Design, Architecture, and Communication in Agile Development

David Lahtinen[*]
Ruihao Zhou[*]
dlahtine@asu.edu
rzhou21@asu.edu

Abstract

CCS Concepts • **Software and its engineering → Programming teams**.

Keywords Agile

ACM Reference Format:
David Lahtinen and Ruihao Zhou. 2018. Design, Architecture, and Communication in Agile Development. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

Our SER 574 instructor tasked us with researching communications in Agile teams, particularly with respect to architecture and design. This is a well established field, and we won't be making any novel observations, so we will not have a "results" section.Figure one can show the basic idea about Agile team working's outline.

2 Background

How do different Agile teams deliver high quality product when lots of problems are ahead of them? Each team has their own "implementation" of Agile and how they dismiss the big difference across them? By taking a research about this top, except for some core principles each team would adhere, there are several important things should be considered to collaboration and communication between Agile teams.

2.1 Goal

Companies should have cross-team meetings every sprint to craft a clear idea about the final objectives and separation of work for the sprint. If each team has its own ideas about project, it's going to be really hard to continue the project. They should spend their efforts towards same goal and work for it. Multiple teams working at same time will duplicate work and it requires each team to create suited independence to minimize this. Therefore, product backlog is popular among developers which can help them identify and plan for cross-team dependencies.

2.2 Standards

Each company should agree on the standards they would adopt and follow before the project starts and apply this accross all teams. For this kind of communication, meetings between teams are not necessary, but it may make sense to hold a meeting with them all anyways to give them all the same message at the same time, and make sure they are all being held to the same standards. For example, which kind of development procedure will teams use and how they will create testing for each phase? If the company decides that test-driven development will be the way their sprints are run, they may have an engineering all-hands meeting to make sure everyone is on board. Of course, standards can also include patterns, code style and more. The whole work efforts across mean it's a huge product and it's appropriate to create checking points in order to check errors more easily and decrease developer's burden. Also, all teams should have full access to the whole project repository and contribute at anytime.

2.3 Workflow

Each team should have communication quickly, frequently, quickly as soon as possible. It's all about teamwork. Using some project management tools such as Slack, Git and more to convey information. Daily and face to face meeting are two core ideas in Agile development. If you need to communicate ideas to the whole engineering team, hold a meeting with representatives from all teams and be there to answer any questions the members have. Create slides, outilines, diagrams, or whatever tools can aid you in communication. [1] Before taking actions, you should get their agreement. "Communication improves, teamwork improves and "Us vs

[*]Both authors contributed equally to this research.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

them" is reduced. And if the distance is not very long, seed visits infrequent intervals across the length of the project, it builds trust between the client and the team and between the team members themselves. Don't forget, that the highest performing teams are those based on trust!" [1]

Feedback, each team should incorporate feedback quickly form other teams and come up with solutions to fix it before it's going to deeper development. Take Scrum for example, work is divided into small iterations called Sprint which includes a lists of tasks. Since each Sprint's size is not big, they can work on it with expected time. Once finish one, they can easily reflect whether it's good or not and make some necessary change. And when one Sprint is finished, each team is able to evaluate it and decide which parts should be kept or altered. Changes can be incorporated into next iteration which can save lots of time and increase productivity.

3 Communication across Agile Teams

As Dr. Mehlhase spake to us in class, Architecture is design at the highest level. At this level, the Agile teams cannot work separately from the rest; there must be communication and coordination. As the government describes in its article about Agile team communication, they recommend holding regular -daily if possible- scrum meetings consisting of representatives from related agile teams so that each team is on board with the bigger picture. [1] If your project has multiple project owners, they should also meet regularly so that releases and features can be planned, but also so that each team can learn from the mistakes and novel solutions of the others. [1] The closer to the modular level that you examine, however, the less communication to other teams about it is required, as they likely only want to know the interfaces.

One of the benefits of having Agile teams is that they closely correspond to modular architecture. Because communication between teams is somewhat restricted by the realities of the large teams, but communication within teams is encouraged, the engineers on the teams are encouraged to have very simple interfaces with engineers working on other teams or in other departments, which increases the maintainability of your software.

4 Conclusion

Large projects cannot be completed in a timely manner without large workforces, but communication within that workforce must be managed effectively. The most effective way to do this that I know of is to put everyone on small scrum teams. The problem this poses to communication between teams can be overcome by having regular contact via representatives, so that the teams can stay in regular communication without spending all their time in meetings.

References

[1] Lulit T. 2018. Collaboration Across Agile Teams. Retrieved January 19, 2019 from https://tech.gsa.gov/guides/Collaboration_Across_Agile_Teams/