Instituto Superior Técnico

Mestrado em Engenharia Eletrotécnica e de Computadores

Sistemas de Informação e Bases de Dados

2018/2019 1º Semestre

Assignment 3 - Using the Database

Grupo: 4

84037 – Eduardo Costa

84038 – Eduardo Melo

84087 – João Sebastião

Docente: Bruno Martins

7 de Dezembro de 2018

1. A Web Application Using the Database

The web application can be found in http://web.ist.utl.pt/ist425337/main.php. Opening it shows us seven different options. We shall check all of them later. For now, in order to check the functionalities specified for the project, selecting "Find Animal", leads us to the page in figure 1.

A client comes to the hospital asking for a consult

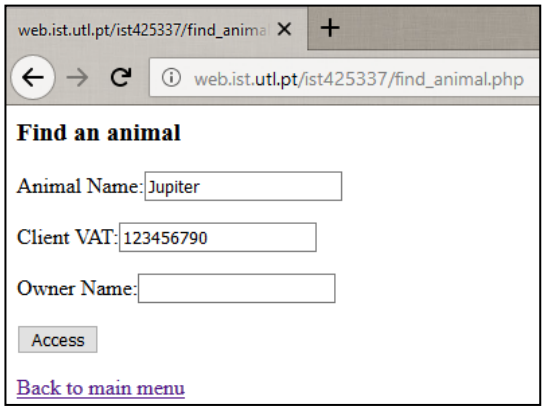

Figure 1 – find_animal.php
(no matches found)

This form requests the animal name, the client VAT and the owner's name (or part of it), in order to list all the animals with corresponding names and owner names.

If client VAT is valid and no existing animals are found, the page shown in figure 2 is opened. If matches are found, page shown in figure 4 is opened.

Introducing an "owner name", however, it's not mandatory, and if it's left in blank every animal that has a matching name will appear in figure 4.

The case displayed in figure 1 will lead to figure 2.

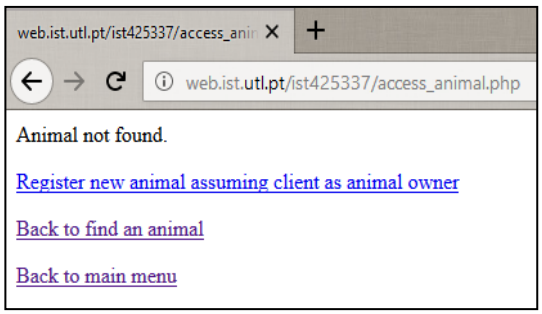

Figure 2 – access_animal.php
(no matches found)

In figure 2, there were no matches for the data inserted in figure 1. We are prompted to register this new animal, assuming the client as the animal owner. Clicking it will open the page in figure 3, where we can see that the animal name and the owner VAT are fixed, and cannot be changed. Filling every field and pressing "register" will register the new animal in our database.

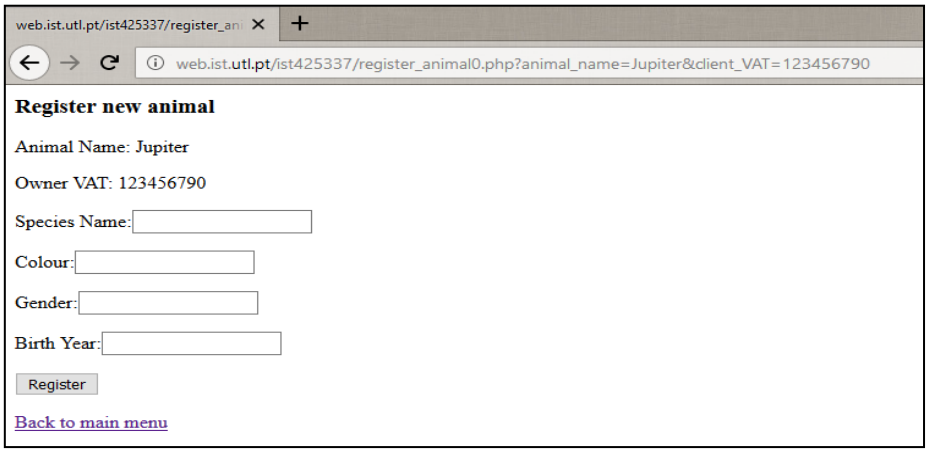

Figure 3 – register_animal0.php
(no matches found)

However, if we insert data that matches, for example “Animal Name” = “Paradox”, “Client VAT” = “123456783” and “Owner Name” = “dinger”, the page in figure 4 will be shown.

