## About the Project/Project Title

*This software is about sorting and collecting data. It also allows to use information that has been stored about different animals in a database. The user can modify, add, or remove animals if they wish to do so.*

## Motivation

*The motivation of this project is to a reliable and easy way to keep track of animals that are currently in the animal shelter. The user can add animals, modify, or remove electronically which makes it more convenient.*

## Getting Started

* Connect to host
* Authenticate yourself
* Select the database
* Import the files
* Use commands to search documents

## Installation

*List the tools you need to use the software and how to install them.*

*First of all you’ll need an internet connection. The tools you need is Junyper Notebook, Python for command line and MongoDB. Once MongoDB is running just use the commands to traverse through the database.*

## Usage

*Use this space to show useful examples of how your project works and how it can be used. Be sure to include examples of your code, tests, and screenshots.*

### Code Example

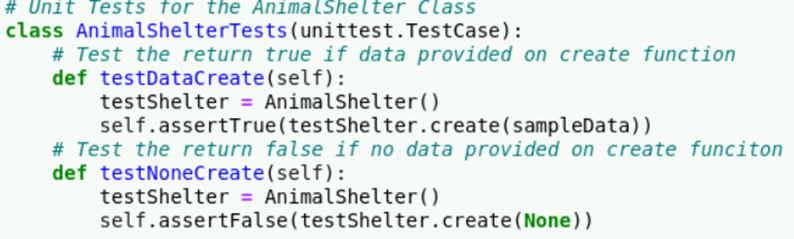
*testData():*

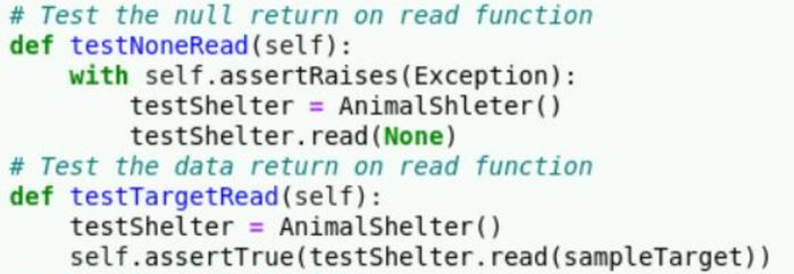
*testShelter=animalShelter()*

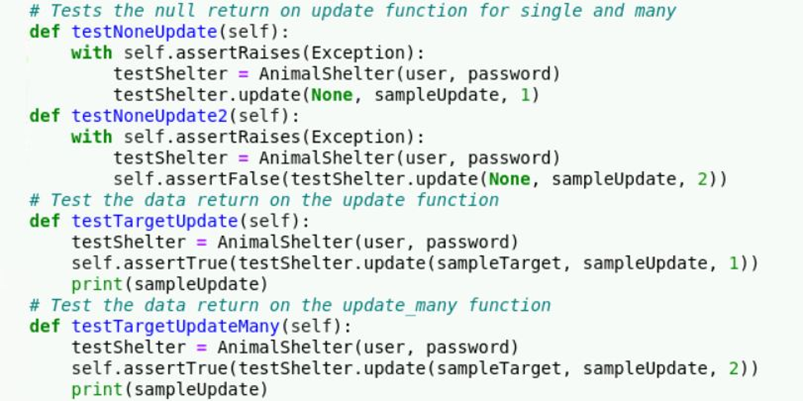
*assertTrue(testShelter.create(sampleDate))*

