International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 – 5161

Available at http://inpressco.com/category/ijcet

Research Article

Agility in Web Application Development–Success Embraces at the Enterprise Level

Veerapaneni Esther Jyothi†, R. Rama Krishna† and K. Nageswara Rao‡

†Department of Computer Applications, V.R.Siddhartha Engineering College, Vijayawada – 7, Andhra Pradesh, India
‡Potti Sriramulu Chalavadi Mallikharjuna Rao College of Engineering and Technology, Vijayawada – 1, Andhra Pradesh, India

Accepted 16 June 2016, Available online 30 June 2016, **Vol.6, No.3 (June 2016)**

Abstract

Innovation in technology is increasingly very high. To compete with the best, companies always have to be much more inventive and faster than ever. This means releasing software with better performance or creating web based applications that meets the modern trends. With the aim of creating the significant software, companies are adopting Agile Methodologies to convene that challenge ever more.

Keywords: *Agile Development, Quality Facilitator, Continuous Integration, Agile Manifesto, Sprint.*

1. Introduction

Agility created revolution in various fields of software development especially in developing web applications reaching the expectations of stakeholders. Maintaining the quality of software and increasing the productivity is the major concern in now-a-days. Developing high quality software and minimizing the risks are the challenging tasks before any software professional.

This research paper is the result of an immense research work on the implementation of agile software development that facilitates software professionals to develop web based applications and to understand current trends in testing online environments.

The paper is organized as follows: Section2 describes a typical agile process, the Scrum. Section3 presents the principles in agile manifesto and explains the risks associated with agile development. Section4 presents the results of the survey commissioned by Sauce Labs and conducted by Dimensional research.

2. Agile Process

Scrum is an agile framework which concentrates on how the team members work in order to produce a flexible system in a constantly changing environment. The key factors to consider in scrum approach are:

- A product owner creates a prioritized wish list called a product backlog.
- During sprint planning, the team pulls a small chunk from the top of that wish list, a sprint backlog, and decides how to implement those pieces.
- The team has a certain amount of time — a sprint (usually two to four weeks) — to complete its work, but it meets each day to assess its progress (daily Scrum).
- Along the way, the ScrumMaster keeps the team focused on its goal.
- At the end of the sprint, the work should be potentially shippable: ready to hand to a customer, put on a store shelf, or show to a stakeholder.
- The sprint ends with a sprint review and retrospective.
- As the next sprint begins, the team chooses another chunk of the product backlog and begins working again.

The following diagram illustrates a typical agile process in a series of sprints.

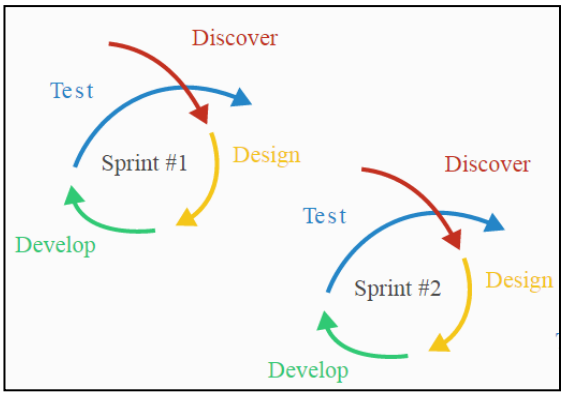

Fig.1 Series of Sprints

*Corresponding author **Veerapaneni Esther Jyothi** and **R. Rama Krishna** are working as Assistant Professors; **K. Nageswara Rao** as Principal

3. Agile Principles and risks associated

Agile Alliance formulated their ideas into values and further to twelve principles that support those values. Values of Agile Manifesto are as follows:

- Individuals and interactions over processes and tools.
- Working software over comprehensive documentation.
- Customer collaboration over contract negotiation.
- Responding to change over following a plan.

The above values are realized in the principles of Agile Manifesto:

1) Value Delivery: Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2) Harnessing Change: Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3) Frequent Delivery: Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4) Business and Development Collaboration: Business people and developers must work together daily throughout the project. 5) Self-Organization: Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6) Communication: The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
7) Measuring Progress: Working software is the primary measure of progress.
8) Sustainable Pace: Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9) Technical Excellence: Continuous attention to technical excellence and good design enhances agility.
10) Simplicity: The art of maximizing the amount of work not done--is essential.
11) Self-Organization: The best architectures, requirements and designs emerge from self-organizing teams.
12) Continuous Improvement: At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly.

