



INFORMATION SYSTEMS AND DATABASES

PROJECT PART 3

GROUP 34

André Ferreira - 81715

Bruno Alves - 81684

Tiago Ferreira - 81579

December 7, 2018

A Web Application Using the Database

For the final project of the course Information Systems and Databases, the previously defined veterinary database is paired with a web application solution, in order to have a user friendly interaction with the data. The homepage is hosted in the url <http://web.tecnico.ulisboa.pt/ist181579/sibd/proj/HomePage.php>, which shows the interface seen on figure 1.

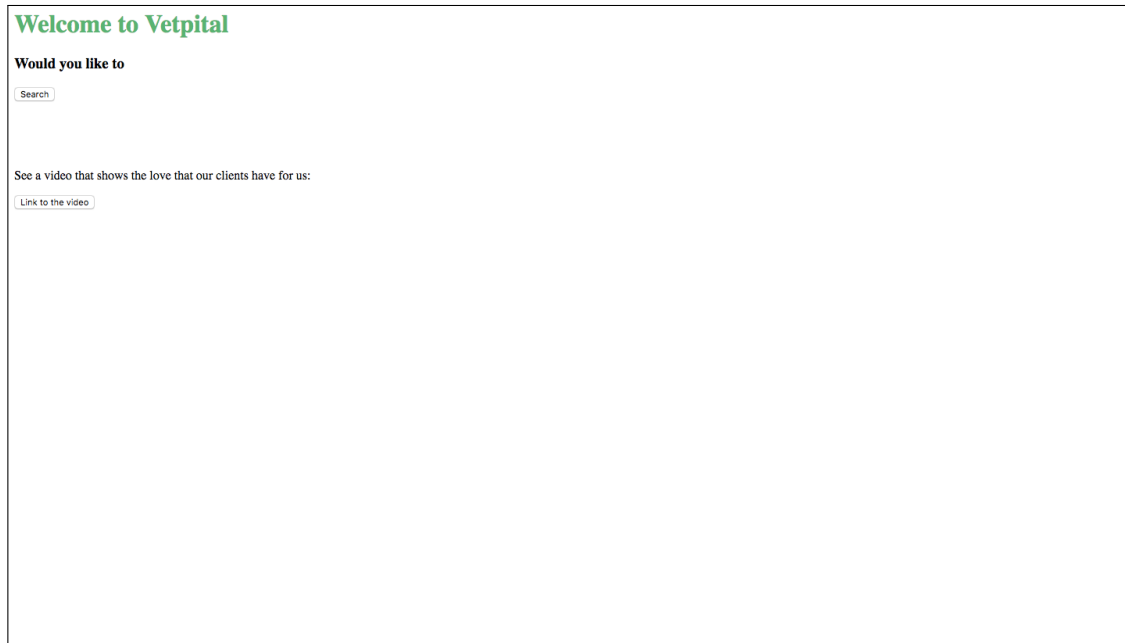


Figure 1: Homepage interface of the veterinary database.

The PHP code that generates the homepage is shown bellow.

```
1 <html>
2     <body>
3         <h1 style="color:MediumSeaGreen;">Welcome to Vetpital</h1>
4         <h3>Would you like to</h3>
5         <form action="Search.php" method="get">
6             <p><input type="submit" value="Search"/></p>
7         </form>
8         <br>
9         <br>
10        <br>
11        <form action="https://www.instagram.com/p/Bq1XlcjAvr3/?
            utm_source=ig_share_sheet&igshid=143zsnmqhmbij" method="
            get">
12            <p>See a video that shows the love that our clients
                have for us:</p>
13            <p><input type="submit" value="Link to the video"/></p>
```

```
14                                     >
15                                     </form>
16 </body>
17 </html>
```

In the following pages, the main features of the application are demonstrated, according to the project requirements. Non trivial decisions are explained in more detail.

1. Searching for animals

When choosing the option "Search" from the home page, shown in figure 1, a new page appears with options to search for an animal, by introducing its name and, optionally, define the owner name to filter the results and the VAT of a client, to see if it has any register of ever bringing the animal to a consult. To see the search results, one has to press the button "Search", as seen in figure 2. Alternatively, the user may register a new user, by pressing the button "Register new client".


The screenshot shows a web interface for 'Vetpital'. At the top left is the logo 'Vetpital' in green. Below it is the heading 'Please introduce the following informations'. There are three input fields: 'Animal name:' with a text box, 'Owner name:' with a text box, and 'Client VAT:' with a text box containing '12345678' and a small blue icon to its right. Below these fields are two buttons: 'Search' and 'Register new client'.

Figure 2: Interface shown when searching for animals in the veterinary database.

Vetpital

Please introduce the following informations

Animal name:

Owner name:

Client VAT ☒ 12345678
☐ 14745000
☐ 35263686
☐ 45620852
☐ 53250530
☐ 54609438
☐ 63250157
☐ 65856663
☐ 69630596
☐ 78526209
☐ 91970633
☐ 98520520
☐ 98585856
☐ 91919191

Search

Register new

Figure 3: Selecting a client VAT while searching for animals in the veterinary database.

The PHP code that generates the animal search web page is shown bellow.

```

1 <html>
2     <body>
3         <h1><a href = "http://web.tecnico.ulisboa.pt/ist181579/sibd/
          proj/HomePage.php" style="color:MediumSeaGreen;text-
          decoration: none;">Vetpital</a></h1>
4         <form action="ResultSearch.php" method="post">
5             <h3>Please introduce the following informations</h3>
6             <p>Animal name: </p>
7             <input type="text" name="animal_name"/>
8             <p>Owner name: </p>
9             <input type="text" name="owner_name"/>
10            <p>Client VAT:
11                <select name="client_vat">
12 <?php
13            # Establishing the connection with the database
14            $host = "db.tecnico.ulisboa.pt";
15            $user = "ist181579";
16            $pass = "utfv5127";
17            $dsn = "mysql:host=$host;dbname=$user";
18            try
19            {
20                $connection = new PDO($dsn, $user, $pass);

```

```

21     }
22     catch(PDOException $exception)
23     {
24         echo("<p>Error: ␣");
25         echo($exception->getMessage());
26         echo("</p>");
27         exit();
28     }
29
30     # Get the existent clients VAT
31     $sql = "SELECT_VAT_FROM_client_ORDER_BY_VAT";
32     $result = $connection->query($sql);
33     if ($result == FALSE)
34     {
35         $info = $connection->errorInfo();
36         echo("<p>Error: ␣{ $info[2]} </p>");
37         exit();
38     }
39     foreach($result as $row)
40     {
41         $client_vat = $row['VAT'];
42         echo("<option_value=␣$client_vat␣>$client_vat </
43         option>");
44     }
45     $connection = null;
46     ?>
47
48         </select>
49
50         </p>
51         <p><input type="submit" value="Search"/></p>
52     </form>
53     <form action="new_client.php" method="post">
54         <p><input type="submit" value="Register_new_client"
55         /></p>
56         <input type="hidden" value="Search.php" name="from"/>
57     </form>
58 </body>
59 </html>

```

When the indicated parameters match with animals registered in the database, a new page is displayed. In figure 4, an example is seen, after searching for animals named "Bobi", that might have been brought in to a consult by the client with VAT "12345678". In this case, one can see three animals named "Bobi", one belonging to the owner "Maria Repolho", which has VAT "12345678" and has taken her animal at least once to a consult (which is why the column "Consults with this Client" appears with value "Yes"), one belonging to "Maria Albertina", that was also brought in at least once by Maria Repolho that has VAT "12345678",

and one belonging to owner "John Smith". If a filter is applied on the owner name, in the input box from the interface of figure 2, the application only displays the results which have associated owners with names that contain the specified input name. For instance, when writing "Maria" in the owner name of the animal search, now only the "Bobi"'s belonging to "Maria Repolho" or "Maria Albertina" are returned (figure 5).

Vetpital
Animal name: Bobi
Owner name:
Client VAT: 12345678
Search results:

Animal Name	Owner VAT	Owner Name	Consults with this Client
<input type="text" value="Bobi"/>	12345678	Maria Repolho	Yes
<input type="text" value="Bobi"/>	35263686	Maria Albertina	Yes
<input type="text" value="Bobi"/>	65856663	John Smith	No

Select the animal that you are looking for

Figure 4: Interface shown when searching for animals in the veterinary database.

Vetpital

Animal name: Bobi
Owner name: Maria
Client VAT: 12345678

Search results:

Animal Name	Owner VAT	Owner Name	Consults with this Client
Bobi	12345678	Maria Repolho	Yes
Bobi	35263686	Maria Albertina	Yes

Select the animal that you are looking for

Figure 5: Interface shown when searching for animals in the veterinary database, belonging to someone with "Maria" on their name.

The PHP code that generates this animal search results page is shown bellow.

```

1 <html>
2     <body>
3         <h1><a href = "http://web.tecnico.ulisboa.pt/ist181579/sibd/
           proj/HomePage.php" style="color:MediumSeaGreen;text-
           decoration: none;">Vetpital</a></h1>
4 <?php
5     # Establishing the connection with the database
6     $host = "db.tecnico.ulisboa.pt";
7     $user = "ist181579";
8     $pass = "utfv5127";
9     $dsn = "mysql:host=$host;dbname=$user";
10    try
11    {
12        $connection = new PDO($dsn, $user, $pass);
13    }
14    catch(PDOException $exception)
15    {
16        echo("<p>Error: _");
17        echo($exception->getMessage());
18        echo("</p>");
19        exit();

```

```

20     }
21
22     # Data received
23     $animal_name = $_REQUEST[ 'animal_name' ];
24     $owner_name = $_REQUEST[ 'owner_name' ];
25     $clientVat = $_REQUEST[ 'client_vat' ];
26     echo("<p>Animal_name: { $animal_name }</p>");
27     echo("<p>Owner_name: { $owner_name }</p>");
28     echo("<p>Client_VAT: { $clientVat }</p>");
29     echo("<h3>Search_results:</h3>");
30
31     # Check for animals that match the search keys
32     $sql = "CALL_SearchConsultName(' $animal_name ', ' $owner_name ', '
        $clientVat ' );";
33     $result = $connection->query( $sql );
34     if ( $result == FALSE )
35     {
36         $info = $connection->errorInfo();
37         echo("<p>Error: { $info[2] }</p>");
38         exit();
39     }
40     if ( $result->rowCount() == 0 )
41     {
42         echo("<h4>There was no animal found<form action='new_animal.
            php' method='post'>\n<p><input type='submit' value='
            Register_New_Animal' /><input type='hidden' value='{
            $animal_name }' name='animal_name' /><input type='hidden'
            value='{ $owner_name }' name='owner_name' /></p>\n</form></h4
            >");
43
44     } else {
45         # Print the results from the search
46         echo("<table border='2'>");
47         echo("<tr><td>Animal_Name</td><td>Owner_VAT</td><td>Owner_Name
            </td><td>Consults_with_this_Client</td></tr>");
48         foreach( $result as $row )
49         {
50             echo("<tr align='center'>");
51             echo("<td align='center'><form action='
                display_consults.php' method='post'>\n");
52             echo("<input type='hidden' name='animal' value='{ $row
                [ 'animal' ] }' />\n");
53             echo("<input type='hidden' name='owner' value='{ $row [ '

