The Rabid Squirrels
Pawpular
Spring 2018

Overview

Pawpular is an animal-based social media application. In this application, a user can browse the map for nearby animal services, and find many pictures and posts of local cute animals! Pawpular also includes a feed tab where a user can post media, comments or services of interest. In the services tab, our users can browse local services for their furry (or not so furry) friend. These include walking, grooming and pet-sitting. In the profile tab, a user can see his/her own posts, along with any saved content they find interesting. Lastly, our settings tab gives users options to change their username/password, and chose privacy options.

Team Members

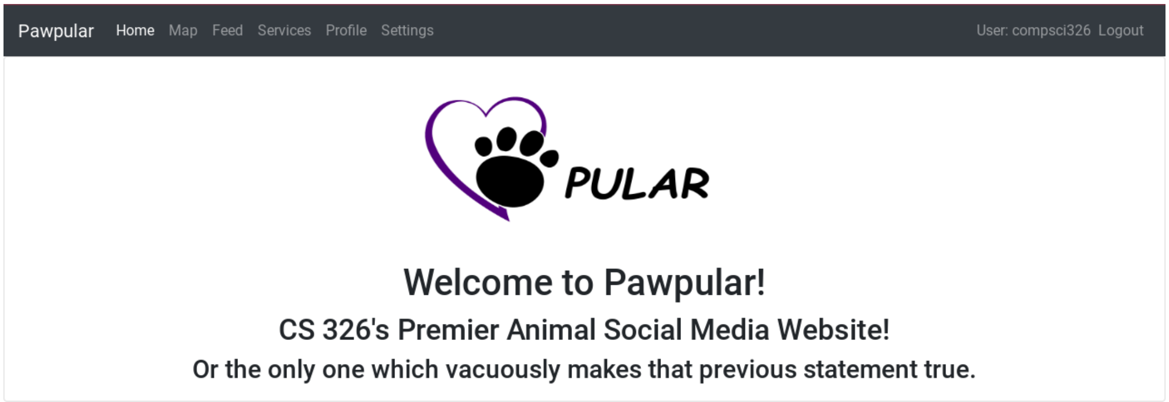
Elisa Coutant
Jillian Hynes
Ryan Mahan
Stephen Ren
Cole Reilly

Github Repository

https://github.com/Colebeep/Pawpular

User Interface

- A final up-to-date list/table describing your application's user interface. This should include the name of the UI view and its purpose. You should include a screenshot of each of your UI views.

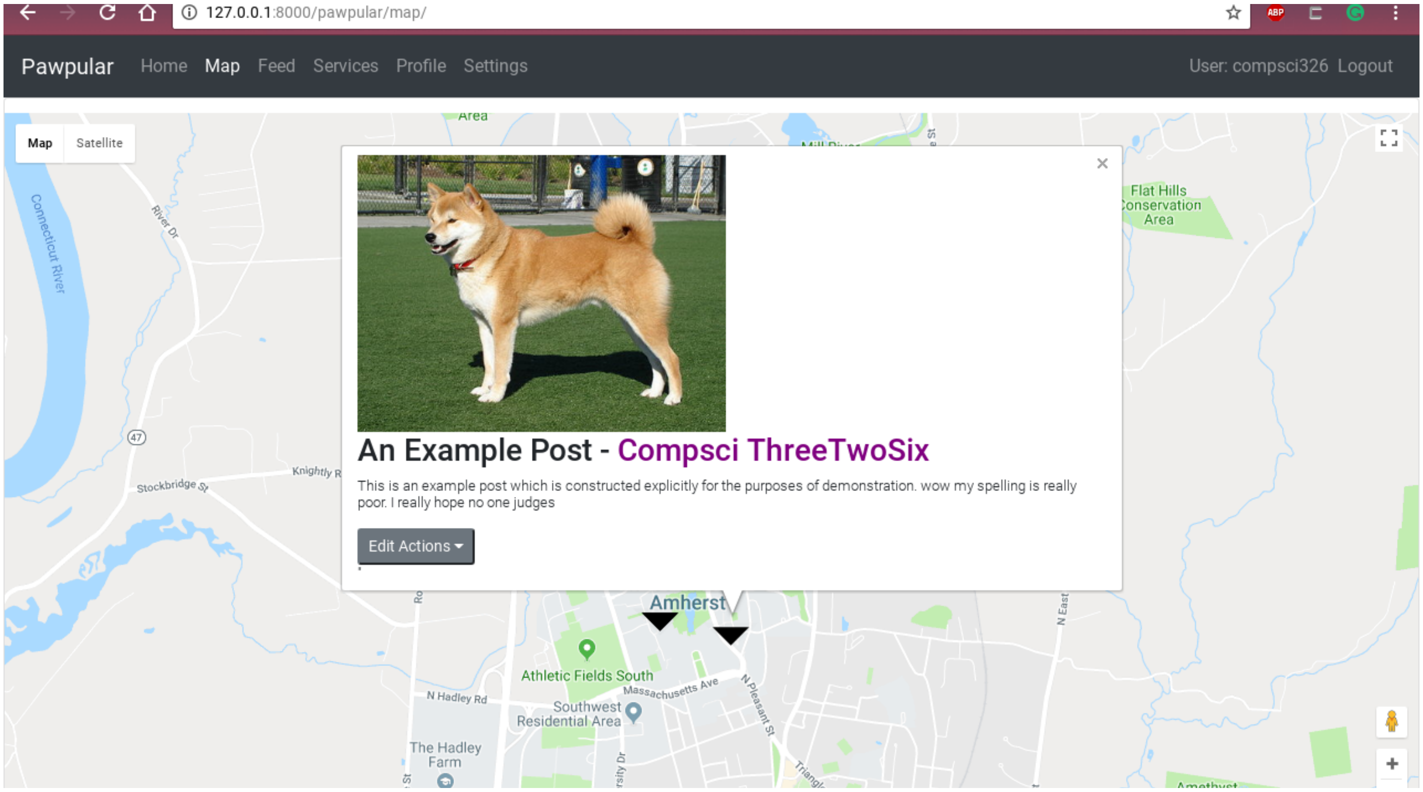


Welcome to Pawpular!

