**ID512 2023 website aromatawai**

**Due Date:** Friday, 23rd June, midnight – code freeze

**Value:** 70% of your final mark

**Group Size:** Individual

**Learning Outcomes:**

1. Use fundamental components of web pages and basic client/server communication.
2. Develop simple web-based applications using industry relevant client/server-side programming languages.
3. Use industry relevant tools and workflows in the development of web-based applications.

**Aromatawai details:**

You are going to **build a website**. The point of this assignment is to **demonstrate your competency** in the areas outlined in this course.

* **Choose a topic** and **three example sites** from which you want to incorporate elements.
* **Write a brief document evaluating and critiquing** the sites – why did you choose these particular sites? What elements appeal to you and why?
* **Build a site using svelte –** your site should have at least **2** pages and demonstrate good component architecture.
* **Correct encapsulation of common functionality** (no repeated code across components/pages)
* **Competency using HTML** to structure components (semantic HTML where appropriate, correct use of tags)
* **Competency using CSS** to style your site (adherence to modern web design principles/conventions, correct encapsulation of common styles, correct use of selectors)
* **Competency using JS** to allow user interaction (adherence to ES6 syntax, code elegance)
* **Competency using asynchronous programming (AJAX) and APIs** to fetch and display data.
* **Competency using Git/Github to maintain your code** (frequent, meaningful commits, descriptive commit messages, README, feature branching)

**>> YOU MAY NOT USE A PRE-BUILT TEMPLATE (e.g. A BOOTSTRAP TEMPLATE).**

**Milestone schedule:**

There are no **required** milestones or due dates that any part of your site needs to be completed by. We will do some **practical work in-class** that will give you a good foundation for tackling your individual assignment sites. However, most importantly, you are encouraged to work **continuously** and **at the pace you feel comfortable with –** if you wish to work **faster** or **slower** than we progress in class, this is fine. It would be good to **discuss with me** if you start to drift ahead or behind the general pace of the class, just so I can keep an eye on your progress.

**Marking Rubric:**

Attached at the end of this document.

**Submission:**

* You will hand in a link to your **code repository on Github**. If I cannot find this, you might risk losing marks.
* You will create your own Github repository – make this public for easy viewing, or **add me as a collaborator** under settings if private (my username is **dfenders**).

**Passing Criteria:**

This assessment is criterion-referenced with a cumulative pass mark of 50%.

**OTHER REQUIREMENTS:**

* **You must have a README in your Git repo that details the purpose of your repo, the technologies used, any known bugs, future roadmap, screenshots, etc… It must be formatted to a professional standard. See resources such as https://www.freecodecamp.org/news/how-to-write-a-good-readme-file/.**
* **You must use exclusively ES6 JavaScript – if you are unclear what this means, ask!**
* **The quality of your code will also affect marks – so including HTML, CSS or JS that does not adhere to good development practices such as DRY, will affect your marks.**

**Marking rubric**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **10-9** | **8-7** | **6-5** | **4-0** |
| **HTML** | Has coded a complex and engaging website using advanced HTML. | Has coded a functional website using HTML. | Has coded a basic website using simple HTML. | Has failed to use HTML to code a website.  Has attempted to use some HTML to code a messy or dysfunctional website. |
| **CSS** | Has coded a complex and engaging website using advanced CSS. | Has coded a functional website using CSS. | Has coded a basic website using simple CSS. | Has failed to use CSS to code a website.  Has attempted to use some CSS to code a messy or dysfunctional website. |
| **Site design** | Has identified a wide range of modern design principles and conventions and implemented them to a high standard. | Has identified a range of modern design principles and conventions and implemented them to a good standard. | Has identified some modern design principles and conventions and implemented them to an acceptable standard. | Has failed to identify a range of modern design principles and conventions and/or has failed to implement them to an acceptable standard. |
| **JavaScript use** | Has coded an interactive and engaging website using complex JavaScript. | Has coded a functional website using JavaScript proficiently. | Has coded a basic website using simple JavaScript. | Has failed to use JavaScript to code a website.  Has attempted to use some JavaScript to code a messy or dysfunctional website. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **AJAX/asynchronous programming** | Has demonstrated advanced use of AJAX/async programming. | Has used AJAX/async programming proficiently in a logical and meaningful way. | Has incorporated simple example of AJAX/async programming in website. | Has failed to incorporate any AJAX or asynchronous programming in website.  Has attempted to incorporate AJAX/async programming in website but functionality is broken. |
| **Git** | Has used Git in a professional manner, and modelled appropriate industry behaviours. | Has used Git consistently to manage code. | Has used Git semi-regularly to manage code. | Has failed to use Git to manage code.  Has made bare attempt to use Git to manage code. |
| **Code commenting** | Has made consistent and thorough commenting throughout, with comprehensive detail. | Has made consistent commenting throughout with moderate detail. | Has made consistent, minimal commenting throughout. | Has not commented code.  Has made bare attempt to comment some code. |