

```
1  <!DOCTYPE html>
2  <html>
3  <?php
4  /*
5   * Author : Lucien Camuglia
6   * Description : index page
7   * Date : April-june 2017
8   * Version : 1.0 LC BaseVersion
9   */
10 include './Includes/header.php'
11 ?>
12
13 <body>
14     <?php
15     DisplayMenu(basename($_SERVER['PHP_SELF']));
16     ?>
17     <?php include './Includes/footer.php'; ?>
18
19     <div class="col-lg-6 col-lg-offset-3">
20     <p>The website was made for my Technician diploma, it allows sharing
21     itineraries between motorcycle enthusiasts. The site is set-up with
22     HTML/PHP/CSS/AJAX and Google's API. It allows users to create trips, to
23     share and modify them.</p>
24
25     <p>Different criteria can be defined by the user to create the itinerary:</p>
26     <ul>
27         <li>Travel Time </li>
28         <li>Type of road </li>
29         <li>Change in altitude, etc.</li>
30     </ul>
31
32     <p>This website gives the user diverse informations. One example could be
33     the theoretical consumption of their motorcycle for the ride or travel time.
34     The rider can also upload or download files directly from his GPS.</p>
35     </div>
36 </body>
37 </html>
```

```

1  <!DOCTYPE html>
2  <?php
3  /*
4   * Author : Lucien Camuglia
5   * Description : Page with all itineraries
6   * Date : April-june 2017
7   * Version : 1.0 LC BaseVersion
8   */?>
9  <html>
10     <head>
11         <?php include './Includes/header.php' ?>
12         <title></title>
13     </head>
14     <body style="height: 500px;" >
15         <?php
16         DisplayMenu(basename($_SERVER['PHP_SELF']));
17         ?>
18         <script src="./JS/Map.js"></script>
19         <script>
20
21         </script>
22         <div class="col-sm-1 "><h2>Filters :</h2>
23             <h6 class="col-xs-9">Enabled</h6> <input class="col-sm-3
24             checkbox-inline" type="checkbox" onclick="EnabledDisabledFilters(this)"/>
25             <div id="filters" hidden="true">
26                 <p><label>Sinuosity</label><input id="sinuosity" type="range"
27                 min="<?php echo GetLessSinuousRoad(); ?>" max="<?php echo
28                 GetMostSinuousRoad() + 1; ?>" /></p>
29                 <p><label>Slope</label><input id="slope" type="range" min="<?php
30                 echo GetLessSteepestRoad(); ?>" max="<?php echo
31                 GetMostSteepestRoad(); ?>" /></p>
32                 <p><label>Highway </label><input id="highway" class="col-sm-3
33                 checkbox-inline" type="checkbox" /></p>
34
35             </div>
36         </div>
37         <div class="routes col-lg-2 " style="height: 110%">
38             <?php
39             $routes = GetRoutes();
40             foreach ($routes as $route) {
41                 echo"<div class=\"route\" name=\"\" . $route["idRoute"] . "\">\" .
42                 $route["RouteName"] . "<span class=\"by\"> by \" . $route["Username"]
43                 . "</span></div>";
44             }
45             ?>
46
47         </div>
48         <input id="pac-input" class="controls" type="text" placeholder="Search Box"
49         hidden>
50         <div id="map" class="col-lg-8" style="height: 110%;"></div>
51         <div><button class="btn btn-primary" onclick="DownloadRoute();" >Download
52         route</button></div>
53
54         <div class="col-lg-5 col-lg-offset-4 routeControl hidden " >
55             <center><h2>Road Infos</h2></center>
56             <h4 class="col-lg-12" id="InfoHighway">This road does not includes
57             highways</h4>

```

```
47      <p><span style="text-align: left;" class="col-lg-6"
      id="InfoLength">Length : 12km</span><span style="text-align: right;"
      class="col-lg-6" id="InfoDuration">Duration : 0h30min</span></p>
48      <p><span style="text-align: left;" class="col-lg-6"
      id="InfoSinuosity">Sinuosity : 0/10</span><span style="text-align:
      right;" class="col-lg-6" id="InfoSlope">Slope : 0/10</span></p>
49      <p><span style="text-align: left;" class="col-lg-8"
      id="InfoConsumption">Consumption : 52 liters</span><span
      style="text-align: right;" class="col-lg-6"></span></p>
50
51      </div>
52      <?php include './Includes/footer.php'; ?>
53  </body>
54  </html>
55
```

```
1  <!DOCTYPE html>
2  <?php
3  /*
4   * Author : Lucien Camuglia
5   * Description : User page for own itineraries
6   * Date : April-june 2017
7   * Version : 1.0 LC BaseVersion
8   */
9  ?>
10 <html>
11     <head>
12         <?php include './Includes/header.php'; ?>
13         <title></title>
14     </head>
15     <body style="height: 500px;" >
16         <?php
17             DisplayMenu(basename($_SERVER['PHP_SELF']));
18             importerTrajetModal();
19             newRouteModal();
20             if (isset($_GET["NewIti"])) {
21                 $idRoute = $_GET["id"];
22             }
23             ?>
24             <script src="./JS/Map.js"></script>
25             <script>
26 <?php
27 if (isset($idRoute)) {
28     ?>
29         var Points = LoadPoints(<?php echo $idRoute ?>);
30         console.log(Points);
31         Points = SnappPoints2Road(Points);
32         SaveNewLocation(<?php echo $idRoute ?>, Points);
33     <?php
34 }
35 ?>
36     </script>
37     <div class=" routes col-lg-offset-1 col-lg-2" style="height: 500px;">
38
39         <?php
40             $routes = GetRoutes($_SESSION["id"]);
41             foreach ($routes as $route) {
42                 echo"<div class=\"route\" name=\"\" . $route["idRoute"] . "\">\" .
43                     $route["RouteName"] . "</div>";
44             }
45             ?>
46
47     </div>
48     <input id="pac-input" class="controls" type="text" placeholder="Search Box"
49     hidden/>
50     <div id="map" class="col-lg-7" style="height: 500px;"></div>
51
52     <div class="btnZone col-lg-2 top-buffer" >
53         <!-- <button id="btnModif" class="hidden btn btn-primary btn-block"
54             name="none">Modifier le trajet</button>-->
55
56         <label class="checkbox-inline SaveRouteControls hidden">
```

```

55         <input id="chkHighway" type="checkbox" checked data-toggle="toggle">
           Highways
56     </label>
57
58     <button id="SaveRoute" class=" btn btn-primary btn-block
SaveRouteControls hidden" data-target="#newRoutemodal"
data-toggle="modal">Save the route</button>
59     <button id="ClearRoute" class=" btn btn-primary btn-block
SaveRouteControls hidden">Clear the route</button>
60
61     <button class="btn btn-block" id="StartCreation">Créer un nouveau
trajet</button>
62     <button class="btn btn-block" data-toggle="modal"
data-target="#modalimport" onclick='$("#msgCont").attr("Hidden",
"true")' >Importer un nouveau trajet</button>
63
64     <button id="RefreshRoute" class=" btn btn-primary btn-block hidden
modifRoute">Refresh the route</button>
65     <button id="DeleteRoute" class=" btn btn-danger btn-block hidden
routeControl">Delete the route</button>
66
67 </div>
68
69 <div id="pointsInfo" class="col-lg-6 col-lg-offset-3 hidden modifRoute" >
70
71     <div id="pointStart">
72         <div class="Pointname col-lg-12" >
73             <span class="col-lg-2 align-middle" style="height: 100px;"
>Start : </span>
74             <div class="col-lg-10" >
75                 <div>
76                     <label class="col-sm-3">Latitude : </label> <input
id="inpLatPoint0" type="text" value="0.000">
77                 </div>
78                 <div>
79                     <label class="col-sm-3">Longitude : </label> <input
id="inpLonPoint0" type="text" value="0.000">
80                 </div>
81             </div>
82         </div>
83
84     </div>
85
86     <div id="pointEnd">
87         <div class="Pointname col-lg-12" >
88             <span class="col-lg-2 align-middle" style="height: 100%;" >End :
</span>
89             <div class="col-lg-10" >
90                 <div>
91                     <label class="col-sm-3">Latitude : </label> <input
id="inpLatPoint1" type="text" value="0.000">
92                 </div>
93                 <div>
94                     <label class="col-sm-3">Longitude : </label> <input
id="inpLonPoint1" type="text" value="0.000">
95                 </div>
96             </div>

```

```
97         </div>
98     </div>
99
100 </div>
101
102 <div class="modalLoad"></div>
103
104 <div class="col-lg-5 col-lg-offset-4 routeControl hidden" >
105     <center><h2>Road Infos</h2></center>
106     <h4 class="col-lg-12" id="InfoHighway">This road does not includes
107     highways</h4>
108     <p><span style="text-align: left;" class="col-lg-6"
109     id="InfoLength">Length : 12km</span><span style="text-align: right;"
110     class="col-lg-6" id="InfoDuration">Duration : 0h30min</span></p>
111     <p><span style="text-align: left;" class="col-lg-6"
112     id="InfoSinuosity">Sinuosity : 0/10</span><span style="text-align:
113     right;" class="col-lg-6" id="InfoSlope">Slope : 0/10</span></p>
114     <p><span style="text-align: left;" class="col-lg-8"
115     id="InfoConsumption">Consumption : 52 liters</span><span
116     style="text-align: right;" class="col-lg-6"></span></p>
117
118 </div>
119 <?php include './Includes/footer.php'; ?>
120 </body>
121 </html>
```

```

1  <!DOCTYPE html>
2  <?php
3  /*
4   * Author : Lucien Camuglia
5   * Description : Managing users page
6   * Date : April-june 2017
7   * Version : 1.0 LC BaseVersion
8   */
9  ?>
10
11 <html>
12     <head>
13         <?php include './Includes/header.php' ?>
14         <script src="./JS/ManageUsers.js"></script>
15         <title></title>
16     </head>
17     <body>
18         <?php
19         DisplayMenu(basename($_SERVER['PHP_SELF']));
20         ?>
21
22         <table class="table table-striped table-bordered table-responsive"
23         cellpadding="0" id="">
24             <thead>
25                 <tr>
26                     <th rowspan="2" class="text-center valign">Username</th>
27                     <th rowspan="2" class="text-center valign">E-Mail</th>
28                     <th colspan="4" class="text-center valign">Role</th>
29                     <th colspan="3" class="text-center valign">Motorcycle</th>
30                     <th rowspan="2" class="text-center valign">Delete</th>
31                 </tr>
32                 <tr>
33                     <th class="text-center valign">Designation</th>
34                     <th colspan="3" class="text-center valign">Action</th>
35                     <th class="text-center valign">Brand</th>
36                     <th class="text-center valign">Model</th>
37                     <th class="text-center valign">Year</th>
38                 </tr>
39             </thead>
40             <tbody id="">
41                 <?php
42                 $users = getUsersNMotorcycle();
43                 foreach ($users as $user) {
44                     echo "<tr>";
45                     echo "<td>" . $user["Username"] . "</td>";
46                     echo "<td>" . $user["email"] . "</td>";
47                     $rank = $user["role"];
48                     echo "<td > <span id=\"" . $user["idUser"] . "Role\"
49                     name=\"$rank\">";
50
51                     switch ($rank) {
52                         case 1 : echo "Administrator";
53                             break;
54                         case 2 : echo "User";
55                             break;

```

```
56         case 3 : echo "Ban";
57         break;
58     }
59     echo "</span></td>";
60
61     echo "<td class=\"text-center no-border action-table ";
62     if ($rank == 1) {
63         echo "disabled";
64     }echo "\" style=\"width: 60px;\"><a name=\"\" . $user["idUser"] .
        \"\" class=\"UpgradeRole\"><span class=\" glyphicon
        glyphicon-chevron-up \" > upgrade</span></a> </td>";
65
66     echo "<td class=\"text-center no-border action-table ";
67     if ($rank != 1) {
68         echo "disabled";
69     }echo "\" style=\"width: 80px;\"><a name=\"\" . $user["idUser"] .
        \"\" class=\"DowngradeRole\"><span class=\" glyphicon
        glyphicon-chevron-down \" > downgrade</span></a></td>";
70     echo "<td class=\"text-center no-border action-table ";
71     if ($rank == 3) {
72         echo "disabled";
73     }echo " \" style=\"width: 50px;\" ><a name=\"\" . $user["idUser"]
        . \"\" class=\"BanRole\"><span class=\" glyphicon
        glyphicon-ban-circle \" > ban</span></a></td>";
74     echo "<td>\" . $user["Brand"] . "</td>";
75     echo "<td>\" . $user["model"] . "</td>";
76     echo "<td>\" . substr($user["year"], 0, 4) . "</td>";
77     echo "<td> <span class=\"glyphicon glyphicon-remove red\"
        ></span></td>";
78     }
79     ?>
80
81     </tbody>
82     <tfoot>
83
84     </tfoot>
85
86     </table>
87     <?php include './Includes/footer.php'; ?>
88 </body>
89 </html>
90
```



```

1  <!DOCTYPE html>
2  <?php
3  /*
4   * Author : Lucien Camuglia
5   * Description : managing motorcycle page
6   * Date : April-june 2017
7   * Version : 1.0 LC BaseVersion
8   */
9  ?>
10 <html>
11     <head>
12         <?php include './Includes/header.php' ?>
13         <script src="./JS/ManageMoto.js"></script>
14         <title></title>
15     </head>
16     <body>
17         <?php
18         DisplayMenu(basename($_SERVER['PHP_SELF']));
19         AddMotorcycleModal();
20         $motorcycles = GetAllMotorcycles();
21         foreach ($motorcycles as $motorcycle) {
22             confirmationDeleteMotorcycleModal($motorcycle["idMoto"],
23             $motorcycle["Brand"], $motorcycle["model"], substr($motorcycle["year"],
24             0, 4));
25         }
26         ?>
27
28         <table class="table table-striped table-bordered" cellspacing="0"
29         id="MotorcycleTable">
30             <thead>
31                 <tr>
32                     <th>Brand <select id="brand" ><option value="%">All</option> <?php
33                     $brands = GetMotorcycleBrand();
34                     foreach ($brands as $brand) {
35                         echo "<option value='" . $brand . "'>" . $brand .
36                         "</option>";
37                     }
38                     ?></select> </th>
39                     <th>Model <select id="model"><option value="%">All</option><?php
40                     $models = GetAllMotorcycleModel();
41                     foreach ($models as $model) {
42                         echo "<option value='" . $model . "'>" . $model .
43                         "</option>";
44                     }
45                     ?></select></th>
46                     <th>Year <select id="year"><option value="%">All</option><?php
47                     $years = GetAllMotorcycleYear();
48                     foreach ($years as $year) {
49                         echo "<option value='" . $year . "'>" . $year .
50                         "</option>";
51                     }
52                     ?></select></th>
53                     <th>Consumption (l/100km) <select id="consumption" ><option
54                     value="%">All</option><?php
55                     $consumptions = GetAllMotorcycleConsumption();
56                     foreach ($consumptions as $consumption) {
57                         echo "<option value='" . $consumption . "'>" .

