Timothy McIntire CS 499 Professor Hawk June 3, 2025 (revised June 19, 2025)

Milestone Three Narrative

1. Briefly describe the artifact. What is it? When was it created?

The artifact I selected for this milestone is the same console-based Java application used in Milestone Two. Originally developed for IT 145, the application simulates a rescue animal management system for a fictional organization called Grazioso Salvare. It enables users to intake new animals, reserve animals, and print animal data via a menu-driven interface. The project has evolved since its original creation to support scalable features, including modularized animal kingdoms (Mammal, Bird, Reptile, etc.). The most recent enhancement for this milestone focused on improving the application's performance by replacing linear search data structures with hash-based alternatives.

2. Justify the inclusion of the artifact in your ePortfolio. Why did you select this item? What specific components showcase your skills and abilities in software development? How was the artifact improved?

I selected this artifact for inclusion in my ePortfolio because it demonstrates my ability to refactor and optimize existing code using appropriate data structures. The original version of the system used ArrayList collections for each animal type, which required linear iteration to search for animals by ID during operations like reservation and display. As the application expanded to support multiple animal kingdoms and a potentially large number of records, the inefficiency of linear searches became evident. To address this, I enhanced the system by replacing each ArrayList with a HashMap<String, RescueAnimal>. The key is the animal's ID, enabling constant-time average-case lookup operations. This change significantly improved the system's scalability and responsiveness. For example, the reserveAnimalById method was rewritten to directly retrieve an animal object using get(id) from a combined map, rather than iterating over all records. This enhancement showcases the following skills and concepts:

- Understanding of algorithmic time complexity and trade-offs $(O(n) \rightarrow O(1))$
- Practical application of hashing and key-value storage
- Refactoring an existing codebase without disrupting functionality
- Ensuring scalability for systems handling growing data sets

In addition to performance improvements, I also considered data integrity when transitioning to HashMap. By using unique animal IDs as keys, I ensured that each entry is stored and retrieved without duplication or conflict. Although HashMap is not inherently a secure storage mechanism, its structure allows for precise control over data consistency, reducing the risk of accidental overwrites or orphaned data. I implemented checks to prevent null access or duplicate keys, which are basic but effective safeguards to maintain logical correctness and prevent runtime errors. In future iterations, stronger validation and exception handling would further protect the integrity of stored animal records.

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?

The enhancements completed align with the following program learning outcomes:

- **Course Outcome 3**: The migration to HashMap demonstrates deliberate algorithmic thinking. I evaluated the inefficiency of linear searches and implemented a structure optimized for frequent lookups, thereby improving real-world performance.
- **Course Outcome 4**: Rather than redesigning the entire system, I integrated HashMap collections into the existing architecture. This shows my ability to improve and modernize software incrementally while preserving original intent and structure.

These outcomes were included in my original outcome coverage plan in Module One and remain accurate. No updates to the planned outcomes are needed.

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 the artifact was both a challenging and rewarding experience. Refactoring a project that was originally structured around ArrayList collections required careful attention to how animals were stored, accessed, and printed. I learned to think critically about data access patterns and how system performance can degrade with scale. The HashMap enhancement taught me how to implement constant-time operations in practice and reinforced the importance of selecting the right data structure for the job.

One of the most valuable lessons was realizing that optimization is not always about rewriting everything from scratch. In this project, the goal was to integrate better performance with minimal disruption to existing functionality. This required thoughtful planning and testing to ensure existing methods like intake, print, and reserve continued to work as expected.

Rubric Criteria Summary

Rubric / Milestone Question	How It Was Met
Briefly describe the artifact	Provided a summary of the original IT 145
	project, its purpose, and how it evolved
Justify inclusion	Selected due to its clear need for optimization
	and its visibility in the ePortfolio
Skills showcased	Demonstrated understanding of data
	structures, hashing, lookup efficiency, and
	refactoring
Course outcomes met	Confirmed alignment with CO3 and CO4
	from Module One. No update to outcome
	plan is required
Reflection on process	Shared lessons learned, explained technical
	decisions, and discussed challenges faced
Technical enhancement submitted	All Java files with enhancements are ready
	and reflect the switch from ArrayList to
	HashMap
Narrative reflects milestone goals	Clear explanation of learning, problem
	solving approach, and technical decisions
Narrative is ready for feedback	Narrative written with focus on clarity,
	milestone prompts, and rubric expectations