**CS 499 Final Project: ePortfolio Documentation**

**By Alexander Varljen**

**August 13, 2022**

**Table of Contents:**

1. Relevant Links (page 3)
2. Professional Self-Assessment (pages 4 – 6)
3. Code Review (page 7)
4. Artifact One Enhancement Narrative (pages 8 – 12)
5. Artifact Two Enhancement Narrative (pages 13 – 17)
6. Artifact Three Enhancement Narrative (pages 18 – 23)
7. References (page 24)
8. **Relevant Links:**

In this section, you can find the relevant links that pertain to the ePortfolio. These include **Link 1**, which is a link to my finished GitHub Pages ePortfolio, **Link 2**, which is a link to a zipped file containing my code review video from my Google Drive account, and **Link 3**, which is a link to an MOV file version of the code review video from my Google Drive account, which can be viewed from the webpage. The code review video file links will also be given in the *Code Review* section of this document.

Link 1:

<https://alexander-varljen.github.io/>

Link 2:

<https://drive.google.com/file/d/19qB4ywqz7rjAAgLcKoWCNpRRDYibPWUh/view>

Link 3:

<https://drive.google.com/file/d/10cxJDtofgnS_sbGbILLz1FpD3kXjxyGa/view>

1. **Professional Self-Assessment:**

I have been in SNHU’s B.S. of Computer Science program since the summer of 2019. During this time, I’ve participated in courses and projects that range from CS 310 Collaboration and Team Project, where we worked together as a class to use versioning, group repositories, and code reviews to work on a software application as a group, to CS 250 Software Development Lifecycle, where I participated in each stage of the SDLC. I have learned how to design affective software for specific purposes, construct algorithms and deploy data structures that complete complex tasks with a relatively short amount of code, and create, read, update, and delete items from databases. To showcase these skills, I enhanced three of my past projects, or *artifacts*, which are a bank application, a program that stores and sorts bid data, and a database with three tables where a full gamut of CRUD commands were used.

To justify why I chose these projects, I should first discuss what the course requirements are and where my interests and career plans lie. First, we were to select one artifact from each of three categories: software design/engineering, algorithms and data structure, and databases. Also, we needed to demonstrate our ability to collaborate in a team environment, effectively communicate to stake holders, and develop a security mindset. During my experiences as a computer science major, and after some of my own research, I’ve determined that I would focus my career plans on data analysis and data science. Data analysis requires skills related to database querying, an understanding of certain programming languages, such as SQL and Python, and the skills required to report findings, such as an ability to perform statistical analysis and create visual representations of data (Grupman, C., 2022). The skills required of data scientists are similar, yet often require more advanced knowledge of mathematics, SQL as well as NoSQL languages (such as MongoDB), and an understanding of machine learning algorithms (Berkeley Extension, 2022). There are other projects that I have worked on that demonstrate skills related to data analysis and data science, such as my final project for MAT 243 Applied Statistics for STEM, where I used Python to perform statistical analysis on the NBA basketball team, the “Wizards”, team performance data set. However, for this ePortoflio, I chose the three projects that I thought would best highlight my skills in the three computer science areas mentioned, as well as skills related to data science and data analysis.

The first artifact enhancement that I chose was the source code for a bank application, where I reconstructed the C++ program using Python. This allowed me to show skills related to software design/engineering, as I was able to maintain the intended functionality of the original project while using a different programming language. It also highlighted my skills in a programming language that is used in data analysis and data science.

The second artifact enhancement that I chose was the storing and sorting bids program, which was written in C++. The original program used chaining and a hash table data structure to store bids. To demonstrate my skills in constructing adequate algorithms and data structures, I increased the functionality of the program by adding vector sorting to sort the bids in ascending order. While I have not had the opportunity to learn machine learning algorithms yet, I plan on continuing my education in the future, and am hoping that I can apply what I’ve learned about data structures and algorithms to work on more advanced algorithms, such as machine learning algorithms.

