**CAHIER DES CHARGES**

Lien TRELLO : <https://trello.com/b/eJd9vV5X/a-faire>

Lien github : <https://github.com/Michelhof1978/GarageParrot>

Lien Figma : <https://www.figma.com/file/rhs91pTc1st89ApCbMHoel/GARAGE-PARROT?type=design&node-id=0-1&mode=design&t=b8DACnrXU32mmG6U-0>

**Documents joints avec le projet :**

-Maquette Figma

-User stories admin + Utilisateurs

-Cahier des charges

-Gestion des tâches avec Trello

-Diagramme de cas d’utilisation

-Diagramme de séquences

-Diagramme de classes avec Mysql

**TECHNOLOGIES UTILISEES POUR LE PROJET**

-Front : React, Bootstrap, Html et Css

-Back : Php 8, MySql en SGBDR

**DESCRIPTION DU PROJET**

Le projet consiste à créer une application web vitrine pour le Garage V. Parrot, un garage automobile basé à Toulouse. Le garage offre des services de réparation automobile, d'entretien et de vente de véhicules d'occasion. Le but de l'application est de mettre en avant la qualité des services proposés par le garage.

Voici un aperçu des principales fonctionnalités à développer :

1 -Se connecter : L’administrateur et les employés doivent pouvoir se connecter à l’application en utilisant un formulaire de connexion. Les identifiants seront email et mot de passe sécurisé. Seul l’admin pourra créer un compte à ses employés.

2- Présenter les services : La page d’accueil de l’application doit afficher clairement les différents services de réparation automobile proposés par le garage. L’administrateur doit pouvoir ajouter, modifier, supprimer ces informations depuis son espace d’administration.

3 -Définir les horaires d’ouverture : Les horaires du garage doivent être visibles sur le site, de préférence dans le pied de page. L’administrateur doit pouvoir spécifier les horaires d’ouverture depuis son espace dédié.

4 -Exposer les voitures d’occasion : L’application doit afficher les véhicules disponibles à la vente, avec une galerie de photos, des descriptions détaillées et des informations techniques (un tableau de caractéristiques ainsi que la liste des équipements et options installés). Chaque voiture doit afficher son prix, une image principale, l’année de mise en circulation et le kilométrage. Les employés doivent pouvoir ajouter de nouvelles voitures depuis leur espace.

5- Filtrer la liste des véhicules d’occasion : Pour faciliter la recherche des visiteurs, un système de filtres doit être mis en place. Les visiteurs ajuster les résultats en fonction d’une fourchette de prix, d’un nombre de kilomètres parcourus ou d’une année de mise en circulation. Pour obtenir la meilleure expérience utilisateur possible, les filtres devront être appliqués sans recharger la page.

Les voitures seront rangés par 5 catégories : Berline, Suv, Familiale, utilitaire et Citadine avec mise en avant zoom en mode bouton pour une meilleur expérience utilisateur.

6 – Permettre de contacter l’atelier : Les visiteurs doivent pouvoir contacter le garage facilement, que ce soit par téléphone ou en remplissant un formulaire de contact. L’utilisateur devra alors fournir son nom, prénom, adresse email, numéro de téléphone et un message. De plus, toutes les informations de contact, formulaire compris, devront également être visibles en bas de chaque annonce d’un véhicule d’occasion automatiquement ajusté en fonction du titre de l'annonce pour permettre une future communication plus efficace avec le garage. Intégrer le RGPD pour les infos persos.

7- Recueillir les témoignages clients : Une section spécifique doit être créer pour permettre aux visiteurs de laisser des témoignages. Les témoignages doivent être modérés par un employé du garage avant d’être affichés sur la page d’accueil. Les employés auront aussi le moyen de laisser des avis client via l’espace Pro. Il y aura un système de notage avec 5 étoiles à sélectionner + nom + commentaire, les infos personnels ne devront pas être affiché en public.

