèles présents.

2. Page d'Accueil :

- Un jumbotron est présent en haut de la page, affichant une image de plusieurs voitures pour accueillir les visiteurs.
- Deux sections distinctes sont présentes en dessous du jumbotron : "Qui sommes-nous" et "Que faisons-nous". Chaque section est accompagnée d'une image pour mieux illustrer.

3. Page "Voiture" :

- La page "Voiture" offre une expérience de navigation fluide avec une barre de navigation permettant de choisir parmi différentes marques de voitures.
- Chaque marque redirige vers une page dédiée présentant quatre modèles de voitures.
- Pour chaque modèle de voiture, une disposition équilibrée est utilisée, affichant une image à gauche et une description détaillée à droite. Un lien est également inclus pour accéder à plus d'informations sur le modèle.

4. Page "Demande d'essai" :

- Cette page permet aux visiteurs de planifier un essai de conduite pour les voitures qui les intéressent.
- Un formulaire simple est proposé, demandant le nom d'utilisateur et le mot de passe du client, ainsi que le choix de la voiture à tester et la date souhaitée pour l'essai.

5. Page "Service" :

- Cette page présente une mise en page similaire à celle de la page d'accueil, avec une image accrocheuse en haut.
- Les trois types de services proposés (lavage, maintenance, réparation) sont présentés avec des photos pour une meilleure visualisation.
- En bas de la page, un formulaire permet aux clients inscrits de soumettre leurs demandes de service en spécifiant leur nom d'utilisateur, mot de passe, le service requis et la date souhaitée.

6. Page "Inscription" :



- Les utilisateurs peuvent s'inscrire en remplissant un formulaire dédié avec leurs informations personnelles telles que leur nom, prénom, nom d'utilisateur, adresse e-mail et mot de passe.
- Une fois inscrits, les utilisateurs ont accès aux services du site et peuvent effectuer des demandes d'essai.

7. Page "Contact" :

- Cette page propose un formulaire de contact classique où les visiteurs peuvent saisir leur nom, prénom, adresse e-mail et un message pour obtenir plus d'informations ou poser des questions spécifiques.

## **Développement**

- Outil

Le site est principalement construit en utilisant PHP, HTML et CSS, avec l'utilisation du framework Bootstrap pour assurer une mise en page responsive et esthétique. La majorité des pages sont en PHP, car cela nous permet d'effectuer des requêtes SQL directement dans notre code, simplifiant ainsi la gestion du site.

En utilisant des requêtes SQL, nous pouvons facilement récupérer les données de la base de données et les afficher directement sur les pages du site. Par exemple, sur la page des voitures, les informations telles que le modèle, le prix, la vitesse, etc., sont extraites de la base de données et affichées dynamiquement. Cela rend la gestion du contenu du site très efficace,

car toute modification, suppression ou ajout de données est immédiatement reflétée sur le site.

La structure HTML est principalement utilisée pour les formulaires et les éléments de navigation administratifs. Par exemple, les formulaires permettant aux administrateurs de choisir entre ajouter, modifier ou supprimer des voitures sont construits en HTML. De même, le menu de navigation de l'interface administrateur est également basé sur HTML.

Pour le style et la mise en forme, nous utilisons des feuilles de style CSS. Chaque page du site dispose de son propre CSS pour gérer la mise en page spécifique, tandis que le menu de l'interface d'administration utilise un ensemble de styles CSS commun. Cela assure une cohérence visuelle sur l'ensemble du site tout en permettant des personnalisations spécifiques à chaque page.

Tout le développement a été réalisé dans l'éditeur de code Visual Studio Code, offrant un environnement de développement pratique et efficace pour écrire, tester et déboguer le code.

- Programmation

La programmation de la partie client a débuté au début de janvier, et pour moi, et ma tâche était de concevoir les pages voitures, Toyota et de Ford et également de concevoir la page service.

