Project Two: README file

Joshua Pardue

CS 340: Client/Server Development

Professor Jeff H. Sanford

August 13, 2022

# CS 340 README

## About the Project/Project Title

This software application utilizes the PyMongo driver to create CRUD functionality in order to access the Austin Animal Center Outcomes data set in MongoDB. The Python module, AnimalShelter, uses object-oriented programming methodology which ensures functionality to create, read, update, and delete.

Additionally, reusability is demonstrated in that the Python code is importable as a module by the Python test script. With proper user authentication, the database is able to interact with client-side code.

## Motivation

In an effort at locating premium search and rescue dogs, the purpose of this software application is to utilize existing data from animal shelters in order to pick out and classify the dogs for such a function.

## Getting Started

There are a few crucial steps to take initially such as setting up the database environment in Linux and then Mongodb. After accessing Linux and the proper directory, access to the Mongo environment must be established. Here, user authentication is necessary as the Python client method requires it. After importing the csv file (or whatever type of file), “aacuser” must be authenticated in order for proper function of CRUD capability. The C and R methods were created with functionality in mind and an end goal of input and return capability. The create input being the argument to function while the return being True if successful insert, else False. The read input function operated similarly with argument to function with key and value and the return dependent on a successful result, or Mongo to return an error message. The U an D methods were added as well with functionality in mind and utilized independently from the C and R methods as they utilize update and delete capabilities.

## Installation

This project utilizes and requires the following tools:

Mongodb can be downloaded following: <https://www.mongodb.com/try/download/community>

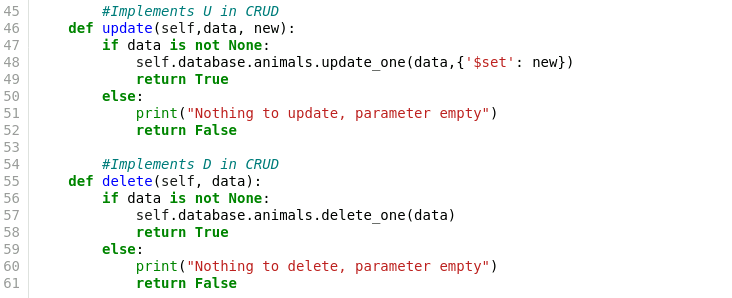
Python can be downloaded following: <https://www.python.org/downloads/>

Jupyter Notebook can be downloaded following: <https://www.jetbrains.com/dataspell/promo/?source=google&medium=cpc&campaign=17612794255&term=jupyter%20notebook%20editor&gclid=EAIaIQobChMIkY3exaCP-QIVKidMCh0K6w7QEAAYASAAEgK1f_D_BwE>

## Usage

The AnimalShelter class initializes the MongoClient in order to access the Austin Animal Center Outcomes data set in MongoDB. The CRUD methods are displayed below as the create function inserts a document into Mongodb database and collection. The create method inserts data into the animals collection while the read method queries documents from the same database and collection using the find () function and specified criteria. The update method updates one parameter in the animas collection and the delete method has the ability to remove it.

### *Text Description automatically generated*



### Code Example

The AnimalShelter class requires 2 libraries. PyMongo is a distribution containing tools to work with MongoDB. It is the recommended way to use MongoDB from Python. Bson.objectid is a unique identifier associated with the records of this database in Mongodb. A unique identifier in an \_id field is used as a primary key. This provides an automatic unique identifier in the form of an ObjectId data type.

**

Initialization is done through a client variable and authentication of username and password for the “accuser” when instantiating the class. The MongoDB URI format is used in creating a MongoClient to the running mongod instance.

