COP4710 Database Design
July 22, 2022

Uliana Ozerova
Helga Mustali
Tural Hagverdiyev

Final Report

Pet Haven of Tampa Area

I. Overview of the website

Pet Haven of Tampa Area is a fictional all-volunteer non-profit organization that works passionately to find fur-ever homes for stray, abandoned, and neglected pets, giving them a chance at the best life possible.

Our website is used to process adoptions by volunteers/administrators using a database to store, access, and modify data. The home page allows users to browse available pets, apply for adoption, and track the status of the application by logging in to the account the system automatically creates when the application is received. For volunteers, the website retrieves the information stored in the database in an organized manner and handles the back-end operations by running SQL queries that are triggered in response to the processing of applications.

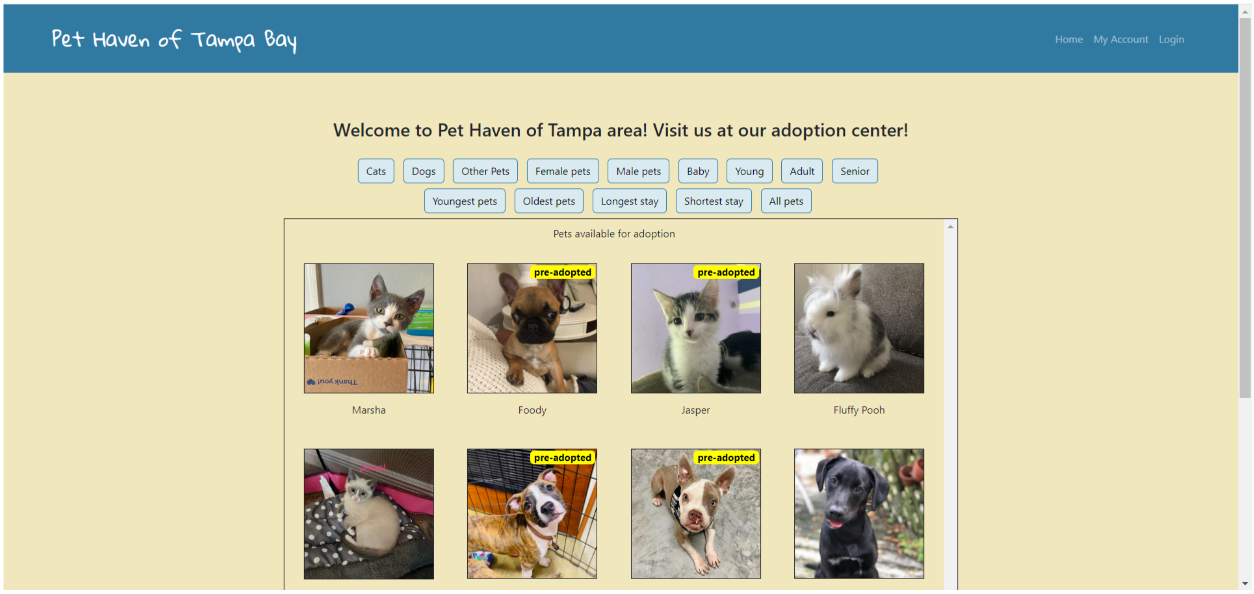
II. System Design/Technical

Back-end: Python
Front-end: Flask framework
Database: sqlite3

We have decided to use python, Flask framework, and sqlite3, an internal database that in many ways operates like most external databases such as MySql or PostgreSql but is not intended for huge corporation-like database size and does not support some of the features other client-server systems do. Besides topics related to the extraction, update, and insertion of data using queries, implementing the web app requires strong knowledge of HTML, CSS, bootstrap, JavaScript, Jinja2, and being familiar with using Shell/Command Prompt.

III. User Interface

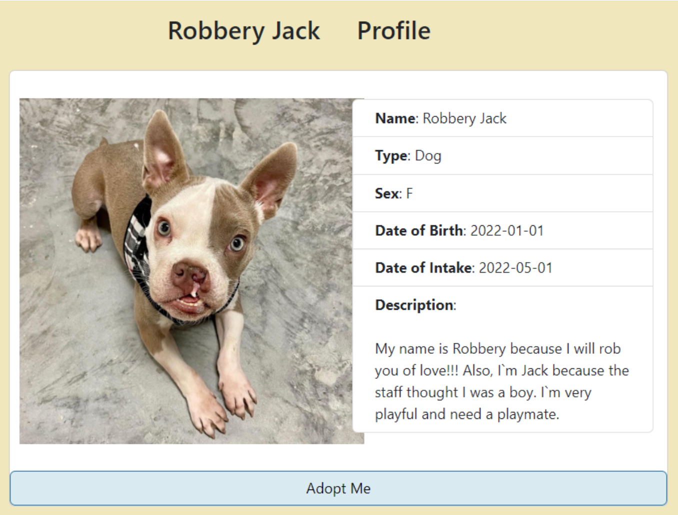
A. Home Page



Welcome to Pet Haven of Tampa area! Visit us at our adoption center!

The first view that the user will see is the home page which displays each of the pets in the database that aren't adopted yet. The page will display each of the filters, the images of the pets, and their respective names. Some pets will have a "pre-adopted" message which means that someone else is currently looking into adopting said pet. The user can still apply to adopt the pet with a "pre-adopted" message, the message will only disappear if the previous applicant's application is declined or if the pet is adopted but if it is adopted then it will no longer show on the home page.

