# CS 340 README Template

## About the Project/Project Title

The project entails creating a Python module that enables CRUD functionality for a document collection. The functionality includes create (C), read (R), update (U), and delete (D). The document collection is from the Austin Animal Center (AAC) and includes key value pairs such as the ID, age, breed, color, animal type, name, and more.

## Motivation

The Python module will eventually be used to connect the user interface component to the database component of the dashboard.

## Getting Started

To get a local copy up and running, follow these simple example steps:

* First, I imported the AAC CSV file into MongoDB using the MongoDB insert tool and used the database name “AAC”.
* Next, I added indexing for optimizing queries.
* Then, I set up user authentication to the database by setting up both administrator and user accounts.
* After the first two steps, I began the process of setting up the animal database for CRUD (Create, Update, Read, and Delete) in PyMongo.
* First, I started without authentication, so it doesn’t get in the way.
* Continue by calling PyMongo, the Mongo client, to connect to the database; then set the database as AAC within Mongo.
* Create functions as methods for the class for create and read in Jupyter.
* Create a test file in Jupyter to ensure the create and read methods work.
* Create functions as methods for the class for update and delete in Jupyter.
* Create a test file in Jupyter to ensure the update and delete methods work.
* Add the Grazioso Salvare logo.

## Installation

The tools needed for this project are quite simple:

* A Linux terminal with a Mongo shell
* PyMongo
* Jupyter notebook

## Usage

### Code Example

1. Insert a CSV file: Open the terminal window to access the Linux shell. Upload the Austin Animal Center (AAC) Outcomes data set into MongoDB by importing a CSV file using the appropriate MongoDB import tool. Use the database name “AAC” and collection name “animals”.

Text

Description automatically generated

1. After importing the data set, start up the mongo shell. Create a simple index on the key “breed”.

Text

Description automatically generated

Text

Description automatically generated

1. Ensure user authentication to the database by setting up both administrator and user accounts: Create an administrator account in the mongo shell.

**Text

Description automatically generated**

**Text

Description automatically generated**

Create a user account called “aacuser” for the database AAC in the Mongo shell.

**Text

Description automatically generated**

**Text

Description automatically generated**

Create and read functions with tests:

*Graphical user interface, text, application

Description automatically generated*

*Graphical user interface, text, application

Description automatically generated*

### Create and Read Tests

*Text

Description automatically generated*

*Text

Description automatically generated*

Update and delete functions:

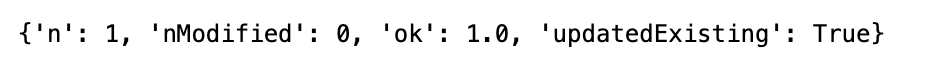
Graphical user interface, text, application, email

Description automatically generated

Update and delete tests with output:

Graphical user interface

Description automatically generated with low confidence



A picture containing company name

Description automatically generated

Adding the logo:

Graphical user interface, text, application, email

Description automatically generated

## Roadmap/Features (Optional)

*Provide an open issues list of proposed features (and known issues). If you have ideas for releases in the future, it is a good idea to list them in the README. What makes your project stand out?  
  
Note: This section is optional for the purposes of this assignment. If you choose not to fill out this section, remove it from your final README file.*

## Contact

Your name: Ashli Campbell