**

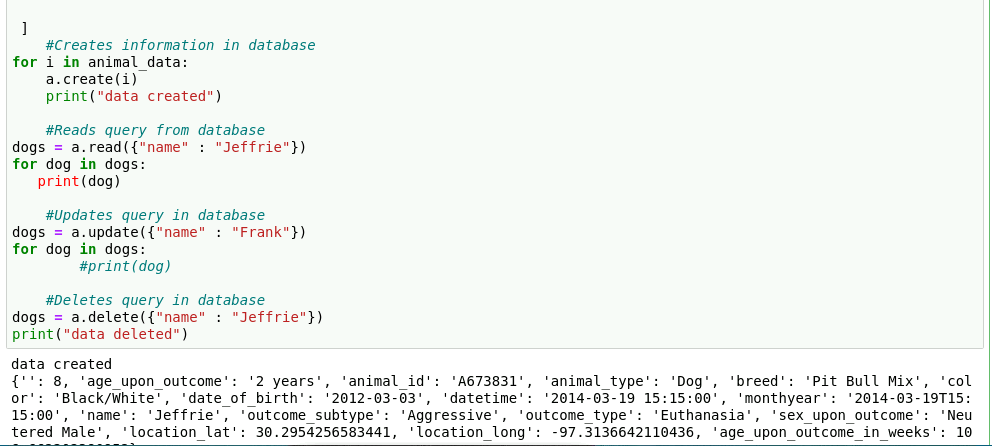
**Tests**

Verification of operation is done through the Testing.ipynd file loaded into Jupyter Notebook. This Python script imports CRUD Python module to call and test all instances of functionality.

The for loop creates the database information from animal\_data and the .read function finds the data in Mongodb verifying it was created. The update method updates the animal\_data while the delete method removes documents from the collection.

