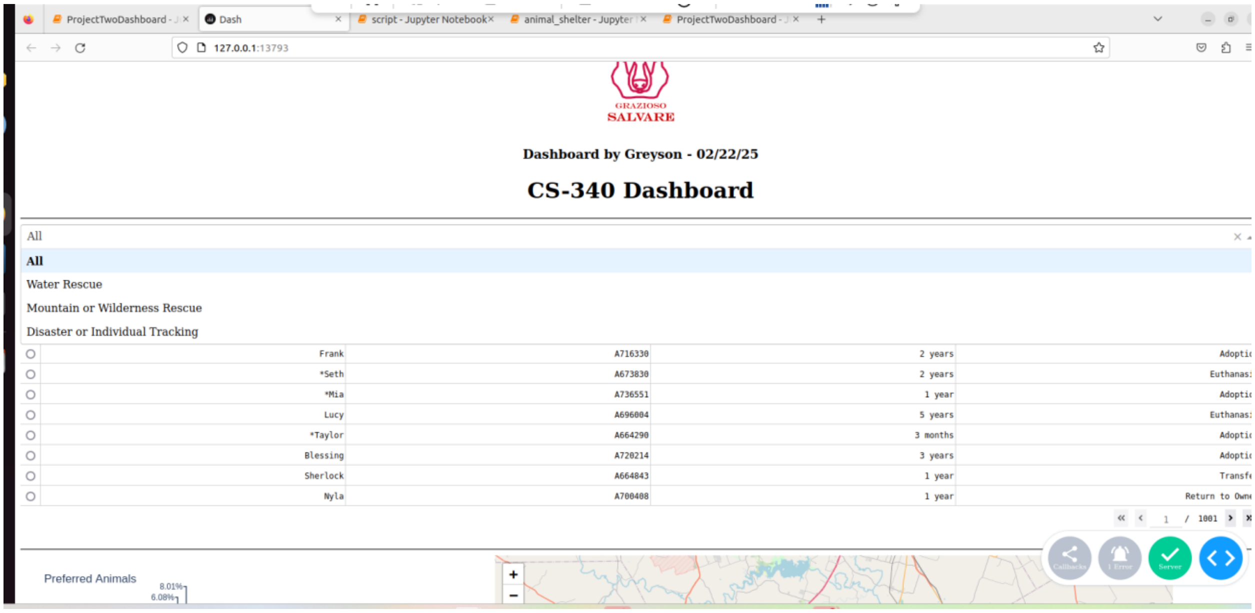
This project is to take a set of data and create a dashboard for the Austin Animal Center using data about the animals. The dashboard needed to be user friendly and show key information in this case it showed the name, ID, age, and most recent event. The first list that shows up is a complete list of all of the different animals from the file. Below the dashboard is a pie chart and an interactive map. The map shows the location of the animal when an animal is clicked on. This makes finding the last known location fairly simple as it is provided. The pie chart changes depending on the filter selected. The filter can be seen above the dashboard and when clicked on a drop down menu appears where the user can select water, mountain, or disaster tracking. This changes the dashboard and the animals that show up are based on the specifications document. The dashboard changes as well as the pie chart which displays the percentage based on the number of animals of a specific breed within the search parameters

