CSSE 477 – Milestone 1 – Web Server

You are provided with a working web server that can handle GET requests. The web server at the moment does not handle POST, PUT, DELETE, and HEAD requests. Note that the main purpose of this homework is to help you understand how a web server works so that you can extend this homework to build a Web Application Server such as Tomcat and Glassfish servers that provide web containers for Java applications in Milestone 2. Please refer to Section 9 of HTTP RFC [http://tools.ietf.org/html/rfc2616] and response codes [http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html] to understand the details of POST, PUT, and DELETE.

Features

Here is the feature request for the server:

F1 – Redesign and Refactor

Redesign and refactor the existing project by implementing appropriate design patterns to **avoid modifying the same source code files multiple times** for implementing the next three requests and corresponding responses. Also make other appropriate design improvements where applicable. [15 points]

F2 – Basic Support for the POST Request

The POST request must **create** a new file or **overwrite** an existing file in the root directory hosted by the web server with the content in the body (payload) of the request. Assume plain text files for this milestone. [10 points]

F3 – Basic Support for the PUT Request

Your PUT request will **create** a new file if one does not exist or **append** the contents in the body of the request to an existing file.

F4 – Basic Support for the DELETE Request

Your DELETE request will delete an existing file. [10 points]

Note that for all of these requests, your server must return an appropriate status response back to the client.

Deliverables

Report [pdf]

Your report must contain the following items:

1. You have already developed architecture for the web server as a part of Paper Review 3. You must have a new found understanding of how the web server works through this exercise. Use this new found understanding to refine your architecture diagram and present it in the report. [2 points]

2. You have also developed a detailed design of the web server as a part of Paper Review 3. Refine the detailed design as well. List all of the **design patterns** you have used in your new design and how it makes your code better. [4 points]

3. List down places where you can improve the current design further (You do not have to implement them.) [2 points]

4. Provide a test report that clearly shows that all the features are complete. **Your server must be tested using a web browser in addition to other test tools.** You may use JUnit, snapshots of browser in action, etc (be creative). [10 points]

Source Code [zip]

5. Provide reasonable amount of in-line comments in the source code. Bundle and turn-in the project on Moodle. [5 points]

Grading

You will be peer-reviewed by another team in class. You will swap your code and use the following rubrics to evaluate each other's grade: 0 – No Try, 1 – Just Started Implementation, 2 – Half Way Done, 3 – Near Completion, 4 – Completely Implemented, 5 – Completely Implemented and Robust (has conducted different tests). Hopefully this in-class exercise will help you learn from each other's work.

F2	PUT	Grade [0 – No Try **to** 5 – Complete and Robust]
F3	POST	Grade [0 – No Try **to** 5 – Complete and Robust]
F4	DELETE	Grade [0 – No Try **to** 5 – Complete and Robust]
D1	Architecture diagram and changes	Grade [0 - No Effort **to** 3 – Excellent Effort]
D2	Detailed design diagrams and changes	Grade [0 - No Effort **to** 3 – Excellent Effort]
D4	Test report	Grade [0 - No Effort **to** 3 – Excellent Effort]
S1	Source code is reasonably documented	Grade [0 - No Effort **to** 3 – Excellent Effort]
F1	Redesign and Refactor	Grade [0 – No Try **to** 5 – Code Matches the Design Doc]

Timeline

On or before Friday, April 24, 2015 – Please show your design improvements to the instructor (For your convenience, *UMLet and UML Cheat Sheet are available on Moodle under the Resources section*)

Monday, April 27, 2015, 8:00 am – Please turn in your report (pdf) and the source code (zip) files on Moodle. We will conduct project evaluations during the class.