Self-Assessment and GitHub Pages

Daryl Miller

Southern New Hampshire University

CS-499-T4546 Computer Science Capstone 22EW4

April 15, 2022

**Content**

[GitHub Pages Welcome 3](#_Toc100910431)

[Professional Self-Assessment 3](#_Toc100910432)

[Code Review 7](#_Toc100910433)

[Course Outcomes 8](#_Toc100910434)

[Enhancement One - Software Design and Engineering 10](#_Toc100910435)

[Enhancement Two - Data Structures and Algorithms 12](#_Toc100910436)

[Enhancement Three - Databases 14](#_Toc100910437)

# GitHub Pages Welcome

Hello, my name is Daryl “DJ” Miller and I want to start by thanking you for taking the time to review my ePortfolio. This ePortfolio serves as the culmination of my hard work during my course work here at Southern New Hampshire University (SNHU). The objective of this ePortfolio is to demonstrate skills and abilities I have learned will completing my coursework. This ePortfolio is comprised of the professional self-assessment, the code review, and three enhancements to artifacts I completed throughout this program. Each of the three artifacts will correspond to demonstrated skills and abilities in the categories of software design and engineering, algorithms and data structures, and databases all the while adhering to the programs’ five course outcomes.

# Professional Self-Assessment

I began my computer science journey here at SNHU in the fall of 2018 and will complete my degree upon the completion of this course in the Spring of 2022. Prior to starting this course, I had only a rudimentary understanding of computers and how things like hardware and software work behind the scenes. Upon successfully completing this degree program, I have gained skills such what different components make up a computer, how to code in different languages, and how important security is to the overall computer science field.

This ePortfolio will focus primarily on the skills and abilities I have learned with respect to the three enhancements contained within. These are not however a complete reflection of the skills and abilities I have learned while here at SNHU. I have also demonstrated proficiency in areas such as Collaborating in a team environment, communicating to stakeholders, data structures and algorithms, software engineering, databases, and security. I will take each of the auxiliary skills and abilities in turn and then focus primarily on the three artifacts to give anyone reading this ePortfolio in-depth information on each artifact and how it clearly aligns to the course outcomes of this program.

Collaborating in a team environment is vital to any employee within a company. Whether that is within the Software Development Life Cycle (SDLC) or somewhere else, a person must know how to work collaboratively in the modern agile workplace. CS-310-J4779 Collaboration and Team Project is a perfect example of this as this course thought me how to collaborate on a software project by applying appropriate change control and versioning practices EGit. This course shows how a programmer can successfully be part of either a co-located or geographically separated team. It taught me skills such as branches, commits, and version control to allow the work I am doing to not negatively affect the work of my teammates.

CS-250-J6435 Software Development Lifecycle reinforced the skills of working within a team and gave me experience filling various roles within an agile team such as the developer, tester, and scrum master. This course also gave me experience communicating with key stake holders such as the delivery product owner, who would communicate with the customer of the product being developed. This is vital as often developers are required to wear many different hats and be able to communicate at a more technical level within the team, but also be able to explain the program in plain English to key stakeholders.

Data structures and algorithms are one of the three key categories covered in this ePortfolio. There are two courses that taught me the basics and advanced concepts of data structures and algorithms. CS-260-J1278 Data Structures and Algorithms taught me how to use code and non-coding development methodologies in algorithmic design and problem solving. This course also taught me the skills to used advanced algorithmic designs to evaluate complex data structures to aid in problem solving. CS-340-T4084 Client/Server Development expanded on my data structure and algorithms skills by reinforcing the key concepts and teaching me how to apply database systems concepts and principles to develop client/server applications that interface client-side code with databases. I was able to create algorithms using the Create, Read, Update, and Delete (CRUD) functionality to interact with the database all the while utilizing a Graphical User Interface (GUI) to manipulate the data.

Software Engineering is the second key categories and was covered heavily throughout this program. Courses such as CS-360-H6813 Mobile Architecture and Programming taught me how to design a mobile application on the Android Operating System (OS) using Android Studio. CS-330-J2957 Computational Graphics and Visualization gave me the skills to develop three-dimensional objects using Application Programming Interfaces (APIs) libraries and best practices. Finally, CS-410-J1744 Software Reverse Engineering taught me the basics of reverse engineering specific software systems to recreate missing documentation and to support legacy software code. This course also taught me how reverse engineering is used to make new software products, enhance the functionality and efficiency of software components, and recreate the code for applications.

Databases is the final key category covered within this ePortfolio and is highlighted by several courses throughout this program. DAD-220-J3007 Introduction to Structured Database Environments taught me the basic skills of creating structured database environments that incorporate basic processing functionality and allow for data management, data manipulation, and data analysis using MySQL. This course was again reinforced using CS-340-T4084 Client/Server Development to expand into the Mongo Database as another database centric skill.

