Journal 4-4: Software Application Requirements

Dhiraj Gurung

Southern New Hampshire University

CS230

Barbara Hecker

January 28, 2025

**Software Application Requirements**

**Client-Server Pattern:**

The client-server pattern is a web-based architecture comprised of two parts the client and the servers. The clients are the computers or programs that send requests for services or data, and the server consists of the computers and programs that handle these data requests. In a web-based game environment, this model allows for effective communication between the user’s computer serves as the client, and sends data requests to the game’s servers. This interaction occurs behind the scenes and is what allows for different actions and reactions to occur within the game in real time. The client-server model allows for a server to speak with multiple clients at once, which is also what makes a multiplayer gaming experience possible. One benefit of this model is that it allows for a distributed workload meaning the client is not solely responsible for computing everything in the game there is a centralized server that handles different requests which eases the burden on the user’s computer.

**Server Side:**

The server communicates with the client by handling the requests that are being sent. The process first begins with the client sending a request that specifies and includes the following: the endpoint, the method, and the payload. Once the server receives these requests it will respond with a status code to verify if the request was handled successfully or not. Additionally, if the request was successful then the server will also include any data that was requested in its response to the client.

**Client Side:**

When creating an application that needs to be accessible on multiple environments you have to make sure that the server is capable of communicating with the client. This means that you have to write the code in such a way that it can handle sending requests to all the various browsers like Chrome and Edge.

To add more users to the database from the server side you can manually input the information of new users as was done with the first 3 users. In the game player management application located in module 4, you can simply go to the GameUserDB.java file and update the hash map using the same code that was used to add the first 3 users. You can simply copy and paste the code’s format but just change the information and key value to match that of the new users being created. In order to do this from the user side we would need to code new functionality into the application. This new functionality would be a new page that would allow new users to sign up and submit their information to the server. Then the server would need to add the new user information to the database.

The gaming application right now is pretty limited but we can add new social features like adding other users as friends and then having some way to communicate with those friends. If these features are implemented then you should also add features like inviting your friends to your team so that you can play the game together with them. Another feature could be adding a global leaderboard and a leaderboard among friends as well. These features would improve the social aspect and may allow users to interact more with the game.

If we needed to host the application for new clients on the Xbox and PS4 consoles we may need to adjust the code to ensure that the server can handle the requests being made from these two platforms. The Xbox and PS4 platforms have different compatibility requirements than a web-based application; therefore, the current code may have to be rewritten to still have the same functionality but now just capable of communicating with the consoles.