# 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

*The purpose of the project is to develop an application that is able to access animal data from shelters that can be used to find animals that meet searchable criteria.*

## Motivation

*The motivation behind this project is to provide an effective way for users to interact with a MongoDB database without needing to write complex queries or handle database connections manually. By streamlining the process of managing animal shelter records, users can focus more on data management than on technical details.*

## Getting Started

1. *Install MongoDB*
2. *Using the terminal import the data*
3. *Use mongsh and switch to db*
4. *Begin search queries*

## Installation

1. *Python- current version can be found at python.org*
2. *MongoDB- free version available at mongodb.com*
3. *Jupyter Notebook- current version available at jupyter.org*

**Tools Used**

*1. MongoDB*

*MongoDB is a NoSQL database that stores data in a flexible, JSON-like format called BSON. It was chosen because of schema flexibility, scalability, powerful query language, and integration with Python.*

*2. Dash*

*Dash, developed by Plotly, is a web application framework for Python designed for building analytical web applications. The reasons for using Dash are easier application management, user interface, real-time interactivity, integration with Plotly.*

*3. JupyterDash*

*JupyterDash is an extension of Dash that allows the dashboard to run within a Jupyter Notebook environment. This was used to facilitate development and testing because of the ability to interactively develop and its ease of use.*

*4. Pandas*

*Pandas is a data manipulation library for Python that was used for handling data in tabular form because of the use of DataFrames and its integration with Dash.*

**Steps**

1. *Gathered Requirements*
2. *Setup Database*
3. *Develop crud.py*
4. *Develop Dashboard*
5. *Testing*
6. *Create README*

## Usage

*The CRUD Python module is designed to provide an easy-to-use interface for interacting with MongoDB databases, specifically enabling Create, Read, Update, and Delete operations. The Python class allows this module to be reusable and gives the ability to use MongoDB databases programmatically. This module should be used when preforming CRUD operations on MongoDB collections. Each method in this module is meant to handle a specific CRUD function. The module handles errors increasing its reliability and allowing it to be expanded for future needs.*

### Code Example

*A screen shot of a computer program

Description automatically generatedA screen shot of a computer code

Description automatically generated*

### Tests

A screenshot of a computer

Description automatically generatedA screenshot of a computer screen

Description automatically generatedA screenshot of a computer program

Description automatically generated

### Screenshots

*A computer screen with white text

Description automatically generatedA computer screen with colorful text

Description automatically generated*

*A red line drawing of a dog

Description automatically generatedA screenshot of a dashboard

Description automatically generatedA screenshot of a dashboard

Description automatically generatedA screenshot of a map

Description automatically generatedA screenshot of a dashboard

Description automatically generatedA screenshot of a map

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated*

## Contact

Your name: Devin Criswell