File I/O Reflection

I thought the File I/O activity was a very useful way to practice source control with group dynamics early during the Full Stack Summer Camp. It took us a couple cycles of branching off from dev and merging back into dev to become comfortable with the process; yet, we ran into less issues with source control than I foresaw. I believe our team worked well dividing up work and completing that work as well in a timely manner.

Unit Converter – Client-Side Reflection

The breaking up of the Client-Side and Server-Side portions of the Unit Converter project proved to be helpful when learning HTML, CSS, and especially JavaScript. The team dynamics yet again proved to be fruitful, yet we were down by one to two people each day. We used w3schools.com as well as moqups.com to our advantage so we could focus on making our project look good and have a great user experience. Our plan for workflow included making an index page first to use as a template for the other pages. This proved to be useful for styling all the other pages but created a bottle neck as we decided to wait to add the JavaScript until the index view’s HTML and CSS were copied over to our other views.

Unit Converter – Server-Side Reflection

I believe this portion was perhaps the largest conceptual jump and perhaps next time we would be given more time to fully immerse ourselves in the idea of a controller. It also bottlenecks the workflow in a team as far as who is setting up the controller and as a result only a portion of the team has a chance to practice writing the actual code and see how it relates to the concepts discussed beforehand. However, splitting up the controller from the rest of the lower layers of the stack allowed for space to learn about controllers and how razor code operates in a .cshtml file. Razor code appeared to be one of the more difficult parts of MVC to practice and learn effectively. Although the best uses for razor code were highlighted, I got the impression that few individuals were actually able to practice implementing the code on their own.

Final Project Reflection

CAN DO AFTER THURSDAY

Full Stack Summer Camp Reflection

Overall this experience has been miles better than last year’s and I’ve learned so much more than I remember from last year. I couldn’t see everyone being as successful, when it comes to the learning portion, if this opportunity was carried out like it was last year. The “random” group aspect is very useful for developing those soft skills needed when working in a team. Taking notes this year proved to help myself learn and pay better attention to the material being presented. The increase in note taking, collaboration and opportunity to learn from those who have previous knowledge of the material allowed for much better quality in our projects and quicker learning arc over the three weeks we have had.

I think the process could be improved by using the summer camp as a “project” where our team could have a sprint every week or two. This would allow us to practice agile development and more specifically scrum when dealing with a multitude of projects such as our portfolio, final project and individual learning activities. This would also allow for teams to practice with Trello and foster more use of slack for communication in contrast to resource sharing. Another improvement could come from…