```



```

54         owner ']]' />\n");
55     echo("<input_type='hidden' _name='ownerVat' _value='{
        $row[ 'vat ']]' />\n");
56     echo("<input_type='hidden' _name='clientVat' _value='{
        $clientVat}' />\n");
57     echo("<input_type='submit' _value='{ $row[ 'animal ']]' />\n
        n</form></td>");
58     echo("<td>{ $row[ 'vat ']} </td>");
59     echo("<td>{ $row[ 'owner ']} </td>");
60
61     # Establishing the connection with the database
62     $host = "db.tecnico.ulisboa.pt";
63     $user = "ist181579";
64     $pass = "utfv5127";
65     $dsn = "mysql:host=$host;dbname=$user";
66     try
67     {
68         $connection_2 = new PDO($dsn, $user, $pass);
69     }
70     catch(PDOException $exception)
71     {
72         echo("<p>Error: _");
73         echo($exception->getMessage());
74         echo("</p>");
75         exit();
76     }
77
78     $sql_2 = "CALL_consult_with_client(' $animal_name', '{
        $row[ 'vat ']}', ' $clientVat ');";
79     $this_result = $connection_2->query($sql_2);
80     if ($this_result == FALSE)
81     {
82         $info = $connection->errorInfo();
83         echo("<p>Error: _{ $info[2]} </p>");
84         exit();
85     }
86     if ($this_result->rowCount() == 0)
87     {
88         $consults_with_client = 0;
89     } else {
90         $consults_with_client = 1;
91     }

```

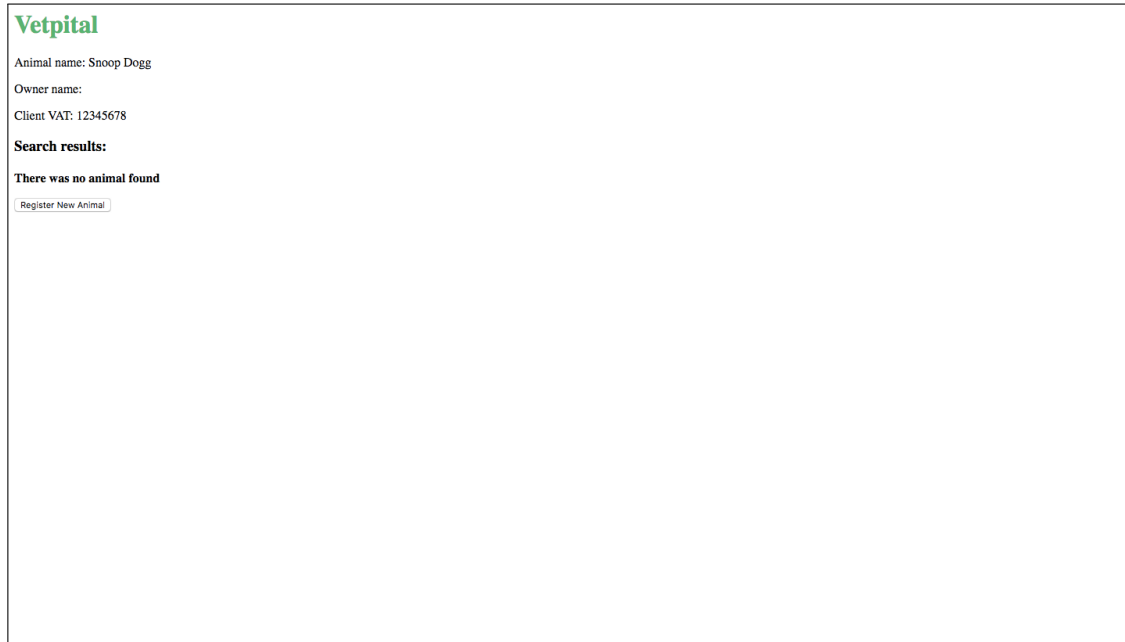
```

92             $this_result = NULL;
93             if( $consults_with_client == 1){echo("<td>Yes</td>");}
94             else{echo("<td>No</td>");}
95             echo("</tr>");
96             $connection_2=NULL;
97         }
98         echo("</table>");
99         echo("<h4>Select_the_animal_that_you_are_looking_for</h4>");
100         $result = NULL;
101     }
102
103     $connection = null;
104     ?>
105     </body>
106 </html>

```

2. Adding new animals

When searching for an animal name that doesn't exist, a message indicating that no results are found is displayed, as figure 6 shows. In this scenario, it's possible to add the animal to the database, with the name used in the search, by pressing the button "Register New Animal".



The screenshot shows a web interface for 'Vetpital'. At the top left is the logo 'Vetpital' in green. Below it, the following text is displayed: 'Animal name: Snoop Dogg', 'Owner name:', and 'Client VAT: 12345678'. Underneath this is a section titled 'Search results:' followed by the message 'There was no animal found'. At the bottom of this section is a button labeled 'Register New Animal'.

Figure 6: Message and button displayed when no animal with the specifyied name is found.

When adding a new animal, first a form to introduce the animal's basic information appears (figure 7). In here, the owner's name, which must be already present in the database, is indicated, as well as the animal's species, colour, gender and birth year. In the case of the species data, there are multiple options available. Initially, the most generic species are selected (e.g. mammal, bird, etc). This is done in SQL by finding the species names that only appear on the right side (second argument) of the "generalization_species" relation. Then, the user has to click the button "Specialize", in order to see more specific species (e.g. dog, cat, etc). This translates in SQL code to find all the names that appear to the left of the selected generic species, in the "generalization_species" relation. In the case of a dog for instance, this specialization goes all the way from mammal to the dog's race (e.g. Estrela mountain dog, boxer, etc), as seen in figures 7, 8, 9 and 10.

Vetpital

New Animal

Animal Name

Owner Name

VAT - Full Name

Species

Colour

Gender ☐ Male ☐ Female

Birth Year

Figure 7: Initial interface for adding a new animal to the database.

Vetpital

New Animal

Animal Name

Owner Name

VAT - Full Name

Species

Colour

Gender ☐ Male ☐ Female

Birth Year

Figure 8: New animal page after specializing the species from "Mammal".

After creating the animal record, the user is taken to the page corresponding to the process of adding new consults. This segment is explained in more detail in section 5.

Vetpital

New Animal

Animal Name:

Owner Name:

VAT - Full Name:

Species:

Colour:

Gender: ☐ Male ☐ Female

Birth Year:

Figure 9: New animal page after specializing the species from "Dog".

Vetpital

New Animal

Animal Name:

Owner Name:

VAT - Full Name:

Species:

Colour:

Gender: ☐ Male ☐ Female

Birth Year:

There is no more possible specialization

Figure 10: No more specializations found after getting "Estrela Mountain Dog" species.

The PHP code that supports this animal registration page is shown bellow.

```
1 <html>
2     <body>
```

```

3      <h1><a href = "http://web.tecnico.ulisboa.pt/ist181579/sibd/
      proj/HomePage.php" style="color:MediumSeaGreen;text-
      decoration:underline;">Vetpital</a></h1>
4      <form action="" method="post">
5          <h3>New Animal</h3>
6          <table style="width:35%">
7              <tr>
8                  <td><p>Animal Name</p></td>
9      <?php
10                  $animal_name = $_REQUEST[ 'animal_name'
11                      ];
12                  if (!empty($_POST[ 'animal_name' ])) {
13                      $animal_name = $_POST[ 'animal_name'
14                          ];}
15                  echo("<td><input type='text' value='{
16                      $animal_name }' name='animal_name
17                      ' /></td>");
18  ?>
19                  <td></td>
20                  </tr>
21                  <tr>
22                      <td><p>Owner Name</p></td>
23                      <?php
24                          $owner_name = $_REQUEST[ 'owner_name' ];
25                          if (!empty($_POST[ 'client_name' ])) {
26                              $owner_name = $_POST[ 'client_name'
27                                  ];}
28                          echo("<td><input type='text' value='{
29                              $owner_name }' name='client_name
30                              ' /></td>");
31  ?>
32                      <td><input type='submit' value="Get _
33                          VAT's" /></td>
34                  </tr>
35              </table>
36          </form>
37          <table style="width:35%">
38              <td><p>VAT - Full Name</p></td>
39              <td>
40                  <form action="" method="post">
41                      <select name='client_vat_option'>
42      <?php
43          # Establishing the connection with the database

```

```

34 $host = "db.tecnico.ulisboa.pt";
35 $user = "ist181579";
36 $pass = "utfv5127";
37 $dsn = "mysql:host=$host;dbname=$user";
38 try
39 {
40     $connection = new PDO($dsn, $user, $pass);
41 }
42 catch(PDOException $exception)
43 {
44     echo("<p>Error: _");
45     echo($exception->getMessage());
46     echo("</p>");
47     exit();
48 }
49
50 # Get the existent client's VAT
51 $client_name = $_POST['client_name'];
52 $animal_name = $_POST['animal_name'];
53 if(!empty($_POST['client_vat_option'])) {
54     $sql="SELECT _name_ as _NAME_ FROM _person_ WHERE _VAT='{
55         _POST['client_vat_option']_}';";
56     $result = $connection->query($sql);
57     if ($result == FALSE)
58     {
59         $info = $connection->errorInfo();
60         echo("<p>Error: _{$info[2]}</p>");
61         exit();
62     }
63     foreach($result as $row)
64     {
65         $client_name_procedure = $row['NAME'];
66         echo("<option_value=\"{$POST['
67             client_vat_option']}\">>{$POST['
68             client_vat_option']_}_
69             $client_name_procedure</option>");
70         echo("<input_type='hidden' _name='
71             client_name_procedure' _value='{
72             $client_name_procedure}' _/>");
73     }
74 } else {
75     $sql ="CALL _get_vat_from_name(' $client_name ');";
76     $result = $connection->query($sql);

```

```

71         if ( $result == FALSE)
72         {
73             $info = $connection->errorInfo();
74             echo("<p>Error: ␣{ $info[2]} </p>");
75             exit();
76         }
77         foreach( $result as $row)
78         {
79             $client_vat_procedure = $row[ 'VAT' ];
80             $client_name_procedure = $row[ 'NAME' ];
81             echo("<option␣value=␣\" $client_vat_procedure␣\">
                $client_vat_procedure␣␣
                $client_name_procedure </option>");
82         }
83     }
84     $connection = null;
85 ?>
86
87         </select>
88     </td>
89     <td></td>
90 </tr>
91 <tr>
92     <td><p>Species </p></td>
93     <td><select␣name=␣" species_name">
94 <?php
95 if (!isset($_POST[ 'specialization_request' ])) {
96     # Establishing the connection with the database
97     $host = "db.tecnico.ulisboa.pt";
98     $user = "ist181579";
99     $pass = "utfv5127";
100    $dsn = "mysql:host=$host;dbname=$user";
101    try
102    {
103        $connection = new PDO($dsn, $user, $pass);
104    }
105    catch(PDOException $exception)
106    {
107        echo("<p>Error: ␣");
108        echo($exception->getMessage());
109        echo("</p>");
110        exit();
111    }