Security is the backbone of any program and should always be on the forefront of a developer’s mind. CS-405-T2675 Secure Coding was the quintessential course that taught security to a new programmer like me. This course focused on common security vulnerabilities that a developer is often faced with. This course taught me techniques and strategies to develop secure code and leverage secure programming principles. I got authentic experience identifying security vulnerabilities and writing secure code to mitigate risks to software and data.

The first enhancement, which can be accessed at this [Enhancement One] (https://github.com/scdaryl2005/SNHU-CS-499-Capstone/tree/main/Enhancement%20One) link on GitHub, is a zoo monitoring program, which satisfies the requirement for the software design and engineering key category. The original artifact was completed in IT-145-R5558 Foundation in Application Development. This artifact is a java application that monitors zoo animals and exhibits. The main java application uses buffer readers, file readers and scanners to perform actions such as accept input from the user and get information from external text sources. The enhancement made to this artifact includes expanding upon the original sample size of the animals and exhibits, as well as adding additional features such as zoo staff. I also streamlined the program to make it leaner and added additional comments to give a first-time programmer more understanding of what each block of code does toward the overall execution of the program.

The second enhancement, which can be accessed at this [Enhancement Two] (https://github.com/scdaryl2005/SNHU-CS-499-Capstone/tree/main/Enhancement%20Two) link on GitHub, is an algorithm-based program that uses vector sorting to satisfy the requirements for the data structures and algorithms key category. The original artifact was completed in CS-260-J1278 Data Structures and Algorithms. This artifact is a vector sorting application written in C++. The main idea behind this application is to load a .csv file to display and sort bids for an auction. The main enhancement I made to this artifact was to incorporate the linked list algorithm and added additional features such as allowing the user to enter, find, remove, and prepend a bid.

The third and final enhancement, which can be accessed at this [Enhancement Three] (https://github.com/scdaryl2005/SNHU-CS-499-Capstone/tree/main/Enhancement%20Three) link on GitHub, is a SQL program used to satisfy the databases key category. The original artifact was completed in DAD-220-J3007 Introduction to Structured Database Environments. I chose to include this artifact into my ePortfolio because it shows a lot of the skills required to be a SQL developer. Though the original artifact is fictious and basic, it can be attributed to many of the current SQL databases implemented by companies today. This artifact highlights some of the Create, Read, Update, Delete (CRUD) skills needed to maintain a SQL database. The main enhancements I made to the SQL was to expand upon its complexity and add advanced concepts of SQL.

# Code Review

I performed my first code review ever in my computer science journey, but quickly understood the importance of code review and how it fits within the SDLC and other aspects of collaboration between a team. The practice of code review is important because it allows developers, either individually or part of a team, to break down their application by certain blocks of code and review it for things such as structure, documentation, variables, arithmetic operations, loops and branches, and defensive programming. This is a part to whole approach that allows the programmers to break down their code into more manageable pieces to begin reviewing it, which in turn will eventually lead to a more defect free product once it is passed to the Quality Assurance (QA) team. This code review will analyze each of the three original artifacts and discusses the enhancements I intended to make to each. My published code review can be found on YouTube by using this [Code Review] (https://youtu.be/XHoT8YvAXIM) link.

# Course Outcomes

CS-499-001: Employ strategies for building collaborative environments that enable diverse audiences to support organizational decision making in the field of computer science

I have demonstrated my skills for this outcome via the code review. The code review is a collaborative review of the program, which would allow a development team to take the code block by block to ensure functionality, conformity, and that the objective meets the scope laid out by key stakeholders. This would also ensure that best practices within the team are adhered to so that each person’s contribution to the program is standardized. Finally, I have demonstrated these skills and abilities in each of the three artifacts below by ensuring the programs, where applicable, provide contextual, in-code comments that result in easily readable and understandable code.

CS-499-002: Design, develop, and deliver professional-quality oral, written, and visual communications that are coherent, technically sound, and appropriately adapted to specific audiences and contexts

I have demonstrated my skills for this outcome through the orally narrated code review that provides an overview of what each artifact is and how I planned on enhancing it. This code review could then be reviewed by other individuals on a development team or presented to key stakeholders to get approval for the proposed enhancements to the original artifacts. I also demonstrated my skills for this outcome by producing three narrated documents for each outcome that clearly identify what each artifact is, the enhancements I made on each, and best practices for future development via a retrospective of my development.

