

Continuing Professional Development

COMP240 - CPD Report

1601016

May 6, 2018

Introduction

Since the last report, I have been looking for relevant summer jobs and career opportunities. This has lead me to revert back to my previous career goal of working in the field of software engineering, as opposed to just games. In terms of achieving this, this course has given me a good understanding of a number of programming languages, as well as a lot of transferable knowledge. Even though my career goal has changed slightly, professional development for teamwork, time management, and programming competency still remain relevant to achieving it.

The 5 key goals I will be expanding on are:

- Create a personal wall plan of assignment requirements.
- Have lunch breaks away from my desk.
- Use a physical Taskboard for the game project.
- Complete 5 "Katas" a week on Codewars.
- Create a UML style diagram before starting a large coding project.

1 Dispositional - Create a personal wall plan of assignment requirements

With multiple assignments going on, I have found myself doing surplus tasks, or missing requirements until close to deadlines. This results in me having to do more work and is an issue for time management. These past few weeks I have been writing down what I need to do on post-it notes and creating a wall plan for each module of what needs to be done. I plan to bring this forward into next year. At the start of the semester I will create a bare bones plan for which I can add tasks to. Within two weeks, I should have enough tasks to cover the requirements of the module. Unlike previous goals, this plan accounts for the time it takes me to understand the requirements of my modules.

2 Affective - Have lunch breaks away from my desk

Burnout has been a big issue for me over the past few semesters, causing major productivity drops over my whole time on this course. In the past I have tried to fix the problem with planning, but this proved counter-productive since it resulted in more work for myself. For this reason I am going to try and tackle the issue by making myself do less work. Eating lunch while I work is one culprit of my burnout, so I will start taking my lunch breaks away from the computer. This also involves getting outside of the studio for a bit to relax myself and properly get away from work before starting again.

3 Interpersonal - Use a physical Taskboard for the game project

For team projects I have struggled to communicate my work to the team and keep track of what other team members are doing. The main reason for this I have found is that when using an online Taskboard, such as Trello, updates happen sporadically.

These updates aren't obvious either, even with slack integration. This semester my team decided to move to a physical Taskboard, which gets updated at every stand up meeting. As a result I have found it far easier to communicate both ways with my team in terms of what's being done. Taking this forward, I will use a physical Tasboard for my game project from the start and update it on a daily basis with the rest of the team.

4 Cognitive - Complete 5 "Katas" a week on Codewars

I have found myself relying too much on "for loops" and "if statements", to tackle each problem I face. Although this was fine when I was starting out, I have found that these methods are not always the most efficient, or clean solution to a task. For this reason I have signed up to Codewars and will complete at least five "Katas" a week, for different languages. Each time I will browse through other users' solutions for alternative methods. This will increase my knowledge of the language functionality and also make me less reliant on IDE auto-complete. This would also help me prove my abilities to prospective employees.

5 Procedural - Create a UML style diagram before starting a large coding project

I have a tendency to enter projects without sufficient planning, causing me to have to do major refactoring down the line. Because of this, before embarking on a large coding project, I will create a UML style diagram. This doesn't need to be a textbook example of UML, which is what I've found trouble with before. These diagrams just need to provide an idea of the structure and how the components interact with each other. With time I will be able to hone the skill of visually planning out a project, which will be valuable when entering a job and working with a team on larger scale projects.

Conclusion

At the start of the next university year I will create a bare bones wall plan for the three modules in that semester. Over the course of two weeks this will be filled out with smaller tasks which will need to be completed before submission. At the same time, I will start up Codewars again, aiming to complete five "Katas" a week and reviewing other solutions. When the game project starts, I will start using a physical Taskboard and have the team update it with what they are doing to aid communication. I will change my habit of working through lunch and eating at my desk to allow myself some proper downtime away from the computer to mitigate burnout. For the next large coding project, which may be the game or a different assignment, I will try to visualise how it is structured before writing the code for it. The long term result of these will increase my knowledge of programming language tools, make me work better in a team and stabilise my productivity. All of which will improve my career prospects in the software engineering industry. The short term benefits are that I will be able to keep on top of my university work better next year, and have better team communication when working on the game.