8- Affichage :

-**Cookies**

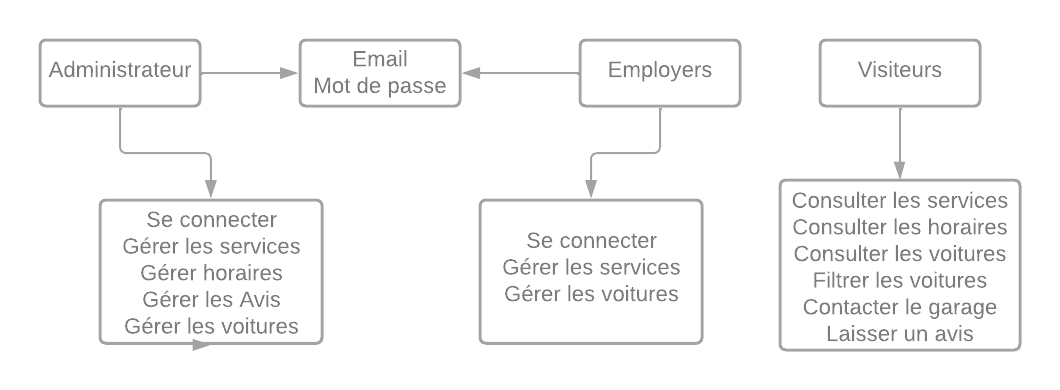
**-Mentions légales**

**-Condition générales de ventes**

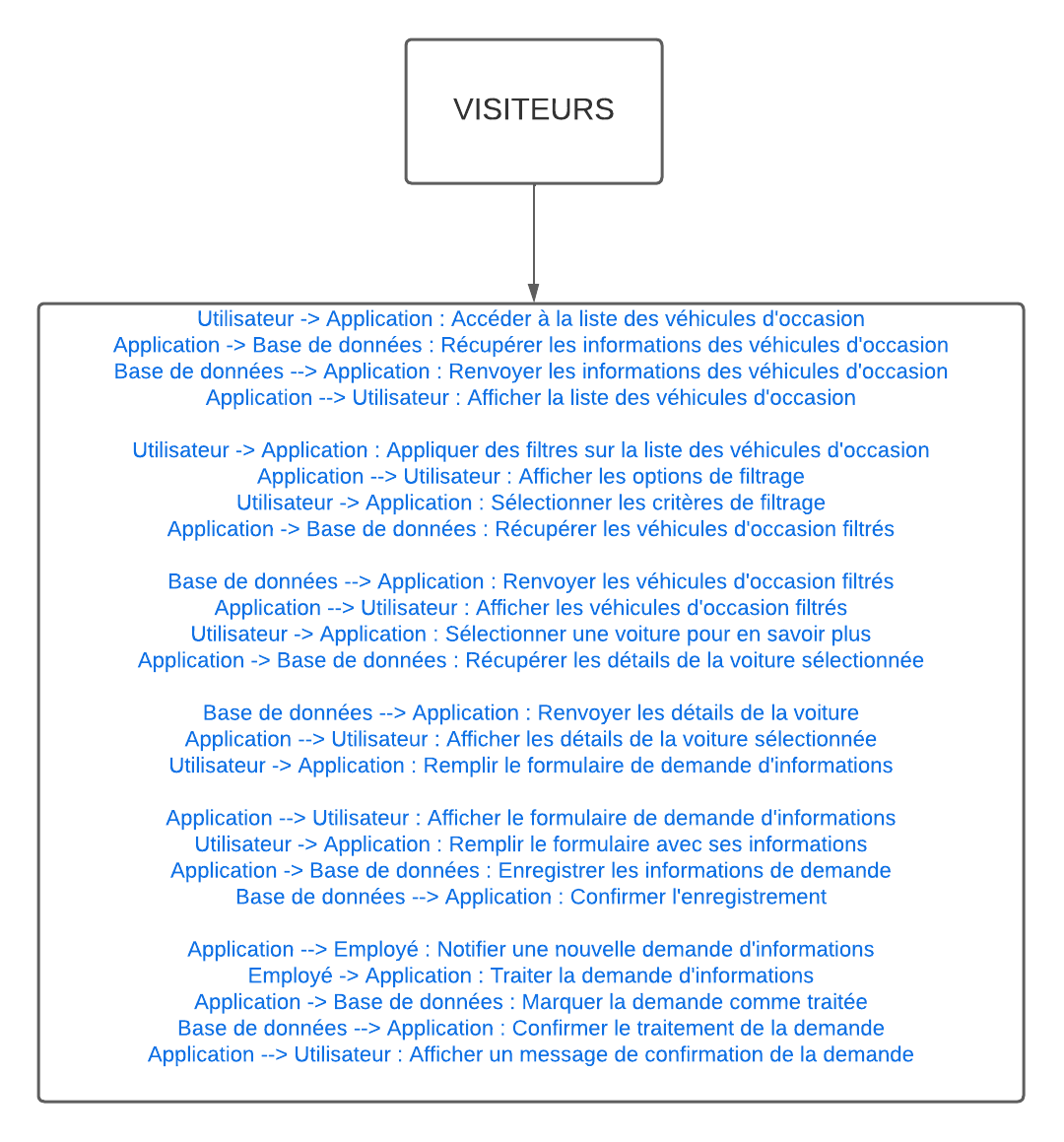
**-Politique de confidentialité**

**DIAGRAMMES**

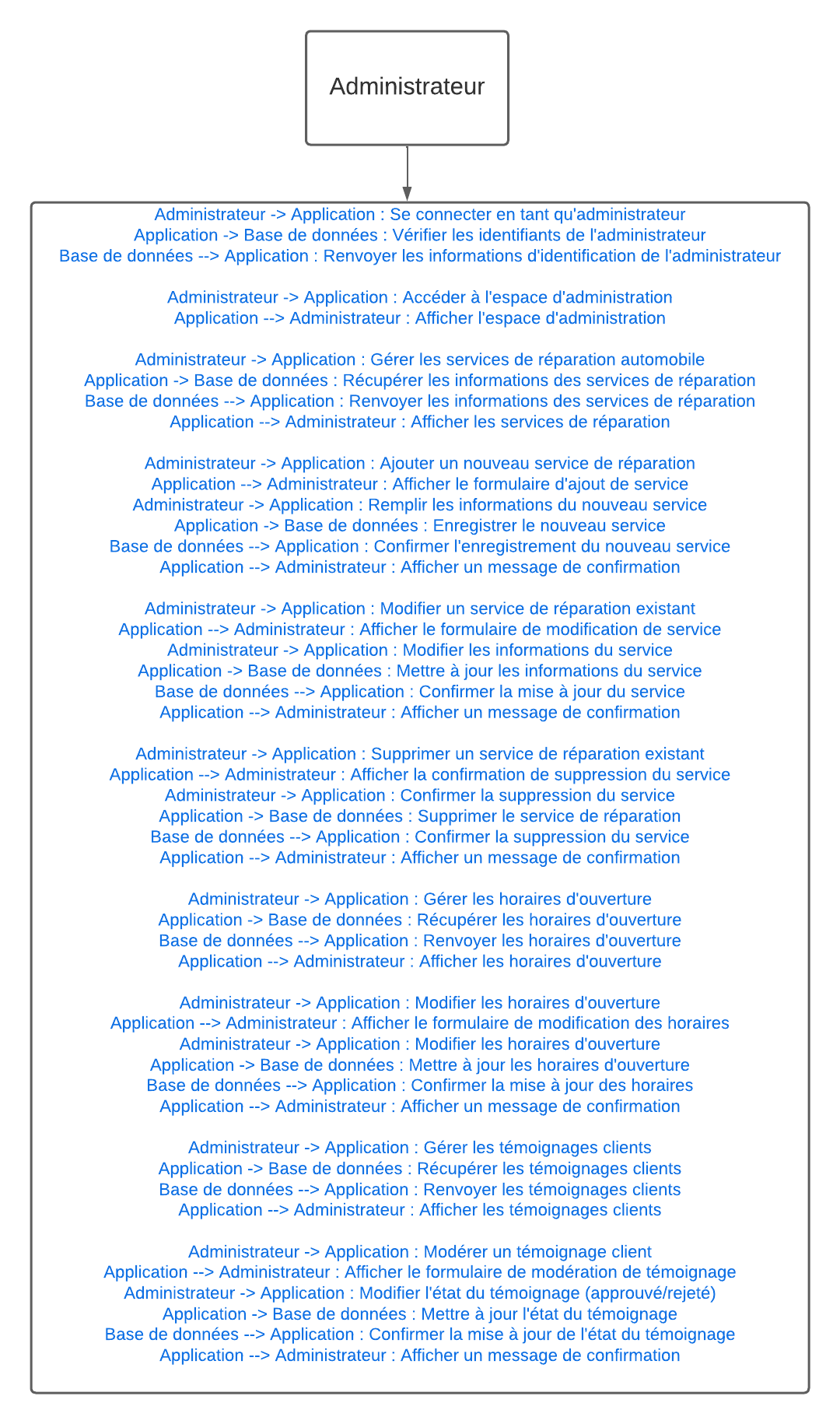
**DIAGRAMME DE CAS D’UTILISATION**



**DIAGRAMMES DE SEQUENCES**

****

Ce diagramme de séquence montre comment un nouveau visiteur peut découvrir la liste des véhicules d'occasion, appliquer