San Jose State University

Final Project Report

“PawsCause”

A functional web application that simulates an online pet adoption site.

By Shreyas Prabhudev, Virali Patel, Carissa Lee

Department of Computer Science, San Jose State University CS 157A: Database Management

December 01, 2023

Project Introduction

Pets are known to be a man's best friend, whether that be a dog, cat, snake, or hamster. Many of these creatures, however, end up in shelters. According to ASPCA Health Insurance, 6.3 million companion animals enter U.S. animal shelters nationwide every year. There are many resources available to redirect attention of these homeless animals towards the public in hopes of being adopted, one being the online presence of shelters. If an online adoption platform is in place, it makes it much more accessible for an individual to be able to see the animals within a shelter and adopt them. “PawsCause” does just that! The application simulates what an ideal pet adoption site would look like from its functionality, user interface and necessary components to allow an employee, administrator, and client to have a fulfilling experience upon the platform.

Objective

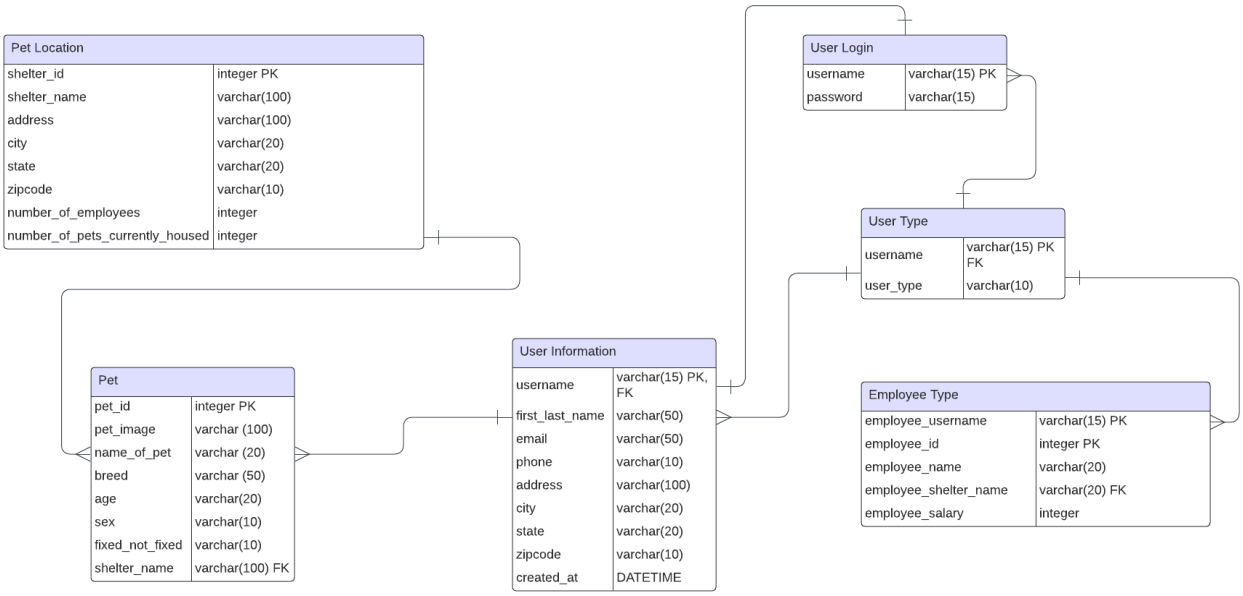
The primary objective of the project is to simulate a pet adoption site featuring functionalities which vary by user type . If registering and logging in as an ordinary client, you have the ability to view various pets by the shelter or the breed you are filtering by, as an employee you may add/remove pets and administrator access allows the admin to edit employee’s personal information or grant a user employee access. Upon adding a pet, it will display the pet’s image, name, age and sex, in which they will also be assigned to a shelter pet.

Project High-Level Design

A high level design for the application includes Bootstrap for the frontend, Django for the backend, and MySQL for the database. We initially used React for the frontend in our proposal, but there was a level of complexity that exceeded the scope of our project. To make React work, we would have to have the React and Django servers running with a RestAPI passing data between the two. To correctly handle sessions for logging in and logging out, we would also need a place to store authentication tokens.

All of these factors made Bootstrap a more appealing option since it could pass data between the Django backend to the HTML frontend through Jinja templating, making the application easier to build out. Django was a good option for the backend because there were pre-built functions that handled logging in, logging out, registering new users, and creating forms which would streamline those features. We also have views that would change based on a user's type, so employees would see a different view from a general customer or client and can only reach the url endpoints if they are validated as the correct user. This project was an exercise of creating a full stack application that connected to a database and creating features that different users had access to.

