Jesel Reyes

CS-230-11520

Professor Elhayboubi

9 June 2024

4-4 Journal: Software Application Requirements

The client-server pattern can be seen in merely all real-world applications such as whenever we connect to our financial institution’s web portal to check our accounts to connecting to a certain YouTube video. The client-server pattern is truly applicable to a multitude of scenarios, one of them being a game application. When it comes to developing game applications that are universal to all consoles and platforms, there needs to be a way in which the clients, those being the PCs or gaming consoles, to communicate to the server. That way being through the client-server pattern, where the client sends a request to the server. The request usually entailing of a header and body; The header would typically have an authorization header that has information about who is trying to request information and the body containing information about what is needed from the server. The server would then process this request through a method such as a PUT, GET, POST to fulfill the request. Then, the server would send back a server code that lets the client know whether the request has been fulfilled. This pattern can be an optimal way to have multiple clients and one server because it decouples the client and server components and allowing there to be a diverse amount of client environments.

In the server side of the pattern, the server oversees completing incoming requests from the client. The incoming requests from the clients in a REST API would typically come in JSON format. The request would have a header and body that would be processed by an HTTP method such as PUT, GET, POST, PATCH, depending on the type of request. Then, the HTTP method would interact with the backend of the server to retrieve the information needed to the client in JSON format. After the HTTP method completes the request, it will send back a server code to the client letting them know whether the request has been completed or not. Additionally, a message could also be sent to the client to explicitly tell them what the issue with their request is or if their request has been successfully completed. This way if there were to be an issue with a request, the client would immediately know what the issue is instead of them having to debug the issue.

On the client side, the developers would oversee ensuring the application is able to handle the requests incoming and receiving across multiple platforms such as mobile devices, desktops, and gaming consoles. To do such a thing, there will need to be a lot of thought put into the design interface as well as the implementation. The user interface would be intuitively designed for the user to be able to navigate it without any issues. The technologies usually used for graphical interface development specifically within gaming development are Unity and Unreal engine. These technologies have been developed with game development in mind. However, there would also need to be an SDK developed if there are unique consoles connecting to the game to establish a connection to the server. If in the future, there were to be an influx of new users, the developers would also need to oversee a user database and management to ensure that user credentials are safe and are following industry security measures. If the Gaming Room were to host an application on an additional client, there would need to be a review of what the new clients are and if they are compatible with the application. If there are not compatible, there would need to be an implementation of an SDK or optimization of graphics to ensure the client has a seamless experience whenever playing the gaming application.

As the game begins to mature and prosper, there might be new features to be implemented into the gaming application. One of them could be such as in-app purchases or user customization. Additionally, to cater to competitive gamers, there could also be gaming tournaments and multiplayer functionality. All in all, developing and designing a gaming application for a multitude of gaming platforms can be a great way to cater to a wide variety of audiences.