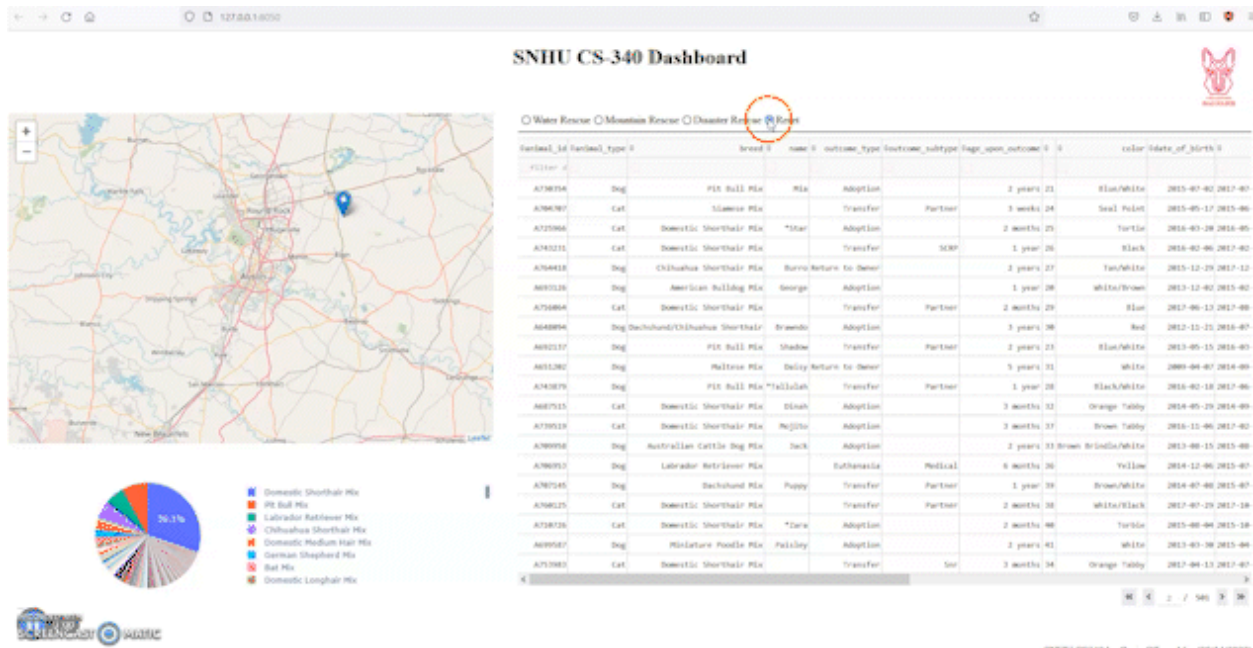


## CS 340 GS Dashboard README

### About the Project/Project Title

GS Dashboard – a quick and easy way to visualize animal shelter data!



### Motivation

GS Dashboard is an open-source tool created for viewing information about shelter animals as a project for the rescue-animal training company Grazioso Salvare (recommend checking out their work on their website). Thanks to them they have allowed this tool to be open-source to aid anyone looking to do similar things!

## Getting Started

GS Dashboard is built in Python 3 using the Dash framework. It further uses MongoDB, a modern, non-relational database to store and access information. To serve as an API for the database, we are using our own MongoCRUD module which is built on top of the PyMongo library. Finally, this project also relies on the popular Python library pandas for data analysis.

### Why MongoDB?

MongoDB is a flexible database with well established APIs to work with many languages. Setup is minimal and much of the backend is taken care of by default without thinking about it. It is a simple and modern solution that is also scalable to the project size.

### Why Dash?

Dash makes it easy to create complex tables and other dataviews directly in Python. It also ports out access to Leaflet, a popular open map engine, directly to Python, that is otherwise accessible in JavaScript. Finally, it makes running a new local server instance as simple as a single line of code.

## Installation

venv – *recommended* – create a new environment to isolate projects and avoid version confusion  
`python -m venv PATH`

pymongo - **required** – use pip to install from pyPI  
`pip install pymongo`  
or  
`python -m pip install pymongo`

mongo\_crud – **required** – currently not available on pyPI  
Check the documentation [here](#)

mongoDB – **required** – database engine  
A free local instance of a mongoDB server can be created using MongoDB Community Server  
<https://www.mongodb.com/try/download/community>  
and following instructions in the associated documentation

Dash framework – **required** – use pip to install from pyPI  
`pip install dash`

pandas – **required** – use pip to install from pyPI  
`pip install pandas, numpy`