CS-499-003: Design an 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

I have demonstrated my skills for this outcome through the enhancements made to artifact 2. I began by taking the pseudocode for the original artifact and ensured that what I initially designed matched the final product. From there, I modified the pseudocode to incorporate the approved enhancements to the program. Once all enhancements had been completed, I went back to verify the enhanced pseudocode matched the enhanced artifact. I clearly demonstrated the ability to solve logic problems within artifact two by developing different algorithms and data structures to not only load bids from a .csv file, but also sort them using vectors and modify them using a linked list algorithm.

CS-499-004: 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

I have demonstrated my skills for this outcome in artifact 1 by creating a more robust and efficient code that delivers values to the zoo personnel and adheres to industry-specific goals of creating an application that monitors animals, habitats, and staff within an exact replica of the San Antonio Zoo. I utilized innovative techniques and skills such as allowing the user to access the system using buffers and scanners to manipulate it to determine the overall health of an animal, habitat, or staff member. Furthermore, this program, or one similar could be developed within the iterative phases of the software development life cycle (SDLC) to ensure the least amount of bugs present as early as possible so that cost could be minimalized and time to market could be quicker.

CS-499-005: 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

I have demonstrated my skill for this outcome in both artifact 1 and artifact 2 by ensuring the program adhered to industry-specific security standards. I addressed these concerns during the requirement phase so that security was at the forefront of my development. In artifact 1 for example, I ensured that all external files that were open, were closed and that each case statement had a break point. In artifact 2 I utilized statistic methods for testing. I could have enhanced the security furthermore in artifact 3 for example by transferring the code from LiveSQL to a mongo database and added authentication features to ensure only those who needed access had access to manipulate the database. All three artifacts are also free of bugs and vulnerabilities. I could also run each of the code through various third-party testing software to ensure there were not any additional vulnerabilities associated with the artifacts.

# Enhancement One - Software Design and Engineering

The artifact I chose for Enhancement 1 was from IT-145-R5558 Foundations in Application Development. I took this course here at SNHU in the fall of 2019 and completed the original artifact at that time.

I chose to include this artifact in my ePortfolio because not only was it on the approved list, but it showcased my initial skills early on in my Computer Science journey. This artifact showcases my ability to design a basic program in Java with minimal experience. The program itself shows my understanding of various concepts such different import statement, buffers, scanners, If-Else statements, and case statement use. These skills also show auxiliary skills such as indexes and indicators. I made a few enhancements to the original artifact that will be submitted to my ePortfolio. The first was to expand on the original data sets. The two .txt files, “ANIMALS” and “HABITATS” only had a few entries into each file. I took the original files and updated them to reflect the current animals and habitats listed at the San Antonio Zoo. The next enhancement I made was to expand the options presented to the user. I coded in a new section to accommodate a new file titled “STAFF.” The final enhancements made were to streamline the program itself to make it more lean and cleaned up the comments so that it is not over commented.

Below is the full list of enhancements I made to this artifact:

- Updated the original “Animals.txt” file to include 38 animals versus the original 5

- Updated the original “Animals.txt” file to include Quantity, Sex, and Age fields

- Updated the original “Habitat.txt” file to include 16 habitats versus the original 3

- Updated the original “Habitat.txt” file to include animals present in habitat

- Created an additional “Staff.txt” file to assign staff to different exhibits and animals

- Updated the Switch / Case statement to include case 3 to accommodate the new “Staff” option

- Updated the menu to accommodate the “Staff” option for the user

- Added additional comments to the code to identify what each block of code is doing

- Verified each block of code worked without errors or warnings

- Added “.close();” statements to ensure each of the text files are closed properly

- Verified each case uses breaks

- Verified each scanner is closed

- Performed various formatting updates to ensure the code conforms to java best practices

- Verified the code works as intended and zero warnings or errors are present

Reflecting on the process of enhancing this artifact, I learned many things from its initial design. The main takeaway was to see where my skills were when I began versus the knowledge I have at the present time. This is a visual representation that shows how much my skills have improved. I did face several self-imposed challenges. First, I did not have any of the IDEs installed on my current computer. I had to go back and research what the original artifact was and how I got it to work properly then. The second was basically relearning Java. I had not written code in Java in well over a year, so I had to brush up on the syntax and structure inherent to Java. Finally, rather than suppressing the warnings I had with the original artifact, I had the knowledge to solve those warnings such as not closing the scanner, which would have allowed for data leaks and created security concerns.

The original, enhanced, and narrated artifacts can be found using this [Enhancement One] (https://github.com/scdaryl2005/SNHU-CS-499-Capstone/tree/main/Enhancement%20One) link.

# Enhancement Two - Data Structures and Algorithms

The artifact I chose for Enhancement 2 was from CS-260-J1278 Data Structures and Algorithms. I took this course here at Southern New Hampshire University (SNHU) in the spring of 2019 and completed the original artifact at that time.