CS 326's Premier Animal Social Media Website!

Or the only one which vacuously makes that previous statement true.

Index: This is the home page of our application.



127.0.0.1:8000/pawpular/map/

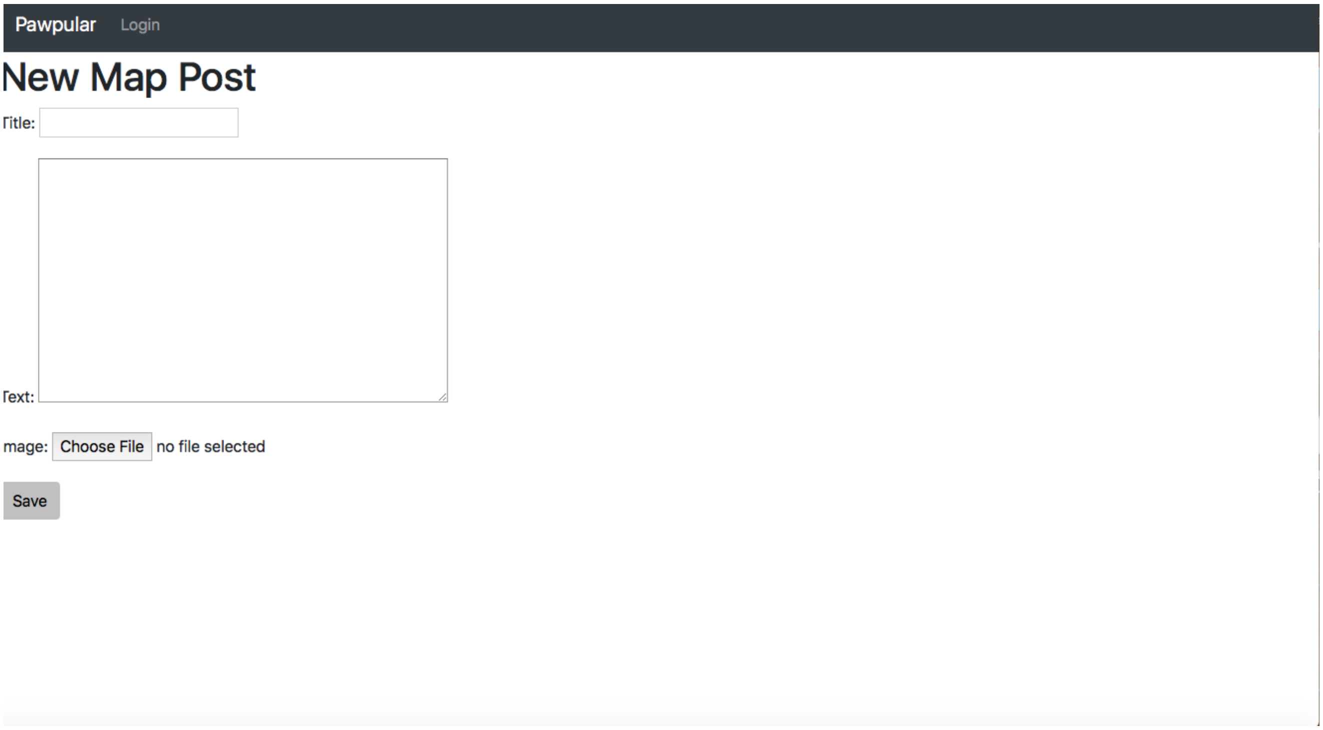
Pawpular Home Map Feed Services Profile Settings User: compsci326 Logout

An Example Post - Compsci ThreeTwoSix

This is an example post which is constructed explicitly for the purposes of demonstration. wow my spelling is really poor. I really hope no one judges

Edit Actions ▾

Map: This is the map view. It allows the user to see posts and where they originated from.