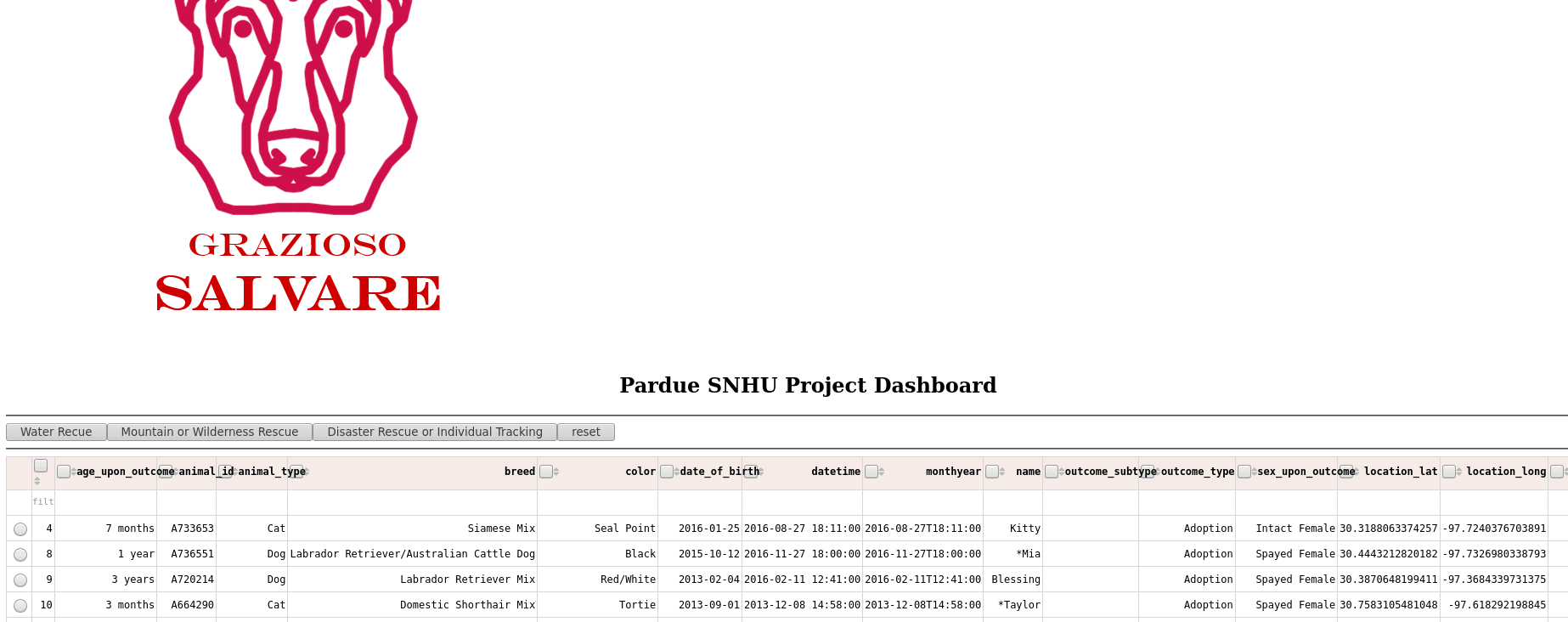
Text

Description automatically generated

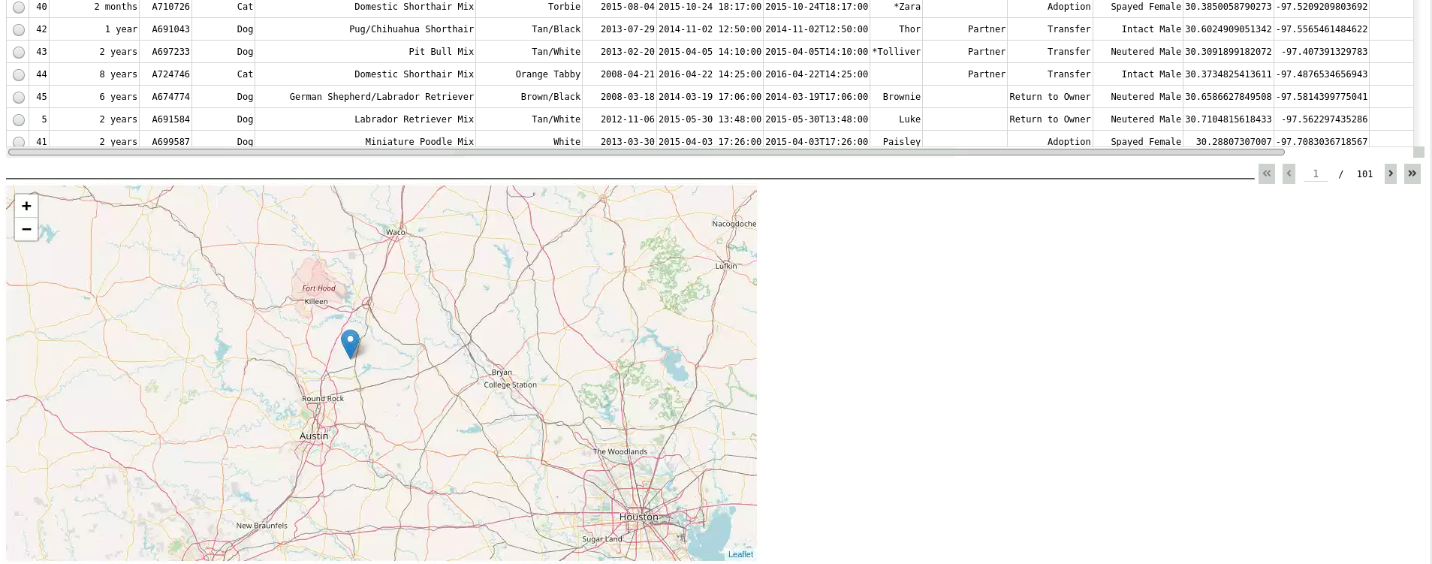


The dashboard components are also verified for operation and widget functionality.

The starting state of the dashboard with included widgets for interactive options to filter data, interactive data table, and charts. Aligning and determining whether or not the widgets and interactions between components and controller was very challenging and took time and planning to resolve. A lot of the issues were resolved through consistent testing and trail and error more than anything.



