Project

- HTTP Server
- 404 page
- Html root dir
- 10 or more concurrent clients
    - Threads
    - Subprocesses - with 1 or more IPC mechanisms
- HEAD and GET at least
- Config file
- HTTP 1.0 at least, ca do higher
- Config front end with ncurses
- Linux + FreeBSD + macOS if available

Process

1) Pick a well defined piece of functionality to work on (state it in a sentence)
    a) Hopefully stand alone or building naturally on what I just did
2) Is it a function? If not, decompose it.
    a) Does it have useful purpose all on its own
    b) Is it going to be large
    c) Has it become large
    d) Does it look like something I already wrote
    e) “Code likes to have a name”
3) Start coding a small part of the functionality
4) Test the code
5) Go back to 3 until done

HTTP Server

- Get the configuration
- Start listening on a port
- Accept request from client (browser)
- Handle request
    - Start subprocess or thread
    - Find the page
    - Send the page or 404 to the client

Config

- What are the names and values, and defaults
    - 404 file
    - Http root dir
    - Port
    - Thread or process
    - IPC?
- Make the struct settings
- Make an empty get_config function that does nothing except set defaults
- Make a file by hand? Or Write to the file using a simple program?
- Make a get_config function that:
    - Command line arguments
    - **Read from the file**
    - Environment vars
    - Hard coded defaults
- Ncurses UI
    - Read
    - Write

Server

- Read config (call get_config)
- Listen on the port for a connection
- Print out the request
- Handle_request
- thread/process
    - Call Handle_request

HTTP

- handle_request (char *, fd)
- Print out the request
- Parse request
- Find the file
- Build response
    - 404 if file not found
    - Send file for the GET
    - Send file info for the HEAD
- Send response back