**Module Six Journal**

Lesley Potts-Langdon

Southern New Hampshire University

CS 465: Full Stack Development 1

Professor John Watson

May 10, 2020

Write a journal entry that summarizes the architecture, functionality, and testing of an SPA for the client-side administration of the web-server. Specifically, summarize the following in your journal entry:

* How is the Angular project structure different from that of the Express HTML customer-facing page?

The significant structural difference between Angular and Express is which side carries the larger load. When we use express for our whole application there is a lot of back and forth communication between the server and the client-side browser. Angular shifts a lot of the load onto the browser by loading most of what is needed as a single page application (SPA) at start up. The only communication between client and server with Angular is to send and receive data.

* What are some advantages and disadvantages of the SPA functionality? What additional functionality is provided by an SPA compared to a simple web application interaction?

I don’t look at it as advantages and disadvantages, it is more about selecting the components that best suit your needs. One of the most noticeable differences from the user’s perspective is load time. A SPA loads more in the beginning and therefore takes longer. If you have a large application that people are not familiar with that could be a real problem. If you are trying to drive traffic to your site to introduce your product and the site takes a long time to load, you could loose people before they even see what you have to offer. If you have a smaller application with limited server capacity you would benefit from shifting the load to the user’s browser and they would likely not notice the delay in loading.

* What is the process of testing to make sure the SPA is working with the API to GET and PUT data in the database? What are some errors you ran into or what are some errors you could expect to run into?

We used Robo 3T, the terminal and Postman to test the POST, GET and PUT methods. I’m including POST because I think it was an important lesson to remember. When we first tried to add a new trip, we could see in the terminal that a POST was sent but we got a 404 not found error because we had not updated the back end to receive it. Once we updated the back end, the POST showed in the terminal and we went into Robo 3T to see that the new trip data was included in the MongoDB database. When we call or post something from within the site, we can see the GET and POST in the terminal. If we want to test, particularly POST, without actually changing anything, we can use Postman. If I go into Postman with the front and back ends both running and send POST to localhost:300/api/trips I get a 400 Bad Request error because it is looking for trip data, but it still shows that communication is happening.

* What questions do you still have that will help you with SPAs in future builds?

As far as actual questions, I don’t have any currently. What I am working on is gaining a better grasp of the times it is appropriate to use a SPA or not. As I visit different websites, I am trying to figure out what methodology they are using. I’m doing a lot more looking at developer tools while I am online.