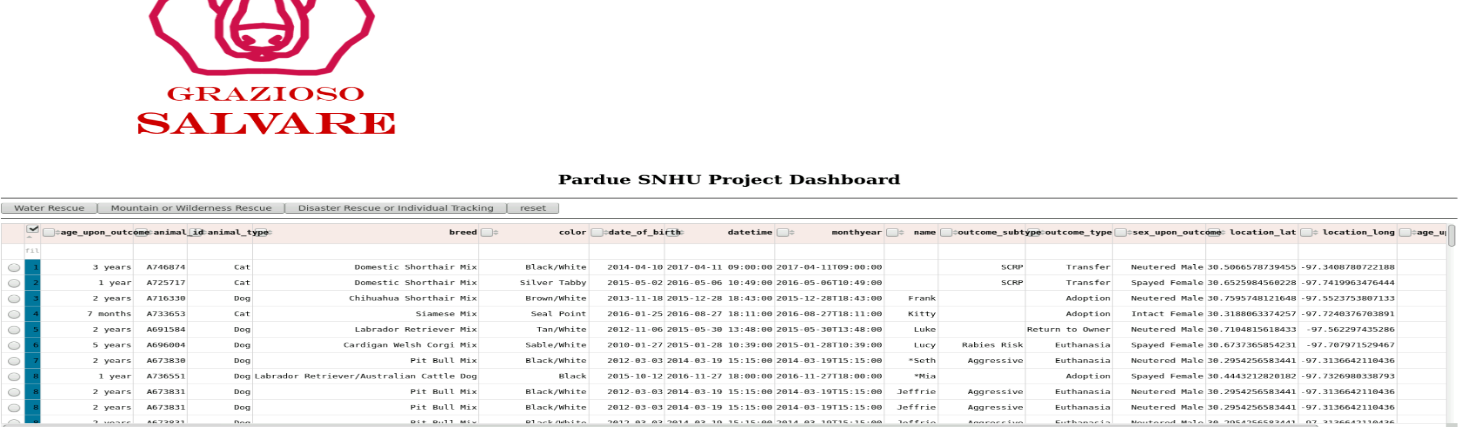
Starting state of dashboard full screen



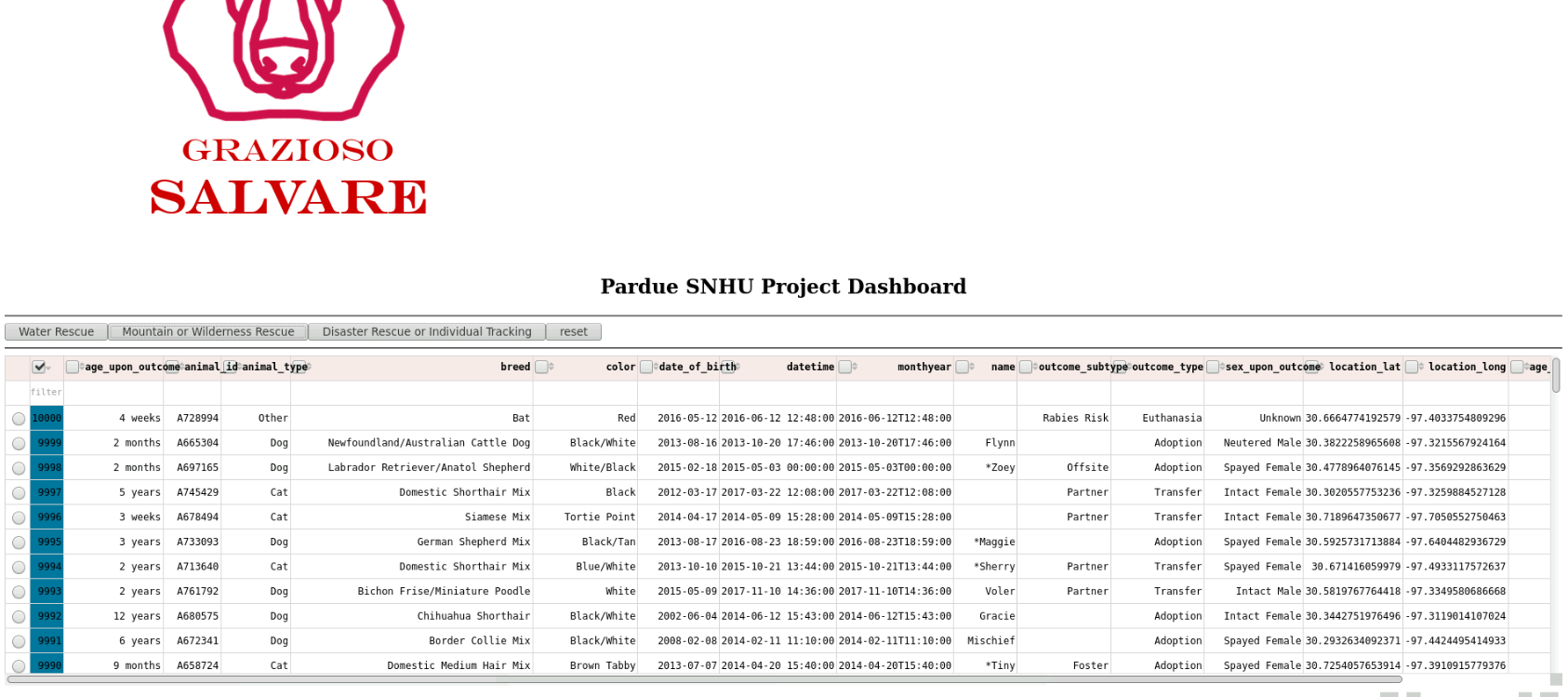
Geolocation chart

