Bagayas, Kevin

CS 230 Module Four Journal

7/28/2024

The client-server model is a pattern consisting of clients and servers. A client is the one requesting services or resources to servers who can provide the needed outputs. Clients can be in different forms like browsers, software or mobile devices. Server examples are proxy, mail, and print servers. One example for a client-server pattern would be when you want to watch videos in YouTube, the device would request the information from the YouTube servers. This is important by segregating the core functions of each element like client’s functions can be the hardware/software that has the interface to input request to the server side.

For the web-based game application scenario, it will work in multiple platforms since the client side will handle mostly the interface, user profiles and other input attributes. The server side will manage the data requests, storage and communication capabilities. The key factors would be having the proper client-server system or protocol that would not be interrupted or accessed by outside threats. Security between communication towards the servers should be secured and safe.

In the server side, using REST API style would be ideal considering it is commonly used for most mobile and web applications to communicate with the server. The server side communicates with the use of APIs and every request and response is independent wherein nothing is stored between each side. They also communicate through HTTP requests and responses. It can also provide URLs to the client side where they can put their requests to.

In the client side, the flexibility of its inputs can be in multiple platforms like web and mobile applications. It is vital that these platforms can communicate seamlessly with the servers via REST APIs. The client side should have the proper user interface, easy to navigate, and the ability to expand or improve its scalability. For example, would be playing with cross platform video game consoles (PS, Xbox, Switch or desktop) to participate in a multiplayer game. It is important that it is developed with compatibility on mind even with different devices or browsers.

In order to add more users to the database would be creating multiple user profiles to accommodate more players in the game. This would also entail other customization options in terms of the game preferences, difficulty and gameplay. Other features that I can incorporate with the game would be voice communication between players, ability to stream gameplay, in-game purchases to boost player inventory, and of course the ability to continue saved gameplay in different platforms.

If the Gaming Room asked me to develop the application in the 4th and 5th client, it would utilize the existing features of the consoles like game data storage, cloud storage and multiplayer capabilities. For this scenario, the client hardware would not matter, it would be the APIs or the communication vehicle that would play a major conduit towards the servers. This would include the ability to send requests and accept responses securely even on multiple client platforms.