Blaine McMahon, Jacob Sword, Nicholas Yameen

Phase 2

2/14/19

**GUI.py**

**UDPserver.py**

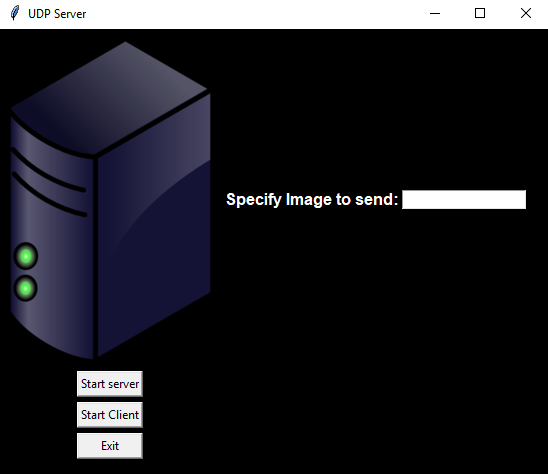
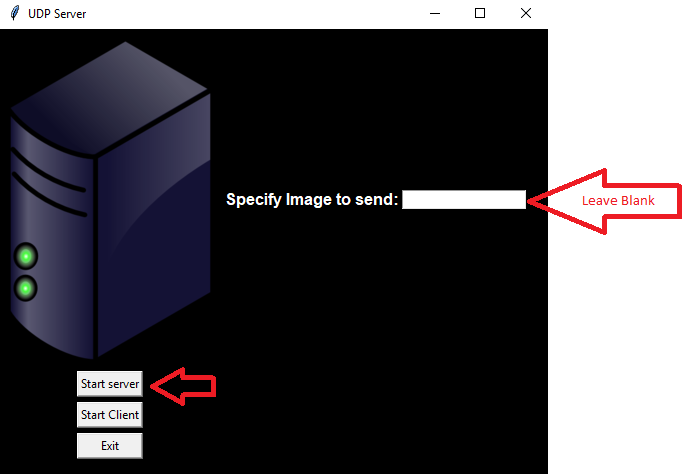
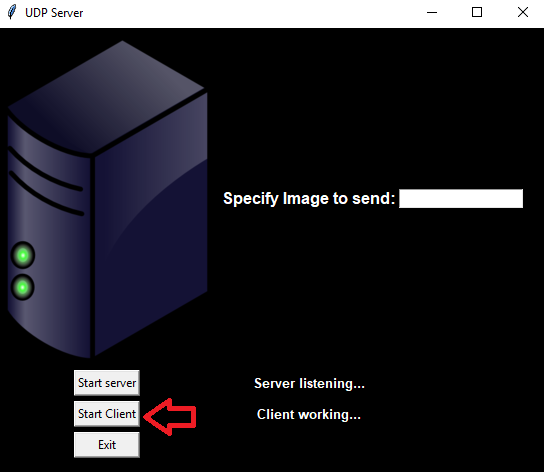
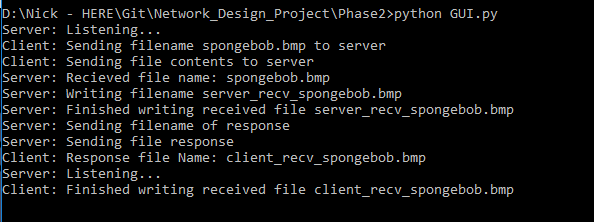
**UDPclient.py**

**rtd.py**

In the file, there are six functions defined. The server and client will use these functions to send RTD 1.0 over a socket. The sender will use *rdt\_send()* function. This function will take in a file, endpoint, and the socket. It will make packets using the *make\_pkt()* function by reading 1024 bytes at a time. It will go into a while loop, which will run while there is a packet to be sent. The *udt\_send()­­* function will send the packet to the endpoint over the socket. It returns the number of bytes sent, then make a new packet. It will continue to send packets until the whole file has been sent.

The *rdt\_rcv()* function will receive a packet. It will be called with the filename that is being received, the file, and the socket. It will go into a while True loop where it will first call *extract()* function. *Extract()* will wait for thread to be ready I/O with the select.select() line. If it is not, it will timeout, which means there is no data to receive on the socket. If the thread is ready, it will receive 1024 bytes on the socket and return the packet. In the *rdt\_rcv()* will then call *deliver\_data()* which will take the data and file. This function will write the file. The receive function will continue to extract and write packets until it has reached the end. At that time, the file will be closed and will be broken out of the while loop.

**How to Run**

1. Issue the command “python GUI.py” on the command line. The following GUI will appear.  
   
2. Leave file to send blank to use default or specify path to own picture. Click server  
   
3. Again, leave image blank to use default, or specify image. Then click client.  
   
4. You will see messages on the GUI, as seen in the image above. In the terminal messages will appear of what is happening as seen below.  
   
5. New files will be in the directory called “client\_recv\_{name}” and “server\_recv{name}”
6. Exit to stop the threads.