Agile Methods

By

Ming Lei

Foundation of Software Engineering

February 16, 2018

Introduction

In a modern economy, market changes quickly. Faced with changing market requirements and endless challenges, the product requirements may not be well understood either by the customers or the developers. Some traditional software development process models, such as pure waterfall model, do not work well on all projects. There is no best "one size fits all." How to accommodate the needs of customers and deliver the project on time using the most efficient quantity of resources are important topics deserving of deep study. This paper will discuss the following five parts: (1) summary of Agile philosophy;(2) two specific Agile development processes;(3) differences between Agile processes and other processes; (4) criteria for successful Agile process; and (5) risks and challenges encountered in Agile methods.

Agile Philosophy

In the book *Software Engineering a Practitioner's Approach (2010)*, Dr. Roger Pressman and Bruce Maxim concludes that the agile philosophy emphasis on four critical issues. First of all, it is essential for self-organizing teams to have control over the work they perform. During the project development process, developers may have difficulty to achieve goal rapidly. Developers should be encouraged and motivated on project. Give them freedom and trust them to get work done.

Second, all parties, including customers, team members, and practitioners, in the project should be able to communicate and collaborate with each other effectively (Pressman & Maxim, 2010). During the development process, business people and developers must work together to brainstorm on ways and means of developing best solutions towards the success of the project.

Within the development team, face-to-face communication is the most effective method of conveying information because it enhances the faster decision making.

Third, change signifies an opportunity. In practice, project requirements may be changing quickly. The customers are reluctant to commit to a set of traditional requirements or the application area is not well understood either by the customer or the developers. Agility principles welcome changing requirements at any stage. The agile philosophy relies on adapting to changes to increase the customer's competitive advantage (Pressman & Maxim, 2010).

Finally, agile philosophy stresses delivery of valuable software quickly and put customers' satisfaction in first place. Agile philosophy wants to give the customer good visibility into project progress. To achieve agility, the Agile Alliance points out that developing team not only delivery of software continuously but also delivery working software with a preference for the shorter schedules (Pressman & Maxim, 2010).

Agile Development Processes

Extreme programming is considered as the most popular agile process model. The Extreme Programming process, relies on object-oriented approach, includes four framework activities: planning, design, coding, and testing (Pressman & Maxim, 2010). Basically, the planning activity is a requirement collection activity that ensure team members to understand the business environment for the software. The broad feel of project features and functionality will help team members measure the development and assign a cost. Design activity prefer "keep it simple" principle. Coding stresses on unit test to exercise stories and focus on use of pairs programming to create story code. In the Testing process, unit tests created before coding are

implemented using automated testing framework to encourage use of regression testing (Pressman & Maxim, 2010).

Another widely used agile software development process model is Scrum. In another book *Software Engineering* (2010), professor Ian Sommerville points out Scrum emphasizes iterative development rather than specific technical methods. Scrum process consists of three phases (shown in Figure 1). The first phase is outline planning and architectural design. The goal is to detail out the requirements of the project guided by the desired results of the project. The starting point for planning is backlog, which is the prioritized list of requirements to be done. The second phase is a series of sprint cycles. Each cycle releases an increment. The last phase is to wrap up the project, finish documentation and deliver software to customer for evaluation.

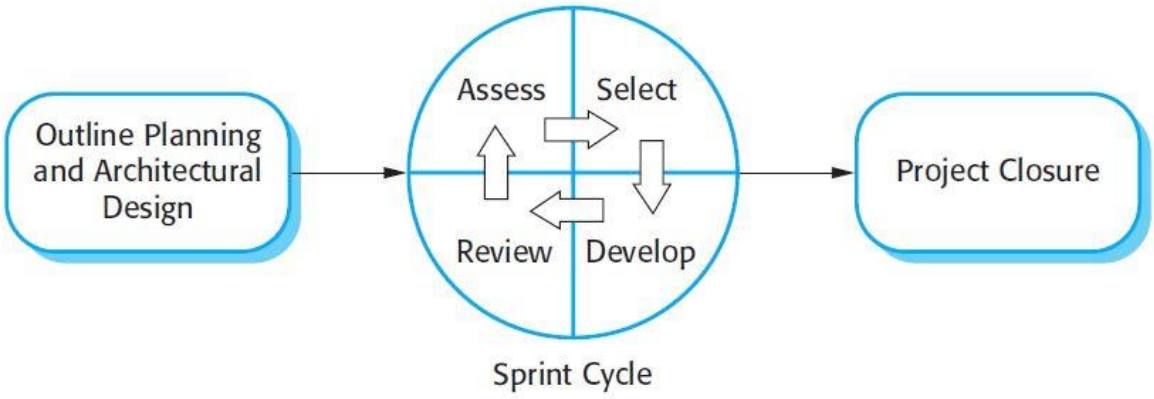

Figure 1. The Scrum Process

Differences between Agile processes and other processes

The Agile process differs from other processes. First, Agile processes emphasize the project plans must be flexible. Welcome changing requirements during development is one of the important Agility principles (Yatin, 2017). In contrast, some traditional processes, such as waterfall approach, are plan oriented. The requirements must be fully-specified up front.