```



```

112
113 # Get the existent client's VAT
114 $sql ="CALL_more_general()";
115 $result = $connection->query($sql);
116 if ($result == FALSE)
117 {
118     $info = $connection->errorInfo();
119     echo("<p>Error: ␣{$info[2]} </p>");
120     exit();
121 }
122 foreach($result as $row)
123 {
124     $species_name_ = $row['species_name'];
125     echo("<option_value=␣\"$species_name_␣\">$species_name_</option>
126         ");
127 }
128 } else {
129     # Establishing the connection with the database
130     $host = "db.tecnico.ulisboa.pt";
131     $user = "ist181579";
132     $pass = "utfv5127";
133     $dsn = "mysql:host=$host;dbname=$user";
134     try
135     {
136         $connection = new PDO($dsn, $user, $pass);
137     }
138     catch(PDOException $exception)
139     {
140         echo("<p>Error: ␣");
141         echo($exception->getMessage());
142         echo("</p>");
143         exit();
144     }
145
146     # Get the existent client's VAT
147     $sql ="CALL_more_specific('{$_POST['species_name']}')";
148     $result = $connection->query($sql);
149     if ($result == FALSE)
150     {
151         $info = $connection->errorInfo();
152         echo("<p>Error: ␣{$info[2]} </p>");
153         exit();

```

```

154     }
155     if ($result->rowCount() == 0) {
156         echo("<option_value=\"{".$POST['species_name']}">{".$POST['species_name']}</option>");
157         echo("<input_type='hidden' _value='yes' _name='final_specialization' />");
158         echo("<input_type='hidden' _value='{$POST['client_vat_option']}' _name='client_vat_option' />");
159     } else {
160
161         foreach ($result as $row)
162         {
163             $species_name_ = $row['species_name'];
164             echo("<option_value=\" $species_name_\">$species_name_</option>");
165         }
166     }
167     $connection = null;
168 }
169 ?>
170
171         </select>
172         <td><input type='submit' value="Specialize" name="specialization_request" /></td>
173 <?php
174     echo("<input_type='hidden' _value='{$POST['animal_name']}' _name='animal_name' />");
175     echo("<input_type='hidden' _value='{$POST['client_name']}' _name='client_name' />");
176 ?>
177         </tr>
178     </form>
179 </table>
180 <?php
181     if ($POST['final_specialization'] == 'yes') {
182         echo("<p>There is no more possible specialization </p>");
183     }
184 ?>
185
186     <table style="width:35%">
187         <form action="" method="post">
188             <tr>
189                 <td><p>Colour</p></td>

```

```

190         <td><input type='text' name='colour'
191             /></td>
192     </tr>
193     <tr>
194         <td><p>Gender</p></td>
195         <td><input type="radio" value="Male"
196             name="gender">Male
197         <input type="radio" name="gender"
198             value="Female"/>Female</td>
199     </tr>
200 </table>
201 <p><input type="submit" value="Register new Animal"
202     /></p>
203 <?php
204     echo("<input type='hidden' value='{$_POST['animal_name']}' name='
205         animal_name'/>");
206     echo("<input type='hidden' name='client_name_procedure' value='{$_POST
207         ['client_name_procedure']}' />");
208     echo("<input type='hidden' value='{$_POST['species_name']}' name='
209         species_name'/>");
210     echo("<input type='hidden' value='{$_POST['client_vat_option']}' name
211         ='client_vat_option'/>");
212 ?>
213     </form>
214 <?php
215     $all_filled=1;
216
217     if(empty($_POST['animal_name'])) $all_filled=0;
218     if(empty($_POST['species_name'])) $all_filled=0;
219     if(empty($_POST['client_vat_option'])) $all_filled=0;
220     if(empty($_POST['colour'])) $all_filled=0;
221     if(empty($_POST['gender'])) $all_filled=0;
222     if(empty($_POST['birth_year'])) $all_filled=0;
223
224     if($all_filled){
225         # Establishing the connection with the database
226         $host = "db.tecnico.ulisboa.pt";

```

```

224 $user = "ist181579";
225 $pass = "utfv5127";
226 $dsn = "mysql:host=$host;dbname=$user";
227 try
228 {
229     $connection = new PDO($dsn, $user, $pass);
230 }
231 catch(PDOException $exception)
232 {
233     echo("<p>Error: _");
234     echo($exception->getMessage());
235     echo("</p>");
236     exit();
237 }
238
239 $sql = "INSERT INTO animal(name,VAT,species_name ,colour ,gender
        ,birth_year) _VALUES(?, ?, ?, ?, ?, ?)";
240 $sth = $connection->prepare($sql);
241
242 if ($sth == FALSE) {
243     $info = $connection->errorInfo();
244     echo("<p>Error: _{ $info[2]} </p>");
245     exit();
246 } else {
247     $sth->execute(array($POST['animal_name'], $POST['
        client_vat_option'], $POST['species_name'], $POST[
        'colour'], $POST['gender'], $POST['birth_year']));
248     $result = $sth->num_rows;
249 }
250 $connection = null;
251
252
253 #Sends the person to register a new consult
254 echo("<form_id='jsform' _action='register_consult.php' _method='
        post'>");
255 echo("<input_type='hidden' _value='{ $POST['animal_name']}' _
        name='animal'/>");
256 echo("<input_type='hidden' _value='{ $POST['client_vat_option
        ']} _name='owner_vat'/>");
257 echo("<input_type='hidden' _value='{ $POST['client_vat_option
        ']} _name='client_vat'/>");
258 echo("</form>");
259 echo("<script_type='text/javascript'>");

```

```

260         echo("document.getElementById('jsform').submit();");
261     echo("</script>");
262 }
263
264 ?>
265     </body>
266 </html>

```

3. Searching for consults

After selecting an animal from the animal search page (section 1), the user is sent to the web page presented in figure 11, where it is shown some information from the selection process with the purpose of confirmation. Then are also listed the consults involving the considered animal, where it is possible to add a new blood test for a specific consult by the button "New" (see section 6 for more details). If the user desires to know more details about a specific consult, the button with the date should be pressed. Last but certainly not least, there is the button "Register New Consult" which opens a new page to support this functionality.



Vetpital

Animal name: PiuPiu
 Owner name: Maria Albertina
 Owner VAT: 35263686
 Client VAT: 12345678

Consults

Date	Veterinary VAT	Veterinary Name	Add blood test
2018-08-01 12:45:02	12032014	Jacinto Leite	<input type="button" value="New"/>
2018-05-15 16:35:15	12032014	Jacinto Leite	<input type="button" value="New"/>

Figure 11: Retrived consults after selecting the animal named "PiuPiu", of owner "Maria Albertina", from the animal search page (section 1).

The PHP code that generates this consult search page is shown bellow.

```

1 <html>
2     <body>
3         <h1><a href = "http://web.tecnico.ulisboa.pt/ist181579/sibd/
          proj/HomePage.php" style="color:MediumSeaGreen;text-
          decoration: none;">Vetpital</a></h1>

```

```

4 <?php
5     # Establishing the connection with the database
6     $host = "db.tecnico.ulisboa.pt";
7     $user = "ist181579";
8     $pass = "utfv5127";
9     $dsn = "mysql:host=$host;dbname=$user";
10    try
11    {
12        $connection = new PDO($dsn, $user, $pass);
13    }
14    catch(PDOException $exception)
15    {
16        echo("<p>Error: _");
17        echo($exception->getMessage());
18        echo("</p>");
19        exit();
20    }
21
22    # Data received
23    $animal_name = $_REQUEST['animal'];
24    $owner_name = $_REQUEST['owner'];
25    $ownerVat = $_REQUEST['ownerVat'];
26    $clientVat = $_REQUEST['clientVat'];
27    echo("<p>Animal_name: _{$animal_name}</p>");
28    echo("<p>Owner_name: _{$owner_name}</p>");
29    echo("<p>Owner_VAT: _{$ownerVat}</p>");
30    echo("<p>Client_VAT: _{$clientVat}</p>");
31    echo("<h3>Consults</h3>");
32
33    # Check for animals that match the search keys
34    $sql = "CALL_AnimalConsultsInfo('{$animal_name}','{$ownerVat}')";
35    $result = $connection->query($sql);
36    if ($result == FALSE)
37    {
38        $info = $connection->errorInfo();
39        echo("<p>Error: _{$info[2]}</p>");
40        exit();
41    }
42    # Print list of consults involving that animal
43    echo("<table border=\"2\">");
44    echo("<tr><td>Date</td><td>Veterinary_VAT</td><td>Veterinary_Name</td>
45        <td>Add_blood_test</td></tr>");
46    foreach($result as $row)

```

```

46 {
47     echo("<tr_align='center'>");
48     echo("<td_align='center'><form_action='consult_info.php' _
        method='post'>\n");
49     echo("<input_type='hidden' _name='animal' _value='{ $animal_name
        }'/>\n");
50     echo("<input_type='hidden' _name='owner_vat' _value='{ $ownerVat
        }'/>\n");
51     echo("<input_type='hidden' _name='date' _value='{ $row[ ' date
        ']}'/>\n");
52     echo("<input_type='submit' _value='{ $row[ ' date ']}'/>\n</form></
        td>");
53     echo("<td>{ $row[ ' vatVet ']} </td>");
54     echo("<td>{ $row[ ' nameVet ']} </td>");
55     echo("<td_align='center'><form_action='new_blood_test.php' _
        method='post'>\n");
56     echo("<input_type='hidden' _name='animal_name' _value='{
        $animal_name }'/>\n");
57     echo("<input_type='hidden' _name='owner_name' _value='{
        $owner_name }'/>\n");
58     echo("<input_type='hidden' _name='owner_vat' _value='{ $ownerVat
        }'/>\n");
59     echo("<input_type='hidden' _name='date' _value='{ $row[ ' date
        ']}'/>\n");
60     echo("<input_type='submit' _value=New>\n</form></td>");
61     echo("</tr>");
62 }
63 echo("</table>");
64
65 # New consult button
66 echo("<p></p>");
67 echo("<form_action='register_consult.php' _method='post'>\n");
68 echo("<input_type='hidden' _name='animal' _value='{ $animal_name }'/>\n");
69 echo("<input_type='hidden' _name='owner' _value='{ $owner_name }'/>\n");
70 echo("<input_type='hidden' _name='owner_vat' _value='{ $ownerVat }'/>\n");
71 echo("<input_type='hidden' _name='client_vat' _value='{ $clientVat }'/>\n"
    );
72 echo("<input_type='submit' _value='Register_new_Conult' />\n</form></td
    >");
73
74 $result = NULL;
75 $connection = null;
76 ?>

```

```

77         </body>
78 </html>

```

4. Consult info

When the button with the date is pressed, the flow continues to the page shown in figure 12. Here, it is printed the primary information of a consult just for confirmation. Then, as required, the information about the animal (Species name, Colour, Gender, Age, Weight), the SOAP notes, the list of Diagnosis and the list of Prescriptions are presented.