Database Design



Database Design

The database tables we included within our project include:

- “Pet Location” : shelter id, name, address, city, state, zip code, employee count, animal count
- “Pet”: pet id, image, name of pet, breed, age, sex, fixed, and shelter name
- “User Information” : username, first last name, email, phone, address, city, state, zip code, created at
- “User Type”: username, user_type
- “User Login”: username, password
- “Employee Type”: username, id, name, employee shelter name, salary

Normalization

Normalization is crucial in order to maintain the data integrity alongside minimizing redundancy within a database. The "Pet Location" table is responsible for shelter information as it exhibits characteristics aligned with 3NF. The table is well-structured and with shelter-related data efficiently organized, there are no transitive dependencies present. The primary key, "Shelter ID," serves its purpose by uniquely identifying each shelter. The "Pet" table establishes a relationship with the "Pet Location" table through the "Shelter Name" attribute as it is its foreign key. While this suggests potential normalization, the actual implementation of this relationship is crucial for ensuring 3NF compliance. Moving to the "User Information" table, which captures user details, each non-prime attribute, such as "First Name" and "Email," is directly dependent on the primary key "Username," indicating a well-organized structure. The relationships established in the "User Type" and "User Login" tables appear to align with 3NF principles, with dependencies centered around the primary key "Username." The "Employee Type" table has the primary key "Employee Username" where it should effectively drive dependencies, ensuring that attributes like "Employee Shelter Name" and "Salary" are directly linked without introducing transitive dependencies.

Results

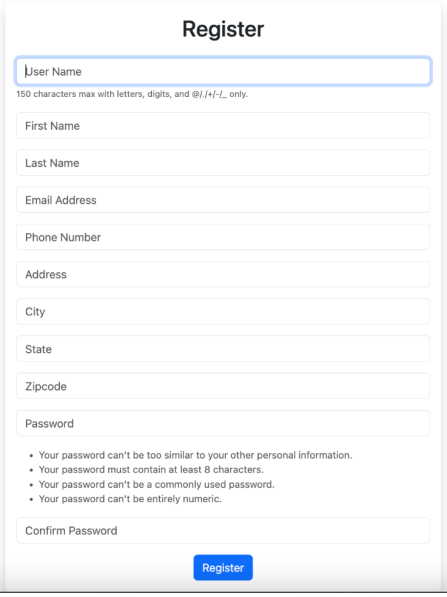
The overall result of our project includes a fully functional application in which a User is able to observe certain pets, an Employee that may add/remove pets, and an Admin that can alter Employee information. This information is stored in a MySQL database and the application itself was built through Django in the backend, Bootstrap in the frontend. We utilized pictures of our own pets for the sample animals on the website, and have functionalities of logging and registering.

Contribution

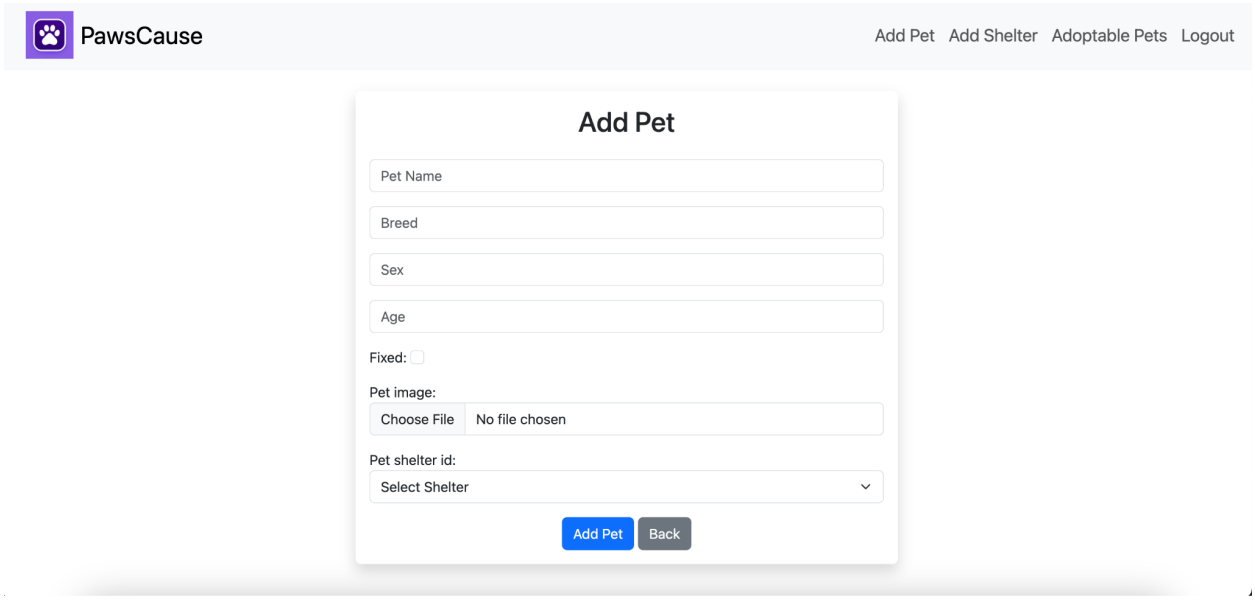
The frontend Bootstrap elements and ERD were done by Virali Patel, the backend and connection of backend and frontend were done by Shreyas Prabhudev including the MySQL database setup/connection, and the homepage were done by Carissa Lee.

