GAM110 – Development Practice Portfolio

Part 1: Agile Development Research and Analysis

The agile workflow method is comprised of 3 main points, sprint, scum and review and is a technique to manage the workflow and productivity of a team. The sprint is a period of time in which each member of the team sets out to complete one or more tasks that are assigned at the start of the sprint. A sprint can be any length of time up to about 4 weeks, dependant on the difficulty of the tasks that the team have set out to complete. The scrum is when the whole team meet up and discuss how they are doing on the tasks that they have, for example they would talk about how much of their task they have completed, what may be difficult and how they are doing on timings for completing the task on time. A scrum is ideally done every morning before the team starts work for the day, however in reality scrums are done every 2 – 3 days and really help other members of the team to understand how the rest of the team are doing and how they are progressing. The third and final part of the agile workflow is the review which consists of teammates showing what they have done during the sprint to either the rest of the team or to a singular other member. This allows the work that has been done in the sprint to be double checked and made certain that it works to completion. This is an extremely important part of the agile workflow as it allows the team to catch any issues that might have slipped through the original team members work, this allows for a more complete and flawless final product.

In order to fully understand the usefulness of the agile workflow we also need to look at the precursor to it, the Waterfall workflow. This workflow states that you should start by laying out the requirements so that you know exactly what the final project needs, then go on to designing the product for those specific requirements. After the design has been done you would move on to implementing it into a final product and then verify that it is what the company or consumer wants. The final step states that after this point all that’s left is maintenance, making sure that the product keeps functioning as is should. However this workflow had its downsides due to the fact that it in linear which means that if one thing higher up the chain changes, you would have to start all over again. For example, if you were making a first person shooter that is based around parkour and movement and then someone higher up in management or the team decides that it should be third person, you would have to go back to the first requirements stage. This is where the agile method of project management works a lot better as the constant communication through the use of scrums means that if something was to change then the whole team would know before they get too far in the project. Also, the use of sprints helps with this as well as at the end of a sprint the team could look at what they have done and make sure it is still what they want, this means that if they didn’t like an aspect then they would only lose at most 4 weeks of work instead of however long it might take to complete the project in its entirety. As stated in an article from APM (Association for Project Management) “Traditional 'waterfall’ approaches will tend to treat scope as the driver and calculate the consequential time and cost; whereas ‘agile’ commits set resources over limited periods to deliver products that are developed over successive cycles.” This means that instead of the looking at the project as a whole and calculating the time and cost of development, you can use resources over smaller periods and always have a product that you can show. This allows companies to save money in the long run as they don’t have to spend money developing a finished product only to see that its not what they want but instead are able to see the product build in stages and cancel production if they don’t think that it is worth the remaining time or money.

Agile works extremely well when it comes to game development because it allows that developers to respond to changes in requirement or finance instead of following a structed plan. Also the agile workflow excels in team communication whilst working on a game project, which is one of the most important aspects of creating a successful game. Through the use of scums team member can help and sort issues that others tell them about, it also allows the teams to get some insight in to what the other member are doing and how they are progressing. One big issue in the games industry is bugs in games that have just been released, this is minimised when using the agile workflow effectively because of the review stage which allows other team member to double check each other’s work so that less errors and mistakes get through to the final product. In an article by Clinton Keith called “Agile game development with scum” he states “The game is reviewed at the end of every iteration, and results influence the goals of future iterations”. This means that by using the agile workflow the final product is constantly evolving and changing as the based on reviews of past iterations. This evolution allows the final product to be better than if you were using the waterfall workflow which would deny the team opportunity to make the product better as they would have already planned out everything. In conclusion, using the agile workflow method means that teams will communicate more making the team for efficient and it will also allow the product to evolve naturally, becoming exactly what the consumer or company wants.

Word Count = 1000.

Part 2: Game Concept Overview

**Fùchóu** is an action-packed stealth game about a liberation story where the player fights to overthrow a ruthless dictator. Ex monk turned assassin the player must use a variety of gadgets and tactics to sneak their way past or butcher a variety of enemy.

Aimed at mature audiences it is the players choice to use non-lethal gadgets and stick to their monk’s code to defeat guards or turn the streets into a blood bath using explosives and steel to clear the way.

