Chris Tapia

Prof. Buksbaum

CS 230

June 5, 2021

Journal: Module Four

The way the client-server pattern is setup, makes it easy when integrating the game application to other operating systems. It effectively separates the main code of the application from the set of rules that are in place for communicating to and from the server. This allows the application to maintain its set of rules while also changing the code when needed while not affecting the client server code. This made the application very easy to work with without having to worry about the other code that’s used to integrate on the selected operating system.

When viewing the application from the server side, it’s setup so it can deliver the information effectively in a very clean and easy to view interface. It’s done so that the client can make different requests like GET and POST to achieve different outcomes in different parts of the application. The GET requests are issues when the client accesses the application in the browser and types in a new URL. While POST requests are made when the user enters in their login information from the client side. These types of requests can also be used to add more users through the app interface. For example, an admin can login and click a button to “add a user”. That kind of function would be a POST request. There can also be a function to delete or remove users which would be examples of a DELETE request.

When implementing more features, it’s important to think of what is relevant to the game while maintaining its simplicity. A feature to save progress could be quite useful. Applying a feature like that would require the use of REST features like POST to add progress information to a database. Another notable feature would be to add other existing users to a personal friends list. This could be done by linking existing users together using ERD database design. This also raises the question of making the game cross-platform compatible. Bringing the application to other clients like Xbox and PS4 would require some changes in code but should allow for the use of the existing client-server pattern to still be just as effective. That’s the beauty of the client server pattern, is that the pattern allows is to keep its same uniform design, while allowing you to add different code to other parts of the program without having to worry about affecting the this kind of design. Having a design like this in place in a program further increases its versatility and its reusability. It allows for other developers and software engineers to expand on what has already been created. This makes the program future-proof and easier to debug. This is because the client server side pattern is kept separate from the rest of the code, which allows others testers and developers alike to easily find issues and problems as well as explore newer ways to add requested features when it gets taken to new clients and operating systems. Designing programs around a REST API style can be so beneficial to any application, and should never be overlooked.