B. Pet Profile



Robbery Jack Profile

Name: Robbery Jack

Type: Dog

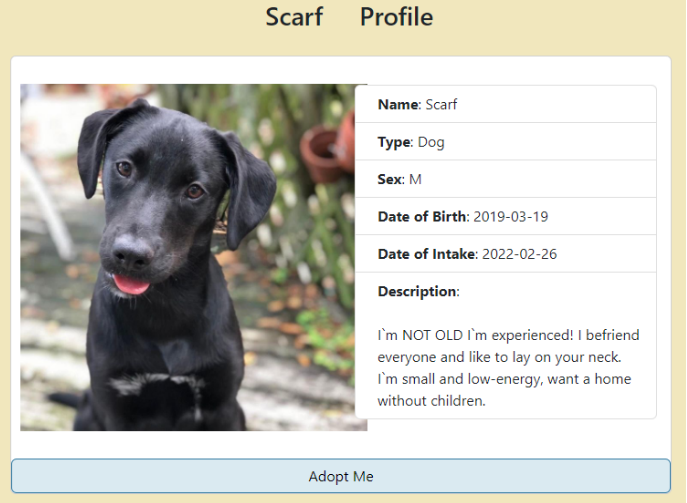
Sex: F

Date of Birth: 2022-01-01

Date of Intake: 2022-05-01

Description:

My name is Robbery because I will rob you of love!!! Also, I`m Jack because the staff thought I was a boy. I`m very playful and need a playmate.



Scarf Profile

Name: Scarf

Type: Dog

Sex: M

Date of Birth: 2019-03-19

Date of Intake: 2022-02-26

Description:

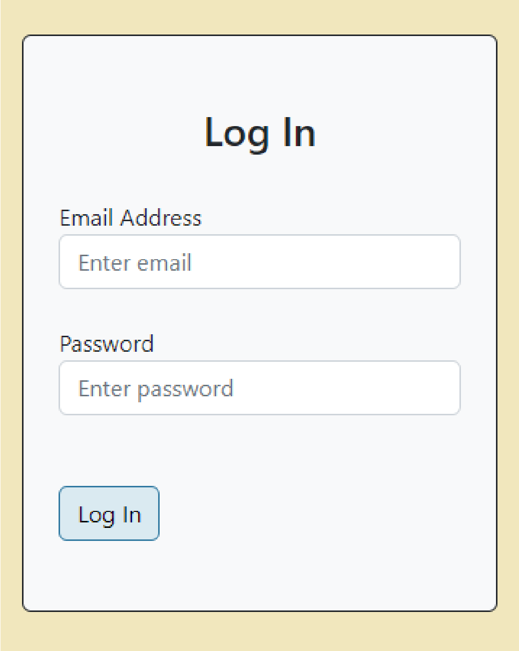
I`m NOT OLD I`m experienced! I befriend everyone and like to lay on your neck. I`m small and low-energy, want a home without children.

When the user clicks on any of the pets on the homepage they will be redirected to the appropriate pet profile page. In the pet profile, the user is able to view information about the pet they clicked on, such as the pet's date of birth or its description. On this page, they will also have the ability to apply for adoption where they can fill in an application for the chosen pet.

—> Query: " SELECT * from Pets WHERE pid = ?', [pid] "

C. Log In

When the user clicks on the "Login" button or if they're not logged in already and they press the "My Account" button they will be redirected to the Login page. On this page the user is able to log in if their email and password is valid, or else the page will be refreshed and they'll have to retry to log in again.



D. My Account

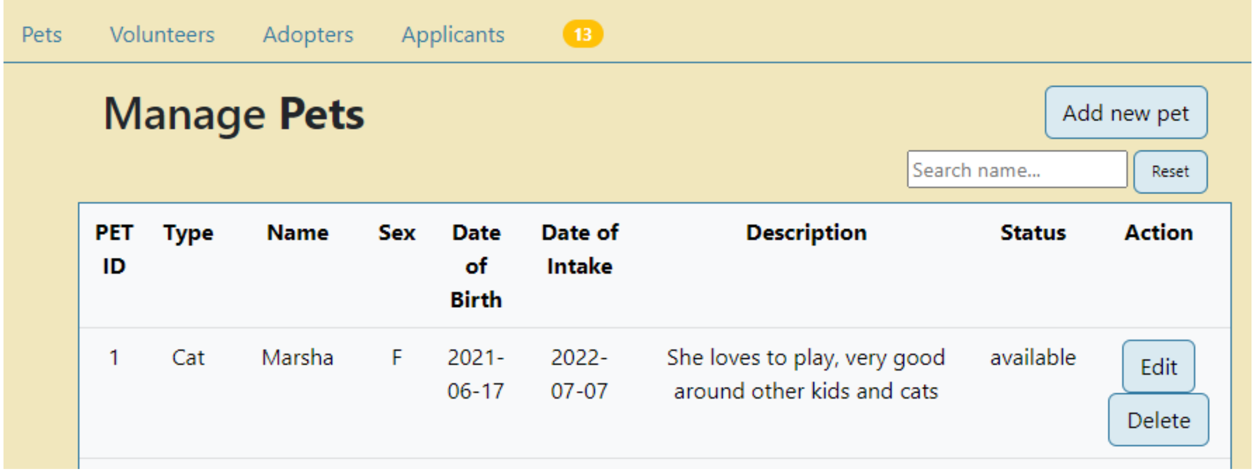
Depending on the user logged in, the views will be different if the user is an administrator or an applicant. The applicants will only be able to see their application status in the My Status tab, while the Administrator will be able to manage pets, volunteers, adopters, and applicants in their respective tabs.

1. Administrator/Volunteer

The administrator is able to make changes to the actual database. When the user is an administrator, if they sign in using the information below, they will have 4 tabs where they can make changes to the pets and applicants in the database. They also will have the ability to see all of the information stored in the database.

