

Créé le 12/02/2024  
Modifié le 29/03/2024

# Documentation de GVH (Gestion des VéHicules)

# TABLE DES MATIERES

## Table des matières

PRESENTATION DU PROJET GVH.....	2
<i>Introduction</i> .....	2
REALISATION DU GESTIONNAIRE DES IMAGES DES VEHICULES .....	2
<i>Contexte</i> .....	2
<i>Table des images</i> .....	2
<i>Compteur des images dans le référentiel</i> .....	3
<i>Partie gestion des images</i> .....	4
<i>Ajouter une image</i> .....	6
<i>Modifier une image</i> .....	10
<i>Supprimer une image</i> .....	12
<i>Système de changement d'ordre (drag and drop)</i> .....	13
<i>Carrousel d'images</i> .....	14
REALISATION DES STATISTIQUES DE GVH .....	16
<i>Contexte</i> .....	16
<i>Démonstration du code par la statistique « véhicules les plus réservé »</i> .....	16
<i>Histogramme</i> .....	24
<i>Secteur</i> .....	26
<i>Tableau</i> .....	28
DONNEES NOTABLE .....	29
<i>Concepteur des données de phpMyAdmin</i> .....	30

## Présentation du projet GVH

### Introduction

**GVH** pour **Gestion des VéHicules** est une application ayant été développé par une entreprise afin de gérer les réservations des véhicules faite par les utilisateurs de leur site respectif. On y retrouve notamment le calendrier des réservations pour les utilisateurs.

The screenshot shows the GVH application's reservation calendar for April 2024. The top navigation bar includes links for 'Gestion des VéHicules (GVH)', 'Calendrier des réservations', 'Accueil', 'Administration', 'Référentiels', 'Réservations', and a shield icon. Below the navigation is a breadcrumb trail: 'Calendrier des réservations'. The main content area displays a grid for April 2024, with columns labeled from L01 to M30. The grid shows vehicle availability with different shades of gray. A legend at the bottom right indicates four status categories: 'Pré-réserve' (blue), 'Confirmé' (orange), 'Cloturé' (green), and 'En attente' (purple). The legend also includes a question mark icon. The grid header for April 2024 - Antenne Loiret lists the days of the month. The left side of the grid has a vertical column for 'VOITURE' and 'Jours chômés' (days off). Several vehicles are listed with their license plates and status: DUSTER (A 1 ⓘ), Kangoo (BB-271-BA ⓘ), Clio IV (ZZ-999-ZZ ⓘ), DUSTER (ZZ-698-YY ⓘ), KANGOO (AA-123-AA ⓘ), and A455. The background of the application features a dark blue header and footer.

## Réalisation du gestionnaire des images des véhicules

### Contexte

Il m'a été demandé pour un meilleur aperçu du véhicule par un utilisateur le consultant lors de sa réservation, de réaliser dans le backoffice, un système de gestion des images pour chaque véhicules permettant d'ajouter, supprimer, modifier et changer l'ordre des images pour qu'elle soit retranscrit dans un carrousel d'image.

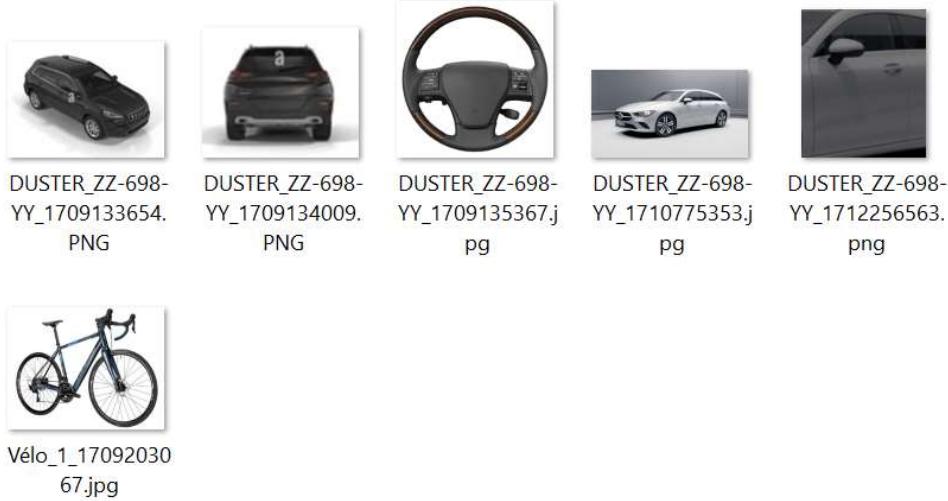
### Table des images

Pour ce gestionnaire il a fallut créer une nouvelle table nommé « **t\_images\_vehicules** » afin de stocker des informations sur les images.

id	img_cree_par	img_cree_le	img_maj_par	img_maj_le	img_libelle	img_description	img_lien	img_ordre	id_moyen_transport
50	1	2024-03-18 16:22:33	NULL	NULL	test	test	photos/DUSTER_ZZ-698-YY_1710775353.jpg	4	23
49	1	2024-02-29 11:37:47	NULL	NULL	Vélo	Photo du vélo	photos/Vélo_1_1709203067.jpg	1	34
48	1	2024-02-28 16:49:27	NULL	NULL	Volant	Photo du volant du véhicule	photos/DUSTER_ZZ-698-YY_1709135367.jpg	5	23
47	1	2024-02-28 16:26:49	1	2024-03-04 14:21:06	Coté; arrière; re	Coté; arrière; devant	photos/DUSTER_ZZ-698-YY_1709134009.PNG	1	23
46	1	2024-02-28 16:20:54	1	2024-03-18 16:20:57	Coté; avant d	Coté; avant du véhicule	photos/DUSTER_ZZ-698-YY_1709133654.PNG	3	23
51	1	2024-04-04 20:49:23	1	2024-04-05 12:50:07	portière; droite	portière; droite	photos/DUSTER_ZZ-698-YY_1712256563.png	2	23

Pour chaque images il y a un id unique, date et id de l'utilisateur l'ayant créé et/ou modifié, un libellé de l'image, une description de l'image, un lien qui indique la localisation de la photos enregistré sur le serveur suivi du nom du fichier sur le format « marque - plaque D'immatriculation - timeStamp »

64 > www > gvh2 > photos   Rechercher dans : photos 



, son ordre d'affichage dans le carrousel et le véhicule auquel il est référé.

### Compteur des images dans le référentiel

Dans la section des référentiels => Moyens de transport, j'ai inclue dans la génération des tableau Bootstrap pour chaque véhicule de chaque type un compteur d'images qui effectue une requête SQL afin de compter le nombre d'image dans la table **t\_images\_vehicules** associé à leur id.

#### VOITURE

VOITURE									
Recherche									
Modèle	Immatriculation	Energie	Kilométrage	Site	Information complémentaire	Prioritaire	Actif	NB Image	Actions
Mégane	BC-129-FC	Diesel	549 876	Antenne Cher	Pas de toit	①	Oui	2	  
C94	ZZ-777-YY	Super 98	55	Antenne Eure-et-Loire	test	①	Oui	Aucune	  
Mégane	1827 ZQ 45	Diesel	3 001	Antenne Indre	voiture de secours	①	Oui	Aucune	  

Dans le fichier « **gestion\_moyens\_transports.php** »

```

456 //Compte le nombre d'image associé à l'id du véhicule
457 $sql_nbPhoto = "SELECT COUNT(id) as nb_image
458 FROM t_images_vehicules img
459 LEFT JOIN t_moyens_transports m ON img.id_moyen_transport = m.id_moyen_transport
460 WHERE m.id_moyen_transport = ".$moyen['id_moyen_transport'];
461 $nbPhoto = $pdo->query($sql_nbPhoto)->fetch(PDO::FETCH_ASSOC)['nb_image'];

462 //gère l'affichage du nombre et de sa coloration
463 if ($nbPhoto == 0)
464 {
465     $nbPhoto = "<span style='color:red;'>Aucune</span>";//Si le véhicule n'a pas de
466 }
467 if ($nbPhoto > 0) {
468     $nbPhoto = "<span style='color:green;'>".$nbPhoto."</span>";
469 }
470

```

Et un bouton d'ajout d'image pour accéder à la partie de gestion des images associé à l'id du véhicule tout à droite de la colonne « Actions »

## Partie gestion des images

En accédant à la partie de gestion des images toujours dans **gestion\_moyens\_transports.php**, on aura accès à cette partie.

```

1125 // ****
1126 // ***** ONGLET D'INTERACTION D'IMAGES DE MOYEN DE TRANSPORT ****
1127 // ****
1128 elseif ($this_onglet['url_onglet'] == "add_image_transport" && array_key_exists($this_onglet['id_onglet'], $onglets)) {

```

Liste des moyens de transport    Ordonnancement des moyens des transport    Liste des moyens de transport archivés    Ajouter image transport

Photo(s) de : Mégane

Ouvrir carrousel d'image(s)

Pour changer l'ordre d'une image, restez appuyé, puis déplacez la.