Agile methods link developers and stakeholders/users to hit the mark quickly. The below mentioned four major risks has to be eliminated for effective implementation of Agile practices which are seen as the least priority risks but are very serious within the agile projects.

1) Project plan – within a short sprint cycles it is difficult to follow a huge documentation work of a project plan and also we may get frequent requirements from the customers. Maintaining such an enormous project plan is a risky factor which is to be eliminated. So it is necessary to have a plan which should be lean, concise and focused.
2) Documentation of meetings and status report – most of the agile development teams are unable to track the action items and also the telephonic conversations between clients and team are not recorded which will lead to unsafe circumstances when any turbulence arises. So it is always necessary to have documentation.
3) Reviews – because of short iterations there will be less enough time for inspections/audits and testing which leads to a defective product. So, periodical reviews have to be conducted.
4) Tracing – the project is not in a position of tracing the part of the source code where a particular requirement has an impact on. This leads to a dilemma in the team members whether to accept the desired requirement change or whether similar change has already been made or not. So, a traceability matrix with regular updates is required.

The research suggests that a Quality Facilitator (QF) role has to be created to halt all these worthwhile risks.

Fig.10 Portrayal of the responsibilities of QF

Figure10 portrays the responsibilities of the Quality Facilitator in different aspects of the agile software development process. Stating briefly QF should be responsible of the effective management of changing requirements which is an important factor to be concentrated on in order to maximize stakeholder Return on Investment. QF has to track the minute of meeting (MOM) and status report in order to find risks or action items pending and should see that it is circulated to the stakeholders. QF needs to conduct quality reviews and testing before each sprint delivery without fail in order to make sure that all tasks and

deliverables reach highest quality standards. QF along with the agile team should as well address the issues of maintenance and support so as to maintain high discipline and good engineering principles.

4. Results of survey by Sauce Labs

Quality of web applications is the responsibility of software professionals which is a challenging task that creates increasing pressure on release cycles. A global survey to understand current trends in online testing environments reveals so many things. This survey on 520 software professionals is commissioned by Sauce Labs and conducted by Dimensional research.

The executive summary of this global survey is as follows:

- Agile is ubiquitous but few have fully embraced testing best practices
- 67% are deploying at least weekly
- 46% want to deploy faster
- Only 21% have achieved all 5 key areas of agile testing maturity including:
- 23% fix bugs right away
- 24% iterate small testable requirements rather than waiting for features to be completed
- 26% have more automated testing than manual
- 77% of the development and QA teams communicate in real-time
- 86% report development and QA teams think of themselves as partners
- Continuous Integration (CI) changes testing
- Adoption of CI results in more automated testing and testing earlier in the development cycle for the majority of those who adopt it.
- CI also results in more systematic and rigorous testing, trying more things for customer feedback, and writing code with fewer errors but it frequently takes more time to achieve these results.
- Cross browser testing remains important, especially for large organizations
- 95% of large companies perform cross browser testing compared to 86% of small companies
- 68% of large companies test on older browser versions vs. only 60% of small companies
- 32% test their applications on Microsoft Edge
- Need for Internet Explorer testing has not decreased in past year
- Use of cloud-based testing platforms holds steady
- The number of companies testing in the cloud increased only slightly in past year from 43% to 45%

Few Detailed results of the survey are as follows:

1) Adoption of agile development for web applications is continuously increasing.

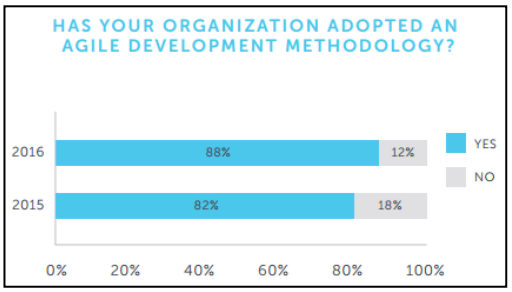

Fig.2 Adoption of Agile Development

2) Software organizations still want to deploy even faster

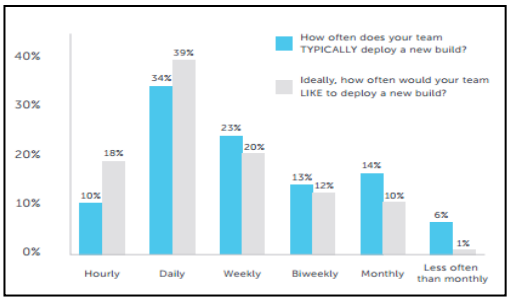

Fig.3 Deploying a new build

3) One of the best practices of agile is to fix the bugs soon after they are identified

