

ANIMAL RESCUE DATABASE

BRIAN BENTLEY – 4/16/2023

Brian Bentley

Grazioso Salvare  Rhode Island

Table of Contents

**CRUDModule1**

Getting Started2

Usage3

Installation………………………………………………………………………………………………………………………………….3

**Web Application6**

Getting Started6

Usage6

Development Log.............................................................................................................................7

Installation…………………………………………………………………………………………………………………………………8

CRUDMODULE

# Getting Started

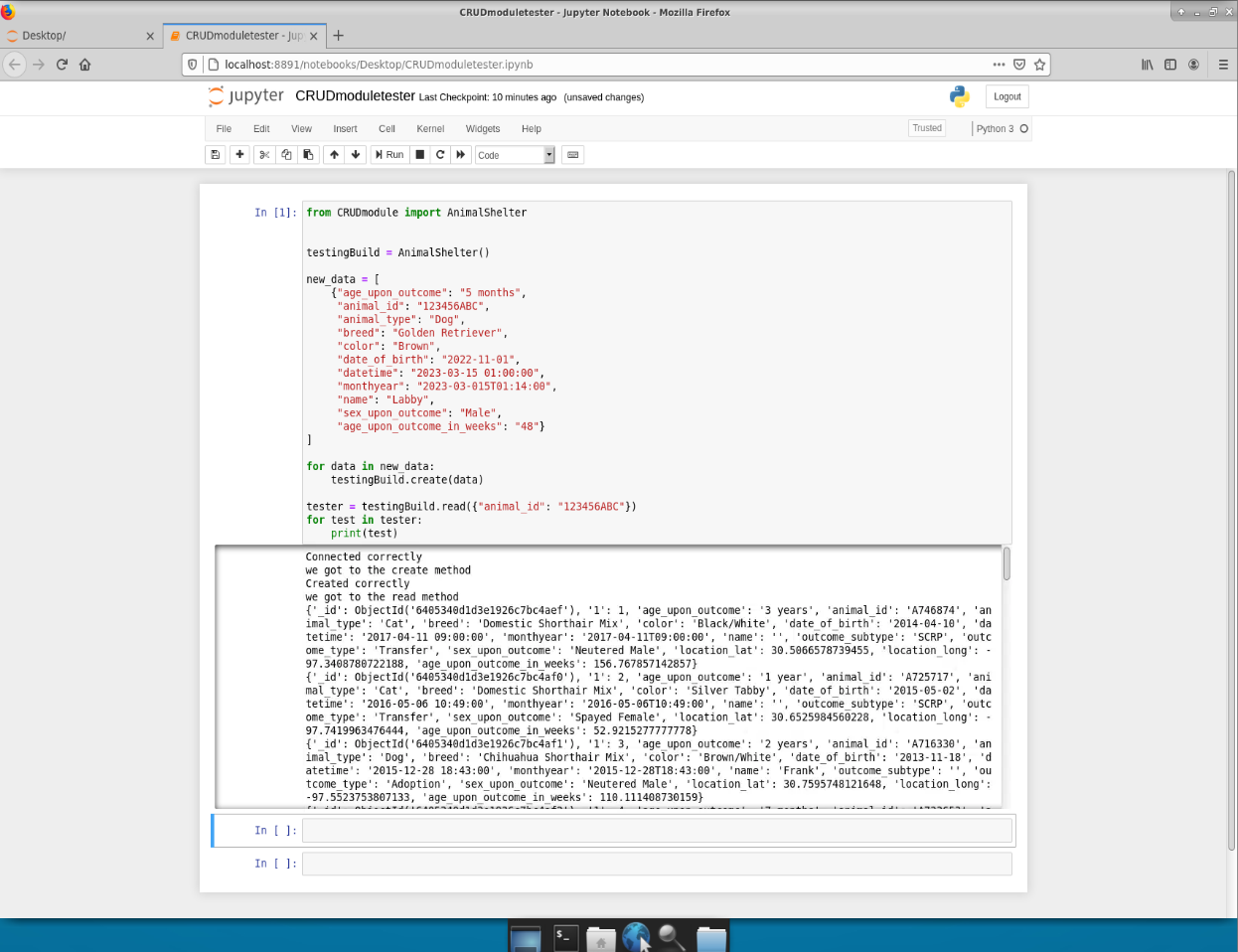
The purpose of this portable python module is to enable CRUD functionality for the data connection in the mongodb. This module will have create and read functionality at first however, later on it will be updated to contain an update and delete functionality. Again, later on we will have a user interface that interacts with our CRUD python module.

