Name: Alejandro Grau

Term: Winter 2023

Continuous Integration Group Project Report

Previous Team Projects

I have worked on several group projects throughout middle school, high school, and college. These projects have ranged from coding in Computer Science classes to presentations in English classes. An ongoing issue with most of these projects was ensuring an even distribution of responsibilities. More often than not it is with someone doing most of the work and/or someone not holding their weight. With our continuous integration project, I was hoping we could combat this issue head-on.

Another issue that has come up in previous group projects is the lack of communication. I've been a part of projects where it isn't clear who is supposed to work on what since we haven't created any form of consistent communication. This ultimately leads to a poor finished product.

Working with Continuous Integration

To combat these issues, my group felt it was very important to create a Teams chat right away. This was our main form of communication as it was very easy to check on a regular basis. We met on a call early on to talk about how we were going to complete this project. To evenly distribute responsibilities, each of us took the role of implementing and writing tests for one of the three functions with the other two reviewing our code. This resulted in a fair distribution of tasks.

One issue that occurred immediately with our continuous integration was our implementation of tests. We decided to write black-box tests based on the requirements of the assignment first, pushing that to our repository, and merging the testing suite together for all three functions. This however meant that we could not merge these branches to main due to our workflow.

After realizing this, we kept our tests to ourselves for our respective functions and continued on. Once our functions were done we submitted pull requests for code reviews before merging them to main. This was the most fruitful part of the project for me and something I feel has been lacking in my coding so far. I've been able to code to meet requirements so far, but I haven't gotten much feedback on my code before. Hearing ways to clean up my code, make my code more readable, and write more efficient algorithms for problems has made me think about my programs in a new light. It also helped to review my teammates' code. Seeing how they tackle problems and how they organize their code helped me grow as a developer.

We did come across other issues with continuous integration. When we merged the first two completed functions, one was in a file named task.py and the other in tasks.py. We all overlooked this discrepancy and our workflow still allowed us to merge into main despite the errors it gave us. I appreciate how my teammates were available to look into the problem with me quickly and we were able to find a solution. We were able to make new branches to patch these issues and approve and merge them right away.

Lessons for the Future

Overall this project was great exposure to continuous integration and deployment. Having to check in with our group regularly ensured everyone was working on what they were supposed to be working on and that we were going to finish before the deadline. Working in a large team, I can see the importance of this technique to make sure that everyone is working with an up-to-date code base and reduce conflicts when merging different pieces together.

Mandatory code reviews allowed everyone in the group to have the opportunity to learn how to be a better developer. We all have different perspectives, strengths, and weaknesses. This allowed us to share that knowledge with each other and have us all grow. This also made sure that only the best, working code was in our main branch ready for submission.