# 

# 

## **Ilyass Hmamou**

## [**ilyass.hmamou@snhu.edu**](mailto:ilyass.hmamou@snhu.edu)

## **CS-340: Client/Server Development**

## **Project Two README**

## 

## 

## 

**About the Project/Project Title**

The “AAC database Dashboard” is a project that allows users to Interactively browse Animals Shelter Database using a Python script. The project provides great details about the database by using visualization tools, including an interactive datatable linked to a geolocation map, and a Pie chart that describes the population of the dataset.

**Motivation**

The purpose of this project is to enhance convenience when interacting with the “AAC” database by utilizing a Python script that interfaces with the Mongo server, enabling seamless database browsing and providing meaningful insights through graphs and charts.

**Getting Started**

**Step1:** Import Austin Animal Center (AAC) dataset into your MongoDB server  
**Step2:** Create a user in the server to allow authentication to the DB

**Step3:** Create a Python script that connects to the AAC database (using the credentials of the created user in Step2), then implement separate methods to allow all CRUD operations (Create, Read, Update and Delete)

**Step4:** Create the UI element of the project that uses the FindAll/ReadAll method created in step3. Your UI can include any type of graphs, maps, charts and interactive datatables.

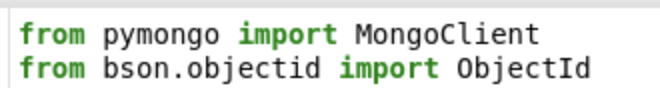
**Installation**

* Set up your MongoDb server where you will import the AAC dataset into
* Set up your Linux environment with Mongodb required commands (ex: mongosh) to be able to access your MongoDB server easily
* The main required tool is Jupyter notebook to implement the Python scripts in.
* Make sure to use mongopy, ObjectId and any other libraries needed to implement your UI

**Usage**

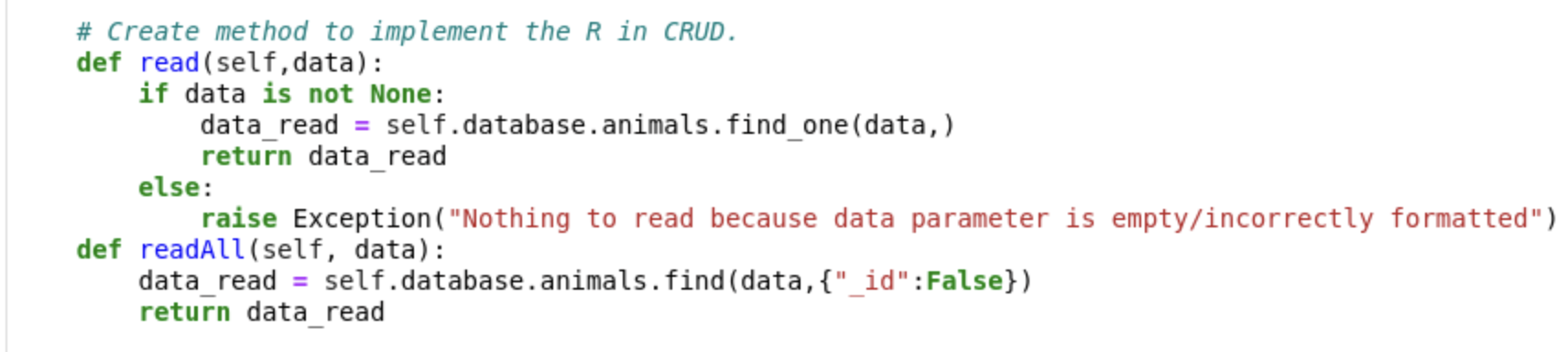
This project offers the capability to interact with the AAC database in a mongoDb server by creating, reading, updating and deleting data stored within the database.