The third artifact enhancement that I chose was the QuantigrationUpdates Database enhancement. The original project consisted of creating a three tabled database with MySQL, querying that database, updating it, deleting records from it, and downloading a table from it. My enhancement was to achieve the same functionality using MongoDB, a popular NoSQL language, along with adding user authentication. This highlights my ability to apply CRUD capable commands and queries in both SQL and NoSQL based languages to databases, which is a useful skill in both data analysis and data science.

While working on these artifact enhancements and constructing my ePortfolio, I was also able to demonstrate the ability to work in a way conducive to a collaborative environment, communicate with stakeholders, and ensure software security. My ePortfolio webpage gives access to my public GitHub page, where one can find, access, download, and, potentially, contribute to the page. I’ve also included explanations, code comments, and data sets for the benefit of any visitors. I’ve also developed oral, visual, written communication in the form of a screencast video of my artifacts’ code review, flowcharts and UML diagrams of my artifacts and their enhancements, narratives accompanying the artifact enhancements, and readme documentation where appropriate. I’ve also worked on exception handling to reduce the effects of bugs in the first two artifacts’ software, as well as password-secured user authentication for my QuantigrationUpdates Database, both of which demonstrate a security mindset when it comes to working with code.

1. **Code Review:**

I created a video recording of a code review that I had performed on the three artifacts that I selected for my enhancements. In the video, I first cover the project that I selected for my Artifact One Enhancement, then the project that I selected for my Artifact Two Enhancement, then the project that I selected for my Artifact Three Enhancement. There are two links pertaining to this video, both of which are located on my Google Drive account. **Link 1** is the location of the zipped file version and **Link 2** is the location for the MOV version of the file, which can be viewed from the webpage itself.

Link 1:

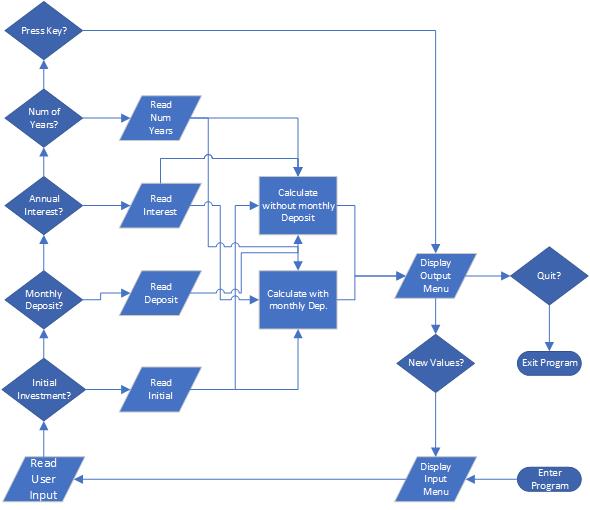
<https://drive.google.com/file/d/19qB4ywqz7rjAAgLcKoWCNpRRDYibPWUh/view>

Link 2:

<https://drive.google.com/file/d/10cxJDtofgnS_sbGbILLz1FpD3kXjxyGa/view>

1. **Artifact One Narrative:**

**Banking Program’s Flowchart:**



**Description of the Artifact:**

The original project that I picked to demonstrate my software development/engineering skills, which was created in February 2022, was the source code of a banking application called “CS 210 Project Two”. This artifact presented a menu which asks the user to give an initial investment amount, a monthly deposit amount, an annual interest rate, and the number of years (up to five years) in which they wish to see the change in their end of year balance and accrued interest. Then, the end of year balance and accrued interest are calculated based on these inputs and an output display is made. My enhancement of this artifact was to recreate the program using Python, as well as to add more security measures and shorten the length of the program’s source code to increase efficiency. This enhancement was made during the second week of July in 2022.