Email address for the Administrator: admin@usf.edu
Password for the Administrator: 12345

Manage Pets tab:

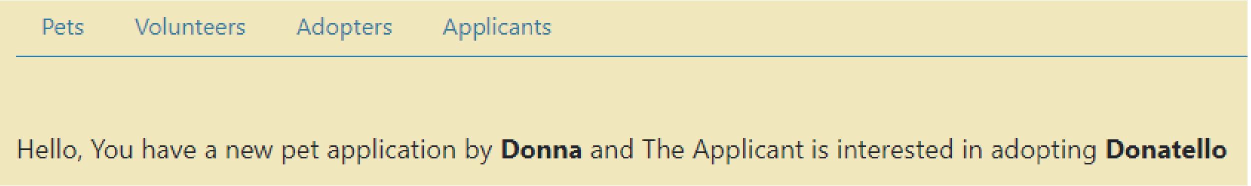


Manage Pets

PET ID	Type	Name	Sex	Date of Birth	Date of Intake	Description	Status	Action
1	Cat	Marsha	F	2021-06-17	2022-07-07	She loves to play, very good around other kids and cats	available	Edit Delete

In the manage pets tab the administrator can edit information about the pets already in the database using the “Edit” button, doing things such as editing the type of the pet or the pet’s description for example. They also have the ability to delete pets from the database using the “Delete” button. Another ability they have is to add a new pet using the “Add new pet” button where they can fill in information about the pet. All of these three changes will also make changes to the actual database since that is where they are retrieved from.

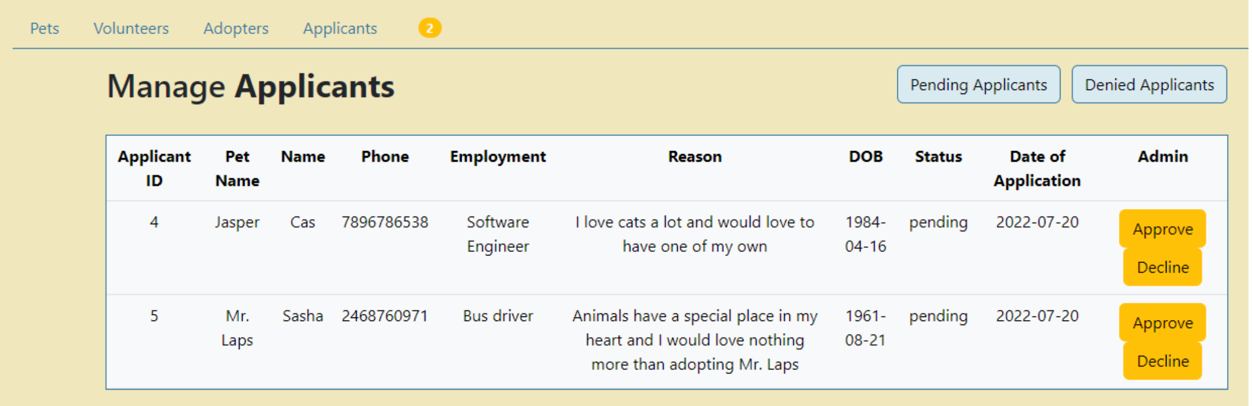
Volunteers tab:



Hello, You have a new pet application by **Donna** and The Applicant is interested in adopting **Donatello**

The administrator is able to see which is the most recent application, they’ll be able to see the applicant’s name, Donna for example, and the pet that they are applying to adopt, Donatello.

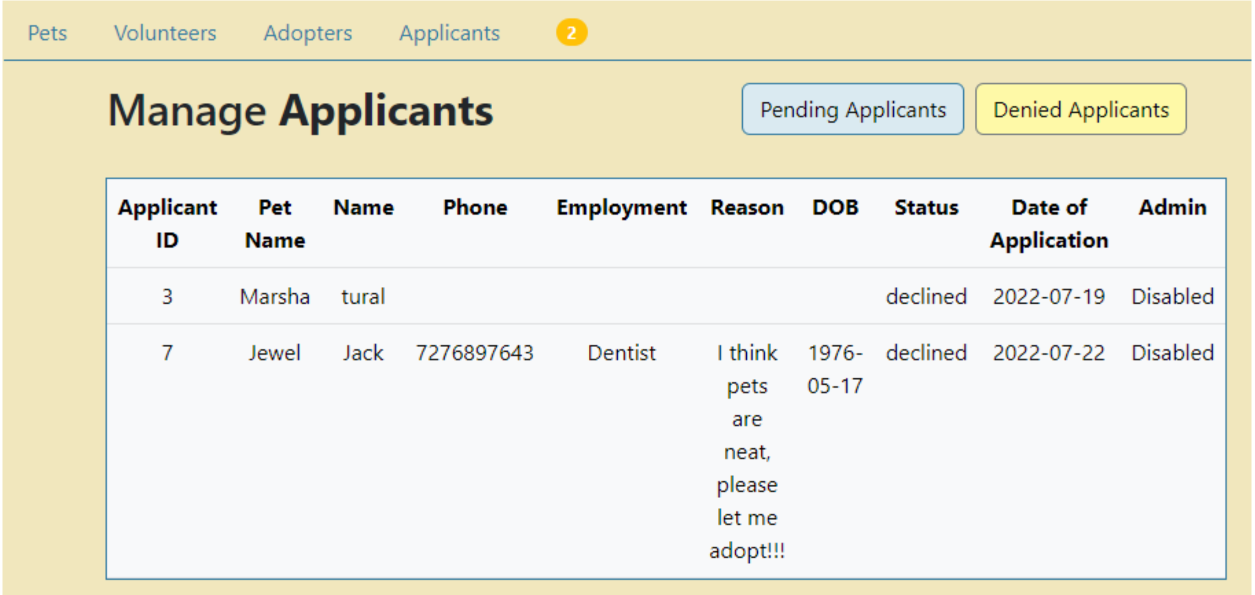
Manage Applicants tab:



Applicant ID	Pet Name	Name	Phone	Employment	Reason	DOB	Status	Date of Application	Admin
4	Jasper	Cas	7896786538	Software Engineer	I love cats a lot and would love to have one of my own	1984-04-16	pending	2022-07-20	Approve Decline
5	Mr. Laps	Sasha	2468760971	Bus driver	Animals have a special place in my heart and I would love nothing more than adopting Mr. Laps	1961-08-21	pending	2022-07-20	Approve Decline

In the manage applications tab the administrator is able to approve or decline applications. These applications are of the users that applied to adopt certain pets. The Name attribute is the applicant's name and the Pet Name shows the pet's name. This tab will only show the applications that are pending, applications with a status of 'approved' or 'declined' will not be shown, unless they click on the 'Denied Applicants' button or 'Adopters' tab.