```

```

        $consumption . "</option>";
51     }
52     ?></select></th>
53     <th>Tiredness <select id="tiredness"><option
value="%>All</option><?php
54         $tirednesses = GetAllMotorcycleTiredness();
55         foreach ($tirednesses as $tiredness) {
56             echo "<option value='" . $tiredness . "'>" .
                $tiredness . "</option>";
57         }
58     ?></select></th>
59     <th>Delete</th>
60 </tr>
61
62 </thead>
63 <tbody id="motorcyclesDatas">
64     <?php
65     foreach ($motorcycles as $motorcycle) {
66         echo "<tr>";
67         echo "<td>";
68         echo $motorcycle["Brand"];
69         echo "</td>";
70         echo "<td>";
71         echo $motorcycle["model"];
72         echo "</td>";
73         echo "<td>";
74         echo substr($motorcycle["year"], 0, 4);
75         echo "</td>";
76         echo "<td>";
77         echo $motorcycle["consumption"];
78         echo "</td>";
79         echo "<td>";
80         echo $motorcycle["Tiredness"];
81         echo "</td>";
82         echo "<td>";
83         echo "<span class='\"glyphicon glyphicon-remove red\"
data-toggle='\"modal\"' data-target='\"#modaldelete\" .
        $motorcycle[\"idMoto\"] . \"\" ></span>";
84         echo "</td>";
85         echo "</tr>";
86     }
87     ?>
88 </tbody>
89 <tfoot>
90     <tr><td colspan="6" ><a href="#" data-toggle="modal"
data-target="#modalMotorcycle" onclick='$("#msgCont").attr("Hidden",
"true")'> + Add a Motorcycle</a></td></tr>
91 </tfoot>
92
93 </table>
94 <?php include './Includes/footer.php'; ?>
95 </body>
96 </html>
97

```

```
1  <!DOCTYPE html>
2  <?php
3  /*
4   * Author : Lucien Camuglia
5   * Description : Signin page
6   * Date : April-june 2017
7   * Version : 1.0 LC BaseVersion
8   */
9  $error = false;
10 $errorMsg = "";
11 if (isset($_GET["error"])) {
12     if (!empty($_GET["error"])) {
13         $error = true;
14         $errorMsg = $_GET["error"];
15     }
16 }
17 $value = false;
18 if (isset($_GET["values"])) {
19     if (!empty($_GET["values"])) {
20         $value = true;
21         $values = json_decode($_GET["values"]);
22     }
23 }
24 ?>
25
26 <html>
27     <head>
28         <?php include './Includes/header.php' ?>
29         <script src="./JS/Signin.js"></script>
30         <title></title>
31     </head>
32     <body style="width: auto;">
33         <?php
34         DisplayMenu(basename($_SERVER['PHP_SELF']));
35         ?>
36         <div class="col-sm-6 col-lg-offset-3 ">
37             <div class="alert alert-danger" <?php echo ($error) ? "" : "hidden"; ?>>
38                 <strong>Erreur !</strong> <label id='msg'><?php echo $errorMsg;
39                 ?></label>
40             </div>
41             <center>
42                 <h1>Sign in</h1>
43                 <form action="./Includes/signIn.php" method="post">
44                     <div class="form-group row">
45                         <label for="Username" class="col-sm-3 col-form-label"
46                         >Username <span data-toggle="tooltip" title="min 5 char"
47                         class="glyphicon glyphicon-info-sign"
48                         id="InfoUsername"></span> </label>
49                         <div class="col-sm-8">
50                             <input class="form-control" type="text" name="username"
51                             id="Username" value="<?php
52                             if ($value) {
53                                 if (isset($values->username)) {
54                                     echo $values->username;
55                                 }
56                             }
57                             ?>">
```

```
53         </div>
54
55         <span class="glyphicon glyphicon-remove red"
56             id="LogoUsername"></span>
57     </div>
58     <div class="form-group row">
59         <label for="Username" class="col-sm-3
60             col-form-label">email</label>
61         <div class="col-sm-8">
62             <input class="form-control" type="email" name="email"
63                 id="email" value="<?php
64                 if ($value) {
65                     if (isset($values->email)) {
66                         echo $values->email;
67                     }
68                 }
69                 ?>">
70         </div>
71         <span class="glyphicon glyphicon-remove red"
72             id="LogoEmail"></span>
73     </div>
74     <div class="form-group row">
75         <label for="Username" class="col-sm-3
76             col-form-label">password <span data-toggle="tooltip"
77                 title="8char min, lupper, llower, lNumeric" class="glyphicon
78                 glyphicon-info-sign" id="InfoPwd"></span> </label>
79         <div class="col-sm-8">
80             <input class="form-control" type="password" value=""
81                 name="password" id="password">
82         </div>
83
84         <span class="glyphicon glyphicon-remove red"
85             id="Logopwd"></span>
86     </div>
87     <div class="form-group row">
88         <label for="Username" class="col-sm-3
89             col-form-label">password confirmation</label>
90         <div class="col-sm-8">
91             <input class="form-control" type="password" value=""
92                 name="password2" id="password2">
93         </div>
94         <span class="glyphicon glyphicon-remove red"
95             id="LogoPwd2"></span>
96     </div>
97
98     <div class="panel panel-primary">
99         <div class="panel-heading">Motorcycle</div>
100         <div class="panel-body">
101             <div class="form-group row">
102                 <label for="Username" class="col-sm-3 col-form-label
103                     ">Brand</label>
104                 <div class="col-sm-9">
105                     <select class='brand' name="brand">
106                         <?php
107                         if ($value) {
108                             if (isset($values->brand)) {
109                                 echo "<option
```

```
value=\".$values->brand\"
>$values->brand</option>";

97         }
98     }
99     ?>
100     <option value="">Select...</option>
101     <?php
102     $brands = GetMotorcycleBrand();
103     foreach ($brands as $brand) {
104         echo "<option value='" . $brand . "'>" .
            $brand . "</option>";
105     }
106     ?>
107     </select>
108 </div>
109 </div>
110
111 <div class="form-group row">
112     <label for="Username" class="col-sm-3
113     col-form-label">Model </label>
114     <div class="col-sm-9">
115         <select id="model" name="model">
116             <?php
117             if ($value) {
118                 if (isset($values->model)) {
119                     echo "<option
120                         value=\".$values->model\"
121                         >$values->model</option>";
122                 }
123             }
124             ?>
125             <option value="">Select a brand</option>
126         </select>
127     </div>
128 </div>
129
130 <div class="form-group row">
131     <label for="Username" class="col-sm-3
132     col-form-label">Year </label>
133     <div class="col-sm-9">
134         <select id="year" name="year" >
135             <?php
136             if ($value) {
137                 if (isset($values->year)) {
138                     echo "<option
139                         value=\".$values->year\"
140                         >$values->year</option>";
141                 }
142             }
143             ?>
144             <option value="">Select a model</option>
145         </select>
146     </div>
147 </div>
148 </div>
149 </div>
```

```
145         <div class="form-group row">
146             <div class="col-sm-4 col-lg-offset-4">
147                 <input class="form-control btn btn-primary disabled "
148                     type="submit" name="submit" value="Sign in" id="Signin">
149             </div>
150         </div>
151     </center>
152 </div>
153 <?php include './Includes/footer.php'; ?>
154 </body>
155 </html>
156
```

```
1  <?php
2  /*
3   * Author : Lucien Camuglia
4   * Description : log out an user
5   * Date : April-june 2017
6   * Version : 1.0 LC BaseVersion
7   */
8
9  session_start();
10 session_destroy();
11 header("location: ../index.php");
12
13
```

```
1  /*
2   * Author : Lucien Camuglia
3   * Description : All map fonctions
4   * Date : April-june 2017
5   * Version : 1.0 LC BaseVersion
6   */
7
8
9  //Google API KEYS, DEFAULT is my default key for this project ans RESCUE is used
  only if DEFAULT'S quotas has been reached
10  var KEY = (function() {
11      var private = {
12          'DEFAULT': 'AIzaSyCRxYbU0CGNMpZINbtBJqn72k1UCi0bMo8',
13          'RESCUE': 'AIzaSyAe7EU4muOht_cbrvH88JTd2FTrvbsYZ_E'
14      };
15      return {
16          get: function(name) {
17              return private[name];
18          }
19      };
20  })();
21
22
23  var map;
24  var mapkey = KEY.get('DEFAULT');
25  var polyline;
26  var pointsArray = [];
27  var midmarkers = [];
28  var oldMarkers = [];
29  var Markers = [];
30  var oldRoute = [];
31  var route = [];
32  var highlighted;
33  var addedPoints = 0;
34  var directionsDisplay = new google.maps.DirectionsRenderer({suppressMarkers: true});
35  var directionsService = new google.maps.DirectionsService();
36  var elevator = new google.maps.ElevationService;
37  var paroursPolyline;
38  var imageNormal = new google.maps.MarkerImage(
39      './images/lines/square.png',
40      new google.maps.Size(11, 11),
41      new google.maps.Point(0, 0),
42      new google.maps.Point(6, 6)
43  );
44  var imageHover = new google.maps.MarkerImage(
45      './images/lines/square_over.png',
46      new google.maps.Size(11, 11),
47      new google.maps.Point(0, 0),
48      new google.maps.Point(6, 6)
49  );
50  var imageNormalMidpoint = new google.maps.MarkerImage(
51      './images/lines/square_transparent.png',
52      new google.maps.Size(11, 11),
53      new google.maps.Point(0, 0),
54      new google.maps.Point(6, 6)
55  );
56  var total = 0;
```



```

57  var DisplayTraffic = false;
58  var avoidHighways = false;
59
60  //initialise the map
61  function initMap(modif, traffic) {
62      modif = modif || false;
63      traffic = traffic || false;
64
65      var Searchinput = document.getElementById('pac-input');
66      var searchBox = new google.maps.places.SearchBox(Searchinput);
67
68      var pos = {
69          lat: 46.203545,
70          lng: 6.145150
71              //geneva
72      };
73
74      if (navigator.geolocation) {
75          navigator.geolocation.getCurrentPosition(function(position) {
76              pos = {
77                  lat: position.coords.latitude,
78                  lng: position.coords.longitude
79              };
80          })
81      }
82
83      map = new google.maps.Map(document.getElementById('map'), {
84          zoom: 4,
85          center: pos,
86          mapTypeControl: true,
87          mapTypeControlOptions: {
88              style: google.maps.MapTypeControlStyle.DROPDOWN_MENU,
89              mapTypeId: google.maps.MapTypeId.TERRAIN
90          }
91      });
92      //if the route can be modified
93      if (modif) {
94          google.maps.event.addListener(map, 'click', function(event) {
95              CreateMarker(event.latLng, true, true, "FE7569", ++addedPoints,
96                  addedPoints);
97              $("#pointsInfo").append("<div id=point" + addedPoints + ">\n\
98                  <div class=\"Pointname col-lg-12\" >\n\
99                  <span class=\"col-lg-2 align-middle\"
100                  style=\"height: 100%;\" >Point  " + addedPoints
101                  + ": </span>\n\
102                  <div class=\"col-lg-10\">\n\
103                  <div> <label class=\"col-sm-3\">Latitude :
104                  </label> <input id='inpLatPoint" + addedPoints +
105                  "' type=\"text\" value=" + event.latLng.lat() +
106                  "> </div>\n\
107                  <div> <label class=\"col-sm-3\">Longitude :
108                  </label> <input id='inpLonPoint" + addedPoints +
109                  "' type=\"text\" value=" + event.latLng.lng() +
110                  ">\n\
111                  </div></div><div></div>  ");
112              if (addedPoints > 1) {
113                  searchIti(Markers[addedPoints - 2].position, Markers[addedPoints -