des filtres pour affiner les résultats, sélectionner une voiture spécifique et remplir le formulaire de demande d'informations pour en savoir plus sur cette voiture. L'application enregistre ensuite la demande et la notifie à un employé du garage qui traitera la demande et marquera la demande comme traitée dans la base de données.

****

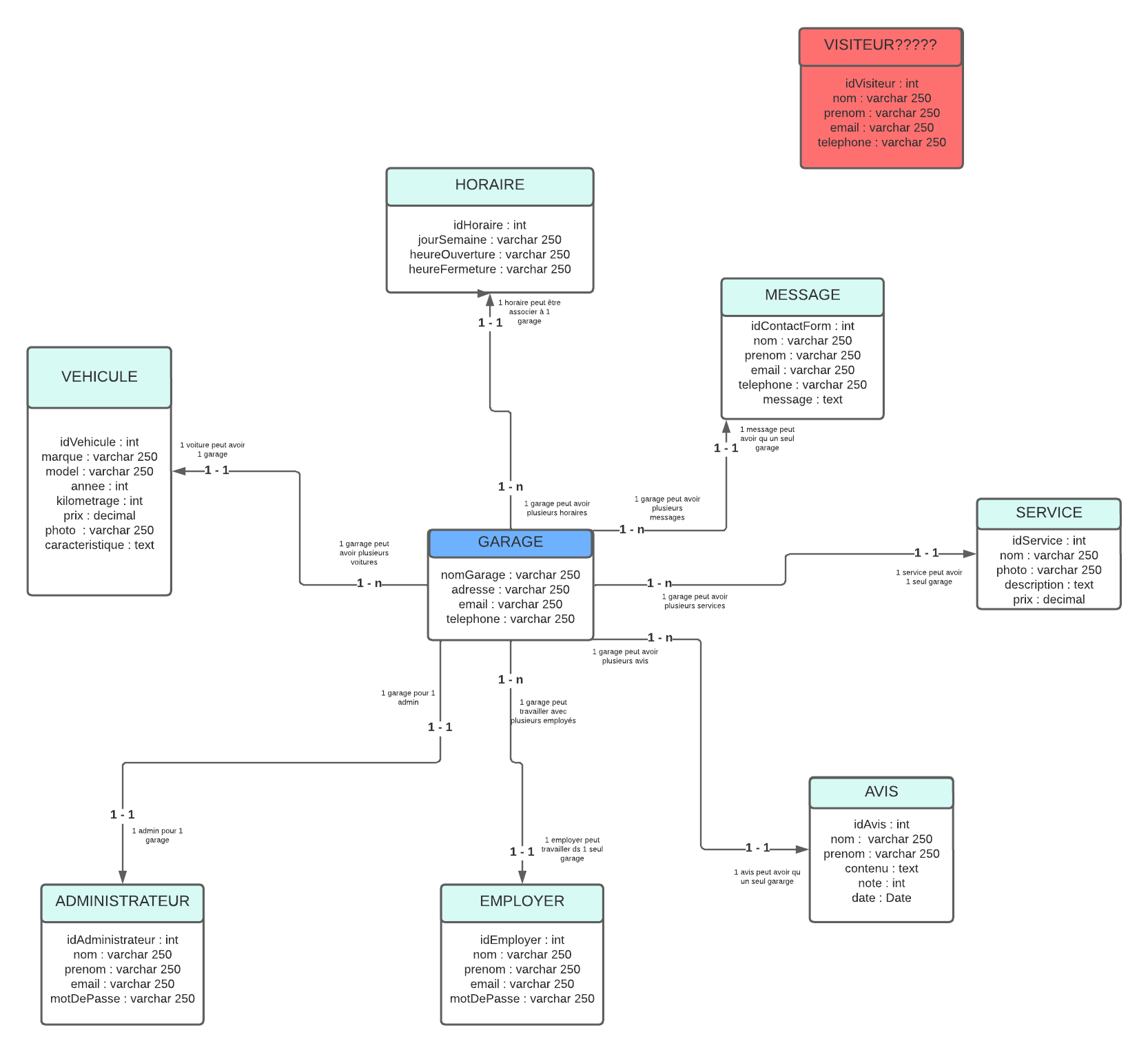
Ce diagramme de séquence montre comment un administrateur peut se connecter à l'application, accéder à l'espace d'administration, gérer les services de réparation automobile (ajouter, modifier, supprimer), gérer les horaires d'ouverture, et modérer les témoignages clients. L'application interagit avec la base de données pour récupérer et mettre à jour les informations pertinentes, et elle affiche des messages de confirmation à l'administrateur pour indiquer le succès des opérations effectuées.Haut du formulaire