**Why This Artifact Was Chosen:**

I chose this artifact because it demonstrates my Python programming skills, which is a language commonly used in data analysis and data science, my main fields of interest. The range of Python programming techniques employed in this enhancement include user defined functions for subprocesses such as checking and receiving valid inputs, input receivers, while-loops, conditional statements, output displays, and basic arithmetic operations for calculating the year-end balance and interest. Also, through working on this artifact, I was able to secure the integrity of the software more faithfully by including exception handling for areas of the code receiving user input, as well as in areas that perform calculations.

**Course Objectives:**

While working on this artifact enhancement, I was able to employ strategies for building collaborative environments that enable diverse audiences to support organizational decision making in the field of computer science. This was done in three main ways. First, there are links to the zipped source file of this enhanced artifact, as well as to the original C++ project, which are both on my public GitHub repository. That public page also has a readme explaining the purpose of those and other files, along with an invitation to download any of the files for oneself. This allows anyone wishing to participate to learn about or potentially add to this project and make it their own or contribute to the project for the benefit of any developers accessing the page. Second, the source code in the enhancement itself employs modularity and clear user defined functions. This allows others to easily add to, remove, or move these modules, as well as more easily navigate the source code, for the purposes of collaborative work or independent efforts. Third, I made descriptive and specific code comments that allow others who would want to understand, change, or add to the code more easily able to do so.

While working on this artifact enhancement, I was able to design, develop, and deliver professional-quality oral, written, and visual communications that are coherent, technically sound, and appropriately adapted to specific audiences and contexts. In terms of oral communication for this artifact, I covered the original artifact in the first part of the code review video, where I went module by module discussing what the functionality of each module does and what improvements could be made. In terms of written communication, I wrote thorough code comments in my enhancement’s source code that describe the functionality of the user-defined functions and variables throughout the source code. I’ve also written this accompanying narrative to describe why I’ve chosen the artifact, what my enhancement of it was, and why I chose this enhancement. In terms of visual communication, I created an accurate flowchart showing the basic functionality of the artifact.

While working on this artifact enhancement, I was able to 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. One of the well-founded techniques that I employed in the creation of the source code for the artifact enhancement was to use exception handling in areas of the code that receive user input and perform calculations. This helped ensure that specific messages were presented by the program if a bad input or an error in calculation has occurred. Also, by creating a program composed of modular user-defined functions, I was able to demonstrate the well-founded technique of making a program modular. The tool that I used to develop, test, and run my artifact enhancement’s source code was the Python 3.6.6 Shell. This tool allowed me to run lines of code one module at a time to watch for errors and output in specific areas of the code, which helped me ensure that, by the end of my work, there were no errors and that the output is what was intended.

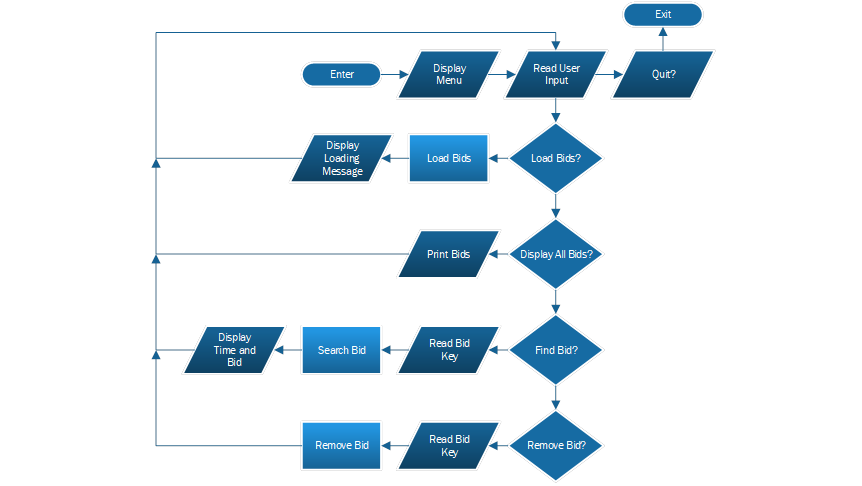
While working on this artifact enhancement, I was able to develop a security mindset that anticipates adversarial exploits in software architecture and designs to expose potential vulnerabilities, mitigate design flaws, and ensure privacy and enhanced security of data and resources. There are no semantic or syntax errors in the enhanced artifact’s source code. I was able to ensure this by running the program module by module using the Python 3.6.6 Shell. Also, I was able to add exception handling and messages for multiple situations involving invalid inputs and calculation errors. For example, for the user-defined function containing the code for receiving the initial investment amount from the user, I created exception handling code that throws out the message “Invalid initial investment” if the values are less than or equal to zero, as only a positive numeric value would make sense for the program’s purposes. I created similar exception handling code for the other three inputs asking for numeric values, as well as for the areas performing calculations based on the inputs.

