* Describe the required functionality of the project. Include the screenshots or screencast taken while testing and deploying your dashboard (Step 6) as proof that you have achieved the required functionality.

The functionality of this project is to read a file and format the contents into a data table, maps, and charts for easier analysis. This dashboard can filter between water, mountain/wilderness, and disaster rescue. It also has the ability to reset itself. The map can show the location of the animal, while the chart can show the number of a specific animal.

Startup

A screenshot of a computer

Description automatically generated

Water Rescue

A screenshot of a computer

Description automatically generated

Mountain/Wilderness Rescue

A screenshot of a map

Description automatically generated

Disaster Rescue

A screenshot of a map

Description automatically generated

Reset

A screenshot of a computer

Description automatically generated

* Describe the tools used to achieve this functionality and a rationale for why these tools were used.
  + Be sure to explain why MongoDB was used as the model component of the development, including what specific qualities or capabilities it provides for interfacing with Python.
  + Be sure to explain the Dash framework that provides the view and controller structure for the web application.
  + Be sure to include links to any resources or software applications that were accessed or used.

MongoDB, Python, and Jupyter Notebook were used to achieve the functionality in the project. MongoDB is an easier database to learn, paired with Python, one of the easiest languages due to its massive libraries. Additionally, MongoDB can read Excel-type files, which is what the data contains. The Dash framework allows for an easy user interface compared to a CLI. It can filter the housed data to only see what the user is looking for. Then, it can convert the filtered data into widgets like a map or chart.

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

* Explain the steps that were taken to complete the project.

The first step was to create a CRUD file. A create, remove, update, and delete method was made inside a PY file. Then the Jupyter Notebook would securely access the database with CRUD abilities. Jupyter would then format the database into easy-to-read formats for the user to see. This was done by manipulating the data to project specifications. Each step needed to be completed before the next one could begin.

* Identify any challenges that were encountered and explain how those challenges were overcome.

I ran into many challenges throughout the project. I had a feeling this would be the case before starting the project. So, I made sure to start early in the week and give myself plenty of time to finish. There were many errors, and I often spent hours on the wrong fix, even thinking that the Jupyter was outdated, and that was the broad problem. Through trial and error, I found fixes that made the program work as intended.