**Code Example**

Import required libraries:   


Connect to the mongoDb Server: (Make sure to replace the connections variables with your own)  


Read/Search for document:



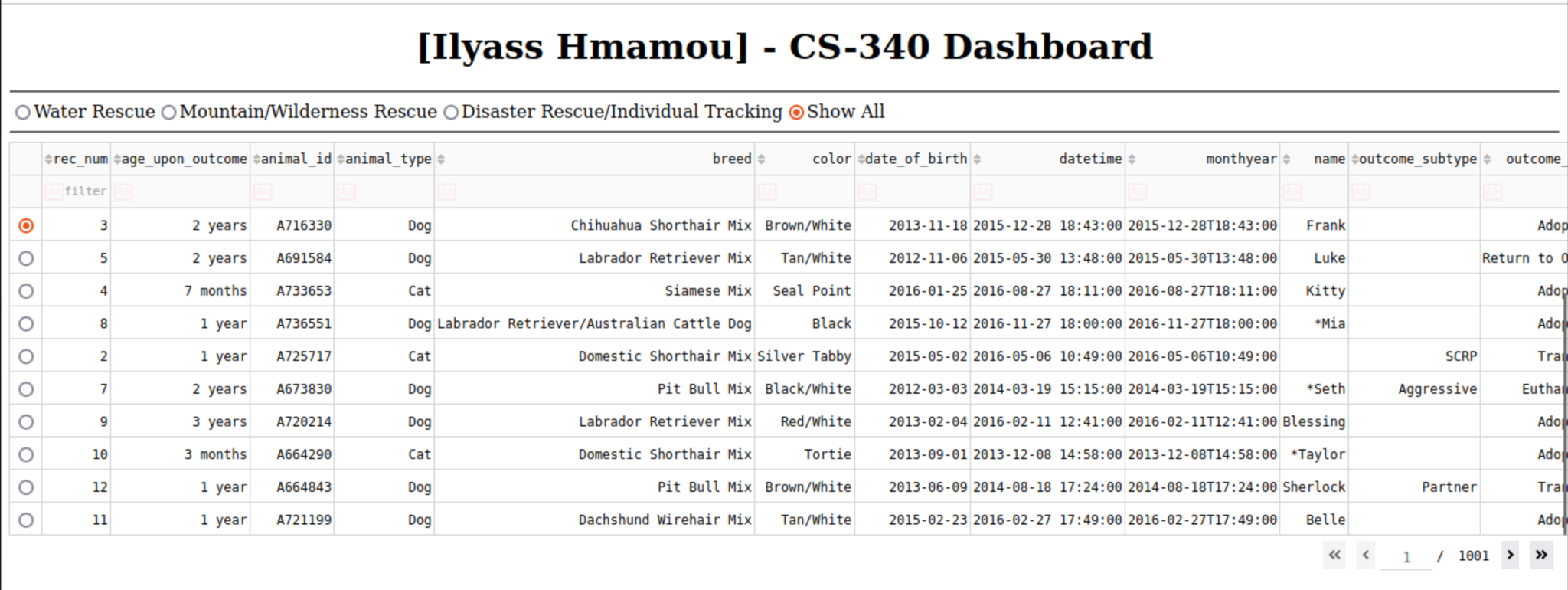
**Tests**

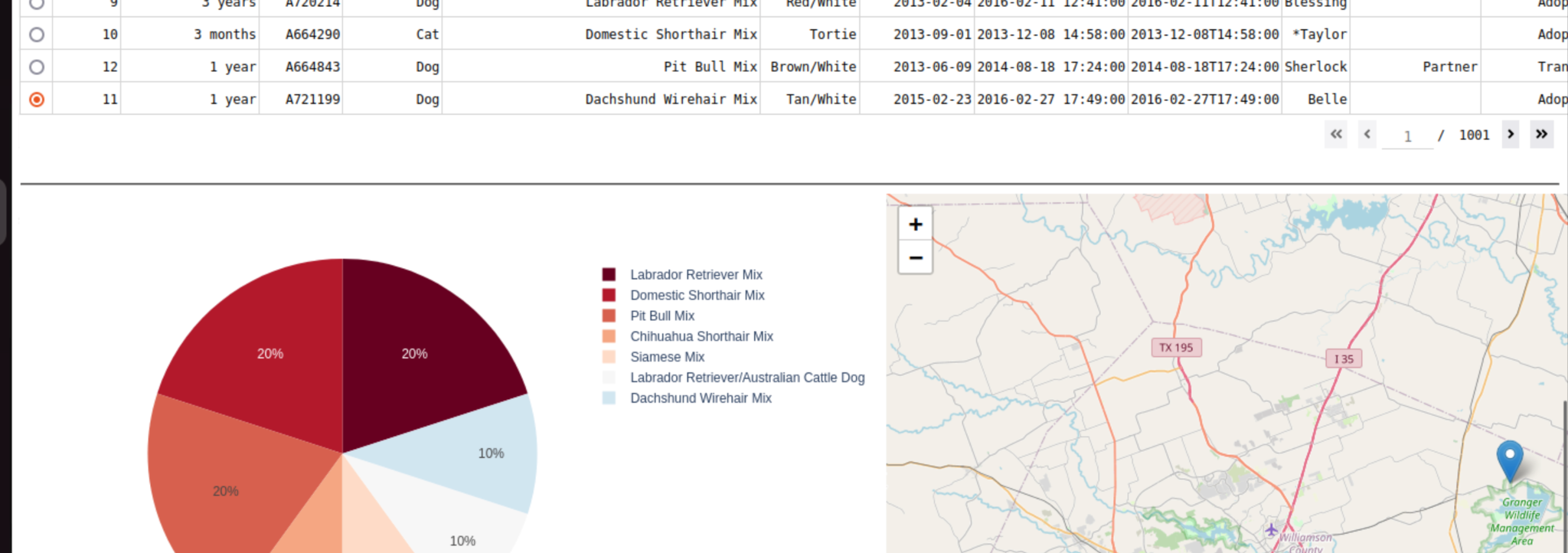
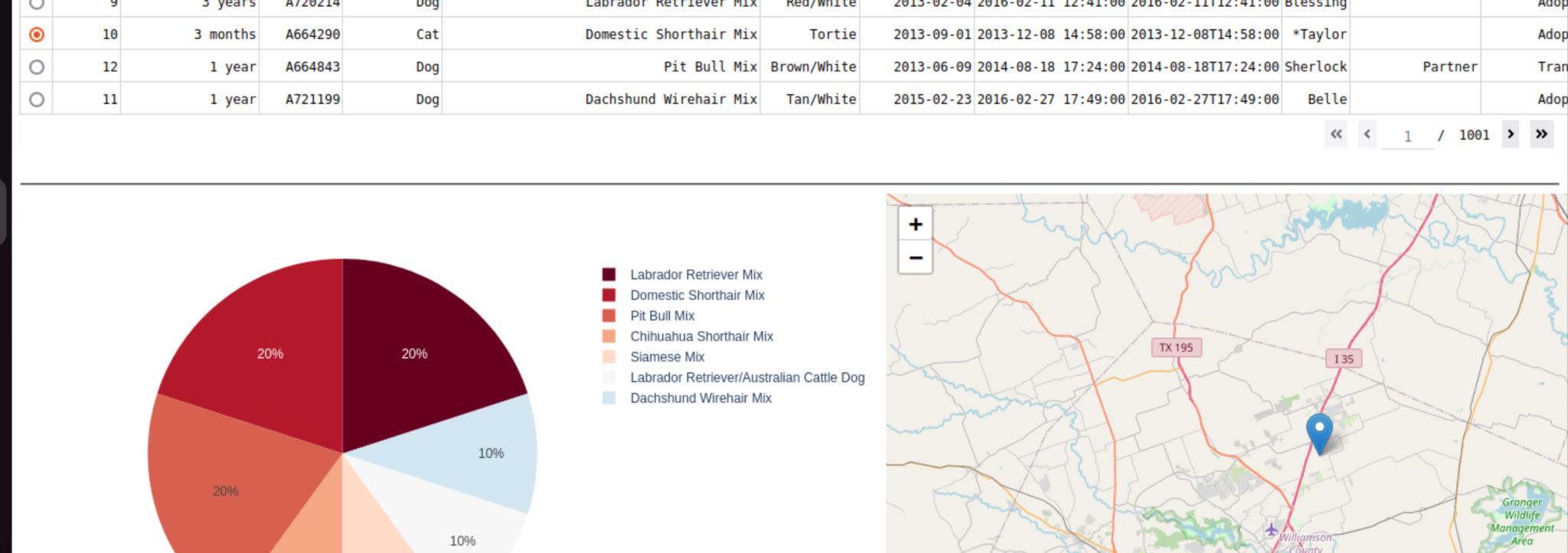
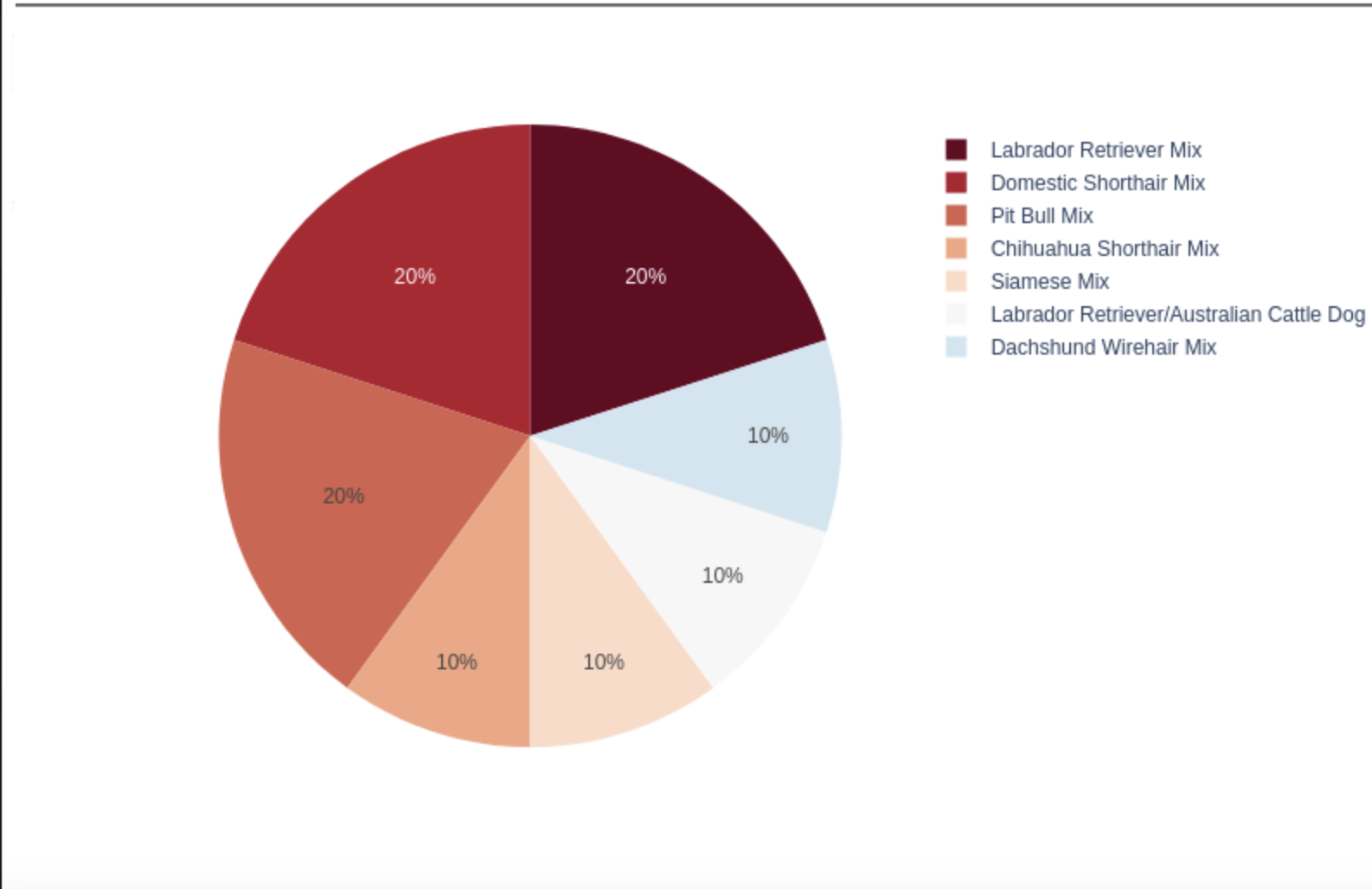
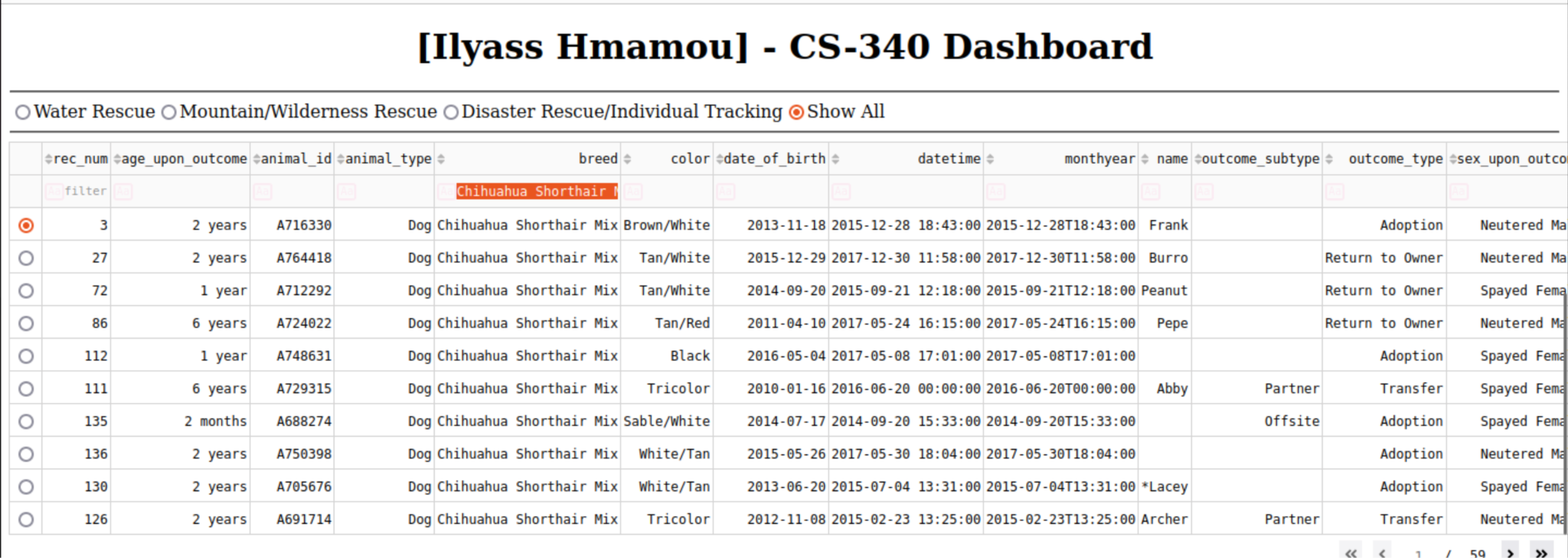