```

```
1].position);
105     }
106 });
107 Searchinput.removeAttribute('hidden');
108 map.controls[google.maps.ControlPosition.TOP_LEFT].push(Searchinput);
109 map.addListener('bounds_changed', function() {
110     searchBox.setBounds(map.getBounds());
111 });
112 searchBox.addListener('places_changed', function() {
113     var places = searchBox.getPlaces();
114     if (places.length == 0) {
115         return;
116     }
117     var bounds = new google.maps.LatLngBounds();
118     places.forEach(function(place) {
119         if (!place.geometry) {
120             return;
121         }
122         var icon = {
123             url: place.icon,
124             size: new google.maps.Size(71, 71),
125             origin: new google.maps.Point(0, 0),
126             anchor: new google.maps.Point(17, 34),
127             scaledSize: new google.maps.Size(25, 25)
128         };
129         if (place.geometry.viewport) {
130             // Only geocodes have viewport.
131             bounds.union(place.geometry.viewport);
132         } else {
133             bounds.extend(place.geometry.location);
134         }
135     });
136     map.fitBounds(bounds);
137 });
138 }
139
140 //if the traffic have to be diplay
141 if (traffic) {
142     var trafficLayer = new google.maps.TrafficLayer();
143     trafficLayer.setMap(map);
144 }
145
146 //import a GPX file
147 function ImporterGpx(file) {
148     $.ajax({
149         url: './Includes/ajax.php',
150         type: 'GET',
151         data: {fonction: "ImportGPX", file: file},
152         dataType: "json",
153         success: function(idRoute) {
154
155         }
156     });
157 }
158
159 }
```

```
161 //Ask google Roads for snapping point to the road
162 function AskGoogle(Path) {
163     var snapped = [];
164     $.ajax({
165         url: 'https://roads.googleapis.com/v1/snapToRoads',
166         type: 'GET',
167         data: {path: Path, interpolate: true, key: mapkey},
168         dataType: "json",
169         async: false,
170         success: function(r2) {
171
172             $.each(r2["snappedPoints"], function(index, value)
173             {
174                 snapped[index] = value.location;
175             });
176         }
177     });
178     return snapped;
179 }
180
181 //load all points of a route
182 function LoadPoints(idRoute) {
183     var resultArray = [];
184     $.ajax({
185         url: './Includes/ajax.php',
186         type: 'GET',
187         data: {fonction: "GetRoutePoints", idRoute: idRoute},
188         dataType: "json",
189         async: false,
190         success: function(result) {
191             $.each(result, function(index, value)
192             {
193                 resultArray[index] = value;
194             });
195         }
196     });
197     return resultArray;
198 }
199
200 //snapp all points of the route to the road
201 function SnappPoints2Road(route) {
202     var Path = "";
203     var tmpSnapped = [];
204     var Snapped = [];
205     $.each(route, function(index, point)
206     {
207         if ((index > 0) && (index % 50 === 0)) {
208             Path = Path.slice(0, -1);
209             tmpSnapped = AskGoogle(Path);
210             Path = "";
211             Snapped = Snapped.concat(tmpSnapped);
212         }
213         Path += point.Latitude + "," + point.Longitude + "|";
214     });
215     Path = Path.slice(0, -1);
216     Snapped = Snapped.concat(AskGoogle(Path));
217     return Snapped;
```

```
218     }
219
220     //save a new location
221     function SaveNewLocation(idroute, route, fromweb) {
222         fromweb = fromweb || false;
223         var geoRoute = [];
224         var echantRoute = [];
225         var tmpRoute = route;
226         $.each(route, function(index, value)
227         {
228             //verify if the request is made from the web api
229             if (fromweb) {
230                 geoRoute.push(new google.maps.LatLng(value.lat(), value.lng()));
231
232             } else {
233                 geoRoute.push(new google.maps.LatLng(value.latitude, value.longitude));
234
235             }
236         });
237         //get length and calcul the sinuosity
238         var length = google.maps.geometry.spherical.computeLength(geoRoute);
239         var sinuosity = route.length / length;
240         var counter = 10;
241
242         if (geoRoute.length <= 10) {
243             counter = 1;
244         }
245         if (geoRoute.length > 4000) {
246             counter = 20;
247         }
248         if (geoRoute.length > 8000) {
249             counter = 30;
250         }
251         if (geoRoute.length > 12000) {
252             counter = 50;
253         }
254
255         for (var i = 0; i < geoRoute.length - counter; i += counter) {
256             echantRoute.push(geoRoute[i]);
257         }
258         //get the levation of each point of the route
259         elevator.getElevationAlongPath({
260             'path': echantRoute,
261             'samples': 256
262         }, function(elevations, status) {
263             if (status !== 'OK') {
264                 console.log("Error elevaton : " + status);
265             } else {
266                 var elevationAverage = 0;
267                 //calcul the elevation average
268                 for (var i = 1; i < elevations.length; i++) {
269                     elevationAverage += Math.abs(elevations[i - 1].elevation -
270                     elevations[i].elevation);
271
272                 }
273                 if (fromweb) {
274                     route = JSON.stringify(route);
275                 }
276             }
277         });
278     }
279 }
```

```

274         //send the infos to the PHP file
275         $.ajax({
276             url: './Includes/ajax.php',
277             type: 'POST',
278             data: {fonction: "SaveNewRoute", idRoute: idroute, route: route,
279                 sinuosite: sinuosity, elevation: elevationAverage, length: length},
280             async: false,
281             success: function(result) {
282                 if (fromweb) {
283                     location.reload();
284                 }
285             });
286         }
287     });
288 }
289
290 //call the PHP fonction for creating a new route
291 function CreateRoute(name, containsHighway) {
292     if (name != "") {
293         $.ajax({
294             url: './Includes/ajax.php',
295             type: 'GET',
296             data: {fonction: "CreateRoute", name: name, containsHighway:
297                 containsHighway},
298             dataType: 'json',
299             async: false,
300             success: function(result) {
301                 SaveNewLocation(result, route, true);
302             }
303         });
304     }
305
306     $(document).ready(function() {
307         initMap();
308         clear();
309         RouteClick();
310         //when the modification button is pressed
311         $("#btnModif").click(function() {
312             if (highlighted != null) {
313                 oldRoute = route;
314                 oldMarkers = Markers;
315                 clear();
316                 initMap(true);
317                 ShowParcours(LoadPoints($(this).attr('name')), true);
318                 $(".modifRoute").removeClass("hidden");
319                 $("#inpLatPoint0").val(Markers[0].position.lat());
320                 $("#inpLonPoint0").val(Markers[0].position.lng());
321                 $("#inpLatPoint1").val(Markers[1].position.lat());
322                 $("#inpLonPoint1").val(Markers[1].position.lng());
323                 editlines();
324             }
325         });
326         //when th refresh button is pressed
327         $("#RefreshRoute").click(function() {
328             if (oldMarkers.length === Markers.length) {

```

```

329         var j = 0;
330         for (var i = 0; i < Markers.length; i++) {
331             var from = [];
332             from["lat"] = Markers[i].position.lat();
333             from["lon"] = Markers[i].position.lng();
334             var to = [];
335             to["lat"] = oldMarkers[i].position.lat();
336             to["lon"] = oldMarkers[i].position.lng();
337             searchIti(from, to);
338             if (i < Markers.length - 1)
339                 while ((oldRoute[j].lat() !== oldMarkers[i + 1].position.lat())
340                     && (oldRoute[j].lng() !== oldMarkers[i + 1].position.lng())) {
341                     route.push(oldRoute[j]);
342                     j++;
343                 }
344             else
345                 while (j < oldRoute.length) {
346                     route.push(oldRoute[j]);
347                     j++;
348                 }
349         }
350     });
351 });
352
353 $("#sinuosity").change(function() {
354     RefreshhRoutesWithFilters();
355 });
356 $("#slope").change(function() {
357     RefreshhRoutesWithFilters();
358 });
359 $("#highway").change(function() {
360     RefreshhRoutesWithFilters();
361 });
362 $("#StartCreation").click(function() {
363     StartCreation();
364 });
365
366 $("#ClearRoute").click(function() {
367     ClearRoute();
368 });
369
370 $("#chkHighway").change(function() {
371     if ($("#chkHighway").is(':checked')) {
372         avoidHighways = false;
373     } else {
374         avoidHighways = true;
375     }
376 });
377 });
378
379 //when a request AJAX start, display a wheel
380 $(document).bind({
381     ajaxStart: function() {
382         $("body").addClass("loading");
383     },
384     ajaxStop: function() {

