Enhancement Three: Databases

Zaffar Shiekh

CS-499-10941-M01 Computer Science Capstone 2024 C-4

Federico Bermudez

8/4/2024

**Artifact Description**

The artifact I selected for the databases category is a CRUD application for an animal shelter management system. It was initially created during my CS 340 - Advanced Programming Concepts coursework. The application uses MongoDB for its database operations and provides Create, Read, Update, and Delete (CRUD) functionalities for managing animal records.

**Justification for Inclusion**

I included this artifact in my ePortfolio to showcase my database management and application development skills using MongoDB. The CRUD operations are fundamental in software development, and this project demonstrates my ability to design and implement these operations effectively. Enhancing this artifact allowed me to refine my database interaction, error handling, and user interface design skills.

**Improvements and Enhancements**

The enhancements I made to this artifact include:

1. **Refactoring the Code for Local MongoDB Access**:
   * Previously, the application was configured to connect to a remote MongoDB instance, which caused authentication issues. I modified the connection settings to use a local MongoDB instance without authentication for easier testing and deployment.
   * **Changes**:

In animal\_shelter.py, I updated the connection string to use localhost and port 27017.

1. **Error Handling and User Feedback**:
   * Improved error handling in the CRUD methods to provide more informative feedback to the user.
   * **Changes**:

Added try-except blocks in each CRUD method to catch and print errors.

1. **Dashboard Integration**:
   * Integrated the CRUD application with a Dash-based dashboard to provide a user-friendly interface for viewing and filtering animal records.
   * **Changes**:

In dashboard.ipynb, connected the dashboard to the local MongoDB instance and added filtering options for different rescue types.

**Course Objectives and Learning Outcomes**

With this enhancement, I aimed to meet several course objectives, including demonstrating the ability to design and implement database solutions and improving my skills in creating user interfaces for database interactions. I have achieved these objectives by successfully enhancing the functionality and usability of the animal shelter management system.

**Reflection on the Enhancement Process**

Enhancing this artifact taught me valuable database management and user interface design lessons. One of the main challenges I faced was resolving the authentication issues with the remote MongoDB instance, which led me to configure a local instance for testing purposes. This experience reinforced the importance of understanding the deployment environment and a flexible problem-solving approach. Additionally, integrating the application with a dashboard required me to think about how users interact with the data and how to present it meaningfully.

**Accessing and Running the Database**

To run and access the enhanced animal shelter management system, follow these steps:

1. **Install MongoDB Locally**:
   * Download and install MongoDB from the [official MongoDB website](https://www.mongodb.com/try/download/community).
   * Start MongoDB without authentication by running the following command in a terminal (or PowerShell on Windows):

mongod --dbpath /path/to/your/db --bind\_ip 127.0.0.1 --port 27017

1. **Set Up Jupyter Notebook**:
   * Install Jupyter Notebook by running the following command in PowerShell:

pip install jupyter

1. **Run Jupyter Notebook**:
   * Navigate to the directory where your notebook file is located:

cd "C:\Windows\System32\Untitled Folder\Untitled Folder"

* + Start Jupyter Notebook:

jupyter notebook

* + This will start the Jupyter Notebook server and provide a URL with a token key to access the notebook. Look for a line that looks like this:

http://localhost:8888/?token=YOUR\_UNIQUE\_TOKEN

1. **Access the Notebook**:
   * Copy the provided URL with the token key and paste it into your web browser to access the Jupyter Notebook interface.
2. **Run the Dashboard**:
   * Open the dashboard.ipynb file in the Jupyter Notebook interface.
   * Execute the cells to start the Dash server.
   * The dashboard will be accessible at the provided local URL.

Following these steps, you can access and interact with the enhanced animal shelter management system, view animal records, and apply various filters through the integrated dashboard.

This approach ensures you can run and test the application locally without authentication issues. Once you confirm the functionality, you can configure authentication for a more secure deployment.