The artifact is an Advising Software developed as a project for CS 300: Data Structures and Algorithms. It’s a tool to help assist with course advising by storing information about courses, including course numbers, names, and prerequisites. Additionally, it provides functionalities such as loading data from a file, displaying course lists, and printing course information.

This artifact was selected for inclusion in the ePortfolio due to its significance in showcasing proficiency in algorithms and data structures. Specifically, the project demonstrates the ability to design and implement efficient data structures (e.g., hash table, graph) and algorithms (e.g., topological sorting) to solve real-world problems. By leveraging these skills, the artifact demonstrates the ability to create practical software solutions with strong functionality.

The artifact was enhanced by integrating a dynamic array that stores a collection of strings representing advice quotes, aligning with the original enhancement plan. This addition expands the functionality of the program by promoting student success.

The enhancements made to the Advising Software successfully met the course objectives outlined. The addition of a dynamic array for the advice system expanded the functionality of the program, demonstrating proficiency in algorithm design and data structure implementation. By integrating advanced data structures such as graphs and algorithms like topological sorting, the project showcased a well-founded understanding of fundamental concepts in computer science. Security was also a key consideration during the development of the Advising Software. Potential vulnerabilities in the software architecture were anticipated, and measures were taken to protect student data. Steps such as thorough input validation and secure file handling were implemented to ensure data privacy and security.

During the process of enhancing and modifying the artifact, many challenges were encountered. I gained a deeper understanding of graph data structures and algorithms, particularly in the context of implementing topological sorting for prerequisite management.

Challenges faced during the enhancement process included focusing on a seamless integration of the advice system with existing functionalities and addressing errors when adding in the topological sorting. These challenges were overcome through debugging, careful consideration of design choices, and following industry standards.