||

5-2 Milestone Four: Enhancement Three: Databases

Shekhar Chaudhary

CS-499 Computer Science Capstone

10/02/2025

||

### ****Introduction****

The database enhancement I implemented for this milestone originated from my **CS-340 (Client/Server Development)** project and was further aligned with the data design concepts explored in **CS-300 (Data Structures)**. The original artifact involved developing a MongoDB-driven application for managing and retrieving animal shelter data using a web dashboard interface. For this enhancement, I focused on **normalizing the schema**, **adding stored procedures for core database operations**, and **introducing role-based access control (RBAC)** to strengthen both the usability and security of the system.

The goal of this enhancement was to improve the scalability, security, and maintainability of the database component by ensuring that the schema adhered to third normal form (3NF) and that user operations were governed by defined roles (e.g., admin, staff, viewer). These improvements directly contribute to the Computer Science program outcomes related to **secure database design, query optimization, and adversarial-aware software architecture**.

### ****Artifact Description****

The database originally consisted of a single “Users” and “Animals” collection managed through MongoDB. While functional, it lacked normalized relationships and centralized security enforcement. For this milestone, I enhanced the artifact by:

1. **Normalizing Tables:** Breaking down unstructured collections into smaller relational entities (Users, Roles, Permissions, and Animals). This eliminates redundancy and ensures data consistency.
2. **Implementing Stored Procedures:** Creating procedures for repeated operations such as adding users and inserting animal records while validating input roles and permissions.
3. **Integrating Role-Based Access Control:** Restricting actions to specific roles — for example, only administrators can add new users, and only registered users can add or modify anima

### ****Planned Enhancement and Pseudocode****

One of the major enhancements was implementing a stored procedure for secure user creation. The pseudocode for this procedure is as follows:

A computer screen with white text

AI-generated content may be incorrect.

This procedure ensures only predefined roles are accepted, preventing unauthorized privilege escalation. Furthermore, each user’s password is hashed using the PBKDF2 algorithm (pbkdf2\_sha256) before storage, ensuring that sensitive credentials are never stored in plain text.

Another enhancement involved modularizing authentication into separate files (Authentication.py) to improve code readability and maintain separation of concerns between the application logic and authentication logic.

Screen Shoots Mongo Shell:

A screenshot of a computer

Description automatically generated

Navigated to file /usr/local/datasets:

A screenshot of a computer

Description automatically generated

Retrieve a document from the collections:

show dbs #lists directory of databases

use enron #this sets db to the enron database

show collections #lists directory of collections

db.emails.findOne() #retrieves a document from the emails collection

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

Execute the command to find the size of a single document:

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer

Description automatically generated

Execute the command to find the size of the collection of documents:

A screenshot of a computer program

Description automatically generated

### ****Conclusion****

This milestone demonstrates my ability to blend **database theory and secure software engineering practices** into a functional and scalable solution. By refining schema design, implementing role-based stored procedures, and securing authentication mechanisms, I successfully improved the reliability, maintainability, and integrity of my CS-340 database application.

Through this enhancement, I showcased:

* Expertise in **database normalization and query optimization**,
* Proficiency in **secure authentication and role-based design**, and
* The ability to **anticipate vulnerabilities and enforce privacy protections** through thoughtful schema and code structure.

These accomplishments highlight my growth as a computer science professional ready to design, build, and secure data-driven systems in real-world applications.

**References:**

* Repository: [github.com/shekharchaudhary/CS-300](https://github.com/shekharchaudhary/CS-300)
* Repository: [github.com/shekharchaudhary/CS-340](https://github.com/shekharchaudhary/CS-340)
* Mongo Shell Operations Reference: CS -340 Module One

||

Thank you

||