Executions of the dashboard, shows the widgets after each of the data filters has been applied

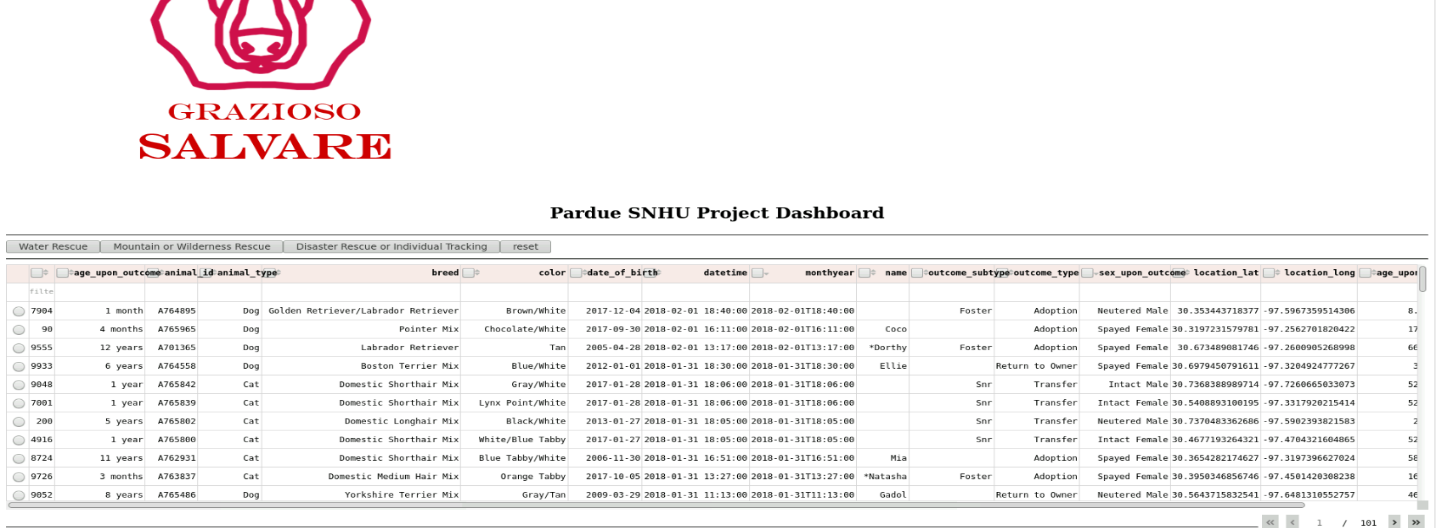
Water Rescue:



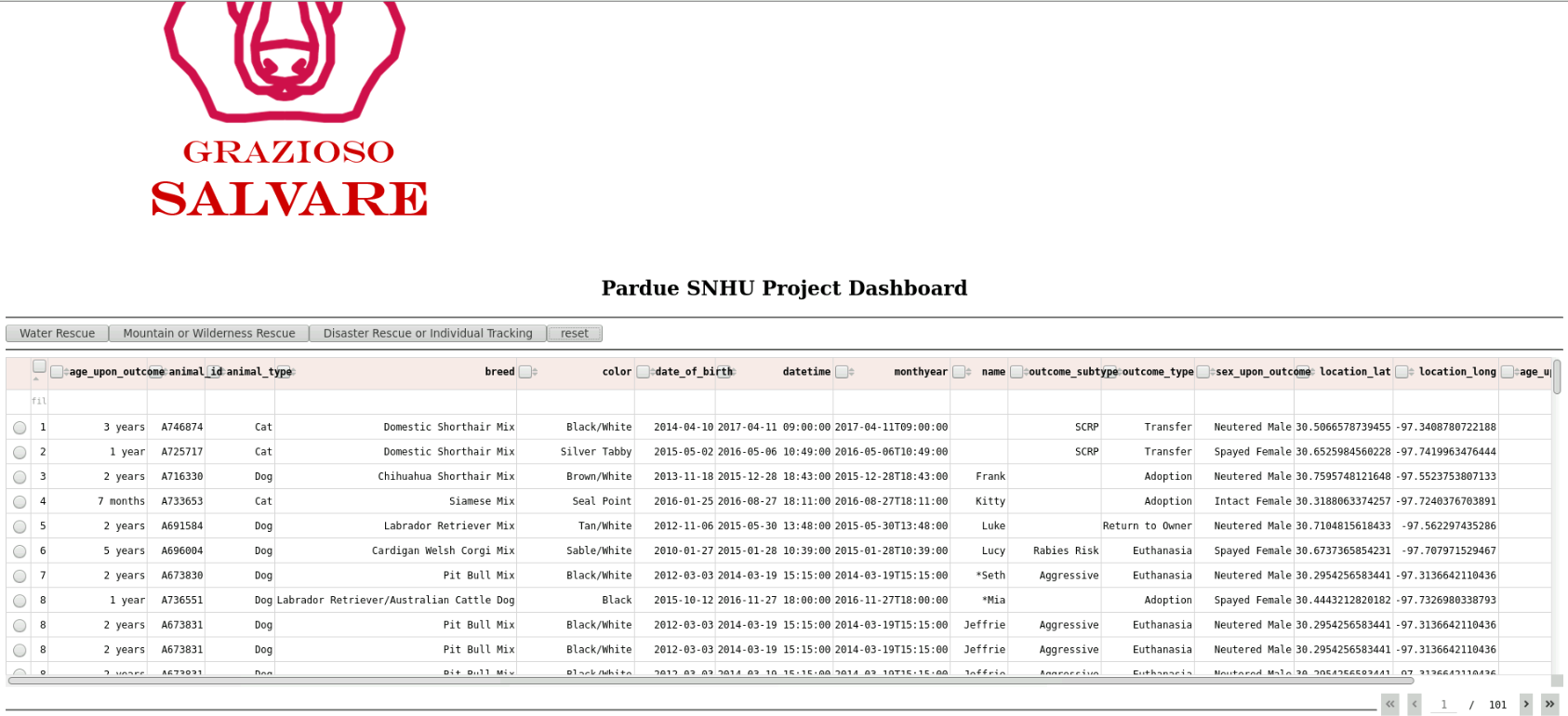
Mountain or Wilderness Rescue:



Disaster or Individual Tracking:



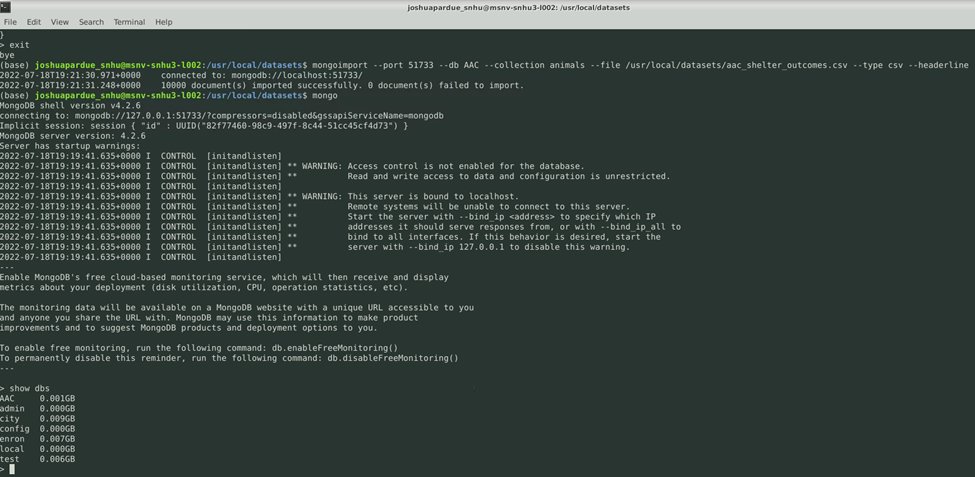
Reset (returns all widgets to their original, unfiltered state):

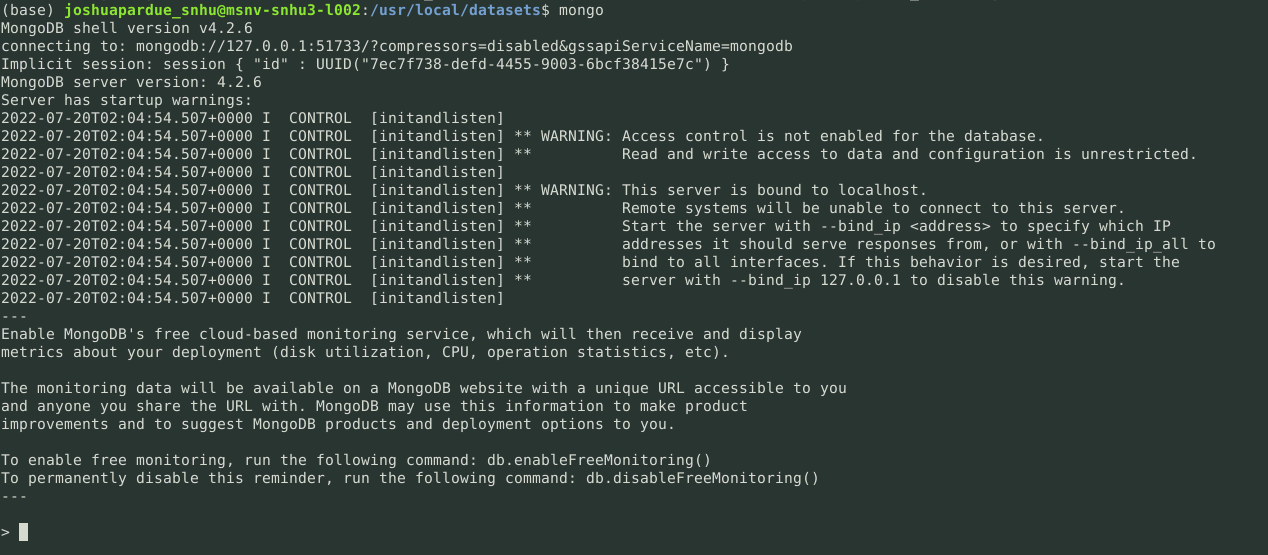


### Screenshots

Verify Mongo shell in running prior to running program.

Upload the Austin Animal Center Outcomes data set into MongoDB by inserting a CSV file using the MongoDB import tool.



**Contact**

Joshua Pardue (Joshua.Pardue@snhu.edu)