Screenshots of our Application

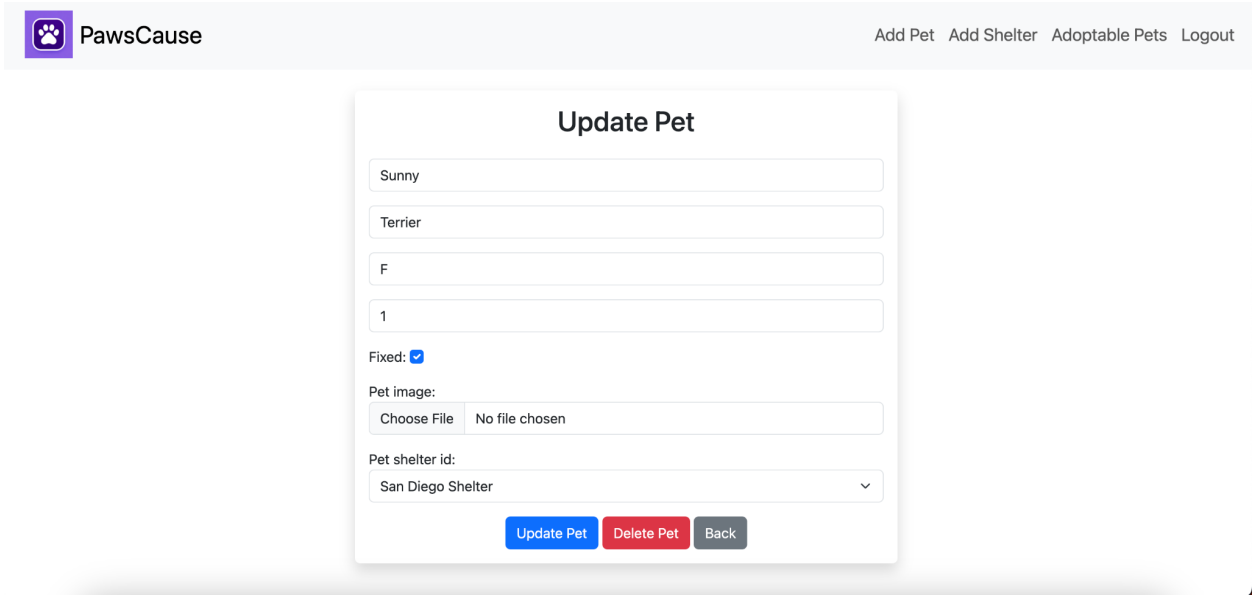
Register Screen



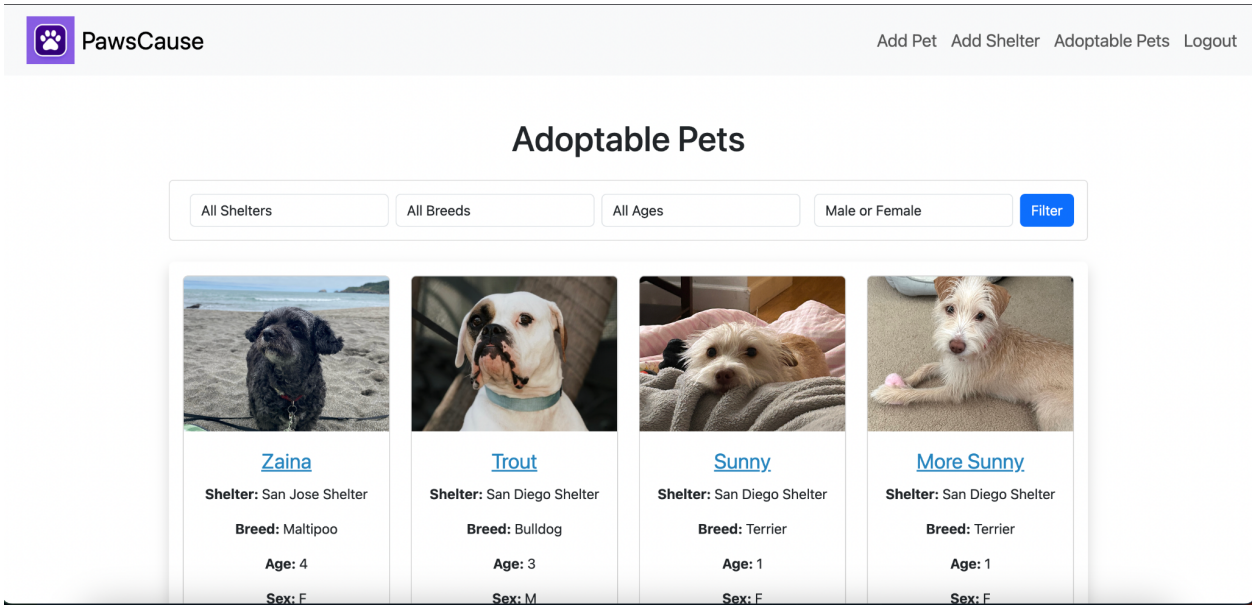
Add Pet Screen



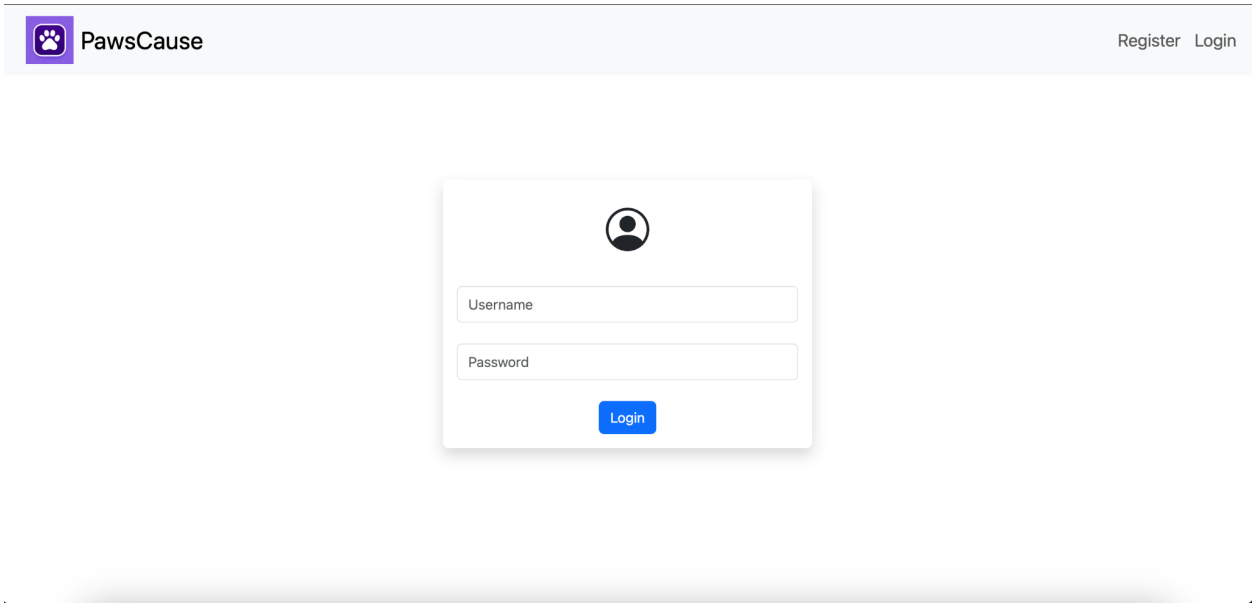
Update Pet Screen



Adoptable Pets Screen



Login Screen



References

Bootstrap Resources:

- https://getbootstrap.com/docs/4.4/getting-started/introduction/
- https://www.w3schools.com/bootstrap/

Django Resources:

- https://www.youtube.com/watch?v=t10QcFx7d5k&ab_channel=freeCodeCamp.org

MySQL Resources:

- https://www.w3resource.com/mysql/mysql-tutorials.php