	Libellé	Description	Actions
	côté avant	une description du côté avant	<a href="#">Modifier</a> <a href="#">Supprimer</a>
			
	côté arrière		<a href="#">Modifier</a> <a href="#">Supprimer</a>

Ajouter une image

Lorsque que l'on est sur cette page on récupère l'id du véhicule pour faire une requête afin de récupérer le nom et au possible les images associé à cette id du véhicule.

```

1129 |         $transport = getpost("id_moyen_transport"); //récupérer l'id du véhicule choisis
1130 |         //***DEBUG***
1131 |         //var_dump($transport);
1132 |
1133 |         //récupérer le nom du véhicule associé a l'id
1134 |         $sql_nom_vehicule = "SELECT libelle_moyen_transport FROM t_moyens_transports WHERE id_moyen_transport = $transport";
1135 |         $nom_vehicule = $pdo->query($sql_nom_vehicule)->fetch(PDO::FETCH_ASSOC);
1136 |     ?>
1137 |         <span>Photo(s) de : <span style="color: blue; font-size: 1.2em;"><?= $nom_vehicule['libelle_moyen_transport'] ?></span></span>
1138 |         <input type="hidden" id="id_moyen_transport" value="<?= $transport ?>">
1139 |     </?php
1140 |
1141 |         //récupérer les informations du véhicule associé a l'id avant de pouvoir les utiliser
1142 |         $sql_image_vehicule = "SELECT id, img_libelle, img_description, img_lien, img_ordre, id_moyen_transport FROM t_images_vehicules WHERE id_moyen_transport = $transport ORDER BY img_ordre ASC";
1143 |         $images_vehicule = $pdo->query($sql_image_vehicule)->fetchAll(PDO::FETCH_ASSOC);
1144 |
1145 |     ?
1170 |         //pour chaque ligne, on la rajoute dans le tableau
1171 |         foreach($images_vehicule as $image)

```

Pour chaque bouton (sauf Supprimer), ils feront appel à un modal au centre de la page dans des div attribué pour leur besoins avec le onclick pour appeler leur fonction.

```

1157 |         <!--création du tableau pour afficher les informations récupérées-->
1158 |         <table id="imageTable" class='table tablegh table-hover table-bordered table-striped table-responsive' style="text-align: center; padding-left: 5%; padding-right: 5%;">
1159 |
1160 |             <thead style="background: #cfcfcf;">
1161 |                 <tr>
1162 |                     <td></td>
1163 |                     <td>Libellé</td>
1164 |                     <td>Description</td>
1165 |                     <td>Actions</td>
1166 |                 </tr>
1167 |             </thead>
1168 |             <ul>
1173 |                 <tr style="cursor: grab;">
1174 |                     <!--afficher l'image-->
1175 |                     <td>
1176 |                         " alt="<?= $image['img_lien'] ?>" style="max-width: auto; max-height: 250px; min-height: 100px; object-fit: contain; cursor: zoom-in;" data-id="<?= $image['id'] ?>">
1177 |                     </td>
1178 |                     <!-- Centrer les éléments au centre du td -->
1179 |                     <!--afficher le titre de l'image-->
1180 |                     <td style="width: 20%; margin: auto;">
1181 |                         <?= $image['img_libelle'] ?>
1182 |                     </td>
1183 |                     <!--afficher la description de l'image-->
1184 |                     <td style="width: 30%; text-align: left; padding-left: 5px; padding-right: 5px; max-width: 20%; margin: auto;">
1185 |                         <?= $image['img_description'] ?>
1186 |                     </td>
1187 |
1188 |                     <!--INTERACTION-->
1189 |                     <td style="width: 20%; background-color: white; margin: auto;">
1190 |                         <!--bouton pour modifier les informations de l'image-->
1191 |                         <button class="btn btn-outline-info" id="btnEditImage" onclick="editImage(<?= $image['id'] ?>, '<?= $image['img_libelle'] ?>', '<?= $image['img_description'] ?>')" style="cursor: pointer;">
1192 |                             Modifier
1193 |                         </button>
1194 |                         <!--bouton pour supprimer l'image de la base de données-->
1195 |                         <button class="btn-delete btn btn-outline-danger" data-id="<?= $image['id'] ?>" style="cursor: pointer;">Supprimer</button>
1196 |                     </td>
1197 |                 </tr>
1203 |                 <!--bouton pour ajouter une image-->
1204 |                 <button class="btn btn-outline-info" id="btnAddImage" onclick="addImage(<?= $transport ?>, '<?= $nom_vehicule['libelle_moyen_transport'] ?>')" style="margin: auto; margin-top: 10px; min-width: 20%;">
1205 |                     Ajouter une image
1206 |                 </button>

```

```

1208      <!-- modal pour le popup de la modification d'image -->
1209      <div class="modal fade" id="modalEditImage" tabindex="-1" aria-labelledby="modalEditImageLabel" aria-hidden="true"></div>
1210
1211      <!--modal pour le popup de l'ajout d'image-->
1212      <div class="modal fade" id="modalAddImage" tabindex="-1" aria-labelledby="modalAddImageLabel" aria-hidden="true"></div>
1213
1214      <!--modal pour l'agrandissement de l'image dès le clic-->
1215      <div class="modal fade" id="scaleImage" tabindex="-1" aria-labelledby="scaleImageLabel" aria-hidden="true"></div>
1216

```

## Ajouter une image

Le bouton en bas de page :



Ajouter une image

Permet à l'utilisateur d'ajouter une image associé au véhicule, ouvrant un modal de formulaire

```

1266      // Script pour gérer l'ajout d'image
1267      function addImage(id_moyen_transport, nom_vehicule)
1268      {
1269          $.ajax({
1270              type: 'get',
1271              url: 'ajax_modal_add_vehicule_image.php',
1272              data: {
1273                  id_moyen_transport: id_moyen_transport,
1274                  nom_vehicule: nom_vehicule
1275              },
1276              //récupère le contenu de ajax_modal_carousel_image_vehicule.php
1277              success: function(dataHtml)
1278              {
1279                  //injecte le contenu dans le modal
1280                  $('#modalEditImage').html(dataHtml);
1281                  //affiche le modal
1282                  $('#modalEditImage').modal('show');
1283              }
1284          });
1285      }
1286