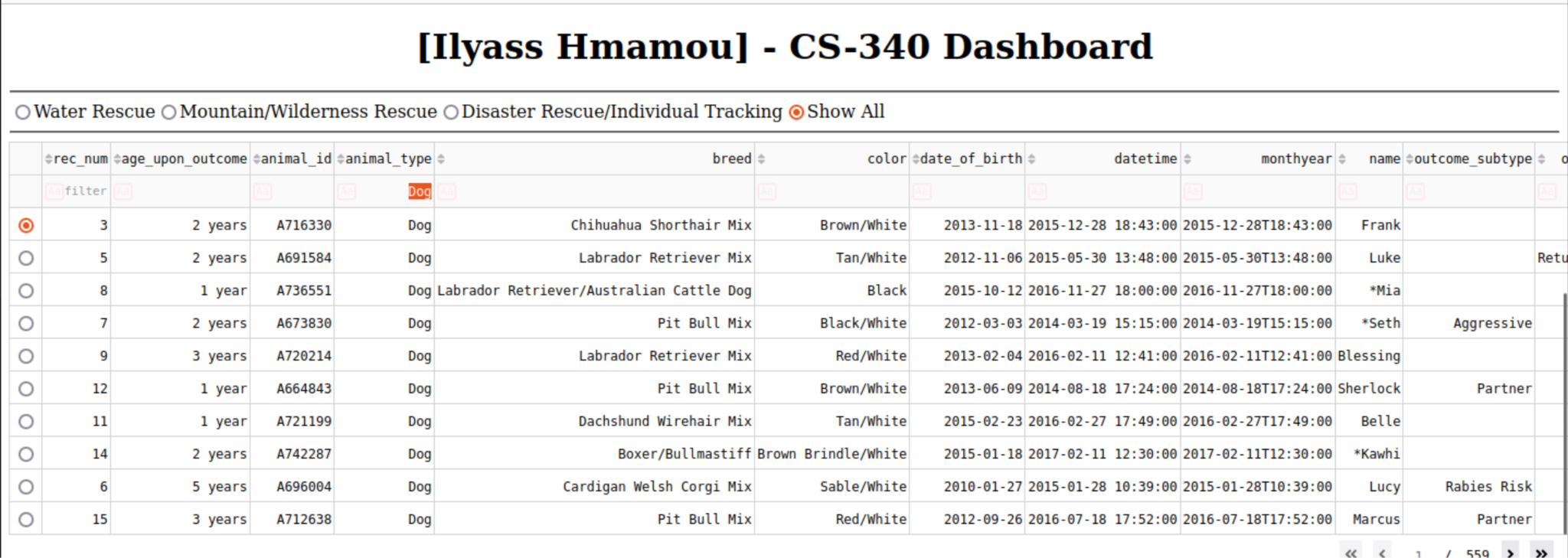
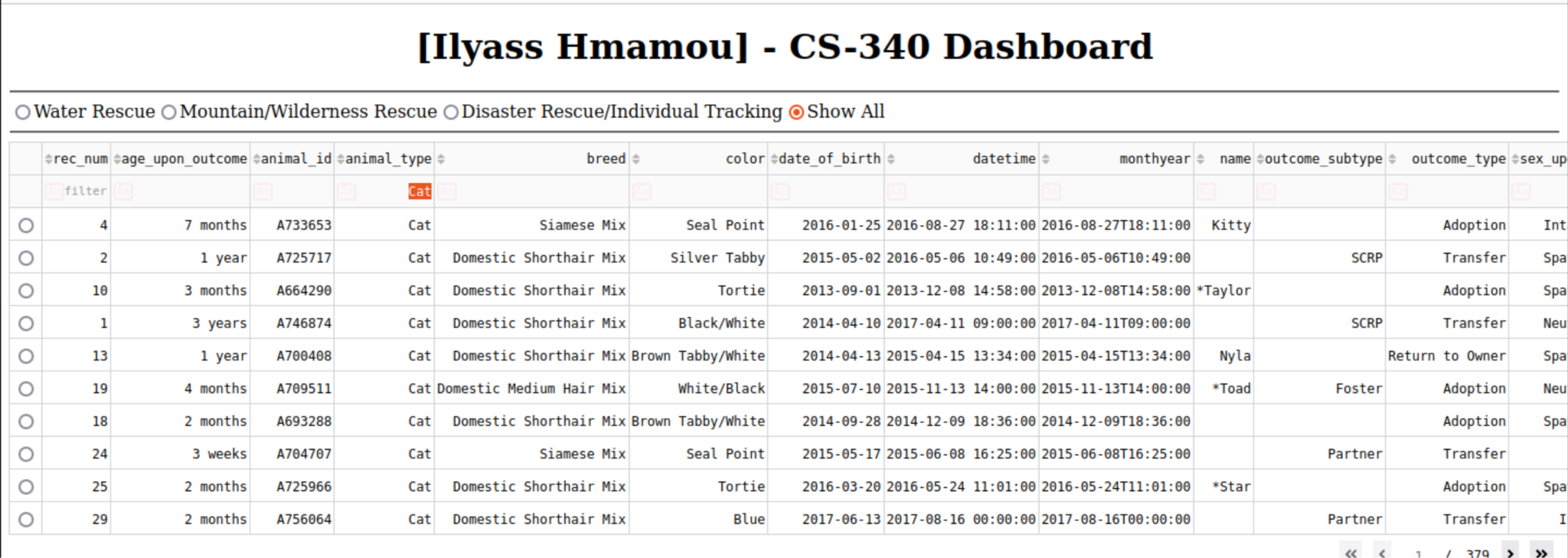
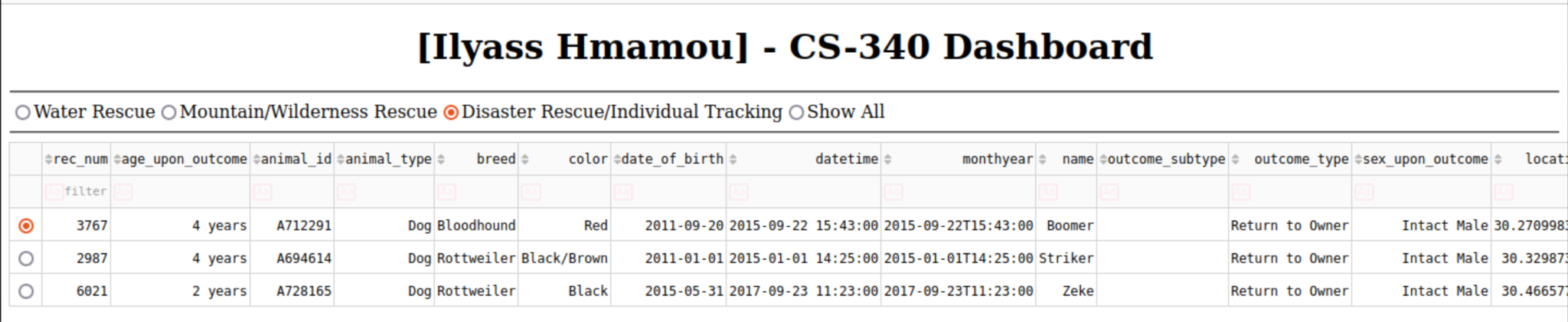
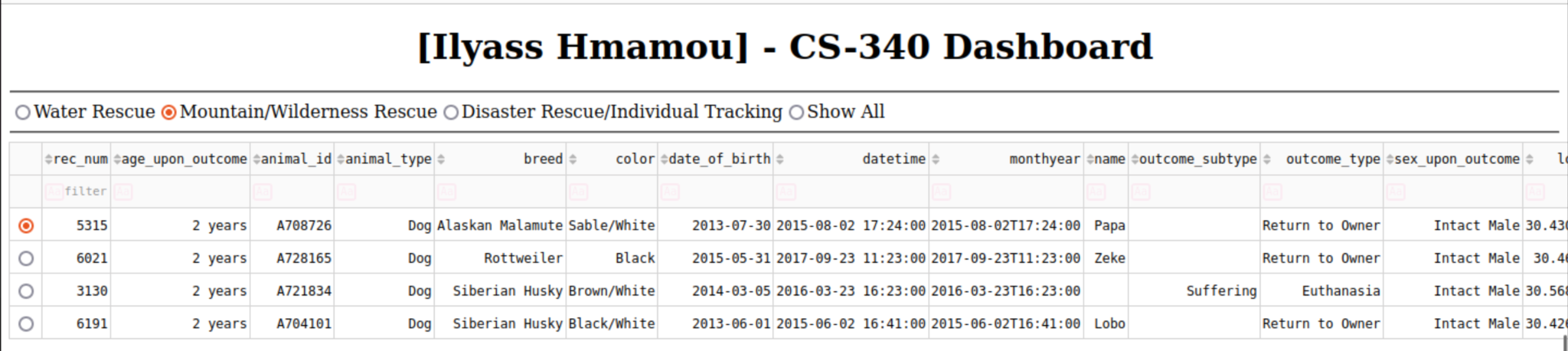
