**4-2 Milestone Three: Enhancement Two: Algorithms and Data Structure**

Tomas J. Estanislao

Southern New Hampshire University

CS 499: Computer Science Capstone

Instructor Maryann Krupa

August 19, 2024

**4-2 Milestone Three: Enhancement Two: Algorithms and Data Structure**

Part 1:

1. Briefly describe the artifact. What is it? When was it created?

The artifact I am using is the same as the previous Enhancement. With the overall goal of modifying an API to use with a new dataset, I took the opportunity to expand upon my previous enhancement. I modified the data structures of my database schema, along with implementing searching and filtering, to emulate a common API use case.

1. Justify the inclusion of the artifact in your ePortfolio. Why did you select this item? What specific components of the artifact showcase your skills and abilities in algorithms and data structures? How was the artifact improved?

I included this item because I felt it was a meaningful and worthwhile pursuit to build off an already complex artifact. By tying in all three enhancements together, I can go through the process of implementing new features— while also dealing with an expanded complexity from my previous changes. Aside from that, re-designing my lists Schema to incorporate custom data fields was a way to demonstrate my skills as they pertain to data structures. These changes required me to redesign how the data was structured within the Novels[] array, and consequently required me to redesign my existing API routes. I completed the U in CRUD, to emulate an editable custom user reading list. Aside from the data structures changes, I went back to my original public /novels endpoint and implemented a custom searching route to query the master database. This required me to research how MongoDB handles querying, and how to interact with different query parameters in Node/Express/Mongoose.

1. Did you meet the course objectives you planned to meet with this enhancement in Module One? Do you have any updates to your outcome-coverage plans?

I completed most of my goals for my enhancement items. I designed custom data structures that is both customizable and editable, and able to persist in the database. I also used appropriate API and database tools to implement querying— searching by Title/Tag, sorting by Title/Stats, ordering by ascending/descending, and limiting the number of results. Although I continued the practice of protecting my API calls with tokens, I also ran into security concerns such as injection and error handling. In my research on reading query parameters and searching the database, I was surprised that there were many different techniques to validate the user input. I focused on basic whitelisting, restricting the types of fields for input, and default values. Some potential security concerns I could address are validating the inputs to remove special characters and limiting the maximum amount of data in the responses.

1. Reflect on the process of enhancing and modifying the artifact. What did you learn as you were creating it and improving it? What challenges did you face?

I learned that basic changes to the structure of data, can significantly change how you can interact with it. Since I updated my user lists schema, I had to deal with nested arrays, and the changes to querying. I could no longer use .findById(), and instead had to use .find() to loop through each element within the novels array— in order to find the correct NovelID to retrieve. These schema changes required me to refactor my previous API routes, not only to access each NovelID, but also to populate the array from the master novels collection. When implementing the querying functionality, I had to practice my debugging and error handling. Not only did I have to research how to define and interact with query parameters, but I also ran into problems with results of my query. I wasn’t sure why it wasn’t working, and after some researching and debugging, I realized that my data wasn’t stored in the database properly. As an example, when sorting by page count, I was expecting the resulting data to be ordered something like: [10000, 100, 10, 1]. Instead, I received something like this: [9, 519, 375, 37, 368, 1492]. This alerted me to investigate how my data was being stored in the database, and question what was happening. After this I confirmed that the basic functionality of my /novels/search endpoint was working, but that I need to investigate how to clean up my data for my third enhancement: Databases.

I completed my enhancements on my previous course repository, due to having the project libraries already installed and set up. All changes are within the **“app\_api**” directory in the “**Enhance2”** branch. I also copied the file folders and published onto my ePortfolio github repo.

Previous Course Repository: <https://github.com/Testanislao/cs465-fullstack/tree/Enhance2>

ePortfolio Repository: <https://github.com/Testanis396/Testanis396.github.io/tree/main>

Below are screenshots of my running enhancement:

**Add new novel**

A screenshot of a computer screen

Description automatically generated

**Get user list**

A screenshot of a computer screen

Description automatically generated

**Modify novel**

A screenshot of a computer screen

Description automatically generated

**Get user list**

**A screenshot of a computer screen

Description automatically generated**

**Delete novel**

**A screenshot of a computer program

Description automatically generated**

**Get user list**

**A screenshot of a computer program

Description automatically generated**

**Search**

A screenshot of a computer

Description automatically generated

