Are User Story distribution methods used in regular Agile Development practice transferable to AAA Games Development?

COMP150 - Agile Development Practice

1606119

November 28, 2016

There are many questions surrounding one specific part of the Agile Development process and that is ”User Stories”, and one of which is do User Story distribution methods used in traditonal software development transfer to the games industry, as after all they are an integral part of Agile Development practice as they are the medium of which developers are given their work in the development cycle to create a finished product, and so are an intergral part of development, and then go further and look at the distribution methods such as Validated Learning and the MoSCow method, I will compare and contrast these individual methods, and by using my research gained from my noted sources, I will be able to ascertain if these methods have any effective place in the games development field and if they are useful at all. And from these reflections on the different methods, a final conclusion will be made on my question and whether further research is required to gain a better understanding of the impact it can have on the usefulness of these methods in the video games industry, .

1 Introduction

As the principles and philosophy of Agile Development is now common practice in both software and games development, and as the need for "valuable software" and software that is created "early and continuous" [1], so that projects are delivered to customers on time, User story distribution methods are being used to distribute the workload around the development team and although several methods exist, it is unclear whether they are transferable to games development contexts, thus this paper poses the question "Are User Story distribution methods used in regular Agile Development practice transferable to AAA Games Development", to answer this the following section will go over the 4 general methods that are popular in traditonal software development, then narrow it down to the methods that are applicable to the games development field, and see if they would be useful, which from the research, they are not.

2 Transfer-ability of Agile practice into games development

Through the research, we have found that there is quite a divide on the topic, and the usefulness of such distribution methods, with the first source outlining a relatively large, self created distribution method that according to the source is "really simple and easy to understand" [2, p. 5], which is always important when working under such tight time constraints and milestones that are found in the modern Games Industry, however they also state that the method is only suitable for the "prioritization of small and medium size project can be calculated efficiently" [2, p. 5], which makes it good for general software development and Indie Games, where a usually multi-skilled team is working on a project and thus can cover several disciplines, however it seems to me that it would be quite redundant for typical AAA Development as staff are normally very specialised and won't normally cross over to other disciplines in normal development practice.

To further elaborate on the previous point, we again turn to Source 2 even goes as to say that "Prioritization of user-stories is a difficult task in agile environment because of its volatile nature." [2, p. 1], which is especially true of games development as there are many factors that can affect the length of a sprint, as mentioned by Al-azawi, Ayesh and Obaidy, such as "Feature Creep" [3, p. 3] which entails "new functionality is added during the development phase to increase project scope and change schedule time" [3, p. 3] and these type of events are what would be described as volatile and such can affect the sprint length on a project, along with causing other issues such as "increased error, possible defects and increased chances of failure" [3, p. 3] which again in turn can also lead to increased sprint time, but then again, they only state that this is applicable to "Unmanaged" [3, p. 3] Feature Creep, and there are many case studies into methods that are meant to keep feature creep to a minimum during development, such as this one by Scott Crabtree, who outlines many methods of reducing feature creeps effects such as "Work through the design as thoroughly as you can, as early as you can"[4, p. 2], which is a feature of an older development method, Waterfall, where a very large design document is created at the start of a project, outlining all of the features the game will have, however as with all collaborative projects, especially on the scale of AAA game development, different parties will make changes and not update the document, so features go unwritten but are planned to be implemented which creates unnessecary work and strain on the team, which is why Agile Development techniques or atleast work ethics from the Agile Philosophy are now being implemented into games development, along with software development.

Interestingly, when research was carried out by Pettrillo and Pimenta, they found that "good practices adopted in the traditional software industry are also found in the games industry"[5, p. 12], which points to a positive as surely the methods used in traditional development carry should carry over to games, however they go on to say "In both studies, the quality of the team was dominant for the success of the project"[5, p. 12], again pointing back to my earlier point that outside factors such as team structure and specialisation are more important to a project, which is why we'll now take a look closer at the way methods used in regular development and see whether they have any significance in Games Development.

3 Distribution Methods in a Games Development Context and their usefulness

Of the methods covered in the previous section, only two will likely transfer to game development contexts. These are: Validated Learning and MoSCow[2, p. 156], with Validated Learning revolving around the development of the most risky features first, and then taking the feedback from that and applies that knowledge into future development, which is good for developers that work on the Early Access or release Beta and Alpha versions of their projects, which could apply to both Indie and AAA development in today's market, but again linking back to my earlier point that, especially early on in development, that feature creep will come into play and slow the development down due to the inclusion of new features demanded by the feedback. And MoSCow works on the basis that User Stories will be divided into 4 parts of importance from "Must have this" to "Won't have this"[2, p. 156], and this method is good for traditional development but as previously, AAA games development now normally outline everything they want in the product to begin with, as to prevent factors such as Feature Creep, and games normally have an end deadline so features that won't have time to implemented won't get included in the stories , so again not very useful in the grand scheme of things.

4 Conclusion

To conclude, the distribution methods used in traditional development would seem to not transfer to games development to any large effect, as there are too many outside factors such as Feature Creep and pre-existing practices in place to create any lasting effect, as found, the use of a qualified and focused team were the main deciding factor on a quality product as the end result, and the methods would simply take the place of already existing practices, and thus there's no point in using them, although methods tailoured specifically to games could be created to aid the development of new projects.

References

[1] Kent Beck, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, Dave Thomas, “Principles behind

the agile manifesto," [Online]. Available: http://agilemanifesto.org/, [Accessed: 21-Nov-2016].

[2] R. Popli, N. Chauhan, and H. Sharma, "Prioritising user stories in agile environment," in *2014 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)*, Feb 2014, pp. 515–519.

[3] R. Al-Azawi, A. Ayesh, and M. A. Obaidy, "Towards agent-based agile approach for game development methodology," in *Computer Applications and Information Systems (WCCAIS), 2014 World Congress on.* IEEE, Feb 2014, pp. 1–6.

[4] Scott Crabtree, "Killing feature creep without ever saying no," [Online]. Available: http://tinyurl.com/jzbzoev, [Accessed: 24-Nov-2016].

[5] F. Petrillo and M. Pimenta, "Is agility out there?: Agile practices in game development," in *Proceedings of the 28th ACM International Conference on Design of Communication*, ser. SIGDOC '10. New York, NY, USA: ACM, 2010, pp. 9–15. [Online]. Available: http://doi.acm.org.ezproxy.falmouth.ac.uk/10.1145/1878450.1878453