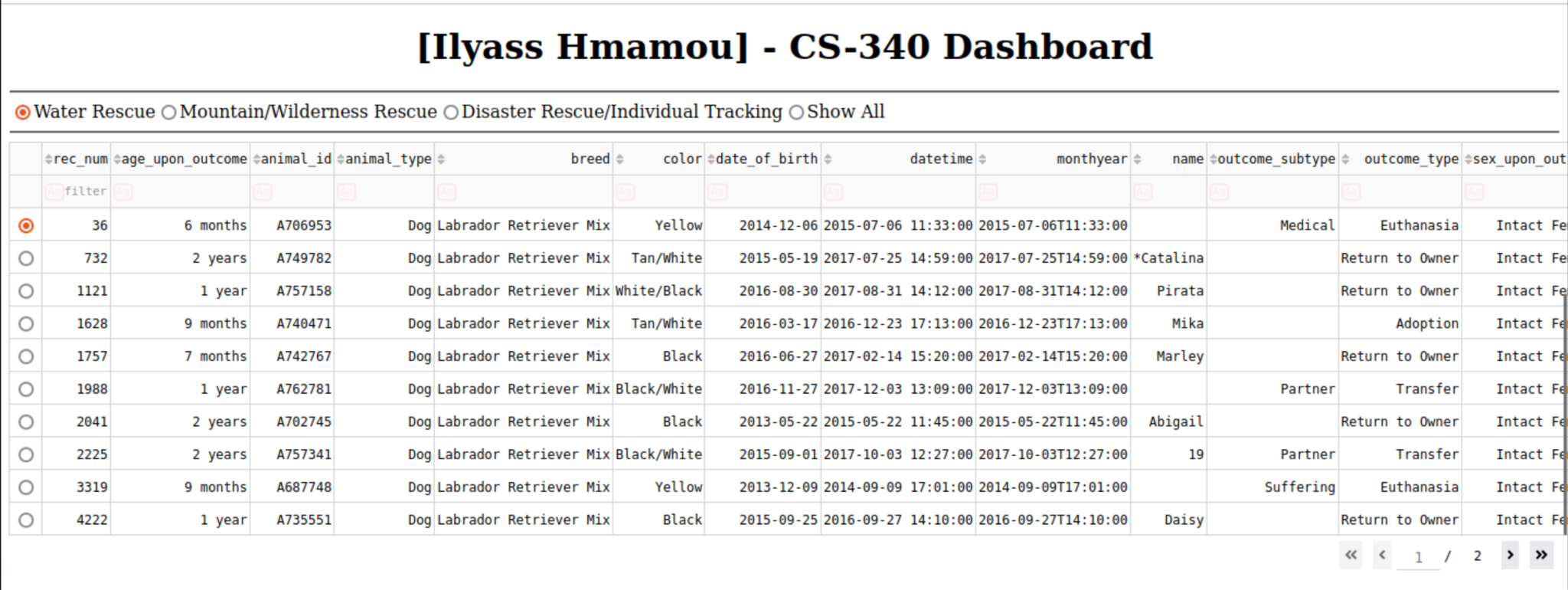
You can run the test in a Jupyter notebook by importing the Python function that was created. By doing this, you can attempt to generate sample data and also explore reading data from the database.

**Screenshots**

Screenshot of the project running







**Roadmap/Features (Optional)**

To allow the end users full access to the “AAC” dataset with better user experience, The recommendation here is to add a feature feature that allows AUTHORIZED USERS to update, or delete records in the dataset. The easiest way would be to make fields on the datatable editable and link it with the Delete and Update methods implemented on the CRUD project.

**Contact:** ilyass.hmamou@snhu.edu

**Your name:** Ilyass Hmamou