**CS-499 Final Narrative: Professional Self-Assessment**

Throughout this course of instruction, I took a rudimentary, Java based authentication system and a made a series of enhancements to it that align with the intended course outcomes, in the following ways:

*Course Outcome 1: Employ strategies for building collaborative environments that enable diverse audiences to support organizational decision-making in the field of computer science*

I employed strategies for building collaborative environments that enable diverse audiences to support organizational decision-making in the field of computer science by integrating an SQL database. This shift from a file-based system enhances security and scalability, enabling collaborative team interactions. For example, the database setup allows for concurrent access and modifications by multiple users without data corruption, essential for teams spread across different geographies and time zones.

*Course Outcome 2: Design, develop, and deliver professional-quality oral, written, and visual communications that are coherent, technically sound, and appropriately adapted to specific audiences and contexts*

I designed, developed, and delivered professional-quality oral, written, and visual communications that are coherent, technically sound, and appropriately adapted to specific audiences and contexts. My code is structured into functions like ‘create\_account’ and ‘user\_login’, each documented with clear comments explaining parameters and processes, making it accessible for future developers and auditors. Additionally, the extensive system documentation provides setup and operation instructions, making it user-friendly for non-technical administrators who need to manage user accounts.

*Course Outcome 3: Design and evaluate computing solutions that solve a given problem using algorithmic principles and computer science practices and standards appropriate to its solution, while managing the trade-offs involved in design choices*

I designed and evaluated computing solutions that solve a given problem using algorithmic principles and computer science practices and standards appropriate to its solution, while managing the trade-offs involved in design choices. By replacing MD5 with bcrypt for password hashing, I enhanced security against brute-force attacks. The username generation mechanism handles duplicates by appending a sequential number, efficiently ensuring uniqueness without extensive database queries. For instance, if ‘jsmith’ exists, the next user ‘John Smith’ becomes ‘jsmith1’, optimizing the process.

*Course Outcome 4: Demonstrate an ability to use well-founded and innovative techniques, skills, and tools in computing practices for the purpose of implementing computer solutions that deliver value and accomplish industry-specific goals*

I demonstrated an ability to use well-founded and innovative techniques, skills, and tools in computing practices for the purpose of implementing computer solutions that deliver value and accomplish industry-specific goals. The use of bcrypt, an industry-standard for secure password hashing, and SQLite, a lightweight relational database, exemplify the use of robust technologies suitable for web applications requiring high security and scalability. This setup significantly reduces the risk of data breaches while maintaining quick access to user information.

*Course Outcome 5: Develop a security mindset that anticipates adversarial exploits in software architecture and designs to expose potential vulnerabilities, mitigate design flaws, and ensure privacy and enhanced security of data and resources*

I developed a security mindset that anticipates adversarial exploits in software architecture and designs to expose potential vulnerabilities, mitigate design flaws, and ensure privacy and enhanced security of data and resources. One specific measure is the enforcement of password changes for new users, which requires them to update their temporary password upon first login. This ensures that any password potentially exposed during account setup is quickly made obsolete. Additionally, by storing old hashed passwords and checking new passwords against them, I prevent password reuse, addressing a common security oversight in many systems.

Each of these points illustrates not only the alignment with the educational objectives outlined in the course outcomes but also provides tangible examples of how the enhancements in my Python authentication system contribute to a secure, efficient, and scalable solution.Top of Form

Bottom of Form