**CS 436 Project 1**

**Done by: Jamison Coombs, Colton Clark**

1. Run the server program. Use port 18000 at the server.

Client.py

Text

Description automatically generated

Server.py

Text

Description automatically generated

2. Run the client program. The client program displays the menu. The user selects option 1. Then the user enters this path: “attachments/file4.html”. The client program displays another menu asking what HTTP version will be used. The user selects “1.1”. The client establishes a TCP connection with the server with TCP SYN messages. Then the client sends an HTTP GET request to get the file contents. The server responds with “404 File not found”. The client 5 displays this message on the screen. The client closes the TCP connection with FIN messages. At all these steps, both client and server display messages reporting what they have done.

Client.py

Text

Description automatically generated

Server.py

Text

Description automatically generated

3. The client program displays the menu. The user selects 1. Then the user enters this path: “attachments/file1.html”. The client program displays another menu asking what HTTP version will be used. The user selects “1.1”. The client establishes a TCP connection with the server with TCP SYN messages. Then the client sends an HTTP GET request to get the file contents. The server opens the file and enters its contents to the response message with status code “200 OK”. The client displays the status code and the received file contents on the screen. The client stores the file. The client closes the TCP connection with FIN messages. At all these steps, both client and server display messages reporting what they have done.

Client.py

Text

Description automatically generated

Server.py

Text

Description automatically generated

4. The client program displays the menu. The user selects 1. Then the user enters this path: “attachments/file2.html”. The client program displays another menu asking what HTTP version will be used. The user selects “1.1”. The client establishes a TCP connection with the server with TCP SYN messages. Then the client sends an HTTP GET request to get the file contents. The server opens the file and enters its contents to the response message with status code “200 OK”. The server also sets “HTTP\_INCLUDED\_OBJECT\_PATH” to “attachments/file3.html”. The client displays the status code and the received file contents on the screen. The client stores the file. Then the client automatically sends another HTTP GET request for “attachments/file3.html”. The server will send the contents of that file as well. The client displays the status code and the received file contents on the screen. The client stores the file. The client closes the TCP connection with FIN messages. At all these steps, both client and server display messages reporting what they have done.

Client.py

Text

Description automatically generated

Server.py

Text

Description automatically generated

5. The client program displays the menu. The user selects 1. Then the user enters this path: “attachments/file2.html”. The client program displays another menu asking what HTTP version will be used. The user selects “1.0”. The client establishes a TCP connection with the server with TCP SYN messages. Then the client sends an HTTP GET request to get the file contents. The server opens the file and enters its contents to the response message with status code “200 OK”. The server also sets “HTTP\_INCLUDED\_OBJECT\_PATH” to “attachments/file3.html”. The client displays the status code and the received file contents on the screen. The client stores the file. Then the client closes the TCP connection with TCP FIN messages. Then, it establishes a new TCP connection with TCP SYN messages. Then, it sends another HTTP GET request for “attachments/file3.html”. The server will send the contents of that file as well. The client displays the status code and the received file contents on the screen. The client stores the file. The client closes the TCP connection with FIN messages. At all these steps, both client and server display messages reporting what they have done.

Client.py

Text

Description automatically generated

Server.py

Text

Description automatically generated