```

Par de l'ajax on envoi les informations vers le fichier de modal qui va afficher le formulaire du fichier « ajax\_modal\_add\_vehicule\_image.php »



```

14  <!-- Modal pour modifier une image, menant vers le formulaire d'ajout d'image (ajax_add_vehicule_image.php) -->
15 <div class="modal-dialog modal-dialog-centered modal-lg" id="addVehiculeImage" tabindex="-1" aria-labelledby="addVehiculeImageLabel"
     aria-hidden="true">
16   <div class="modal-dialog">
17     <div class="modal-content">
18       <form id="addForm" method="post" action="ajax_add_vehicule_image.php" enctype="multipart/form-data">
19         <div class="modal-header">
20           <h5 class="modal-title" id="addVehiculeImageLabel">Ajouter une image</h5>
21           <button class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
22         </div>
23         <div class="modal-body">
24           <!--libellé de l'image-->
25           <label for="image" class="form-label">Libellé</label>
26           <input type="text" name="libelle" id="libelle" class="form-control" required>
27           <br>
28           | <!--ajout de l'image, une redimension manuelle est possible si besoin de réajuster-->
29           <label for="image" class="form-label">Image</label>
30           <input type="file" name="image" id="image-input" class="form-control"
31             accept="image/.jpg, image/.jpeg, image/.png, image/.gif" required><!--accepte les images de type jpg, jpeg, png, gif-->
32           <input type="hidden" name="id_moyen_transport" value="<?= $id_moyen_transport ?>"><!--récupérer l'id du véhicule de la
33             page-->
34           <br>
35           | <!--ajout de l'image, une redimension manuelle est possible si besoin de réajuster-->
36           <label for="image" class="form-label">Image</label>
37           <input type="file" name="image" id="image-input" class="form-control"
38             accept="image/.jpg, image/.jpeg, image/.png, image/.gif" required><!--accepte les images de type jpg, jpeg, png, gif-->
39           <input type="hidden" name="id_moyen_transport" value="<?= $id_moyen_transport ?>"><!--récupérer l'id du véhicule de la
40             page-->
41           <br>
42           | <!--description de l'image-->
43           <label for="description" class="form-label">Description</label>
44           <textarea name="description" id="description" class="form-control"></textarea>
45           | <!-- récupérer l'id du véhicule pour l'ajouter dans la base de données lors de l'ajout de l'image-->
46           <input type="hidden" name="id_moyen_transport" value="<?= $id_moyen_transport ?>">
47           | <!-- récupérer le nom du véhicule pour le nom du fichier image-->
48           <input type="hidden" name="nom_vehicule" value="<?= $nom_vehicule ?>">
49           </div>
50           <div class="modal-footer">
51             <button type="button" class="btn btn-outline-danger" data-bs-dismiss="modal">Fermer</button>
52             <button type="submit" class="btn btn-outline-info">Ajouter</button>
53           </div>
54         </form>
55       </div>
56     </div>
57   </div>
58 </div>

```

Le code du formulaire est récupéré par le success de l'ajax pour l'appliquer dans le modal correspondant, dès l'envoi des éléments du formulaire on fait appel au fichier « **ajax\_add\_vehicule\_image.php** », son fonctionnement est précisé dans le code ci dessous

```

9 // Vérifier que les données sont bien envoyées
10 if (empty(getpost('libelle'))) {
11     echo "Requête invalide.";
12     exit;
13 }
14
15 // vérifier dans $_FILES si le fichier a bien été envoyé et s'il n'y a pas d'erreur
16 if (isset($_FILES['image']) && $_FILES['image']['error'] === 0) {
17
18     $imageContent = file_get_contents($_FILES['image']['tmp_name']); //récupération du contenu du fichier
19
20     // Vérifier l'extension du fichier
21     $extensions_authorized = array('jpg', 'jpeg', 'png', 'gif', 'JPG', 'JPEG', 'PNG', 'GIF');
22     $extension = pathinfo($_FILES['image']['name'], PATHINFO_EXTENSION);
23     if (in_array($extension, $extensions_authorized)) {
24         //vérification de la taille du fichier
25         $tailleMax = 5000000; //SMO
26         if ($_FILES['image']['size'] > $tailleMax) {
27             //compression de l'image pour aller en dessous de la taille max au minimum
28             $imageContent = imagecreatefromstring($imageContent);
29             ob_start(); //démarrer la temporisation de sortie il permet de stocker les données dans un tampon au lieu de les envoyer
29             directement
30             imagejpeg($imageContent, NULL, 50); // $imageContent est l'image à compresser, NULL pour la sortie, 50 pour la qualité de
30             compression
31             $imageContent = ob_get_contents(); //récupère le contenu du tampon de sortie
32             ob_end_clean(); //vide le tampon de sortie et éteint la temporisation de sortie
33
34     }
35
36     //récupération et/ou génération des autres valeurs
37     $id_user = $_SESSION['ident'][ 'id_user']; //récup de l'id de l'utilisateur dans la session
38     $dateActuelle = date("Y-m-d H:i:s"); //récup de la date actuelle année-mois-jour heure:minute:seconde
39     $libelle = $_POST[ 'libelle']; //récup du libellé
40     $description = $_POST[ 'description']; //récup de la description
41     $id_moyen_transport = $_POST[ 'id_moyen_transport']; //récup de l'id du moyen de transport
42
43     //récupération de l'immatriculation associée à l'id du moyen de transport
44     $sqlMoyenTransport = "SELECT immatriculation_véhicule, libelle_moyen_transport FROM t_moyens_transports WHERE id_moyen_transport = $id_moyen_transport";
45     $recupMoyenTransport = $pdo->prepare($sqlMoyenTransport); $recupMoyenTransport->execute();
46     //récup de l'immatriculation dans la colonne immatriculation_véhicule
47     $immatriculation = $recupMoyenTransport->fetchColumn();
48
49     //récup du libellé du moyen de transport
50     $libelleMoyenTransport = $_POST[ 'nom_véhicule'];
51
52     // Renommer le fichier par le nomVéhicule + immatriculationVéhicule + timestamp + extension, séparés par des _
53     $nomFichier = "photos/" . $libelleMoyenTransport . "_" . $immatriculation . "_" . time() . "." . $extension;
54
55     //récup de l'ordre par une requête sql en allant au dessus du plus grand nombre pour l'ajout à la suite
56     $sqlOrdre = "SELECT MAX(img_ordre) FROM t_images_vehicules WHERE id_moyen_transport = $id_moyen_transport";
57     $recupOrdre = $pdo->prepare($sqlOrdre); $recupOrdre->execute();
58     $ordre = $recupOrdre->fetchColumn() + 1; //(+1 pour l'ajout à la suite)

```

```

60 //vérification des caractères spéciaux
61 $libelle = htmlentities($libelle, ENT_QUOTES, "UTF-8");
62 $description = htmlentities($description, ENT_QUOTES, "UTF-8");
63
64
65
66
67 // Ajouter les infos de l'image dans la base de données
68 try {
69     $sqlInsert = "INSERT INTO t_images_vehicules (img_cree_par, img_cree_le,#MAJ NULL car création seul
70     img_libelle, img_description, img_lien, img_ordre, id_moyen_transport)
71     VALUES ('$id_user', '$dateActuelle', '$libelle', '$description', '$nomFichier', '$ordre', '$id_moyen_transport')";
72     $ajoutPhotos = $pdo->prepare($sqlInsert);
73     $ajoutPhotos->execute();
74 }catch (Exception $e) {
75     echo "Erreur lors de l'ajout des infos de l'image dans la base de donnée : " . $e->getMessage();
76     exit;
77 }
78
79 // Déplacer le fichier dans le dossier photos
80 try {
81     file_put_contents(PGI_FOLDER . "/" . $nomFichier, $imageContent);
82 }
83 catch (Exception $e) {
84     echo "Erreur lors de l'ajout du fichier de l'image dans le dossier photos du serveur : " . $e->getMessage();
85     exit;
86 }
87
88 //revenir à la page précédente
89 header("Location: " . $_SERVER["HTTP_REFERER"]);
90
91 }
92 else {
93     echo "L'extension du fichier n'est pas autorisée.";
94     exit;
95 }
96 }
97 else {
98     echo "Erreur lors de l'envoi du fichier.";
99     exit;
100 }
101
102 ?>

```

En résumé cette partie va prendre les informations du véhicule et du formulaire pour ajouter les information de l'image dans la base de donnée et la stocker dans le dossier images du serveur.

Elle est désormais dans le tableau.

<u>id</u>	<u>img_cree_par</u>	<u>img_cree_le</u>	<u>img_maj_par</u>	<u>img_maj_le</u>	<u>img_libelle</u>	<u>img_description</u>	<u>img_lien</u>	<u>img_ordre</u>	<u>id_moyen_transport</u>
54	1	2024-04-07 16:58:39		NULL NULL	côté droit	vue du côté droit	photos/Mégane_BC-129-FC_1712501919.png	3	17



côté droit

vue du côté droit

[Modifier](#) [Supprimer](#)

## Modifier une image

Le bouton sur le coté droit d'une image :