I chose to include the vector artifact from CS-260 to satisfy the requirements for data structures and algorithms. This artifact was initially created to showcase the skills of implementing data structures and algorithms. Utilizing the vector data structure, different bids could be either loaded directly into the program, or called upon using the load function, which is what was used to import all the bids from the attached CSV file. It is because Vectors were the foundation which everything else was built upon, I have included it within my portfolio of work. The specific components that showcased my skills in data structures and algorithms was the vector sorting data structure, which allowed me to utilize the algorithm to partition the .csv file and sort it by other methods than simply a high or low. The selection sort and quick sort logic were used to divide the list into a low and high section that was then used to display the desired sorting method. The main enhancement I made to this artifact was to incorporate the linked list algorithm and added additional features such as allowing the user to enter, find, remove, and prepend a bid.

Below is the full list of enhancements I made to this artifact:

- I added the “# include <string>” header file

- I validated that all additional header files were present and needed

- I added an additional class for the “LinkedList” algorithm

- I initialized the bid

- I added a section to allow the user to add their own bid

- I created a block to partition the imported .csv file into halves using the heads and tails node

- I added a block of code to allow the user to delete a particular bid

- I added a block of code to allow the user to search for a particular bid

- I updated the menu to allow for additional features such as entering a bid, finding a bid, removing a bid, and prepending a bid and therefore expanding on its original complexity

- I ensured that all loops and cases were properly closed and iterated through each as intended

- I made various additional inline comments to ensure readability for subsequent developers

- Finally, once all enhancements were complete, I verified that the program worked as intended and was free of any errors or warnings

Reflecting on the process of enhancing this artifact, I learned many things from its initial design. The biggest take away was how valuable reusable code can be. Taking the initial artifact and expanding its complexity to incorporate additional features shows how reusable code can be invaluable to a programmer. Rather than starting from scratch, I could take the linked list code, modify it, and incorporate it into my main artifact. The only challenges I faced were minor adjustments to the code to get it to compile properly once I incorporated the linked list features.

The original, enhanced, and narrated artifacts can be found using this [Enhancement Two] (https://github.com/scdaryl2005/SNHU-CS-499-Capstone/tree/main/Enhancement%20Two) link.

# Enhancement Three - Databases

The artifact I chose for Enhancement 3 was from DAD-220: Introduction to SQL. I took this course here at Southern New Hampshire University (SNHU) in the spring of 2019 and completed the original artifact at that time.

I chose to include this artifact into my ePortfolio because it shows a lot of the skills required to be a SQL developer. Though the original artifact is fictious and basic, it can be attributed to many of the current SQL databases implemented by companies today. This artifact highlights some of the CRUD skills needed to maintain a SQL database. Though the original data was limited, I chose to expand on the data to add additional features to make it more extensive, which highlights my skills as a SQL developer. I have shown innovative ways to manipulate a database and perform the everyday CRUD functionality to maintain it. I have also addressed the outcomes by programming a solution to solve problems involving storing, manipulating, and accessing data. Finally, this written narrative demonstrates my ability to clearly articulate my ideas and accomplishments as demonstrated by the working product.

Below is the full list of enhancements I made to this artifact:

- Since the original artifact was given to me in a third-party website and was no longer available, I had to recreate the SQL from scratch using LiveSQL from Oracle

- Once the original database was built, I expanded on its complexities by incorporating additional features such as adding another table, adding more Olympic athletes, adding additional contacts in the contact list, and adding additional messages to messages table

- I created a step-by-step guide with screenshots included to recreate this database

- I performed advanced CRUD actions such as deleting things based on a query versus row by row

- Finally, I developed a complete script within LiveSQL so that anyone who searches for my script, can easily run it to produce the same results

- Future enhancements would include incorporating a mongo database to add additional security features such as authentication to the database and a JavaScript interface

Reflecting on creating the original artifact was hard because I was brand new to computer science and specifically SQL. Even though I got a great grade on the original artifact, I still wanted to learn more about SQL and thought that I could improve the artifact. During Enhancement 3, I took my original artifact and improved it by adding additional features and performing advanced SQL concepts. In the end I am happy with the enhancements made to the artifact and could see myself as a SQL developer in the future. The only challenge I faced when enhancing this artifact was learning the nuances between coding in MySQL versus LiveSQL. For example, MySQL uses INT (8), whereas LIVESQL uses NUMBER (8) when creating a table. After I sorted out the language difference, it was strait forward in replicating and expanding upon the original artifact.

The original, enhanced, and narrated artifacts can be found using this [Enhancement Three] (https://github.com/scdaryl2005/SNHU-CS-499-Capstone/tree/main/Enhancement%20Three) link.