Pour pouvoir connecter nos pages à la base de données MySQL nous avons créé une page connexion.php

```
<?php

//définir vos paramètres serveur de connexion avec la base de données
$host="localhost";
// nom utilisateur
$login="root";
// mot de passe
$pass="";
// nom de la base de données
$dbname="supercar";

// établir la connexion avec la base de données
$bdd = mysqli_connect($host, $login, $pass, $dbname);
// vérification de la connexion avec la BD
if (!$bdd) {
    die("Échec de la connexion : " . mysqli_connect_error());
}

// changer le jeu de caractères à utf8
mysqli_set_charset($bdd, "utf8");
?>
```

Ceci est le code de connexion, elle établit une connexion a la base de données supercarré.

Explication du code :

- \$host: L'hôte où se trouve la base de données. Dans ce cas, c'est localhost, ce qui signifie que la base de données est hébergée sur la même machine que le serveur web.
- \$login: Le nom d'utilisateur utilisé pour se connecter à la base de données. Dans cet exemple, c'est "root", qui est souvent le nom d'utilisateur par défaut pour les installations locales de MySQL.
- \$pass: Le mot de passe pour l'utilisateur de la base de données. Dans cet exemple, il est vide, ce qui signifie qu'aucun mot de passe n'est utilisé. En production, il est essentiel d'utiliser un mot de passe sécurisé.
- \$dbname: Le nom de la base de données à laquelle se connecter. Dans cet exemple, c'est "supercar".

**Programmation de la partie Client (Pages Voiture et Services) :**

Voici le code de voiture :

```
<?php

include("connexion.php");

$query = "SELECT distinct marque FROM VOITURE";
$result = mysqli_query($bdd, $query);

    if (!$result) {
        die("Query failed: " . mysqli_error($bdd));
    }
?>

<li class="nav-item dropdown">
    <a class="nav-link mx-lg-2 dropdown-toggle active" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false">Voitures</a>
    <ul class="dropdown-menu">
        <?php

            while ($row = mysqli_fetch_assoc($result)) {
                $marque = $row['marque'];
                $lien_voiture = "c_voiture_marque.php?marque=" . strtolower($marque);
                ?>
                <li><a class="dropdown-item" href="<?php echo $lien_voiture; ?>"><?php echo $marque; ?></a></li>
            }
        ?>
    </ul>
</li>
```

Cette partie de code est inclut dans la barre de navigation.

La ligne `include("connexion.php");` inclut le fichier `connexion.php`, qui contient le code pour établir la connexion à la base de données MySQL. Cela permet d'accéder à la variable `$bdd`, qui représente la connexion à la base de données, dans ce fichier.

La requête SQL `SELECT distinct marque FROM VOITURE` est utilisée pour sélectionner les marques distinctes des voitures de la table `VOITURE` dans la base de données.

La condition `if (!$result)` vérifie si la requête a échoué. Si tel est le cas, le script s'arrête en affichant un message d'erreur.

À l'intérieur de la boucle `'while'`, le nom de la marque est extrait du tableau et stocké dans la variable `$marque`.

Un lien est généré pour chaque marque de voiture en utilisant la fonction `strtolower()` pour convertir le nom de la marque en minuscules et en l'ajoutant à l'URL `'c_voiture_marque.php?marque='`. Ce lien est stocké dans la variable `$lien_voiture`.

Chaque marque est affichée comme un élément de liste déroulante avec un lien cliquable vers la page `'c_voiture_marque.php'` avec le nom de la marque en tant que paramètre.

```

<?php
// Mettre en exécution la connexion avec la base de données
include("connexion.php");

// Votre requête SQL pour récupérer les détails spécifiques de la marque (par exemple, pour Audi)
$marque = isset($_GET['marque']) ? $_GET['marque'] : '';

// Validation ou nettoyage éventuel
// ...

// Préparer votre requête pour sélectionner des données dans la table VOITURE avec la condition sur la marque
$select = "SELECT * FROM voiture WHERE marque = '$marque'";

// Exécuter la requête avec la fonction PHP pour récupérer les données dans la variable $curs
$curs = mysqli_query($bdd, $select);

// Assurez-vous que la requête s'est bien exécutée
$heroSectionDisplayed = false;
if ($curs) {
    while ($row = mysqli_fetch_array($curs)) {
        if (!$heroSectionDisplayed) {
            $logo_marque = $row["logo_marque"];
            $row_marque = $row["marque"];

            echo "<section class='hero-section' style='background: url($logo_marque) center/cover no-repeat;'>";
            echo "<div class='container d-flex align-items-center justify-content-center fs-1 text-white flex column'>";
            echo "<h1>" . strtoupper($row_marque) . "</h1>"; // Corrected variable name
            echo "</div>";
            echo "</section>";

            $heroSectionDisplayed = true; // Set to true after displaying the hero section
        }
    }
}