```

```
385         $("body").removeClass("loading");
386     }
387 });
388
389 function RouteClick() {
390     $(".route").click(function() {
391         clear();
392         initMap(false, DisplayTraffic);
393         $("#DeleteRoute").click(function() {
394             DeleteRoute($(highlighted).attr('name'));
395
396         });
397         $("#btnModif").removeClass("hidden");
398         $(".routeControl").removeClass("hidden");
399         $("#btnModif").attr('name', $(this).attr('name'));
400         if (highlighted != null)
401             $(highlighted).removeClass("highlight");
402         $(this).addClass("highlight");
403         highlighted = this;
404         ShowParcours(LoadPoints($(this).attr('name')), false);
405         DisplayRouteInfo($(this).attr('name'));
406
407     });
408 }
409
410 //search an itinary
411 function searchIti(from, to) {
412
413     var request = {
414         origin: from,
415         destination: to,
416         travelMode: google.maps.TravelMode.DRIVING,
417         avoidHighways: avoidHighways
418             //WALKING / DRIVING / BICYCLING / TRANSIT /
419     };
420
421     directionsDisplay.setMap(this.map);
422     directionsService.route(request, function(result, status) {
423
424         if (status === google.maps.DirectionsStatus.OK) {
425             var distanceM = result.routes[0].legs[0].distance.value;
426             var tempsS = result.routes[0].legs[0].duration.value;
427             //add all point in the current route
428             $.each(result.routes[0].overview_path, function(index, point) {
429                 route.push(point);
430             });
431             //display the route
432             DisplayRoute();
433         } else {
434             console.log(status);
435         }
436
437     });
438 }
439
440 /* display the points
441 @param array tableauPoints : Markers array to display
```

```
442  @param bool parcouresModif : if the road can be modified
443  */
444  function ShowParcoures(tableauPoints, parcouresModif, color) {
445      color = color || "#fflece";
446      var liste_des_points = tableauPoints;
447      if (parcouresModif === true) {
448          //create the first and last markers
449          CreateMarker(liste_des_points[0], parcouresModif, false, "36af2d",
            "%E2%80%A2", 0);
450          CreateMarker(liste_des_points[liste_des_points.length - 1], parcouresModif,
            false, "a52424", "%E2%80%A2", 1);
451      }
452
453      $.each(liste_des_points, function(index, value)
454      {
455          route.push(new google.maps.LatLng(value.Latitude, value.Longitude));
456      });
457      var bounds = new google.maps.LatLngBounds();
458      for (var i = 0; i < liste_des_points.length; i++) {
459          bounds.extend(new google.maps.LatLng(liste_des_points[i].Latitude,
            liste_des_points[i].Longitude));
460      }
461      map.fitBounds(bounds);
462      DisplayRoute(color);
463  }
464
465  //Display the polyline
466  function DisplayRoute(color) {
467      color = color || "#fflece";
468      parcouresPolyline = new google.maps.Polyline({
469          path: route,
470          strokeColor: color,
471          strokeOpacity: 1.0,
472          strokeWeight: 5,
473          map: map
474      });
475  }
476
477  //create a new marker
478  function CreateMarker(value, parcouresModif, parcouresDelete, pinColor, texte,
    position) {
479      //load the image and add the position in the road
480      var pinImage = new
    google.maps.MarkerImage("http://chart.apis.google.com/chart?chst=d_map_pin_letter&
    chld=" + texte + "|" + pinColor,
481          new google.maps.Size(21, 34),
482          new google.maps.Point(0, 0),
483          new google.maps.Point(10, 34));
484      var pos;
485      if (value.Latitude) {
486          pos = new google.maps.LatLng(value.Latitude, value.Longitude);
487      } else {
488          pos = value;
489      }
490
491      parcouresMarker = new google.maps.Marker({
492          position: pos,
```



```
493     draggable: parcouresModif,
494     zIndex: position,
495     map: map,
496     icon: pinImage
497 });
498 //if the marker can be delete
499 if (parcouresDelete) {
500     parcouresMarker.addListener('click', function() {
501         this.setMap(null);
502         Markers.splice(Markers.indexOf(this), 1);
503         $("#point" + this.zIndex).remove();
504         length--;
505         addedPoints--;
506     });
507 }
508
509 // if the marker can be move
510 if (parcouresModif) {
511     parcouresMarker.addListener('dragend', function() {
512         refreshValues(this.zIndex);
513         Markers[this.zIndex].previous = Markers[this.zIndex].position;
514         Markers[this.zIndex].latLng = this.position;
515     });
516 }
517
518 //add the marker in the table
519 Markers.push(parcouresMarker);
520 }
521
522 //display the values in the fields
523 function refreshValues(index) {
524     $("#inpLatPoint" + index).val(Markers[index].position.lat());
525     $("#inpLonPoint" + index).val(Markers[index].position.lng());
526 }
527
528 //Download the route
529 function DownloadRoute() {
530     if (highlighted == null) {
531         alert("Please select a trip");
532     } else
533     {
534         name = $(highlighted).html();
535         name = name.substr(0, name.indexOf("<span"));
536         $.ajax({
537             url: './Includes/ajax.php',
538             type: 'POST',
539             data: {fonction: "Download", name: name, path: JSON.stringify(route)},
540             dataType: "html",
541             async: false,
542             success: function(result) {
543                 document.location.href = './Includes/download.php?file=" + result;
544             },
545             error: function(result, status, error) {
546                 console.log(result);
547                 console.log(status);
548                 console.log(error);
549             }
550         })
551     }
552 }
```

```
550         });
551     }
552 }
553
554 //display or not the filters
555 function EnabledDisabledFilters(caller) {
556     if (caller.checked) {
557         $("#filters").removeAttr("hidden");
558         RefreshhRoutesWithFilters();
559     } else {
560         $("#filters").attr("hidden", true);
561         RefreshhRoutesWithoutFilters();
562     }
563 }
564
565 //refresh the displayed routes depending to the filters
566 function RefreshhRoutesWithFilters() {
567     var sinuosity = $("#sinuosity").val();
568     var slope = $("#slope").val();
569     var highway = $("#highway").prop("checked");
570     $.ajax({
571         url: './Includes/ajax.php',
572         type: 'GET',
573         data: {fonction: "FilterRoad", sinuosity: sinuosity, slope: slope, highway:
574             highway, time: 1},
575         dataType: "json",
576         success: function(result) {
577             $(".routes").empty();
578             $.each(result, function(index, value)
579             {
580                 $(".routes").append("<div class=\"route\" name=\"\" + value.idRoute +
581                     \">\" + value.RouteName + "<span class=\"by\"> by \" + value.Username
582                     + "</span></div>");
583             });
584             RouteClick();
585         }
586     });
587 }
588
589 //refresh the displayed routes
590 function RefreshhRoutesWithoutFilters() {
591     $.ajax({
592         url: './Includes/ajax.php',
593         type: 'GET',
594         data: {fonction: "GetRoutes"},
595         dataType: "json",
596         async: false,
597         success: function(result) {
598             $(".routes").empty();
599             $.each(result, function(index, value)
600             {
601                 $(".routes").append("<div class=\"route\" name=\"\" + value.idRoute +
602                     \">\" + value.RouteName + "<span class=\"by\"> by \" + value.Username
603                     + "</span></div>");
604             });
605             RouteClick();
606         }
607     });
608 }
```

```
602     },
603     error: function(XMLHttpRequest, textStatus, errorThrown) {
604         alert("Status: " + textStatus);
605         alert("Error: " + errorThrown);
606     }
607 });
608 }
609
610 //start the creation of the new route (display the buttons
611 function StartCreation() {
612     ClearRoute();
613     $(".SaveRouteControls").removeClass('hidden');
614 }
615
616 //display all infos about the road
617 function DisplayRouteInfo(idroute) {
618     $.ajax({
619         url: './Includes/ajax.php',
620         type: 'GET',
621         data: {fonction: "GetRoadsInfos", idRoute: idroute},
622         dataType: "json",
623         async: false,
624         success: function(result) {
625             $('#InfoHighway').html((result.Highway == 1) ? 'This road includes
highways' : 'This road does not include highways');
626             $('#InfoLength').html("Length : " + result.Length + "km");
627             $('#InfoSinuosity').html("Sinuosity : " + result.Sinuosity + "/10");
628             $('#InfoSlope').html("Slope : " + result.Slope + "/10");
629             $('#InfoDuration').html("Travel time : " + result.Time);
630             if (result.MotorcycleConsumption == 0) {
631                 $('#InfoConsumption').html("Consumption : Please login for more
informations");
632             } else {
633                 $('#InfoConsumption').html("Théorical consumption : " +
Math.round(result.Length * result.MotorcycleConsumption / 100) + "
liters");
634             }
635         },
636         error: function(result, status, error) {
637             console.log(result);
638             console.log(status);
639             console.log(error);
640         }
641     });
642 }
643
644 }
645
646 //call the PHP fonction for delete a route
647 function DeleteRoute(idRoute) {
648     $.ajax({
649         url: './Includes/ajax.php',
650         type: 'GET',
651         data: {fonction: "DeleteRoute", idRoute: idRoute},
652         dataType: 'json',
653         async: false,
654         success: function(result) {
```

```
655         location.reload();
656     }
657 });
658 }
659
660 //clear all fields
661 function ClearRoute() {
662     initMap(true, DisplayTraffic);
663     pointsArray = [];
664     midmarkers = [];
665     oldMarkers = [];
666     oldRoute = [];
667     addedPoints = 0;
668     clear();
669 }
670
671 function clear() {
672     Markers = [];
673     route = [];
674 }
675
676
677
678
679
680 // Editting the lines, found on :
681 http://www.birdtheme.org/useful/v3tool.html
682 function editlines() {
683     Markers = [];
684     route = parcoursPolyline.getPath();
685     if (route.length > 0) {
686         for (var i = 0; i < route.length; i++) {
687             var marker = setmarkers(route.getAt(i));
688             Markers.push(marker);
689             if (i > 0) {
690                 var midmarker = setmidmarkers(route.getAt(i));
691                 midmarkers.push(midmarker);
692             }
693         }
694     }
695     function setmarkers(point) {
696         var marker = new google.maps.Marker({
697             position: point,
698             map: map,
699             icon: imageNormal,
700             raiseOnDrag: false,
701             draggable: true
702         });
703         google.maps.event.addListener(marker, "mouseover", function() {
704             marker.setIcon(imageHover);
705         });
706         google.maps.event.addListener(marker, "mouseout", function() {
707             marker.setIcon(imageNormal);
708         });
709         google.maps.event.addListener(marker, "drag", function() {
710             for (var i = 0; i < Markers.length; i++) {
```

```
711         if (Markers[i] == marker) {
712             parcoursPolyline.getPath().setAt(i, marker.getPosition());
713             movemidmarker(i);
714             break;
715         }
716     }
717     route = parcoursPolyline.getPath();
718     var stringtobesaved = marker.getPosition().lat().toFixed(6) + ',' +
719     marker.getPosition().lng().toFixed(6);
720     pointsArray.splice(i, 1, stringtobesaved);
721 });
722 google.maps.event.addListener(marker, "click", function() {
723     for (var i = 0; i < Markers.length; i++) {
724         if (Markers[i] == marker && Markers.length != 1) {
725             marker.setMap(null);
726             Markers.splice(i, 1);
727             parcoursPolyline.getPath().removeAt(i);
728             // removemidmarker(i);
729             break;
730         }
731     }
732     polyPoints = parcoursPolyline.getPath();
733     if (Markers.length > 0) {
734         pointsArray.splice(i, 1);
735         logCode1();
736     }
737 });
738 return marker;
739 }
740 function setmidmarkers(point) {
741     var prevpoint = Markers[Markers.length - 2].getPosition();
742     var marker = new google.maps.Marker({
743         position: new google.maps.LatLng(
744             point.lat() - (0.5 * (point.lat() - prevpoint.lat())),
745             point.lng() - (0.5 * (point.lng() - prevpoint.lng()))
746         ),
747         map: map,
748         icon: imageNormalMidpoint,
749         raiseOnDrag: false,
750         draggable: true
751     });
752     google.maps.event.addListener(marker, "mouseover", function() {
753         marker.setIcon(imageNormal);
754     });
755     google.maps.event.addListener(marker, "mouseout", function() {
756         marker.setIcon(imageNormalMidpoint);
757     });
758     google.maps.event.addListener(marker, "dragend", function() {
759         for (var i = 0; i < midmarkers.length; i++) {
760             if (midmarkers[i] == marker) {
761                 var newpos = marker.getPosition();
762                 var startMarkerPos = Markers[i].getPosition();
763                 var firstVPos = new google.maps.LatLng(
764                     newpos.lat() - (0.5 * (newpos.lat() - startMarkerPos.lat())),
765                     newpos.lng() - (0.5 * (newpos.lng() - startMarkerPos.lng()))
766                 );
767                 var endMarkerPos = Markers[i + 1].getPosition();
```

```
767         var secondVPos = new google.maps.LatLng(  
768             newpos.lat() - (0.5 * (newpos.lat() - endMarkerPos.lat())),  
769             newpos.lng() - (0.5 * (newpos.lng() - endMarkerPos.lng()))  
770         );  
771         var newVMarker = setmidmarkers(secondVPos);  
772         newVMarker.setPosition(secondVPos); //apply the correct position to  
773         the midmarker  
774         var newMarker = setmarkers(newpos);  
775         Markers.splice(i + 1, 0, newMarker);  
776         parcouresPolyline.getPath().insertAt(i + 1, newpos);  
777         marker.setPosition(firstVPos);  
778         midmarkers.splice(i + 1, 0, newVMarker);  
779         break;  
780     }  
781     }  
782     polyPoints = parcouresPolyline.getPath();  
783     var stringtobesaved = newpos.lat().toFixed(6) + ',' + newpos.lng().toFixed(6);  
784     pointsArray.splice(i + 1, 0, stringtobesaved);  
785     logCode1();  
786     });  
787     return marker;  
788 }  
789 function movemidmarker(index) {  
790     var newpos = Markers[index].getPosition();  
791     if (index != 0) {  
792         var prevpos = Markers[index - 1].getPosition();  
793         midmarkers[index - 1].setPosition(new google.maps.LatLng(  
794             newpos.lat() - (0.5 * (newpos.lat() - prevpos.lat())),  
795             newpos.lng() - (0.5 * (newpos.lng() - prevpos.lng()))  
796         ));  
797     }  
798     if (index != Markers.length - 1) {  
799         var nextpos = Markers[index + 1].getPosition();  
800         midmarkers[index].setPosition(new google.maps.LatLng(  
801             newpos.lat() - (0.5 * (newpos.lat() - nextpos.lat())),  
802             newpos.lng() - (0.5 * (newpos.lng() - nextpos.lng()))  
803         ));  
804     }  
805 }  
806 function removemidmarker(index) {  
807     if (Markers.length > 0) { //clicked marker has already been deleted  
808         if (index != Markers.length) {  
809             midmarkers[index].setMap(null);  
810             midmarkers.splice(index, 1);  
811         } else {  
812             midmarkers[index - 1].setMap(null);  
813             midmarkers.splice(index - 1, 1);  
814         }  
815     }  
816     if (index != 0 && index != Markers.length) {  
817         var prevpos = Markers[index - 1].getPosition();  
818         var newpos = Markers[index].getPosition();  
819         midmarkers[index - 1].setPosition(new google.maps.LatLng(  
820             newpos.lat() - (0.5 * (newpos.lat() - prevpos.lat())),  
821             newpos.lng() - (0.5 * (newpos.lng() - prevpos.lng()))  
822         ));  
823     }  
824 }
```

