WWDC A2 Reflection

For A2 I selected a project in the form of a video game. During in class tutorials and testing different methods of coding and project types, I really connected to the video game forum and created music through data code. For this given project, creating a video game seemed more challenging but also more interesting to me. I heavily utilised the skills learnt in the in class tutorial on MakeCode Arcade coding in Python as the foundations of my project as it introduced me to the basics of video game coding. This included creating a background, title and sprites as well as setting time limits to games and certain levels and much more,

Throughout this project creating certain elements of code was more difficult than I had anticipated, especially elements that we had not covered in class tutorials or in lectures. As outlined in my A1 I allocated myself 5 weeks to complete all of the coding aspects of this assignment and left another 2 weeks for creating an explainer video and ironing out any issues that may have affected my prototype's functionality. Specifically, producing code that allows certain objects to bounce off each wall of the screen and also ensures those same objects disappear once captured by the player. What started as a simple prototype and set of ideas, evolved into something more complex than I expected. Initially, I believed my prototype design was somewhat basic, but as I added more features and elements I had to continually monitor my progress and reevaluate the time management schedule I had originally put into place. Through the time in the in class tutorials I was able to set more realistic goals for each week and break down this project into smaller tasks which ultimately allowed me to stay on track and focused on exactly what needed to be done.

With reference to in class tutorials, Week 5 was crucial in building the foundations of this assignment. In week 5 we focused on MakeCode Arcade and coding video games through Python. Week 5's in class tutorials went in depth into using MakeCode arcade which not only allows for python coding but also allowed me to visualise the functionality of my code through the block building feature. This was imperative to my progress throughout this project as sometimes it became difficult to locate certain errors within my coding, the blocks showed me directly where I may have been going wrong and allowed me to fix mistakes from there.

Peer reviews during class time were also highly important for my progress throughout this project as it allowed me to see a fresh perspective on my project and gave me insights into how I could more efficiently progress. Specifically, in week 8 during in class peer reviews, when I first began struggling with the bouncing element of my code, my peers explained the success they had found in using YouTube tutorials for specific elements. Peers also revealed to me the use of Python Code libraries such as codeacademy and pygame which deeply altered the way in which I viewed coding in general. This library taught me to view my ideas and aspirations for this project as commands that I use to communicate with a computer or program (Berry, 2011). These reviews conducted in class taught me the importance of peer feedback, testing the game regularly and making improvements or changes based upon peer input and my own observations.

Throughout this entire process, I was met with many obstacles and lines of failed code. Refining code and 'debugging' was a skill that I learnt and found crucial throughout the process. Although frustrating and at times disheartening, making mistakes throughout this

process ultimately evolved my understanding of python and the most efficient ways to code. In particular when initially beginning to code the bouncing element, lines were clunky, extremely long and hard to read. Through the use of code libraries, Youtube tutorials and going back through MakeCode Arcade tutorials I was able to refine these lines to make it easier for myself to understand.

Overall, through coding in python this project has strengthened my understanding of algorithms and logic. Writing functions, loops, conditionals, and working with data structures like lists, dictionaries, and sets have all honed my logical thinking. These concepts are transferable to many areas beyond software development, including fields like data analysis, machine learning, and especially decision-making in any other area of life. It has given me a solid grasp of how to approach problems more algorithmically and less emotionally. I feel more prepared for unexpected, technical challenges in any industry. Completing this project has given me the freedom to experiment and be creative in my problem-solving and decision making. The nature of Python, its simplicity and flexibility allow me to focus on innovation, rather than struggling with language barriers. This fosters a creative mindset for me that I believe is crucial for pushing boundaries and developing innovative ideas, not just in coding but in any problem-solving scenario I may find myself in, in the future.

REFERENCES & RESOURCES

https://www.pygame.org/news

https://www.codecademy.com/

Kiersten Baschnagel - MakeCode Arcade Bounce
https://www.youtube.com/watch?v=ACFqo6fph9Y

https://arcade.makecode.com/

Berry, D. (2011). *What is Code?* In the Philosophy of Software (pp. 29-63). Palgrave Macmillan UK.

Week 5 in class tutorial.