Name: Angelia-Grace Martin (934383455)

Term: Fall 2023

Previous Team Projects

Prior to this class, I've worked on three major team projects in computer science: the term projects for CS 340, CS 351, and CS 361. Each experience was unique, with different structures, workflows, and goals.

CS 340 provided experience with databases and web design and served as an introduction to teamwork in computer science. I worked with a single partner, and together we designed the front-end UI and back-end database of a software for businesses to track customers, stock, and orders.

CS 351, the UX course, was more in-depth, with peer reviews and design showcases at every step of our project, and required a rigorous schedule of meetings and mini deadlines to keep everyone on track.

CS 361 served as an introduction to microservice architecture and the Agile Scrum workflow, and the teamwork aspect was more like a relationship between clients. Each of us provided a microservice for another team member's software, following precise specifications.

Working with Continuous Integration

My experience working with CI this term has been overall excellent. I'm very glad to have learned this strategy, and I'm sure it will be useful once I finish my degree and enter the industry.

Overall the group project went well, but there were a few bumps in the road.

To be completely honest, my teammate and I both experienced some extenuating circumstances that added complexities to this project. My wife was diagnosed with a catastrophic illness a couple of weeks ago that has upended much of my personal life, and my partner has dealt with his own difficulties. So getting together as a team to chat about our work has been difficult, as has scheduling. But such is life! We made a commitment at the start to get our work done on schedule and create the best product possible, and we followed through. The Continuous Integration process has been a crucial part of that. We were able to work with one another seamlessly at nearly every step. We were able to stay on the same page throughout the project, getting everything done with plenty of time for thorough testing and checking one another's work.

We were intentional in reviewing one another's code, and I think that played a large part in our success. Initially, we worked mostly in our own separate branches, focusing on getting the individual functions functional. At this stage, there were a few miscommunications, as a lot of our work was very disparate. Moving to the next step introduced some confusion as neither of us had used a CI workflow before and were unsure about what exactly was expected in code reviews and dealing with merge conflicts.

But when we got the hang of it, CI really came in useful. Careful code reviews, the automatic builds, and checking for merge conflicts made sure that nobody's code got screwed up without both team members being 100% aware of what was going on. This kept us on the same page even when life was extremely hectic, and strictly following it helped to ensure that we progressed steadily.

Lessons for the Future

As before, I've come away from this project with unique lessons which will influence my work in the future. The strategy of Continuous Integration is a valuable tool that I look forward to implementing in future projects. Though it was confusing at first and a bit of a struggle at times, and my teammate and I ran into some rocky situations throughout the project, all in all the project went very smoothly. Any bumps in the road were easily solved with a bit of communication and delegation, and using CI made sure that that communication happened quickly and clearly.

I've also learned more about software like Git and GitHub in general, and I feel much more confident. As it turned out, learning Git in the command line at the beginning of the term was also my first experience using vim, which was an unexpected learning experience!

Finally, and perhaps most importantly, I have learned a lot about creating test suites and selecting test cases. The three testing assignments at the beginning of this course covered unique types of tests for unique situations, all of which I made use of in this project: black box testing for eliminating bias, truth tables and diagrams for full branch and condition coverage where necessary, and—maybe most useful of all—how to look at an instance of an unknown bug and identify what the bug is. (I didn't use any brute force random testing in this project, but that skill still absolutely came in handy. One of the trickier bugs I needed to identify appeared only when an input had a repeat digit somewhere… my initial guesses for the trigger were way off, though, thinking it had to do with the length of the input.) The transition from carefully designed assignments with autograder, to a “real world” project in which our grade kind of depends on how well we can rely on our test suite, was smooth and intuitive, and I learned a lot along the way.

Having taken this class and completed this project, I feel very confident that I'm much more ready for a career in software engineering.