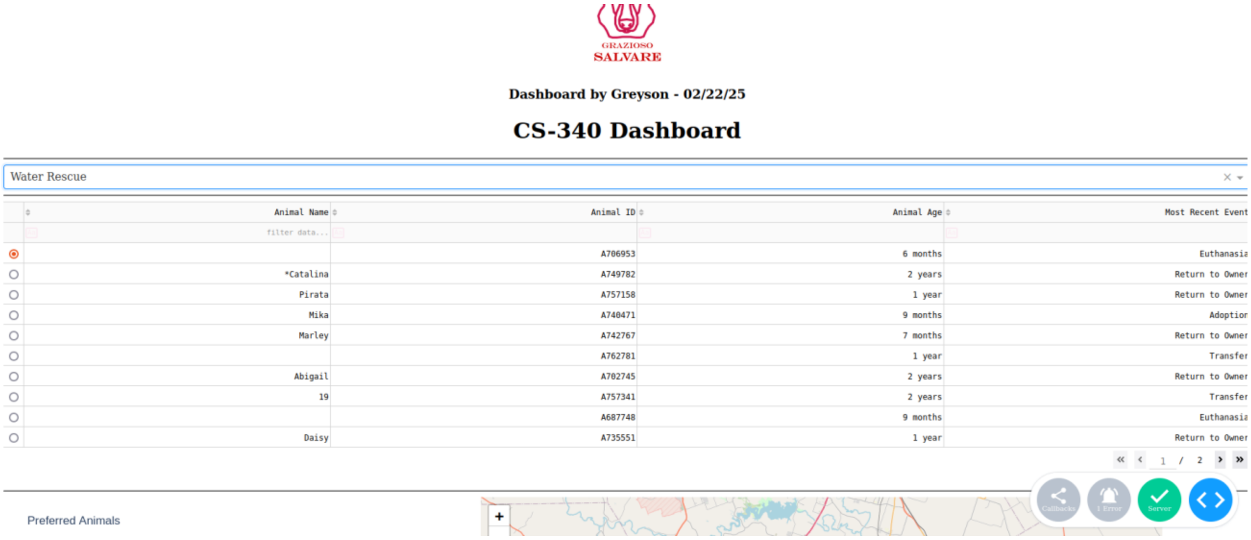


Dashboard by Greyson - 02/22/25

CS-340 Dashboard

All
All
Water Rescue
Mountain or Wilderness Rescue
Disaster or Individual Tracking

○	Frank	A716330	2 years	Adoptic
○	*Seth	A673830	2 years	Euthanasi
○	*Mia	A736551	1 year	Adoptic
○	Lucy	A696004	5 years	Euthanasi
○	*Taylor	A664290	3 months	Adoptic
○	Blessing	A720214	3 years	Adoptic
○	Sherlock	A664843	1 year	Transfe
○	Nyla	A700408	1 year	Return to Owne



Dashboard by Greyson - 02/22/25

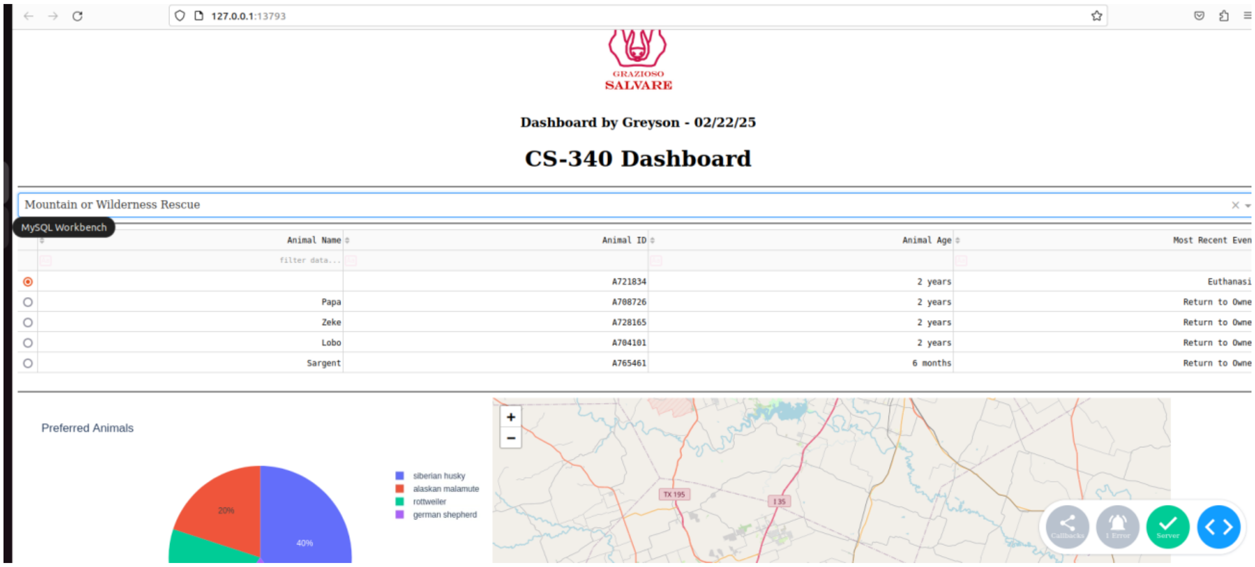
CS-340 Dashboard

Water Rescue

	Animal Name	Animal ID	Animal Age	Most Recent Event
	filter data...			
●		A706953	6 months	Euthanasia
○	*Catalina	A749782	2 years	Return to Owner
○	Pirata	A757158	1 year	Return to Owner
○	Mika	A740471	9 months	Adoption
○	Marley	A742767	7 months	Return to Owner
○		A762781	1 year	Transfer
○	Abigail	A702745	2 years	Return to Owner
○	19	A757341	2 years	Transfer
○		A687748	9 months	Euthanasia
○	Daisy	A735551	1 year	Return to Owner