The motivate of this project is to ensure that there is an easy way to pass data back and forth to the mongo databases you are working with. This data will allow us to create, read, undo, and delete entry’s quickly and easily to our database without having to manually enter mongdb and insert the entries by hand.

We accomplish this by writing a python module that will securely login to mongo using existing credentials and a password and further utilizing the mongoDB api methods included to write and read data passed into this module through an external file.

# Usage

In order to use this module you must first setup your mongoDB user accounts and import or create a database to utilize. Ensure that the user you create had readWrite permissions for the database. Once that is completed you can simply insert this info when creating the database object by calling up the module and passing the username and password directly into the object.



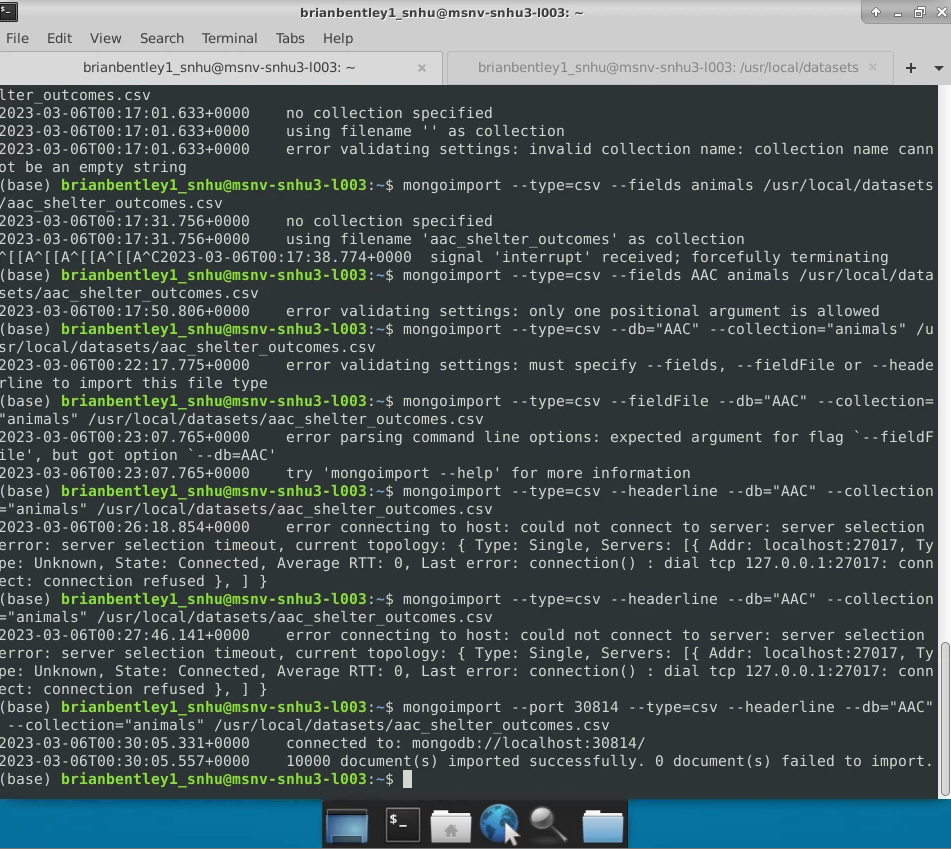
Here we have our module the “CRUDmodule” which we imported the AnimalShelter class from (will be renamed later). This class then is created and has data from the following dictionary passed through it to create a database entry. The read method then reads this information.

We have now also included an update and delete functionality to our software. The update function updates an entry in the database. While the delete function removes an entry in the currently selected database.

