Brianna De La Riva

Professor Krupa

CS-499 Computer Science Capstone

22 September 2024

**Artifact One Narrative**

**What is it?**

The artifact is a Java program designed to handle recurring appointments. It calculates the next occurrence of an appointment based on a start date, frequency, and interval. The program supports weekly, monthly, and yearly recurring appointments. The original project would only allow for new appointments, include an appointment ID, and allow for specific criterias to be met (i.e., description max length, date must be in future, etc). However, with the new enhancement, it allows for added functionality to the program.

**When was it created?**

The program was developed as part of a scheduling system project. It was originally created within my CS320 Software Quality Assurance and Automation course in December 2023.

**Why did you select this item?**

I selected this artifact because it showcases my ability to apply Java programming skills to solve a practical problem related to scheduling. The CS-320 course was one I enjoyed taking in the past, and it fell within the course list of class options I could pick an artifact from. I believe this project fit within the criteria for software engineering/design, as well as meeting the needs of the selection criteria.

**What specific components of the artifact showcase your skills and abilities in software development?**

The artifact demonstrates:

Data Structures: I utilized Java collections such as `ArrayList` and `HashMap` to store and manage appointment data and scheduling rules.

Algorithmic Principles: Implements algorithms to calculate the next occurrence of recurring appointments based on various intervals and there frequencies.

Object-Oriented Design: Shows the ability to design classes and methods that handle different types of recurring appointments, improving code modularity and readability.

**How was the artifact improved?**

The artifact was enhanced by adding the functionality of allowing users to create recurring appointments. This helped enhance the user functionalities, and their input handling to ensure that all the appointment information is correct.

**Meeting Course Outcomes**

**Did you meet the course outcomes you planned to meet with this enhancement in Module One?**

Yes, the enhancement met several course outcomes, for example:

Algorithmic Principles: The improved logic for calculating recurring appointments demonstrates a deeper understanding of algorithms and increased code readability.

Design and Evaluation: The use of data structures like `HashMap` and `ArrayList` showcases the ability to design and evaluate computing solutions effectively.

**Do you have any updates to your outcome-coverage plans?**

Currently, I do not have any upcoming updates at the moment. The enhancements have covered the planned outcomes related to software engineering and design.

**Reflection on the Enhancement Process**

**What did you learn as you were creating it and improving it?**

During the enhancement, I learned the importance of designing flexible algorithms and leveraging Java’s data structures to manage and process recurring appointments efficiently.

**What challenges did you face?**

Some of the challenges I faced involved handling the recurrence patterns, as well as ensuring that there were correct and accurate data calculations being made. It was complicated at first to figure out the specific logic to be used for recurring appointments, but it turned out to be a success.

**Alignment with Computer Science Program Outcomes**

Below is a list of some of the computer science program outcomes and how I related to them.

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

This artifact showcases individual coding skills that are crucial for collaborative environments, such as designing and maintaining scheduling systems.

1. Design, develop, and deliver professional-quality oral, written, and visual communications that are coherent, technically sound, and appropriately adapted to specific audiences and contexts.

The Java program showcases technical communication through well-structured code and frequent comments listed throughout. This shows an ability to develop clear and professional-quality software solutions.

1. 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 (data structures and algorithms)

The enhanced artifact effectively uses Java data structures and algorithms to solve the problem of recurring appointments, demonstrating an understanding of appropriate practices and standards.

1. 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

While the project primarily focuses on the scheduling Java logic, incorporating the user input validation and error handling contributes to making the code more reliable and secure against any potential issues that may arise.