Pawpular Login

New Map Post

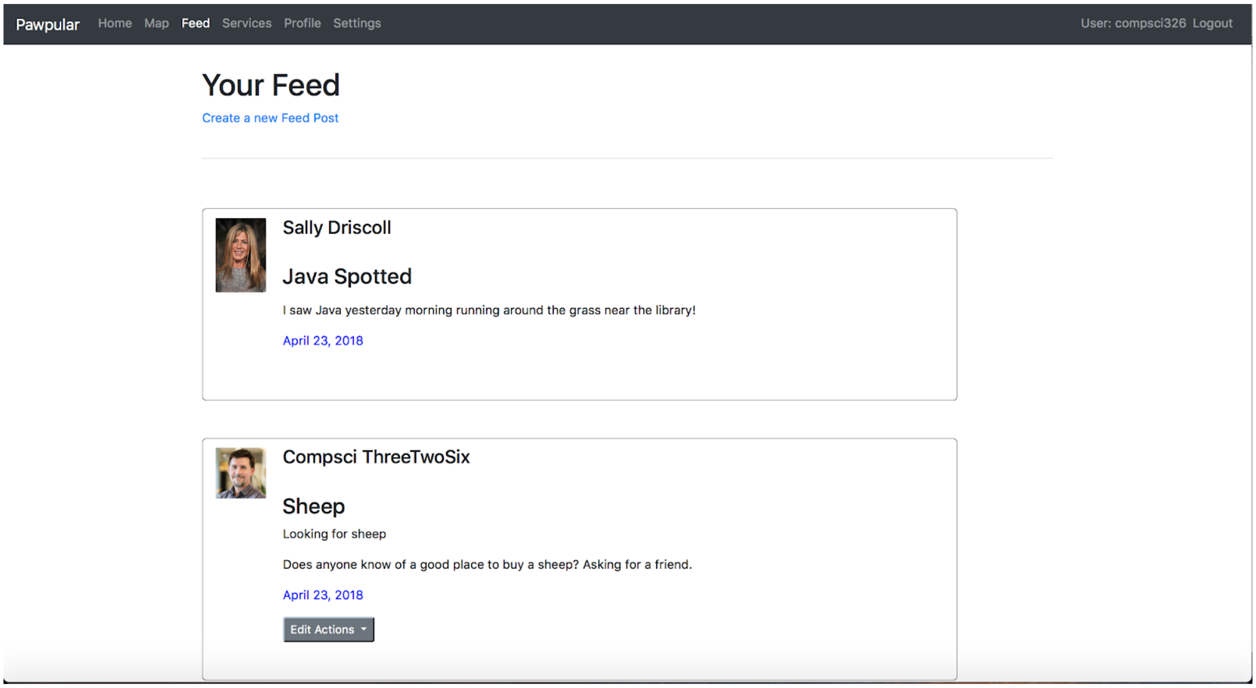
Title:

Text:

Image: Choose File no file selected

Save

New/Edit Map Post: This page allows a user to add a marker on the map with an image or text. Edit looks identically but with the prewritten data in the feilds



Your Feed

Create a new Feed Post

Sally Driscoll

Java Spotted

I saw Java yesterday morning running around the grass near the library!

April 23, 2018

Compsci ThreeTwoSix

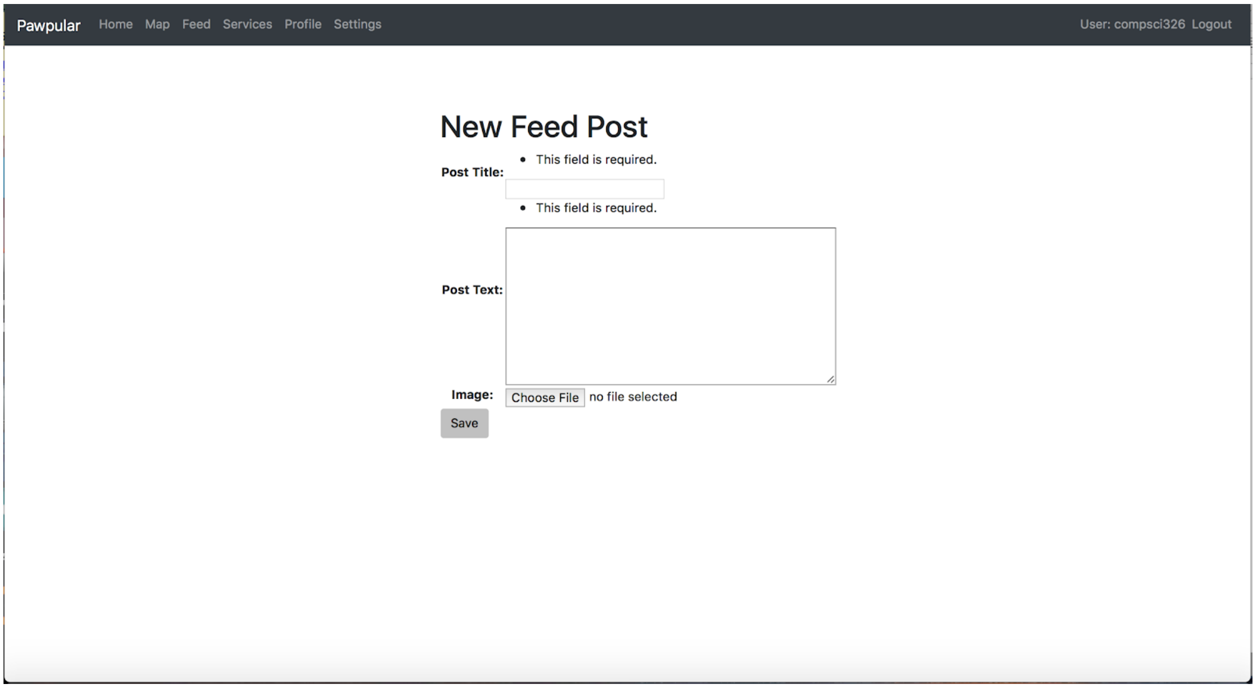
Sheep

Looking for sheep

Does anyone know of a good place to buy a sheep? Asking for a friend.

April 23, 2018

Feed: This page allows the user to see various posts posted by other users.



New Feed Post

- This field is required.