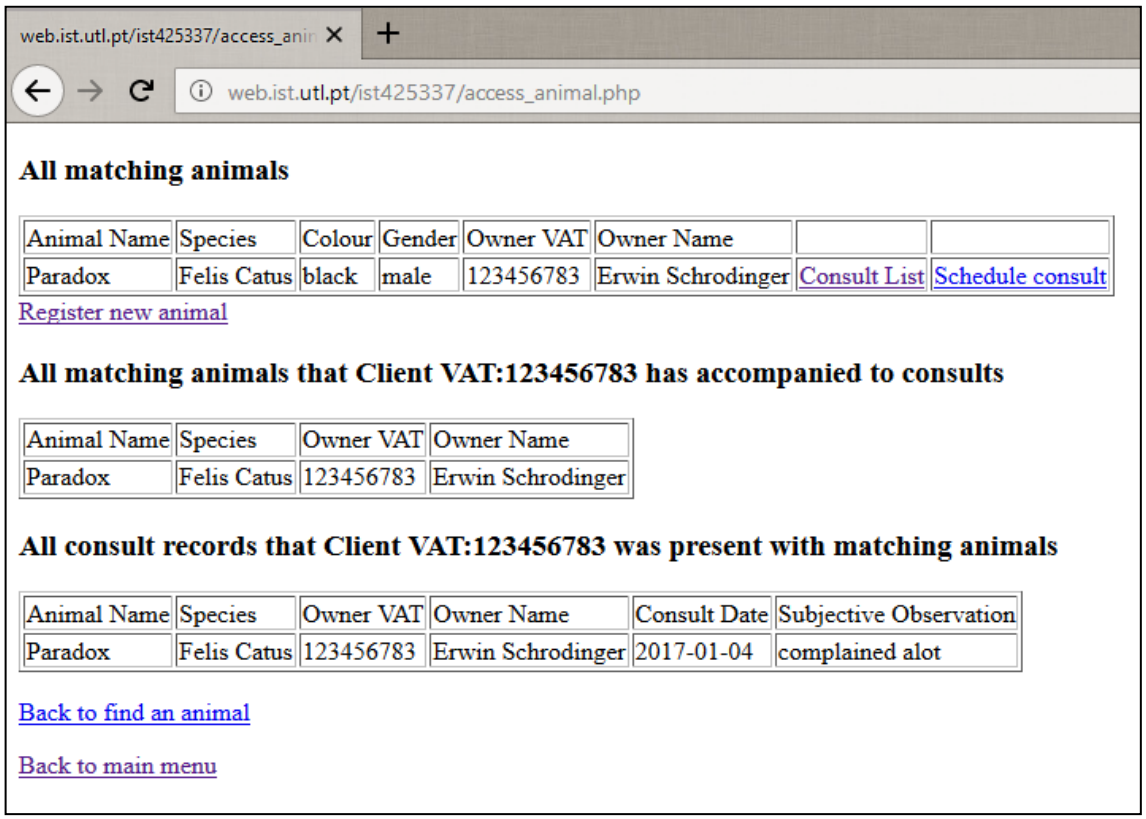

Figure 4 – access_animal.php

All matching animals:

Lists all animals in the database that the parameters inserted before correspond to “Animal Name” and (a part of) “Owner Name”.

All matching animals that Client VAT:xxxxxxxxx has accompanied to consults

Lists only the animals listed in “**All matching animals**” that were accompanied by the client at least once to a consult before.

All consult records that Client VAT:xxxxxxxxx was present with matching animals

Lists all the consults records that match with “**All matching animals that Client VAT:xxxxxxxxx has accompanied to consults**”. This means that there could be listed more than one consult record for the same animal, if that’s the case.

It’s possible to register a new animal in “Register new animal”, or to schedule a consult for any animal that matched in “Schedule consult”.

If “Consult List” is selected, it will lead to the second task of the project.

Access and registry of information associated to a consult

Selecting “Consult List” on the page of figure 4, will lead to the page in figure 5, that lists all the previous consult of the selected animal.

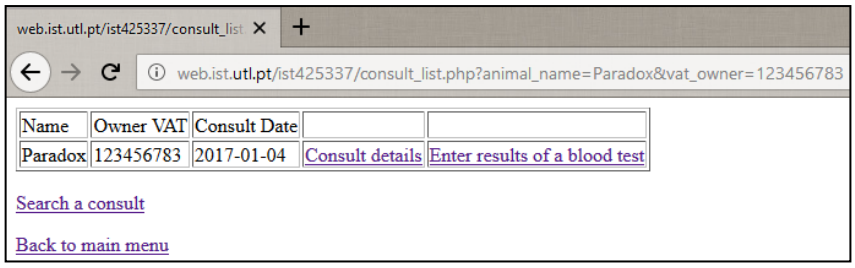

Figure 5 – consult_list.php

If “Enter results of a blood test“ is selected, it will lead to the third task of the project.