```

Comme dans le code précédent, le fichier connexion.php est inclus pour établir la connexion à la base de données.

La variable \$marque est initialisée en vérifiant si un paramètre marque est passé via la méthode GET. Si tel est le cas, sa valeur est attribuée à \$marque.

Une requête SQL est préparée pour sélectionner toutes les colonnes de la table voiture où la colonne marque correspond à la marque spécifique récupérée. La requête est stockée dans la variable \$select.

Une boucle while est utilisée pour parcourir chaque ligne de résultat de la requête SQL.

Pour chaque voiture, une section est affichée avec l'image de la voiture, son modèle, une brève description et un lien vers une autre page pour plus d'informations.

Voici le code :

```

include("connexion.php");

// Your original code used $marque, but here it should be $modele for consistency
$modele = isset($_GET['modele']) ? $_GET['modele'] : '';

// Validation or sanitization could be added here

// Initialize the flag for the hero section
$heroSectionDisplayed = false;

// Prepare your SQL query to select data from the "voiture" table with the condition on the "modele"
$selection = "SELECT * FROM voiture WHERE modele = '$modele'";

// Execute the query to retrieve data into the $curseur variable
$curseur = mysqli_query($bdd, $selection);

// Make sure the query executed successfully
if ($curseur) {
    while ($row = mysqli_fetch_array($curseur)) {
        // BEGIN HERO SECTION
        if (!$heroSectionDisplayed) {
            $logo_marque = $row["logo_marque"];
            $row_marque = $row["marque"];

            echo "<section class='hero-section' style='background: url($logo_marque) center/cover no-repeat;'>";
            echo "<div class='container d-flex align-items-center justify-content-center fs-1 text-white flex column'>";
            echo "<h1>" . strtoupper($row_marque) . "</h1>"; // Use $row_marque instead of $marque
            echo "</div>";
            echo "</section>";

            $heroSectionDisplayed = true; // Set to true after displaying the hero section
        }
    }
}

```

```

// BEGIN CAR DETAILS
$modele = $row["modele"];
$annee = $row["annee"];
$description_interieur = $row["description_interieur"];
$description_exterieur = $row["description_exterieur"];
$image_front = $row["image_front"];
$image_back = $row["image_back"];
$image_interieur = $row["image_interieur"];
$image_profile = $row["image_profile"];
$prix = $row["prix"];
$logo_prix = $row["logo_prix"];
$logo_vitesse = $row["logo_vitesse"];
$vitesse = $row["vitesse"];

echo "<div class='container-fluid mt-3 mx-auto'>";

```

Ces variables contiennent les données spécifiques d'une voiture, qui seront utilisées pour afficher les détails de cette voiture sur la page Web.

```

echo "<table class='table car-details'>";
echo "<tr>";
echo "<td><img src='$image_front' width='100%' alt='Front View'></td>";
echo "<td>";
echo "<h1><b><center>$modele</center></b></h1>";
echo "<p><center><b>Année:</b> $annee</center></p>";
echo "</td>";
echo "</tr>";

echo "<tr>";
echo "<td>";
echo "<p><img src='$logo_prix' alt='logo Prix' width='60'>$prix</p>";
echo "<p><img src='$logo_vitesse' alt='logo vitesse' width='55'>$vitesse</p>";
echo "</td>";
echo "<td>";
echo "<img src='$image_back' width='100%' alt='Back View'></td>";
echo "</tr>";

echo "<tr>";
echo "<td>";
echo "<img src='$image_profile' width='100%' alt='Back View'></td>";
echo "<td>";
echo "<p><h3><b><center>Extérieure</center></h3></b></p>";
echo "<p><center>$description_exterieur</center></p>";
echo "</td>";
echo "</tr>";