Vetpital

Animal name: PiuPiu
Owner VAT: 35263686
Consult date: 2018-08-01 12:45:02

Animal Characteristics

Animal Name	Owner Name	Species name	Colour	Gender	Age	Weight
PiuPiu	Maria Albertina	Mockingbird	Yellow	Male	0	6.00

SOAP Notes

S There is excess of weight
O Piupiu is obese and has a Potassium issue in the blood analysis
A Has low obesity and Insect Bites
P Give him desparasit - once a month

Diagnosis

Diagnosis Code	Diagnosis Name
00045	Heartworm Infestation
00056	Lyme Disease
00059	kidney failure
09002	Obesity

Prescriptions

Diagnosis Code	Medication Name	Medication Lab	Medication Dosage	Regime
00045	Desparasit	TiraBicho	3 Capsulas de 20 mL	1 capsula por mês
09002	laxil	intestiniti	10 capsulas de 100 mL	0.5 comprimido por dia
09002	levicim	emagricon	5 comprimidos de 20 mg	0.5 comprimido por dia

Figure 12: Detailed consult information regarding a visit of animal "PiuPiu", on the 1 of August of 2018, at 12:45.

The PHP code that generates this consult information page is shown bellow.

```

1 <html>
2     <body>
3         <h1><a href = "http://web.tecnico.ulisboa.pt/ist181579/sibd/
           proj/HomePage.php" style="color:MediumSeaGreen;text-
           decoration: none;">Vetpital</a></h1>
4 <?php
5     # Establishing the connection with the database
6     $host = "db.tecnico.ulisboa.pt";
7     $user = "ist181579";
8     $pass = "utfv5127";
9     $dsn = "mysql:host=$host;dbname=$user";
10    try

```



```

11     {
12         $connection = new PDO($dsn, $user, $pass);
13     }
14     catch(PDOException $exception)
15     {
16         echo("<p>Error: ␣");
17         echo($exception->getMessage());
18         echo("</p>");
19         exit();
20     }
21
22     # Data received
23     $animal_name = $_REQUEST['animal'];
24     $ownerVat = $_REQUEST['owner_vat'];
25     $date = $_REQUEST['date'];
26     # Check received data
27     echo("<p>Animal_name: ␣{$animal_name}␣</p>");
28     echo("<p>Owner_VAT: ␣{$ownerVat}␣</p>");
29     echo("<p>Consult_date: ␣{$date}␣</p>");
30
31     echo("<h3>Animal_Characteristics </h3>");
32     # Get the animal information
33     $sql = "CALL_AnimalInfo('{$animal_name}', '{$ownerVat}', '{$date}')";
34     $result = $connection->query($sql);
35     if ($result == FALSE)
36     {
37         $info = $connection->errorInfo();
38         echo("<p>Error: ␣{$info[2]} </p>");
39         exit();
40     }
41     # Display animal information
42     echo("<table␣border=␣\"2\">");
43     echo("<tr><td>Animal_Name</td><td>Owner_Name</td><td>Species␣
44         name</td><td>Colour</td><td>Gender</td><td>Age</td><td>
45         Weight</td></tr>");
46     foreach($result as $row)
47     {
48         echo("<tr␣align='center'>");
49         echo("<td>{$animal_name}</td>");
50         echo("<td>{$row['name_owner']}</td>");
51         echo("<td>{$row['species']}</td>");
52         echo("<td>{$row['colour']}</td>");
53         echo("<td>{$row['gender']}</td>");

```

```

52         echo("<td>{$row['age']}</td>");
53         echo("<td>{$row['weight']}</td>");
54         echo("</tr>");
55     }
56     echo("</table>");
57     $result = NULL;
58
59     # Get the soap notes
60     $sql = "CALL_SoapInfo(' $animal_name ', ' $ownerVat ', ' $date ');";
61     $result = $connection->query($sql);
62     if ($result == FALSE)
63     {
64         $info = $connection->errorInfo();
65         echo("<p>Error: ␣{$info[2]}</p>");
66         exit();
67     }
68     if ($result->rowCount() == 0)
69     {
70         echo("<h3>SOAP_Notes</h3>");
71         echo("<p>There are no SOAP notes</p>");
72     } else
73     {
74         # Display soap notes
75         echo("<h3>SOAP_Notes</h3>");
76         echo("<table border=\"0\" ␣cellspacing=\"5\">");
77         foreach($result as $row)
78         {
79             echo("<tr><td style='font-weight: bold'>S</td><td>{$row
80                 ['s']}</td></tr>");
81             echo("<tr><td style='font-weight: bold'>O</td><td>{$row
82                 ['o']}</td></tr>");
83             echo("<tr><td style='font-weight: bold'>A</td><td>{$row
84                 ['a']}</td></tr>");
85             echo("<tr><td style='font-weight: bold'>P</td><td>{$row
86                 ['p']}</td></tr>");
87         }
88         echo("</table>");
89     }
90     $result = NULL;
91
92     # Get the diagnosis information
93     $sql = "CALL_DiagnosisInfo(' $animal_name ', ' $ownerVat ', ' $date ');";
94     $result = $connection->query($sql);

```

```

91     if ( $result == FALSE)
92     {
93         $info = $connection->errorInfo();
94         echo("<p>Error : { $info[2]} </p>");
95         exit();
96     }
97     # Display diagnosis information
98     echo("<h3>Diagnosis </h3>");
99     echo("<table border='2'>");
100     echo("<tr><td>Diagnosis_Code</td><td>Diagnosis_Name</td></tr>"
        );
101     foreach( $result as $row)
102     {
103         echo("<tr align='center'>");
104         echo("<td>{ $row[ 'code' ]} </td>");
105         echo("<td>{ $row[ 'name' ]} </td>");
106         echo("</tr>");
107     }
108     echo("</table>");
109     $result = NULL;
110
111     # Get the prescriptions information
112     $sql = "CALL_PrescriptionsInfo( '$animal_name', '$ownerVat', '$date' );";
113     $result = $connection->query( $sql );
114     if ( $result == FALSE)
115     {
116         $info = $connection->errorInfo();
117         echo("<p>Error : { $info[2]} </p>");
118         exit();
119     }
120     # Display prescriptions information
121     echo("<h3>Prescriptions </h3>");
122     echo("<table border='2'>");
123     echo("<tr><td>Diagnosis_Code</td><td>Medication_Name</td><td>
        Medication_Lab</td><td>Medication_Dosage</td><td>Regime</
        td></td></tr>");
124     foreach( $result as $row)
125     {
126         echo("<tr align='center'>");
127         echo("<td>{ $row[ 'code' ]} </td>");
128         echo("<td>{ $row[ 'name_med' ]} </td>");
129         echo("<td>{ $row[ 'lab' ]} </td>");
130         echo("<td>{ $row[ 'dosage' ]} </td>");

```

```

131         echo("<td>{$row['regime']}</td>");
132         echo("</tr>");
133     }
134     echo("</table>");
135     $result = NULL;
136     $connection = null;
137 ?>
138 </body>
139 </html>

```

5. Adding new consults

When the user wants to create a new consult, he is confronted with a webpage as represented in figure 13. Again, in the beginning, there are some attributes which are printed just for confirmation. Then the user can choose between registering a consult with the current date or registering a previous consult (useful in cases where the veterinary forgets to register the consult) by using the buttons.

When the "Previous Consult" is pressed, it is asked for a date and hour like in figure 14. This pair should be filled with a previous date than the current one. After this selection the user is directed to the page represented in figure 15

When the "Current Consult" is pressed, the form with the required information to register a consult is presented like in figure 15.

After filling the new consult form and pressing the button "Register Consult", it is asked if the user wants to add a diagnosis to that consult as shown in figure 16. Here, the user is allowed to select one from the list existing in the database. After pressing "Register Diagnosis", the diagnosis is inserted in the consult and it is exhibited another button for the possibility of adding another diagnosis to that consult, see figure 17.

Vetpital

Animal name: Snoop Dogg

Owner VAT: 54609438

Client VAT: 54609438

Select a type of consult

Figure 13: Initial step in the page for adding consults, as seen from the phase after creating the animal seen on figure 10.

Vetpital

Animal name: Snoop Dogg

Owner VAT: 54609438

Client VAT: 54609438

Select a type of consult

Previous Consult

Consult date and hour:

Figure 14: Adding the consult's date, after choosing the "Previous Consult" option.

The PHP code that allows a consult registration is shown bellow.

1 `<html>`


Vetpital

Animal name: Snoop Dogg
 Owner VAT: 54609438
 Client VAT: 54609438

Select a type of consult

New Consult

2018-11-07 14:23

Veterinary: 

Weight:

SOAP Notes:

S
O
A
P

Figure 15: Adding the veterinary's identification, the animal's current weight and optional vet notes, after choosing the date.

Vetpital

Animal name: Snoop Dogg
 Owner VAT: 54609438
 Client VAT: 54609438

Select a type of consult

Consult was successfully created!

Associate Diagnosis


Diagnosis code: 

Figure 16: Options to add diagnosis to the consult.

```

2      <body>
3      <h1><a href = "http://web.tecnico.ulisboa.pt/ist181579/sibd/
      proj/HomePage.php" style="color:MediumSeaGreen;text -

```

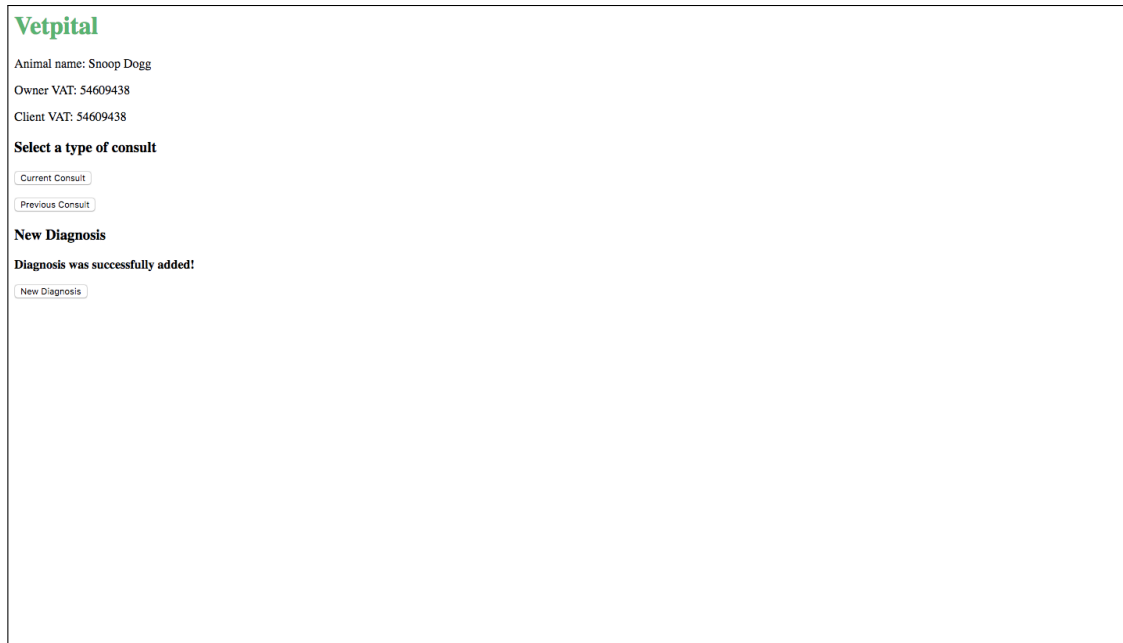


Figure 17: Page shown after successfully adding a diagnosis to the newly created consult.

```

                                decoration: _none;"]>Vetpital</a></h1>
4 <?php
5     # Establishing the connection with the database
6     $host = "db.tecnico.ulisboa.pt";
7     $user = "ist181579";
8     $pass = "utfv5127";
9     $dsn = "mysql:host=$host;dbname=$user";
10    try
11    {
12        $connection = new PDO($dsn, $user, $pass);
13    }
14    catch(PDOException $exception)
15    {
16        echo("<p>Error: _");
17        echo($exception->getMessage());
18        echo("</p>");
19        exit();
20    }
21
22    # Data received
23    $animal_name = $_REQUEST['animal'];
24    $ownerVat = $_REQUEST['owner_vat'];
25    $clientVat = $_REQUEST['client_vat'];

