

## Design Document for Assignment 3

William Kudsk

CRUZID: wkudsk

CSE 130, Fall 2019

### 1. Goal:

The goal of this assignment is to build on our http server so that it caches files

### 2. Assumptions:

That the client will send me an end of file message once I am done receiving data for a PUT if no content length is specified.

The server is not expected to log a PUT request that does not contain a content length

That there is no / at the beginning of the resource name

### 3. Design:

First I need to fix the mistakes I made on the last assignment. Then I need to create a queue for the cache. I decided to make two queues, one that stored file names and another that stored contents of the cached file. These two queues stayed in sync where each time there is a new file in the cache a new element is pushed to the back of both queues, and if the queues are over capacity both queues have the first elements removed and written to disc.

For logging, we no longer have threads needing to share access to the log file at the same time, so we use an offset that is no longer used a critical section.

### 4. Pseudocode:

```
handleRequest(socket) {  
    if(badRequest) {  
        sendRightError();  
    }  
    if(log) {  
        getOffset();  
        pwrite(loginput offset);  
    }  
}
```

```

        increaseOffset();
    }
}
else {
    if(inCache) {
        Do request from cache...
    }
    Else {
        Do request like normal and store info to cache...
    }
}
}
}
handleClient(socket) {

    if(log) {
        getOffset();
        pwrite(loginput, offset);
        increaseOffset();
    }

    handleRequest(socket);

}

createsocket()
bindsocket()
listensocket()

while(1)
{
    accept();
    handleClient(socket);
}

```