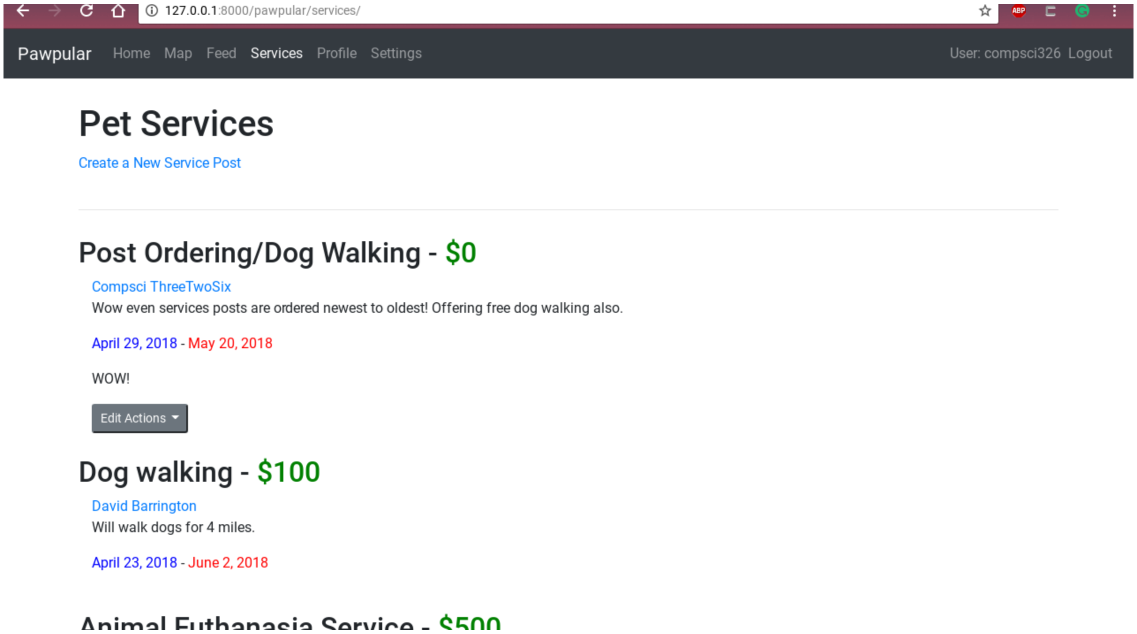
Post Title:

- This field is required.

Post Text:

Image: Choose File no file selected

New/Edit Feed Post: This page allows the user create a new post to add to the feed and his/her profile. Edit looks identically but with pre populated fields.



Pet Services

Create a New Service Post

Post Ordering/Dog Walking - $0

Compsci ThreeTwoSix

Wow even services posts are ordered newest to oldest! Offering free dog walking also.

April 29, 2018 - May 20, 2018

WOW!

Dog walking - $100

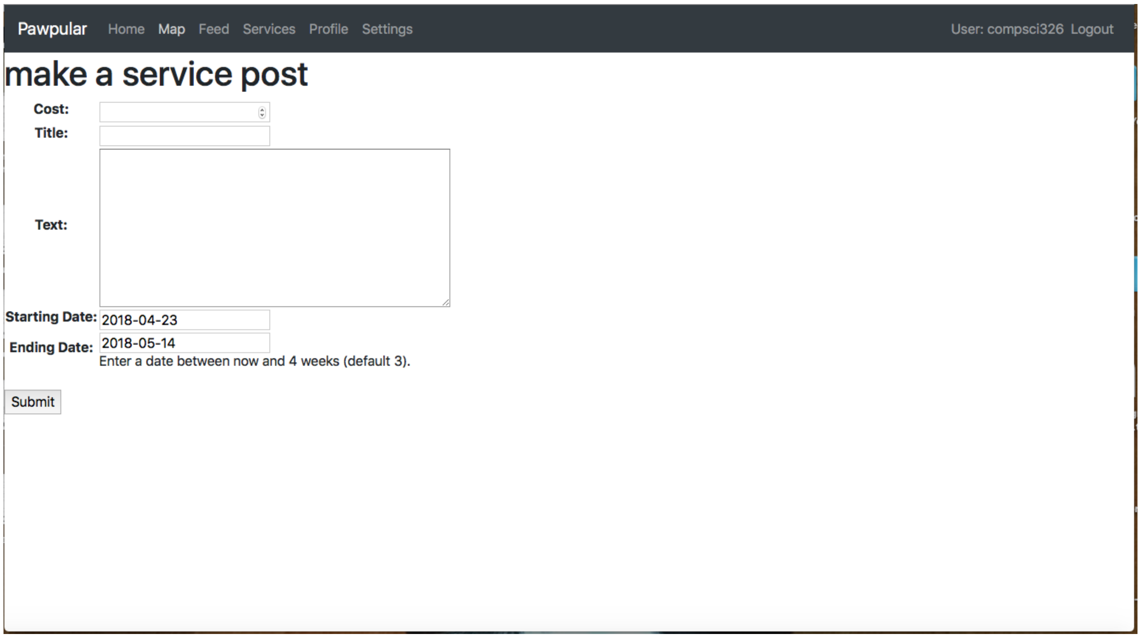
David Barrington

Will walk dogs for 4 miles.