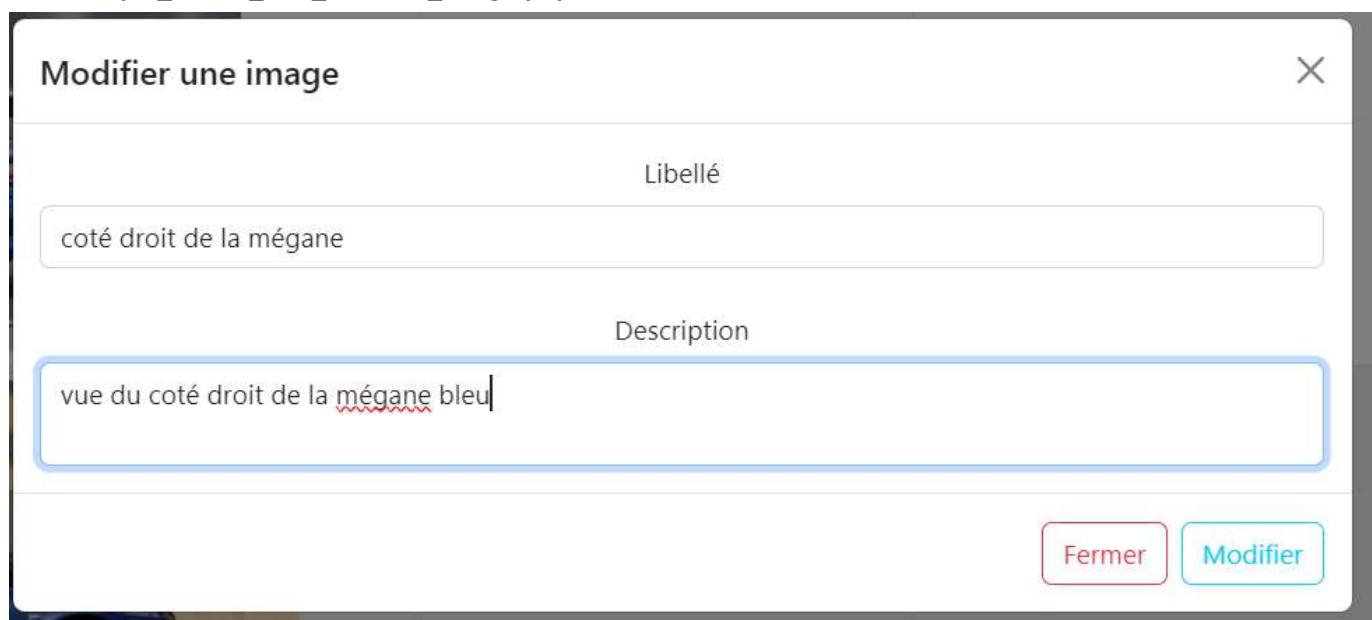
Modifier

Permet à l'utilisateur de modifier une image associé au véhicule, ouvrant un modal de formulaire

```

1219 // Script pour gérer la modification des images
1220 function editImage(id, libelle, description)
1221 {
1222     $.ajax({
1223         type: 'get',
1224         url: 'ajax_modal_edit_vehicule_image.php',
1225         data: {
1226             id: id,
1227             libelle: libelle,
1228             description: description
1229         },
1230         //récupère le contenu de ajax_modal_carousel_image_vehicule.php
1231         success: function(dataHtml)
1232         {
1233             //injecte le contenu dans le modal
1234             $('#modalEditImage').html(dataHtml);
1235             //affiche le modal
1236             $('#modalEditImage').modal('show');
1237         }
1238     });
1239 }
```

Par de l'ajax on envoi les informations vers le fichier de modal qui va afficher le formulaire du fichier « **ajax\_modal\_edit\_vehicule\_image.php** »



```

13 $id = getpost('id');
14 $libelle = getpost('libelle');
15 $description = getpost('description');
16
17 ?>
18 <!-- Modal pour modifier une image -->
19 <div class="modal-dialog modal-dialog-centered modal-lg" id="modalEditImage" tabindex="-1" aria-labelledby="modalEditImageLabel"
      aria-hidden="true">
20   <div class="modal-content">
21     <form id="editForm" method="post" action="ajax_edit_vehicule_image.php">
22       <div class="modal-header">
23         <h5 class="modal-title" id="modalEditImageLabel">Modifier une image</h5>
24         <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
25       </div>
26       <div class="modal-body text-center">
27         <!-- libellé de l'image-->
28         <label for="image" class="form-label">Libellé</label>
29         <input type="text" name="libelle" id="libelle" class="form-control" value="<?= $libelle ?>">
30         <br>
31         <!--description de l'image-->
32         <label for="description" class="form-label">Description</label>
33         <textarea name="description" id="description" class="form-control"><?= $description ?></textarea>
34
35         <!-- récupère data-id pour l'image à modifier -->
36         <input type="hidden" name="id" id="id" value="<?= $id ?>">
37       </div>
38       <div class="modal-footer">
39         <button type="button" class="btn btn-outline-danger" data-bs-dismiss="modal">Fermer</button>
40         <button type="submit" class="btn btn-outline-info">Modifier</button>
41       </div>
42     </form>
43   </div>
44 </div>

```

Le code du formulaire est récupéré par le success de l'ajax pour l'appliquer dans le modal correspondant, dès l'envoi des éléments du formulaire on fait appel au fichier

« ajax\_edit\_vehicule\_image.php », son fonctionnement est précisé dans le code ci dessous

```

9 // Vérifier que les données sont bien envoyées
10 if (empty(getpost('id')) || empty(getpost('libelle'))) {
11 echo "Requête invalide.";
12 exit;
13 }
14 //récupère l'id de l'image, son libelle et sa description
15 $id = getpost('id');
16 $libelle = getpost('libelle');
17 $description = getpost('description');
18
19 //vérification des caractères spéciaux
20 $libelle = htmlentities($libelle, ENT_QUOTES, "UTF-8");
21 $description = htmlentities($description, ENT_QUOTES, "UTF-8");
22
23 //édité le et édité par
24 $id_user = $_SESSION['ident']['id_user'];//récup de l'id de l'utilisateur dans la session
25 $dateActuelle = date("Y-m-d H:i:s");//récup de la date actuelle année-mois-jour heure:minute:seconde
26
27 // Mettre à jour les images avec les nouvelles valeurs
28 $sqlEdit = "UPDATE t_images_vehicules
29 SET img_maj_par = '$id_user', img_maj_le = '$dateActuelle', img_libelle = '$libelle', img_description = '$description'
30 WHERE id = $id";
31 $edit = $pdo->prepare($sqlEdit);
32 $edit->execute();
33
34 //revenir à la page précédente
35 header("Location: " . $_SERVER["HTTP_REFERER"]);

```

Les informations de l'image sont bien modifiée.



## Supprimer une image

Le bouton sur le coté droit d'une image :

**Supprimer**

localhost indique

Voulez-vous vraiment supprimer cette image ?

OK

Annuler

```

1241 // Script pour gérer la suppression des images
1242 $(function() {
1243     $(".btn-delete").click(function(e) {
1244         e.preventDefault();
1245         var imageId = $(this).data("id");
1246
1247
1248         // Confirmer la suppression
1249         if (confirm("Voulez-vous vraiment supprimer cette image ?")) {
1250             // Envoyer l'ID de l'image à un script PHP pour la suppression
1251             $.ajax({
1252                 url: "ajax_delete_vehicule_image.php",
1253                 method: "POST",
1254                 data: {
1255                     id: imageId
1256                 },
1257                 success: function() {
1258                     // Recharger la page après la suppression
1259                     location.reload();
1260                 }
1261             });
1262         }
1263     });
1264 });

```

Si on supprime, par l'ajax on envoi les informations vers le fichier ajax

« **ajax\_delete\_vehicule\_image.php** » qui va supprimer l'image associé à l'id de l'image dans la base de donnée (ses information) et dans son dossier où la photo est stockée.

```

9 // Vérifier la requête et les données
10 if (empty(getpost('id'))) {
11     echo "Requête invalide.";
12     exit;
13 }
14
15 $id = $_POST['id'];
16
17 //sélectionner le lien de l'image (par rapport à l'id de la photo)
18 $lien = $pdo->query("SELECT img_lien FROM t_images_vehicules WHERE id = $id")->fetchAll(PDO::FETCH_ASSOC);
19 //var_dump($lien);//DEBUG
20
21 //générer le chemin de l'image
22 $image = PGI_FOLDER . "/" . $lien[0]['img_lien'];
23 //echo $image;//DEBUG
24
25
26 // Supprimer les données de l'image de la base de données
27 $pdo->query("DELETE FROM t_images_vehicules WHERE id = $id");
28
29 // Supprimer l'image du serveur
30 if (file_exists($image)) {
31     unlink($image); // unlike supprime un fichier
32 }
33 else {
34     echo "Le fichier n'existe pas.";
35     exit;
36 }