Bas du formulaire

**DIAGRAMME DE CLASSES**

Ce diagramme de classes illustre la structure de base du modèle de données pour le

**Garage V. Parrot**



-Dans ce diagramme de classes, le Garage est la classe principale et possédera des méthodes pour gérer les services, les horaires, véhicules d’occasion, les contacts avec le service client ainsi que les avis clients.

-La classe Service représente un service de réparation automobile. Elle a des méthodes pour accéder et modifier ces attributs.

-La classe Vehicule représente un véhicule d’occasion. Elle a des méthodes pour accéder et modifier ces attributs.

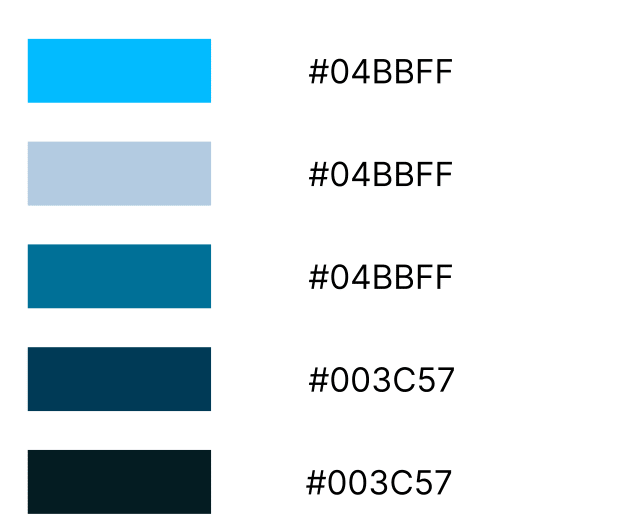
-La classe Message représente les informations de contact, telles que les demandes de contact de clients.

-La classe Horaire représente les horaire d’ouverture du garage . Elle peut avoir des méthodes pour accéder et modifier ces attributs.

-La classe Avis représente les avis ou les témoignages des clients concernant le garrage ou ses services. Elle peut avoir des méthodes pour accéder et modifier ces attributs, ainsi que pour gérer les opérations liées aux avis.

Haut du formulaire

**PALETTE DE COULEURS**



**Polices Utilisés Pour Le Site :**

**- LEAD** de Bootstrap

### - LEAD n’est pas reconnu par Figma, je mettrais en remplacement sur la maquette la police ‘**Time New Roman’**

Haut du formulaire