**Reflection:**

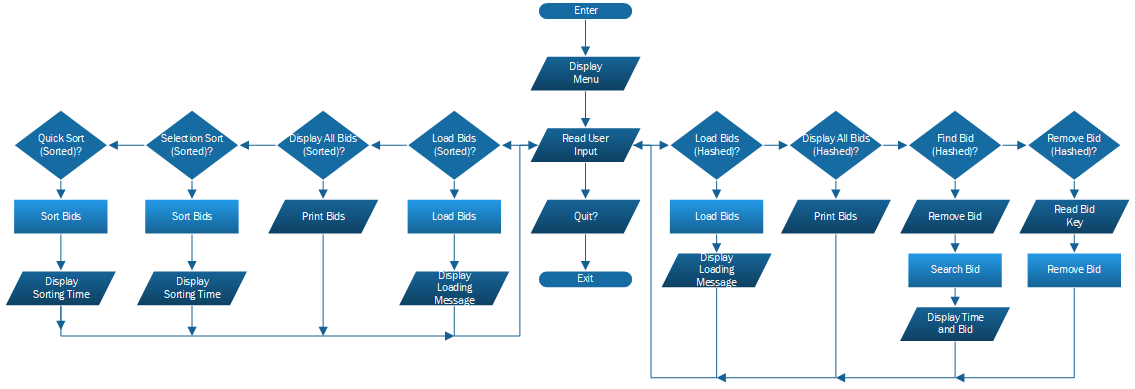
In the process of enhancing and modifying the artifact, I’ve had to refresh my knowledge of Python, as most of my previous experience with Python has been for constructing middle layer code for database queries and for statistical analysis. Therefore, I had to take some time and read documentation and practice with Python for more general purposes. Much of the new information that I needed to learn was also on exception handling in Python. I’ve had previous experience with exception handling using Java, but never with Python before working on this artifact. These challenges have caused me to, initially, work slowly on this artifact as I got my bearings. However, I’ve got more confident and competent along the way as I continued to work throughout the week of this artifact’s completion.

1. **Artifact Two Narrative:**

**Flowchart of Original Bids Program Artifact:**



**Flowchart of Bids Program Artifact After Enhancement:**



**Description of the Artifact:**

The original project from which this artifact enhancement was taken from was called “CS 260 Data Structures and Algorithms: Hash Tables and Chaining”, which I created for an assignment back in February of 2021. The goal of the project was to create a program that, via four menu options, allowed the user to load bid data from a csv file into a Hash Table data structure, display all the bids from within this data structure, find one bid from the list stored in the Hash Table, and remove one bid from the Hash Table. My enhancement to this artifact was adding vector sorting capacity with four new menu options, one for loading the bids into the vector that was used for the vector sorting, one for displaying all the bids in this vector, one for sorting the bids in that vector via selection sort, and one for sorting the bids in that vector via quick sort. This enhancement was made in the third week of July in 2022.

