What communication difficulties does a distributed development team face using Agile, furthermore what methods are there to improve this?

COMP150 - Agile Essay

1507516

May 6, 2016

Agile methods requires constant effective communication of the development team in order to work effectively, there are a wide range of communication methods that are available, such as; video conferencing, telephone and chat rooms. This paper will address which method of communication will be most appropriate for game developers and the problems that occur with using such methods.

1 Introduction

This paper reviews the adoption of the Agile/scrum development method, and the problems it encounters when it is being used with distributed teams.

Agile is a set of principles, which allows for change and constant iteration of software in development. One of the principles of scrum, which is an agile development method, is that it should have daily communication between the team members [1], in the form of a daily stand up meeting. This is where each member tells the scrum master what they will work on that day, what they have been working on and any problems they are encountering. However as this is face-to-face communication is not possible with distributed teams, what are the best alternatives to overcome common communication issues.

This method of working has become very popular with game developers, so this paper aims to address how communication between teams that are working in different locations. Face to face communication is suggested to be the most effective form of communication [2] But as this is not possible, this paper will propose alternatives to aid in distributed teams.

2 Communication Issues and possible solutions

2.1 Communication Issues

One common communication issue with new game developers that are adhering to scrum, is that the developers will tend to act as the product owner and try to "improve" the design without consulting the product owner [3]. This miscommunication can lead to a product that was not desired by the product owner, and the failure of a project.

Another issue Story cards and social activity are key to the success of co-located teams [1].

Scrum being too *"cumbersome"* to follow and keep everyone updated [4].

Another very common communication issue is that the daily scrum meetings

cannot happen, for example the developers are located in different locations as in the case of scharffs paper [4] where they have a team of students located around the world, and various communication issues occured.

Here is a list of the main possible communication issues [2]:

1. Culture
2. Language
3. Working Hours
4. Lack of Face-to-Face contact
5. Low quality communication medium
6. Unprepared communication tools
7. Miscommunication of Requirement

Communication with people of different cultures and languages is a very common issue with distributed teams, [5]

Communication with developers that are absent to scrums and sprint retrospectives [4]

The story cards consist of three parts, *the card, conversation and confirmation* [1]

2.2 Alternative solutions to help improve communication for distributed teams

Results show that passive modes of communication are preferred for distributed teams and Active forms are preferred for localised teams. [2] Agile Communication Model for Distributed Software Development [6] proposes an agile model that works in a distributed environment efficiently. This method involves intra and inter pair programming among the distributed members.

Post it notes are key to the agile process and need to be replaced.

Agreed active and passive forms of communication such as video conferencing (Active) E-Mail (passive) [2]

Social media tools such as Facebook and slack are good at sharing knowledge to your team.

As Mike Cohn and Doris Ford say, "bring as many people as possible together for the first week or two of the project can increase the likelihood of success." [5] as the first few weeks of a project normally requires the most amount communication.

Establish a routine and stay organized to be productive. [**?**]

3 Conclusion

conclusion incomplete...

References

[1] N. N. B. Abdullah, S. Honiden, H. Sharp, B. Nuseibeh, and D. Notkin, "Communication patterns of agile requirements engineering," in *Proceedings of the 1st workshop on agile requirements engineering*, p. 1, ACM, 2011.

[2] P. Joshi, A. Aggarwal, and S. Goel, "Communication issues in agile methodology: A survey," *International Journal of Latest Research in Science and Technology*, vol. 2, no. 4, pp. 15–20, 2013.

[3] I. Krasteva and S. Ilieva, "Adopting an agile methodology: why it did not work," in *Proceedings of the 2008 international workshop on Scrutinizing agile practices or shoot-out at the agile corral*, pp. 33–36, ACM, 2008.

[4] C. Scharff, S. Heng, and V. Kulkarni, "On the difficulties for students to adhere to scrum on global software development projects: preliminary

results," in *Collaborative Teaching of Globally Distributed Software Development Workshop (CTGDSD), 2012*, pp. 25–29, IEEE, 2012.

[5] M. Cohn and D. Ford, "Introducing an agile process to an organization," *Computer*, no. 6, pp. 74–78, 2003.

[6] S. Bhalerao and M. Ingle, "Analyzing the modes of communication in agile practices," in *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on*, vol. 3, pp. 391–395, IEEE, 2010.