Jacob Goldberg

Professor Bryant

CS 499

1 February 2025

**Milestone Three Narrative**

For the category Algorithms and Data Structure, I am using the same artifact I used for the last milestone, which is the Java Rescue Animal program that allows users to submit Dogs and Monkeys to be taken in and added to a Rescue Animal system, where they can then be reserved and a list of the animals in the system can be printed. I chose this artifact to use for this category because the program is centered around the management of data structures through the use of algorithms in the code.

The unmodified version of this program only deals with Dog and Monkey objects. These objects are stored and accessed through array-lists in the original Java code. In order to demonstrate my skills and abilities when it comes to utilizing algorithms and data structures in a project, I have enhanced the program by adding support for additional animal types, including cats and birds, as well as made it so that the user can enter their own custom animal type to be taken into the system, reserved, and listed. These new object types are stored in their own respective vectors instead of array-lists, with the custom animal types all sharing an Animal object type. This is because I rewrote the program in C++ as part of my first enhancement. As I modified and enhanced the artifact, I learned that it made more sense for each specific animal subclass (dog, monkey, cat, and bird) to have their own respective vector object associated with them, as creating a vector that could house all the different subclasses together connected by their shared parent class was beyond the scope of the enhancement, especially considering the implementation of a database system to manage objects in the next enhancement will render such a development pointless.

As it stands, by creating the additional data types, restructuring the process of intaking a new animal to consider the additional types, and allowing the user to input their own custom animal type into the system securely using C++'s secure character input functionality, I was able to meet the following course outcomes as planned:

* 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.
* 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.