CS 499

Professor Troy Hawk

Michael Thomas

8 June 2025

Milestone Four Narrative

1. Briefly describe the artifact. What is it? When was it created?
   * The GraziosoSalvare artifact is a command-line inventory and rescue animal management system originally created for *IT 145: Foundation in Application Development* during the summer of 2023, and it managed basic animal data using arrays and file output. While functional, the original version lacked persistent storage and any formal database structure, making it difficult to scale or query efficiently. For my improvement, I replaced the flat data structure with a normalized SQLite database and implemented secure, parameterized SQL commands to persist and retrieve animal records. The refactored version includes a full CRUD interface, validation logic, and a modular layout, aligning more closely with professional-grade applications.
2. Justify the inclusion of the artifact in your ePortfolio. Why did you select this item? What specific components of the artifact showcase your skills and abilities in software development? How was the artifact improved?
   * I selected this artifact to showcase my growth in working with databases, structured data models, and secure query design. The original version was a procedural script using arrays and file writes for temporary storage, which meant data was lost on each session reset and was not easily searchable. By refactoring the program to use Python's sqlite3 module and introducing a persistent SQLite backend, I demonstrated my ability to normalize and model real-world data, apply relational schema design, and implement parameterized SQL queries to prevent injection vulnerabilities. The enhancement also splits functionality across separate modules (database.py, models.py, and main.py), and applying software engineering best practices for maintainability and clarity. These improvements reflect practical development approaches seen in entry-level software and database roles.
3. Did you meet the course outcomes you planned to meet with this enhancement in Module One? Do you have any updates to your outcome-coverage plans?
   * Yes, I successfully met my planned outcomes, specifically Course Outcomes 3 and 4. For Outcome 3, the transition from array-based logic to normalized relational database design involved applying algorithmic reasoning and trade-offs—such as managing schema constraints, ensuring data integrity, and optimizing for search and update operations using SQL. For Outcome 4, the project demonstrates the use of industry-standard tools and techniques, such as structured schema definitions, parameterized SQL, modular Python design, and secure user inputs. I also included basic validation to protect against malformed data. No updates are needed to my original outcome-coverage plan, as all enhancement goals were met or exceeded.
4. Reflect on the process of enhancing and modifying the artifact. What did you learn as you were creating it and improving it? What challenges did you face?
   * Enhancing this artifact deepened my appreciation for the role of persistent storage and data abstraction in scalable applications. I learned how SQLite can offer a powerful yet lightweight solution for structured data management, and how to build logic that bridges user input with relational queries in a secure and reusable manner. One challenge I faced was abstracting database logic from user interaction without losing clarity, especially when designing the models.py layer to interface cleanly with the CLI. I also had to ensure that constraints like species type and reserved status were enforced through both code and schema design. The experience reinforced the value of separating concerns and the importance of secure, validated data flows in software applications.