```

```

26         # Check received data
27         echo("<p>Animal_name: { $animal_name } </p>");
28         echo("<p>Owner_VAT: { $ownerVat } </p>");
29         echo("<p>Client_VAT: { $clientVat } </p>");
30
31     echo("<h3>Select_a_type_of_consult </h3>");
32     # Button to create a consult with the actual date
33     echo("<form_action=\"\" _method=\"post\">");
34     echo("<input_type='hidden' _name='animal' _value='{ $animal_name }' />\n");
35     echo("<input_type='hidden' _name='owner_vat' _value='{ $ownerVat }' />\n");
36     echo("<input_type='hidden' _name='client_vat' _value='{ $clientVat }' />\n"
37         );
38     echo("<p><input_type=\"submit\" _name=\"btnRegisterActual\" _value=\""
39         Current_Conconsult\" /></p>");
40     echo("</form>");
41     # Button to create a consult with a previous date
42     echo("<form_action=\"\" _method=\"post\">");
43     echo("<input_type='hidden' _name='animal' _value='{ $animal_name }' />\n");
44     echo("<input_type='hidden' _name='owner_vat' _value='{ $ownerVat }' />\n");
45     echo("<input_type='hidden' _name='client_vat' _value='{ $clientVat }' />\n"
46         );
47     echo("<p><input_type=\"submit\" _name=\"btnRegisterPrevious\" _value=\""
48         Previous_Conconsult\" /></p>");
49     echo("</form>");
50
51     $animal_name = $_REQUEST[ 'animal' ];
52     $ownerVat = $_REQUEST[ 'owner_vat' ];
53     $clientVat = $_REQUEST[ 'client_vat' ];
54     $date = $_REQUEST[ 'date_' ];
55     $time = $_REQUEST[ 'hour_' ];
56     $date_timestamp = $_REQUEST[ 'date_timestamp' ];
57     $s = $_REQUEST[ 's' ];
58     $o = $_REQUEST[ 'o' ];
59     $a = $_REQUEST[ 'a' ];
60     $p = $_REQUEST[ 'p' ];
61     $vetVat = $_REQUEST[ 'vet' ];
62     $weight = $_REQUEST[ 'weight' ];
63     $d_code = $_REQUEST[ 'd_code' ];
64
65     # It is desired a new consult with the actual date
66     if (isset($_REQUEST[ 'btnRegisterActual' ]))
67     {
68         $date = date("Y-m-d");

```



```

65     $time = date("H:i:s");
66 }
67
68 # It is desired a new consult with a previous date – select
        date
69 if (isset($_REQUEST['btnRegisterPrevious']))
70 {
71     $date = date("Y-m-d");
72     echo("<h3>_Previous_Conult</h3>");
73     echo("<form_action=\"\"_method=\"post\">");
74     echo("<p>Consult_date_and_hour:</p>");
75     echo("<input_type='date'_name='date_'_max='{ $date }'/>\n");
76     echo("<input_type='hidden'_name='animal'_value='{ $animal_name
        }'/>\n");
77     echo("<input_type='hidden'_name='owner_vat'_value='{ $ownerVat
        }'/>\n");
78     echo("<input_type='hidden'_name='client_vat'_value='{
        $clientVat }'/>\n");
79     echo("<p><input_type=\"submit\"_name=\"btnRegisterPreviousHour
        \"_value=\"Proceed\"/></p>");
80     echo("</form>");
81 }
82
83 # It is desired a new consult with a previous date – select
        time
84 if (isset($_REQUEST['btnRegisterPreviousHour']))
85 {
86     $CurrentDate = date("Y-m-d");
87     $time = date("H:i:s");
88     echo("<h3>_Previous_Conult</h3>");
89     echo("<p>Selected_date: _$date</p>");
90     echo("<form_action=\"\"_method=\"post\">");
91     echo("<p>Consult_hour:</p>");
92         if ($CurrentDate == $date){
93             echo("<input_type='time'_name='hour_'_
                max='{ $time }'/>\n");
94         } else {
95             echo("<input_type='time'_name='hour_'_
                />\n");
96         }
97     echo("<input_type='hidden'_name='date_'_value='{ $date }'/>");
98     echo("<input_type='hidden'_name='animal'_value='{ $animal_name
        }'/>\n");

```

```

99     echo("<input_type='hidden' _name='owner_vat' _value='{ $ownerVat
100         }'/>\n");
101     echo("<input_type='hidden' _name='client_vat' _value='{
102         $clientVat }'/>\n");
103     echo("<p><input_type='submit' _name='Proceed' _value='
104         Proceed' /></p>");
105     echo("</form>");
106 }
107
108 # Collect data required to create a new consult
109 if (isset($_REQUEST['Proceed']) || isset($_REQUEST['
110     btnRegisterActual']))
111 {
112     echo("<h3>New_Conult</h3>");
113     $date_timestamp = $date . " ";
114     $date_timestamp = $date_timestamp . $time;
115     echo("<p>$date_timestamp</p>");
116     echo("<form_action='\"' _method='post'>");
117     echo("<p_style='font-weight: bold'>Veterinary: <span_style='
118         margin-left:0.5em'>");
119     echo("<select_name='vet'>");
120     $sql = "SELECT _person.VAT_AS_VAT, _name FROM veterinary INNER_
121         JOIN _person ON veterinary.VAT=_person.VAT ORDER BY _person
122         .VAT";
123     $result = $connection->query($sql);
124     if ($result == FALSE)
125     {
126         $info = $connection->errorInfo();
127         echo("<p>Error: { $info[2]} </p>");
128         exit();
129     }
130     foreach($result as $row)
131     {
132         $vet_vat = $row['VAT'];
133         $option = $vet_vat . " _";
134         $option = $option . $row['name'];
135         echo("<option_value='\" $vet_vat\">
136             $option</option>");
137     }
138     echo("</select>");
139     echo("</p>");
140     echo("<p_style='font-weight: bold'>_Weight: <span_style='margin
141         -left:2em'>");

```

```

133     echo("<input_type='number' _step='0.01' _name='weight' />");
134     echo("<p_style='font-weight: bold'>_SOAP_Notes:</p>");
135     echo("<table_border='0' _cellspacing='5'>");
136     echo("<tr_align='center'>");
137     echo("<tr_align='center'><td_style='font-weight: bold'>_S</td><
        td>input_type='text' _name='s' /></td></tr>\n");
138     echo("<tr_align='center'><td_style='font-weight: bold'>_O</td><
        td>input_type='text' _name='o' /></td></tr>\n");
139     echo("<tr_align='center'><td_style='font-weight: bold'>_A</td><
        td>input_type='text' _name='a' /></td></tr>\n");
140     echo("<tr_align='center'><td_style='font-weight: bold'>_P</td><
        td>input_type='text' _name='p' /></td></tr>\n");
141     echo("</table>");
142     echo("<input_type='hidden' _name='animal' _value='{ $animal_name
        }' />\n");
143     echo("<input_type='hidden' _name='owner_vat' _value='{ $ownerVat
        }' />\n");
144     echo("<input_type='hidden' _name='client_vat' _value='{
        $clientVat }' />\n");
145     echo("<input_type='hidden' _name='date_timestamp' _value='{
        $date_timestamp }'");
146     echo("<p><input_type='submit' _name='RegisterCons' _value='\"
        Register_Conconsult\" /></p>");
147     echo("</form>");
148     $result = NULL;
149 }
150
151 # Register the new consult in the database
152 if (isset($_REQUEST['RegisterCons']))
153 {
154     $sql = "CALL_InsertConsult('$animal_name', '$ownerVat
        ', '$date_timestamp', '$s', '$o', '$a', '$p', '
        $clientVat', '$vetVat', '$weight')";
155     if ($connection->query($sql) == TRUE){
156         echo("<p_style='font-weight: bold'>_
            Consult_was_successfully_created
            !</p>");
157     }else{
158         $info = $connection->errorInfo();
159         echo("<p>Error: { $info[2]}</p>");
160         exit();
161     }
162 }

```

```

163
164 # Collect diagnosis codes for registration
165 if (isset($_REQUEST['RegisterCons']) | isset($_REQUEST['
    CollectDiag']))
166 {
167     echo("<h3>Associate_Diagnosis</h3>");
168     echo("<form_action=\"\"_method=\"post\">");
169
170     echo("<p_style='font-weight: bold'>Diagnosis_code:<span_style
        ='margin-left:0.5em'>");
171     echo("<select_name='d_code'>");
172     $sql = "SELECT_code, _name FROM diagnosis_code ORDER BY_code";
173     $result = $connection->query($sql);
174     if ($result == FALSE)
175     {
176         $info = $connection->errorInfo();
177         echo("<p>Error: { $info[2]}</p>");
178         exit();
179     }
180     foreach($result as $row)
181     {
182         $code = $row['code'];
183         $option = $code . " _";
184         $option = $option . $row['name'];
185         echo("<option_value=\" $code\">$option
            </option>");
186     }
187     echo("</select>");
188     echo("</p>");
189     echo("<input_type='hidden' _name='animal' _value='{ $animal_name
        }'/>\n");
190     echo("<input_type='hidden' _name='owner_vat' _value='{ $ownerVat
        }'/>\n");
191     echo("<input_type='hidden' _name='client_vat' _value='{
        $clientVat }'/>\n");
192     echo("<input_type='hidden' _name='date_timestamp' _value='{
        $date_timestamp }'");
193     echo("<p><input_type=\"submit\" _name=\" RegisterDiag\" _value=\"
        Register_Diagnosis\"/></p>");
194     echo("</form>");
195 }
196
197 # Register new diagnosis code

```

```

198         if (isset($_REQUEST['RegisterDiag']))
199         {
200             echo("<h3>New_Diagnosis</h3>");
201             $sql = "CALL_InsertDiagnosis('$_d_code','$_animal_name
202                 ','$_ownerVat','$_date_timestamp');";
203             if ($connection->query($sql) == TRUE){
204                 echo("<p_style='font-weight: bold'>_
205                     Diagnosis_was_successfully_added
206                     !</p>");
207             }else{
208                 $info = $connection->errorInfo();
209                 echo("<p>Error: {$_info[2]}</p>");
210                 exit();
211             }
212             echo("<form_action=\"\"_method=\"post\">");
213             echo("<input_type='hidden'_name='animal'_value='{$_animal_name
214                 }'/>\n");
215             echo("<input_type='hidden'_name='owner_vat'_value='{$_ownerVat
216                 }'/>\n");
217             echo("<input_type='hidden'_name='client_vat'_value='{
218                 $_clientVat }'/>\n");
219             echo("<input_type='hidden'_name='date_timestamp'_value='{
220                 $_date_timestamp }'");
221             echo("<p><input_type=\"submit\"_name=\"CollectDiag\"_value=\"
222                 New_Diagnosis\"/></p>");
223             echo("</form>");
224         }
225
226         $connection = null;
227     ?>
228     </body>
229 </html>

```

6. Adding a blood test

As seen on figure 11, each consult has an associated "New Blood Test" button. By pressing it, the page from figure 18 pops up. In here, the user can add all the info necessary for a blood test analysis, namely the white blood cell count, number of neutrophils and number of monocytes. As expected, it's also necessary to indicate the VAT of the assistant that performed the test procedure, in order to complete the data that needs to be filled in the database.

