**General Assembly Front-End Web Development Assessment**

**Overview**

Candidates for a General Assembly Certificate in Front-End Web Development will apply principles of HTML, CSS, and JavaScript in creating a one-page website to showcase a topic, business, cause, or person of their choice. In building this webpage, they will demonstrate their proficiency with the following learning objectives:

* Write clean, efficient, concise, logically structured, and easily readable code.
* Write HTML, CSS, and JavaScript to match a given mockup or visual presentation.
* Design a responsive application with an appropriate load-time, as measured by the YSlow plugin.
* Organize file structure in an intuitive manner, with separate files for HTML, CSS, and JavaScript.
* Use SEO techniques to optimize the site’s visibility in search engines.

**Assessment**

Projects will be assessed on the following rubric.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Functionality** | **Appearance** | **Performance** | **Structure/Organization** |
| 1 | Markup errors (missing/extra closing tags) | Visual errors (unnecessary scroll distances, misplaced content) | Responsiveness | File Structure |
| 2 | CSS errors (non-existent properties, incorrect rule syntax) | Matching specifications | Load Time | Separation of Concerns |
| 3 | File references (images, CSS files, linked scripts) |  |  | SEO Techniques |
| 4 | Matching specifications |  |  |  |
| 5 | JavaScript Console Errors |  |  |  |
| 6 | Links |  |  |  |
|  |  |  |  |  |
|  | **Code Quality: HTML** | **Code Quality: CSS** | **Code Quality: Javascript** |  |
| 1 | Structure | Structure | Organization |  |
| 2 | Choice of HTML tags | Use of CSS Properties | Program Logic |  |
| 3 | Use of classes and IDs | Readability | Readability |  |
| 4 | Readability | Concision | Efficiency |  |
| 5 | Separation of content |  | Maintainability |  |
|  |  |  |  |  |
|  | **jQuery** | | | |
| 1 | Student's creativity regarding using the jQuery library to affect the content and/or the behavior of their HTML pages. | | | |
| 2 | The number of calls to function the student's jQuery code included to return an object that they then display elements from. | | | |
| 3 | The number of times the student included the jQuery UI library and widgets from the project (accordion, date picker, etc). | | | |
| 4 | The number of custom animations the project includes (using $ animate()) | | | |
| 5 | The number of the following concepts the student's project includes: | | | |
|  | - Basic control flow statements of JavaScript (if/else, loops, switches) | | | |
|  | - Data types (including functions and objects) | | | |
|  | - Logical operators (==, ===, !=, <, >, etc.) | | | |
|  | - Using jQuery to traverse the DOM | | | |
|  | - Using jQuery to add/remove elements | | | |