

Do the benefits from early and frequent user research outweigh the potential cost on the return on investment for games?

COMP150 - Agile Development Practice

1703170

November 12, 2017

One of the major factors that contribute to a games success at launch is the ability to recognise and fix problems with the game before release. But does too much user research and playtesting negatively impact development? This paper outlines how user research is implemented using Agile development methods and how these methods affect development time and costs compared with other less rigorous methods in the games industry.

1 Introduction

User research is an incredibly important part of a games development and allows a developer to gather important data about the game [see fig.1]. But how much user research is appropriate for your game? Production teams that take an Agile approach [?] to game development often implement user research early on in the process and yield positive result when implemented

well, while others that prefer plan-based approaches to game development do the majority of their playtesting and user research towards the end of the development process. Often these can lead to issues at launch. The purpose of this paper is to discuss the positives and negatives of an Agile and iterative approach to game design with reference specifically to that topic of user research. The paper will compare and contrast these findings to the methodology they used in development [?,?].

Table of Player Deaths				
	Min	Average	Median	Max
colombia-museum-break-in-roof	1	1.9	1	7
colombia-chase-fence	2	2.2	2	3
colombia-rooftops-tiles	2	2.8	3	4
syria-syria-turret1-outside	1	2.3	2	7
syria-syria-rpgesus-trapped	2	5.9	6.5	9
syria-syria-area2-start	1	2.9	2	8
syria-syria-area2-return	1	7.5	8	14
syria-syria-escape-hub-exit-mid	1	3.0	2.5	8
syria-syria-escape-bridge	1	3.1	2	8
yemen-temple-yem-temp-exit-combat-mid	1	2.2	1	8
grave-grave-01-freighter-section-2-exit	1	3.4	3.5	6
grave-grave-01-firstyard-start	1	3.7	3.5	9
grave-grave-01-firstyard-combat-mid-left	0	1.4	0	10
grave-grave-01-firstyard-combat-mid-right	0	5.3	6.5	11
grave-grave-01-firstyard-wreck-hatch	0	3.1	2	9
cruise-ship-cruise-container-fight-mid	2	5.0	4.5	8
cruise-ship-cruise-ballroom-fight-start	1	5.4	6.5	10
cruise-ship-cruise-ballroom-fight-mid	3	7.6	8	11
cruise-ship-cruise-chandelier-climb	1	2.8	2	9
airport-car-field-start	1	2.3	1.5	8
airport-car-field-mid	1	2.3	1.5	7
sandlantis-san-desert-battle-start	1	4.9	6	7
sandlantis-san-cistern-noria-tower-start	0	3.1	2.5	6

Figure 1 [GDCAAttention]

2 Postmortem Comparisons

Because in Agile, user research is usually done early in the development process, the data gathered from early playtesting can be used in sprint reviews and can mitigate the need for a large crunch at the end of development as bugs and design flaws can be spotted and fixed much sooner [?,?]. More

traditional methods may not catch these in this early stage. The longer that problems go unnoticed, the more work is put into problematic code, the more difficult and time consuming it becomes to fix later on [?,?]. This clearly advocates the idea that early and often playtesting is actually beneficial with regards the development time and thus development costs.

In more traditional methods like waterfall, testing is usually planned to be done at the end of a project [?,?]. A lack of earlier playtesting can easily overwhelm a studio if playtesting is only used at the end of development for Quality Assurance [?,?]. Playtesting is more about improving quality whereas QA is about getting the game to work. Only allowing for a couple of months to playtest a game before the final development stage is a risk to developers and investors. If the scope of the project has been underestimated then when it comes to the testing stage, a lack of funds or time to adequately playtest and do user research can easily become a reality [?,?]. This can mean a lot more issues at launch particularly for large AAA games [?,?] or the game not being released at all. With an Agile approach, the continuous iteration can ensure that a shippable build is always available and with continuous testing and feedback, each iteration is more likely to be the best the game can be at that time. This significantly lowers the risk for investors, always knowing that launch is viable [?,?].

This suggests that test-driven development in particular seems especially useful [?,?]. Day one playtesting of a game by the developers through an early prototype can also be a good way to focus the team [?,?]. It makes it difficult to ignore problems with the gameplay if you or other team members as developers are experiencing the issues firsthand. This works particularly well for small indie studio's who are making games with a smaller scope and constanly iterating over a basic build [?,?].

3 How Early and How Often?

The question this paper asks can be seen to largely depend on how early and how often user research should start. With regard to the former, it may be considered that the most appropriate time to start playtesting is as soon as the developer has a question about the game [?,?]. This is likely to be very early on in the project and with Agile, and a flexible design document, user research about the game can be gathered in early iterations,. Comparatively, with plan-driven development, this is a problem because the team has a large or rigid design document, teams are less likely to want to change things before the end of development.

How often a games gets playtested depends on how often content or features are integrated as well as when special events are held. Playtesting throughout gives the developer a much better grip on the players experience, saves time because developers wont be wasting time on the wrong things and gets the developer to focus on the nuances of a game. It takes time to find out what parts of a game make it the most fun [?,?].

4 The Community

Early Access on Steam is also a good way of gaining user research data [?,?]. Games that don't use Agile in some way won't be able to go Early Access, as ideally this will happen well before the game is complete. This adds another way of funding the game whilst simultaneously adding an early player base of testers who can give the developer feedback during development [?,?]. This obviously does not suit other methods such as waterfall, which would be unable to benefit from EA.

There are risks however. If the initial release on EA isn't very appealing,

with perhaps the gameplay mechanics leaving something to be desired. Normally this wouldn't be a huge problem, developers would be able to fix this in later iterations if playtesting was conducted within production. When the game has been released to the public in Early Access however, this can be an issue. The game will be judged by both players and press. People may forget about the game a few months down the line when production is finished and may not go back to see the improvements.

5 Testing and Team Communication

Testing early on in the development process can also be a boon to team communication as test-driven development promotes early communication between the different teams in a studio or company [?,?]. Communication between the Product Owner, Testers, Designers and Programmers is paramount to ensuring a successful game is made [?,?]. Whereas if a studio elects to take a more plan-based approach or even an Agile approach with strict and inelastic procedures they may suffer [?,?] [?,?], communications may break down and team morale can be adversely affected [?,?]. This would clearly hinder the development process as a whole and thus increase development times and costs.

6 Conclusion

It's clear that studios should use iteration [?,?] [?,?] but should user research be part of that iterative process. This paper would argue that it should be. Although much of the evidence presented in this paper is anecdotal and comes out of research of various developers postmortems of their games, it is nonetheless a valuable insight into how user research is used in the

development process of games today. Although using Agile doesn't mean problems can't occur in larger companies, it does appear to limit them [?, ?] [?, ?]. In the end, the findings of this paper clearly suggest that continuous playtesting and user research throughout the development process is preferable to the problems that can occur without it, especially in smaller teams and an argument can be made that larger AAA companies should also be playtesting more often, even if they opt for a more waterfall approach to production.