Learning Outcomes Mapped

Learning Outcome	Evidence of Meeting the Learning Outcome	Your Own Assessment of the Grade You Believe Would Be Appropriate	Tutor's Justification of Grading (optional)
Apply a structured approach to identifying needs, interests, and functionality of a website.	Unit 1. Identified personas, identified scenarios. Site structure mapped out .	**A**, B, C, D	
Design dynamic websites that meet specified needs and interests.	Unit 1. Multi-paged website with features specific to fit use cases.	**A**, B, C, D	
Write well-structured, easily maintained, standards-compliant, accessible HTML code.	Unit 2. HTML is solid. No deprecated tags, no in-line or internal CSS. No inline or internal js.	**A**, B, C, D	
Write well-structured, easily maintained, standards-compliant CSS code to present HTML pages in different ways.	Unit 2. Use table on Timeline page for consistent data. Used a column-approach for the gallery page. Used a sub-menu in Categories Page.	**A**, B, C, D	
Use JavaScript to add dynamic content to pages.	Unit 5 On tools page, there a javascript game with media-rich content, inserted every second. Media page dynamically pulls from Instagram based on hashtags.	**A**, B, C, D	
Critique JavaScript code written by others, identifying examples of both	Unit 4 Reflection. The code I used was not build for my purpose entirely, had to modify	**A**, B, C, D	

good and bad practice.	it. Commented all the code.		
Select appropriate HTML, CSS, and JavaScript code from public repositories of open source and free scripts that improves your site and that enhances the experience of site visitors.	Unit 6 Proposal. Used ZeroClipboard library to improve the experience for users in two scenarios.	**A**, B, C, D	
Modify existing HTML, CSS, and JavaScript code to extend and alter its functionality, and to correct errors and cases of poor practice.	Unit 4 Learning Diary Had to hack apart and back together the snippet I used to achieve more dynamic functionality, added some classes to HTML and altered CSS to get a more roman-esque look & feel.	**A**, B, C, D	
Write well-structured, easily maintained JavaScript code following accepted good practice, including	Unit 5. Javascript code is solid.	**A**, B, C, D	
· general appearance and form: commented, properly laid out, appropriate capitalization	Unit 5. Camel case used, all commented.	**A**, B, C, D	
· structure: modular, using functions and objects effectively	Unit 5. Used json to hold my game settings data and in-game vars. Structured well into functions for different game states.	**A**, B, C, D	

· standards-compliant	Unit 5. Works in top browsers.	**A**, B, C, D	
· accessible	CSS Degrades somewhat gracefully given there's no CSS framework like Bootstrap.	A, **B**, C, D	
Write JavaScript code that works in all major browsers (including IE, Mozilla-based browsers such as Firefox, Opera, Konqueror, Safari, Chrome).	Unit 5. Works in top browsers.	**A**, B, C, D	
Effectively debug JavaScript code, making use of good practice and debugging tools.	Unit 5 & Reflection. Used Chrome Developer tools to debug everything. Pretty much have it open 100% of the time I'm coding.	A, B, C, D	
Use JavaScript libraries (e.g., JQuery) to create dynamic pages.	Unit 7. Media page dynamically pulls content from Instagram based on hashtag.	**A**, B, C, D	
Use JavaScript to access and use web services for dynamic content (AJAX, JSON, etc.).	Unit 7. Instagram & Rotten Tomatoes feature achieved through AJAX functior in jQuery. JSON is the data format received from Instagram. JSON also used in Unit 5 for game data and in-game vars.	**A**, B, C, D	
Overall		**A**, B, C, D	