# CS 340 README Template

## About the Project/Project Title

The purpose of the Grazioso Salvare Rescue Animal Training MongoDB CRUD Python Module is to provide a set of tools for managing and manipulating data related to rescue animals. This module is part of the larger Grazioso Salvare project, which focuses on identifying and categorizing dogs suitable for search and rescue training.

## Motivation

The Grazioso Salvare Rescue Animal Training project is driven by a dual mission of enhancing search and rescue effectiveness by using trained rescue dogs, as well as prioritizing animal welfare by providing purpose and better lives for shelter dogs. Collaboration with nonprofit animal shelters in Austin, Texas strengthens community ties. Data-driven decisions optimize training suitability, while technology like MongoDB and Python streamlines the data management. Python’s extensive libraries for data manipulation and ability to read BSON documents allow for easy interaction with MongoDB NoSQL database. Open-sourcing the code on GitHub encourages collaboration. Ultimately, the goal is a positive community impact by improving search and rescue outcomes, enhancing animal welfare, and fostering collaboration among stakeholders.

## Getting Started

To get the local database running follow these steps:

1. Ensure that MongoDB is installed and running locally or on a remote server.
2. Clone the repository from https://github.com/Beth-Camp1989/CS340 into MongoDB using mongoImport.
3. Change the current database to AAC database in mongoShell.
4. Test and ensure that AAC has been imported.
5. Enable a user and password with readWrite capabilities in the AAC database using admin in mongoShell.
6. Run Jupyter Notebook with Python.
7. In Jupyter Notebook, open the animal\_shelter.py file.
8. Change the username, password, and port to fit the user you set-up in step 5.
9. Open the Project Two.ipyng.
10. In Project Two, run the program and go to the dashboard.

## Installation

To use the Grazioso Salvare Rescue Animal Training MongoDB CRUD Python Module in your project, you’ll need the following tools and libraries:

* Jupyter with Python: You may install this from jupyter.prg/install
  + Python, facilitated by the pymongo library acts as a bridge to enable seamless integration with MongoDB and empowering the project to retrieve and manipulate data with precision. Python’s versatility is capable of performing robust data analysis due to libraries like NumPy and Pandas. These tools delve into intricacies of animal data, conduct comprehensive analysis, and generate insightful reports. Python’s web framework seamlessly integrates with MongoDB to create user-friendly management and access of animal data. Scripts capitalize on MongoDB’s scalability options. Real-time updates to MongoDB’s database provide current and accurate data. Security is ensured by integrating MongoDB and Python.
* MongoDB: Ensure you have MongoDB installed and running. You may download it from mongodb.com.
  + MongoDB plays a pivotal role as the project’s primary database system and efficiently manages vast datasets related to rescue animals. It’s flexibility and scalability make it ideal for housing a wide array of information such as age, breed, and name. MongoDB integrates with Python to facilitate user-friendly management and access to animal data. The robust authentication and access control features fortify against unauthorized access and data breeches. MongoDB’s scalability equips the project to accommodate expanding datasets gracefully.

## Usage

1. Data Management: Store, retrieve, update, and delete data related to rescue animals, such as their age, breed, name, and rescue type.  
A close up of a text

Description automatically generated

2. Data Analysis: Perform data analysis on the stored data to make data-driven decisions regarding training suitability.A screenshot of a computer

Description automatically generated

3. User-Friendly Interface: Provide a user-friendly interface for managing and accessing animal data. A screenshot of a computer

Description automatically generated

4. Security: Ensure the security of data by integrating MongoDB for robust authentication and access control.

5. Scalability: Leverage MongoDB's scalability options to accommodate expanding datasets.

**Screenshots:**

A screenshot of a computer

Description automatically generatedA map with lines on it

Description automatically generatedA screenshot of a map

Description automatically generated

## Roadmap/Features (Optional)

Open Issues:

* Remote buttons not working as expected. Later updates to come

## Contact

Your name: Beth Campbell