If “Consult details” of a consult is selected, a new page will open, containing all the information associated to the consult selected divided in 4 tables, as shown in figure 6.

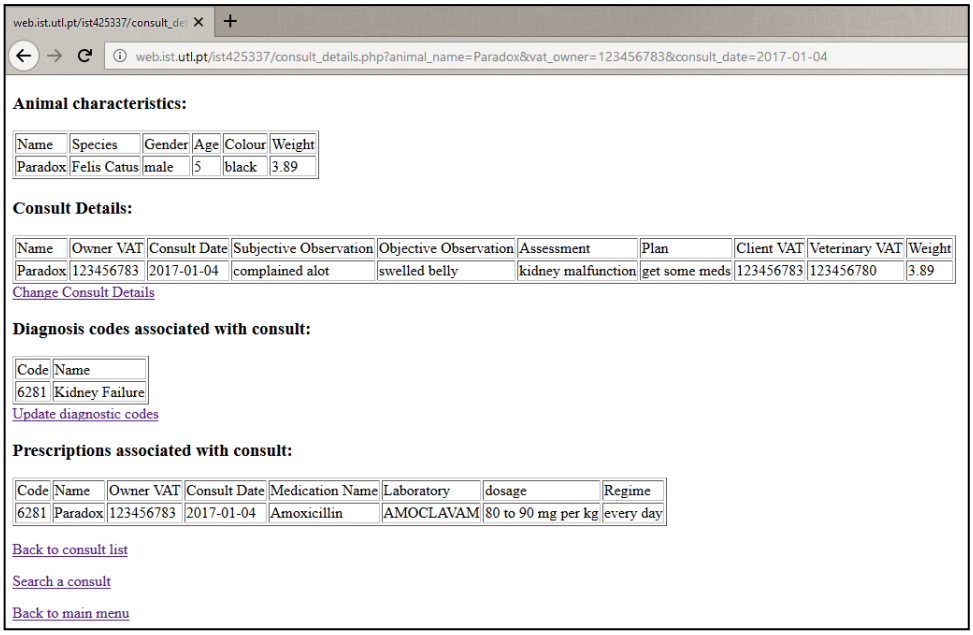

Figure 6 – consult_details.php

It is possible to change the consult details by selecting “Change Consult Details” or to update the diagnostic codes by selecting “Update diagnostic codes”.

Change Consult Details: A new page opens with a form. Here it’s possible to add/edit the SOAP notes, the veterinary VAT and the weight of the animal. (Vet. VAT must exist).

Update diagnostic codes: A new page opens with 2 tables: one with all the codes associated to the consult, along with an option to remove them - “Remove Code”; the other table with all the codes that are not associated to the consult, along with an option to add them - “Add Code”. Selecting each of this options will remove or add the correspondent diagnostic code to the consult, respectively.

Registry of information associated to a blood test procedure

Selecting “Enter results of a blood test” on the page of figure 5, will lead to the page in figure 7, which has a form that requests to insert information about the blood test. This information consists of the VAT of the assistant that conducted the procedure and a fixed set of indicators.