Second, Agile processes provide better transparency for customers compared with other processes. Agile advocates involvement of customers in every stage of the software development process. However, in some other processes, such as waterfall approach, customers are only involved in the planning stage. After the implementation begins, customer involvement become very low (Yatin, 2017). In addition, Agile advocates extensive communication among each party. High level of communication increases the project improvement and customers' feedbacks enhance the project further.

Finally, the delivery time of agility is different with other processes. According to Agility principles, agile processes prefer delivering product in shorter schedules, such as every two or three weeks (Pressman & Maxim, 2010). But some traditional processes, such as waterfall approach, deliver product slowly. In the waterfall approach, customers and stakeholders must wait to see the product at the end of the life cycle. Staged delivery model improves waterfall approach and provides some useful functionality to the customer earlier. In addition, some traditional process, such as waterfall approach and staged delivery approach, can have clear deliver time. In contrast, agility's flexibility, such as changing requirements, may open the door to procrastination. To response new requirements and feedbacks from customers, the delivery time can become nebulous (Shiotsu).

Criteria for Successful Agile Process

According to a joint study conducted by the University of Applied Sciences Zurich and the University of Applied Sciences Northwestern Switzerland in 2015, professor Martin Kropp and professor Andreas Meier interviewed eight successful agile IT companies and concluded the following three categories as success factors in agile process: engineering practices, management

practices and the culture they live. Professor Kropp and Meier structured these three categories as the following Agile Criteria Pyramid (see Figure 2). First, on the engineering level, successful agile IT companies place emphasis on quality assuring activities, such as unit testing, continuous integration and writing clean code. It is essential for the company to use agile methods in a discipline way.

Second, on the management level, effective collaboration and commutation among all parties is considered to be one of the key elements. The team needs interaction regularly updating each other on the outcomes of the project and adjust the changes in the development process. Finally, on the value level, professor Kropp and Andreas found that successful agile team live a culture of transparency and openness. It is important to inform customers the project progress and give the customers good visibility. By doing so, developing team can get feedback quickly. Openness means the team members not only own the accountability of a specific area of the project but also require joint responsibility on the results of the project. All team members should have a big picture of the project in mind.

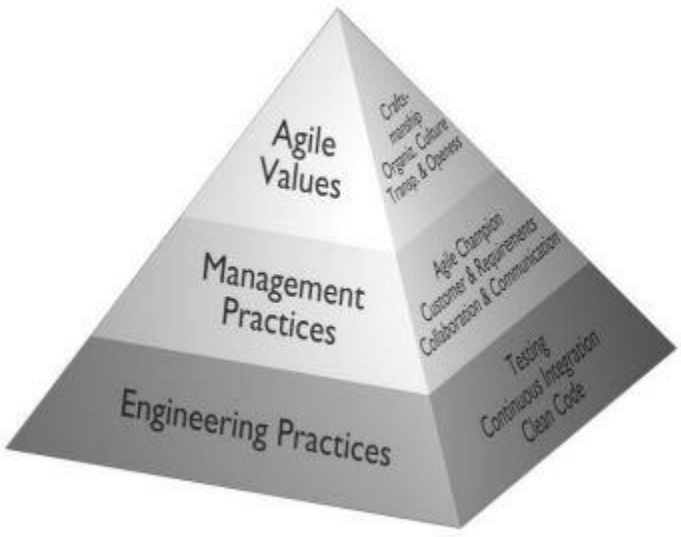

Figure 2. Pyramid of Agile Criteria

Risks and Disadvantages of Using the Agile Approach

Even though the Agile approach is popular, it does have some risks and disadvantages. First, as discussed earlier in this paper, the delivery time of Agile approaches can become nebulous if the customers change requirements frequently (Shiotsu). Second, Agile approaches advocate group work and face-to-face collaboration. However, in practice, face-to-face collaboration can be more time-consuming than other methods (Shiotsu). Phone call or video chat may be more convenient as customers and team members may not be in the same city. Finally, solving the conflict of customer needs can be one risk of using the agile approaches. If the team works on a large project with a lot of customers, assimilating all customer needs is difficult (Pressman & Maxim, 2010).

Conclusion

Agile methods take an iterative approach to software development and they stress on developing rapidly, releasing software frequently, reducing process overheads, and producing high-quality code (Sommerville, 2010). Agile methods encourage the customers involve in the process and welcome changing requirements in every stage. There is no best “one size fit all”. The decision on whether to use Agile methods should be based on the specific project, management, developer, and customer characteristics.

References

Kropp, M. and Meier, A. (2015). *Agile Success Factors.* Retrieved Feb 13, 2018, from Swiss Agile Study Web site:http://www.swissagilestudy.ch/files/2015/05/AgileSuccessFactors2014.pdf

Pressman, R. S. and Maxim, B.R.(2010). *Software engineering: a practitioners approach.* New York: McGraw-Hill Higher Education.

Sommerville, Ian (2010). *Software engineering (10th ed.).* Addison Wesley.

Shiotsu, Y. *Agile vs. Waterfall: A Side-by-Side Comparison.* Retrieved Feb 13,2018, from UpWork Web site: https://www.upwork.com/hiring/development/agile-vs-waterfall/

Yatin(2017). *What is the difference between Traditional and Agile Project Management.* Retrieved Feb 13,2018, from UPRAISE Web site: http://upraise.io/blog/traditional-vs-agile-project-management/