Agile Principles as A Leadership Value System in The Software Development: Are We Ready to be Unleashed?

S B Patil
4th Yr PhD Research Scholar,
Head & Associate Professor
MPSTME, SVKM's NMIMS
+91 9421490622
patil.suryakant@gmail.com

Srikantha Rao
PhD Guide,
Director, TIMSCDR
Thakur College, Mumbai
+91 99209 71285
dr_s_rao@yahoo.com

P S Patil
3rd Yr PhD Research Scholar,
Associate Professor, MPSTME
SVKM's NMIMS, India
+91 9421490623
preeti.patil.india@gmail.com

ABSTRACT

Agile methods generally promote a disciplined project management process that encourages frequent inspection and adaptation, a leadership philosophy that encourages teamwork, self-organization and accountability, a set of engineering best practices that allow for rapid delivery of high-quality software, and a business approach that aligns development with customer needs and company goals. When it is considered Agile Principles as a leadership value System in the Software Development then the question arise that are one is ready to be unleashed? As criticisms include several issues regarding the same like Agile Principles often used as a means to bleed money from customers through lack of defining a deliverable, Lack of structure and necessary documentation, Only works with senior-level developers Incorporates insufficient software design, Requires meetings at frequent intervals at enormous expense to customers, Requires too much cultural change to adopt, Can lead to more difficult contractual negotiations?, Can be very inefficient—if the requirements for one area of code change through various iterations, the same programming may need to be done several times over. Impossible to develop realistic estimates of work effort needed to provide a quote, because at the beginning of the project no one knows the entire scope/requirements and can increase the risk of scope due to the lack of detailed requirements documentation?

Here it is observed and surveyed the various categories of Projects for different kind of group members to adopt and follow the agile principles. It proposes the strength and weaknesses of all 12 Agile Principles based on Indian scenario.

Categories and Subject Descriptors

D.2.9 [**SOFTWARE ENGINEERING: Agile Principles**]: Human Issues, Coordination and Practices.

General Terms

Human Factors, Design, Management, Reliability, Standardization.

Keywords

Agile Practices, Development Phases, Agile Criticism, Indian realistic scenario.

1. INTRODUCTION

"Agile Development" is an umbrella term for several iterative and incremental software development methodologies. While each of the agile methods is unique in its specific approach, they all share a common vision and core values. They all fundamentally incorporate iteration and the continuous feedback that it provides to successively refine and deliver a software system [1 and 4]. They all involve continuous planning, continuous testing, continuous integration, and other forms of continuous evolution of both the project and the software. They are all lightweight (especially compared to traditional waterfall-style processes), and inherently adaptable. As important, they all focus on empowering people to collaborate and make decisions together quickly and effectively. Agile Management or Agile Project Management is a method of delivering projects in a highly flexible and interactive manner.

2 STUDIES AND SURVEY OF AGILE METHODS IN INDIAN SCENARIO

The difference between Agile and iterative development is that the delivery time in Agile is in weeks rather than months. Since Agile Management derives from Agile software development, it follows the same standards defined in the Agile Manifesto when it comes to collaboration and documentation. Several software methods derive from Agile, including Scrum and Extreme Programming [1, 3 and 5].

Agile methods grew out of the real-life project experiences of leading software professionals who had experienced the challenges and limitations of traditional waterfall development on project after project. The approach promoted by agile development is in direct response to the issue associated with traditional software development – both in terms of overall philosophy as well as specific processes. Agile development, in its simplest form, offers a lightweight framework for helping teams, given a constantly evolving functional and technical landscape, maintain a focus on the rapid delivery of business value (i.e., "bang for the buck"). As a result of this focus and its associated benefits, organizations are capable of significantly reducing the overall risk associated with software development. In particular, agile development accelerates the delivery of initial business value,

and through a process of continuous planning and feedback, is able to ensure that value is continuing to be maximized throughout the development process.

As a result of this iterative planning and feedback loop, teams are able to continuously align the delivered software with desired business needs, easily adapting to changing requirements throughout the process. By measuring and evaluating status based on the undeniable truth of working, testing software, much more accurate visibility into the actual progress of projects is available [2 and 4]. Finally, as a result of following an agile process, at the conclusion of a project is a software system that much better addresses the business and customer needs.

3 AGILE AND TRADITIONAL APPROACH

The Fig. 1 below displays the differences between agile and waterfall development processes. By delivering working, tested, deployable software on an incremental basis, agile development delivers increased value, visibility, and adaptability much earlier in the life cycle, significantly reducing project risk.