Vetpital

New blood test

Animal name: Bobi

Owner name: Maria Repolho

Owner VAT: 12345678

Date: 2018-11-15 17:46:00

Please introduce the following informations

Assistant VAT:

53250530

White blood cell count:

Number of neutrophils:

Number of monocytes:

Register

Figure 18: Blood test procedure creation form, shown in an example of adding to the consult of animal "Bobi", in 15 of November of 2018.

Functions, Triggers and Stored Procedures

As a way to prove the behavior of the codes corresponding to this chapter, some tests were created. In the following sub-chapters there are figures showing the results of the referred tests.

1. Animal Age Update Trigger

The figure 19a shows the age of all animals before a new consult. Afterwards, a new consult is added to the animal named "Matumbe", that belongs to a person with 54609438 as VAT number. The result of adding the consult is shown in the figure 19b. It should be noted that only the age corresponding to the animal who has the new consult was updated. This behavior was chosen because, if we have a big database, with a lot of animals, it would be very expensive to update all the animals every time a consult is created.

```

1 DELIMITER $$
2
3 CREATE TRIGGER ageUpdate BEFORE INSERT ON consult
4 FOR EACH ROW
5 BEGIN
6     UPDATE animal
7     SET age = YEAR(current_date)-birth_year
8     WHERE name = new.name
9     AND VAT = new.VAT_owner;
10 END$$
11

```

VAT	name	age
12345678	Bobí	5
35263686	Bobí	22
65856663	Bobí	1
69630596	FalaFala	7
54609438	Iris	NULL
63250157	Lantejoulas	18
12345678	Luna	3
78526209	Luna	3
69630596	Malandro	5
54609438	Matumbe	NULL
45620852	Miau	3
96520520	Onitório	5
96520520	Pala	2
54609438	Papinhas	NULL
53250530	Peggy	3
35263686	PiuPiu	0
35263686	Qualquer Coisa	22
98585856	Qualquer Coisa	NULL
45620852	Rapina	4
45620852	Riscas	2
54609438	Snoop Dogg	0

(a) Before the new consult.

VAT	name	age
12345678	Bobí	5
35263686	Bobí	22
65856663	Bobí	1
69630596	FalaFala	7
54609438	Iris	NULL
63250157	Lantejoulas	18
12345678	Luna	3
78526209	Luna	3
69630596	Malandro	5
54609438	Matumbe	9
45620852	Miau	3
96520520	Onitório	5
96520520	Pala	2
54609438	Papinhas	NULL
53250530	Peggy	3
35263686	PiuPiu	0
35263686	Qualquer Coisa	22
98585856	Qualquer Coisa	NULL
45620852	Rapina	4
45620852	Riscas	2
54609438	Snoop Dogg	0

(b) After the new consult.

Figure 19: Age of all animals in the database before and after a new consult.

12 DELIMITER ;

2. Veterinary Not Assistant Triggers

Figure 20 shows the VATs corresponding to the people who are a Veterinary or an Assistant. We should remember that one person cannot be simultaneously an assistant and a Veterinary. In the figure 21 is shown the result of trying to create a assistant that already is a veterinary, and vice versa.

VAT
53250530
63065186
63520543
85202652

(a) Assistant's VAT

VAT
12032014
85264856

(b) Veterinary's VAT

Figure 20: VAT numbers from persons who are assistant or veterinary.

```
MySQL [ist181579]> INSERT INTO assistant(VAT) VALUES (12032014);
ERROR 1644 (45001): This person is already registered as a Veterinary. A person cannot be a Assistant and a Veterinary at the same time.
MySQL [ist181579]> INSERT INTO veterinary(VAT) VALUES (53250530);
ERROR 1644 (45001): This person is already registered as a Assistant. A person cannot be a Assistant and a Veterinary at the same time.
```

Figure 21: Results while testing the triggers.

```
1 DELIMITER $$
2
3 CREATE TRIGGER vetNotAssist BEFORE INSERT ON assistant
4 FOR EACH ROW
5 BEGIN
```

```

6      DECLARE vet_cannot_be_a_assistant CONDITION FOR SQLSTATE '45001';
7      DECLARE EXIT HANDLER FOR vet_cannot_be_a_assistant
8      BEGIN
9      SIGNAL vet_cannot_be_a_assistant SET MESSAGE_TEXT = 'This person is
      already registered as a Veterinary. A person cannot be a Assistant
      and a Veterinary at the same time.';
10     END;
11 IF EXISTS(
12     SELECT VAT
13     FROM veterinary
14     WHERE veterinary.VAT = new.VAT
15 )THEN
16     SIGNAL vet_cannot_be_a_assistant;
17
18 END IF;
19 END$$
20
21 CREATE TRIGGER assistNotVet BEFORE INSERT ON veterinary
22 FOR EACH ROW
23 BEGIN
24     DECLARE assistant_cannot_be_a_vet CONDITION FOR SQLSTATE '45001';
25     DECLARE EXIT HANDLER FOR assistant_cannot_be_a_vet
26     BEGIN
27     SIGNAL assistant_cannot_be_a_vet SET MESSAGE_TEXT = 'This person is
      already registered as a Assistant. A person cannot be a Assistant
      and a Veterinary at the same time.';
28     END;
29 IF EXISTS(
30     SELECT VAT
31     FROM assistant
32     WHERE assistant.VAT = new.VAT
33 )THEN
34     SIGNAL assistant_cannot_be_a_vet;
35
36 END IF;
37 END$$
38
39 DELIMITER ;

```

3. Unique Phone Numbers Triggers

Figure 22 shows the existing phone numbers in the database. Removing a number and trying to insert one that already exists, the trigger prevents the duplicate number creation, as seen on figure 23. This figure also

pictures the result obtained when trying to update a phone number with an existing phone number.

VAT	phone
12032014	269358742
12345678	912345670
14740000	999999999
35263686	965423580
45620852	930536378
53250530	+63220185206
54609438	936723009
63065186	918206630
63250157	+351 50666232
63520543	963025225
65856663	920520648
69630596	251068266
78526209	270352066
85202652	210524896
85264856	263596307
91970633	963258741
96520520	936210875
98585856	918520856
91919111	632501578

Figure 22: The phone number from the persons within the database.

```
MySQL [ist181579]> DELETE FROM phone_number WHERE VAT = 12032014;
INSEQuery OK, 1 row affected (0.01 sec)

MySQL [ist181579]> INSERT INTO phone_number VALUES(12032014,'930536378');
ERROR 1644 (45000): This phone number already exists.
MySQL [ist181579]> INSERT INTO phone_number VALUES(12032014,'123456999');
Query OK, 1 row affected (0.01 sec)

MySQL [ist181579]> UPDATE phone_number SET phone = '930536378' WHERE VAT = 12032014;
ERROR 1644 (45000): This phone number already exists.
```

Figure 23: Results while testing the triggers.

```
1 DELIMITER $$
2
3 CREATE TRIGGER phoneNotDuplicateInsert BEFORE INSERT ON phone_number
4 FOR EACH ROW
5 BEGIN
6     DECLARE phone_already_exists CONDITION FOR SQLSTATE '45000';
7     DECLARE EXIT HANDLER FOR phone_already_exists
8     BEGIN
9         SIGNAL phone_already_exists SET MESSAGE_TEXT = 'This_phone_number_
            already_exists.';
10    END;
11 IF EXISTS(
12     SELECT phone
13     FROM phone_number
14     WHERE phone = new.phone
15 )THEN
16     SIGNAL phone_already_exists;
17 END IF;
18 END$$
19
```

```

20 CREATE TRIGGER phoneNotDuplicateUpdate BEFORE UPDATE ON phone_number
21 FOR EACH ROW
22 BEGIN
23     DECLARE phone_already_exists CONDITION FOR SQLSTATE '45000';
24     DECLARE EXIT HANDLER FOR phone_already_exists
25     BEGIN
26         SIGNAL phone_already_exists SET MESSAGE_TEXT = 'This_phone_number_
                already_exists.';
27     END;
28 IF EXISTS(
29     SELECT phone
30     FROM phone_number
31     WHERE phone = new.phone
32 )THEN
33     SIGNAL phone_already_exists;
34 END IF;
35 END$$
36
37 DELIMITER ;

```

4. Yearly Number of Consults Function

In order to visualize the function working, an example is portrayed using the consults corresponding to the animal named "Miau". In figure 24a, one can see the dates from the consults where "Miau" was present. The next figure (24b) represents the calculated number of "Miau"'s consults in 2017.

date_timestamp
2016-02-06 14:50:30
2017-06-15 16:25:16

(a) Miau's consults dates.

number_consult_year('45620852','Miau','2017')
1

(b) Number of Miau's consults in 2017.

