Eirika Sawh

SLogo Analysis

October 30, 2014

Project Journal

Time Review

The project was started near September 30th, and we spent up through October 25th working on it. It’s hard to gauge how long we spent on the project since we didn’t meet as regularly as I did for Cell Society, but I’d say roughly 40-50 hours individually. Most of the time at the beginning was spent coding new features, but closer to the deadline, I worked more on refactoring and improving the design quality. Unlike in Cell Society, I made sure to document methods as I made them, making it easier to find out what things were doing, as well as reducing the work at the end since I didn’t have to write descriptions for every method right at the deadline. Personally, I found it hard to write JUnit tests to test my code and instead sort of assumed it worked and traced how code would flow by myself. Once the parser was working, however, I tried different commands to see what worked and what didn’t work and tried to fix the problems as I noticed them. The easiest tasks were coming up with basic ways in which the Turtle should move, whereas the hardest tasks were definitely rotating the Turtle and ESPECIALLY making the Turtle’s lines loop around the screen (as it is now, there are still some errors that I really have no clue how to fix after literally days of just working on that feature). Thus, tunneling too hard on looping lines was probably the biggest waste of my time, but taking the time to really think about where different objects should be (such as Pens and Turtles) as well as which class should interact with another were good in my opinion.

Teamwork

Unlike Cell Society, we met much less frequently and the work appeared to be mainly solitary. Thus, I’m not entirely sure how long each team member spent working, but when we would meet as a team, it would usually be for around 3 hours. We spent the first week designing the project, and after that we didn’t have any more design related meetings, and instead just notified each other over Groupme if we were making any large changes (which didn’t really happen as much as I can remember). Yoon worked on most of the GUI things such as buttons and the color picker, I worked on things related to Turtle movement and drawing lines and receiving commands from the parser, and Davis and Keng worked on the backend parser and commands. Disregarding Keng, communication within the team was generally okay, with Yoon, Davis, and I talking frequently in the Groupme. However, Keng never spoke within the chat even though he was a part of it and made literally no effort to ever meet with us at any team meetings except the first and last one. The project felt like a 3-man project and as far as I know, although Davis had more interaction with Keng than Yoon and me, it was much less than Yoon and my interactions. The fact that Keng didn’t really tell any of us what he was doing really irked me, and I don’t think I could work with him again on another project. As for a plan to complete the project, the roles we assigned to each other at the beginning of the project we held constant (unlike in Cell Society where it kind of flip flopped), and it was generally very stable. If we had problems, we talked it out in the Groupme or in person (again, generally without Keng), and this was maintained as the project’s extensions were revealed.

Commits

I pushed to the master branch a total of 35 times, and typically the first push would be somewhat large as it would be after I completed a certain section of code, and subsequent pushes (usually in the same day) would be much smaller that were just small fixes, typically to what I had already pushed. Most of the time, we pushed to the master branch if we wanted others to look at something we did and to integrate whenever changes were made on both the front end and the back end. Seeing as my commit messages clearly state what I was pushing, I feel as though they effectively say what I contributed to the project.

Commit message 1: “Turtle loops correctly if moving in an X or Y direction < 0. Still working on line looping - have a framework but it's not all put together yet.” The purpose of this commit was to push at least part of a feature that I was working on so that others in the project could look at it (especially Yoon). The commit should not have caused merge conflicts as I did not tamper with anyone else’s code except mine, and I feel it was done in a timely manner seeing as even though it wasn’t perfect, I pushed it anyway to at least have a framework down.

Commit message 2: “Got rid of generic error messages.” This commit was to clean up the code parts that had generic “catch (Exception e) e.printStackTrace()” portions. I don’t remember this on its own causing any merge conflicts, but it might have. Seeing as this was at the end of the project, I feel it was well timed as that was when we spent the most time cleaning up the code.

Commit message 3: “Set the Turtle’s (0, 0) to the center of the display by adding offsets in the ImageUpdater class. This also affects the Line’s drawing by adding an offset.” Although the message may not have been entirely clear, the purpose of the commit was more so to explain that we were shifting the coordinate system to the center of the screen than to introduce a major feature. Since Yoon and I worked on it together, there were no merge conflicts on our side, and there shouldn’t have been any on the backend either, although I’m not sure. Seeing as it was closer to the beginning of the project than towards the end, it was on-time with the project’s demands in my opinion.

Conclusions

I don’t think we over or underestimated the project, just that it was difficult in general. I feel as though I took enough responsibility because I not only did my section, but helped with other sections as well. Generally moving which classes had what variables required the most editing, like relocating Pens into Turtles, having each Turtle have its own canvas + line canvas, etc. To be a better designer, I should try more fully to incorporate different models and patterns when I design classes, and to be a better teammate I should try harder to get those who don’t come to meetings to actually come. If I could work on one part right now, it would be to more fully generalize multiple turtles to work in the parser as well as the front end.