```

## Système de changement d'ordre (drag and drop)

Le système de drag and drop fonctionne avec Jquery UI par cette utilisation : « Pour changer l'ordre d'une image, restez appuyé, puis déplacez la. »



Le code va actualiser l'ordre des ligne dans le DOM par rapport au déplacement de l'utilisateur et les récupérer un par un par leur id d'images dans l'ordre puis les mettre dans une tableau pour ensuite dans « **ajax\_delete\_vehicule\_image.php** » actualiser l'ordre des images dans la base de données par leur ordre actuel du DOM récupéré du tableau. (Voir code ci-dessous)

```

1287 // Script pour gérer le drag and drop des images pour l'ordre
1288 $(function() {
1289     // Vérifie que la table a pour id "imageTable" et qu'elle n'est pas vide
1290     if ($("#imageTable").length > 0) {
1291         // Rend la table "sortable" pour pouvoir changer l'ordre des images avec jQuery UI
1292         $("#imageTable tbody").sortable({
1293             // Change l'ordre des images lorsqu'on les déplace
1294             update: function(event, ui) {
1295                 var orderImages = []; // Tableau pour stocker les IDs dans l'ordre actuel
1296
1297                 // Met à jour l'ordre des lignes déplacées et altérées par l'opération
1298                 $("#imageTable tbody tr").each(function(index) {
1299                     var id = $(this).find("img").data("id"); // Récupère l'ID de l'image
1300                     orderImages.push(id); // Ajoute l'ID à la fin du tableau
1301                 });
1302
1303                 // Envoie le tableau des images dans la requête ajax
1304                 $.ajax({
1305                     url: "ajax_update_order_vehicule_image.php", // URL de la requête ajax
1306                     method: "POST",
1307                     data: {
1308                         orderImages: orderImages, // Envoie le tableau avec les IDs des images dans l'ordre actuel
1309                         id_moyen_transport: $('#id_moyen_transport').val() // Récupère l'ID du véhicule
1310                     },
1311                     success: function() {
1312                         console.log("Les images ont été réorganisées avec succès.");
1313                     }
1314                 });
1315             }
1316         });
1317     });
1318 });
1319 });
1320 });

```

### (ajax\_delete\_vehicule\_image.php)

```

9 //récupère les informations de l'image
10 $orderImages = getpost('orderImages');
11 $idVehicle = getpost('id_moyen_transport');
12
13 // Vérifier que les données sont bien envoyées
14 if (empty($orderImages) || empty($idVehicle)) {
15     echo "Requête invalide.";
16     exit;
17 }
18
19 // met à jour l'ordre par ceux de la liste
20 try {
21     // pour chaque ligne de la liste, on met à jour l'ordre à l'id correspondant
22     $i = 0;
23     foreach ($orderImages as $idList) {
24         $sqlUpdate = "UPDATE t_images_vehicules SET img_ordre = $i + 1 WHERE id = '$idList' AND id_moyen_transport = '$idVehicle'";
25         $update = $pdo->prepare($sqlUpdate);
26         $update->execute();
27         $i++;
28     }
29
30 } catch (PDOException $e) {
31     echo "Erreur lors de la mise à jour de l'ordre des images : " . $e->getMessage();
32     exit;
33 }
34 ?>

```

## Carrousel d'images

Le carrousel d'image est un modal qui va permettre de prendre toutes les images d'un véhicule pour les mettre en valeur afin que les utilisateurs puissent les consulter.

Le bouton sur le haut de la page :

Ouvrir carrousel d'image(s)

Le code pour appeler le modale « `ajax_modal_carousel_image_vehicule.php` », il a seulement besoin de l'id du véhicule en paramètre pour pouvoir fonctionner :

```
1146      <!-- modal pour le popup du carousel d'image -->
1147      <div class="modal fade" id="modalCarouselImageVehicule" data-bs-backdrop="static" data-bs-keyboard="false"
1148          | aria-labelledby="modalCarouselImageVehiculeLabel" aria-hidden="true" tabindex="-2" role="dialog">
1149      </div>
1150      <!-- bouton pour afficher le carousel d'image -->
1151      <button class="btn btn-outline-info" id="btnAffichageCarousel" onclick="affichageCarousel(<?= $transport ?>)"
1152          | style="margin: auto; min-width: 20%;"
1153          |     Ouvrir carousel d'image(s)
1154      </button>
```

(ajax\_modal\_carousel\_image\_vehicule.php)

```

61      <!--on affiche les boutons de navigation-->
62      <div class="carousel-indicators" style="margin: 0; margin-right:10px; margin-left:10px; position: relative;">
63          <?php
64              $i = 0;
65              $imageVehicule->execute(); //réinitialiser le curseur pour afficher les boutons de navigation
66              while ($row = $imageVehicule->fetch()) {
67                  if ($i == 0) {
68                      ?><button type="button" style="height: 10px; background-color: grey;" data-bs-target="#carouselExampleIndicators"
69                          data-bs-slide-to=<?= $i ?> class="active" aria-current="true" aria-label="Slide <?= $i ?>"></button><?php
70                  } else {
71                      ?><button type="button" style="height: 10px; background-color: grey;" data-bs-target="#carouselExampleIndicators"
72                          data-bs-slide-to=<?= $i ?> aria-label="Slide <?= $i ?>"></button><?php
73                  }
74                  $i++;
75              }
76              $imageVehicule->closeCursor();
77          ?>
78      </div>
79  </div>
80
81      <?php if ($imageVehicule->rowCount() > 1) [&] ?>
82      <div>
83          <!-- les bouton sont en noir et collé en bas du carrousel -->
84          <button class="carousel-control-prev" type="button" data-bs-target="#carouselExampleIndicators" data-bs-slide="prev"
85              style="background-color: black; width: 50px; height: 100px; margin-top: auto; margin-bottom: auto;">
86              <span class="carousel-control-prev-icon" aria-hidden="true"></span>
87              <span class="visually-hidden">Précédent</span>
88          </button>
89          <button class="carousel-control-next" type="button" data-bs-target="#carouselExampleIndicators" data-bs-slide="next"
90              style="background-color: black; width: 50px; height: 100px; margin-top: auto; margin-bottom: auto;">
91              <span class="carousel-control-next-icon" aria-hidden="true"></span>
92              <span class="visually-hidden">Suivant</span>
93          </button>
94      </div>
95      <?php [&] ?>
96      <?php } else { ?>
97          <h2 style="text-align: center; background-color: lightgrey; color: red; margin-bottom: 0;">Aucune image n'est associée à ce
98          véhicule</h2>
99      <?php } ?>
    <!-- fermer le carrousel -->
    <button type="button" class="btn btn-outline-danger" data-bs-dismiss="modal" style="margin-top: 10px; margin-bottom: 10px;
        margin-left: auto; margin-right: auto; width: 30%;">Fermer</button>
</div>
</div>

```

## Réalisation des statistiques de GVH

### Contexte

Il m'a été demandé dans un deuxième temps, de concevoir une nouvelle section pour permettre de récupérer et mettre en forme des données en tant que statistiques qui seraient utiles de savoir, tel que les véhicules les plus réservés, l'énergie la plus utilisée...

### Démonstration du code par la statistique « véhicules les plus réservés »

Dans la barre de navigation en haut du site sur Administration => Statistiques , se trouve 4 statistiques différentes (au niveau de la requête SQL et données nécessaires), mais comme ils fonctionnent sensiblement de la même manière, un exemple suffit pour montrer l'essentiel.

Véhicules les plus réservés    Véhicules parcourant le plus de km    Km par énergie    Nb réservation par conducteur

Liste des véhicules les plus réservés

Site(s)    Date de début    Date de fin

Choisissez un ou plusieurs sites    +    01/01/2024    31/12/2024

HISTOGRAMME    SECTEUR    TABLEAU

Dans le fichier « **gestion\_statistiques.php** » on arrive par défaut sur le premier onglet ici présent

```

939 // **** ONGLETS ****
940 // ***** Liste des véhicules les plus réservés ****
941 // ****
942 if ($this_onglet['url_onglet'] == "vehicules_les_plus_reserve" && array_key_exists($this_onglet['id_onglet'], $onglets))
{ ?>

```

J'active une fonctionnalité « Slim select » pour plus d'interaction avec le select pour le formulaire associé.

```

949     <script>
950         $(init); //on charge la fonction init au chargement de la page
951         function init(){
952             //SlimSelect permet la sélection multiple de valeur pour le select
953             new SlimSelect({
954                 select: '#selectVehiculePlusReserve',
955                 placeholder: 'Choisissez un ou plusieurs sites',
956                 searchPlaceholder: "Rechercher un site",
957                 selectByGroup: true, //permet de sélectionner par groupe
958                 closeOnSelect: false, //ne pas fermer la liste après sélection
959                 allowDeselect: true, //permet de désélectionner
960                 allowDeselectOption: true,
961             });
962         }
963     </script>
964

```

Le formulaire ci-dessous (après résumé) est en POST, il a un id unique qui servira pour la fonction de génération du formulaire.