Figure 24: Age of all animals in the database before and after a new consult.

```

1 DELIMITER $$
2
3 CREATE FUNCTION number_consult_year(a_vat INTEGER,a_name CHAR(50),c_year YEAR)
4 RETURNS INTEGER
5 BEGIN
6
7     DECLARE count_result INTEGER;
8
9     SELECT COUNT(date_timestamp) INTO count_result
10    FROM consult

```

```

11 WHERE (date_timestamp) IN(
12 SELECT date_timestamp
13 FROM consult
14 WHERE name = a_name
15 AND VAT_owner = a_vat
16 AND YEAR(date_timestamp) = c_year
17 );
18
19 RETURN count_result;
20
21 END$$
22
23 DELIMITER ;

```

5. Milligrams to Centigrams Stored Procedure

The images in figure 25 show the indicators and the produced indicators before the procedure. After calling the procedure, the result is the one shown in the figure 26.

name	reference_value	units	description
Blood-Chloride	355.00	milligrams	Medido em mg/dL
Blood-cholesterol	140.00	milligrams	Medido em mg/dL
Blood-Ionized Calcium	4.40	milligrams	Medido em mg/dL
Blood-Magnesium	2.00	milligrams	Medido em mg/dL
Blood-Potassium	17.50	milligrams	Medido em mg/dL
Blood-Sodium	320.00	milligrams	Medido em mg/dL
Blood-Total Calcium	9.00	milligrams	Medido em mg/dL
Blood-Total Serum Iron	100.00	micrograms	Medido em ug/dL
creatinine level	0.50	milligrams	Medido em mg/24h
Urine-pH	6.00	pH	Sem unidade
Urine-Potassium	60.00	millimole	Medido em mmol/24h
Urine-Protein	10.00	milligrams	Medido em mg/dL
Urine-Urea	16.00	grams	Medido em g/24h
Urine-Uric Acid	400.00	milligrams	Medido em mg/24h

(a) indicator's table before the procedure.

name	VAT_owner	date_timestamp	num	indicator_name	value
Bobí	35263686	2018-07-19 19:06:00	42	Number of monocytes	999.99
Bobí	35263686	2018-07-19 19:06:00	42	Number of neutrophils	10.00
Bobí	35263686	2018-07-19 19:06:00	42	White blood cell count	5.00
Luna	12345678	2018-09-18 15:52:56	6136	Blood-Ionized Calcium	4.40
Luna	12345678	2018-09-18 15:52:56	6136	Blood-Sodium	322.00
Luna	12345678	2018-09-18 15:52:56	6136	Blood-Total Serum Iron	150.00
Luna	12345678	2018-12-04 10:48:54	42	Number of monocytes	333.00
Luna	12345678	2018-12-04 10:48:54	42	Number of neutrophils	222.00
Luna	12345678	2018-12-04 10:48:54	42	White blood cell count	111.00
PiuPiu	35263686	2018-08-01 12:45:02	8256	creatinine level	1.10

(b) produced.indicator's table before the procedure.

Figure 25: Tabels before the procedure.

name	reference_value	units	description
Blood-Chloride	35.50	centigrams	Medido em mg/dL
Blood-cholesterol	14.00	centigrams	Medido em mg/dL
Blood-Ionized Calcium	0.44	centigrams	Medido em mg/dL
Blood-Magnesium	0.20	centigrams	Medido em mg/dL
Blood-Potassium	1.75	centigrams	Medido em mg/dL
Blood-Sodium	32.00	centigrams	Medido em mg/dL
Blood-Total Calcium	0.90	centigrams	Medido em mg/dL
Blood-Total Serum Iron	100.00	micrograms	Medido em ug/dL
creatinine level	0.05	centigrams	Medido em mg/24h
Urine-pH	6.00	pH	Sem unidade
Urine-Potassium	60.00	millimole	Medido em mmol/24h
Urine-Protein	1.00	centigrams	Medido em mg/dL
Urine-Urea	16.00	grams	Medido em g/24h
Urine-Uric Acid	40.00	centigrams	Medido em mg/24h

(a) indicator's table after the procedure.

name	VAT_owner	date_timestamp	num	indicator_name	value
Bobí	35263686	2018-07-19 19:06:00	42	Number of monocytes	999.99
Bobí	35263686	2018-07-19 19:06:00	42	Number of neutrophils	10.00
Bobí	35263686	2018-07-19 19:06:00	42	White blood cell count	5.00
Luna	12345678	2018-09-18 15:52:56	6136	Blood-Ionized Calcium	0.44
Luna	12345678	2018-09-18 15:52:56	6136	Blood-Sodium	32.20
Luna	12345678	2018-09-18 15:52:56	6136	Blood-Total Serum Iron	150.00
Luna	12345678	2018-12-04 10:48:54	42	Number of monocytes	333.00
Luna	12345678	2018-12-04 10:48:54	42	Number of neutrophils	222.00
Luna	12345678	2018-12-04 10:48:54	42	White blood cell count	111.00
PiuPiu	35263686	2018-08-01 12:45:02	8256	creatinine level	0.11

(b) produced.indicator's table after the procedure.

Figure 26: Tabels after the procedure.

```

1 DELIMITER $$
2

```

```

3 CREATE PROCEDURE change2centigrams()
4 BEGIN
5     DECLARE ind_name CHAR(30);
6     DECLARE done INT DEFAULT FALSE;
7
8     DECLARE c CURSOR FOR
9     SELECT name
10    FROM indicator
11   WHERE units = 'milligrams';
12    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;
13
14    OPEN c;
15    REPEAT
16    FETCH c INTO ind_name;
17    IF NOT done THEN
18    UPDATE produced_indicator
19    SET value = value/10
20    WHERE indicator_name = ind_name;
21    END IF;
22    UNTIL done
23    END REPEAT;
24    CLOSE c;
25
26    UPDATE indicator
27    SET units = 'centigrams', reference_value = reference_value/10
28    WHERE units = 'milligrams';
29 END$$
30
31 DELIMITER ;

```

Appendix

AnimalConsultsInfo

```

1 DROP PROCEDURE AnimalConsultsInfo;
2 DELIMITER $$
3
4 CREATE PROCEDURE AnimalConsultsInfo(IN animal_name VARCHAR(50),IN owner_vat
    INTEGER)
5 /*RETURNS TABLE(date timestamp, vatVet INTEGER, nameVet VARCHAR(100))*/
6 BEGIN
7 /*RETURN TABLE(*

```

```

8      SELECT consult.date_timestamp as date,consult.VAT_vet as vatVet ,person
      .name as nameVet
9      FROM consult
10     INNER JOIN veterinary ON consult.VAT_vet = veterinary.VAT
11     INNER JOIN person ON veterinary.VAT = person.VAT
12     WHERE consult.name = animal_name
13     AND consult.VAT_owner = owner_vat
14     ORDER BY consult.date_timestamp DESC;
15  /*)*/
16  END$$
17  DELIMITER ;

```

AnimalInfo

```

1  DROP PROCEDURE AnimalInfo;
2  DELIMITER $$
3
4  CREATE PROCEDURE AnimalInfo(IN animal_name VARCHAR(50),IN owner_vat INTEGER,
      IN date_ TIMESTAMP)
5  /*RETURNS TABLE(name_owner VARCHAR(100), species VARCHAR(50), colour VARCHAR
      (20), gender VARCHAR(20), age INTEGER, weight NUMERIC(5,2))*/
6  BEGIN
7  /*RETURN TABLE(*/
8      SELECT person.name AS name_owner, species_name AS species , colour ,
      gender , age , consult.weight AS weight
9      FROM animal
10     INNER JOIN person ON animal.VAT = person.VAT
11     INNER JOIN consult ON animal.VAT = consult.VAT_owner AND animal.name =
      consult.name
12     WHERE animal.name = animal_name
13     AND animal.VAT = owner_vat
14     AND consult.date_timestamp = date_ ;
15  /*)*/
16  END$$
17  DELIMITER ;

```

consult_with_client

```

1  DROP PROCEDURE consult_with_client;
2  DELIMITER $$
3
4  CREATE PROCEDURE consult_with_client(IN animal_name VARCHAR(50),IN owner_vat
      INTEGER,IN clientVat INTEGER)
5  BEGIN

```

```

6      SELECT DISTINCT VAT_client
7      FROM consult
8      WHERE name = animal_name
9      AND VAT_owner = owner_vat
10     AND VAT_client = clientVat;
11 END$$
12 DELIMITER ;

```

DiagnosisInfo

```

1 DROP PROCEDURE DiagnosisInfo;
2 DELIMITER $$
3
4 CREATE PROCEDURE DiagnosisInfo(IN animal_name VARCHAR(50),IN owner_vat INTEGER
    , IN date_ TIMESTAMP)
5 /*RETURNS TABLE(code CHAR(5) , name CHAR(100))*/
6 BEGIN
7 /*RETURN TABLE(*/
8     SELECT diagnosis_code.code AS code , diagnosis_code.name AS name
9     FROM consult_diagnosis
10    INNER JOIN diagnosis_code ON consult_diagnosis.code = diagnosis_code.
        code
11    WHERE consult_diagnosis.name = animal_name
12    AND consult_diagnosis.VAT_owner = owner_vat
13    AND consult_diagnosis.date_timestamp = date_;
14 /*)*/
15 END$$
16 DELIMITER ;

```

get_vat_from_name

```

1 DROP PROCEDURE get_vat_from_name;
2 DELIMITER $$
3
4 CREATE PROCEDURE get_vat_from_name(IN owner_name CHAR(100))
5 BEGIN
6     SELECT person.VAT AS VAT, person.name AS NAME
7     FROM person INNER JOIN client USING(VAT)
8     WHERE person.name LIKE CONCAT( '%',owner_name, '%') ;
9 END$$
10 DELIMITER ;

```

GetMaxProcedureNum

```

1 DROP PROCEDURE GetMaxProcedureNum;

```

```

2 DELIMITER $$
3
4 CREATE PROCEDURE GetMaxProcedureNum()
5 BEGIN
6     SELECT max(num)
7     FROM _procedure;
8 END$$
9 DELIMITER ;

```

InsertBloodTest

```

1 DROP PROCEDURE InsertBloodTest;
2 DELIMITER $$
3
4 CREATE PROCEDURE InsertBloodTest(IN animal_name VARCHAR(50), IN VAT_owner
    INTEGER, IN date_timestamp TIMESTAMP, IN num INTEGER, IN VAT_assistant
    INTEGER, IN white_blood_cell_count NUMERIC(5, 2), IN number_neutrophils
    NUMERIC(5, 2), IN number_monocytes NUMERIC(5, 2))
5 BEGIN
6     INSERT INTO _procedure VALUES(animal_name, VAT_owner, date_timestamp,
    num, 'Blood_test');
7     INSERT INTO test_procedure VALUES(animal_name, VAT_owner,
    date_timestamp, num, 'blood');
8     INSERT INTO produced_indicator VALUES(animal_name, VAT_owner,
    date_timestamp, num, "White_blood_cell_count",
    white_blood_cell_count);
9     INSERT INTO produced_indicator VALUES(animal_name, VAT_owner,
    date_timestamp, num, "Number_of_neutrophils", number_neutrophils);
10    INSERT INTO produced_indicator VALUES(animal_name, VAT_owner,
    date_timestamp, num, "Number_of_monocytes", number_monocytes);
11    INSERT INTO performed VALUES(animal_name, VAT_owner, date_timestamp,
    num, VAT_assistant);
12 END$$
13 DELIMITER ;

```

InsertConsult

```

1 DROP PROCEDURE InsertConsult;
2 DELIMITER $$
3
4 CREATE PROCEDURE InsertConsult(IN name CHAR(50), IN VAT_owner INTEGER, IN
    date_timestamp TIMESTAMP, IN s TEXT, IN o TEXT, IN a TEXT, IN p TEXT, IN
    VAT_client INTEGER, IN VAT_vet INTEGER, IN weight NUMERIC(5,2))
5 BEGIN

```

```

6          INSERT INTO consult VALUES(name, VAT_owner, date_timestamp, s, o, a, p
          , VAT_client, VAT_vet, weight);
7 END$$
8 DELIMITER ;