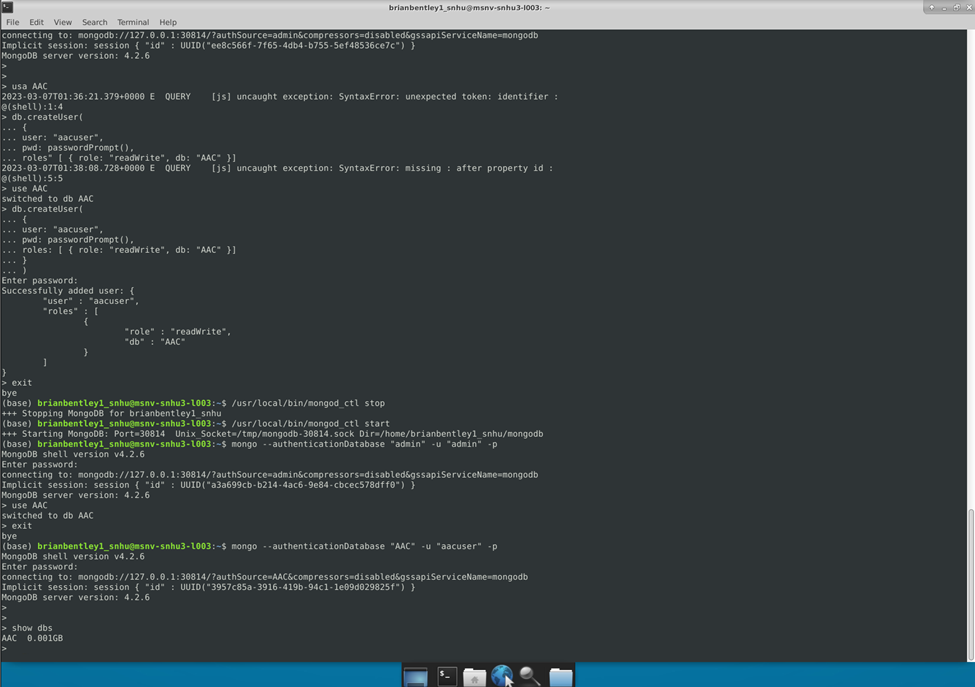
# Installation

To install this module for now simply copy it and paste it to your scripts folder. Then when writing scripts for you database you can simply import CRUDmodule or use a from import statement to just grab the specific class that you are looking for.

Image of how to load the database as a .csv file is below



Below is an image of how to create a user account and set the password in mongo

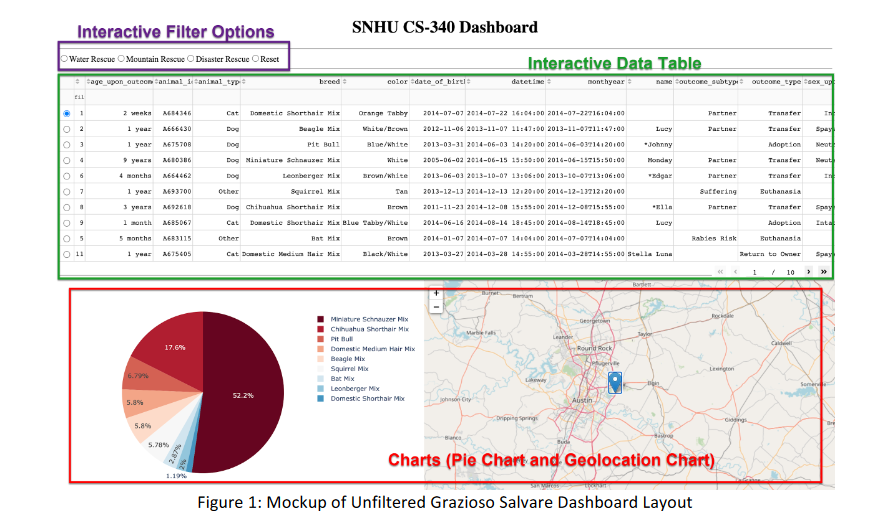


Later we will implement a way to !pip or download the CRUDmodule off of a linux repo packaged in an rpm for your ease. For now, enjoy!

Web Application Dashboard

# Getting Started

Welcome! This project has been commissioned by our customer Grazioso Salvare. The customer wanted a user based web application that could query a database in MongoDB. This database contains information on animals in the animal shelter in Austin. The customer also wanted to be able to filter for specific cases in the database and have this information populate a web page Like so,



The data that is filtered must be based on the four rescue types that Grazioso Salvare operates. Specifically, Water Rescue, Mountain or Wilderness Rescue, Disaster or Individual Tracking, and a reset function.

These rescue types also will filter out based on the preferred dog breed for those rescue situations

# Usage

Using this web application is fairly simple. The data chart will populate with the latest information from the Austin Animal Center locations with the dogs up for adoption being featured in the lower map. There will also be a pie chart showing the types of dog breeds and the percentage at those locations.

At the top of the page you will be able to pick a radial button that will reload the new database queries.

The database table will also be interactive, allowing users to filter and re-sort entries.

# Development Log

This took a week or so to develop in our off time. The project creation really consisted of researching the documentation available online for the different technologies that we used for this project. The different documentation necessary is as follows: MongoDB, Python, HTML, Jupyter, and Dash. It was overall an interesting project and will be further re-factored and delivered in the future.

# Installation

This application will be hosted online on a separate server soon. We will most likely be running this site on a native linux apache server at the Grazioso Salvare headquarters. The user for this application will not have to do anything besides ensure that the database is running and the web server is up.