// SNHU

// CS 340: Client/Server Development

// Student Name: THOMAS SEIBERT

// MOD 7 – Project Two (README)

**# Describe the required functionality of the project.**

The required functionality of the project was to essentially build a full-stack program which utilizes MongoDB on the bottom layer, a Python CRUD module as middleware, and a web application dashboard as the top layer. This functionality allowed the client, Grazioso Salvare, an animal rescue organization, to interact with and visualize the database they need to organize and carry out their work more efficiently.

* Now I will include the screenshots of the functional dashboard, starting with my unique identifier and the client’s logo at the top of the page:
* (screenshots begin on next page)

A screenshot of a computer

AI-generated content may be incorrect.

* Next, a screenshot of the data table from which the client can view the data they choose to view based on radio button selections which triggered various queries:
* (screenshots on next page)

A screenshot of a computer

AI-generated content may be incorrect.

* Under the data table, both a histogram chart and a map appear, and are both subject to change based on which choice (radio button) the client chooses to select. The first histogram shows the results of Water Rescues:
* (screenshot on next page)

A screenshot of a computer

AI-generated content may be incorrect.

* Next, the results of Mountain or Wilderness Rescues:

A screenshot of a computer

AI-generated content may be incorrect.

* Next, the results of Disaster or Individual Tracking:

A screenshot of a computer

AI-generated content may be incorrect.

* Finally, the results of the default setting, or reset, which show the “big picture” of the total number of animals in the database:
* (screenshots on next page)

A screenshot of a computer

AI-generated content may be incorrect.

**# Describe the tools used to achieve this functionality and a rationale for why these tools were used.**

The primary tools which were used were:

• A virtual Linux desktop in support of Python, Spyder, MongoDB shell, and Jupyter Notebook.

* MongoDB was used as the model component in this program due to its flexibility, scalability, and ease of integration with Python. It features a document-oriented model, which makes it particularly suitable for modern applications where data structures evolve over time. In this case, intuitive queries were integrated with Python which instantly gathered the necessary information for the client.
* The Dash framework was the top layer, or client-facing component of the application. It is through the Dash web application that the client can, with the click of a button, request and gather the information they need.

**# Explain the steps that were taken to complete the project.**

The primary steps that were taken were:

* Accessing of the AAC database through MongoDB.
* Building of the Python CRUD module which acts as the middle layer between MongoDB and the Dash application.
* Deployment of the client’s Dash application, which utilizes the Python module, which in turn allows the client to access the information they need to more efficiently organize their operation.

**# Identify any challenges that were encountered and explain how those challenges were overcome.**

The most difficult challenge for me personally was the development of the final deployment of the Dash application. To complete the project, I referenced Module Seven’s resources and looked over every line of code thoroughly, referencing it with the Python CRUD module, to make sure I understood what needed to be changed in both files to accomplish the final goal. Overall, the project was not easy because I had no prior experience with full stack programming but for that reason, it was also an invaluable lesson and learning experience.