International Journal of Engineering Sciences Paradigms and Researches Vol. 09, Issue 01, November 2013
(An Indexed, Referred and Impact Factor Journal)
ISSN (Online): 2319-6564
www.ijesonline.com

Trends in Agile Development Methodology

Srinivasa Rao Kosiganti

Enterprise Solution Architect, TechMahindra, Hyderabad (India)
srinivasa.kosiganti@techmahindra.com

Abstract

Agile is an adaptive software development methodology, with focus on early realization of business value and continuous improvement through ongoing end user feedback. The goal of an agile development model is to rapidly deliver quality software driven by business needs and priorities. The project lifecycle approach for agile development is incremental and iterative in nature. The management style is entirely autonomous with focus on team and individual oriented practices.

Keywords: *Agile Development Methodology, Agile principles and values, Agile implementations.*

1. Introduction

Agile Development Methodology is based on the following core principles:

- Ensure that all Stakeholders and Development teams are committed to Agile principles and values.
- Pilot and Prove rather than Falter and Fail – We run Agile implementations in pilot mode to ensure customer and team are in harmony before we run critical engagements.
- Higher empowerment of team and closer collaboration with customer to ensure user stories (requirements) are understood well prioritized and delivered right.
- Create a small but capable initial design, and then maintain and evolve that design over the life of the system.
- Keep release cycles small, and follow an iterative approach to incorporate improvements using experience gained from earlier releases.
- Choose flexible engagement models that permit for right value delivery keeping scope flexible.
- At the start of each release, let stakeholders and project teams negotiate over the features they want to see implemented by the end of each release cycle.

2. Objectives

- Provide an overview of the agile development model.
- Identify current market participants and their agile capabilities for the benefit of potential customers over traditional development models
- Evaluate the advantages and the challenges of agile engagements and provide a best-practices guide that makes for a successful agile engagement.
- Clearly show higher EBIDTA values to the Vendor and high productivity gains to the end Customer

3. Scope

We assess the applicability of the Agile delivery model and the choice of the Agile Method, taking into consideration the advantages as well as disadvantages of using agile for a particular customer engagement while making decisions for development or maintenance methodology.

Advantages of Agile development:

- Agile with its focus on self organizing teams brings in a perspective to work collectively with an aim to get the job done - this choice is exercised when we have mature and relatively experienced members on the team.
- Ongoing and evolutionary identification of needs and wants, coupled with

incremental planning and design enables better change management and reduces waste.

- Better focus on customer priority and end user needs, ensuring pull driven delivery, quicker ROI, better quality and shorter releases with continuous user feedback.
- Iterative and adaptive development in shorter release cycles resulting in early business realization.
- "Small" scoped, value-defined "user story" driven iteration targets, enhance team commitment and ensure measurable progress of NPV and IRR.
- As sprints/iterations are shorter and with constant feedback easier to manage change
- Leads to reduction in waste as features are prioritized often and planned for each sprint with business involvement.
- Daily standup meetings coupled with updating Burn-down charts result in early visibility of progress and common issues so that proactive actions can be taken.
- Faster to develop and allows early detection of failure – ensures quick-wins and fail-fast.
- Helpful in developing quality product in schedule time by managing sprints/iterations effectively.
- Test Driven Development practices with automation testing, vastly improving the quality and the version compatibility of code with rework and waste reduction.
- Weekly Show and Tell meetings are helpful to share demonstrable artifacts with business and key stakeholders and get quick feedback. This in particular builds better credibility with users and precious end-user feedback is directly integrated with the delivery process.
- Retrospective meetings aid in capturing lessons learned and take steps for continuous improvement. Small steps to improve the process/practice, mastered one at a time, ensures a large benefit over a period of time across iterations – Kaizen.

4. Methodology

The Software community has developed an Agile Framework that begins with Assessment and Define phases to understand the agile readiness and the engagement characteristics to tune the agile development methodology.