Preferred Animals

-



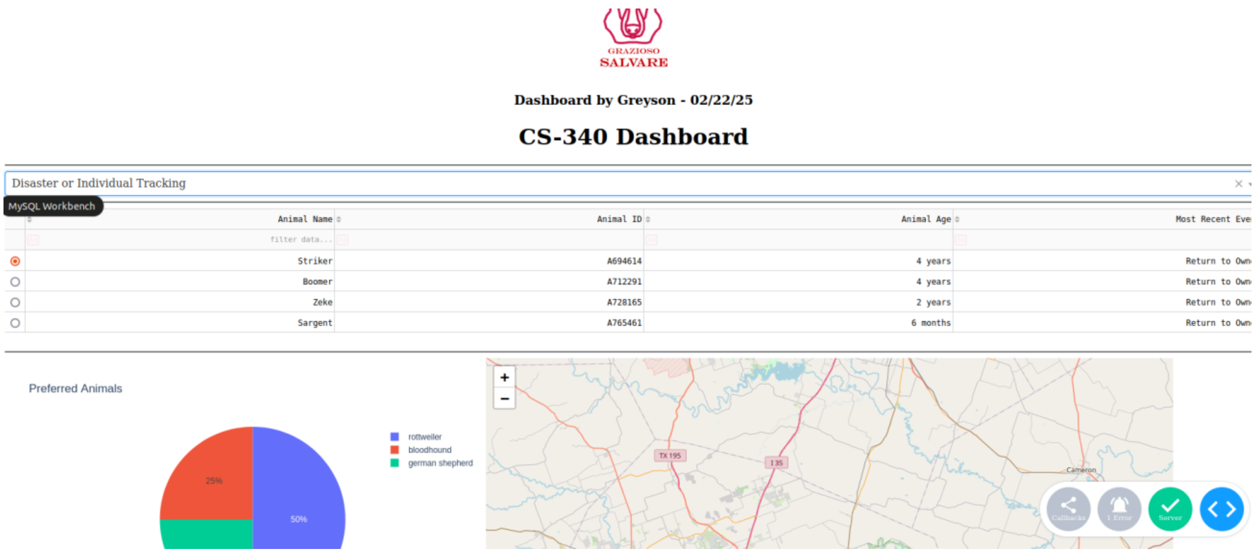
Dashboard by Greyson - 02/22/25

CS-340 Dashboard

Mountain or Wilderness Rescue

	Animal Name	Animal ID	Animal Age	Most Recent Even
	filter data...			
●		A721834	2 years	Euthanasi
○	Papa	A708726	2 years	Return to Owne
○	Zeke	A728165	2 years	Return to Owne
○	Lobo	A704101	2 years	Return to Owne
○	Sargent	A765461	6 months	Return to Owne

Preferred Animals


GRAZIOSO SALVARE

Dashboard by Greyson - 02/22/25

CS-340 Dashboard

Disaster or Individual Tracking

	Animal Name	Animal ID	Animal Age	Most Recent Eve
●	Striker	A694614	4 years	Return to Own
○	Boomer	A712291	4 years	Return to Own
○	Zeke	A728165	2 years	Return to Own
○	Sargent	A765461	6 months	Return to Own

MongoDB was picked because it is a NoSQL database that is flexible and allows for the handling of unstructured data. SQL would have thrown a fit and would have been harder to use for this use case (probably) I cannot say for sure as I have not done this with SQL to this extent. MongoDB can handle a lot of data really quickly and process it which makes it perfect for things like this where large queries need to be done. This is a relatively small query, but it is large for the scope of the project. Python was used because it provides an easy way to access MongoDB by using the pymongo library. This made python the natural choice for MongoDB. Within python there were a few libraries that I used such as pandas for data manipulation, numpy for data operations and numerical operations, and matplotlib for visualizations. Dash gets it's own little entry because of how much it was used within this project. Dash allowed for the interactivity displayed on the dashboard and was instrumental in connecting Python to the database and making it look nice and simple. I do not have a lot of skill regarding web development so this was extremely useful for me and I relied on it to help me make things look semi-professional.

The project setup was done by using an already created layout from module 6 and modifying it to fit the needs of the project. In this case the drop down menu should be on the top of the dashboard for ease of use. The data processing was also important but I had done a lot of that throughout the course. However, the specific data filtering was new but because of the general idea provided through the supporting materials section I knew what needed to be filtered. The map was already provided and the pie chart was mostly provided as well and did not require a huge amount of setup. Testing was a pain in the butt but I will discuss the challenges I had in the next paragraph.

The challenges, the biggest challenge I faced was getting a json error. This was because the data was not formatted in a way that json likes and this created issue after issue of me trying to find a way to fix this problem. The solution that I used was to convert and handle the date and time formats. It took a lot of trial and error to get to figuring it out. This was the biggest challenge and took a majority of my time. I solved it by using the tutoring services SNHU provide and talking with different tutors who were familiar with either Python or MongoDB. This was hard because I needed to take what the Python tutors said and the MongoDB tutors and combine it.

https://dash.plotly.com/dash-core-components

https://pandas.pydata.org/docs/reference/api/pandas.tseries.offsets.Week.html#pandas.tseries.offsets.Week