Name: Vincent Yasi

Term: Spring 2020

Previous Team Projects

I have had to work with a group on several projects in the past in this program. Sometimes it is just me and another student as a pair, and other times it has been up to an eight-person team. The projects have ranged from group work on homework problems up to creating whole working websites with a developed backend database and the like. Each project and group has had its own difficulties which needed to be overcome, but those hurdles were passed each time, and I developed in my ability to work as a group.

When starting out as a group, it is often very disjointed, no one really sure how to proceed or what rules we should work by. It often takes some time to sort these things out. I have found a good starting point is to get to know each other and just talk about the project in broad terms. This has often helped us to get onto similar terms and to see each person's perspective on the project ahead. From simply discussing, a rudimentary plan forms on its own, and from here a more concrete one can be created. It often lays out specific roles and tasks for each person, making sure all the necessary parts are covered. An accountability plan also usually arises, to make sure everyone stays on track and each of the necessary tasks are completed. This also typically includes a schedule of what needs to be done when in order to reach project milestones at the required time.

I have also found from these discussions often arises one or two people who take the role of project leader, making sure everything stays on track and keeping the big picture of the project in mind. This has often been myself. I think it comes from my innate tendency to always keep in mind what is due next or needed next, and working towards that as a goal, moving to the next signpost when it is done. While doing this, it becomes simple to do the same for the whole group, and unconsciously I often find myself falling into this role. I do not mind it, and it has aided my projects, as it gives an overall driving force to the project and keeps everything moving along smoothly.

In the end, I have had good experiences so far with group work. My groupmates and myself have gotten along well, and we have each been willing to do our share and more. We have worked well using the tactics mentioned above, and have ended up with good results and good projects. We have never really run into difficulties, partially because if one member was having an issue with a part, the others were willing to step in and help them. We have all always treated it as a learning experience to work on the project, and so we help others if they are having issues with it.

Working with Continuous Integration

Going into this project, I was feeling pretty good. I had been able to follow the lessons and material presented in the course well, and I was looking forward to integrating them in a real-world example. I got my chance to do so with this project, and overall it went well.

Starting out, I was a tiny bit hesitant, as I saw we would be working all three on a single file, but then I noticed the assignment itself was actually to create three functions in this file, and as there were three of us, we could each work on one function as the other two worked on the two others. This gave us each our own space in which to work without hampering or crowding the other two. However, at the same time, as it was one file, housed in a single online repository, we also still had to follow Continuous Integration and be able to work all three as a single team, modifying a single file. The project gave us the space to work on our own at our own pace, while also still needing to do so in a shared environment. I felt this was a good way to ease us into Continuous Integration and teamwork as a whole, and it went well for my group.

As we each wrapped up our respective functions, we even began to break the separate workspaces described above and began a bit of work on each other's code. This mostly amounted to writing some extra testcases the person may have missed in their test suite, or noticing a way a part of code could be done more efficiently, but we organically slowly shifted from independent work to a more collaborative style. It was minor, and most of the work on each section was still from one person, but it certainly made us feel like a team, all working toward one single piece. The way the assignment was set up let us do this.

As for the actual process itself, it was an enlightening one. As said above, we had learned about the techniques we needed to employ, but had yet to really use them. Now, we did. The pushing to a central master branch and the peer review took a bit of getting used to at first. Initially, we didn't realize we should only approve pushes if it passed all the tests, and were approving "bad" code, with an aim to fix it in a future push. However, we soon moved past that and established higher standards. We only approved code that passed all the tests in GitHub, and even then only if we felt it was in such a state as to be ready to merge it onto the main master branch. Admittedly, we never really found code "unworthy", though we would give comments and feedback from time to time about pieces that should be changed before it was actually merged to the master branch, and we would take each other's comments to heart. In fact, the toughest critic was not often each other, but GitHub. It was a challenge at times to meet the strict linting standards, especially the "too complex" one, but we would work on each of our requests to get it in the standards, and I feel this made us better for it, making us think more about our code and what was really written there.

Overall, I felt this was a good experience. It was my first time really dealing with my teammates' code (in the past, each person would often write a piece and put it all together, and it would often work). This time we had to critically look at each other's code and see how it was part of the greater whole. We needed to review each piece of code before it was added to the whole, as opposed to just letting it in like with most projects. We also saw how multiple people can write multiple parts of a single codebase and keep a uniform style for the whole thing. It helped us to critically look at group coding and projects in general in a way other group projects had not.

Lessons for the Future

Going into the future, this project has taught me many lessons which will stick with me, and make me a better team member on group code. It has also taught me more about the group coding process, and individual ones, too.

As mentioned in the previous section, working with other people, all on the same file and code, and needing to do manual reviews of each other's code gave us all experience with the overall group coding process. This made us each an overall better worker with future group projects going forward, specifically in the real-world jobs we will have once we graduate from this program. Working alone, or at least on your own little piece of code, without exposing yourself to the work of the others on a project, can give you a narrow view of the overall project and can lead to mismatches and problems with the whole codebase. By using Continuous Integration and peer code reviews, it has opened us to how to have a more wholistic view of code projects, and how to avoid the solitary view that can develop. It has also allowed us to critically examine the work of others (and of ourselves) to build the better project and to better be able to integrate our code into the whole. It can be easy to close yourself into your corner of code, but the practices we employed with this project have made it easier to break out of that practice.

Overall, this project gave us a microcosm of how a real-world group project would work, and gave us the skills we will need to work on such a thing. It taught us better how to work as a group, and how to integrate our code into the whole better, giving us tools and practices we can use going forth. Code reviews make us critically look at our code and others. Continuous Integration makes us see how to work with a constantly changing and dynamic document, and what practices we need to employ to make sure it is built smoothly.