What is your view on the agile development model and how do you see this trend evolving in the future? How do you see the use of agile changing in the next 3-4 years?

The Software community is witnessing an increase in customer expectations on agile delivery models. Agile has been a development model of choice for most of our customers, as they perceive that early delivery of working code allows them to realize business benefits earlier, manage business priorities for application features better and thus minimize the overall risk of delivery. Although "agile" methods tend to stress on co-located teams, calling for customers, product owners, and business users sitting along with development/maintenance teams, it has been seen that most customers are reluctant to keep the entire project team at onsite for skill availability and cost reasons. We have experimented over the years with multiple customer engagements and have fairly succeeded in distributed agile delivery models.

The Software community has been involved in agile development engagements for both Time &Material and Fixed Bid projects. As of now, about **30%** of overall development projects at the globe are executed using agile methodologies. Agile engagements are best run in the T&M mode with upper cap, for ensuring logical pay-back and cut-off. However, despite the risks, in our experience we observe that the demand for Agile based delivery is more for Fixed Bid projects compared to T&M projects. This puts us, the service provider, to prove the model and take the onus of delivery. Once the model is proven across a few delivery releases, some customers have accepted and moved from T&M to Fixed Bid models.

We feel that enterprise adoption of agile will only increase in the coming years, with customers realizing the favorable outcomes and early business value realization. Initially, customers are

not very forth-right in accepting agile methods, but in the last 2 years the trends have clearly changed with nearly 50% of new development engagements discussing or trying out the "agile models". The primary reasons for this trend are:

- Enterprise shift to reach global markets for business expansion.
- IT aware end-users and impatient Gen-Y workforce, who ask for quick outcomes and are eager to give feedback
- Adoption of global workforce to capitalize local talent and meet regional regulations.
- Global economic pressures and recession requiring businesses to change and adapt faster.
- Early wins in agile projects that catalyses the further funding of Agile projects, which become self-sponsoring through the value-realization of initial releases.
- Cuts in IT spending and growing Business demands of ROI, for more value from IT,
- Increase in number of Fixed Bid engagements and the realization that Agile can reduce the project risk.
- Maturity of IT Services outsourcing forcing enterprises to work with distributed teams for skill availability and cost purposes.

5. Current Trends

Here are some examples on distributed agile delivery from different customer engagements:

- Cloud leading to Agile processes in Infrastructure provisioning & automation
- Agile drive shift from isolated dev teams to CXO(Chief Executive Officer) level in the organizations
- In addition to Scrum and XP, Lean principles from manufacturing are influencing Agile practices
  - While Scrum addresses the team level issues, Lean is believed to provide a better perspective on the whole value by cutting down waste.
- Increase in acceptance of open source tools for Agile development
  - Tools such as Sonar, Jenkins and Selenium have become common and are being used extensively by multiple users
- ALM(Application Lifecycle Management) Tools are recognized as essential means for process standardization and automation

6. Future Trends

- DevOps to drive Agile adoption further to bridge the gap between development teams and IT operations teams
- Maturity in IT outsourcing and increase in Managed Services will force service providers to use Agile practices to collaborate closely with customer teams, reduce project risk, and help customers in early realization of business value
- Customers will demand Key Product Initiatives and Service Level Agreements to link software delivery to business outcomes thus taking Agile from within development teams to between business and IT organizations

We see the future as more about distributed agile development and not merely as agile development. Distributed Agile development is here to stay. Distributed Agile development can also take the form of specialized activities being executed from different entities in a distributed engagement model. The new way is of different specialized teams working together as multidisciplinary teams to deliver output based agile projects is the future. This combines the speed of Agile with specialization of linear development. The other future trend in Agile would be output based delivery. CFO (Chief Financial Officer)s would not give IT a blank check just because somebody wants to do Agile. So organizations will have to think of ways of balancing flexibility with cost controls.

