Utkarsh Jain

October 2, 2019

Agile Development in Large Organization

CS-C3150 Software Engineering

In today's world, the business operates on a global level where they face new competition, requirements and markets every day. Since software is a critical part of all of this, it is therefore crucial to develop new software quickly to respond to the new requirements and get ahead of the competition. Plan-driven software development approaches, which requires specifying all the requirements upfront, can't to be used in an agile environment where new requirements emerge with time. As new requirements come up, the system design has to be reworked which involves going back to the planning stage, updating the documents, etc. This consumes too much time and delays the delivery of software to the customer. To cope with this agile methods were introduced. These methods were typically introduced to be used by small teams of programmers could work in close vicinity of each other and communicate informally. However, the need for faster delivery of software, which is closer to the demands of the customer, also applies to a larger organization. This suggests that the need for agile methods to be used in a large organization is significant. Over the past few years, a lot of work has gone into evolving agile methods so that they can be used in large organizations. A popular agile method approach which is used extensively nowadays by large organizations is the scrum agile model.(Sommerville, 2016)

There are many advantages of using this model in large organizations. The whole product is divided into sets of builds. The software is designed, implemented and tested incrementally in several builds. This essentially breaks the product in smaller modules which makes it easier for the stakeholders to understand. Since the product is developed in iterations, this enables the product to be released much earlier in the development cycle. The faster and frequent delivery of increments shortens the customer feedback loop. The customers work closely with the developing team explaining to them the requirements and way the software is expected to respond in some particular situation. This involvement of customers in the product development and reviewing develops trust between them and developers, and a positive culture is created in which everyone expects the project to succeed. During every iteration, the team holds a short daily meeting in which they review progress. All the team members describe the work they have done, the problems they faced and their goals for the following day. Thus, the team has a clear vision of the project's status which enriches team communication and boosts morale. The incremental development of the product ensures that unstable requirements don't hold up the progress because these requirements can always be taken into account in the next iterations.(Sommerville, 2016)

There are also some disadvantages to using this model. Generally, when a customer outsources their project to other companies, there is a contract signed between the two parties. The contract also includes system requirement documents. Since in agile model, development of requirements and code goes hand in hand, there is no definite statement of requirements that can be included. In the agile model, the customer pays for the time and resources required in the development of the system. In case if some delay happens in delivering the product, it can lead to dispute over who will have to pay for the extra time and resources required. (Sommerville, 2016)

Using this model can be disadvantageous in long-term systems which requires maintenance. Formal documentation is needed to describe the system to the people who are changing it. However in agile methods, either no formal documentation is produced and if produced, they are rarely updated and so does not accurately reflect the code. This can make the maintenance and evolution process difficult and expensive. Another potential problem is keeping the development team intact. Programmers prefer working on new systems and are unwilling to work in software maintenance after the first release has been delivered. During development of a software, the developers are expected to understand the requirements without necessarily documenting them. If a teams breaks up, they take away with them the knowledge of the software and it can be difficult for the new team members to build up the same understanding of the system. Another key challenge is to keep the customer involved in the process. Customer's interest gets dissolved in the system and are reluctant to participate in the system maintenance where changes are not continuous. (Sommerville, 2016)

It can be a challenge to adopt agile methods in large organizations for several reasons. Large organizations generally have quality procedures and standards for all the projects. Because of this, they are less likely to adopt the more "informal" nature of agile methods. The project managers who are inexperienced with agile development may be hesitant to accept the risks involved with it. Using agile methods is effective when the team members have a high skill set. In large organizations people have a wide range of skills and abilities, and those with lower skill sets may not be able to cope up. Organizations that have a long history of using conventional system engineering processes may pose cultural resistance.(Sommerville, 2016) There can be other challenges like underestimating the amount of planning in Agile, impatience, misunderstanding partial Agile adoption with full adoption, relying on an Agile tool to become Agile, etc.(Anon., no date)

Introducing and sustaining agile methods in a large organization requires cultural change. It can not be adopted overnight in large companies who have been using plan-based models for a long time. Generally, it needs a long time to implement and also may require a change of management before it can be accomplished. Rather than forcing agile methods on the developers, it is best to introduce the concept bit by bit by starting with an enthusiastic group of people. A successful agile project can act as a catalyst which will then attract more and more programmers to adopt the agile concept.(Sommerville, 2016) Agile methods involve breaking a larger system into smaller ones which can be implemented individually and parallel-ly. Educating the developers about this increased efficiency and giving them a controlled and gradual exposure would make them less skeptical and more encouraged to use agile methods.

References

Anon., [no date], *Challenges of agile adoption* [online]. [viewed on: 4th Oct 2019]. Available from: http://www.intelliware.com/challenges-of-agile-adoption/

SOMMERVILLE, I., 2016. Software Engineering. 10th ed. Boston: Pearson.