BREEDOG.

Bianchi Tommaso
Cenzi Filippo
Facchin Carlo
Fiorese Tobia
Manzato Andrea
Secchiero Elia
Xie Aniello

Objective

The objective of the project is the creation of a website for dog breeder and dog information:

1. Designed for registered breeder

2. Info about dogs and their breeders

3. List of the events to which a dog took part to

4. Search for dogs of other breeders

5. Let non-registered user to search for an adoptable dog

Design

The design choices are made to create a interface user-friendly, readable and intuitive:

1. Color palette restricted and consistent

2. Material design icons

3. Use of buttons instead raw links

4. Mobile friendly
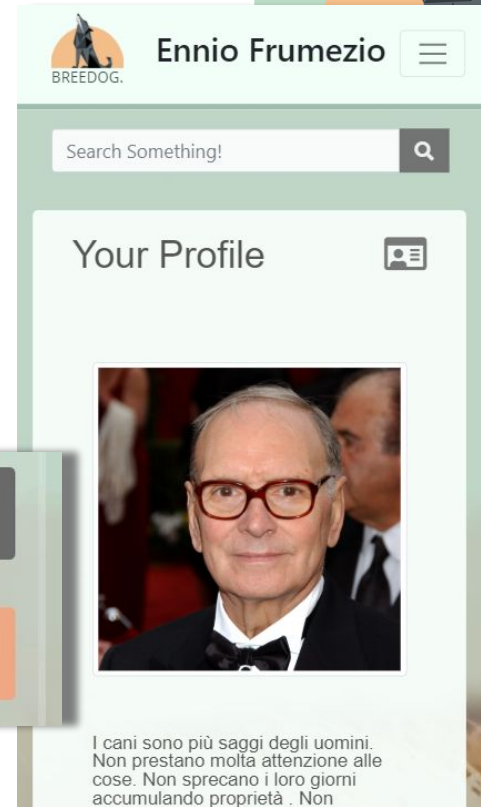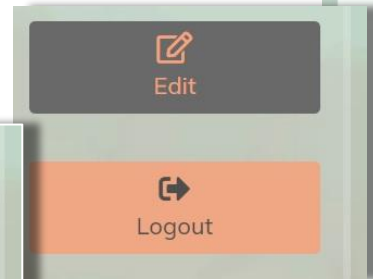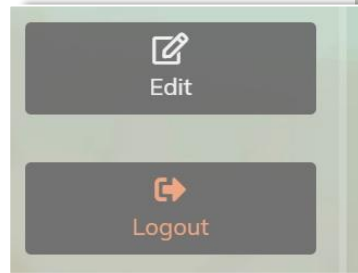
Model-View-Controller

The application adopts the Model-View-Controller paradigm

- Model: access method to the data and notification of the state updates.

- View: let the user see the output of an action or the state querying the model and acts as a bridge between user and application.

- Controller: gets the user data from the View, process it and in case updates the state, communicating it to the model.
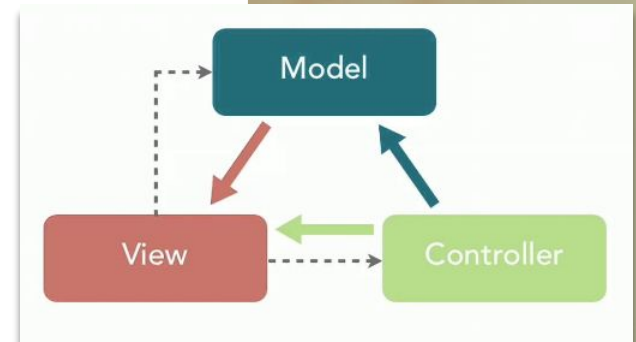
User Authentication

We implemented the user authentication using the simplest Basic Authorization header where username and password are concatenated with a colon and encoded Base64.
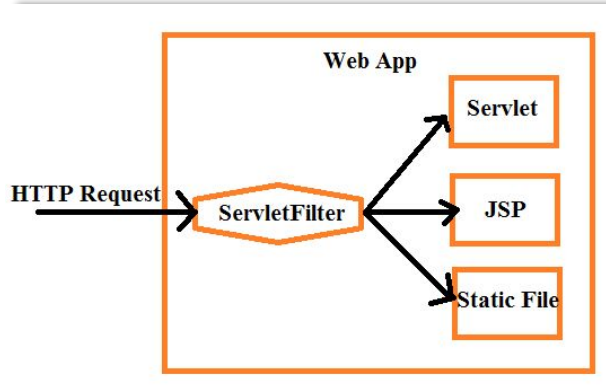
User pages are inside /user/* path.
Requests to this section are filtered using Java Servlet Filter that check if we already logged in.
Not logged request will be redirected to the homepage.

HomePage

The homepage shows all the user information and the breeds that the user own.

Breeder profile seen by another user.
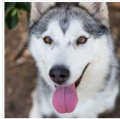
Breeds info.

Edit page

Provides the possibility to change user informations.
The possibility to add the vat number is only for the users that haven't registered it yet.

