EECS Project 4- Group 17

Integration Strategy

The integration strategy used by our team was Top-Down Integration. We would start at the top of our module and work our way down to the bottom. We would first make sure that a page would be linked to other pages properly, then move our way down to see if on the new page the functionality for the wellness section was working then we would check the workout section. For the wellness side we had to test to make sure the links at the top connected to the right part of the page, and that all the subsections had working functionality and could save correctly. On the workout side we had to integrate the timer with the workout display, this posed as a challenge initially, but using top-down allowed for us to integrate the new objects slowly to ensure that not only the functionality was working before moving to the next section, but also that the GUI was still presenting how we wanted it to. Top-down testing allows for interface errors to be found quickly and earlier in the process, which is where most of our defects were being found. The program itself is mostly GUI with javascript functions to help save user inputted data and to make the website more interactive for the user. For top-down once our saving data function was written we could then write test functions for the saving to make sure it was all working as it should. A drawback to this kind of testing is that it works from the top-down so once something has been integrated at a higher level, a lower-level integration could potentially affect an upper-level object which would require us to backtrack. This happened a few times when integrating code. When a new item was added to the GUI it would sometimes rearrange the page which means we would have to go back and try to integrate the new item without it changing what has already been integrated.