823

}

824

825

```
1  /*
2   * Author : Lucien Camuglia
3   * Description : Sign in javascript fonctions
4   * Date : April-june 2017
5   * Version : 1.0 LC BaseVersion
6   */
7
8  //field validation regex
9  var passwordRegex = new RegExp(/^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)[a-zA-Z\d]{8,}$/i);
10 //8char 1upper 1 lower 1 number
11
12 var emailRegex = new
13 RegExp(/^([a-zA-Z0-9_+-.])+\@(([a-zA-Z0-9-])+\.)+([a-zA-Z0-9]{2,4})+$/i); //found on
14 http://stackoverflow.com/questions/2507030/email-validation-using-jquery
15
16 /**
17  * Connexion function
18  * @param {array} datas array who contains username and data
19  */
20 function connexion(datas) {
21     $.ajax({
22         url: './Includes/connexion.php',
23         type: 'POST',
24         data: {username: datas[0], password: datas[1]},
25         dataType: "json",
26         success: function(result) {
27             if (result.status === "success") {
28                 location.reload();
29             } else {
30                 $("#msgCont").removeAttr("hidden");
31                 $("#msg").html(result.message);
32             }
33         }
34     });
35 }
36
37 //Load motorcycle model from brand
38 function LoadModelFromBrand(brand) {
39     $.ajax({
40         url: './Includes/ajax.php',
41         type: 'GET',
42         data: {fonction: "GetModel", Brand: brand},
43         dataType: "json",
44         success: function(result) {
45             //clear the select
46             $("#model").html("");
47             $("#year").html("<option value=\"\">Select a model</option>");
48             $("#model").append("<option value=\"\">Select...</option>");
49             //add all returned motorcycle
50             $.each(result, function(i, item) {
51                 $("#model").append("<option value=\"" + item + ">" + item +
52                                     "</option>");
53             });
54         }
55     });
56 }
57
58 //Load motorcycle year from brand and model
59 function LoadYearFromModel(brand, model) {
```



```
54     $.ajax({
55         url: './Includes/ajax.php',
56         type: 'GET',
57         data: {fonction: "GetYear", Brand: brand, Model: model},
58         dataType: "json",
59         success: function(result) {
60             $("#year").html("");
61             $("#year").append("<option value=\"\">Select...</option>");
62             $.each(result, function(i, item) {
63                 $("#year").append("<option value=\"" + item + ">" + item + "</option>");
64             });
65         }
66     });
67
68
69 }
70
71 //check if the username exists
72 function UserNameExists(username) {
73     exist = false;
74     $.ajax({
75         url: './Includes/ajax.php',
76         type: 'GET',
77         data: {fonction: "UserExists", Username: username},
78         dataType: "json",
79         async: false,
80         timeout: 30000,
81         success: function(result) {
82             if (result.exist) {
83                 exist = true;
84             }
85         }
86     });
87     return exist;
88 }
89
90 //check if the submit button can be pressed
91 function checkForEnabled() {
92     $("#Signin").removeClass("disabled");
93     if ($("#LogoUsername").hasClass("red")) {
94         $("#Signin").addClass("disabled");
95     }
96     if ($("#Logopwd").hasClass("red")) {
97         $("#Signin").addClass("disabled");
98     }
99     if ($("#LogoEmail").hasClass("red")) {
100         $("#Signin").addClass("disabled");
101     }
102     if ($("#LogoPwd2").hasClass("red")) {
103         $("#Signin").addClass("disabled");
104     }
105     if ($("#brand").val() === "") {
106         $("#Signin").addClass("disabled");
107     }
108     if ($("#model").val() === "") {
109         $("#Signin").addClass("disabled");
110     }
111 }
```

```
111     if ($("#year").val() === "") {
112         $("#Signin").addClass("disabled");
113     }
114 }
115
116 $(document).ready(function() {
117
118     //active Tooltip
119     $('[data-toggle="tooltip"]').tooltip();
120
121     $(".brand").click(function() {
122         LoadModelFromBrand($("#this").val());
123         checkForEnabled();
124     });
125
126     $("#model").click(function() {
127         LoadYearFromModel($(".brand").val(), $("#this").val());
128         checkForEnabled();
129     });
130
131     $("#year").click(function() {
132         checkForEnabled();
133     });
134
135     //check if the username exists or not when the input change
136     $('#Username').on('input', function(e) {
137         text = $("#Username").val();
138         $("#LogoUsername").removeClass("green");
139         $("#LogoUsername").removeClass("glyphicon-ok");
140         $("#LogoUsername").addClass("red");
141         $("#LogoUsername").addClass("glyphicon-remove");
142         if (text.length >= 5) {
143             exist = UserNameExists(text);
144             console.log(exist);
145             if (!exist) {
146                 $("#LogoUsername").removeClass("red");
147                 $("#LogoUsername").removeClass("glyphicon-remove");
148                 $("#LogoUsername").addClass("green");
149                 $("#LogoUsername").addClass("glyphicon-ok");
150             }
151         }
152         checkForEnabled();
153     });
154
155     //check if the password is correct or not when the input change
156     $('#password').on('input', function(e) {
157         $("#Logopwd").removeClass("green");
158         $("#Logopwd").removeClass("glyphicon-ok");
159         $("#Logopwd").addClass("red");
160         $("#Logopwd").addClass("glyphicon-remove");
161         if (passwordRegex.test($("#password").val())) {
162             $("#Logopwd").removeClass("red");
163             $("#Logopwd").removeClass("glyphicon-remove");
164             $("#Logopwd").addClass("green");
165             $("#Logopwd").addClass("glyphicon-ok");
166         }
167         checkForEnabled();
168     });
169 }
```

```
168     });
169
170     //check if the email is correct or not when the input change
171     $('#email').on('input', function(e) {
172         $('#LogoEmail').removeClass("green");
173         $('#LogoEmail').removeClass("glyphicon-ok");
174         $('#LogoEmail').addClass("red");
175         $('#LogoEmail').addClass("glyphicon-remove");
176         if (emailRegex.test($('#email').val())) {
177             $('#LogoEmail').removeClass("red");
178             $('#LogoEmail').removeClass("glyphicon-remove");
179             $('#LogoEmail').addClass("green");
180             $('#LogoEmail').addClass("glyphicon-ok");
181         }
182         checkForEnabled();
183     });
184
185     //check if the 2nd password equals the 1st or not when the input change
186     $('#password2').on('input', function(e) {
187         $('#LogoPwd2').removeClass("green");
188         $('#LogoPwd2').removeClass("glyphicon-ok");
189         $('#LogoPwd2').addClass("red");
190         $('#LogoPwd2').addClass("glyphicon-remove");
191         if ($('#password2').val() === $('#password').val()) {
192             $('#LogoPwd2').removeClass("red");
193             $('#LogoPwd2').removeClass("glyphicon-remove");
194             $('#LogoPwd2').addClass("green");
195             $('#LogoPwd2').addClass("glyphicon-ok");
196         }
197         checkForEnabled();
198     });
199
200 });
```

```
1  /*
2   * Author : Lucien Camuglia
3   * Description : managing users fonctions
4   * Date : April-june 2017
5   * Version : 1.0 LC BaseVersion
6   */
7
8  //update the user role
9  function ModifyUser(idUser, role) {
10     $.ajax({
11         url: './Includes/ajax.php',
12         type: 'GET',
13         data: {fonction: "UpdateUserRole", idUser: idUser, Role: role},
14         dataType: "json",
15         success: function(result) {
16             $("#" + idUser + "Role").html(result.datas.NewRole);
17             location.reload();
18         }
19     });
20 }
21
22
23 $(document).ready(function() {
24     $(".BanRole").click(function() {
25         var idUser = $(this).attr('name');
26         ModifyUser(idUser, 3);
27     });
28
29     $(".UpgradeRole").click(function() {
30         var idUser = $(this).attr('name');
31         var role = $("#" + idUser + "Role").attr('name');
32         ModifyUser(idUser, --role);
33     });
34
35     $(".DowngradeRole").click(function() {
36         var idUser = $(this).attr('name');
37         var role = $("#" + idUser + "Role").attr('name');
38         ModifyUser(idUser, ++role);
39     });
40 });
41
```

```
1  /*
2   * Author : Lucien Camuglia
3   * Description : managing motorcycles fonctions
4   * Date : April-june 2017
5   * Version : 1.0 LC BaseVersion
6   */
7
8  //create a new motorcycle
9  function AddMotorcycle(brand, model, year, consumption, tiredness) {
10
11      $.ajax({
12          url: './Includes/ajax.php',
13          type: 'GET',
14          data: {fonction: "AddMotorcycle", Brand: brand, Model: model, Year: year,
15              Consumption: consumption, Tiredness: tiredness},
16          dataType: "json",
17          async: false,
18          success: function(result) {
19              if (result.error.status === true) {
20              } else {
21                  window.location.reload(false);
22              }
23          }
24      });
25  }
26
27
28  //load all motorcycles
29  function LoadMotorcycles(brand, model, year, consumption, tiredness) {
30      $.ajax({
31          url: './Includes/ajax.php',
32          type: 'GET',
33          data: {fonction: "GetMotorcycles", Brand: brand, Model: model, Year: year,
34              Consumption: consumption, Tiredness: tiredness},
35          dataType: "json",
36          async: false,
37          success: function(result) {
38              $("#motorcyclesDatas").html("");
39              $.each(result, function(i, item) {
40
41                  $("#motorcyclesDatas").append("<tr> <td>" + item.Brand + "</td>\n\
42                      <td>" + item.model + "</td>\n\
43                      <td>" + item.year.substring(0, 4) +
44                      "</td> \n\
45                      <td>" + item.consumption + "</td>\n\
46                      <td>" + item.Tiredness + "</td>\n\
47                      <td> <span class=\"glyphicon glyphicon-remove red\" ></span>
48                      </td></tr>");
49
50              });
51          }
52      });
53  }
```

```
53 $(document).ready(function() {
54     //filters for motorcycles
55     $("#select").click(function() {
56         LoadMotorcycles($("#brand").val(), $("#model").val(), $("#year").val() +
57             "-01-01", $("#consumption").val(), $("#tiredness").val());
58     });
59 }
```

```
1  <?php
2  /*
3   * Author : Lucien Camuglia
4   * Description : header of html pages
5   * Date : April-june 2017
6   * Version : 1.0 LC BaseVersion
7   */
8
9  /*
10   * *****
11   *                                     CLEF API
12   * *****
13   * AIzaSyCRxYbU0CGNMpZINbtBJqn72k1UCi0bMo8
14   */
15  session_start();
16  include './Includes/menu.php';
17  include './Includes/functions.php';
18  ?>
19  <meta charset="UTF-8">
20  <meta name="description" content="Sharng road between motorcycle enthousiasts">
21  <meta name="keywords" content="Motorcycle,road,moto,ride,findyourride">
22  <meta name="author" content="Lucien Camuglia">
23
24  <link rel="stylesheet" href="./Includes/bootstrap/css/bootstrap.min.css"/>
25  <link rel="stylesheet" href="./CSS/main.css"/>
26  <script
27  src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCRxYbU0CGNMpZINbtBJqn72k1UCi0bMo8&libraries=geometry,places&v=3&sensor=false"></script>
28  <script src="./JS/jquery.min.js"></script>
29  <script src="./JS/Signin.js"></script>
30  <link href="./Includes/bootstrap/bootstrap-toggle/css/bootstrap-toggle.min.css"
31  rel="stylesheet">
32  <link rel="icon" type="image/png" href="./images/logo.png" />
33
34  <script
35  src="./Includes/bootstrap/bootstrap-toggle/js/bootstrap-toggle.min.js"></script>
36  <script
37  src="https://github.com/douglascrockford/JSON-js/blob/master/json2.js"></script>
38
39  <script src="./Includes/bootstrap/js/bootstrap.min.js"></script>
```

```
1 <footer class="footer">
2   <span>Lucien Camuglia &copy; CFPT Technician Diploma 2017</span><span
   style="text-align: right;"><a
   href="https://github.com/LucienCamuglia/FindYourRide" target = "_blank">Follow
   me on GitHub</a></span>
3 </footer>
4
```



```
1  <?php
2  /*
3   * Author : Lucien Camuglia
4   * Description : navigation menu
5   * Date : April-june 2017
6   * Version : 1.0 LC BaseVersion
7   */
8  require_once './Includes/Modals.php';
9
10 function DisplayMenu($page) {
11     includeConnexionModal();
12     $erreur = false;
13     if (isset($_SESSION['erreur'])) {
14         $erreur = true;
15         $erreurMsg = $_SESSION['erreur'];
16         session_unset($_SESSION['erreur']);
17     }
18     ?>
19     <nav class="navbar navbar-default">
20         <div class="container-fluid">
21             <div class="navbar-header">
22                 <button type="button" class="navbar-toggle collapsed"
23                     data-toggle="collapse" data-target="#bs-example-navbar-collapse-1"
24                     aria-expanded="false">
25                     <span class="sr-only">Toggle navigation</span>
26                     <span class="icon-bar"></span>
27                     <span class="icon-bar"></span>
28                     <span class="icon-bar"></span>
29                 </button>
30                 <a class="navbar-brand" href="./Index.php">Find your ride</a>
31             </div>
32
33             <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
34                 <ul class="nav navbar-nav">
35                     <li <?php
36                         if ($page == "Index.php") {
37                             echo 'class="active"';
38                         }
39                         ?><a href="./Index.php">Home <span
40                             class="sr-only">(current)</span></a></li>
41                     <li <?php
42                         if ($page == "map.php") {
43                             echo 'class="active"';
44                         }
45                         ?><a href="./map.php">Map</a></li>
46
47                     <?php
48                     if (isset($_SESSION["id"])) {
49                         ?>
50
51                     <li <?php
52                         if ($page == "itineraire.php") {
53                             echo 'class="active"';
54                         }
55                         ?><a href="./itineraire.php">My routes</a></li>
```

```
55         <?php if ($_SESSION["role"] == 1) { ?>
56         <li <?php
57         if ($page == "manageUsers.php") {
58             echo 'class="active"';
59         }
60         ?><a href="./manageUsers.php">Manage users</a></li>
61
62         <li <?php
63         if ($page == "manageMoto.php") {
64             echo 'class="active"';
65         }
66         ?><a href="./manageMoto.php">Manage motorcycles</a></li>
67         <?php } ?>
68     </ul>
69
70
71     <ul class="nav navbar-nav navbar-right">
72         <li><a href="./logout.php">Log Out</a></li>
73     </ul>
74     <ul class="nav navbar-nav navbar-right">
75         <li><a>Logged as <?php echo $_SESSION["username"]; ?></a></li>
76     </ul>
77     <?php
78 } else {
79     ?>
80 </ul>
81 <ul class="nav navbar-nav navbar-right">
82     <li <?php
83     if ($page == "Inscription.php") {
84         echo 'class="active"';
85     }
86     ?><a href="./Inscription.php">Sign in</a></li>
87
88 </ul>
89 <ul class="nav navbar-nav navbar-right">
90     <li <?php
91     if ($page == "connexion.php") {
92         echo 'class="active"';
93     }
94     ?> <a href="#" data-toggle="modal"
95         data-target="#modalConnexion"
96         onclick='$("#msgCont").attr("Hidden",
97             "true")'>Connection</a></li>
98
99 </ul>
100 <?php } ?>
101 </div>
102 </div>
103 </nav>
104 <?php
105 }
```

```
1  <?php
2  /*
3   * Author : Lucien Camuglia
4   * Description : contains all modal
5   * Date : April-june 2017
6   * Version : 1.0 LC BaseVersion
7   */
8
9  //Fenêtre modal de connexion
10 function includeConnexionModal() {
11     ?>
12     <div class="modal fade" id="modalConnexion" tabindex="-1" role="dialog"
13         aria-labelledby="exampleModalLabel" aria-hidden="true">
14
15         <div class="modal-dialog" role="document">
16             <div class="modal-content">
17                 <div class="modal-header">
18                     <div class="alert alert-danger" id='msgCont' hidden="true">
19                         <strong>Error !</strong> <label id='msg'></label>
20                     </div>
21                     <h5 class="modal-title" >Connexion</h5>
22                     <button type="button" class="close" data-dismiss="modal"
23                         aria-label="Close">
24                         <span aria-hidden="true">&times;</span>
25                     </button>
26                 </div>
27                 <div class="modal-body">
28                     <div class="form-group row">
29                         <label class="col-2 col-form-label">Username</label>
30                         <div class="col-4">
31                             <input class="form-control" type="text" id="username"
32                                 name="username" required>
33                         </div>
34                     </div>
35                     <div class="form-group row">
36                         <label class="col-2 col-form-label">Password</label>
37                         <div class="col-4">
38                             <input class="form-control" type="password" id="pwd"
39                                 name="password" required>
40                         </div>
41                     </div>
42                 </div>
43                 <div class="modal-footer">
44                     <button type="button" class="btn btn-secondary"
45                         data-dismiss="modal">Close</button>
46                     <button type="button" class="btn btn-primary"
47                         onclick='connexion([$("#username").val(),
48                             $("#pwd").val()])'>Connexion</button>
49                 </div>
50             </div>
51         </div>
52     </div>
53     <?php
54 }
55
56 //fenêtre modale d'ajout de moto
57 function AddMotorcycleModal() {
```

```

51     ?>
52     <div class="modal fade" id="modalMotorcycle" tabindex="-1" role="dialog"
    aria-labelledby="Modal motorcycle" aria-hidden="true">
53
54         <div class="modal-dialog" role="document">
55             <div class="modal-content">
56                 <div class="modal-header">
57                     <div class="alert alert-danger" id='msgCont' hidden="true">
58                         <strong>Erreur !</strong> <label id='msg'></label>
59                     </div>
60                     <h5 class="modal-title" >Add a Motorcycle</h5>
61                     <button type="button" class="close" data-dismiss="modal"
    aria-label="Close">
62                         <span aria-hidden="true">&times;</span>
63                     </button>
64                 </div>
65                 <div class="modal-body">
66                     <div class="form-group row">
67                         <label class="col-2 col-form-label">Brand</label>
68                         <div class="col-4">
69                             <input class="form-control" type="text" id="AddBrand"
    name="Brand" required>
70                         </div>
71                     </div>
72                     <div class="form-group row">
73                         <label class="col-2 col-form-label">Model</label>
74                         <div class="col-4">
75                             <input class="form-control" type="text" id="AddModel"
    name="Model" required>
76                         </div>
77                     </div>
78                     <div class="form-group row">
79                         <label class="col-2 col-form-label">Year</label>
80                         <div class="col-4">
81                             <select id="Addyear" name="year" required>
82                                 <?php
83                                     for ($i = 1900; $i <= date("Y"); $i++) {
84                                         echo "<option value=\"\$i\">\$i</option>";
85                                     }
86                                 ?>
87                             </select>
88                         </div>
89                     </div>
90                     <div class="form-group row">
91                         <label class="col-2 col-form-label">Consumption</label>
92                         <div class="col-4">
93                             <input class="form-control" type="number"
    id="AddConsumption" name="Consumption" required>
94                         </div>
95                     </div>
96
97                     <div class="form-group row">
98                         <label class="col-2 col-form-label">Tiredness</label>
99                         <div class="col-4">
100                             <input class="form-control" type="number"
    id="AddTiredness" name="Tiredness" required>
101                         </div>