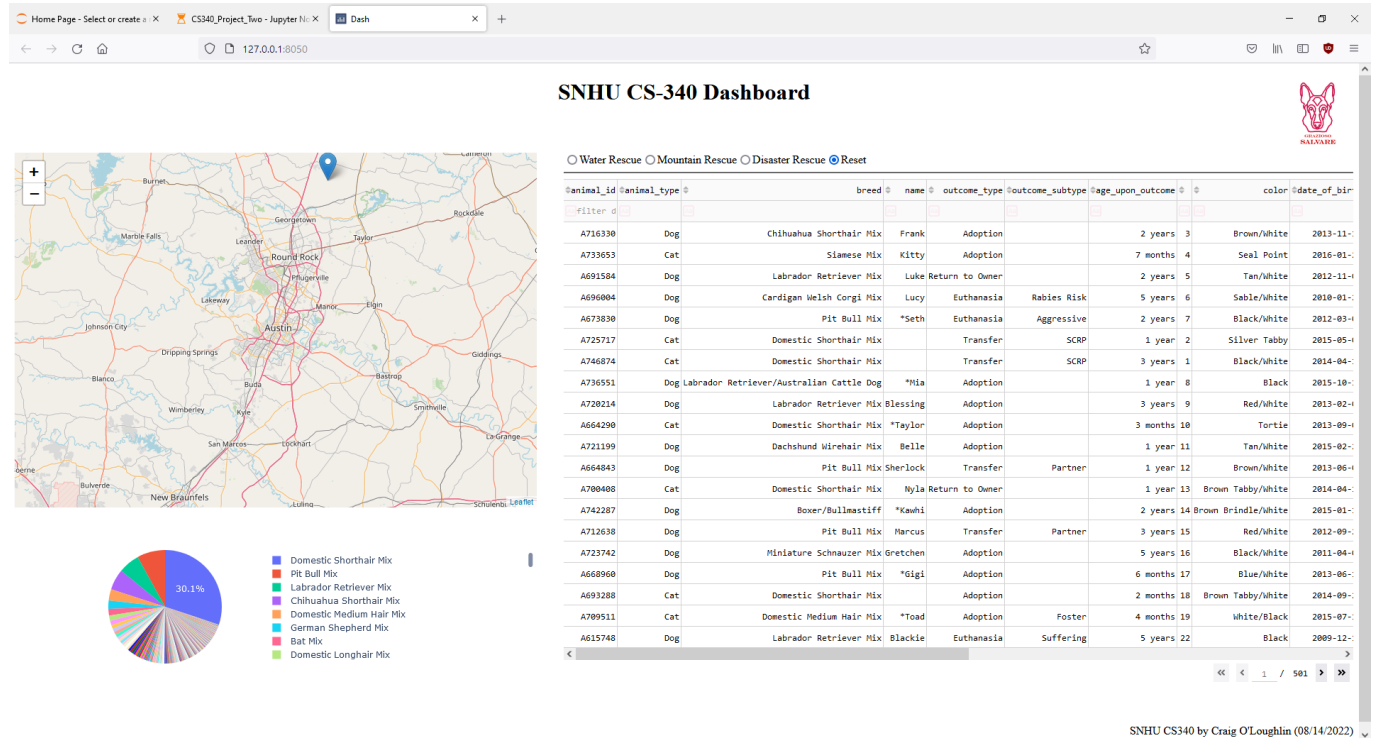
## Usage

GS Dashboard will need a MongoDB server instance with an active database along with configuring the MongoCRUD module to the proper server location and login information. Check out the MongoCRUD documentation for assistance with this.

Afterwards, simply run GS Dashboard to start up a local Dash server.