What are the major factors that play an important role in the success or failure of agile engagements? What efforts need to be taken by clients to ensure success? Do you help clients in this process?

The primary drivers for the success of agile delivery are:

- Commitment from Business Teams and other Stakeholders to actively participate in prioritization reviews and provide feedback.
- Choice of right engagement models, team structure, team distribution and contracts as suit a collaborative approach, rather than a traditional/risk based approach.
- Commitment from the Product Owner (sponsor/customer nominee) on enabling the Agile Team suitably to overcome obstacles and provide key business stories and decisions early.
- Focus on learning and continuous improvement to learn from current iteration, take corrective steps, and apply process improvements for the subsequent iterations.
- Agile trained teams enabled with appropriate Agile Delivery platforms, automation and collaboration tools, especially in a distributed agile project.

Other factors that play an important role in the success of an agile engagement/project are:

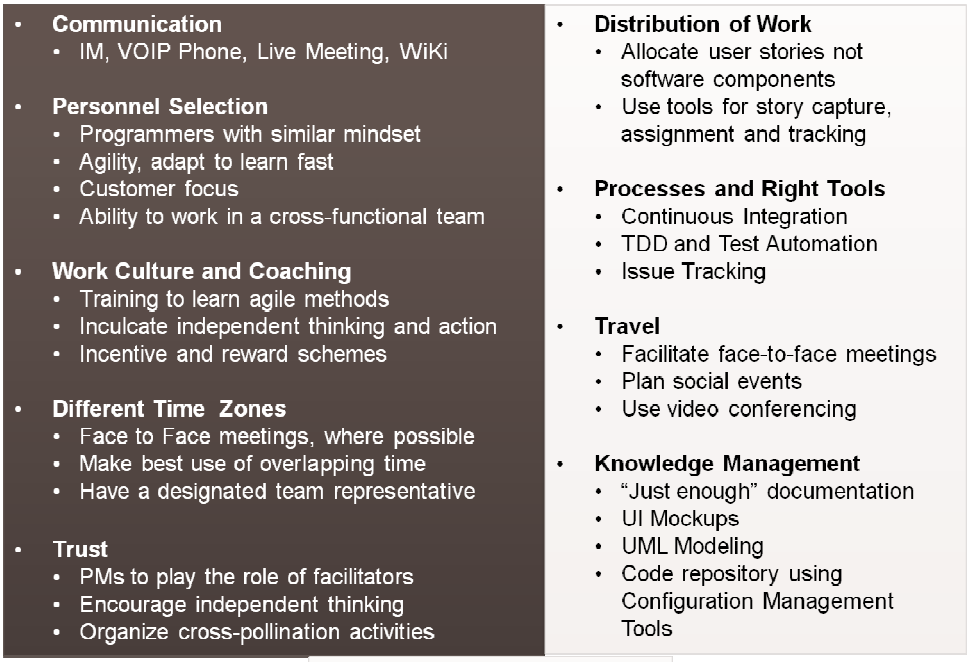
- Enabling Constant interaction with the customer and Clarity on User Stories acquired early.
- Team members being open, responsive, communicative and flexible.
- Sprint ceremonies are to be religiously followed.
- Effective Sprint Planning.
- Ensuring Team uses consensus based approach for estimating, user stories identification and task elaboration for the common 'definition of done'.
- Spike based design and Peer code reviews are planned during the sprint to provide the best design and solution.
- Adopt Test Driven Development from XP(Extreme Programming) methodology with automated testing tools for user story fulfillment. XUnit frameworks are mandated with continuous integration platforms.
- Mid-sprint reviews/demos helped to keep the focus intact.
- Any impediments in implementation are discussed in detail and corrective action is taken thus alleviating fears of non-completion.
- Sprint Retro - The areas of improvement for subsequent sprints with action items are tracked to closure. The areas of improvement from the previous sprint are reviewed during the next sprint's retro.