Le select va récupérer les sites de la structure requêté au début du code

```
9 $structure_user = $_SESSION['ident']['id_structure'];
10
11 // Récupération des sites par rapport à la structure de l'utilisateur connecté
12 $sql_structure = "SELECT DISTINCT st.id_structures, st.libelle_structures
13 FROM t_sites s
14 JOIN r_departements de ON de.id_departement = s.id_departement
15 JOIN r_structures st ON st.id_structures = de.id_structures
16 WHERE st.id_structures = $structure_user
17 ORDER BY libelle_structures ASC";
18 $structures_stats = $pdo->query($sql_structure)->fetchAll(PDO::FETCH_ASSOC);
```

il va chercher les sites de la structure correspondant à celle de l'utilisateur.

La date de début et de fin seront également nécessaire.

```

        $sites = $pdo->query($sql_site)-
>fetchAll(PDO::FETCH_ASSOC);
        ?>
        <?php
        foreach($sites as $site)
        { ?>
            <option value="<?=
$site['id_site'] ?>"><?= $site['libelle_site'] ?></option>
            <?php
            }
            ?>
            </optgroup>
        <?php
        }
        ?>
        </select>
    </div>
    <div class='col-md-4' style="width: 200px; margin-
top: 5px; margin-bottom: 5px;">
        <label for="date_debut">Date de début</label>
        <!-- date de début par défaut : 1er jour de
l'année en cours -->
        <input type="text" class="form-control"
id="date_debut" name="date_debut" required pattern="\d{2}/\d{2}/\d{4}"
title="j/mm/aaaa"
            value="<?= date('01/01/Y') ?>">
    </div>
    <div class='col-md-4' style="width: 200px; margin-
top: 5px; margin-bottom: 5px;">
        <label for="date_fin">Date de fin</label>
        <!-- date de fin par défaut : dernier jour de
l'année en cours -->
        <input type="text" class="form-control"
id="date_fin" name="date_fin" required pattern="\d{2}/\d{2}/\d{4}"
title="jj/mm/aaaa"
            value="<?= date('31/12/Y') ?>">
    </div>
    <!-- générer un graphe ou tableau -->
    <div class='col-md-4' style="width: 350px; height:
35px; margin-bottom: 8px; margin-top: auto; padding: 0;">
        <button type="button" class="btn btn-primary"
style="width: 140px; margin-right:5px;"
onclick="modalVehiculePlusReserve('bar')">HISTOGRAMME</button>
        <button type="button" class="btn btn-primary"
style="width: 90px; margin-right:5px;;
onclick="modalVehiculePlusReserve('pie')">SECTEUR</button>
        <button type="button" class="btn btn-primary"
style="width: 90px; margin-right:5px;;
onclick="modalVehiculePlusReserve('tableau')">TABLEAU</button>

```

```

        </div>
    </div>
</form>
</div>
</div>
</div>

<!-- Fin formulaire -->
```

Une fois le formulaire rempli, un onclick menant à la même fonction sera pour chacun des 3 bouton pour choisir le type d'affichage des statistiques pour au choix : en **HISTOGRAMME (bar)** , **SECTEUR (pie)** ou **TABLEAU (tableau)**.

Les donnée entrée en tant que exemple :

### Liste des véhicules les plus réservés

Site(s)		Date de début	Date de fin					
<input type="checkbox"/> Antenne Cher	<input type="checkbox"/> Antenne Eure-et-Loire	<input type="checkbox"/> Antenne Indre	<input type="checkbox"/> Antenne Indre-ét-Loire	<input type="checkbox"/> Antenne Loir-et-Cher	<input type="checkbox"/> Antenne Loiret	<input type="checkbox"/>	01/01/2010	31/12/2024
		<input type="button" value="HISTOGRAMME"/>	<input type="button" value="SECTEUR"/>	<input type="button" value="TABLEAU"/>				

Cette statistique a comme fonction « **modalVehiculePlusReserve** » elle va récupérer le **type de statistique, la date de début, la date de fin et la sélection d'un ou plusieurs sites du select**

```

55 // Modal pour vehicules_les_plus_reserve
56 function modalVehiculePlusReserve(type_stat){
57     //récupérer les valeurs du formulaire associé
58     date_debut = $('#date_debut').val();
59     date_fin = $('#date_fin').val();
60     select = $('#selectVehiculePlusReserve').val();
```

On va vérifier que les éléments sont bien remplis et que la date est bonne

```

67 if (select != "" && date_debut != "" && date_fin != "") {//si tout les champs sont remplis
68     if (verifDateFin_Superieur_DateDebut(date_debut, date_fin)) {// si la date de fin est inférieure à la date de début
69         $('#formVehiculePlusReserve').find('button').attr('disabled', true); //on désactive le bouton pour éviter le spam
70         $('.telechargergraphe').hide(); $('.telechargergraphe').attr('disabled', true); //on cache le bouton pour télécharger l'image au cas où il serait affiché
71         $('.spinner-border').show(); //on affiche le spinner de chargement
72
73     }
74
75     $.ajax({//---ajax---
76         type: 'get',
77         url: 'ajax_modal_statistique_vehiculePlusReserve.php',
78         data: {
79             date_debut: date_debut,
80             date_fin: date_fin,
81             selectSite: select,
82             type_stat: type_stat
83         },
84     });
85 }
```

Fonction de vérification de la date du fichier « **javascript.js** »

```

2500: ~ function verifDateFin_Superieur_DateDebut(dateDebut, dateFin) {
2501:
2502:     //on concatène les dates pour les comparer
2503:     dateDebut = dateDebut.split("/");//on sépare la date en jour mois et année
2504:     dateDebut = dateDebut[2] + dateDebut[1] + dateDebut[0];//on inverse l'ordre pour avoir l'année en premier
2505:     dateDebut = parseInt(dateDebut);//on transforme la date en nombre
2506:     // console.log(dateDebut);
2507:
2508:     dateFin = dateFin.split("/");
2509:     dateFin = dateFin[2] + dateFin[1] + dateFin[0];
2510:     dateFin = parseInt(dateFin);
2511:     // console.log(dateFin);
2512:
2513:     //le format sera aaaa/mm/jj , on peut donc comparer les date par leur grandeur en nombre directement
2514:     //si la date de fin est inférieure à la date de début
2515:     if (dateFin < dateDebut) {
2516:         return false;
2517:     }
2518:     else {
2519:         //valide les dates
2520:         return true;
2521:     }
2522: }

```

On récupère les données nécessaire et les envoi vers le fichier modal

« ajax\_modal\_statistique\_vehiculePlusReserve.php » en ajax.

```

13  //si les données sont présentes dans le post, on les récupère pour faire la requête pour le graphe
14  if (getpost('date_debut') && getpost('date_fin') && getpost('selectSite') && getpost('type_stat')) {
15      //mettre les dates au format anglais pour la requête
16      $date_debut = fr_date_to_eng(getpost('date_debut'));
17      $date_fin = fr_date_to_eng(getpost('date_fin'));
18
19      //var_dump(getpost('selectSite'));//debug
20      $sites_select = implode(", ", getpost('selectSite'));//on récupère le ou les sites sélectionnés et on
21      //les sépare par ','
22
23      //type de statistique
24      $type_stat = getpost('type_stat');

```

On vérifie que les données sont récupérable.