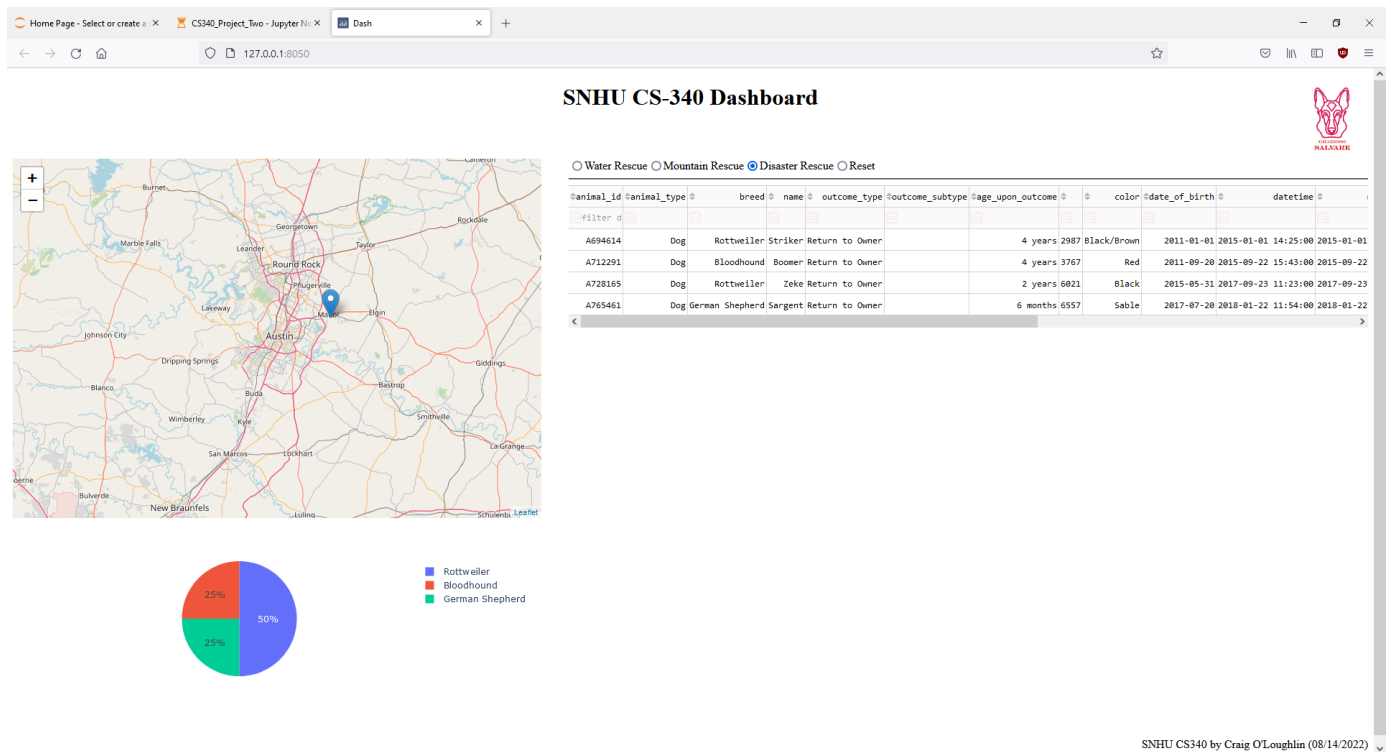
## Examples

### 1. The default dashboard view upon loading. All database entries are displayed.

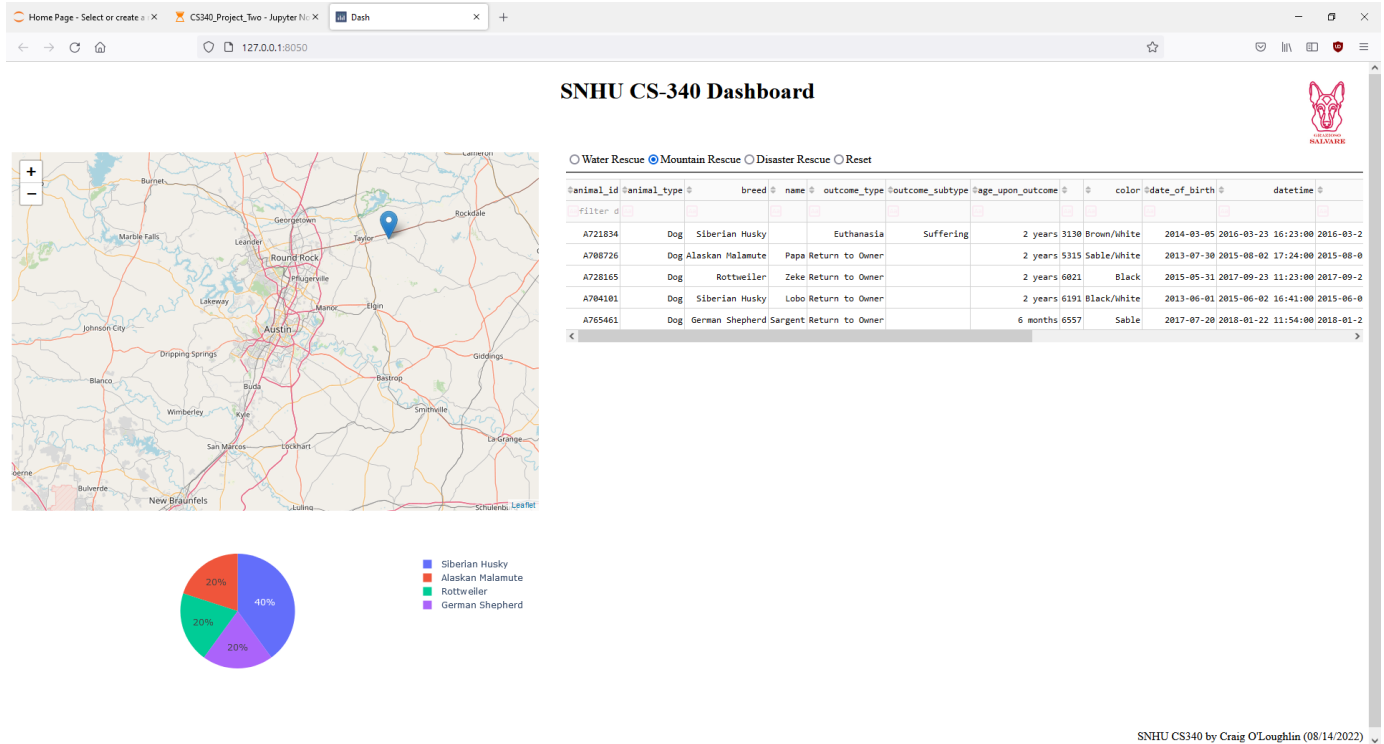


2. Use the filter radio buttons to sort only