Milestone 2 Enhancement

William P. Giittinger

william.giittinger@snhu.edu

Southern New Hampshire University

**Zoo Authentication Website Login Artifact**

The project files that were submitted contain the groundwork for a zoo login authentication system as a website. The general goal of this first submission was to have an express site that allowed a user to register for the website, login, and have his/her session saved. This is a common functionality that is required by many websites and is typically the most fundamental security layer. The application was originally written in Java as a simple application and was converted into JavaScript. The website also implements page routing, controllers which handle the logic for those pages, and use of the EJS view engine for rendering HTML and placing basic logic into the HTML (e.g. referencing user specific name once logged in).

This website showcases a variety of skills that I have developed throughout my education. Firstly, as mentioned, I was able to transfer logic written in one language and convert it into another language. On top of this, the logic was converted into the needs of a website which includes multiple different pages and systems to work. The program has moved from being merely a NetBeans Java program written with a couple included libraries for parsing files, into a Visual Studio Code JavaScript program using Express.js, Node.js, HTML, EJS view engine, Passport, Express extensions (session and flash), and many more tools. It also showcases many improvements in code writing including syntax, style, convention, etc. Admittedly, there are a few areas that could be improved with more direct commenting, but the goal was to comment only as necessary for now, as some temporary systems will change rapidly in the following enhancements with the inclusion of more pages, routes, logic, and essentially a functioning backend with a MongoDB. This artifact belongs in my ePortfolio to showcase that I am capable of understanding, implementing, and iterating the implementation of a secure login system that also enables session support.

As for this artifact, the enhancements made go beyond what was originally planned. The primary reason for this was the tools and libraries needed for this program to work as intended were underestimated during the initial planning phase. Getting each library, function, route, etc. working in unison was difficult as this was the first time that I have ever used many of those tools. Even the libraries and functions I have used before were challenging to integrate in new ways that I have not been exposed to previously. Currently, there are no plans to remove any planned updates from the upcoming enhancements. On the contrary, there were additions made that will need to be accounted for moving forward. Specifically, the ability to register new users to the system and the ability to logout of a session and start over.

There are a couple design decisions that were made for this project to make future improvements much easier. The first of which was to have the user’s data (id, name, email, password) stored in the application memory rather than a separate file or database. The database will be implemented for this website in a later artifact, and there are no easy ways of implementing plain text file management with JavaScript without extensive dedicated methods. Thus, in order to make the transition into storing the data on the MongoDB, the methods and in-app data storage were designed to specifically mimic the general structure of the data as they are planned to appear in the MongoDB. There is also minimal use of styling efforts regarding pages and the contents they display. The primary reason for this is the use of a view-engine which was not originally intended to be needed for the project but ended up as a necessary inclusion. The visual presentation will be improved in the upcoming enhancement which is more about handling data structures and showcasing the effects of algorithms.

Developing this first enhancement was admittedly quite difficult. Many of these systems I have only been briefly exposed to and in a very structured, step-by-step, manner. Here I was able to take what I have learned previously, and implement a totally different system, using the best practices that I have learned so far. The use of new and different libraries and frameworks was also a large learning curve. While each step was somewhat painstaking, I learned a lot about how each tool used in this website is designed to work together and have a greater appreciation for each’s role in the site. For example, the login system went through at least 3 major reconstructions as I better understood how to use each system and the shortcomings of some of them. As mentioned previously, there is no easy way to parse and manage plain text files with JavaScript, so after imperfectly writing long and complicated methods to do so, I learned the functionality I desired was both imperfect and programmatically too complicated. Instead, I moved the user information data into the app’s memory for now. While this isn’t a perfect solution by any means, and there were issues with language support (e.g. JavaScript’s .find() function wasn’t working as intended when accessing objects), it will save me time and effort in future improvements and changes.