```

```

102         </div>
103     </div>
104     <div class="modal-footer">
105         <button type="button" class="btn btn-secondary"
106             data-dismiss="modal">Close</button>
107         <button type="button" class="btn btn-primary"
108             onclick='AddMotorcycle($("#AddBrand").val(),
109                 $("#AddModel").val(), $("#Addyear").val(),
110                 $("#AddConsumption").val(),
111                 $("#AddTiredness").val())'>Add</button>
112     </div>
113 </div>
114 </div>
115 <?php
116 }
117
118 //fenêtre modal d'import de trajet
119 function importerTrajetModal() {
120     ?>
121     <div class="modal fade" id="modalimport" tabindex="-1" role="dialog"
122         aria-labelledby="modalimport" aria-hidden="true">
123         <form method="post" action="./Includes/ImportGPX.php"
124             enctype="multipart/form-data">
125             <div class="modal-dialog" role="document">
126                 <div class="modal-content">
127                     <div class="modal-header">
128                         <div class="alert alert-danger" id='msgCont' hidden="true">
129                             <strong>Erreur !</strong> <label id='msg'></label>
130                         </div>
131                         <h5 class="modal-title">Importer fichier .GPX</h5>
132                         <button type="button" class="close" data-dismiss="modal"
133                             aria-label="Close">
134                             <span aria-hidden="true">&times;</span>
135                         </button>
136                     </div>
137                     <div class="modal-body">
138                         <div class="form-group row">
139                             <label class="col-2 col-form-label">Fichier</label>
140                             <div class="col-4">
141                                 <input class="form-control" type="file"
142                                     id="GpxFile" name="GpxFile" required>
143                             </div>
144                             <label class="col-2 col-form-label">This route contains
145                                 highways </label> <input class="col-lg-offset-1"
146                                     type="checkbox" name="highway">
147                         </div>
148
149                         <div class="modal-footer">
150                             <button type="button" class="btn btn-secondary"
151                                 data-dismiss="modal">Close</button>
152                             <input type="submit" class="btn btn-primary"
153                                 value="Importer" name="import">
154                         </div>
155                     </div>
156                 </div>
157             </div>
158         </div>
159     </div>
160 </div>

```

```

146         </form>
147     </div>
148     <?php
149 }
150
151 //fenetre modale de confirmation de suppression d'une moto
152 function confirmationDeleteMotorcycleModal($id, $brand, $model, $year) {
153     ?>
154     <div class="modal fade" id="modaldelete<?php echo $id ?>" tabindex="-1"
155         role="dialog" aria-labelledby="modaldelete<?php echo $id ?>" aria-hidden="true">
156         <form method="post" action="./Includes/deleteMotorcycle.php" >
157             <div class="modal-dialog" role="document">
158                 <div class="modal-content">
159                     <div class="modal-header">
160
161                         <h5 class="modal-title" >Delete a motorcycle</h5>
162                         <button type="button" class="close" data-dismiss="modal"
163                             aria-label="Close">
164                             <span aria-hidden="true">&times;</span>
165                         </button>
166                     </div>
167                     <div class="modal-body">
168                         <div class="form-group row">
169                             <label class="col-2 col-form-label">Are you sure to
170                             delete :</label>
171                             <label class="col-2 col-form-label"><?php echo "$brand,
172                             $model, $year" ?></label>
173                             <input hidden value="<?php echo $id ?>" name="toDelete"/>
174                         </div>
175
176                         <div class="modal-footer">
177                             <button type="button" class="btn btn-primary"
178                                 data-dismiss="modal">Close</button>
179                             <input type="submit" class="btn btn-danger" value="Yes"
180                                 name="delete">
181                         </div>
182                     </div>
183                 </div>
184             </div>
185         </form>
186     </div>
187     <?php
188 }
189
190 //fenêtre modal de création d'une nouvelle route
191 function newRouteModal() {
192     ?>
193     <div class="modal fade" id="newRoutemodal" tabindex="-1" role="dialog"
194         aria-labelledby="newRoutemodal" aria-hidden="true">
195         <div class="modal-dialog" role="document">
196             <div class="modal-content">
197                 <div class="modal-header">
198                     <div class="alert alert-danger" id='msgCont' hidden="true">
199                         <strong>Erreur !</strong> <label id='msg'></label>
200                     </div>
201                     <h5 class="modal-title" > New route</h5>
202                     <button type="button" class="close" data-dismiss="modal"

```

```
196         aria-label="Close">
197             <span aria-hidden="true">&times;</span>
198         </button>
199     </div>
200     <div class="modal-body">
201         <div class="form-group row">
202             <label class="col-2 col-form-label">Route name : </label>
203             <div class="col-4">
204                 <input class="form-control" type="text"
205                     id="newRouteName" name="newRouteName" required>
206             </div>
207             <label class="col-2 col-form-label" >This route contains
208                 highways </label> <input class="col-lg-offset-1"
209                     type="checkbox" name="highway">
210             </div>
211
212         <div class="modal-footer">
213             <button type="button" class="btn btn-secondary"
214                 data-dismiss="modal">Close</button>
215             <input type="submit" class="btn btn-primary"
216                 value="Importer" name="import"
217                 onclick="CreateRoute ($ ( '#newRouteName' ) .val () , $ ( '#chkHighw
218                     ay' ) .is ( ':checked' ) ) ;">
```

```
1  <?php
2  /*
3   * Author : Lucien Camuglia
4   * Description : Connexion file
5   * Date : April-june 2017
6   * Version : 1.0 LC BaseVersion
7   */
8
9  session_start();
10 include_once './functions.php';
11
12 if (!empty($_POST['username'])) {
13     if (!empty($_POST['password'])) {
14         $connexion = ConnexionUser($_POST['username'], sha1($_POST['password']));
15         if (!$connexion) {
16             $reponse_array['status'] = 'error';
17             $reponse_array['message'] = 'wrong username or password';
18         } else {
19             if ($_SESSION["role"] == 3) {
20                 session_destroy();
21                 $reponse_array['status'] = 'error';
22                 $reponse_array['message'] = 'You are banned';
23             } else {
24                 $reponse_array['status'] = 'success';
25             }
26         }
27     } else {
28         $reponse_array['status'] = 'error';
29         $reponse_array['message'] = 'password is required';
30     }
31 } else {
32     $reponse_array['status'] = 'error';
33     $reponse_array['message'] = 'username is required';
34 }
35
36 header('Content-type: application/json');
37 echo json_encode($reponse_array);
38
```



```
1  <?php
2  /*
3   * Author : Lucien Camuglia
4   * Description :sing in fonction
5   * Date : April-june 2017
6   * Version : 1.0 LC BaseVersion
7   */
8
9  require_once './functions.php';
10
11 //check all input values
12 if (isset($_POST["submit"])) {
13     if (isset($_POST["username"])&&!empty($_POST["username"])) {
14         $values["username"] = $_POST["username"];
15         if (isset($_POST["email"])&&!empty($_POST["email"])) {
16             $values["email"] = $_POST["email"];
17             if (isset($_POST["password"])&&!empty($_POST["password"])) {
18                 $values["password"] = sha1($_POST["password"]);
19                 if (isset($_POST["password2"])&&!empty($_POST["password2"])) {
20                     $values["password2"] = sha1($_POST["password2"]);
21                     if (isset($_POST["brand"])&&!empty($_POST["brand"])) {
22                         $values["brand"] = $_POST["brand"];
23                         if (isset($_POST["model"])&&!empty($_POST["model"])) {
24                             $values["model"] = $_POST["model"];
25                             if (isset($_POST["year"])&&!empty($_POST["year"])) {
26                                 $values["year"] = $_POST["year"];
27
28                                 if ($values["password"] == $values["password2"]) {
29                                     if(strlen($values["username"])>=5){
30                                         signin(json_encode($values));
31                                         header("location:
32                                             ../Index.php?validation=Inscription%20réussi
33                                             e");
34                                     }else {
35                                         redirect("Username length minimum 5 characters",
36                                             $values);
37                                     }
38                                 } else {
39                                     redirect("Password aren't the same", $values);
40                                 }
41                             } else {
42                                 redirect("Year is required", $values);
43                             }
44                         } else {
45                             redirect("model is required", $values);
46                         }
47                     } else {
48                         redirect("brand is required", $values);
49                     }
50                 } else {
51                     redirect("Password confirmation is required", $values);
52                 }
53             } else {
54                 redirect("Password is required", $values);
55             }
56         } else {
57             redirect("Email is required", $values);
58         }
59     }
60 }
```

```
55         }
56     } else {
57         redirect("Username is required");
58     }
59 }else {
60     redirect();
61 }
62
63 exit();
64
65 function redirect($error = NULL, $values = NULL) {
66     $values = json_encode($values);
67     header("location: ../Inscription.php?error=$error&values=$values");
68 }
69 ?>
70
```

```
1  <?php
2
3  /*
4   * Author : Lucien Camuglia
5   * Description : Import a GPX file
6   * Date : April-june 2017
7   * Version : 1.0 LC BaseVersion
8   */
9
10 /**
11  * Connexion à la base de données
12  * @staticvar PDO $pdo objet de connexion
13  * @return PDO connexion à la base
14  */
15 function connexionDb() {
16
17     try {
18         //database login infos
19         $serveur = '127.0.0.1';
20         $pseudo = 'fyr';
21         $pwd = 'NekoFyr';
22         $db = 'findyourride';
23
24         static $pdo = null;
25
26         if ($pdo === NULL) {
27             // Connection to database.
28             $pdo_options[PDO::ATTR_ERRMODE] = PDO::ERRMODE_EXCEPTION;
29             $pdo = new PDO("mysql:host=$serveur;dbname=$db", $pseudo, $pwd,
30                 $pdo_options);
31             $pdo->exec("Set Character set UTF8");
32         }
33         return $pdo;
34     } catch (Exception $exc) {
35         echo $exc->getTraceAsString();
36     }
37
38     $pdo = connexionDb();
39
40 /**
41  * Prépare et exécute une requête
42  * @param string $query requête a éxécuter
43  * @param tableau $params éventuel paramètre pour la requête
44  * @return PDOStatement
45  */
46 Function PrepareExecute($query, $params = NULL) {
47     global $pdo;
48     // Préparation de la requête SQL.
49     $st = $pdo->prepare($query);
50     // Execution de la requête SQL.
51     $st->execute($params);
52     return $st;
53 }
54
55 /**
56  * Connecte un utilisateur
```

```
57  * @param String $User Nom d'utilisateur
58  * @param Sha1_String $Password Mot de passe crypté
59  * @Return Boolean Vrai si la connexion a marché
60  */
61  function ConnexionUser($User, $Password) {
62      global $pdo;
63
64      $query = 'SELECT idUser, Username, Password, role FROM users ';
65
66      $st = PrepareExecute($query);
67      $connecter = false;
68
69      while ($data = $st->fetch(PDO::FETCH_ASSOC)) {
70          if ($data['Username'] == $User && $data['Password'] == $Password) {
71              $id = $data['idUser'];
72              $username = $data["Username"];
73              $role = $data["role"];
74
75              $connecter = true;
76              break;
77          }
78      }
79      if ($connecter) {
80          $_SESSION["role"] = $role;
81          $_SESSION["id"] = $id;
82          $_SESSION["username"] = $username;
83          return true;
84      } else {
85          return false;
86      }
87  }
88
89  /**
90   * Return all brand
91   * @return array all motorcycle brand
92   */
93  function GetMotorcycleBrand() {
94      $query = "select distinct Brand from moto";
95      $st = PrepareExecute($query);
96      while ($data = $st->fetch(PDO::FETCH_ASSOC)) {
97          $array_response[] = $data["Brand"];
98      }
99      return $array_response;
100  }
101
102  /**
103   * Return all motorcycle model
104   * @return array with all motorcycle model
105   */
106  function GetAllMotorcycleModel() {
107      $query = "select distinct model from moto order by model";
108      $st = PrepareExecute($query);
109      while ($data = $st->fetch(PDO::FETCH_ASSOC)) {
110          $array_response[] = $data["model"];
111      }
112      return $array_response;
113  }
```

```
114
115  /**
116   * return all years of motorcycle
117   * @return array with all years
118   */
119  function GetAllMotorcycleYear() {
120      $query = "select distinct year from moto order by year";
121      $st = PrepareExecute($query);
122      while ($data = $st->fetch(PDO::FETCH_ASSOC)) {
123          $array_response[] = substr($data["year"], 0, 4);
124      }
125      return $array_response;
126  }
127
128  /**
129   * return all consumptions of motorcycle
130   * @return array with all consumptions
131   */
132  function GetAllMotorcycleConsumption() {
133      $query = "select distinct consumption from moto order by consumption";
134      $st = PrepareExecute($query);
135      while ($data = $st->fetch(PDO::FETCH_ASSOC)) {
136          $array_response[] = $data["consumption"];
137      }
138      return $array_response;
139  }
140
141  /**
142   * return all Tiredness of motorcycle
143   * @return array with all Tiredness
144   */
145  function GetAllMotorcycleTiredness() {
146      $query = "select distinct Tiredness from moto order by Tiredness";
147      $st = PrepareExecute($query);
148      while ($data = $st->fetch(PDO::FETCH_ASSOC)) {
149          $array_response[] = $data["Tiredness"];
150      }
151      return $array_response;
152  }
153
154  /**
155   * Return all motorcycle
156   * @return array with all motorcycle ordered by brand
157   */
158  function GetAllMotorcycles() {
159      $query = "select * from moto order by Brand";
160      $st = PrepareExecute($query);
161      while ($data = $st->fetch(PDO::FETCH_ASSOC)) {
162          $array_response[] = $data;
163      }
164      return $array_response;
165  }
166
167  /**
168   * Sign in a user
169   * @param array $values
170   * @return int as boolean false if the user exists and the id if not
```

```

171  */
172  function signin($values) {
173      $values = json_decode($values);
174      //get the user in the bd who have the same username
175      $query = "Select Username FROM users where Username=:username;";
176      $params = array('username' => $values->username);
177      $st = PrepareExecute($query, $params);
178      //if something is returned, the user already exists
179      while ($data = $st->fetch(PDO::FETCH_ASSOC)) {
180          return false;
181      }
182      //get the id moto of the selected motorcycle
183      $query = "SELECT idMoto FROM moto WHERE Brand=:brand AND model=:model AND
year=:year";
184      $params = array(
185          'brand' => $values->brand,
186          'model' => $values->model,
187          'year' => $values->year . "-01-01"
188      );
189      $st = PrepareExecute($query, $params);
190      $idMoto = $st->fetch(PDO::FETCH_ASSOC) ["idMoto"];
191      //insert in the db the new user
192      $query = "INSERT INTO users(Username, Password, email, idMoto, role) VALUES
(:username,:password,:email,:idmoto,2)";
193      $params = array(
194          'username' => $values->username,
195          'password' => $values->password,
196          'email' => $values->email,
197          'idmoto' => $idMoto
198      );
199      $st = PrepareExecute($query, $params);
200
201      global $pdo;
202      //return the last inserted id
203      return $pdo->lastInsertId();
204  }
205
206  /**
207   * Get all routes
208   * @param $idUser facultative user id
209   * @return JSON array
210   */
211  function GetRoutes($idUser = NULL) {
212      if ($idUser == NULL) {
213          $query = "select * from route natural join users order by RouteName ";
214          $st = PrepareExecute($query);
215      } else {
216          $query = "select * from route where idUser = :iduser order by RouteName";
217          $params = array(':iduser' => $idUser);
218          $st = PrepareExecute($query, $params);
219      }
220      $array_response = [];
221      while ($data = $st->fetch(PDO::FETCH_ASSOC)) {
222          $array_response[] = $data;
223      }
224      return $array_response;
225  }

