4-4 Journal: Software Application Requirements

The client-server architectural pattern is a very commonly known pattern, especially with network type architectures. It contains two components, client and server components. The server will listen to requests from clients and provides the prompted services. This will help enhance the function of being able to run on multiple operating platforms, as the server can be based off of Linux for example, but a Mac computer can reach out to that server as a client, ask for a request or service, and the Linux OS based server will return the request.

On the server side, the server provides clear communication to the client with REST API style by prompting an interface for the user to interact with to answer the requests made by the client. REST API stands for Representational state transfer, and is a software architectural style that defines a set of constraints used for creating web services. When a web service conforms to the REST API style, it is considered “RESTful”. One of the criteria for REST API is to be stateless, which allows the for client-server communication to not be constrained to no client requests being store on the server in between requests. By being stateless, RESTful web services become fast, reliable, and have the ability to grow.

On the client side, the client sends out requests for the server to fulfil, whether be transferred to a database to be interacted with or fulfill the information in the request, and send it right back to the client. For the application to be used on all three clients, there may be a need to have more than one instance of a game at one point in time. If need-be, developers may wish to add features like a save feature, to store data in a database for later retrieval, so a user may continue a game later. To accommodate a fourth or fifth client, the game may need to be expanded to match the protocols that those clients request.