# CS 340 README Template

*Use this template to complete your README file. When completing the template, keep the headings as they are so that your document has a clear organization. Remove the italicized prompt text after you have completed each section for a polished final document.*

## About the Project/Project Title

*With access to the dashboard, authorized users can access the Austin Animal Shelter database in a single dashboard. The dashboard allows them to filter to specific queries for different types of animal rescue based on breed, sex, and age in weeks. It also allows them to search animals that do not fit the criteria of the included queries with filters.*

## Motivation

*This project was created to assist the Grazioso Salvare employees in finding the best animals for their animal rescue programs by querying the databases of five different animal shelters in Austin, Texas.*

## Getting Started

*Follow these steps to get started:*

* *Enter MongoDB using a terminal*
* *Import the aac\_shelter\_outcomes csv file using MongoDB commands*
* *Sign in with the aacuser credentials*
* *Ensure Python is installed and up to date*
* *Run the code from the Dashboard Jupyter Notebook*
* *Add applicable filters (widgets are assigned for quick filtering)*

## Installation

*The following programs must be present on your machine to use the dashboard:*

* *Up to date version of MongoDB*
* *Up to date version of Python*
* *Up to date version of Jupyter Notebooks*
* *Included animal\_shelter.py file*
* *Included Dashboard Jupyter Notebook*

## Usage

### Code Example

To import the data, you must use the “mongoimport” function in the terminal window. Normal steps should be taken as if signing in with username, password, port, and host ids. Next, you define the database, collection, if there is a header line, and the type of file. In the included screenshot, the file was a comma separated value (CSV) file:

A screenshot of a computer screen

Description automatically generated

To check your connection to your account, you can use “db.runCommand({connectionStatus:1})”. This will return the authentication information along with any roles that the account has access to:

A screenshot of a computer program

Description automatically generated

**Create**

*The create function requires a dictionary entry of all data points that need to be added. You can create your own animal data using the create() function defined below:*

*A computer code with text

AI-generated content may be incorrect.*

**Read**

*When reading data from the database, use the animals.read() function. This function requires a query to find specific data. See the defined function below:*

A computer code with text

AI-generated content may be incorrect.

**Update**

*You can also update any data that may have changed or has errors using the animals.update() function. You first define your updateKey, which is the data you are looking for, then your updateValue, which is the updated data:*

**A computer screen shot of a code

AI-generated content may be incorrect.**

**Delete**

*Data can be deleted using the animals.delete() function. All data that matches the query will be deleted, so be sure to use a unique identifier or be certain you want to delete all data that matches:*

**A computer screen shot of a error

AI-generated content may be incorrect.**

### Tests

*When creating data, the animals.create() function is used. It is best to use predefined data for ease of reuse. In the screenshots below, animal\_data has been defined for a cat named Leah:*

A screenshot of a computer

Description automatically generated

*This data can then be added to the database using the animals.create() function mentioned above:*

A close-up of a white box

Description automatically generated

Note: If the result comes back as ‘False’, there was an issue in the creation.

*Reading data can be done a multitude of ways. Depending on the granularity of your query, less results will be displayed. The below screenshot shows a query for all animals that match the “Cat” animal type:*

A screenshot of a computer screen

Description automatically generated

*Note: If looking for a specific animal, try to make the query more granular. This will return all data*

*that matches your search query.*

*When updating data, you have to define the update key, which is the value to be changed, then the update value, which is the value that takes the place of the changed data. Continuing with our newly created cat named Leah, we can change the animal ID to demonstrate:*

*A screenshot of a computer code

AI-generated content may be incorrect.*

*The update is made and the number of files changed will be returned under the block. You can see below that the animal\_id has been changed from the original value to the updated value:*

*A screenshot of a computer

AI-generated content may be incorrect.*

*Delete can be used to remove any data that matches your query. In this case, we’ll delete the file we created:*

*A screen shot of a computer

AI-generated content may be incorrect.*

*Again, the request is processed and the number of files deleted is returned to the screen. Now, when we search for that specific animal ID, we get no data:*

A computer code with black and red text

AI-generated content may be incorrect.

*Note: Be sure to filter to the exact data you want to delete. This function will delete all data that matches the query.*

### Screenshots

*When using the dashboard, you may want to quickly filter to the type of rescue animal that would suit you best. The quick filter widgets can help you with that. The dashboard includes a table, a pie chart, and a geolocation chart to help with the search as well:*

**A screenshot of a computer

AI-generated content may be incorrect.**

**A map with a pin on it

AI-generated content may be incorrect.**

*The widgets allow for easier filtering like so:*

***Water Rescue***

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a map

AI-generated content may be incorrect.**

***Mountain or Wilderness Rescue***

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a map

AI-generated content may be incorrect.**

***Disaster or Individual Tracking***

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a map

AI-generated content may be incorrect.**

*There is also a “Reset” widget that will set the table back to the default view. Using these widgets can significantly speed up the search for the correct animal.*

***Challenges***

*The main challenge I faced was getting the dashboard to generate. I decided to add the port into the app.run\_server() function after some googling and that fixed the issue. In hindsight, it seems like the port was being ignored in the app creation and could have been added like the username and password, but I managed to get it running.*

## Contact

Your name: Jon Marvin