```

```
226
227 /**
228  * Create a new route
229  * @param string $name name of the trip
230  * @param int $iduser
231  * @param boolean $containsHighway
232  * @return int Route Id
233  */
234 function CreateRoute($name, $iduser, $containsHighway) {
235     $query = "INSERT INTO route(RouteName, idUser, highway) VALUES (:name,:user,
236     :highway)";
237     $params = array(
238         'name' => $name,
239         'user' => $iduser,
240         'highway' => $containsHighway
241     );
242     $st = PrepareExecute($query, $params);
243     global $pdo;
244     return $pdo->lastInsertId();
245 }
246 /**
247  * Delete all points of a road
248  * @param int $idRoute
249  */
250 function deletePlaces($idRoute) {
251     $query = "DELETE FROM `place` WHERE idRoute = :idRoute";
252     $params = array(
253         'idRoute' => $idRoute
254     );
255     $st = PrepareExecute($query, $params);
256 }
257 /**
258  * Add a new point of a road
259  * @param double $lat latitude
260  * @param double $lon longitude
261  * @param int $position position of the point in the route
262  * @param int $idRoute
263  */
264 function addPlaceToRoute($lat, $lon, $position, $idRoute) {
265     $query = "INSERT INTO place(latitude, longitude, position, idRoute) VALUES
266     (:lat,:lon,:pos,:route)";
267     $params = array(
268         'lat' => $lat,
269         'lon' => $lon,
270         'pos' => $position,
271         'route' => $idRoute
272     );
273     $st = PrepareExecute($query, $params);
274 }
275 /**
276  * Save a GPX file in the database
277  * @param string $file path and filename
278  * @param int $iduser
279  * @param boolean $containsHighway
```

```

281  * @return int routeID
282  */
283  function Gpx2Sql($file, $iduser, $containsHighway) {
284      //create a new dom Document
285      $dom = new DOMDocument();
286      //open the file
287      $dom->load($file);
288      //get the route name in the file
289      $names = $dom->getElementsByTagName("name");
290      $name = $names->item(0)->nodeValue;
291      //get all trakpoint
292      $trackpoint = $dom->getElementsByTagName("trkpt");
293      if ($trackpoint == NULL) {
294          //get all route point
295          $trackpoint = $dom->getElementsByTagName("rtept");
296      }
297      //create the route and get the id
298      $idRoute = CreateRoute($name, $iduser, $containsHighway);
299      $position = 0;
300      //add all points in the database
301      foreach ($trackpoint as $point) {
302          $lat = $point->getAttribute("lat");
303          $lon = $point->getAttribute("lon");
304          addPlaceToRoute($lat, $lon, $position++, $idRoute);
305      }
306      return $idRoute;
307  }
308
309  /**
310   * Convert a path into a GPX file
311   * @param string $name road name
312   * @param JSON array $path
313   * @return string filename
314   */
315  function Path2Gpx($name, $path) {
316      $path = json_decode($path);
317      $file = $name . ".gpx";
318      if (file_exists($file)) {
319          unlink($file);
320      }
321      $handle = fopen($file, "w");
322      $string = "<?xml version=\"1.0\" encoding=\"UTF-8\"?>";
323      fwrite($handle, $string);
324      $string = "<gpx version=\"1.0\" creator=\"FindYourRide.org\"
xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\"
xmlns=\"http://www.topografix.com/GPX/1/0\"
xsi:schemaLocation=\"http://www.topografix.com/GPX/1/0
http://www.topografix.com/GPX/1/0/gpx.xsd\">";
325      fwrite($handle, $string);
326      $string = "<trk> <name>$name</name> <trkseg>";
327      fwrite($handle, $string);
328      foreach ($path as $point) {
329          $string = "<trkpt lat=\"\" . $point->lat . \"\" lon=\"\" . $point->lng . \"\">
</trkpt>";
330          fwrite($handle, $string);
331      }
332  }

