PSY 252 Social Psychology Essay 1 Option 2

Cognitive Dissonance in Programming

Tyler Holland

Cognitive dissonance is defined as "anxiety that results from simultaneously holding contradictory or otherwise incompatible attitudes, beliefs, or the like" (dictionary.com). As a Computer Science student, there have been many times when I have experienced cognitive dissonance while doing a programming project. In particular, my recent work on my senior project has caused me great cognitive dissonance at times. In this essay, I will describe how my senior project made me experience dissonance due to effort, dissonance due to decision-making, and how the two combined at one point in time.

The first type of dissonance I experienced was dissonance due to effort. This form of dissonance occurs when you work hard to achieve something, and you end up increasing your liking of this achievement because you worked so hard for it. The dissonance occurs when you think that maybe what you achieved is actually worthless, so a split occurs between your actual self and ideal self. In my case, this happened over the course of a quarter while I was working on my senior project. I was working on a grocery list application for Android smartphones, because I felt that none of the apps currently on the market were easy enough to use. At this point, my ideal self would have the project finished in 2 quarters, and have it be the easiest to use grocery list app on the Android market. Since I had told others about what I wanted my app to become, including my senior project advisor, my ought self was very similar to my ideal self at this time. I began to work on my senior project at the beginning of winter quarter, and I worked on it at least 10 hours a week during those 11 weeks. I invested a lot of time and effort into this app, and I wanted all of my effort to be worth it. At the end of the quarter, however, I did not feel that the work I had put in was good enough. I made some programming errors that made it hard for me to continue working on the app, but at the same time I had already put so much effort into writing this code that I didn't want to just abandon it. This is where I had the most cognitive dissonance due to effort. I began to try to justify my effort to reduce my cognitive dissonance. I told myself that getting this far with the app was a good learning experience, and that I now had a better understanding of how to make the app better if I did choose to rewrite it. These justifications helped to resolve my dissonance and came in handy during the next part of my cognitive dissonance that followed directly after.

After resolving my dissonance due to effort, I was now faced with the decision to either completely rewrite my app, and improve its long term potential, or to keep the code as is and sacrifice the app's long term potential. If I chose to rewrite the app, I would essentially be throwing away a quarter's worth of work, which isn't in line with my ideal self that would have the app finished at the end of this quarter. I had previously lessened my cognitive dissonance from effort by justifying all of the work I had done on the app already. Rewriting the app would cause cognitive dissonance due to effort again. Also, it was possible that my rewrite wouldn't actually solve the problem that I was trying to solve by doing the rewrite. I was fairly certain that it would indeed fix the app, but I wouldn't know until I was almost done with the entire rewrite. On the other hand, if I kept the current code and didn't rewrite it, I would have a better chance of having the app done by the end of the quarter. This would still cause some cognitive dissonance due to having a poorly written app go against my ideal and ought self, because myself and others want me to be a good computer scientist. I had two choices, and neither were entirely positive or entirely negative, which lead to great cognitive dissonance. I ended up choosing to rewrite the app, but I was able to rid myself of some of the dissonance by doing a few things. First, I made a backup of my current work, so that if I absolutely needed to, I could get rid of the rewrite and go back to the original. Second, I salvaged what I knew I wasn't going to rewrite from the old code. There were a few elements of the app, such as images, formatting, and layouts that I knew were already fine. By reusing these elements, I was able to save myself a bunch of time doing the rewrite, which was one of the strongest negative arguments against doing the rewrite. All of these combined lead me to not have that much cognitive dissonance, as I realized I could still be in line with my ought and ideal self if I worked extra hard this quarter. Additionally, over time my dissonance lessened, because I could see the app becoming much better than it was because of the rewrite.

Throughout my senior project, I dealt with two types of cognitive dissonance, one due to effort, the other due to decision-making. In the end, all of my dissonance was resolved by me finally releasing my application to the Android market, and becoming one with my ideal and ought self. By doing this I was able to rationalize all of my previous actions, and resolve my dissonance.

I am glad that I went through this process of resolving my cognitive dissonance, because this is a common occurrence in the software engineering/computer science workplace. It is common to have to throw away large portions of code, and it is ideal to not have to deal with dissonance every time this happens. Having this experience will help me rationalize similar problems in the future.