Edit page

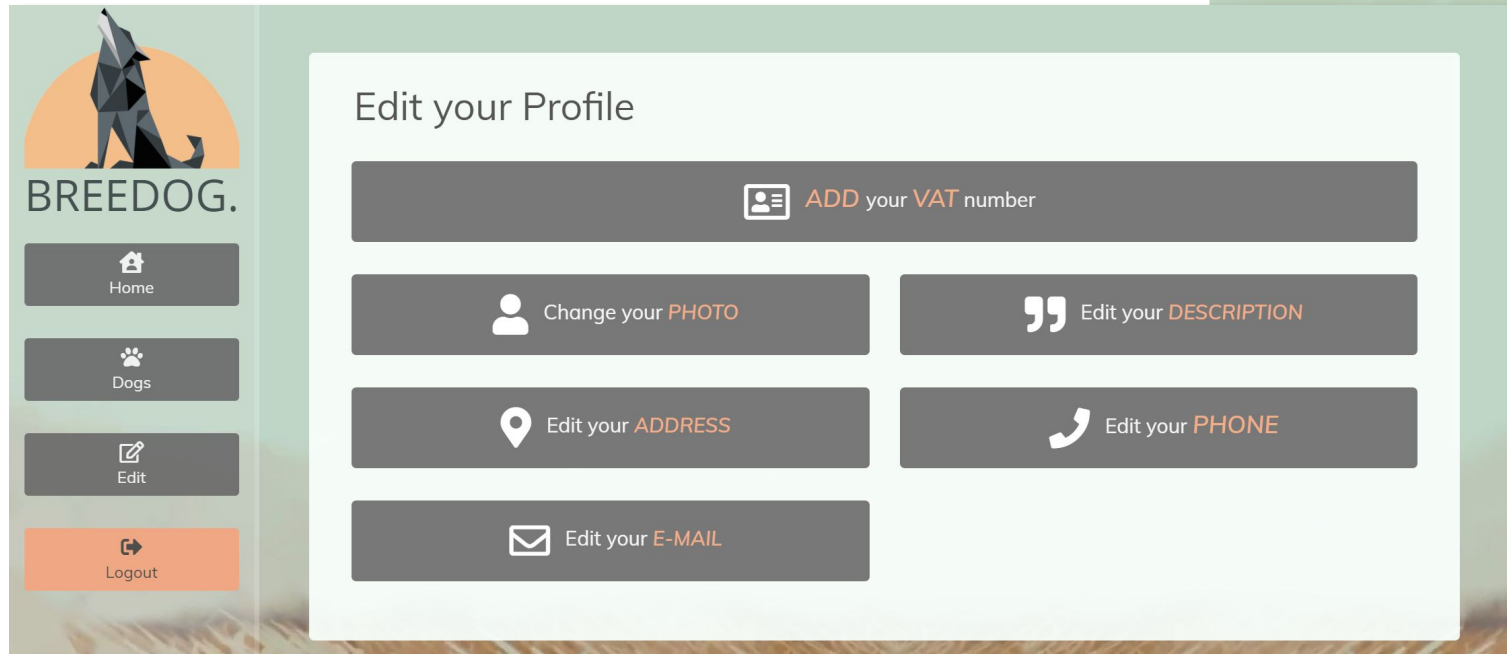The load of the breeder information is generated querying the database with the breeder fiscal code while for each insertion is used a POST request of json data.

Edit page

In the edit page there is also the possibility to change the user password. The password confirmation is verified with js while the correctness of the current password uses the authentication check.

Dog Profile Page

Peter ✏️

Basic Infos

Kennel Name	Famiglia Frumezio
Microchip Code	380260043256632
Tatoo Code	LJ2E3KCNL9
Birth Date	1999-06-12
Gender	Male
Owner	Tiziano
Status	Pensione

Breed

FCI Code	111
Group	8
Breed Name	Golden retriever

Traits

Height	61.0 cm
Weight	38.0 Kg
Coat Color	Bianco
Character	Attento e vigile
Teeth Structure	Normali
Particular Sign	

The dog home page can be subdivided into :
- Dog basic informations
- Pathology
- Awards
- Genealogy

Pathology ✏️

Code	Name	Severity
4053	Displasia dei gomiti	D

Awards ✏️

Show Victory Only ⬜

Competition	Type	Event
PLTHSGN	Show	0000000002

Parents

Name	Microchip	Role

Sons ✏️

Name	Microchip	Birth
Tobia	380260000000000	2005-03-29

Almost one query per category

Show Victory Only 🟢

Slider to show only victory

Dog Profile Page

Modals to show competitions and events extra information.