the animal information you like. These radio filters are customizable in the dashboard module code if you would like to change search criteria or add more.

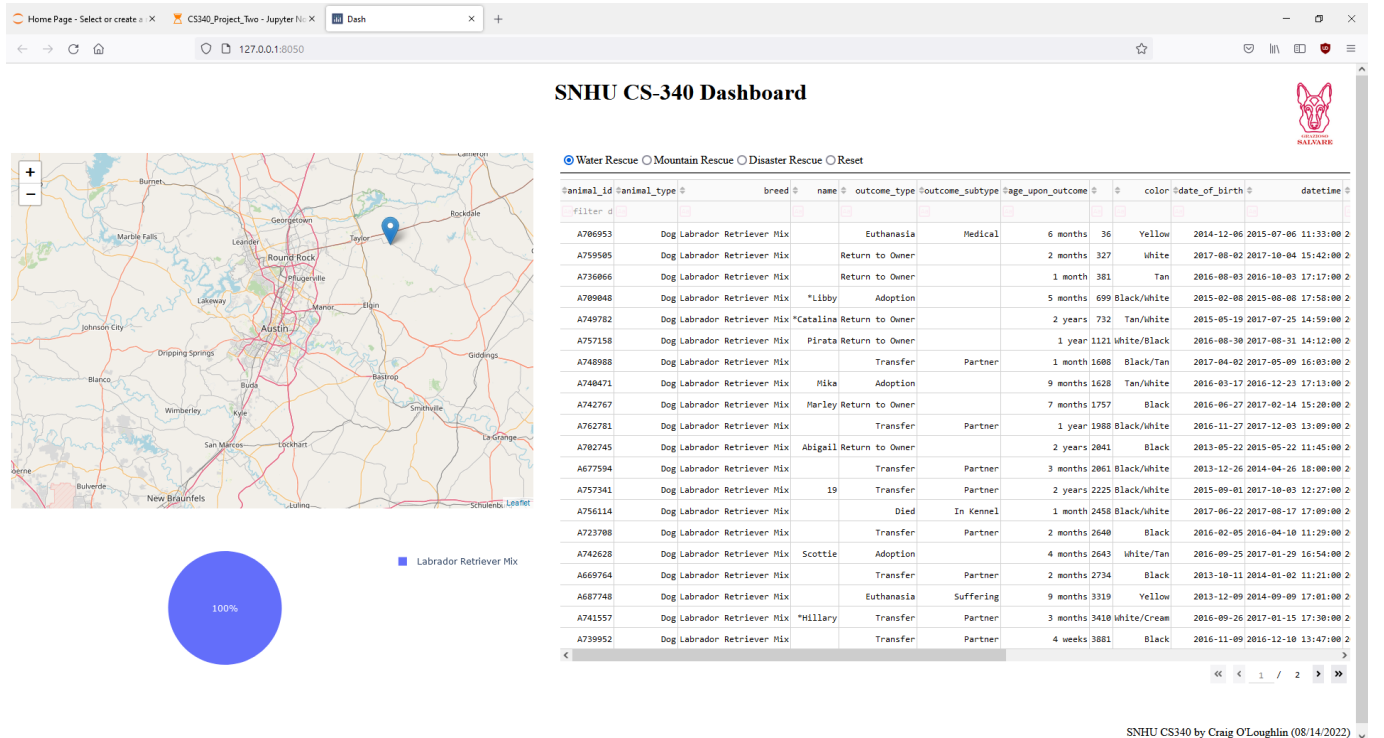
3. Disaster Rescue selected.