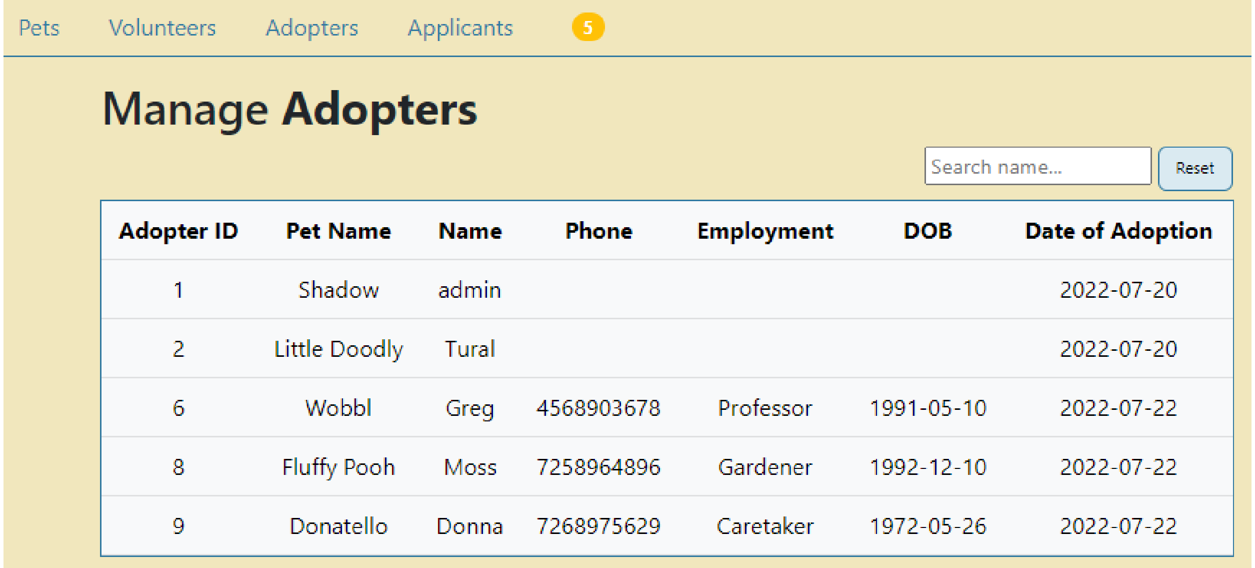
Denied Applicants:



Applicant ID	Pet Name	Name	Phone	Employment	Reason	DOB	Status	Date of Application	Admin
3	Marsha	tural					declined	2022-07-19	Disabled
7	Jewel	Jack	7276897643	Dentist	I think pets are neat, please let me adopt!!!	1976-05-17	declined	2022-07-22	Disabled

If the administrator clicks on the Denied Applicants button then only the applications with the "denied" status will be shown.

Manage Adopters tab:



Manage Adopters

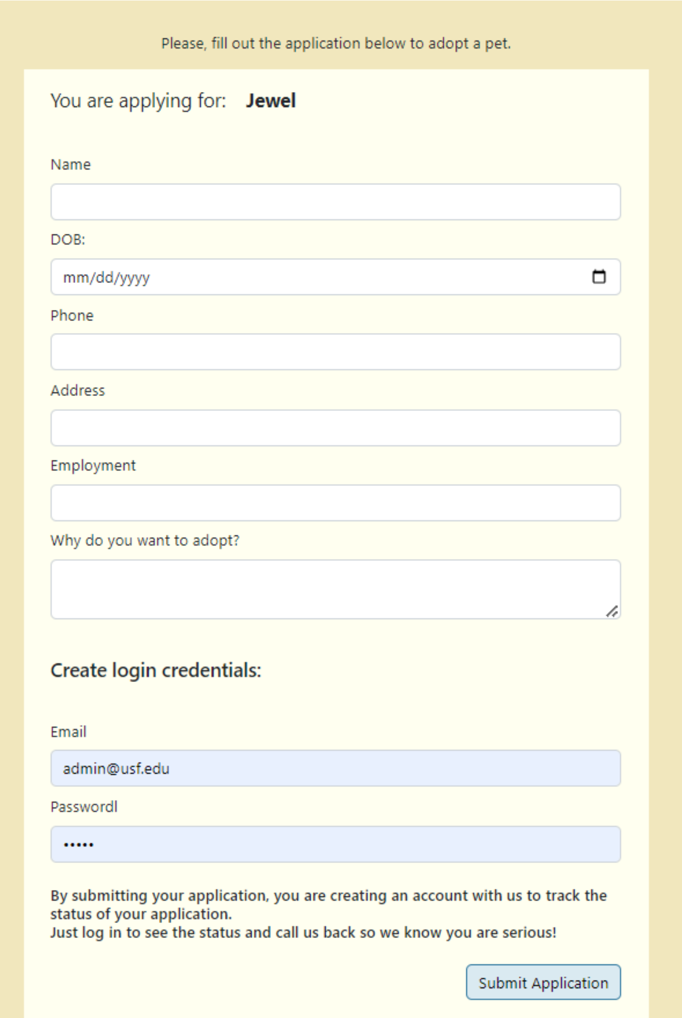
Adopter ID	Pet Name	Name	Phone	Employment	DOB	Date of Adoption
1	Shadow	admin				2022-07-20
2	Little Doodly	Tural				2022-07-20
6	Wobbl	Greg	4568903678	Professor	1991-05-10	2022-07-22
8	Fluffy Pooh	Moss	7258964896	Gardener	1992-12-10	2022-07-22
9	Donatello	Donna	7268975629	Caretaker	1972-05-26	2022-07-22

The manage adopters tab shows all of the applicants who were approved, by the administrator, and their information is retrieved from the Adopters table. The date of the application will be on the day when the application is approved.

