**Jacob Rook**

**Systems & Networks II**

**Project 1**

**User’s Manual**

**Setup and Compilation:**

1. Download and unzip the submission from eLearning on a Linux box in the multi-platform lab.
2. The includes:
   1. httpClient.c
   2. httpClient.h
   3. httpServer.c
   4. httpServer.h
   5. makefile
   6. UsersManual\_README.docx
   7. Project\_Protocol.txt
   8. text.txt
   9. blank.html
   10. image.png
3. Environment: This project was created using on a combination of the Eclipse and jGrasp IDE. However, the project was mainly tested on a personal Linux VM.
4. Compiling. This program includes a makefile. At the command line in Linux in the folder with the extracted files, type make into the command line. The program produces executables: httpClient and httpServer

**Running the program:** Issue the command ./httpServer to initiate the server program and ./httpClient to initiate the client program. No command line arguments are required or checked, just make sure any files that are of interest to transfer are in the folder with the httpServer executable. Make sure to start the server program before trying to connect the Client with the server.

**NOTE:** I was only able to have the client and server programs communicate when they were on the same host. Not sure why this occurred, but the client was never able to find the server if the server was on the UWF SSH server and the client was on my personal VM or visa versa.

**NOTE:** I was not able to transfer larger files to the client, they were corrupted in some way. So, when the client program tries to open the file received by the server, the file most likely will not open correctly. Small files like text files were transferred correctly.

User Input Server: The server program is user non-interactive. The server program will just handle the client’s requests as the come.

User Input Client: The client program will prompt the user to enter the IP address of the server and the file that the user wants from the server.

**Example of GET method working on small file**