#### 4. Mountain Rescue selected.



#### 5. Water Rescue selected.



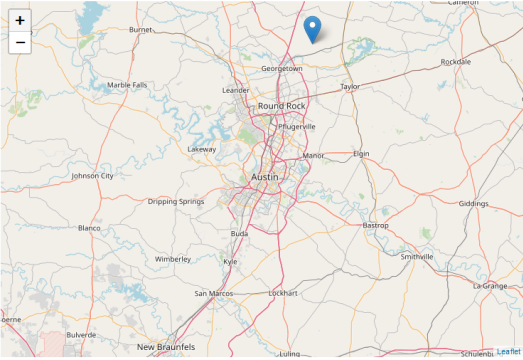
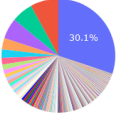
6. Click anywhere on a row of data to select it. This animal's geolocation appears on the map. If no row is selected, by default the first animal on that page appears on the map instead.

Home Page - Select or create x CS340\_Project\_Two - Jupyter Notebook x Dash x +

127.0.0.1:8050

### SNHU CS-340 Dashboard

Water Rescue Mountain Rescue Disaster Rescue **Reset**

animal_id	animal_type	breed	name	outcome_type	outcome_subtype	age_upon_outcome	age	color	date_of_birth
A716338	Dog	Chihuahua Shorthair Mix	Frank	Adoption		2 years	3	Brown/white	2013-11-
A733653	Cat	Siamese Mix	Kitty	Adoption		7 months	4	Seal Point	2016-01-
A691584	Dog	Labrador Retriever Mix	Luke	Return to Owner		2 years	5	Tan/white	2012-11-
A696004	Dog	Cardigan Welsh Corgi Mix	Lucy	Euthanasia	Rabies Risk	5 years	6	Sable/white	2018-01-
A673838	Dog	Pit Bull Mix	*Seth	Euthanasia	Aggressive	2 years	7	Black/white	2012-03-
A725717	Cat	Domestic Shorthair Mix		Transfer	SCRIP	1 year	2	Silver Tabby	2015-05-
A746874	Cat	Domestic Shorthair Mix		Transfer	SCRIP	3 years	1	Black/white	2014-04-
A736551	Dog	Labrador Retriever/Australian Cattle Dog	*Hia	Adoption		1 year	8	Black	2015-10-
A728214	Dog	Labrador Retriever Mix	Blessing	Adoption		3 years	9	Red/white	2013-02-
A664298	Cat	Domestic Shorthair Mix	*Taylor	Adoption		3 months	10	Tortie	2013-09-
A721199	Dog	Dachshund Wirehair Mix	Belle	Adoption		1 year	11	Tan/white	2015-02-
A664843	Dog	Pit Bull Mix	Sherlock	Transfer	Partner	1 year	12	Brown/white	2013-06-
A700408	Cat	Domestic Shorthair Mix	Nyla	Return to Owner		1 year	13	Brown Tabby/white	2014-04-
A742287	Dog	Boxer/Bulldog	*Kawhi	Adoption		2 years	14	Brown Brindle/white	2015-01-
A712638	Dog	Pit Bull Mix	Marcus	Transfer	Partner	3 years	15	Red/white	2012-09-
A723742	Dog	Miniature Schnauzer Mix	Gretchen	Adoption		5 years	16	Black/white	2011-04-
A658968	Dog	Pit Bull Mix	*Gigi	Adoption		6 months	17	Blue/white	2013-06-
A693288	Cat	Domestic Shorthair Mix		Adoption		2 months	18	Brown Tabby/white	2014-09-
A709511	Cat	Domestic Medium Hair Mix	*Toad	Adoption	Foster	4 months	19	White/Black	2015-07-
A615748	Dog	Labrador Retriever Mix	Blackie	Euthanasia	Suffering	5 years	22	Black	2009-12-

SNHU CS340 by Craig O'Loughlin (08/14/2022)

Version: 1.0

Date: 08/14/2022



## Development Process

Development of this project was done primarily using the Dash framework with the documentation available at <https://dash.plotly.com/>. We also used MongoDB and our [own CRUD module](#) to build and access a database of animals. Development started simple with one widget at a time and making sure that each feature was working correctly before moving to the next.

Any comments or additional info, or collab requests, find me at:

## Contact

Craig O'Loughlin

[craig.oloughlin@snhu.edu](mailto:craig.oloughlin@snhu.edu)

<https://github.com/oloughlinc>