April 23, 2018 - June 2, 2018

Animal Euthanasia Service - $500

Services: This page allows the user to see services being offered and sought out by other users.



make a service post

Cost:

Title:

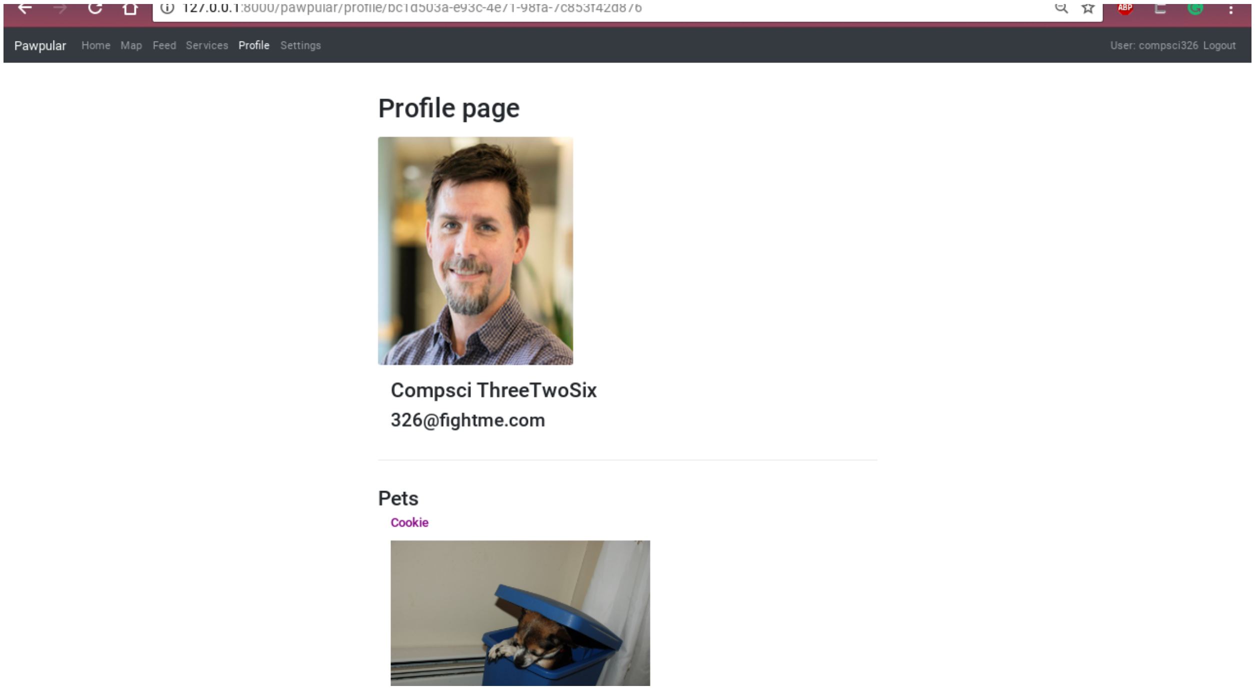
Text:

Starting Date: 2018-04-23

Ending Date: 2018-05-14

Enter a date between now and 4 weeks (default 3).

New/Edit Service Post: This is the page where a user can make a service post of their own. Edit Looks identical but with pre populated fields.


127.0.0.1:8000/pawpular/profile/bc1d503a-e93c-4e71-98fa-7c853f42d876
Pawpular
Home
Map
Feed
Services
Profile
Settings
User: compsci326
Logout

