# CS 340 README

## PROJECT OVERVIEW

Grazioso Salvare is seeking a software application that can work with existing data from five animal shelters to identify and categorize available dogs. By utilizing several user-friendly components to create a visualization pipeline using MongoDB, Juypter Notebook and Dash Framework, all shelters can manipulate data for the common goal on a web-based application dashboard.

PyMongo was select as the as the projects driver because it is the official MongoDB driver for synchronous Python applications. Being able to connect to the database instance and perform seamless actions such as Create new animals, Update existing data, Read or query animals, and Delete data, enables all shelter authenticated users to execute their primary mission.

Dash Framework is well suited for this application due to its versatility. Using the Python ecosystem, this framework functions well with our main data handling library Pandas. Due to the large amount of data from all shelters, this is the perfect tech stack to meet all expectations. Additionally, Dash is specifically tailored to web-based applications which provides Grazioso Salvare a common hub for data.

## Motivation

This web-based application will allow Grazioso Salvare to access a centralized database consisting of animals from multiple shelters. By defining attributes, Grazioso can identify possible search and rescue training candidates.

## Installation

*Install MongoDB* [*https://www.mongodb.com/docs/manual/tutorial/install-mongodb-on-os-x/*](https://www.mongodb.com/docs/manual/tutorial/install-mongodb-on-os-x/)

*Install Juypter Notebook* [*https://jupyter.org/install*](https://jupyter.org/install)

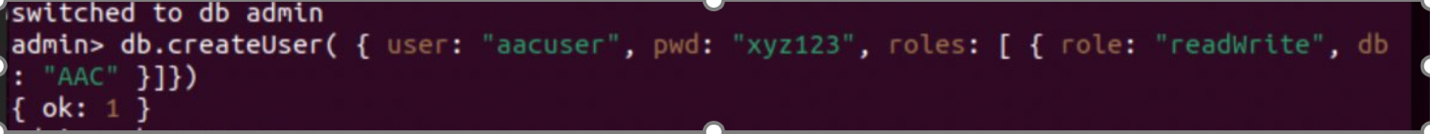
Getting Started

1. Access Terminal and import database using .env variables or pre-defined user data

A computer code on a dark background

Description automatically generated

1. Access admin database and create new user.



1. Authenticate user and verify access.

A screen shot of a computer code

Description automatically generated

Usage

Utilizing a Model View Conroller design patter, the use of creating, reading, updating and deleting data is performed in the Model portion. As you can see, this minimum code allows for a wide distribution use throughout the code. Pulling data from Mongo Data Base to be converted through the Pandas Data Framework.

C: Create a new object or animal in the database.A computer screen shot of text

Description automatically generated

R: Read or query for a specific animal from the database.

A computer screen shot of a code

Description automatically generated

U: Update an animal from the database.

A screen shot of a computer code

Description automatically generated

D: Delete an animal from the database.

A screen shot of a computer

Description automatically generated

*DEMO*

1. *Access through designated URL*

**

1. *Use predefined searches radio buttons*
   1. *Ex: Water Rescue*

*A close-up of a card

Description automatically generated*

* 1. *Returns eligible animals that meet Water Rescue criteria.*

*A white background with black text

Description automatically generated*

1. *Select ONE animal of your choice and its location will populate on a local map.*

*A map with a blue pin on it

Description automatically generated*

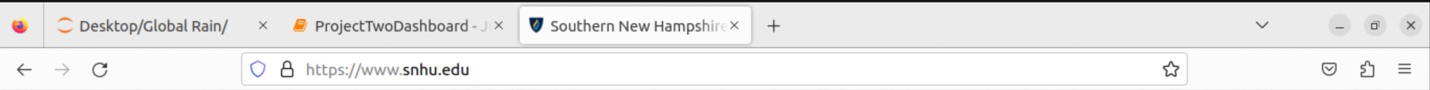
1. *Additional Features include monthly statistics:*

*A pie chart with text overlay

Description automatically generated*

1. *Links: Each Logo links to SNHU Home Page.*

*A logo of a dog

Description automatically generated*

## Roadmap/Features:

Upcoming Feature 1: Current searchable modeling on 10k plus animals is strictly defined by three categories. Implementation of your won criteria by having multiple input components with drop down menus for your detailed searches.

Upcoming Feature 2: Pie Chart manipulation based on your own selected criteria.

Upcoming Feature 3: Once search has been narrowed down and selection has been made, a picture of the animal with 360 deg rotation for viewing.

Upcoming Feature 4: Scheduling.

Webpage Update: Implement header and footer for Website tabs such as About us, Contact, Mission, and FAQ. Footer will implement additional context for resources and company policies.

*Development:*

*Current issues: Code doesn’t meet full MVC model. Code needs to be further segregated into smaller files such as Crud, callbacks, app.*

*The application evolved from several shelters repurposing animals for a greater good. Combining all the data into one database (Mongodb) and then transforming that data into usable and visual data on the web (Pandas Framework) and then creating widgets (searchable buttons/map/charts- Leaflets) to allow the user to have control over their options has created a well-defined web based application.*

*After importing the database, I created a reusable CRUD file and imported in into the main application. Using several libraries such as Juypter, pandas, and dash allowed for the required framework to meet the required tasks.*

## Global Rain:

Daniel Dickinson

[Daniel.dickinson@GlobalRain.com](mailto:Daniel.dickinson@GlobalRain.com)