```

```

echo "<tr>";
echo "<td>";
echo "<h3><b><center>Intérieure</center></b></h3>";
echo "<p> <center>$description_interieur</center></p>";
echo "</td>";
echo "<td><img src='$image_interieur' width='100%' alt='Interior View'></td>";
echo "</tr>";

echo "</table>";
echo "</div>";

```

Ces lignes de code génèrent la structure HTML pour afficher les détails de la voiture dans un tableau `<table class= 'table car-détails'>` est un tableau avec une class CSS

Les données de la voiture sont classées dans un tableau CSS.

Chaque texte est illustré par une image.

- `<tr>`: Démarre une nouvelle ligne dans le tableau.
- `<td><img src='$image_front' width='100%' alt='Front View'></td>`: Insère une cellule contenant une image de la vue avant de la voiture. L'image est définie par la variable `$image_front` et est affichée à une largeur de 100%. L'attribut `alt` fournit un texte alternatif pour l'image.
- `<td>`: Ouvre une nouvelle cellule dans la même ligne.

- `<h1><b><center>$modele</center></b></h1>`: Affiche le modèle de la voiture en tant que titre principal centré dans la cellule.
- `<p><center><b>Année:</b> $annee</center></p>`: Affiche l'année de fabrication de la voiture centrée dans la cellule.
- `</td>`: Ferme la cellule.
- `</tr>`: Ferme la première ligne du tableau.

Passons à la programmation de service :

```
<?php
include("connexion.php");
$select = "SELECT image,type_service, description_service FROM p_service";
$curseur = mysqli_query($bdd, $select);

if ($curseur) {
// Assurez-vous qu'il y a des résultats à afficher
if (mysqli_num_rows($curseur) > 0) {
while ($row = mysqli_fetch_array($curseur)) {
    $v_image = $row["image"];
    $v_type_service = $row["type_service"];
    $v_description = $row["description_service"];

    echo "<div class='container-fluid mt-3 mx-auto'>";
    echo "<div class='row'>";

    echo "<div class='col-sm-4'><img src='$v_image' width='100%'></div>";
    echo "<div class='col-sm-4 p-3 milieu'>";
    echo "<div class='container-fluid'>
        <h5> $v_type_service </h5>
        <p> $v_description </p>
    </div>
    </div>";
}
}
```

La `$select = "SELECT image,type_service, description_service FROM p_service"` : Cette ligne définit la requête SQL pour sélectionner les colonnes image, type\_service et description\_service de la table p\_service.

`$v_image`, `$v_type_service`, `$v_description` : Ces variables contiennent les valeurs des colonnes image, type\_service et description\_service respectivement pour chaque ligne de résultat.

Après avoir la partie client, nous allons voir la partie admin :

Pour la partie admin, nous avons créé des interfaces qui vont permettre aux administrateurs d'ajouter, modifier et supprimer des voitures, la page d'accueil, l'inscription, service, demande d'essai et contact.

Voici le code pour pouvoir visualiser les voitures :



```

<div class="container on p-5">
  <a href="a_menu_voiture.html">Retour au Menu</a>

  <table class="table table-striped table-bordered">
    <thead>
      <tr>
        <th>Identifiant</th>
        <th>Marque</th>
        <th>Modèle</th>
        <th>Année</th>
        <th>Mini Description</th>
        <th>Description Intérieure</th>
        <th>Description Extérieure</th>
        <th>Vitesse</th>
        <th>Prix</th>
      </tr>
    </thead>