**Why This Artifact Was Chosen:**

I selected this artifact as my algorithms and data structures enhancement because it demonstrates my ability to store and manipulate data using multiple forms of data structure and different algorithms. These included an unmapped vector containing bid data and a mapped, or “hashed”, vector, also containing the bid data from a csv file. The unmapped vector was used by the vector sorting algorithms “quick sort” and “selection” sort, with selection sort being a simpler, but less efficient, algorithm and selection sort being a more complex but more efficient sorting algorithm, both of which were used to sort the bid data in ascending order. The mapped, or “hashed”, vector is a more complex vector to construct, but provides a much faster and more efficient way of finding an item, like a bid, than compared to using an unmapped vector. By including both types of vectors and their corresponding algorithms in this project, I was able to demonstrate that I can work with multiple algorithms and data structures, which each have their trade-offs, but can be utilized for different problems and desired outcomes.

**Course Objectives:**

While working on this artifact enhancement, I was able to employ strategies for building collaborative environments that enable diverse audiences to support organizational decision making in the field of computer science. Those who want to interact with the source file, download it for themselves, make desired changes and even, potentially, recommit changes on an updated version of the file can easily do so via my public GitHub repository. Also, the source code of the artifact enhancement has been made modular, which means that others change modules, delete modules, or add their own modules with a few code changes and without much impact on the other modules. Also, the code comments that I had written for the project make it easier for others to add to, change, navigate, or learn from the code. For example, when one opens the source file, they’ll notice a short description of the source code’s functionality at the top of the file, two boxed-in code comments highlighting two main divisions in the program, which are the modules for the hashed table data structure and those for the vector sorting algorithms, and code comments describing each module. These help others working with and navigating through the source code understand how everything works together so that they can better participate.

While working on this artifact enhancement, I was able to 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 second artifact covered in the code review video is the original artifact for this enhancement, where I discussed the original source code, module by module, and assessed my own previous work. I’ve created two flowcharts for this artifact enhancement, one of which is a depiction of the functionality of the artifact before the enhancements, and the other a depiction of its functionality after the enhancements where made. I also wrote this descriptive narrative which helps explain my artifact, its enhancements, and my intent behind the enhancements.

While working on this artifact enhancement, I was able to 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. To solve the problem of sorting the bids, as well as show competence in creating functional algorithms, I created two vector sorting algorithms. This, however, was met with a trade-off in the form of having to add an extra data structure other than the hash table, which meant that the bids needed to be loaded by the program a second time. This led me to add two more menu options other than the two for the vector sorts, one for loading bids to the new vector and one for displaying the bids from this new vector, along with changing the menu labels to help the user understand the difference. This solution allowed me to manage this trade-off and let the user decide when to load the bids a second time while also allowing me to add two new functional sorting algorithms and meet my artifact enhancement goals.

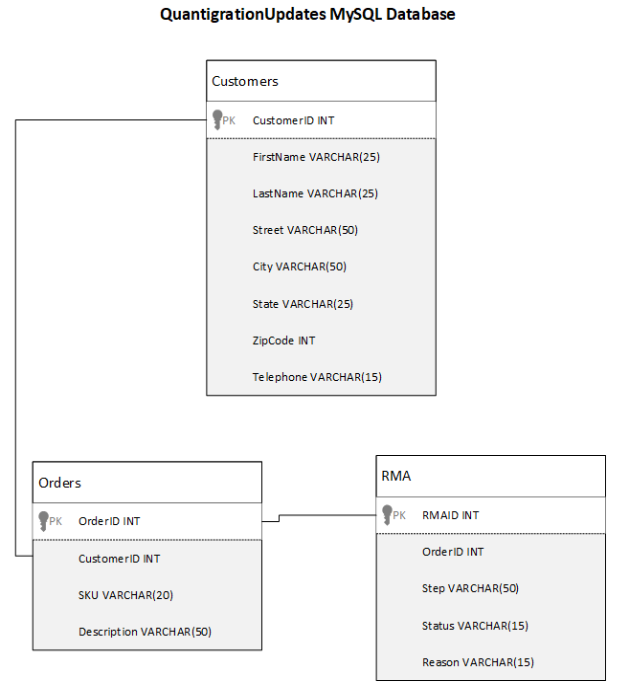
While working on this artifact enhancement, I was able to develop a security mindset that anticipates adversarial exploits in software architecture and designs to expose potential vulnerabilities, mitigate design flaws, and ensure privacy and enhanced security of data and resources. This was done by adding exception handling in the artifact enhancement’s source in key areas, which are in the modules that contain the code used to load bid data. These let the user know if there is an issue with the program when loading the bids. As the csv file containing bid data could, potentially, hold thousands of “bid id’s”, “titles”, “funds”, and “amounts”, which could be filled with the wrong type of data or come from a corrupted file, this seemed to be the area of the most vulnerability. I also made sure that the code ran without any errors.