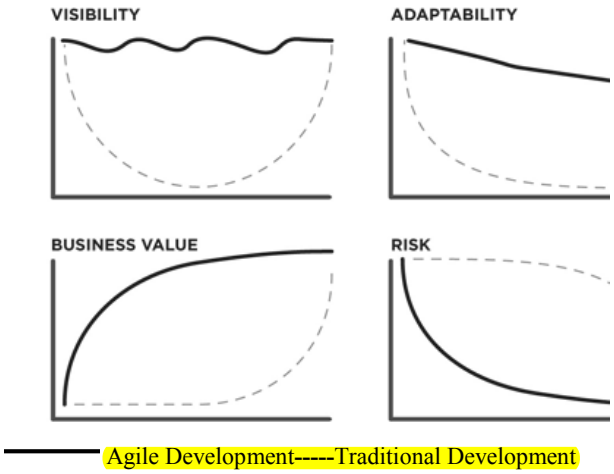

Fig. 1. Agile Development - Value Proposition

Axosoft's own customer base, as illustrated by a recent survey, is also of the belief that rigid project management techniques don't pay. More than 60% of Axosoft customers don't use any particular software development methodology. But, of those who do, Scrum, a relatively new agile development technique, is the one that's gaining the most popularity [1]. As per the survey shown in Fig. 2 Scrum's popularity is rooted in its back-to-basics philosophy; its simplicity and flexibility in execution.

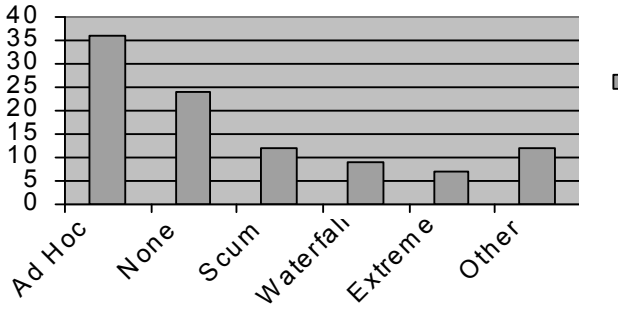

Fig. 2. Survey for % use and popularity of software development methodology

The various agile methodologies share much of the same philosophy, as well as many of the same characteristics and practices [2, 3 and 4]. But from an implementation standpoint, each has its own recipe of practices, terminology, and tactics. Here it is summarized a few of the main contenders:

4 CONCLUSIONS

Agile has the potential of having a major positive impact on the success rate of software projects. For this to happen, agile development will have to overcome many hurdles to become mainstream, but it expects tremendous progress in 2011 onwards. Major SIs and vendors are investing in agile development, consolidation may happen in a scattered agile process landscape enabled by EPF, and agile transformations are hopefully increasingly presented as a journey rather than an 'all-or-nothing' strategy. Tools specifically built for agile development are emerging. Much work is left to do in the areas of compliance, support for large-scale agile development, and geographically distributed development, but here also it is observed that emerging practices and supporting infrastructures that will help making agile development mainstream. All in all, there is good hope that 2011 will prove to be a year where agile development is crossing the chasm and reaching mainstream development organizations.

REFERENCES

[1] Sutherland J. et al., "Distributed Scrum: Agile Project Management with Outsourced Development Teams". Proceedings of the Hawaii International Conference, 2007.

[2] Alfonso, Maria Isabel; Botia, Antonio; An Iterative and Agile Process Model for Teaching Software Engineering, 18th Conference on Software Engineering Education & Training, Digital Object Identifier: 10.1109/CSEET.2005.5 Publication Year: 2005, Page(s): 9 – 16

[3] Zaki, K.M.; Moawad, R.; A hybrid disciplined Agile software process model, The 7th International Conference on Informatics and Systems (INFOS), 2010 Publication Year: 2010, Page(s): 1 – 8

[4] Racheva, Zornitza; Daneva, Maya; Herrmann, Andrea; Wieringa, Roel J.; A conceptual model and process for client-drivenagile requirements prioritization, Fourth International Conference on Research Challenges in Information Science (RCIS), 2010 Digital Object Identifier: 10.1109/RCIS.2010.5507388, Publication Year: 2010, Page(s): 287 – 298

[5] Aoyama, M.; Agile Software Process model, The Twenty-First Annual International Computer Software and Applications Conference, 1997. COMPSAC '97. Proceedings, Digital Object Identifier: 10.1109/CMPSAC.1997.625042. Publication Year: 1997, Page(s): 454 – 459

[6] Don Wells diagram of XP process http://www.extremeprogramming.org/map/project.htm