```

```

<?php
include("connexion.php");
$result = mysqli_query($bdd, "SELECT * FROM voiture ORDER BY id_voiture DESC");

while ($res = mysqli_fetch_array($result)) {
  echo "<tr>";
  echo "<td>" . $res['id_voiture'] . "</td>";
  echo "<td>" . $res['marque'] . "</td>";
  echo "<td>" . $res['modele'] . "</td>";
  echo "<td>" . $res['annee'] . "</td>";
  echo "<td>" . $res['mini_description'] . "</td>";
  echo "<td>" . $res['description_interieur'] . "</td>";
  echo "<td>" . $res['description_exterieur'] . "</td>";
  echo "<td>" . $res['vitesse'] . "</td>";
  echo "<td>" . $res['prix'] . "</td>";
  echo "</tr>";
}
?>
</tbody>
</table>

```

‘mysqli\_query(\$bdd, "SELECT \* FROM voiture ORDER BY id\_voiture DESC")’ : Exécute une requête SQL pour sélectionner toutes les colonnes de la table voiture, triées par ordre décroissant de l'identifiant de voiture.

- Le ‘echo "<td>" . \$res['...'] . "</td>” : Affiche chaque valeur de colonne dans une cellule du tableau.
-

Pour le code de supprimer une voiture, l'administrateur est face à un choix de plusieurs voitures placer dans un tableau et dans la colonne d'action, l'administrateur devra cliquer sur supprimez et un pop s'affichera, annonçant si vous voulez confirmez votre choix, et lorsque vous cliquez sur « valider » la voiture choisie sera définitivement supprimée de la base de données.

Voici le code de supprimer qui est séparée en deux pages

Une page contient un code qui permet de faire l'action de supprimer une voiture.

```
<?php
include("connexion.php");
$id_voiture = $_GET['id_voiture'];
$result = mysqli_query($bdd, "DELETE FROM voiture WHERE id_voiture=$id_voiture");
header("Location:a_voiture_supprimer.php");
?>
```

Pour pouvoir récupérer l'identifiant de la voiture à supprimer, le '\$id\_voiture = \$\_GET['id\_voiture']' permet de récupérer l'identifiant de la voiture à supprimer à partir de la requête GET. Ainsi, le '\$result = mysqli\_query(\$bdd, "DELETE FROM voiture WHERE id\_voiture=\$id\_voiture")', exécute une requête SQL DELETE pour supprimer la voiture de la table 'voiture' où l'identifiant de voiture correspond à celui récupéré précédemment.

L'image dessous est la deuxième page de suppression,

C'est une page qui présente les données de la table voiture en forme de tableau et cette page est le côté plus le côté visuelle ayant ainsi un stylesheet qui s'appelle « a\_style\_menu.css »

```
<div class="container one p-5">
<a href="a_menu_voiture.html">Retour</a>
<h4 class="text-center fw-bold">Veuillez supprimer la voiture de votre choix</h4>
<table>
<thead>
<tr>
<th>Identifiant</th>
<th>Marque</th>
<th>Modèle</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<?php
// Requête pour récupérer les données après la suppression
$result = mysqli_query($bdd, "SELECT * FROM voiture");

while ($res = mysqli_fetch_array($result)) {
    echo "<tr>";
    echo "<td>".$res['id_voiture']. "</td>";
    echo "<td>".$res['marque']. "</td>";
    echo "<td>".$res['modele']. "</td>";
    echo "<td>";
    echo "<a href='\"a_voiture_supp.php?id_voiture=$res[id_voiture]\"' onClick='\"return confirm('Veuillez confirmer ?')\"'>Supprimer</a>";
    echo "</td>";
    echo "</tr>";
}
?>
</tbody>
</table>
</div>
```

L'ajout de voiture, cette page consiste à l'administrateur d'ajouter une nouvelle voiture tout en remplissant les caractéristiques nécessaires de la voiture,