On effectue ensuite la requête souhaité (ici pour les véhicule les plus réservé)

```

60  $sql_vehicules_plus_reserves = "SELECT tmt.libelle_moyen_transport, tmt.immatriculation_vehicule, tmt.
61  date_mise_circulation, COUNT(tr.id_reservation) as nb_utilisations,
62  ts.libelle_site, compteur_en_cours, libelle_marque
63  FROM t_reservations tr
64  JOIN t_moyens_transports tmt ON (tmt.id_moyen_transport = tr.id_moyen_transport)
65  JOIN t_sites ts ON (ts.id_site = tmt.id_site)
66  JOIN r_marques ON (r_marques.id_marque = tmt.id_marque)
67  WHERE tr.debut_mission_reservation >= '$date_debut' AND
68  tr.fin_mission_reservation <= '$date_fin'
69  AND tmt.id_site IN ($sites_select)
70  GROUP BY tmt.libelle_moyen_transport, ts.libelle_site, tmt.immatriculation_vehicule
71  ORDER BY nb_utilisations desc";
72
73
74  $element_vehicules_plus_reserves = $pdo->query($sql_vehicules_plus_reserves);

```

Il y a 2 issues possible, si les données vont être utilisé en tant que **graphe** type **Histogramme**, **Secteur**, on les renvoi avec du json.

```

76     //si le type de statistique est un graphe compatible
77     if ($type_stat == "bar" || $type_stat == "pie") {
78         //on renvoie les données en json
79         echo json_encode($element_vehicules_plus_reserves->fetchAll(PDO::FETCH_ASSOC));
80     }

```

Sinon on les dispose dans un **Tableau** en bootstrap.

```

81     //si le type de statistique est un tableau
82     elseif ($type_stat == "tableau") {
83         //on renvoie les données sous forme de bootstrap table
84     ?>
85         <table class='table table-striped table-bordered table-hover table-condensed' id="tableStat"
86             data-group-by="true"
87             data-group-by-toggle="true"
88             data-group-by-field="libelle_site"
89             data-order-by="nb_utilisations"
90             data-search="true"
91             data-search-align="left"
92             data-pagination="true"
93             data-page-list="[10, 25, 50, 100, ALL]"
94             data-page-size="25"
95             data-show-export="true"
96             data-export-types="['excel', 'csv']">
97             <thead>
98                 <tr>
99                     <th data-visible="false" title="Site" data-field="libelle_site"></th>
100                    <th>Marque</th>
101                    <th>Véhicule</th>
102                    <th data-width="5" data-width-unit "%">Nombre<br>d'utilisations</th>
103                    <th data-width="2" data-width-unit "%">Total %</th>
104                    <th data-width="5" data-width-unit "%">Compteur<br>en cours</th>
105                    <th data-width="5" data-width-unit "%">Âge du<br>véhicule</th>
106                </tr>
107            </thead>

```

Pour obtenir les pourcentages dans le tableau par rapport à chaque site de la **valeur référent** (ici le **nombre d'utilisation**), on créer un tableau pour récupérer les valeur et faire le total pour chaque site. Ensuite faire le calcule par rapport au totaux du site (auquel la valeur est référé) dans un foreach pour chacune d'entre elle.

```

110             // tableau pour stocker les totaux des nombres d'utilisations par site
111             $total_utilisations_par_site = [];
112
113             // calcule les totaux de valeur par site
114             foreach ($element_vehicules_plus_reserves as $row) {
115                 $site = $row['libelle_site'];//on récupère le site de la ligne
116                 if (!isset($total_utilisations_par_site[$site])) { //si le site n'est pas dans le tableau,
117                     on l'ajoute
118                     $total_utilisations_par_site[$site] = 0;//on initialise le total à 0 pour le nouveau site
119                 }
120                 $total_utilisations_par_site[$row['libelle_site']] += $row['nb_utilisations'];//on ajoute
121                 des valeur sur le site de la ligne
122             }
123             $element_vehicules_plus_reserves->execute(); //remettre le curseur au début

```

```

123     foreach ($element_vehicules_plus_reserves as $row) {
124         ?>
125         <tr>
126             <td style="text-align: center;"><?= $row['libelle_site'] ?></td>
127             <td style="text-align: left;"><?= $row['libelle_marque'] ?></td>
128             <td style="text-align: left;"><?= $row['libelle_moyen_transport'] ?> (<?= $row
129                 ['immatriculation_vehicule'] ?>)</td>
130             <td style="text-align: right; font-weight: bold;"><?= nombreAvecFormat($row
131                 ['nb_utilisations']) ?></td>
132             <td style="text-align: right;"><?php echo round(($row['nb_utilisations'] /
133                 $total_utilisations_par_site[$row['libelle_site']]) * 100, 2) ?>%</td>
134             <td style="text-align: right;"><?= nombreAvecFormat($row['compteur_en_cours']) ?></td>

```

Calcule de l'âge du véhicule.

```

132         <!-- calcule de l'age du véhicule par rapport a la date de mise en circulation -->
133         <?php
134             //aura le nombre d'année du véhicule par la différence de date entre la date
135             //actuelle et la date de mise en circulation
136             $age_vehicule = date_diff(date_create($row['date_mise_circulation']), date_create
137             (date('Y-m-d')));
138             //si c'est 1 an ou plus, on affiche le nombre d'années
139             if ($age_vehicule->y >= 1) {
140                 echo "<td style='text-align: right;'>".$age_vehicule->y." ans</td>";
141             }
142             //si c'est un mois ou plus mais moins d'un an, on affiche le nombre de mois
143             elseif ($age_vehicule->m >= 1) {
144                 echo "<td style='text-align: right;'>".$age_vehicule->m." mois</td>";
145             }
146             //sinon, on affiche le nombre de jours
147             else {
148                 echo "<td style='text-align: right;'>".$age_vehicule->d." jours</td>";
149             }
150         ?>
151     </tr>

```

Lors du success on va récupérer les valeurs en fonction du type de stat voulu.

```

82             //récupère le contenu de ajax_modal_statistique_vehiculePlusReserve.php
83             success: function(dataHtml)
84             {
85                 //on supprime les messages d'erreur de la classe alert pour pas avoir de doublons
86                 $('.alert').remove();
87                 //on vide les conteneurs de stats par défaut au cas où il sont remplis
88                 $('#grapheVehiculesPlusReserve').remove();
89                 $('#tableauVehiculesPlusReserve').html("");

```

Si on veut un **graphe**, on va utiliser l'outil « **Chart.js** » pour les générer, les 2 type de graphe n'ont en revanche pas la même syntaxe.

On récupère les données json du datahtml.

```

91             //SI LE TYPE DE STATISTIQUE EST UN GRAPHE
92             if (type_stat == "bar" || type_stat == "pie") {
93                 datas = JSON.parse(dataHtml)//réponse de la requête ajax en JSON
94                 //console.log(datas);//affichage de l'ajax
95
96                 //si les données ne sont pas vides, on génère le graphe
97                 if(datas.length > 0){

```

Pour ces graphe il y a une fonction en cliquant sur un bouton pour télécharger le graphe

[Télécharger le graphe](#)

```

42 //fonction pour télécharger l'image du graphe
43 function telechargerGraphe(canvas_id){
44     //on récupère l'image du graphe dans le canvas
45     var canvas = document.getElementById(canvas_id);
46     var image = canvas.toDataURL("image/png", 1.0); //format png et qualité par défaut
47     //on crée un lien pour télécharger l'image
48     var link = document.createElement('a'); //création d'un lien
49     //le nom du fichier sera le nom de l'id du canvas + période sélectionnée + ..png
50     link.download = canvas_id+'_du_+'+$('#date_debut').val()+'_au_+'+$('#date_fin').val()+'_.png';
51     link.href = image; //on met l'image dans le lien
52     link.click(); //cela va déclencher le téléchargement
53 }

```

## Histogramme

Si on veut un **Histogramme**, on crée le canvas pour le graphe et on met les données souhaité dans un tableau.

```

98
99     if (type_stat == "bar"){
100         $('#corps').append('<canvas id="grapheVehiculesPlusReserve" style="max-height: 50vh; display="none"></canvas>');// plus la largeur est grande, plus la hauteur est petite
101
102         //Génération du graphe
103         var ctx = document.getElementById('grapheVehiculesPlusReserve').getContext('2d');//prend l'id du canvas pour écrire dedans
104
105         // Déclaration d'un tableau
106         datasets = [];
107         // On boucle sur les données récupérées pour les placer dans le tableau
108         datas.forEach((elem) => {
109             // push => envoyer les données dans le tableau pour générer un graphe
110             datasets.push({//label = nom du véhicule (plaques d'immatriculation)
111                 label: elem.libelle_moyen_transport + ' (' + elem.immatriculation_vehicule + ')',
112                 data: [elem.nb_utilisations]
113             })
114         })

