Ben Robbins

09/22/2024 – Updated 10/16/2024

Milestone Two

**Enhancement One: Software Design & Engineering**

For my enhancement in software design and engineering, I selected my Travlr Getaways full-stack web application project for my artifact. I originally completed this project during CS-465 Full Stack Development using the MEAN stack. The purpose of the initial project was to convert a static html page into a dynamic webpage and create an administrative single-page application. Since only one page was originally created, I chose this project as my artifact as I would be able to rapidly increase the scale and complexity of the project by converting the remaining 6 pages to dynamic webpages.

For this specific category of enhancement, I focused on the front-end of the application. My enhancement plan was to convert the static html pages to dynamic pages and utilize MVC routing architecture. Additionally, I would replace the static data with dynamic loops using Handlebars and implement conditional formatting in the header and footer.

To implement my planned enhancements, I used the MVC routing architecture as a guide during development. For each page, I first converted the static html file into a view. Then, for pages with static data that I would load dynamically with handlebars, I created a temporary JSON data file. However, I didn’t remove the static data from the view at this point. I then created a controller file for each page. For each page with data, I created functions to read the data from the JSON file and passed it to the rendered view. Next, I created a route file for each page that implemented the controller. With the basic MVC architecture implemented, I updated the app.js file to include the router files and implement the URLs for each page. To finish my enhancements, I then updated the views to utilize Handlebars to dynamically load the data. I replaced the static data with a loop that referenced the JSON data rather than hardcoded information. Additionally, I replaced the header and footer with partials on each page. I also updated the header and footer to reference the new page URLs and added conditional formatting. I added the conditional formatting by passing another Boolean value through the render function in each controller.

One interesting thing I learned was when I implemented the conditional formatting to the header and footer. At first, I thought I needed to pass a Boolean value for each page for the header and footer to correctly indicate which page was being viewed. However, I was able to improve this by only passing the true value that was required. Each time the pages load, if a value is not passed, the conditional formatting in the header and footer partials defaults to false. One challenge that I faced during this process was not implementing the back-end logic for the database schemas and APIs first. As I got a head start on further enhancements, I found it much easier to continue improving the application once this logic was implemented.

By utilizing tools such as Handlebars in addition to the MEAN stack to implement conditional formatting, I corrected an existing issue within the application and further improved the user’s experience. The successful implementation of these enhancements using innovative tools demonstrates my skills in full-stack development, MVC routing, and UX design while also showcasing my ability to design and develop a collaborative environment. In addition, the clear and concise comments, file structure, and naming conventions I used meet industry standards for documentation and cross-functional communication.