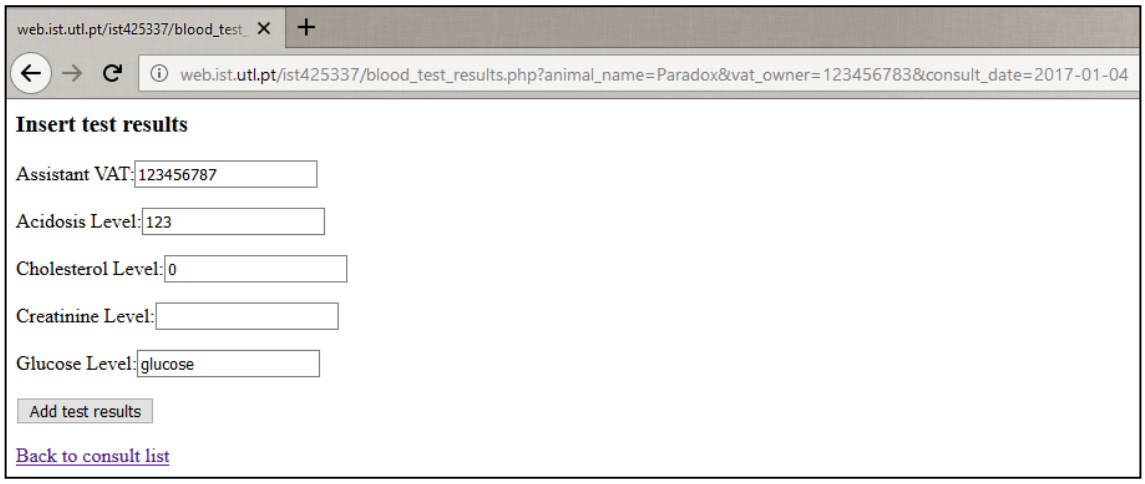

Figure 7 – blood_test_results.php

A valid assistant VAT must be inserted, along with at least one valid indicator value. Inserting for example the values indicated in figure 7, will open the page in figure 8.

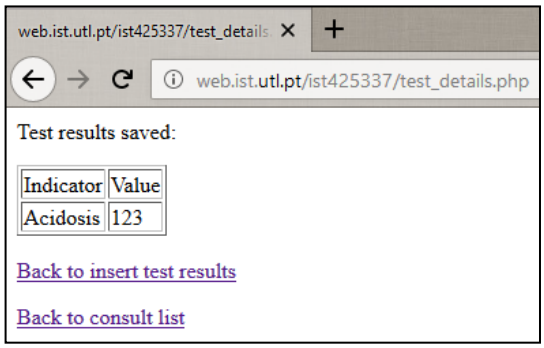

Figure 8 – test_details.php

As we can see, only the “Acidosis Level” indicator was saved in the database. The remaining 3 indicators were discarded.

It was used a transaction to perform all the inserts on the tables. With this, inserting a valid assistant VAT with invalid indicators (all 4 of them) will result into a rollback, erasing all the data modifications made from the start, i.e., the inserts in medical_procedure, performed and test_procedure.

The PHP code of this database uses prepared statements when required, in order to prevent SQL injection.

The remaining functionalities of the database that are not stated in the project specifications are included in appendix.

2. Functions, Triggers and Stored Procedures

Trigger to update age of an animal when it goes to a consult:

```
DROP TRIGGER IF EXISTS update_age;
delimiter $$
create trigger update_age after insert on consult
for each row
begin
	update animal as a, consult as c
	set a.age = year(current_timestamp) - a.birth_year
	where a.name = c.name
	and a.VAT = c.VAT_owner;
end$$
delimiter ;
```

Triggers to ensure that there are no individuals simultaneously assistants and veterinaries in the database:

```
DROP TRIGGER IF EXISTS vet_check;
DROP TRIGGER IF EXISTS assist_check;
delimiter $$
-- checks if the vat already belongs to an assistant
create trigger vet_check before insert on veterinary
for each row
begin
	if exists(select a.VAT from assistant as a where a.VAT = new.VAT) then
		set new.VAT = NULL;
	end if;
end$$
delimiter ;
```

```
delimiter $$
-- checks if the vat already belongs to a veterinary
create trigger assist_check before insert on assistant
for each row
begin
	if exists(select v.VAT from veterinary as v where v.VAT = new.VAT) then
		set new.VAT = NULL;
	end if;
end$$
delimiter ;
```

Trigger to ensure that different individuals cannot have the same phone number:

```
DROP TRIGGER IF EXISTS phone_check;
delimiter $$
create trigger phone_check before insert on phone_number
for each row
begin
	if exists(select * from phone_number as p where p.phone = new.phone) then
		set new.VAT = NULL, new.phone = NULL;
	end if;
end$$
delimiter ;
```

For each of this last 3 triggers (vet_check, assist_check, phone_check), if the conditions that they check are verified, a value of NULL is inserted in a primary key column of their respective table. With this, SQL prints an error and the query is stopped, avoiding the insertion of the unwanted data in the database.

Function to compute the total number of consults for a given animal within a year:

```
DROP FUNCTION IF EXISTS count_consults;
delimiter $$
create function count_consults(c_name varchar(255), c_year int)
returns integer
begin
	declare c_count integer;
	select count(name) into c_count
	from consult
	where c_name = name
	and c_year = year(date_timestamp);
	return c_count;
end$$
delimiter ;
```

Stored procedure to change milligrams to centigrams:

```
DROP PROCEDURE IF EXISTS mg2cg;
delimiter $$
create procedure mg2cg()
begin
	update produced_indicator as p
	set p.indicator_value = p.indicator_value*0.1
	where exists(
		select p.indicator_name
		from indicator as i
		where i.name = p.indicator_name
		and i.units = 'milligrams');
	update indicator
	set units = 'centigrams', reference_value = reference_value*0.1
	where units = 'milligrams';
end$$
delimiter ;
```

A. Appendix

The remaining functionalities of the database are stated here, below figure 7.

