# CS 340 README

## About the Project

The client Grazioso Salvare provides a service to train rescue dogs. They work with animal shelters to find the best fit dogs to become rescue dogs. The client wants a program to help search and filter through the animal shelters database to make searching for the best fit dogs a lot easier. The CRUD Python module provides a set of functions that will create, read, update, and delete data from a database. It is created to be easy to use and work with other applications and make managing the animal shelters database easier.

## Motivation

This module was developed to provide a simple and consistent way to interact with databases. The module erases the need to rewrite code for common CRUD operations. This makes it easier to interact with and manipulate the data related to the animal shelter. Along with the database the client wishes for a dashboard that provides a filterable search and map of animals the animal shelter has.

## Installation

## The following tools below are needed to create this program:

* Python
* MongoDB
* Mongoimport
* Jupyter notebook

## Getting Started

* Using mongo import, import the new data into a newly created database.
* Create a new user using the create user command. Give the new user their username and password, as well as access to the newly created database. (The same database the animal shelter data was imported into)
* Create a python file that connects to your database with the Mongo Client, make sure to include the username, password, host, port, and database name. This python file should include all features of CRUD.
* Jupyter can allow you to create a dashboard to simplify CRUD functions by importing your python file and class.
* Use Jupyter and import the python CRUD file to create a dashboard to filter and show a map of the animal shelter rescues.

## Usage

### MongoDB Python Driver

The MongoDB Python driver used in this module is **pymongo**. It is a widely used driver that offers a high-level API for interacting with MongoDB databases. **pymongo** is chosen for its ease of use.

**MongoDB import and user creation.**

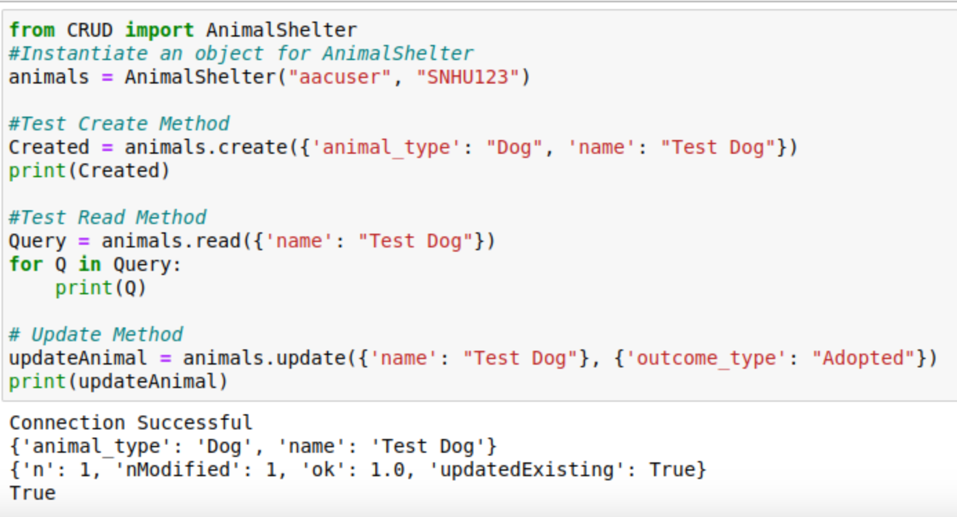
MongoDB offers a way to import data files from a source into the Mongo database with ease. MongoDB also has a method to create users with different access levels to modify and interact with the database.

**Crud Methods**

The CRUD Python module has the following attributes:

1. **Create**: Inserts a new document into a specified MongoDB database and collection. It takes a dictionary of key-value pairs representing the document data.
2. **Read**: Queries for documents from a specified MongoDB database and collection based on a provided query dictionary.
3. **Update**: Queries for and updates documents in a specified MongoDB database and collection
4. **Delete**: Queries for and deletes documents from a specified MongoDB database and collection based on a provided query dictionary. It returns the number of deleted documents.

### Code Example:





(Example of Juypter file simplifying the python CRUD file)

A white sheet with black text and a rabbit head

Description automatically generated with medium confidence

A map with a map and a map with a map and a map with a map and a map with a map and a map with a map and a map with a map and a map with

Description automatically generated

(Dashboard showing the reset filter that shows all animals from the animal shelter database)

A screenshot of a computer

Description automatically generated

A screenshot of a map

Description automatically generated

(Filter for only water rescue dogs)

A screenshot of a computer

Description automatically generated

A screenshot of a map

Description automatically generated

(Filter for only mountain/wilderness rescue dogs)

A screenshot of a computer screen

Description automatically generated(Filter for only the disaster rescue and individual tracking dogs)

### Tests

One method to ensure the module is working is to have checks along the code.

A black background with white text

Description automatically generated

The above code is for the create method, if the create method works then it will return true, if it fails it will return false. I then test the code by printing my create method and see which return I receive. There were some challenges mainly in ensuring the syntax of the code was correct, but a simple search for mongo commands helped solve these problems.

## Contact

Kyle Lund