A Cleaner Web: Shifting the Focus from Formatting to Design.

Blaise Mathai, Computer Systems
Box 7415, 1 University Hts., Asheville, NC 28804
828.231.6774, bmathai@unca.edu
Expected Graduation Date: December 2015

Problem Statement:

When developing for the web, logic and design challenges are commonplace. The job of the developer is to overcome these challenges and create quality material. Unfortunately, there are also regular inconveniences that can snowball into massive problems if not kept in check. Lines being left out in CSS files that allow cross browser compatibility, poor formatting, and vague or infrequent comments can all leave the developer frustrated and confused. These tiresome issues take away from the efficiency and experience of web development.

Purpose:

I plan to create a program that is essentially a toolkit for web developers. The toolkit will allow them to focus on the major tasks of their project, instead of wasting time with the inconvenient but necessary details. This toolkit will be built into a text editor from the ground up using Java. The available tools will be options in a drop down menu format. I plan to include options that behave based on the type of file the user is working with. Below is a list of what I would like to include, followed by the reason I believe that it's worth including.

Basic format tool for CSS and HTML:

I plan on creating a basic CSS parser from the ground up, and use jSoup to parse HTML. From there I would like the user to be able to perform a basic format to fix indention, line spacing, etc.

Comment tool for Javascript, CSS, and HTML:

I would like to include a tool (tied to a hotkey) that will detect the file type, and add the appropriate comment where the cursor is placed. I believe this will save people a lot of time.

Method Documentation for JavaScript:

This tool will make it easy to create documentation for JavaScript code by sweeping through it and creating standard documentation comments at the beginning of methods.

Cross Browser Format for CSS:

I would also like to add a tool that finds lines of CSS that work a limited number of browsers, and inject additional lines, when possible, to make it compatible with all mainstream browsers.

Many of these tools set up the ground work for more complicated procedures, and if possible, I would love to expand on them.

Software & Hardware:

A standard, up to date, personal computer will be adequate to run the toolkit software. I plan to use Java in the NetBeans IDE to develop it. The reason I have chosen to use Java is because it allows me to release the project on Linux systems as well as Windows and Mac. Java also has adequate libraries to build this toolkit with.

Methodology:

1) Research common web development inconveniences.

2) Build algorithms to solve these problems.

3) Write code for each tool individually.

4) Thoroughly test code for each individually.

5) Rewrite/retest code as needed.

6) Package finished individual programs into a single interface.

7) Design layout, menu, etc. for the package.

8) Debug finished toolkit.

9) Present finished toolkit.

Testing Method:

I will test my program through user made HTML, CSS, and JavaScript files. I will allow a test group of people, at least 4, to create files and manipulate them using my program. I will also be testing it with people who are not familiar with programming, to make sure that it can hold its ground as a standard text editor.

Indication of Competency:

This project will demonstrate competency in Computer Science in multiple ways. To start with, I will need to have an understanding of web development languages and common issues that developers run into. This project will also test my problem solving and algorithm writing skills. Most importantly, upper level knowledge of Java will be used heavily. In addition, testing, debugging, and design skills will be demonstrated through the completion of this project.