Dylan Dunagan

CS-230

28 July 2024

Module 4 Journal

               The client-server pattern can be used to satisfy software requirements and efficiently solve a problem. Because the client does not have the same resources as the server, the client-server pattern will bridge that gap. The client requests data from the server which in turn returns what the client was requesting. For example, this week, in Module 4, the class worked on creating a REST-style API for the Gaming Room application. This API allows clients to input their username and password to access their account. The client requests their account or game instance by inputting their unique data, and the server returns either what that client requested, or some sort of error message when the client’s data is incorrect or non-existent. The client-server pattern works extremely well for web-based game applications because it allows multiple platforms to be used. The correct response should come from the server so long as the client request can access it.

               To allow for all three clients to be able to use the website, the developers must declare clear roles for each client. First, the principal object is when the client provides their credentials, username, and password, as a request into their account. The application then authenticates the user’s information via the username a password. The authorizer then matches that user with their assigned role. For example, the “ADMIN” and the “USER” roles. These will determine who is accessing the website and what that person accessing the website can and cannot do.

The developers could add a way to cache all the usernames and passwords as users create them so that the developers, or the Gaming Room team, would not have to manually add each client that wants to use the program. Additionally, there would need to be a way to streamline assigning roles to the clients. The easiest way would probably be to automatically assign each client as a general user and have them request to be upgraded to admin status as it is required by the application’s team. This would allow only authorized clients to be able to make changes to the website.

               The Gaming Room wants to Draw It or Lose It on multiple platforms and as the game grows, it is reasonable to assume that it will expand from mobile devices and computers to console systems like the Xbox, Switch, and the Playstation. If the server can be accessed through those systems, there should not be a need for many changes on the server. There would just need to be a new way for those systems to request their accounts using the username and password method. Usually, console systems, have accounts that automatically log in when the console is turned on, therefore they could access the game by linking that console’s account to the Draw It or Lose it account. This might require a separate list on the server for those accounts, that way the correct data can be returned to the client depending on the type of device being used.