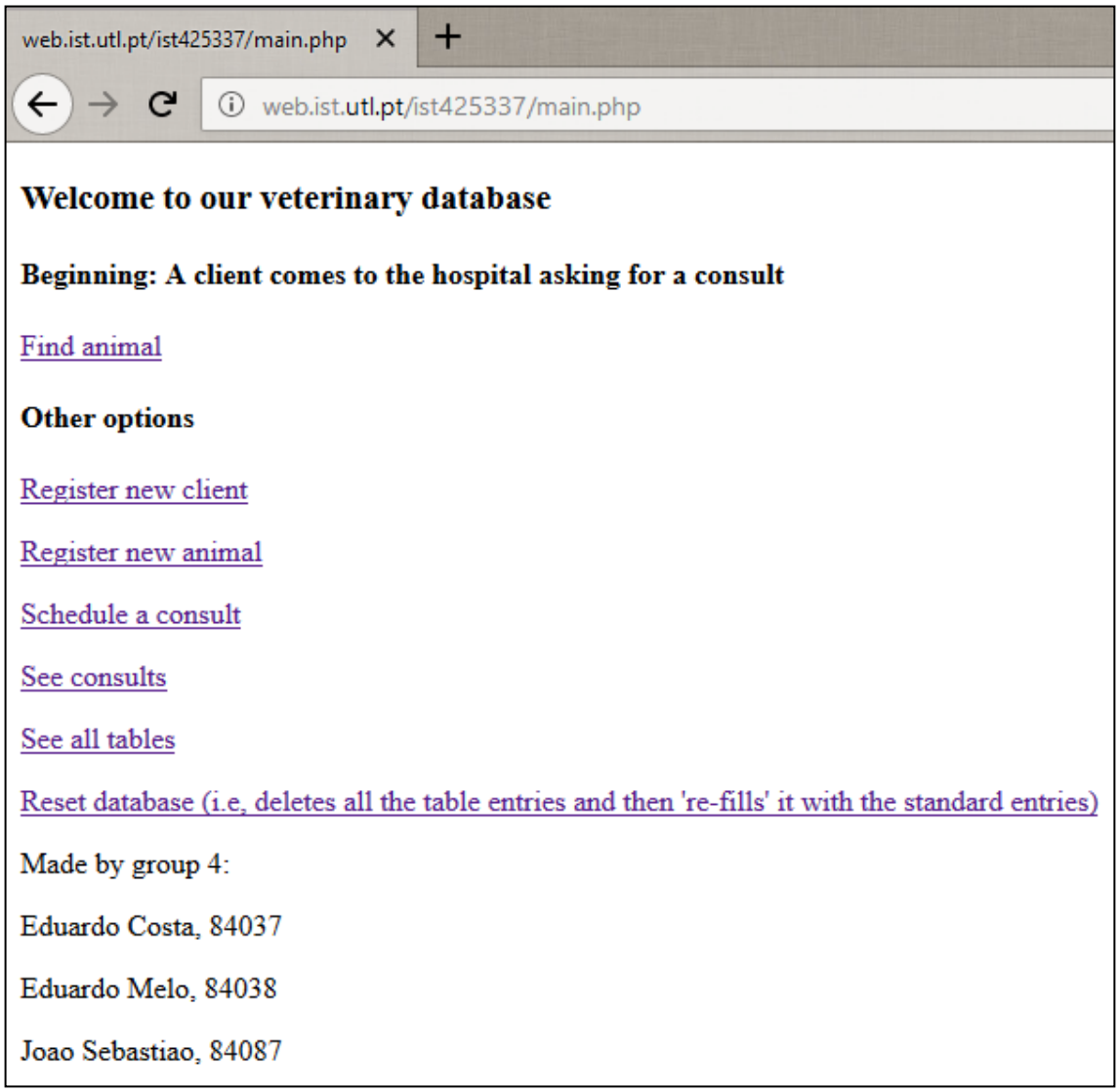

Figure 7 – main.php

Find Animal – 1st task of the project;

Register new client – Registers a new client in the database;

Register new animal – Registers a new animal in the database;

Schedule a consult – Schedules a new consult for a given animal;

See consults – Lists all consults in the database, and allows to select the consult list of any animal with previous consults;

See all tables – Allows to select for view one table of the database;

Reset database – Deletes every table entry, refilling them after with all our standard entries.