The original artifact that I am using for all of these enhancements is a project from April 2024 that was originally developed to read a text file of grocery items purchased, track the frequency of purchases, and export the frequency data to a text file.

I selected this item because I wanted to work on a single project for all three of the improvement areas. I think that a single project makes more sense for the purpose of showcasing my abilities in an interview context rather than multiple smaller projects. During an interview I may not have time to discuss five different projects and how my work on them proves my capabilities so I figured that a single project would help me showcase my abilities in a more concise manner. Another reason that I chose this artifact was that it was a pretty simple project to begin with, and I wanted something that would allow me to really explore and expand upon the initial development.

The specific components of this artifact that relate directly to the category of Algorithms and Data Structures are the two sorting algorithms that I implemented. The two algorithms are both selection sort algorithms that sort the data by frequency and by name. The selection sort algorithm has a time complexity of O(n^2) which makes it not viable for large lists, but due to the smaller list size of this application’s data, it should not have any issues. If a very large amount of items were being sorted and the client wanted better performance, a quick sort or merge sort algorithm could be implemented to increase the average time complexity. But, since the list size is relatively small, the simple implementation of the selection sort is sufficient. Also, I implemented an Item class that houses the data of each item while it is being manipulated outside of the database. So, in a maybe unconventional definition of a data structure, I used these objects to provide a more structured interface for interacting with the data that is stored on the program level.

I believe that the improvements made to this artifact fall in line with the course outcome that I originally identified as my goal to achieve: 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).

When I first was thinking about and designing my improvements to this artifact, I was originally planning on implementing a dictionary to store the data at the program level i.e. outside of the database. However, when I began implementing this, I realized that since I was already providing an abstraction through the item class itself to store the name and frequency of the item, a dictionary was not needed, and I instead just implemented a list of item objects. If I had used a dictionary, I would have been storing the data for each item in a dictionary and within each item object. This allowed for a simpler interaction between functions and data than trying to pull the data to a dictionary also. The implementation of the sorting algorithms also proved to be somewhat time consuming, but a little online research allowed me to solve any issues I had with implementing the algorithms. Other than these, I did not find too many challenges during development as it relates to algorithms and data structures.