```

InsertDiagnosis

```

1 DROP PROCEDURE InsertDiagnosis;
2 DELIMITER $$
3
4 CREATE PROCEDURE InsertDiagnosis(IN code CHAR(5), IN name CHAR(50), IN
    VAT_owner INTEGER, IN date_timestamp TIMESTAMP)
5 BEGIN
6     INSERT INTO consult_diagnosis VALUES(code, name, VAT_owner,
        date_timestamp);
7 END$$
8 DELIMITER ;

```

more_general

```

1 DROP PROCEDURE more_general;
2 DELIMITER $$
3
4 CREATE PROCEDURE more_general()
5 BEGIN
6     SELECT DISTINCT name2 AS species_name
7     FROM generalization_species
8     WHERE name2 != ALL (
9         SELECT name1
10        FROM generalization_species
11    );
12
13 END$$
14 DELIMITER ;

```

more_specific

```

1 DROP PROCEDURE more_specific;
2 DELIMITER $$
3
4 CREATE PROCEDURE more_specific(IN general_name CHAR(50))
5 BEGIN
6     SELECT DISTINCT name1 AS species_name
7     FROM generalization_species
8     WHERE name2 = general_name;

```



```

9
10 END$$
11 DELIMITER ;

```

PrescriptionsInfo

```

1 DROP PROCEDURE PrescriptionsInfo;
2 DELIMITER $$
3
4 CREATE PROCEDURE PrescriptionsInfo(IN animal_name VARCHAR(50),IN owner_vat
    INTEGER, IN date_ TIMESTAMP)
5 /*RETURNS TABLE(code CHAR(5), name_med CHAR(20), lab CHAR(20), dosage CHAR
    (100), regime CHAR(100))*/
6 BEGIN
7 /*RETURN TABLE(*/
8     SELECT code, name_med, lab, dosage, regime
9     FROM prescription
10    WHERE prescription.name = animal_name
11    AND prescription.VAT_owner = owner_vat
12    AND prescription.date_timestamp = date_;
13 /*)*/
14 END$$
15 DELIMITER ;

```

Project_part2

```

1 /*1*/
2 DELIMITER $$
3
4 CREATE TRIGGER ageUpdate BEFORE INSERT ON consult
5 FOR EACH ROW
6 BEGIN
7     UPDATE animal
8     SET age = YEAR(current_date)-birth_year
9     WHERE name = new.name
10    AND VAT = new.VAT_owner;
11 END$$
12
13 DELIMITER ;
14
15 /*TEST 1*/
16 SELECT VAT, name, age
17 FROM animal;
18

```

```

19 INSERT INTO consult(name,VAT_owner,VAT_client,VAT_vet,weight) VALUES('Bobi'
    ,12345678,12345678,12032014,10);
20
21 /*2*/
22 DROP TRIGGER vetNotAssist;
23 DROP TRIGGER assistNotVet;
24
25 DELIMITER $$
26
27 CREATE TRIGGER vetNotAssist BEFORE INSERT ON assistant
28 FOR EACH ROW
29 BEGIN
30     DECLARE vet_cannot_be_a_assistant CONDITION FOR SQLSTATE '45001';
31     DECLARE EXIT HANDLER FOR vet_cannot_be_a_assistant
32     BEGIN
33         SIGNAL vet_cannot_be_a_assistant SET MESSAGE_TEXT = 'This_person_is_
            already_registered_as_a_Veterinary._A_person_cannot_be_a_Assistant_
            and_a_Veterinary_at_the_same_time.';
34     END;
35 IF EXISTS(
36     SELECT VAT
37     FROM veterinary
38     WHERE veterinary.VAT = new.VAT
39 )THEN
40     SIGNAL vet_cannot_be_a_assistant;
41
42 END IF;
43 END$$
44
45 CREATE TRIGGER assistNotVet BEFORE INSERT ON veterinary
46 FOR EACH ROW
47 BEGIN
48     DECLARE assistant_cannot_be_a_vet CONDITION FOR SQLSTATE '45001';
49     DECLARE EXIT HANDLER FOR assistant_cannot_be_a_vet
50     BEGIN
51         SIGNAL assistant_cannot_be_a_vet SET MESSAGE_TEXT = 'This_person_is_
            already_registered_as_a_Assistant._A_person_cannot_be_a_Assistant_
            and_a_Veterinary_at_the_same_time.';
52     END;
53 IF EXISTS(
54     SELECT VAT
55     FROM assistant
56     WHERE assistant.VAT = new.VAT

```

```

57 )THEN
58     SIGNAL assistant_cannot_be_a_vet;
59
60 END IF;
61 END$$
62
63 DELIMITER ;
64
65 /*TEST 2*/
66 SELECT VAT
67 FROM veterinary;
68
69 SELECT VAT
70 FROM assistant;
71
72 INSERT INTO assistant(VAT) VALUES (12032014);
73
74 /*3*/
75 DROP TRIGGER phoneNotDuplicateInsert;
76 DROP TRIGGER phoneNotDuplicateUpdate;
77
78 DELIMITER $$
79
80 CREATE TRIGGER phoneNotDuplicateInsert BEFORE INSERT ON phone_number
81 FOR EACH ROW
82 BEGIN
83     DECLARE phone_already_exists CONDITION FOR SQLSTATE '45000';
84     DECLARE EXIT HANDLER FOR phone_already_exists
85     BEGIN
86         SIGNAL phone_already_exists SET MESSAGE_TEXT = 'This_phone_number_
            already_exists.';
87     END;
88 IF EXISTS(
89     SELECT phone
90     FROM phone_number
91     WHERE phone = new.phone
92 )THEN
93     SIGNAL phone_already_exists;
94 END IF;
95 END$$
96
97 CREATE TRIGGER phoneNotDuplicateUpdate BEFORE UPDATE ON phone_number
98 FOR EACH ROW

```

```

99 BEGIN
100     DECLARE phone_already_exists CONDITION FOR SQLSTATE '45000';
101     DECLARE EXIT HANDLER FOR phone_already_exists
102     BEGIN
103         SIGNAL phone_already_exists SET MESSAGE_TEXT = 'This_phone_number_
            already_exists.';
104     END;
105 IF EXISTS(
106     SELECT phone
107     FROM phone_number
108     WHERE phone = new.phone
109 )THEN
110     SIGNAL phone_already_exists;
111 END IF;
112 END$$
113
114 DELIMITER ;
115
116 /*TEST 3*/
117 SELECT *
118 FROM phone_number;
119
120 DELETE FROM phone_number WHERE VAT = 12032014;
121 INSERT INTO phone_number VALUES(12032014,'930536378');
122 UPDATE phone_number SET phone = '930536378' WHERE VAT = 12032014;
123
124
125 /*4*/
126 DELIMITER $$
127
128 CREATE FUNCTION number_consult_year(a_vat INTEGER,a_name CHAR(50),c_year YEAR)
129 RETURNS INTEGER
130 BEGIN
131
132     DECLARE count_result INTEGER;
133
134     SELECT COUNT(date_timestamp) INTO count_result
135     FROM consult
136     WHERE (date_timestamp) IN(
137     SELECT date_timestamp
138     FROM consult
139     WHERE name = a_name
140     AND VAT_owner = a_vat

```

```

141         AND YEAR(date_timestamp) = c_year
142     );
143
144     RETURN count_result;
145
146 END$$
147
148 DELIMITER ;
149
150 /*5*/
151 DELIMITER $$
152
153 CREATE PROCEDURE change2centigrams()
154 BEGIN
155     DECLARE ind_name CHAR(30);
156     DECLARE done INT DEFAULT FALSE;
157
158     DECLARE c CURSOR FOR
159     SELECT name
160     FROM indicator
161     WHERE units = 'milligrams';
162     DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;
163
164     OPEN c;
165     REPEAT
166     FETCH c INTO ind_name;
167     IF NOT done THEN
168     UPDATE produced_indicator
169     SET value = value/10
170     WHERE indicator_name = ind_name;
171     END IF;
172     UNTIL done
173     END REPEAT;
174     CLOSE c;
175
176     UPDATE indicator
177     SET units = 'centigrams', reference_value = reference_value/10
178     WHERE units = 'milligrams';
179 END$$
180
181 DELIMITER ;
182
183 /* TEST 5 */

```

```

184 SELECT * FROM indicator;
185 SELECT * FROM produced_indicator;
186
187 CALL change2centigrams();

searchConsultName

1 DROP PROCEDURE SearchConsultName;
2 DELIMITER $$
3
4 CREATE PROCEDURE SearchConsultName(IN animal_name VARCHAR(50),IN owner_name
    VARCHAR(100),IN clientVat INTEGER)
5 /*RETURNS TABLE(animal VARCHAR(50), vat INTEGER, owner VARCHAR(100),
    thisClient BOOLEAN)*/
6 BEGIN
7 /*RETURN TABLE(*/
8     SELECT consult.name as animal,consult.VAT_owner as vat ,person.name as
        owner, VAT_client = clientVat as thisClient
9     FROM consult
10    INNER JOIN client ON consult.VAT_owner = client.VAT
11    INNER JOIN person ON client.VAT = person.VAT
12    WHERE consult.name = animal_name
13    AND person.name LIKE CONCAT( '%',owner_name, '%' );
14 /*)*/
15 END$$
16 DELIMITER ;
17     /*DECLARE neg_balance numeric(20,2);
18     SELECT sum(balance) into pos_balance
19     from account natural join depositor
20     where customer_name = c_name;
21     select sum(amount) into neg_balance
22     from loan natural join borrower
23     where customer_name = c_name;
24
25     if pos_balance is NULL then
26     set pos_balance = 0;
27     end if;
28     if neg_balance is NULL then
29     set neg_balance = 0;
30     end if;
31     return pos_balance - neg_balance;*/
32 /*end$$*/

searchQuery

```

```

1 DROP PROCEDURE SearchConsultName;
2 DELIMITER $$
3
4 CREATE PROCEDURE SearchConsultName(IN animal_name VARCHAR(50),IN owner_name
    VARCHAR(100),IN clientVat INTEGER)
5 BEGIN
6     SELECT DISTINCT animal.name as animal, animal.VAT as vat , person.name as
        owner
7     FROM animal
8     INNER JOIN client ON animal.VAT = client.VAT
9     INNER JOIN person ON client.VAT = person.VAT
10    WHERE animal.name = animal_name
11    AND person.name LIKE CONCAT( '%', owner_name, '%') ;
12 END$$
13 DELIMITER ;

```

SoapInfo

```

1 DROP PROCEDURE SoapInfo;
2 DELIMITER $$
3
4 CREATE PROCEDURE SoapInfo(IN animal_name VARCHAR(50),IN owner_vat INTEGER, IN
    date_ TIMESTAMP)
5 /*RETURNS TABLE(s TEXT, o TEXT, a TEXT, p TEXT)*/
6 BEGIN
7 /*RETURN TABLE(*/
8     SELECT s, o, a, p
9     FROM consult
10    WHERE consult.name = animal_name
11    AND consult.VAT_owner = owner_vat
12    AND consult.date_timestamp = date_;
13 /*)*/
14 END$$
15 DELIMITER ;

```