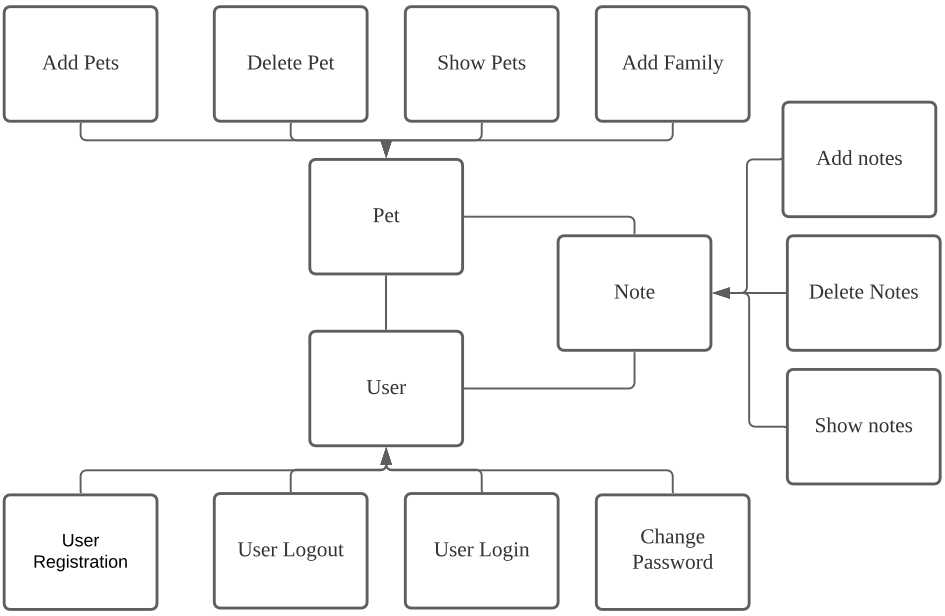
Introduction of the website

This is a website for people who have pets with the name of Poop Picker. With the growing popularity of pets, the number of pet owners is increasing. Just like raising a child, there are many interesting things to record when you are with a pet. The first time of shaking hands, the first time of climbing a mountain, etc. A person may meet all kinds of people in his or her life, but a pet has only one owner in his or her life, and their longest life span may only be twenty years, so I hope poopers will use this website to record their pets and their warm stories.

Function module diagram

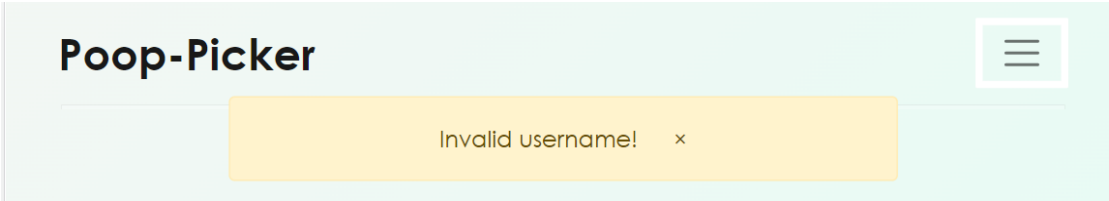


List of features

1. bootstrap

The bootstrap is used to achieve the responsive requirements of multiple web components, such as navigation bar, timeline, and to

remind the user of the operation using the bootstrap's own alert class, such as form validation, deletion.



2. jQuery

The jQuery application implements several dynamic effects to increase the aesthetics of the interface. For example, jumping pet menus, adding multi-step forms in the pet interface, etc.

3.advanced feature of html5 - svg

When the cursor is placed on the component, the color of the elements will slowly change, which enhances the interaction with the user, and also complements the effect of adding a filter to the image hovering below.

4. AJAX

In order to ensure smooth update of the web page and update of the database, in the form of adding family members for pets submitted using ajax asynchronous update rather than the traditional refresh page, the user submits the form by sending json string to the back-end,

the back-end completes data validation and database updates, which the feedback message is passed to the user in the form of alert, without the requirements of refreshing