Voici le code :

```
include("connexion.php");
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $marque = $_POST["marque"];
    $modele = $_POST["modele"];
    $annee = $_POST["annee"];
    $logo_marque = $_POST["logo_marque"];
    $image_front = $_POST["image_front"];
    $image_interieur = $_POST["image_interieur"];
    $image_front = $_POST["image_front"];
    $image_back = $_POST["image_back"];
    $image_profile = $_POST["image_profile"];
    $logo_vitesse = $_POST["logo_vitesse"];
    $vitesse = $_POST["vitesse"];
    $prix = $_POST["prix"];
    $logo_prix = $_POST["logo_prix"];
    $mini_description = $_POST["mini_description"];
    $description_interieur = $_POST["description_interieur"];
    $description_exterieur = $_POST["description_exterieur"];
```

Les lignes suivantes récupèrent les valeurs des champs du formulaire en utilisant la superglobale \$\_POST et les stockent dans des variables PHP. Ces variables contiennent les informations sur la voiture saisies par l'utilisateur dans le formulaire.

Les lignes suivantes récupèrent les valeurs des champs du formulaire en utilisant la superglobale \$\_POST et les stockent dans des variables PHP. Ces variables contiennent les informations sur la voiture saisies par l'utilisateur dans le formulaire.

E les noms des variables correspondent aux noms des champs du formulaire HTML.

De plus, les variables \$marque, \$modele, \$annee, ..., \$description\_exterieur contiennent les valeurs saisies par l'utilisateur pour différents attributs d'une voiture, tels que la marque, le modèle, l'année, les images, la vitesse, le prix, et les descriptions intérieure et extérieure.

```

$image_interieur1 = 'images/'. $image_interieur;

$image_front1 = 'images/'. $image_front;

$image_back1 = 'images/'. $image_back;

$image_profile1 = 'images/'. $image_profile;

$logo_vitesse1 = 'images/'. $logo_vitesse;

$logo_prix1 = 'images/'. $logo_prix;

$logo_marque1 = 'images/'. $logo_marque;

```

Ce code montre le chemin d'accès aux images

`$image_interieur1 = 'images/'. $image_interieur;` :

Cette ligne concatène le chemin relatif 'images/' avec le nom de fichier de l'image stockée dans la variable `$image_interieur`. Cela crée un chemin complet menant à l'image d'intérieur de la voiture.

De manière similaire, les autres lignes construisent des chemins complets pour les autres images, comme l'image de devant (`$image_front1`), l'image de derrière (`$image_back1`), l'image de profil (`$image_profile1`), le logo de vitesse (`$logo_vitesse1`), le logo de prix (`$logo_prix1`), et le logo de marque (`$logo_marque1`).

```

// Prepare your SQL query to insert data into the "CONTACT" table
$insertQuery = "INSERT INTO VOITURE (marque, modele, annee, logo_marque,image_interieur,image_front,image_back,image_profile,logo_vitesse,vitesse,prix,
logo_prix,mini_description,description_interieur,description_exterieur)
VALUES ('$marque', '$modele', '$annee', '$logo_marque1', '$image_interieur1', '$image_front1', '$image_back1', '$image_profile1', '$logo_vitesse1', '$vitesse', '$prix',
'$logo_prix1', '$mini_description', '$description_interieur', '$description_exterieur')";

// Execute the query using mysqli_query
if (mysqli_query($bdd, $insertQuery)) {
    echo "Registration successful!";
} else {
    echo "Error: " . mysqli_error($bdd);
}
}
?>

```

La variable `$insertQuery` contient une chaîne de caractères SQL qui définit une instruction d'insertion dans la table VOITURE.

Les valeurs à insérer sont extraites des variables PHP définies précédemment, telles que `$marque`, `$modele`, `$annee`, etc.