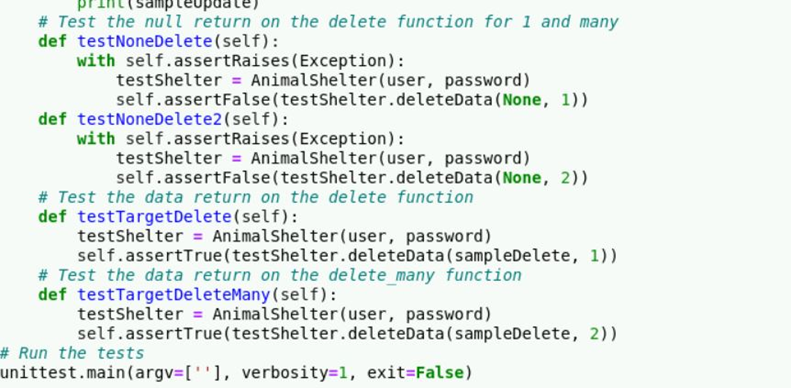
### Tests

*Describe and show how to run the tests with code examples.*

**

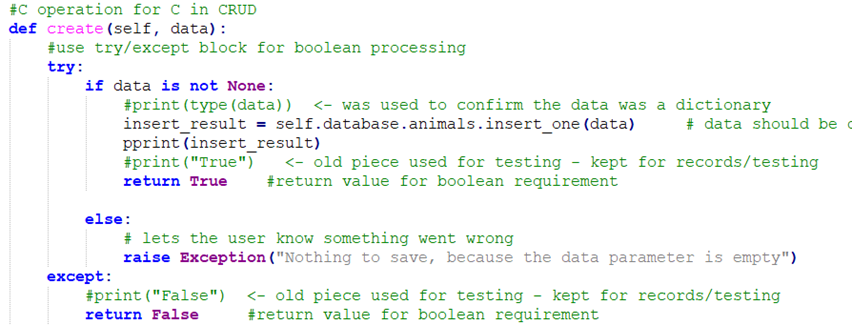
**

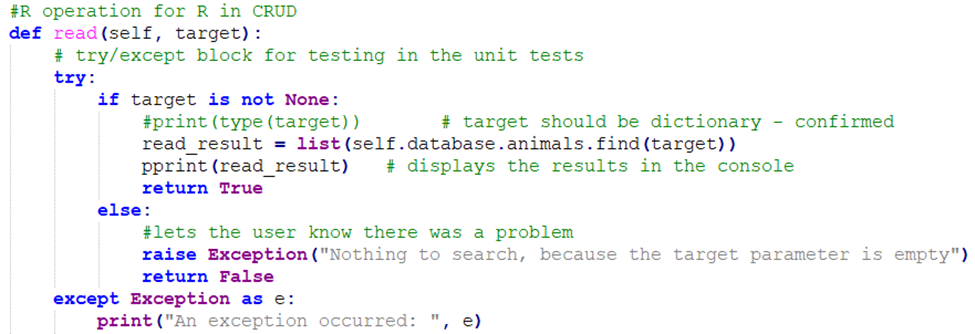
**

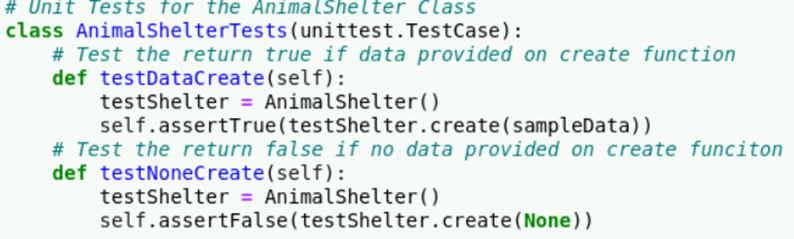
**

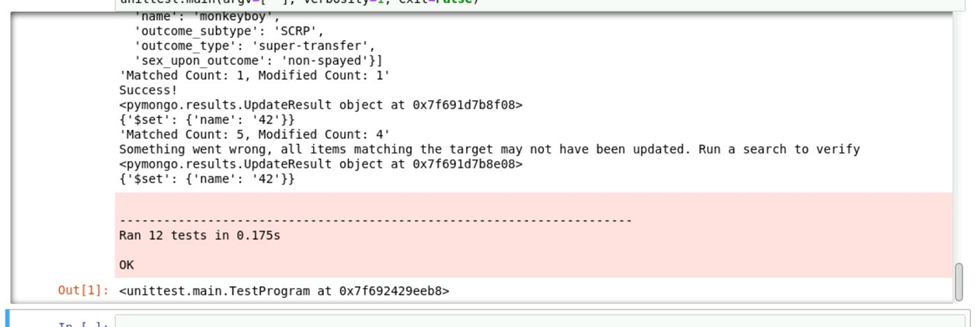
### Screenshots

*Provide screenshots that demonstrate your work.*

**

**

**

**

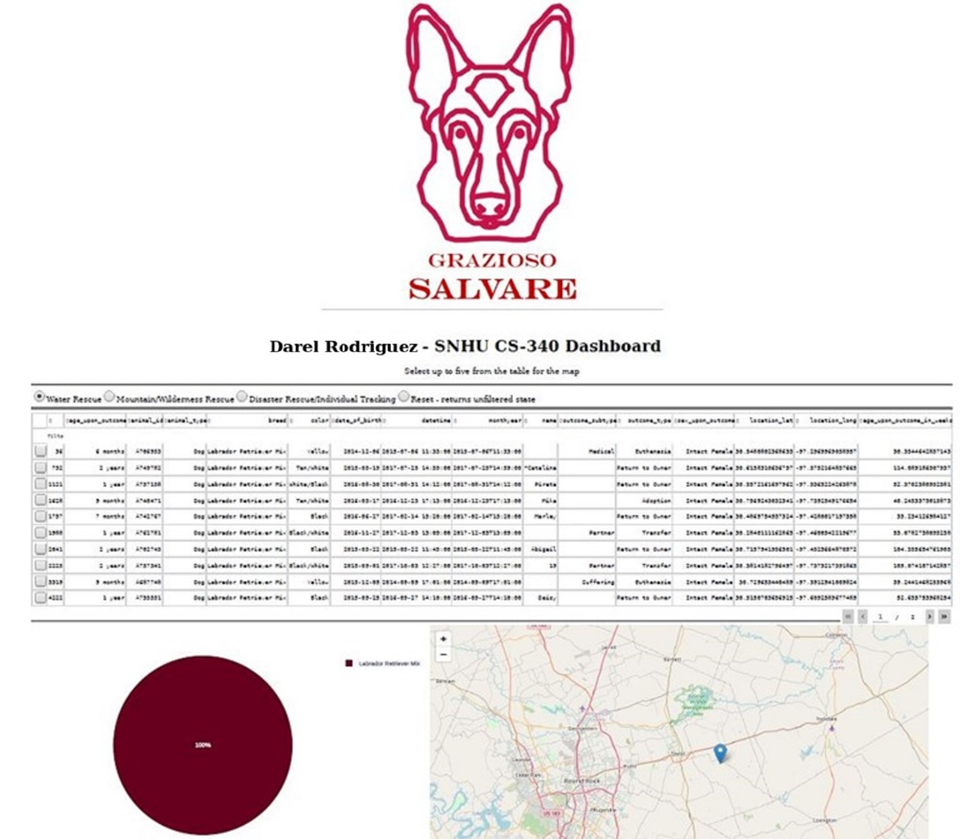
The dashboard features include:

* A table with the pets information
* A pie chart that display the breed of the pets.
* Finally, a map displaying the location of the pet.

The motivation to use MongoDB because it provides a quick setup for the database from a document and it also provides a python friendly interface.

The reason dash was used is because is due to its dynamic nature. It provides a responsive framework and involves html dash tags that control output segments.

* Water Rescue



* Mountain or Wilderness Rescue

A picture containing chart

Description automatically generated

* Disaster or Individual Tracking

A picture containing diagram

Description automatically generated

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

A picture containing diagram

Description automatically generated

The challenges that I encountered in this project is getting the table to display correctly. What I did to resolve this problem is do research about how to display a table on python.