2. Applicant/Adopter

The applicant is the person who is applying for a pet, they can only be an applicant if they registered by clicking “Adopt Me” in the pet's profile. If the applicant’s application is approved by the administrator then they will become an adopter. Anonymous users, users who are not applicants, can look at the pet profiles and once they see a pet they would like to adopt then they can apply to adopt and if their application is approved then they will become an adopter.

Email address for an Applicant: cas@mail.com
Password for an Applicant: 1234567
Email address for another Applicant: sasha@mail.com
Password for another Applicant: 1234567



Please, fill out the application below to adopt a pet.

You are applying for: **Jewel**

Name

DOB: mm/dd/yyyy

Phone

Address

Employment

Why do you want to adopt?

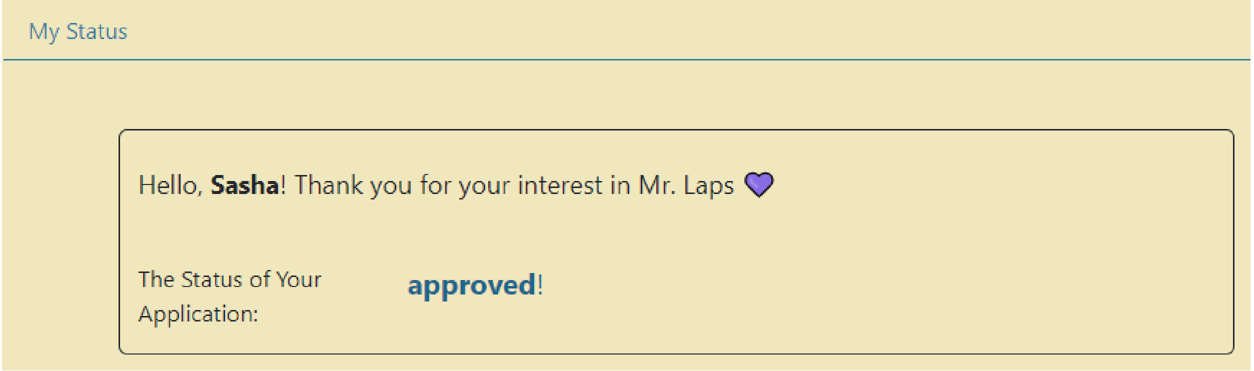
Create login credentials:

Email: admin@usf.edu

Passwordl: •••••

By submitting your application, you are creating an account with us to track the status of your application.
Just log in to see the status and call us back so we know you are serious!

My Status tab:



My Status

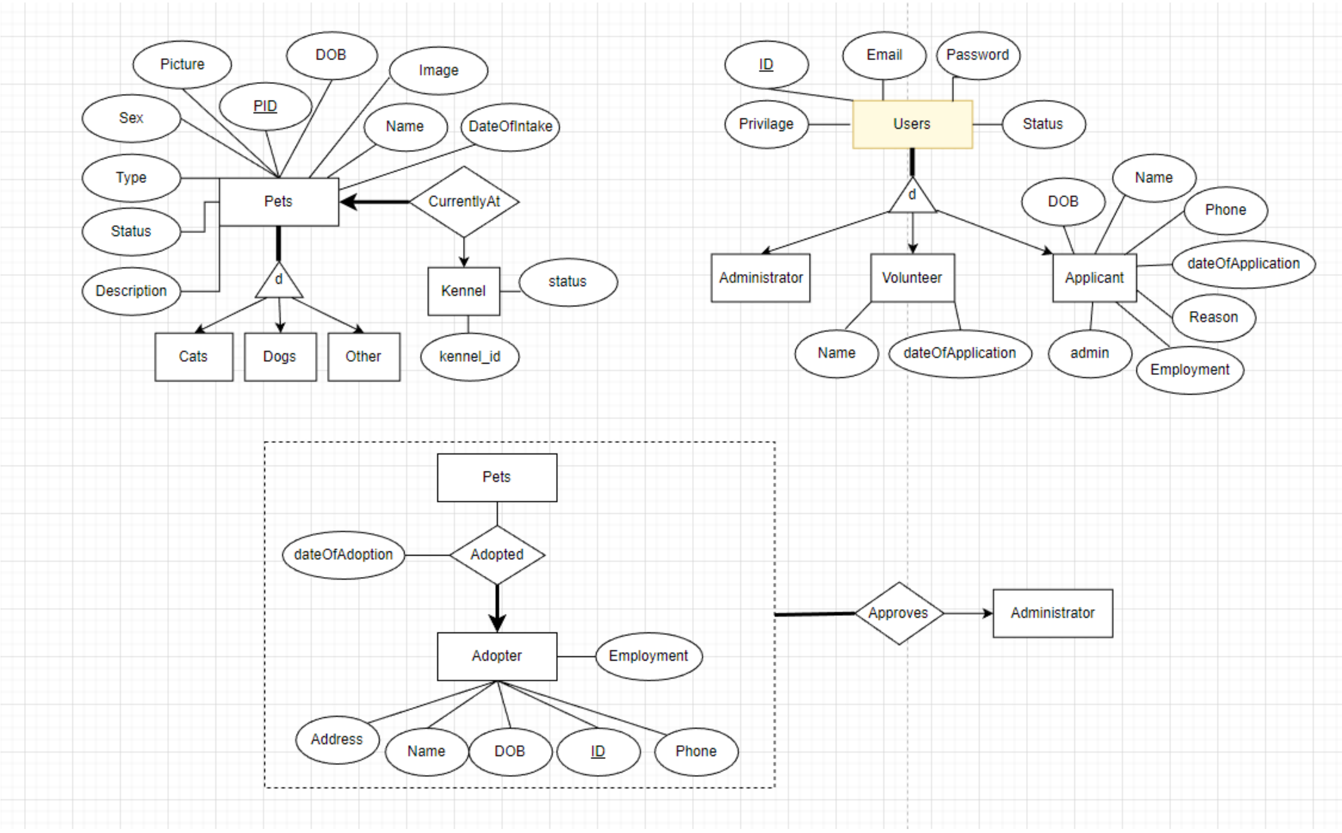
Hello, **Sasha**! Thank you for your interest in Mr. Laps 💜