```



```
333     $string = "</trkseg> </trk> </gpx>";
334     fwrite($handle, $string);
335     fclose($handle);
336     return $file;
337 }
338
339 /**
340  * Delete a motorcycle
341  * @param int $id motorcycle id
342  */
343 function deleteMotorcycle($id) {
344     $query = "Delete from moto where idMoto=:id";
345     $params = array(
346         'id' => $id
347     );
348     $st = PrepareExecute($query, $params);
349 }
350
351 /**
352  * Get a user and his motorcycle
353  * @return JSON array values
354  */
355 function getUsersNMotorcycle() {
356     $array_response = [];
357     $query = "Select * from users natural join moto;";
358     $st = PrepareExecute($query);
359     while ($data = $st->fetch(PDO::FETCH_ASSOC)) {
360         $array_response[] = $data;
361     }
362     return $array_response;
363 }
364
365 /**
366  * Get the role of an user
367  * @param int $id user id
368  * @return int role of the user
369  */
370 function getUserRoleById($id) {
371     $query = "Select role from users where idUser=:id;";
372     $params = array(
373         'id' => $id
374     );
375     $st = PrepareExecute($query, $params);
376     $data = $st->fetch(PDO::FETCH_ASSOC);
377     return $data["role"];
378 }
379
380 /**
381  * Add sinuosity of a road
382  * @param int $idRoute
383  * @param int $sinueusite road sinuosity
384  */
385 function AddSinuosity($idRoute, $sinueusite) {
386     $query = "Update route SET Sinuosity=:sinuosity where idRoute=:idRoute;";
387     $params = array(
388         'sinuosity' => $sinueusite,
389         'idRoute' => $idRoute
```

```
390     );
391     PrepareExecute($query, $params);
392 }
393
394 /**
395  * Add elevation of a road
396  * @param int $idRoute
397  * @param int $elevation road elevation
398  */
399 function AddElevation($idRoute,$elevation) {
400     $query = "Update route SET Slope=:elevation where idRoute=:idRoute;";
401     $params = array(
402         'elevation' => $elevation,
403         'idRoute' => $idRoute
404     );
405     PrepareExecute($query, $params);
406 }
407
408 /**
409  * Add length of a road
410  * @param int $idRoute
411  * @param int length road length
412  */
413 function AddLength($idRoute,$length) {
414     $query = "Update route SET Length=:length where idRoute=:idRoute;";
415     $params = array(
416         'length' => $length,
417         'idRoute' => $idRoute
418     );
419     PrepareExecute($query, $params);
420 }
421
422 /**
423  * Get the most sinuous road
424  * @return int sinuosity value
425  */
426 function GetMostSinuousRoad() {
427     $query = "SELECT max(Sinuosity)*10000 AS Road FROM route;";
428     $st = PrepareExecute($query);
429     $val = $st->fetchAll(PDO::FETCH_ASSOC);
430     return (int) $val[0]["Road"];
431 }
432
433 /**
434  * Get the less sinuous road
435  * @return int sinuosity value
436  */
437 function GetLessSinuousRoad() {
438     $query = "SELECT min(Sinuosity)*10000 AS Road FROM route;";
439     $st = PrepareExecute($query);
440     $val = $st->fetchAll(PDO::FETCH_ASSOC);
441     return (int) $val[0]["Road"];
442 }
443
444 /**
445  * Get the most steepest road
446  * @return int steepest value
```

```
447  */
448  function GetMostSteepestRoad() {
449      $query = "SELECT max(Slope) AS Road FROM route;";
450      $st = PrepareExecute($query);
451      $val = $st->fetchAll(PDO::FETCH_ASSOC);
452      return (int) $val[0]["Road"];
453  }
454  /**
455   * Get the less steepest road
456   * @return int steepest value
457   */
458  function GetLessSteepestRoad() {
459      $query = "SELECT min(Slope) AS Road FROM route;";
460      $st = PrepareExecute($query);
461      $val = $st->fetchAll(PDO::FETCH_ASSOC);
462      return (int) $val[0]["Road"];
463  }
464
465  ?>
```

```
1  <?php
2  /*
3   * Author : Lucien Camuglia
4   * Description : contains all function used in ajax
5   * Date : April-june 2017
6   * Version : 1.0 LC BaseVersion
7   */
8
9  session_start();
10 include './functions.php';
11
12 if (isset($_REQUEST["fonction"])) {
13     $function = $_REQUEST["fonction"];
14
15     switch ($function) {
16         case "GetModel" : GetModel($_GET["Brand"]);
17             break;
18         case "GetYear" : GetYear($_GET["Brand"], $_GET["Model"]);
19             break;
20         case "UserExists" : UserExists($_GET["Username"]);
21             break;
22         case "GetMotorcycles" : GetMotorcycles($_GET["Brand"], $_GET["Model"],
23             $_GET["Year"], $_GET["Consumption"], $_GET["Tiredness"]);
24             break;
25         case "GetRoutePoints" : GetRoutePoints($_GET["idRoute"]);
26             break;
27         case "SaveNewRoute" : SaveNewRoute($_REQUEST["idRoute"], $_REQUEST["route"],
28             $_REQUEST["sinuosite"], $_REQUEST["elevation"], $_REQUEST["length"]);
29             break;
30         case "AddMotorcycle" : AddMotorcycle($_GET["Brand"], $_GET["Model"],
31             $_GET["Year"], $_GET["Consumption"], $_GET["Tiredness"]);
32             break;
33         case "UpdateUserRole" : UpdateUserRole($_GET["idUser"], $_GET["Role"]);
34             break;
35         case "GetUserRoleById" : GetUserRole($_GET["idUser"]);
36             break;
37         case "Download" : downloadRoute($_REQUEST["name"], $_REQUEST["path"]);
38             break;
39         case "FilterRoad" : FilterRoad($_GET["sinuosity"], $_GET["slope"],
40             $_GET["highway"], $_GET["time"]);
41             break;
42         case "GetRoutes" : GetRoutesJSON();
43             break;
44         case "GetRoadsInfos" : GetRoadsInfos($_GET["idRoute"]);
45             break;
46         case "CreateRoute" : CreateNewRoute($_GET["name"], $_GET["containsHighway"]);
47             break;
48         case "DeleteRoute" : DeleteRoute($_GET["idRoute"]);
49             break;
50         default : exit();
51             break;
52     }
53 } else {
54     header('location: ../index.php');
55 }
56 exit();
```

```
54
55 //Retrun motorcycle models by brand
56 function GetModel($brand) {
57     $query = "select distinct model from moto where Brand = :brand ";
58     $params = array('brand' => $brand);
59     $st = PrepareExecute($query, $params);
60     while ($data = $st->fetch(PDO::FETCH_ASSOC)) {
61         $array_response[] = $data["model"];
62     }
63     echo json_encode($array_response);
64 }
65
66 //Retrun motorcycle's year by brand and model
67 function GetYear($brand, $model) {
68     $query = "select distinct year from moto where Brand = :brand and model =
        :model order by year";
69     $params = array('brand' => $brand, 'model' => $model);
70     $st = PrepareExecute($query, $params);
71     while ($data = $st->fetch(PDO::FETCH_ASSOC)) {
72         $array_response[] = substr($data["year"], 0, 4);
73     }
74     echo json_encode($array_response);
75 }
76
77 //check if an user exists
78 function UserExists($username) {
79     $query = "select idUser from users where Username = :username";
80     $params = array('username' => $username);
81     $st = PrepareExecute($query, $params);
82     $array_response["exist"] = false;
83     while ($data = $st->fetch(PDO::FETCH_ASSOC))
84         $array_response["exist"] = true;
85
86     echo json_encode($array_response);
87 }
88
89 //Return all motorcycles according to input criteria
90 function GetMotorcycles($brand, $model, $year, $consumption, $tiredness) {
91     $query = "Select * from moto where Brand like :brand and model like :model and
        year like :year and consumption like :consumption and Tiredness like :tiredness
        order by Brand";
92     $params = array('brand' => $brand, 'model' => $model, 'year' => $year,
        'consumption' => $consumption, 'tiredness' => $tiredness);
93     $st = PrepareExecute($query, $params);
94     $array_response = "";
95     while ($data = $st->fetch(PDO::FETCH_ASSOC)) {
96         $array_response[] = $data;
97     }
98     echo json_encode($array_response);
99 }
100
101 //Get all points of a road
102 function GetRoutePoints($idRoute) {
103     $query = "Select * from place where idRoute=:idRoute order by position;";
104     $params = array('idRoute' => $idRoute);
105     $st = PrepareExecute($query, $params);
106     $array_response = "";
```

```
107     while ($data = $st->fetch(PDO::FETCH_ASSOC)) {
108         $array["Latitude"] = $data["latitude"];
109         $array["Longitude"] = $data["longitude"];
110         $array_response[] = $array;
111     }
112
113     echo json_encode($array_response);
114 }
115
116 //save a new route
117 function SaveNewRoute($idroute, $route, $sinueusite, $elevation, $length) {
118     //delete all places from this road
119     deletePlaces($idroute);
120
121     AddSinuosity($idroute, $sinueusite);
122     AddElevation($idroute, $elevation);
123     AddLength($idroute, $length);
124     $position = 0;
125     $fromweb = false;
126     if (is_string($route)) {
127         $route = json_decode($route);
128         $fromweb = true;
129     }
130     foreach ($route as $point) {
131         if ($fromweb) {
132             $lat = $point->lat;
133             $lon = $point->lng;
134         } else {
135             $lat = $point["latitude"];
136             $lon = $point["longitude"];
137         }
138         addPlaceToRoute($lat, $lon, $position++, $idroute);
139     }
140 }
141
142 //add a new motorcycle
143 function AddMotorcycle($brand, $model, $year, $consumption, $tiredness) {
144     $array_response = [];
145     $array_response["error"]["status"] = false;
146     $array_response["error"]["message"] = "";
147
148     if (!empty($brand) && !empty($model) && !empty($year) && !empty($consumption) &&
149         !empty($tiredness)) {
150         $query = "Insert Into moto (Brand, model, year, consumption, Tiredness)
151             values (:brand, :model, :year, :consumption, :tiredness)";
152         $params = array('brand' => $brand, 'model' => $model, 'year' => $year .
153             '-01-01', 'consumption' => $consumption, 'tiredness' => $tiredness);
154         PrepareExecute($query, $params);
155     } else {
156         $array_response["error"]["status"] = true;
157         $array_response["error"]["message"] = "Please fill all field";
158     }
159
160     echo json_encode($array_response);
161 }
162
163 //update a user role
```

```
161 function UpdateUserRole($idUser, $idRole) {
162     $array_response = [];
163     $array_response["error"]["status"] = false;
164     $array_response["error"]["message"] = "";
165
166     //check if the user is admin
167     if (isset($_SESSION["role"]) && $_SESSION["role"] == 1) {
168
169         $query = "UPDATE users SET role = :role WHERE idUser=:idUser;";
170         $params = array('role' => $idRole, 'idUser' => $idUser);
171         $st = PrepareExecute($query, $params);
172         $Newrole;
173         switch (getUserRoleById($idUser)) {
174             case 1: $Newrole = "Administrator";
175                 break;
176             case 2: $Newrole = "User";
177                 break;
178             case 3: $Newrole = "Ban";
179                 break;
180             default : $Newrole = "error";
181                 break;
182         }
183         $array_response["datas"]["NewRole"] = $Newrole;
184     } else {
185         $array_response["error"]["status"] = true;
186         $array_response["error"]["message"] = "Rights needed";
187     }
188
189     echo json_encode($array_response);
190 }
191
192 //get user role by id
193 function GetUserRole($idUser) {
194     $result;
195     switch (getUserRoleById($idUser)) {
196         case 1: $result = "Administrator";
197             break;
198         case 2: $result = "User";
199             break;
200         case 3: $result = "Ban";
201             break;
202     }
203     echo json_encode($result);
204 }
205
206 //download prepare to download and return filename
207 function downloadRoute($name, $path) {
208     $file = Path2Gpx($name, $path);
209     echo $file;
210 }
211
212 //return a filtered road
213 function FilterRoad($sinuosity, $slope, $highway, $time) {
214     $query = "Select * from route natural join users where Sinuosity <= :sinuosity
215 AND Slope <= :slope AND Time <= :time AND highway = :highway ORDER BY RouteName;";
216     // $query = "Select * from route where Sinuosity <= 135 AND Slope <= 0 AND Time
217     <= 1 AND highway = 0;";
```

```
216     $params = array(
217         "sinuosity" => $sinuosity / 10000,
218         "slope" => $slope,
219         "time" => $time,
220         "highway" => ($highway == "true") ? '1' : '0');
221     $st = PrepareExecute($query, $params);
222     $array_response = [];
223     while ($data = $st->fetch(PDO::FETCH_ASSOC)) {
224         $array["RouteName"] = $data["RouteName"];
225         $array["idRoute"] = $data["idRoute"];
226         $array["Username"] = $data["Username"];
227         $array_response[] = $array;
228     }
229
230     echo json_encode($array_response);
231 }
232
233 //return all routes in JSON
234 function GetRoutesJSON() {
235     $routes = GetRoutes();
236     echo json_encode($routes);
237 }
238
239 //get roads info
240 function GetRoadsInfos($idRoute) {
241     $query = "Select * from route where idRoute = :idRoute";
242     $params = Array("idRoute" => $idRoute);
243     $st = PrepareExecute($query, $params);
244     $datas = $st->fetch(PDO::FETCH_ASSOC);
245     $array_response["Length"] = $datas["Length"] / 1000;
246     $array_response["Highway"] = $datas["highway"];
247     $array_response["Time"] = $datas["Time"];
248     $array_response["Sinuosity"] = (int) (($datas["Sinuosity"] * 10000 /
249     GetMostSinuousRoad()) * 10);
250     $array_response["Slope"] = round($datas["Slope"] / GetMostSteepestRoad() * 10);
251
252     //if th euser is connected, return motorcycle consumption if not, return 0
253     if (isset($_SESSION["id"])) {
254         $query = "select * from moto natural join users where idUser = :idUser ";
255         $params = Array("idUser" => $_SESSION["id"]);
256         $st = PrepareExecute($query, $params);
257         $datas = $st->fetch(PDO::FETCH_ASSOC);
258         $motorcycleConsumption = $datas["consumption"];
259     } else {
260         $motorcycleConsumption = 0;
261     }
262
263     $array_response["MotorcycleConsumption"] = (int) $motorcycleConsumption;
264
265     echo json_encode($array_response);
266 }
267
268 //add a route in database and return id
269 function CreateNewRoute($name, $containsHighway) {
270     $id = CreateRoute($name, $_SESSION["id"], $containsHighway);
271     echo json_encode($id);
272 }
```



```
272
273 //delete a route
274 function DeleteRoute($idRoute) {
275     $query = "Delete from place where idRoute=:idRoute;";
276     $params = array("idRoute" => $idRoute);
277     PrepareExecute($query, $params);
278
279     $query = "Delete from route where idRoute=:idRoute;";
280     PrepareExecute($query, $params);
281
282     echo json_encode("Success");
283 }
284
```

```
1  <?php
2  /*
3   * Author : Lucien Camuglia
4   * Description : Delete a motorcycle
5   * Date : April-june 2017
6   * Version : 1.0 LC BaseVersion
7   */
8
9  session_start();
10 include_once './functions.php';
11
12 if(isset($_SESSION["role"]) && $_SESSION["role"]==1) {
13     if(isset($_POST["delete"])) {
14         if(isset($_POST["toDelete"])) {
15             deleteMotorcycle($_POST["toDelete"]);
16             header('location: ../manageMoto.php');
17             exit();
18         }
19     }
20 }
21
22 header('location: ../index.php');
23
24 exit();
```

```
1  <?php
2  /*
3   * Author : Lucien Camuglia
4   * Description : Download a file
5   * Date : April-june 2017
6   * Version : 1.0 LC BaseVersion
7   */
8
9  if (isset($_GET['file'])) {
10     $file = $_GET['file'];
11     //verify if file exists
12     if (file_exists($file) && is_readable($file)) {
13         //prepare to download
14         header('Content-Type: application/octet-stream');
15         header("Content-Disposition: attachment; filename=\"$file\"");
16         //download the file
17         readfile($file);
18         //delete it
19         unlink($file);
20     } else {
21         //display an 404 error
22         header("HTTP/1.0 404 Not Found");
23         echo "<h1>Error 404: File Not Found: <br /><em>$file</em></h1>";
24     }
25 }
26
27
```

```
7      */
8      include_once './functions.php';
```

```
1  <?php
2  /*
3   * Author : Lucien Camuglia
4   * Description : Import a GPX file
5   * Date : April-june 2017
6   * Version : 1.0 LC BaseVersion
7   */
8  include_once './functions.php';
9  session_start();
10 //verify if import has been pressed
11 if (isset($_POST["import"])) {
12     //veriy if a file is selected
13     if (isset($_FILES["GpxFile"])) {
14         //verify if the file format is gpx
15         if (pathinfo($_FILES["GpxFile"]["name"])[ "extension" ] == "gpx") {
16             //set if the road contains highway
17             $containsHighway = isset($_POST["highway"]);
18             // save the file on the server
19             $idRoute = Gpx2Sql($_FILES["GpxFile"]["tmp_name"], $_SESSION["id"],
20                               $containsHighway);
21             //redirect the user
22             header("location: ../itineraire.php?NewIti&id=$idRoute");
23         } else {
24             header("location:
25                 ../itineraire.php?modal=modalimport&error=extension%20GPX%20nécessaire");
26         }
27     }
28 }
29
30
```