Long Vu
Marcilino Lamiy
Kristen Camarena
Parish Gutierrez
Daniel Corona
CPSC 471-02
12/2/2023

Design Document

For our project, we decided to code in the language Python because we did feel the most comfortable with that language. We had the client ‘cli.py’ and the server ‘serv.py’, in both of them we accessed the Python standard library and used the module ‘socket’. The ‘socket’ module helps provide low-level networking operations that allow us to create and manage network sockets.

The client script module is used in a couple of different ways. It is used to create a TCP socket, as you can see on line 27 of the code. Then on line 30 that is where the client connects to the server using the ‘connect’ method. There is the start of data transmission on line 43 where the client sends and receives data. For the next couple of lines, the process continues.

The server script module is also used in a couple of ways as well in ‘serv.py’. On line 12 that is where the server creates a socket for the incoming connections. On lines 13-14 that is where the server binds the socket to a specific address and port, then listens for any incoming connections. On line 35 is where the server will accept incoming connections by using the ‘accept’ method. We also implemented data transmission where the server sends and receives data similar to the client using ‘send’, ‘recv’, and ‘sendall’.

The interaction between server and client is pretty self-explanatory. It first starts with the server waiting on incoming connections, the client will then initiate a connection to the server. Once connection is established then the server and client can send and receive data and commands. The client would request operations like ‘get’, ‘put’, and ‘ls’. The server responds to the client by sending file data, sizes, or lists. This will continue until the user decides to quit.