Link of the website and password

bamboo.pythonanywhere.com

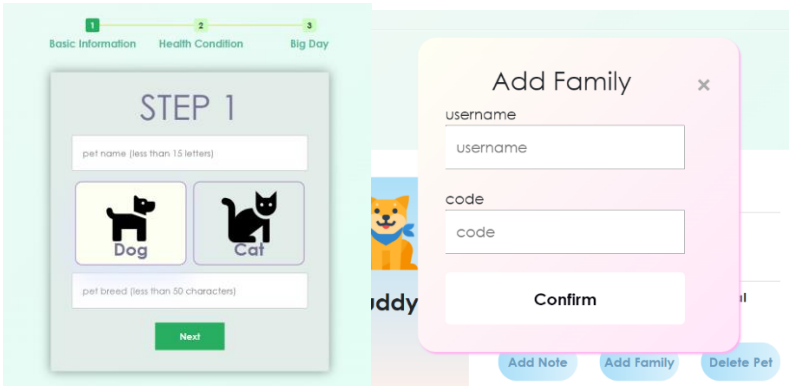
username: bamboo

password: bamboo081103

Evaluation

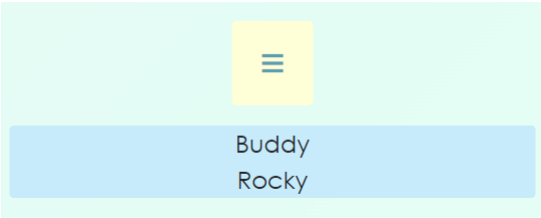
1. style

The style of the whole website is closely related to the theme, such as the video of the puppy playing on the landing page, the illustrations of the registration and login pages, and the pictures of the cats and dogs in the cards in the display of the pets. Since the website is aimed at pet owners, almost all of whom are loving people, the overall colour scheme is macaroon with various gradients. There are various forms on the page, such as pop-up forms, multi-step forms, and forms for the whole page.



2. user experience

The number of refreshing and fluency of the page is satisfactory.In the interface for displaying pets, all pets owned by a user are loaded at once, so users do not need to refresh the interface when switching pets by clicking on the menu, and pop-up forms and submission forms do not need to be refreshed when adding family members to pets in this page.



3. Use of database relationships

There are three models in the database respectively, users, pets and notes, a user can have more than one pet, a pet can have more than one note, a pet can also have a lot of family members, although the relationship between the user and the pet is many-to-many, but in view of the main purpose of the site is for pet owners, so it does not show the user through the pet, saving the site's resources and highlighting

the focus

4. orderly content by query

The content of the front-end page is achieved through database queries, the most complex of which is the interface for displaying pets, again the interface to obtain all the pets of a user is completed through many-to-many relationships, which will be applied in the jinja template, the following displayed notes, not in the order of creation but in accordance with the chronological order of the notes, first sorted by year and month, and then by day. In addition, the date should be converted to a more readable form, such as the full name of the month and week

5. diversification of forms and records

There are several forms elements in the site, including text boxes, numbers, dates, radio boxes, checkbox, files, passwords, etc. Because of these diverse components, it is easier for users to enter and record richer information, such as whether the pet is neutered, the weight of the pet, uploading pictures of notes, etc.

6. the uncompleted menu navigation bar

After the user logs in, if he or she wants to complete the operation of viewing pets, adding pets, logging out, changing passwords. The user must first jump to the dashboard page, causing a number of inconvenience, in the future maintenance and upgrade process, this

website will add a more user-friendly navigation bar

7. information and functions to be improved

There is no good template for the daily life of pets, such as eating, going to the toilet, walking, deworming, vaccination, etc. Note is a kind of broad category, which can be divided into several sub-categories to make the records more organized and clear.

Analysis of the architecture

1. presentation tier

This is the user interface of the website, which is realized by HTML,CSS and JS in the project. Users can complete interactions in this structure, such as submitting forms (login, register, add pets, add family, add notes), triggering events (delete pets, delete notes)

2. application tier

This is the main part, the back-end python file, where the data submitted by the front-end is processed, and the result is returned to the user as an alert message or redirect, where database related operations may also be performed. For example, user.py contains the main operations of the user, such as registering, logging in, changing password. note.py contains adding notes, deleting notes, displaying notes. pet.py is the most complex, involving operations on pets, where the user can add pets, display pets, add family members, delete pets.

3. data tier

This level is a sqlite database, which stores important information. There are four tables, which are user, pet, notes and associated tables between user and pet, and the content of each table is the front-end data, which is stored in the database after the back-end processing, such as direct storage, conversion format and so on.

Security issues

1. Authentication

Users log in to the website to view their data by entering their username and password, which are stored securely in the database. For sensitive information such as passwords, hash encryption is used to prevent leakage of sensitive information or loss of the database, which can lead to leakage of user privacy.

2. Form validation

WTF form is used in the registration and login form, which holds most security threat. It has a crsf_token, which is of high security and can prevent Cross-site request forgery.

3. Access control

In order to prevent the loss of current user's data, it is not allowed to register and log in new users if the current user is authenticated.

4.Logging

Aiming at accurately track the attacks on the site, most of the user's actions are recorded in the log file, so that the attack can be accurately located.

Unit test

In order to ensure the correctness of the back-end data processing and database, and the front-end display, the site was unit tested with the help of unittest framework before deployment. The main testing methods are simulating form submission, jumping routes. And the accuracy of the code is judged by testing the returned flash messages and the data in the database. Eventually, all tests passed, with nearly 100% coverage.

Reference

Anon. "Creating Many-To-Many Relationships in Flask-SQLAlchemy." Www.youtube.com, www.youtube.com/watch?v=OvhoYbjtiKc.

"Coding Snow | Creative Web Design and Development Tutorials." Creative Web Design and Development Tutorials, www.codingsnow.com/.

"Testing Flask Applications — Flask Documentation (2.0.x)." Flask.palletsprojects.com, 2010, flask.palletsprojects.com/en/2.0.x/testing/.