Duncan McDonald  
Week 10 Assignment: Agile and Waterfall

Waterfall methodology, as we covered in class, is best suited to situations where a hard deadline is necessary. Waterfall focuses on a linear approach to development where each phase must be completed before the next can begin, after all requirements have been thoroughly vetted. This linear approach makes it far more feasible to estimate the feasible work within a specified deadline.

A specific example of where this methodology might be suitable is the project I did for Liminal VR last year. For that project, a small team of students was tasked with developing a creative concept for an experience that would fit into their existing virtual reality platform. Only after approval of the creative concept from the CEO and developers could our team move on to development of the experience. The absolute timeframe for development was 13 weeks as we started week two of the second semester. In addition to having clear and decisive requirements, the short time frame left no room for agile experimentation.

Once development began, there were clear steps that had to be taken before subsequent development could take place. With Liminal, we were building for an existing platform so had specific Unity version requirements along with a Liminal SDK that would allow us to build for the Liminal platform. Before level development of any kind could begin, I and the other programmer had to finalize level editing tools and player controls. Once that had been completed, level designers could begin designing level layouts. Finally, artists were able to go in and populate those levels with art assets.

The linear development path along with strict requirements are clear indicators that the waterfall methodology would be well suited.

Agile methodology is in many ways the polar opposite of Waterfall. Where Waterfall lays out all requirements and progresses through them linearly, Agile embraces dynamic requirements and concurrent iteration. Agile focuses on breaking down development into smaller, manageable segments. This approach is best suited to situations with constantly evolving requirements and is very often used in massive and complex projects. One example that would be particularly well suited to Agile methodologies would be MMORPGs.

In the case of a massive multiplayer online role-playing game (MMORPG), the game is almost definitely in a state of constantly shifting requirements. There is additionally a great demand for developers to stay agile and adapt to change in technology, the gaming market, and demands of current players. The Agile methodology would ensure that the development process is flexible and can adapt to changes rapidly.

It is also important to note that the initial development process for an MMORPG is not necessarily better suited to one methodology or another. The development process that I feel best suits an Agile methodology is the long term development stage of an MMORPG. Many of these games are initially developed over some time frame but then must be maintained, added to, and updated extensively to satisfy and retain players. During this phase, there could be development of new art assets, story content, or gameplay features that may or may not be connected or dependent. In these cases, a non-agile methodology would result in an extreme waste of time and money.

Overall, Agile methodology best fits in situations where malleability is key and requirements are in constant fluctuation. There are few better examples of this environment than an MMORPG that requires constant change to satisfy players in a shifting market.