The Status of Your Application: **approved**!

The applicant is able to see their status of the application in the My Status tab, unlike the Administrator, they can only see this tab. This tab will display the current user's name and the pet they are applying to adopt. Depending on whether the administrator approves the application or not, the applicant's application status will be displayed as "approved" or "declined," otherwise it will be pending.

IV. Database design

A. Relational diagram



B. **Database Tables**

Users(id, email, password, status, privilage)

- **Primary key:** id
- **Not null:** email, password, status, privilage
- **Check**(privilage in (‘low’, ‘admin’)

Applicants(id, name, phone, employment, reason, dob, dateofapplication, admin, pet)

- **Foreign keys:** id (Users), pet (Pets)
- **Not null:** name, reason, status, dateofapplication, admin
- **Check**(status in (‘approved’, ‘declined’, ‘pending’))

Volunteer(id, name, dateofapplication)

- **Foreign keys:** id (Users)
- **Not null:** dateofapplication

Adopters(id, pid, name, phone, employment, dob, dateOfAdoption)

- **Primary key:** id
- **Foreign keys:** id (Users), pid (Pets)
- **Not null:** name

Pets(pid, kennel_id, type, status, pname, sex, dob, image, dateofintake, description)

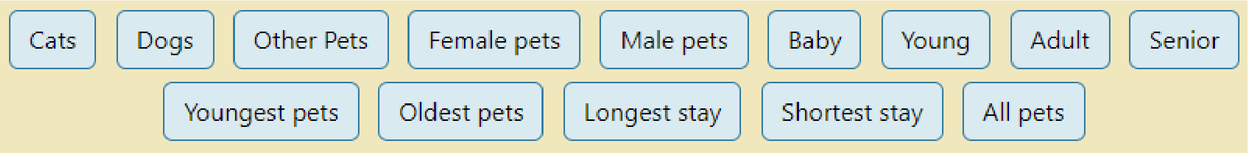
- **Primary key:** pid
- **Foreign keys:** kennel_id (Kennels)
- **Check**(status in (‘available’, ‘pre-adopted’, ‘adopted’)
- **Check**(type in (‘Dog’, ‘Cat’, ‘Other’)
- **Not null:** type

Kennels(id, status)

- **Primary key:** id
- **Not null:** status

C. Features of the website → queries used.

Filters:



On the home page, any of the users are able to view the pets and filter them or sort them by clicking on any of the filter buttons. By clicking on the buttons, only pets that fit the criteria of the filter will be displayed.

Filter by type of pet: *Cat, Dog, Other*
Query: SELECT * FROM Pets WHERE type = 'Cat' | ‘Dog’ | ‘Other’;

Filter by sex of pet: *Female, Male*
Query: SELECT * FROM Pets WHERE sex = 'F' | ‘M’;

Filter by age group of pet: *Baby*
Query: SELECT * FROM Pets WHERE strftime('%Y.%m%d', 'now') - strftime('%Y.%m%d', dob) **> 0** AND strftime('%Y.%m%d', 'now') - strftime('%Y.%m%d', dob) **< 1**');
NOTE: There is no DATE_PART in sqlite3, so strftime was used instead

Filter by age group of pet: *Young*
Query: SELECT * FROM Pets WHERE strftime('%Y.%m%d', 'now') - strftime('%Y.%m%d', dob) **>= 1** AND strftime('%Y.%m%d', 'now') - strftime('%Y.%m%d', dob) **< 3**');

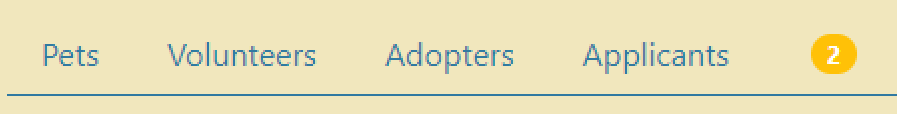
Filter by age group of pet: *Adult*
Query: SELECT * FROM Pets WHERE strftime('%Y.%m%d', 'now') - strftime('%Y.%m%d', dob) **>=3** AND strftime('%Y.%m%d', 'now') - strftime('%Y.%m%d', dob) **< 5**');

Filter by age group of pet: *Senior*
Query: SELECT * FROM Pets WHERE strftime('%Y.%m%d', 'now') - strftime('%Y.%m%d', dob) **>= 5**’);

Sort by: *Youngest to Oldest Pets, Oldest to Youngest Pets*
Query: SELECT * FROM Pets ORDER BY dob [DESC];

Sort by: *Longest to Shortest Stay, Shortest to Longest Stay*
Query: SELECT * FROM Pets ORDER BY dateofintake [DESC];