Ces valeurs sont placées dans la requête SQL à l'aide de marqueurs de substitution ('\$marque', '\$modele', etc.)

```
<div class="container one p-5">
<a href="a_menu_voiture.html">Retour</a>
<h4 class="text-center fw-bold">Ajout de voiture</h4>

<form class="form-group" method="post" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>">
<div class="row mb-3">
  <div class="col-md-4">
    <label for="marque" class="form-label">Marque</label>
    <input type="text" name="marque" class="form-control" placeholder="Marque" required>
  </div>
  <div class="col-md-4">
    <label for="modele" class="form-label">Modèle</label>
    <input type="text" name="modele" class="form-control" placeholder="modèle" required>
  </div>
  <div class="col-md-4">
    <label for="annee" class="form-label">Année</label>
    <input type="text" name="annee" class="form-control" placeholder="année" required>
  </div>
  <div class="mb-3">
    <label for="mini_description" class="form-label">Mini Description</label>
    <textarea name="mini_description" class="form-control" rows="5" placeholder="Mini Description" required></textarea>
  </div>
</div>
```

Ce code montre une partie de comment est construit la visualisation, qui sont en « row » c'est-à-dire rangée et « col » en colonne.

Le `<label for="marque" class="form-label">Marque</label>` est une balise `<label>` qui associe un libellé ("Marque") à l'élément de saisie. L'attribut `for="marque"` spécifie que cet élément de label est lié à l'élément de formulaire avec l'attribut `name="marque"`.

Et le `<input type="text" name="marque" class="form-control" placeholder="Marque" required>` est un champ de saisie de type texte qui permet à l'utilisateur d'entrer la marque de la voiture. Les attributs `name="marque"` spécifient le nom de l'élément de formulaire, qui sera utilisé pour récupérer la valeur saisie côté serveur.

Et pour les autres interfaces ce constituent de la même façon avec moins de variables, et changer le nom de la table qui va récupérer les données et les variables également.

Dans l'ensemble, la partie admin du site offre un contrôle complet sur le contenu et les utilisateurs, permettant une gestion efficace et réactive de la plateforme.

- **Test**

Après avoir programmé le code du site, nous avons lancé le programme pour vérifier son affichage conforme à nos attentes, en nous assurant notamment de la cohérence esthétique. Nous avons également testé les fonctionnalités des formulaires, tels que ceux pour les demandes de service, les essais, les inscriptions et les contacts, afin de garantir que les données saisies sont correctement enregistrées dans la base de données.

Concernant la partie administration, nous avons vérifié l'homogénéité de l'interface et nous avons testé toutes les fonctionnalités, notamment la modification, l'ajout et la suppression de

contenu. Nous avons particulièrement veillé à ce que les modifications, les suppressions et les ajouts apportées dans l'interface administrative se reflètent instantanément dans la base de données.

## **Déploiement et maintenance**

- **Déploiement**

Pour l'instant le site s'aberge en local avec localhost

C'est une pratique courante de développer un site en le faisant tourner en local avec "localhost". Cela permet de tester et de développer le site sans avoir besoin d'une connexion Internet et offre un environnement de développement sécurisé. Une fois que le site est prêt, il peut être déployé sur un serveur en ligne pour être accessible au public. L'hébergement en local offre également une flexibilité pour expérimenter et effectuer des modifications sans affecter le site en direct. Une fois que tout est parfait, le transfert vers un serveur en ligne peut se faire assez facilement.

- **Maintenance**

Le module admin est essentiel car il permet de gérer facilement les pages du site. Cela signifie qu'on peut effectuer des modifications, supprimer ou ajouter des données sans avoir à accéder directement à la base de données ou au code. Nous proposons des formulaires pour chaque rubrique pour que l'admin puissent voir les éléments qu'elle doit modifier, ou qu'elle doit supprimer ou pour ajouter l'administrateur n'a qu'à remplir les caractéristiques demandées.

L'objectif est de créer des fichiers distincts pour la modification, la suppression et l'ajout, ou bien les inclure dans un fichier de connexion.php pour se connecter à la base de données. Pour effectuer une modification, on utilise l'instruction "update", et pour la suppression, on utilise "delete from" pour sélectionner les données à supprimer. Cette approche est plus efficace et rapide, car elle simplifie le processus de gestion des contenus du site sans nécessiter de connaissances techniques avancées en SQL ou en programmation.

## **Conclusion**

En conclusion, mon équipe et moi avons réussi à finaliser le projet au cours de la semaine du 18 avril, et nous en sommes fiers, malgré les nombreuses difficultés rencontrées en cours de route.

