

CSCI 3010 Spring 2023

Web Server Python Assignment

You will develop a web server that handles one HTTP request at a time. Your web server should accept and parse the HTTP request, get the requested file from the server's file system, create an HTTP response message consisting of the requested file, and then send the response directly to the client. If the requested file is not present in the server, the server should send an HTTP "404 Not Found" message back to the client.

HTML File

Create an HTML file and store this in the same directory as your server program. You can create any HTML file you would like for this assignment, but it should at a minimum contain a body of text.

Server code

- Create the server socket (using TCP) and bind it to the IP address of the server and port number.
- Establish a listening socket to listen for incoming connections from a client.
- Establish a separate connection socket for the client-server.
- The server should then receive a request from the client. This message is the file name of the HTML file that the client is requesting. The HTML file needs to be stored in the same directory as the server script.
- The server will need to extract the requested object from the message and open the file to then read the contents of the requested file.
- The server then sends an HTTP response (the status code) if the file is found and then returns the contents of the file to the client.
- The connection socket is then closed since we are only servicing one request per connection.
- The server should send the appropriate status code to the client if the file is not found and close the connection socket.

* There is nothing in the code shutting down the server program if the connection is successful, so make sure to kill the process manually when not working on the program. *

Client Code

Write your own HTTP client to test your server.

- Your client will connect to the server using a TCP connection. You will need to accept the IP address of the server and port number as input from the user.

- Send a request to the server. This is a message containing the file path of the file the client wants to retrieve. The file path is obtained as input from the user.
- Display the server response and contents of the file as output. The client should be able to receive all the bytes of data sent from the server regardless of the number of bytes of your specific HTML file.
- Close the connection.

What to turn in:

ReadMe

Server code

Client code

HTML file that you used to test your program.