**Reflection:**

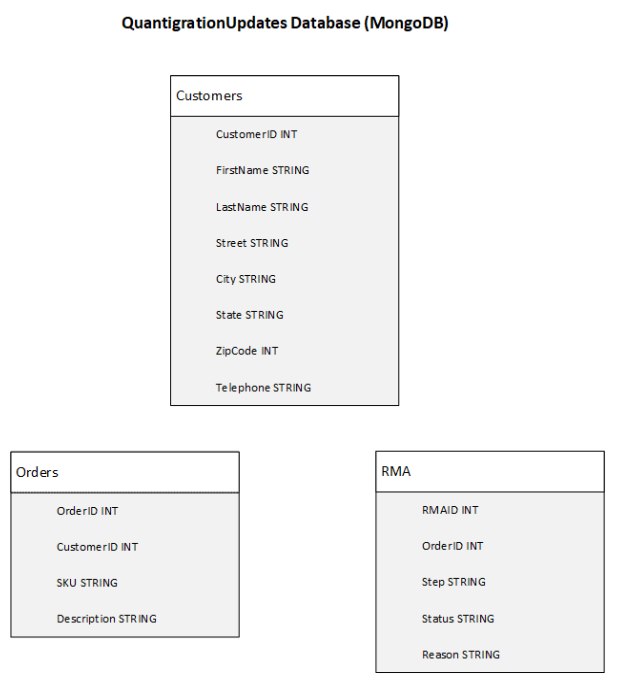
As part of the process of completing my enhancements to this artifact, I had to look over the program thoroughly to understand how it works, what enhancements where possible without changing the whole program, and, when those enhancements were decided on, what programming decisions would be necessary to complete those enhancements. This was made easier by my familiarity with C++, which is the programming language that I am most familiar with. While going through this process of reviewing the projects code, I noticed that the Hash Table data structure doesn’t easily lend itself to vector sorting, so I added an additional vector. This became central to many of my other programming decisions, which includes four new menu options and an extra algorithm for loading the new vector with the csv file data. I don’t believe that a set of menu options like this would normally take place. However, I wanted to display multiple forms of data structures and algorithms while using the same source of data and the same theme of the program, which was storing, adding to, removing, and looking up bids. So, to have a coherent program with one main theme while still demonstrating multiple data structures, I added a second set of options that were labeled differently for the users benefit.

1. **Artifact Three Narrative:**

**QuantigrationUpadates MySQL Database UML Diagram:**



**QunatigrationUpdates MongoDB Database UML Diagram:**



**Description of the Artifact:**

The original project from which the artifact enhancement was taken was called “DAD 220 Project One”, which I created for an assignment back in August of 2020. The goal of this project was to create a database called “QuantigrationUpdates” with three tables called “Customers”, “Orders”, and “RMA”, query their records, update their records, delete their records, and download the records from the “Orders” table onto a csv file using MySQL. My enhancement to this artifact was creating a similar database, also called “QuantigrationUpdates”, with three collections that contain the same fields, use similar queries, updates, deletions, and a download while using MongoDB commands on a MongoDB shell. My enhancement also included adding two levels of user accounts to provide extra security to the database. This enhancement was completed in the last week of July 2022.