Set in a **19th** century alternate timeline of china where a brutal dictator kickstarts an industrial revolution the player must outsmart the corrupt guard using steampunk gadgets like the Transmutation Disc, that lets you traverse the environment quickly, or the Shock hook, that lets you stun enemies from afar.

Furthermore, the games environment features more ways to outsmart the enemies such as using polluted clouds of dust to get behind enemies, or hidden pathways by climbing over pipes.

The world in **Fùchóu is entirely new, the environment and culture are made of a mix of Chinese tradition and 19th century industrial revolution London. Using the stealth genre, we will let the player explore the city using every alley and route to complete each level and discover the world.**

Word Count = 210.

Part 3: Game Project

In our team game project I am in charge of coding the player actions and movements whereas the other 2 programmers are focusing on the AI enemies to get them functioning correctly. For our game we have decided that the player will have basic movement such as walk, run, sprint and crouch which are all fairly easy to do. However, we have decided that the player should be able to wall run horizontally and vertically which is where the biggest challenges will come in. In terms of the player actions we have agreed upon the player having a bow and arrow as his weapon, as well as having 2 main gadgets which are a shock gauntlet that can shock enemies and paralysis them, and a teleporting disk that they can throw and teleport to the location. I will also be dealing with the UI elements functionality, such as cooldown timers on the gadgets and ammo capacity.

The basic player movement should be easy to code as I have made player controllers a lot in the past, however the wall running is where the first on the challenges will come up as I have never done everything like that before. I am planning on using raycasts the come out of the player to the left and right and if they collide with a climbable object then I can lock the y value of the transform to suspend the player in the air whilst he moves along the wall. The approach will be the same with the vertical wall running expect I will have a raycast going forward and I will increase the y value if the raycast hits a climbable surface. If I have enough time then I would like to change the horizontal wall running to have an arc to the run instead of just keeping the same y value, this would be done by working out the differential of y = -0.1(x + -4.5)2 + 2 which should give a nice arc for the wall running. This will be very difficult to implement successfully as I have never done anything like this before so this is where I see the most complications occurring.

The bow will be challenging to code because I would need to have some type of “draw” variable to control how far the player has drawn the bow back to effect the velocity of the arrow. I am planning for the player to control the bow with only the left click on the mouse, when it is held down the player will draw the bow and when the left click is released then he will fire the arrow. The actual firing of the arrow should be simple to implement, the challenge will come from the drawing of the arrow and working out the velocity of the arrow in respect to how far the player has drawn it back.

In terms of the gadgets, their effects will be easy to implement as the teleport disk will be done with changing the players positions to the disks position, and the shock gauntlet will just be a case of calling either a function or variable on the enemy that states if they can move or not. The challenges with the teleport disk will be making the arc of the disk feel natural and good for the player to use, I want to implement a line arc to show where the disk will land however I have never done that before so it might not get done. The shock gauntlet should not have many challenges with it as I will use a sort raycast on a button click that will trigger the function or variable on the enemy. I would like to make the icon glow is some respect when you are close enough to the enemy to shock them but that is a stretch goal if I have coded everything else to completion.

The UI elements that I have to make functional will be difficult as I will have to work closely with either the artists or designers in order to make sure that what I code works correctly with the UI that they design. This will be a challenge, not in the coding aspect but in the communication and team work aspect because there will be a look of back and forth between me and the designer of the UI so that we can make it work without any bugs. This challenge is one that I am looking forward to as it will allow me to work closely with another specialism so I can learn how they work and better understand their role within the team. The UI is a big part of any game as it has to work to perfection otherwise the player will get incorrect information, this makes this task even more important as the player has to get the correct information otherwise the whole game will be worthless as the player would not know what they can and cannot do.

My contributions to this game are quite large as I am in charge of the a lot of the core gameplay mechanics which allow the game to actually be played and enjoyed by the consumer. My largest challenge with this project will be making the player controls feel natural and fluid which is one of the most important aspects of a game as players will not want to play a game where the controls feel clunky and unintuitive.

Word Count = 914.

Bibliography:

The Difference Between Agile and Waterfall Approaches to Project Management –

<https://www.apm.org.uk/resources/find-a-resource/agile-project-management/difference-between-agile-and-waterfall-approaches/>

Agile Game Development with Scum –

<http://index-of.co.uk/Agile/Agile%20Game%20Development%20With%20Scrum.pdf>