Pathology ✏️

Code	Name	Severity
4053	Displasia dei gomiti	D

Awards ✏️

Show Victory Only ⬤

Competition	Type	Event
PLTHSGN	Show	0000000002

Parents

Name	Microchip	Role

Sons ✏️

Name	Microchip	Birth
Tobia	380260000000000	2005-03-29

Competition Info ✕

ID	PLTHSGN
Type	Show
Group	8
Class	Campioni
Won	Yes

Event Info ✕

ID	0000000002
Location	Verona
Begin	2018-11-14
Finish	2018-11-16
Zip	37100

Dog Profile Page

Dynamic Modal to manage Dog information update

Update Info

Scegli il file Nessun file scelto

Name	Peter
Birth	12/06/1999
Owner	Tiziano
Status	Retired
Height	61.0
Weight	38.0
Color	Bianco
Character	Attento e vigile
Teeth	Normali
Sign	

Save Changes

Update Competitions

Action	Add Competition
Competion ID	
Competition Type	Select Type
Competition Group	Select Group
Competition Class	Select Class
Competition Won	Choose Yes/No
Event ID	
Event Location	
Event Begin	gg/mm/aaaa
Event Finish	gg/mm/aaaa
Event Zip	

Add Competition

Edit Pathology

Action	Add Pathology
Pathology	Enter a pathology
Severity	Select pathology severity

Add Pathology

Update Sons

Action	Add Son
Microchip	

Add Son

Insertion page

It permits the insertion of a new dog from the breeder logged into the application. It uses a JavaScript form control and sends information to the server through JSON data with a POST request.

Insertion page

The HTTP request is triggered by the press of the submit button. The request is handled by a Java Servlet that creates the object to parse to the database. After a successful insertion the user is redirect to the home page.
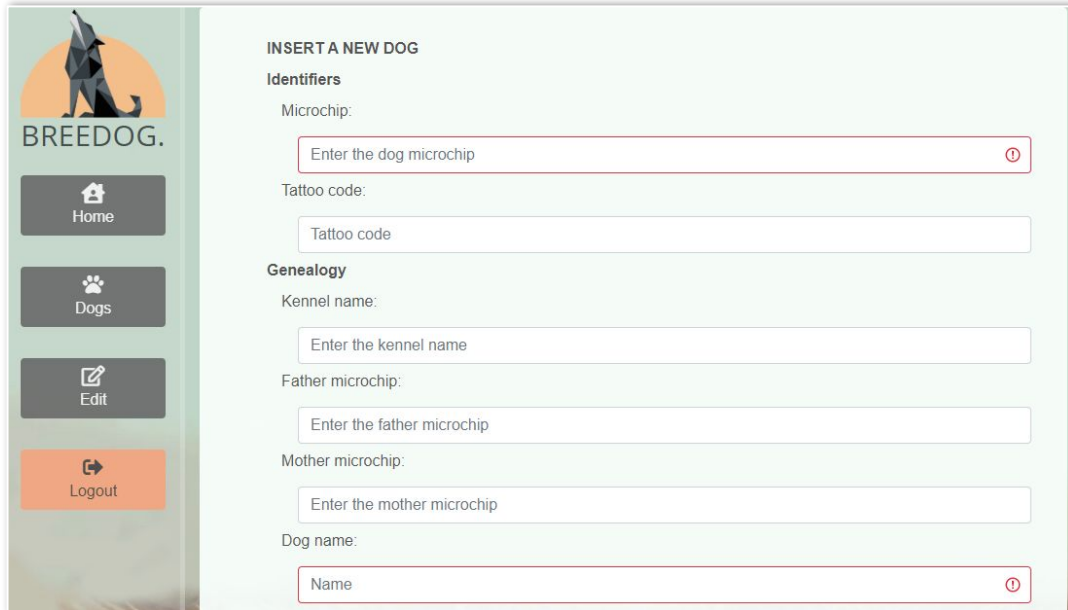
Insertion page

A simple check is implemented in the javascript code for don't let a user insert empty parameters of dog that are mandatory. Forms of bootstrap are retrieved through javascript and they are set to invalid if their are empty.

Search page

Provides access to the dogs information in the database through a search page based on a query and filters.
Not registered users can only search for adoptable dogs and only see basic information.

Search results

The search page is provided at search/ as a JSP page and the requests to the database are made through AJAX GET requests to searchDogs/.
The response from the backend is a json file containing a list of resulting resources.

The results are appended to the DOM tree of the page in article elements and only 6 dogs are shown per page, to keep the json response file small.

With the page pagination the user makes AJAX GET requests to get the next results of his query.

Previous   Next

OP

/Kennel-WA-1.0/user/dog.html?dogmc=380293294293023

Search query

The query to the database has a variable number of instruction to satisfy different queries and different type of users.

```
1   select *
2   from dog
3   where
4
5   ? IS NULL OR to_tsvector(microchip || ' ' || name|| ' ' || kennel || ' ' ||
6   coat || ' ' || character || ' ' || owner || ' ' || breederfc) @@ to_tsquery(?))
7
8   AND (? IS NULL OR SEX = ?::gender)
9   AND (? IS NULL OR STATUS = ?::dogstatus )
10  AND (? IS NULL OR STATUS = 'Adottabile')
11  AND (? IS NULL OR DATE_PART('year', CURRENT_DATE) -
12    - DATE_PART('year', birth::date) < ?::INTEGER) ";
13   [...]
14  + " ORDER BY birth ?
15  LIMIT ? OFFSET ?;
```

DEMO