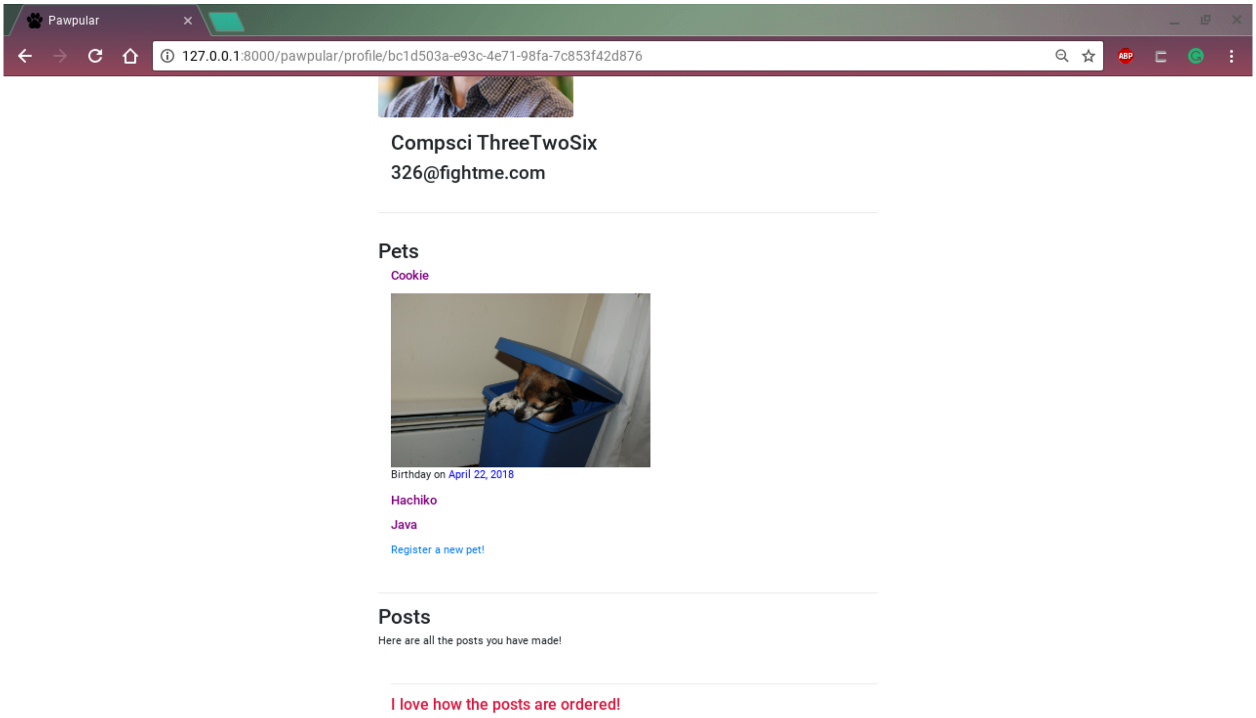
Profile page

Compsci ThreeTwoSix

326@fightme.com

Pets

Cookie


Pawpular
127.0.0.1:8000/pawpular/profile/bc1d503a-e93c-4e71-98fa-7c853f42d876

Compsci ThreeTwoSix

326@fightme.com

Pets

Cookie

Birthday on April 22, 2018

Hachiko

Java

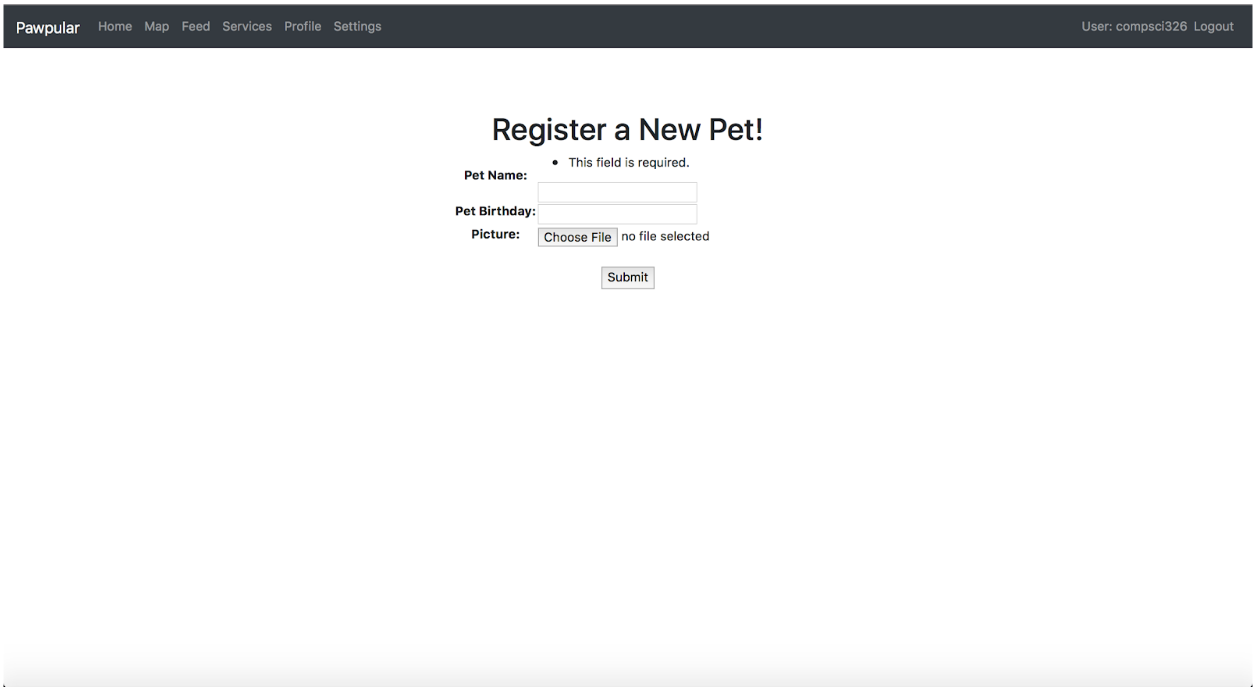
Register a new pet!

Posts

Here are all the posts you have made!

I love how the posts are ordered!

Profile: This page shows the user's profile, including his/her posts, pets, and services.



Register a New Pet!

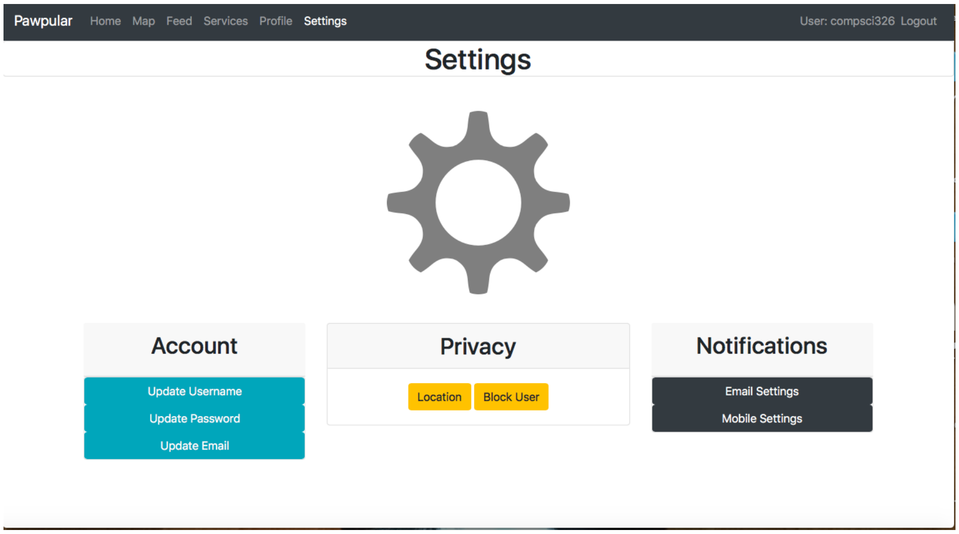
- This field is required.