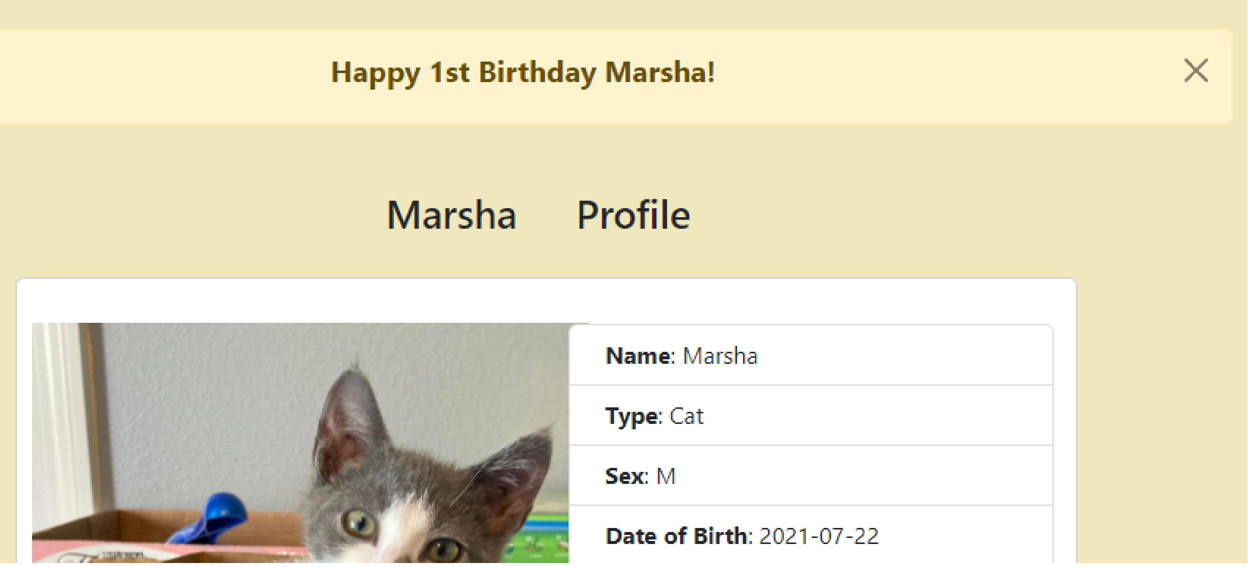
Tab count:



The orange circle at the end of the tabs shows the total current number of pets or adopters in the database.
Depending on the table, the query was: SELECT COUNT(*) FROM Pets | Adopters;

For Applicants, the query depended on the status of the application:
SELECT COUNT(*) FROM Applicants WHERE status = ‘approved’ | ‘declined’ | ‘pending’;
This query shows the total current number of applications in the database that have the status as ‘approved’, ‘declined’, or ‘pending’.

Pet birthday:



Happy 1st Birthday Marsha!

Marsha Profile

Name: Marsha

Type: Cat

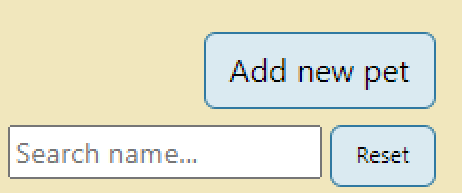
Sex: M

Date of Birth: 2021-07-22

Query to check whether the current pet has a birthday today:
SELECT * FROM Pets WHERE strftime('%d%m', dob) = strftime('%d%m', 'now') AND pid = (current pet id variable);

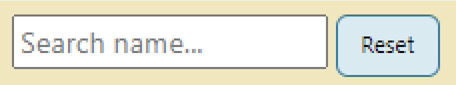
Query to calculate the pets age it it is its birthday today:
SELECT CAST(strftime('%Y.%m%d', 'now') - strftime('%Y.%m%d', dob) AS int) FROM Pets WHERE pid = (current pet id variable);

Search name (Pets tab):



A small feature that the administrator is able to use is the ability to search for a specific pet by their name or by the letter(s) that the pet's name starts with, by using this function only the specified pets with the input name will be retrieved from the database and displayed on the page.
Query used: SELECT * FROM Pets WHERE pname LIKE (petname)%;

