**ECS417U MiniProject Report - Janoth Ketheeswaran 210193499**

Semantic Markup:

The semantic elements I have included in my website are: section, header, nav, figure, article, footer, main, aside, figcaption and details.

Semantic elements indicate their purpose clearly to the browser and users as well. Although they do not make any visible changes to the website (on their own), they help organise source code into different parts/sections, improving readability and making debugging easier.

An example of this with my own website is with the navigation bar at the top of every page of my website. I put the source code for it in between nav and header tags to demonstrate that it serves the purpose of navigation around the site and that it should appear at the top of every page. This helped with debugging as if my navigation bar was working, I would be able to skip the code in between the respective tags when trying to find errors, improving efficiency.

Another good use for semantic elements is with the look of a website. Assigning different IDs and classes to these elements allow programmers to write style rules that apply to certain areas of the page.

To get a black bar on the top of every page (to put the navigation bar in) I sectioned off the page into two sections, colouring one in black and the other with the background image to make it look as if there is a navigation bar at the top of the page.

Web Technologies/Frameworks:

If I were to conduct the project again, I would use Bootstrap instead of CSS for styling. The main reason for this would be the better grid system, allowing me to explore more page designs than a traditional 3 column layout or a 2 row layout.

Another reason for this would be that bootstrap is more lightweight and customisable, and it has more responsive structures and styles than CSS. This would allow me to style my page with more exotic styles, exploring different themes and making the feel of the website an extension of my design preferences.

Another framework I would use is CakePHP to develop my web application. The main reason for this would be that emails, cookies and sessions are easily handled in this framework, which would make the process of having multiple users registered to the website easier to program and easier to monitor/update in the future.

Another reason would be the inbuilt templating engine, which would take away most of the stress with using CSS. It would allow me to worry more about the functionality and features of the website than the look, allowing me to create a website that delivers more in the same amount of developing time, drastically increasing efficiency.