Pet Name:

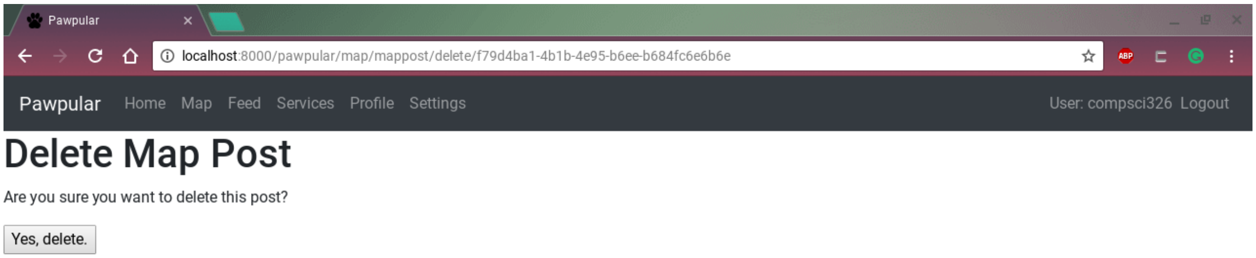
Pet Birthday:

Picture: no file selected

New Pet: This form allows a user to create a new pet to add to his/her profile.



Settings: This is the settings page that contains information on a user’s account.


Pawpular
localhost:8000/pawpular/map/mappost/delete/f79d4ba1-4b1b-4e95-b6ee-b684fc6e6b6e
Pawpular
Home
Map
Feed
Services
Profile
Settings
User: compsci326
Logout

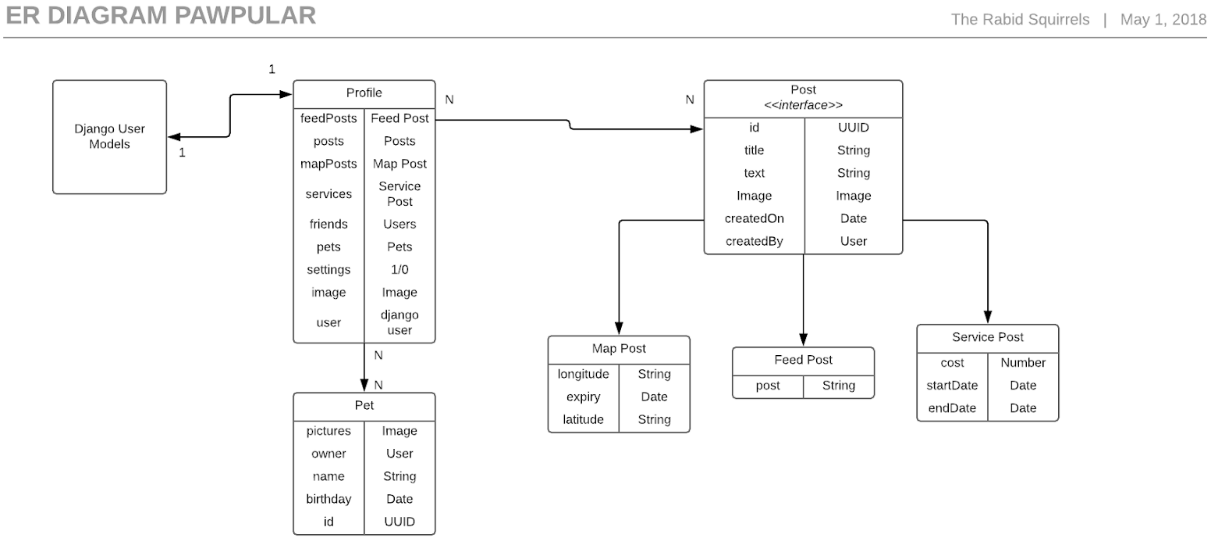
Delete Map Post

Are you sure you want to delete this post?

Delete Map/Service/Feed Post: This page confirms the attempt to delete a post, all three posts look identical save the title.

Data Model

- A final up-to-date diagram of your data model including a brief description of each of the entities in your model and their relationships.



Profile	
feedPosts	Feed Post
posts	Posts
mapPosts	Map Post
services	Service Post
friends	Users
pets	Pets
settings	1/0
image	Image
user	django user

Pet	
pictures	Image
owner	User
name	String
birthday	Date
id	UUID

Post <<interface>>	
id	UUID
title	String
text	String
Image	Image
createdOn	Date
createdBy	User

Map Post	
longitude	String
expiry	Date
latitude	String

Feed Post	
post	String

Service Post	
cost	Number
startDate	Date
endDate	Date

Name	Purpose
User	Django User model can allow us to access user info from the request, verify login, redirect if not logged in, and other django backend functions.
Profile	With a one to one relationship with the user model this allows us to access more data if we have the user, and store relationships with other models and easily access it.
Post	Post acts as an abstract model for the different types of posts on our page, map posts, feed posts, and service posts. This allows us to organize our data models better and maintain the redundancies between the types of posts in one class. Posts are related to profiles, so we can access posts from the profile and display it whether we have a profile or a post. Each type of post adds a unique field to their type of post, relevant to it.
Pet	Pet stores the data of a pet, connecting it with an owner, so we can identify its owner and display the pet if we have the owner.

URL Routes/Mappings

- A final up-to-date table of all the URL routes that your application supports and a short description of what those routes are used for. You should also indicate any authentication and permissions on those routes.)