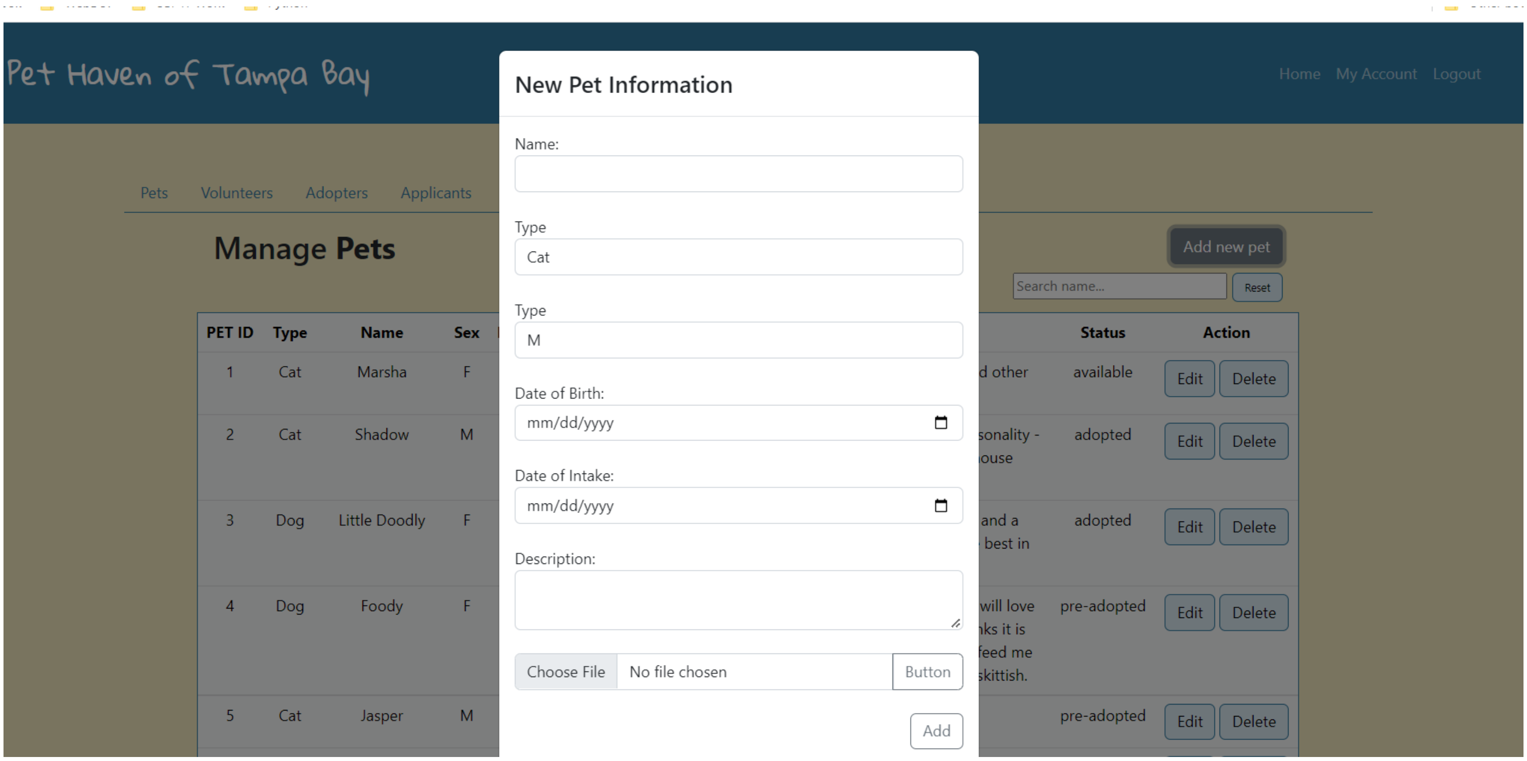
Search name (Adopters tab):



Using this functionality, the administrator is able to search for adoptions with a certain adopter's name or pets name, or letter(s) which either of the names starts with.

Query used: SELECT A.id, P.pname, A.name, A.phone, A.employment, A.dob, A.dateOfAdoption FROM Adopters A, Pets P WHERE A.pid = P.pid AND (P.pname LIKE (findNames)% OR A.name LIKE (findNames)%;

Adding a new pet/updating a new pet.



New Pet Information

Name:

Type
Cat

Type
M

Date of Birth:
mm/dd/yyyy

Date of Intake:
mm/dd/yyyy

Description:

Choose File No file chosen Button

Add

PET ID	Type	Name	Sex		Status	Action
1	Cat	Marsha	F	d other	available	Edit Delete
2	Cat	Shadow	M	sonality - ouse	adopted	Edit Delete
3	Dog	Little Doodly	F	and a best in	adopted	Edit Delete
4	Dog	Foody	F	will love ks it is eed me skittish.	pre-adopted	Edit Delete
5	Cat	Jasper	M		pre-adopted	Edit Delete

Query for updating: ' UPDATE Pets SET pname = ?, sex = ?, dob = ?, dateofintake = ?, description = ?, image = ? WHERE pid = ?', [name, sex, dob, dateofintake, description, image.filename, pid])

Query for adding: INSERT INTO Pets (kennel_id, pname, sex, dob, dateofintake, description, type, image) values ((select id from Kennels where status = 'empty' limit 1), ?,?,?,?,?,?,?)"

Updating Kennels:

UPDATE Kennels set status = 'busy' where id in (select kennel_id from Pets)

V. Ideas for improvement

After we presented the demo, we were given very helpful feedback by the professor. We were told some of the aspects for potential improvement for our project. For example, more advanced topics can be implemented for safety, security, and efficiency purposes and in addition to that, views, triggers, and more privileged concepts should be applied to the software to make it work with high performance.

Instead of fictional pets, we could pull data from actual pet databases that store pets available for adoptions using API, and connect website visitors with actual rescues, giving them an opportunity to create their own accounts and manage their own databases.

Execution plan used

Database Notes:

- Adopters have to be stored with the ID of the animal they`ve adopted.
- Store pictures for pets(**when creating a pet, a picture could be added**). If no picture -> say “picture will be posted soon”.

Homepage:

- All animals in our care(‘with status = *available or pre-adopted*’)
- Sort functions:
    - Shortest stay first
    - Longest stay first
    - —-- some other ones.
- Filter Functions:
    - Type(cat, dog, other)
    - Age group(baby(0-1), young(1-3year), adult(3-5), senior(5-...)
    - Sex

Apply to adopt/apply to volunteer:

- Automatically register applicants. All applicants are users but not all users are applicants. Users are also volunteers, adopters, and supervisors.
- **Apply to adopt** page: application has to be for a certain pet! Hence, only accessible through **Pet`s Profile**!

My account:

- **Is only seen after the log-in(login disappears, my account and logout appear)**
- **Displays:** Hello, nameOfUser.
- Is only seen by users.
- The menu depends on the privileges
    - Supervisors can: manage volunteers(can grant access and make them a supervisor), pets, and adopters —-> change their status.
        - Supervisors should see the number of applicants(badge icon by Applicants tab)
    - Applicants can: see their status and the pet they applied for.

Manage Applicants - need to store the type of application and, if adopter, who they are applying for(can be null):

- Can not edit info - has to be an approve/decline button.
    - The decision about the application is stored in the database and should be displayed to the user.
    - All applications should be deleted after a month.

- If the status of the application is approved:
    - The status of the animal is changed to ‘*pre-adopted*’
    - Applicants' info is stored in the Adopters table along with the ID of the pet they`ve applied for.
    -
- If an application is declined:
    - Change the status of the application. Denied tab.