```

Mise en œuvre des valeurs selon la syntaxe pour un **Histogramme**.

```
//création du graphe qui va être dans la période des dates sélectionnées
avec une ligne pour chaque véhicule
var myChart = new Chart(ctx, {
    type: 'bar',
    data: {
        labels: [
            'Sur la période : ' + date_debut + ' - ' + date_fin
        ],
        datasets: datasets
    },
    plugins: [ChartDataLabels],//utilise le plugin pour le dataLabels
    options: {
        scales: {
            y: {
                beginAtZero: true,
                min: 0
            }
        },
        //titre du graphe
        plugins: {
            title: {
                display: true,
                text: 'Véhicules les plus réservés'
            },
            legend: {
                display: true,
                position: 'top'
            },
            //affichage des valeurs dans chaque barre
            datalabels: {
                anchor: 'end',
                align: 'start',
                color: 'black',//couleur noir
                font: {
                    weight: 'bold'//gras
                },
            }
        }
    }
})
```

```

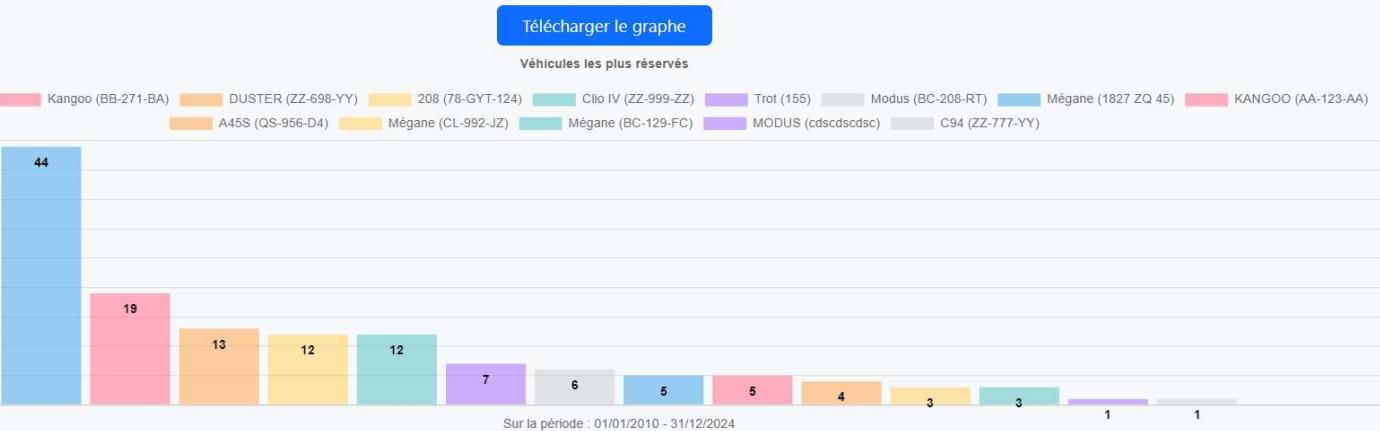
        formatter: function(value, context) {
            return nombreAvecVirgule(value, 0, '');//espace pour
            séparer les ","
        }
    }
});
```

Et on affiche le canvas avec la possibilité de le télécharger

```
//on affiche le graphe
$('#grapheVehiculesPlusReserve').show();

//on affiche le bouton pour télécharger l'image
$('.telechargergraphe').show(); $('.telechargergraphe').attr('disabled', false);
```

Résultat :



## Secteur

Si on veut un **Secteur**, on crée le canvas pour le graphe et on met les données souhaité dans un tableau.

```

157
158 } else if (type_stat == "pie"){
159     $('#corps').append('<canvas id="grapheVehiculesPlusReserve"
160                         style="max-height: 80vh;" display="none"></canvas>');//on rajoute le canvas
161                         pour le graphes
162     var ctx = $('#grapheVehiculesPlusReserve').get(0).getContext('2d');//on
163                         récupère le contexte du canvas
164
165     // création des tableau de chaque type de données
166     var labels = [];//nom
167     var data = [];//valeur
168
169     datas.forEach(element) => {
170         labels.push(element.libelle_moyen_transport + ' (' + element.
171                     immatriculation_vehicule + ')');//nom
172         data.push(element.nb_utilisations);//valeur
173     });
174 }
```

Mise en œuvre des valeurs selon la syntaxe pour un **Secteur**.

```

//on met en place le graphe
var data = {
    labels: labels,//labels sera pour les noms des valeurs
    datasets: [{//datasets sera pour les valeurs
        label: 'Nombre d\'utilisations',//nom du dataset
        data: data
    }]
};

// console.log(data);
var myPieChart = new Chart(ctx, {
    type: type_stat,//type de graphe
    data: data,//place les valeurs
    plugins: [ChartDataLabels],//utilise le plugin pour le datalabels
    options: {
        responsive: true,
        plugins: {
            title: {
                display: true,
                text: 'Véhicules les plus réservés sur la période : ' +
                date_debut + ' - ' + date_fin
            },
            legend: {
                display: true,
                position: 'top'
            },
            //affichage des valeurs dans chaque barre
            datalabels: {
                anchor: 'end',
                align: 'start',
                color: 'black',//couleur noir
                font: {
                    weight: 'bold'//gras
                },
                formatter: function(value, context) {
                    return nombreAvecVirgule(value, 0, '');//espace pour
                    séparer les milliers
                }
            }
        }
    }
});

```

Et on affiche le canvas avec la possibilité de le télécharger

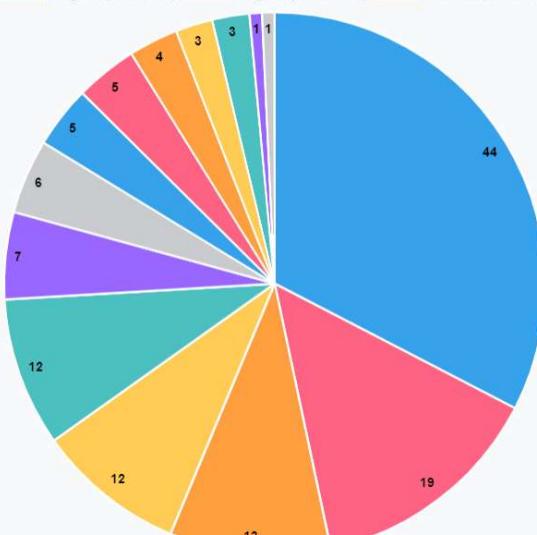
```
//on affiche le graphe
$('#grapheVehiculesPlusReserve').show();

//on affiche le bouton pour télécharger l'image
$('.telechargergraphe').show(); $('.telechargergraphe').attr('disabled', false);
```

Résultat :

[Télécharger le graphe](#)

Véhicules les plus réservés sur la période : 01/01/2010 - 31/12/2024



## Tableau

Si on veut un **Tableau**, on récupère le tableau déjà généré de « `ajax_modal_statistique_vehiculePlusReserve.php` » par le `datahtml`.

```
224
225
226
239
240
else {
    //SI LE TYPE DE STATISTIQUE EST UN TABLEAU
    if (type_stat == "tableau") {
        $('#tableauVehiculesPlusReserve').html(dataHtml); //récupère le tableau de
        données générée par la requête ajax
        $('#tableStat').bootstrapTable(); //on applique le style bootstrap au tableau
```

Résultat :

Recherche						
Marque	Véhicule	Nombre d'utilisations	Total %	Compteur en cours	Âge du véhicule	
<b>Antenne Cher</b>						
RENAULT	Mégane (BC-129-FC)	3	100%	549 876	6 mois	
<b>Antenne Eure-et-Loire</b>						
CITROEN	C94 (ZZ-777-YY)	1	100%	55	6 ans	
<b>Antenne Indre</b>						
RENAULT	Mégane (1827 ZQ 45)	5	62.5%	3 001	8 ans	
RENAULT	Mégane (CL-992-JZ)	3	37.5%	445 854	8 ans	
<b>Antenne Indre-et-Loire</b>						
RENAULT	Modus (BC-208-RT)	6	100%	54 652	6 ans	
<b>Antenne Loiret</b>						

(Les données sont exportable en csv et tableau excel)

## Données Notable

## Concepteur des données de phpMyAdmin