URL	Description	Permissions
index/	The homepage of our application	Must be logged in to access this page
chat/	This page contains both textual and image posts from users	Must be logged in to access this page
chat/feedpost/new	This page is where a new feedpost can be created	Must be logged in to access this page
chat/feedpost/edit/<uuid:pk>	This page is where an already existing feedpost of uuid pk can be edited	Must be logged in to access this page
chat/feedpost/delete/<uuid:pk>	This page allows user to confirm deleting a feed post with uuid pk	Must be logged in to access this page

map/	This page contains our map with posts from users	Must be logged in to access this page
map/mappost/new	This page allows the user to create a new mappost with a location, an image and text	Must be logged in to access this page
map/mappost/edit/<uuid:pk>	This page allows user to edit a map post with uuid pk	Must be logged in to access this page
map/mappost/delete/<uuid:pk>	This page allows user to confirm deleting a map post with uuid pk	Must be logged in to access this page
services/	This page contains a list of services being offered or sought by users	Must be logged in to access this page
services/new	This page allows a user to create a new service post	Must be logged in to access this page
services/servicepost/edit/<uuid:pk>	This page allows user to edit a service posts with uuid pk	Must be logged in to access this page
services/servicepost/delete/<uuid:pk>	This page allows user to confirm deleting a service post with uuid pk	Must be logged in to access this page
settings/	This page contains the settings for a user's account	Must be logged in to access this page
profile/	This page contains the user info such as their name and current posts	Must be logged in to access this page
profile/create_pet	This page allows the user to create new pets for their profile	Must be logged in to access this page

Authentication/Authorization

Users are authenticated using the default django authentication mechanisms. Each user object kept in the database was mapped one to one to a profile in the database. So the User would contain the contact login and name details while the profile kept data pertaining to posts and pets. All the pages besides the home page require the user to login, and the profile page is constructed according to the logged in user's data. If the user matches the createdby field of a post, a clickable drop down is added, allowing the user to edit and delete said post.

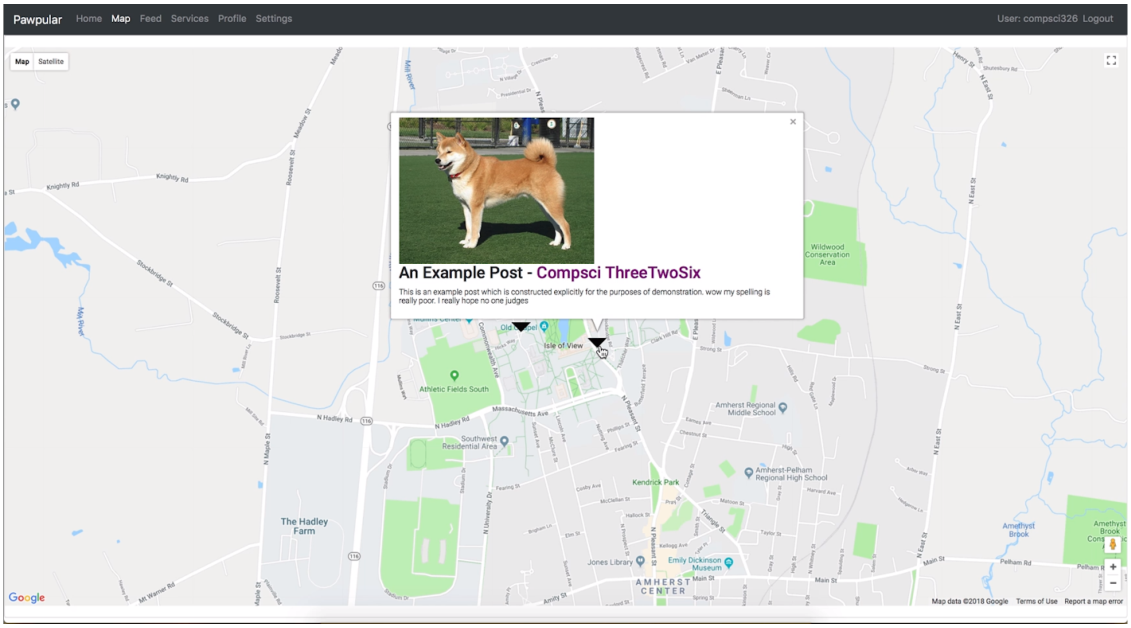
Team Choice

For our team choice we decided to include a map in the application which required us to set aside its own view and url mapping. It includes extensive usage of the Google Map api, and a javascript implementation of it in the frontend page. This involved writing multiple custom functions within the Google Maps API and writing javascript functions to accompany those. We also had to include Django's database grabbing functionality (the {% %} notation) into the javascript which led to a whole suite of new problems. To implement a django form to be able to map a post to a location on the map we had to use django's urls to pass data from the map view to the form, which effectively allows the form to grab all the information it needs from a custom url displaying latitude/longitude. All of the finished work for our team choice can be seen in the map view template, in the views function mappost_new, and in the forms.py file

Conclusion

This project could best be described as fitting all the puzzle pieces together. Each step of the way, we followed the provided tutorials to put together a working application. Using the material learned from the tutorials and fitting them to our application was a great way to better understand web programming. Some difficulties arose whenever we implemented pieces of our project without the use of the material provided in class. This led to extended research on different topics not covered previously and working off of each others' code to make our application the best it could be. At times it was difficult to build off of one another's work due to ambiguities in the code but strong communication among our team and willingness to learn new programming techniques allowed us to overcome these hurdles.

Slide! (included in the github folder)



An Example Post - Compsci ThreeTwoSix

This is an example post which is constructed explicitly for the purposes of demonstration. wow my spelling is really poor. I really hope no one judges