Sample Design Document

Haofan Wang
CruzID: hwang108

CSE 130, Fall 2019

1. Goal

The goal for Assignment 3 is to modify the HTTP server that I already implemented to have one additional feature: caching. Caching means that I'm going to maintain a buffer in my server that contains a subset of the pages. When a request is received, if the requested page is in the cache, then it is read from the cache (if it is a GET request) or updated in the cache (if it is a PUT request). Otherwise, the page is first read from disk into the cache. In the log record of each request, I should indicate whether the page was in the cache at the time the request was received.

2. Assumptions

I'm assuming we need to get rid of the multi-thread from asg2, then make a queue that is able to store array of characters. When the client gets some file, check if this file in the queue(cache), if is in the queue, return that content. When the client puts a file, check if this file in the queue(cache), if it's not in the queue, update the queue.

3. Design

The approach I'm taking is to delete the muti-thread from my asg2, keep the write log function. Then implment queue that can store char arrays. I'll put push, pop, isEmpty, isExist function in the queue.

Then when the client is getting some file from the server, check if the file is in the cache, if is in the cache, take it out from the queue, if not in the cache, take it from the disk. When put file, if the file is exist, update the queue, if not exist, put into the disk.

4. **Pseudocode**

This is the core pseudocode for this program

```
Procedure httpserver
    If -l found in argv
        isLogNeeded = true
    If -c found in argv
        isCacheNeeded = true
    Declare a queue
    Declare a unordered map
    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)
                If file is in the queue
                    Update the queue
                else
                    write(file, data)
                    close(fd)
                    If logIsNeed == 1
                        Write the info into the logfile
                    send(socket, buffer)
            Else
```

```
                open(file) to create a file
                write(file, date)
                close(fd)
                If logIsNeed == 1
                        Write the info into the logfile
                send (socket, buffer)
If is get
        open(file) with read-only
        If the fd is -1
                If the error is EACCES
                        If logIsNeed == 1
                                Write the info into the logfile
                        Send 403 forbidden
                If the error is ENOENT
                        If logIsNeed == 1
                                Write the info into the logfile
                        Send 404 not found
        Read the fd and save all the data to the file
        If cache is needed
                If can't find file in the map
                        If file is in the queue
                                If cache.size is >= 4
                                        Cache.pop
                                        map.earse(the poped file)
                                If cache.size is < 4
                                        Cache.push(file)
                                        map.insert(file)
                                If isLogNeeded = true
                                        Write file was not in the cache
Else if not need cache
        Send the buffer to the server
        If isLogneeded = true
                Write file was in the cache
        close(fd)
        Send 200 ok
Else if it's not get and put
        If logIsNeed == 1
                Write the info into the logfile
        Send 500 internal server error
```

```
    Else
        If logIsNeed == 1
            Write the info into the logfile
        Send 400 bad request
    Close(socket)
End procedure
```