The following are some of the sanity factors and best practices for the successful execution of agile engagements:

- Drive from business for smaller releases for early business realization
- Releases planned with a fixed schedule. Commitment from stakeholders and development teams to work together to prioritize and release features with the most business value
- Smaller, fixed sprints and adaptive development for early visibility and incremental development
- Commitment from Business Owners to provide detailed user stories on-time; Willingness to prioritize, negotiate and review with development teams for maximum business value from every sprint
- Willingness to adjust from business and dev teams for balancing sprint workload and business priorities during fixed sprint cycles
- Communication at All Levels – Product Owner, Scrum Master and Scrum team
- Communication including participation in daily standup meetings by all key stakeholders (business owners, application owners, dev teams). Willingness to rotate working schedule to accommodate standup meeting times for teams located in different time zones
- Seamless face to face communication systems should be adopted for delivery
- Use tools for feature requests and issues instead of communicating one-on-one on phone or email or chat conversations

- Continuous Integration environment and effective usage of tools. Enforcing daily build cycles and using tools for assessing code quality
- Test Driven Development and willingness to invest in Test Automation for reducing testing cycle times
- Avoid large design upfront - Design part of the sprint. But, initial E2E design view is important for avoiding design issues within sprints
- Frequent Code reviews – Pro-active steps for code improvement
- Plan separate sprints for Release Planning and Deployment as these phases will require more coordination and focused effort
- Improve Tools usage – Tools have to be integrated and used effectively with distributed teams

The following are some of the best practices for the successful execution of distributed agile engagements:



7. Limitations

Limitations of Agile development:

- Can lead to uncontrollable scope creep unless controlled by strong change management practices and "agile-educated" product owners/sponsor nominees.
- Ideal for small teams and smaller projects and needs, and is difficult to manage with larger projects unless supported by experienced agile teams with strong practices, collaboration platforms, well-defined backlogs/story-lists and automation tools.
- Improper definition of user stories and tasks can lead to bad estimates and result in increased product backlog, failed commitments, poor trust and/or repeated rework.
- Team commitment is essential for success as sprint cycles are shorter which requires proper work load balancing and team appreciation mechanisms

- Work disruption and impact to team velocity when experienced team members leave the project teams
- Inexperienced teams can over-promise and under deliver or create worse outcomes, sooner and thus fail faster.

The following are some of the challenges typically faced by project teams involved with agile development engagements:

Challenge	Response
Involvement of end users	Active socialization Frequent "show and tell" sessions for early visibility and feedback, and to capture new requirements
Short Sprint cycles	Limit features initially to ensure that the team is successful and ensure customer buy-in
Testing and Test Automation	Involve Business in Test preparation Experimentation with Tools - QC and QTP
Continuous integration	Automation of builds & deployment
Geographical spread	Co-location of teams by User Story, Using collaboration tools Joint Release planning Conference calls
Initiating Agile	Initial Agile consulting and tools setup help. Take a few releases for Agile model to stabilize and win team confidence
Need for Core Team for overall responsibility	Agile Consulting, Release Management, E2E Design, E2E Testing, Build and Configuration management, Quality, Change Management
No track record of agile	Take cautious baby step approach with learn as you go mindset Start with smaller sprints. Start with lighter features in first few (1,2) sprints and get successful before increasing sprint load
Scope Creep	Do not allow scope creep within Sprint. Allow changes in earlier (first few) sprints. Do not allow changes in later sprints

8. Conclusion

The current trends in Agile Methodology would help the IT personnel in many ways. The bottom line which is profits would tremendously increase for the Vendors. The trends would help different IT Personnel to achieve the targets in a quicker way and yield to better productivity too. This indirectly gives a benefit to the customers in terms of achieving the objectives of a project or a product in a much formidable way. This also leads to improve upon traditional development life cycle models.

References

[1] Some of the contents obtained from Wikipedia.
[2] Content obtained from Internet Websites.