

Sample Design Document

Haofan Wang
CruzID: hwang108

CSE 130, Fall 2019

1. Goal

The goal for Assignment 1 is to implement a simple single-threaded HTTP server. The server will respond to simple GET and PUT commands to read and write (respectively) “files” named by 27-character ASCII names.

2. Assumptions

I’m assuming we need to connect the client and server using a socket. After connected, we can put the file on the server by typing the command. We can also get the file, if the file doesn’t exist, an error message will be printed out.

3. Design

The approach I’m taking is to get the server file descriptor, then attach the socket to the port 8080. While connecting, extract the connection socket, receive the message from the socket, and save the message to buffer. Then extract 3 important information in the buffer. (command, content length, and filename)

If it’s a put command, open the file with all the permissions, write the data into the file descriptor, and send it to the socket.

If it’s a get command, just open the file and send to the client, if there’s an error, determine which error and send the message to the client.

4. Pseudocode

This is the core pseudocode for this program

Procedure httpserver

- Creat socket file descriptor

- Print out an error message if socket failed

- Set the socket port and attach it to the port 8080

- bind(server, address)

- While connecting

 - Get the content length

 - Get the command

 - Get the file name

 - If the length of the file is greater than 27, send bad request 400 to client

 - Else determine if it's a put or get operation

 - If is put

 - If able to access

 - Size = recv(socket, data)

 - access (file)

 - open(file)

 - write(file, data)

 - close(fd)

 - send(socket, buffer)

 - Else

 - open(file) to create a file

 - write(file, date)

 - close(fd)

 - send (socket, buffer)

 - If is get

 - open(file) with read-only

 - If the fd is -1

 - If the error is EACCES

 - Send 403 forbidden

 - If the error is ENOENT

```
        Send 404 not found
    Read the fd and save all the data to the file
    close(fd)
    Send 200 ok
Else if it's not get and put
    Send 500 internal server error
Else
    Send 400 bad request
Close(socket)
End procedure
```