**Why This Artifact Was Chosen:**

I chose this artifact because it allowed me to show a range of CRUD skills with Databases using MongoDB, a popular NoSQL programming language. By using MongoDB commands, I was able to create a database for a fictional business with three collections, each containing multiple fields of data, as well as create a view of one of the collections. I was also able to add records to each database using correct commands, display records with queries, update specific records using query and update commands, and delete records using query and delete commands. I was also able to export a collection to a csv file with a MongoDB command. I was also able to go beyond the functionality of the original project by adding password secured user accounts.

**Course Objectives:**

While working on this artifact enhancement, I was able to employ strategies for building collaborative environments that enable diverse audiences to support organizational decision making in the field of computer science. My enhanced artifact zipped file is open for anyone to download and view for themselves from my public GitHub page. This file contains the three csv files used in the artifact’s enhancement, as well as a txt file containing all the MongoDB commands used, so that anyone who wishes to reconstruct the database themselves and conduct the same or different queries can do so. Also, if someone wishing to work with these MongoDB commands needs more information about what the commands are meant to accomplish, there is a word document with screenshots of the commands being used with written explanations for them located in the same zipped file. Also, the public GitHub page that this artifact enhancement can be found on grants others the opportunities to ask questions and propose suggestions pertaining to this and all the other projects and files located on the page.

While working on this artifact enhancement, I was able to design, develop, and deliver professional-quality oral, written, and visual communications that are coherent, technically sound, and appropriately adapted to specific audiences and contexts. In terms of oral communication for this artifact, this was the last artifact covered in the code review video, where I went over screenshots of each of the MySQL commands used and discussed their purpose and potential areas of change. In terms of written communication, the zipped file for this artifact contains a short readme describing the other five files that are contained in the zipped file, one of which is a word document, which contains screenshots of the MongoDB commands used and their output, along with explanations for the commands. I’ve also written this accompanying narrative to describe the original artifact, explain my enhancements, and justify why the enhancements where made. In terms of visual communication, I constructed two UML diagrams, on for the original MySQL QunatigrationUpdates database, and the other for the enhanced MongoDB database so that others can visualize the structure of these databases.

While working on this artifact enhancement, I was able to 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. Some of the well-founded techniques and skills that were used during this enhancement was a full display of CRUD operations, as I created a database using MongoDB commands, queried the collections of that database to find specific information, updated and added several fields in that database, and deleted some fields. The tools that I used during this artifact enhancement include a MongoDB Enterprise Server 6.0 and a MongoDB Shell, which I downloaded to my local device, and a Microsoft command prompt that was used to access the MongoDB Shell and perform MongoDB commands.

While working on this artifact enhancement, I was able to 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 demonstrated this by creating two password secured user accounts, one of which was an administrative account that was granted read/write access to all databases in the server, and the other was a user account that was granted read/write access to the QuantigrationUpdates database. I would also note that, after running each MongoDB command, a message displaying either a warning for errors or the task that was completed by the command is displayed. You can see these message outputs after the commands I used in the word document containing screenshots marked “CS 499 Artifact Three”, which is located in this artifact enhancement’s zipped file.

**Reflection:**

Since I’d already learned how to use MongoDB commands previously, much of the new knowledge I gained from working on this enhancement has been while adding a local Mongo shell, Mongo Server, and other necessary tools to my computer. This was also the main source of the challenges that I had in completing this project, as I have never downloaded and used MongoDB tools on my local device before. However, after seeking help from tech support, the internet, and some trial and error, I was able to use my own working MongoDB shell and server to complete this artifact enhancement.

1. **References:**

Grupman, C. (2022). 8 Data Analyst Skills Employers Need to See in 2022. *Dataquest.* <https://www.dataquest.io/blog/data-analyst-skills/>

Berkely Extension. (2022). 11 Data Scientist Skills Employers Want to See in 2022. *Berkely Extension.* <https://bootcamp.berkeley.edu/blog/data-scientist-skills/>