Milestone Two: Software Design/Engineering Enhancement

Robert Heavner

SNHU

CS-499

Professor Bryan

03/15/2025

Artifact Overview

A Python-based program called the Rescue Animal System was created to monitor and control the availability, training progress, and admission of rescue animals. Users can register new animals, keep track of their training progress, and reserve animals for service using the system, which supports dogs, cats, and monkeys.

This artifact was first created as a project of a software development project in a prior computer science course. The system's initial iteration prioritized reporting features, input validation, and fundamental object-oriented architecture.

Justification for Inclusion in ePortfolio

This project was chosen as it had the ability to showcase an executable program via command line and showcases skills of porting from one language to another, in this case Java to Python. It also shows the ability to add features and functionality that makes the program more robust.

It specifically demonstrates my competence in several important areas. This project shows strong data validation and organized input processing, ensuring the accuracy of user-provided data for important fields like training statuses, dates, and weights. The inclusion of a progress bar for tracking training status demonstrates the artifact's efficient data reporting and visualization, which greatly enhances readability and the user experience in general. It also shows my dedication to improving and maintaining the system by reworking and streamlining features associated with reservations, training updates, and list filtering. These changes highlight my outstanding programming abilities and demonstrate my ability to write clear, effective, and maintainable code.

Enhancements & Improvements

Several changes were made to this artifact, resulting in improvements. I was able to give a more understandable and user-friendly depiction of animal training statuses, a visual training progress monitoring system was implemented. This system includes a progress bar with percentage indicators. A logic-based dynamic reservation system was created to effectively screen and reserve available animals. Modularization greatly improved the code structure by separating functions better, which made the code easier to read and maintain. Strong data validation and error handling procedures were put in place, enhancing user input validation to stop inaccurate data entries. Together, these enhancements greatly improve the system's general functionality, data quality, and usability while keeping the codebase clear and manageable. Moving this project from Java to Python should make maintaining the code easier; less lines of code means less cost and less cost overall is

better when making these types of changes. Moving to python has also allowed for easier readability of the code and moving it to a more modern code base, this logic is equally as trusted as Java and is here to stay.

Alignment with Course Outcomes

Several program outcomes in the Computer Science curriculum are supported and aligned with the improvements made to this artifact. An enhanced design is demonstrated by the creation and use of a progress bar visualization for training status, meeting the goal associated with developing and assessing computing solutions. Strong software engineering and system implementation is also demonstrated by the reworking of modular functions to produce clearer and more effective system logic. Finally, the enhanced data representation made possible by progress bars improves user experience and facilitates better decision-making, meeting the goal of clear and professional communication. The result coverage plan doesn't need to be updated because the intended Module One enhancement goals have been fulfilled with this project.

Reflection on Enhancement Process

Enhancing this artifact turned out to be a good experience that offered real-world use in programming, software restructuring, and user experience design. Important lessons were learned, like the importance of user experience, which was illustrated by the training progress bar's notable improvement in the clarity of training status updates. It reaffirmed the significance of programming by showcasing how dividing tasks into reusable modules improves the scalability and maintainability of systems. Smaller chucks of code make reading and making quick changes especially in Agile environments can really help you make small improvements that can be reviewed and tested easily.

The experience proved the value of strong user input validation and emphasized how crucial error management is to averting system disasters. To ensure a clear and effective representation, dynamic indexing and bar width calculations were used to overcome implementation challenges, such as the difficulty of turning training statuses into a visual progress bar. I really think this project helped showcase the ability to port logic from one language to another while keeping the logic modern and adding some functionality that will make this program a bit better. Just like with any logic you can always add more, but you must know when enough is enough. If I were going to make this a real project for a company, we would need to move data into a database along with having real UI for the customer to use. Having global temporary data is fine for this project, but the data is reset each time you close the program. In terms of my goals in CS499, this project showcases everything it should.

Summary

The Rescue Animal System was greatly improved by this milestone, becoming a more useful, scalable, and user-friendly program. The improvements make it a perfect addition to my ePortfolio since they highlight important software development abilities including OOP, data validation, reporting, and system optimization along with showcasing the ability to port from one language to another while keeping functionality and improving overall readability and having less lines of logic to maintain.

This project prepares me for professional software engineering employment by reaffirming my proficiency in software design, problem-solving, and real-world application development.