Andrew Bolton

CS 230

5-30-21

The client server pattern is a network architecture that can consists of a server or may servers and its clients. These servers are powerful and capable of handling the requests of each client simultaneously. The one benefit of using a client server pattern is the ability to divide the workload and processes, this is very important when handling a big and small programs needs. Since the client server pattern dose most of the processing on the server side, this allows for the efficiency on the client’s side by requiring less processor power and the ability to run the client on a wider variety of hardware.

Server:

While using the REST API the client/server can be setup independent of each other which can be great for cyber security and vulnerability protection. Now as long as they are using the same format of communication, they will be able to work together. The server communicates with the client by sending packets to the request that the client has made. These request can be anything from a path to resources toa HTTP. JSON is another important form of encryptrd communication between the server and client. JOSON is a encrypted that adds an extra security to communication to and from the server’s and client’s. Rest api untilizes JOSON in their requests. With another notable point rest systems are stateless which means that the sever does not know what state the client is in. With the server and client independent of each other they can receive any message sent to it. Since rest systems are able to interact through a standard operation, they do not need to rely on interfaces with resulting in a more reliable and quick performance.

Client:

Now on the client side communicates with the server by sending a request to the server. To use multiple clients, the developers need to introduce proper load balancing with the amount of data sent back and forth. When you start adding graphics rendering and heavy processing load should be done on the server since it has way more power than the client and to ensure the client will not be overloaded.

For security purposes using HTTPS to encrypt the game as well as the server to client connection is very important to protect the user information but in some other cases a vpn might be called for. Another important security feature is a way to authenticating user input to make sure the user and client matchup and restrict access to client when bad information is provided.

There should be a backend terminal that allows for easy user addition to the database. Depending on the server side, one server should be able to handle about 7 to 8 clients “with proper load balancing”. This is very reliable on the server bandwidth but most high end severs will be able to handle multiple clients. Having to many clients can cause severe performance issues resulting in lag and even loss connection. With having to many clients on at once will create stress on the machine and shorting the life of the hardware. So it is very important that there is away to measures the load on the server to keep it from overloading. BY putting a 8 client cap on the program would be the best way to keep the server from overloading.