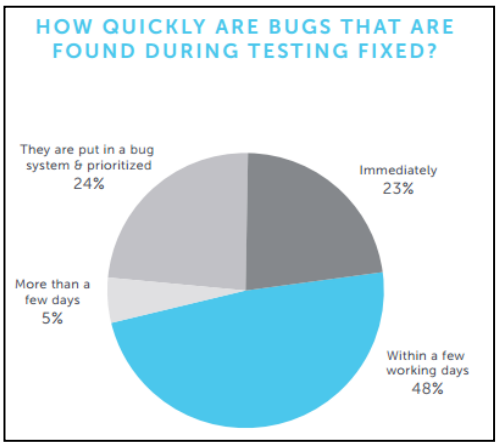

Fig.4 Fixing bugs faster

4) Adoption of Continuous Integration continues

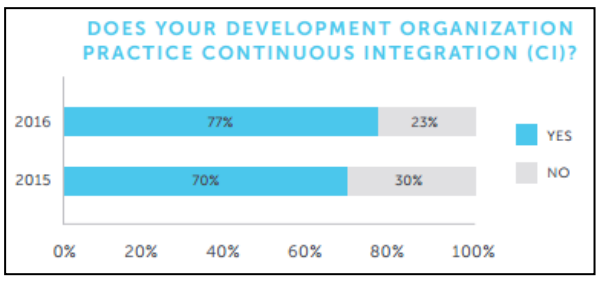

Fig.5 Adoption of Continues Integration

5) Use of Continuous Integration servers evolved with Jenkins taking lead

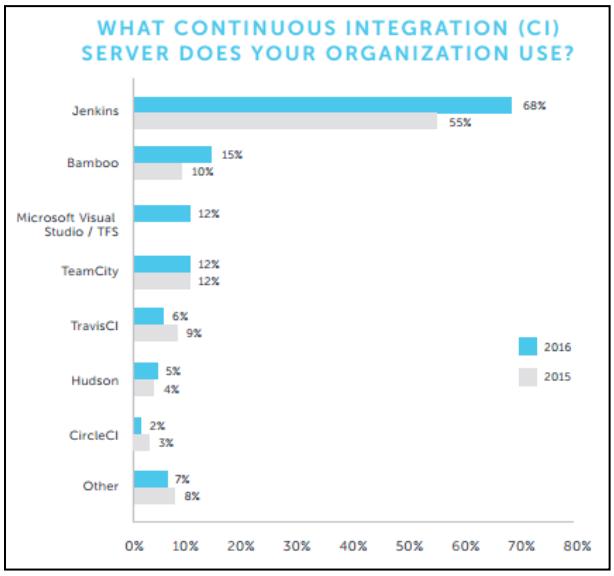

Fig.7 Use of Continues Integration servers

6) Cross Browser testing is important in large organizations

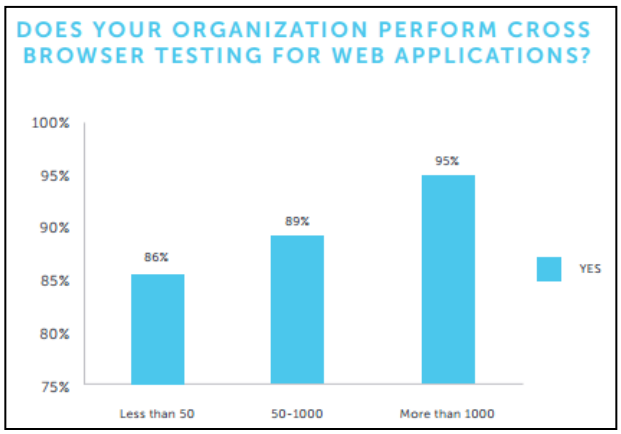

Fig.7 Cross Browser Testing in organizations

7) Need for Cross Browser testing remains constant

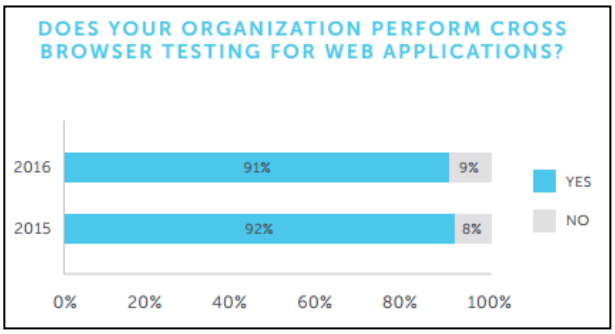

Fig.8 Need for Cross Browser Testing

8) Testing for Microsoft browsers increases

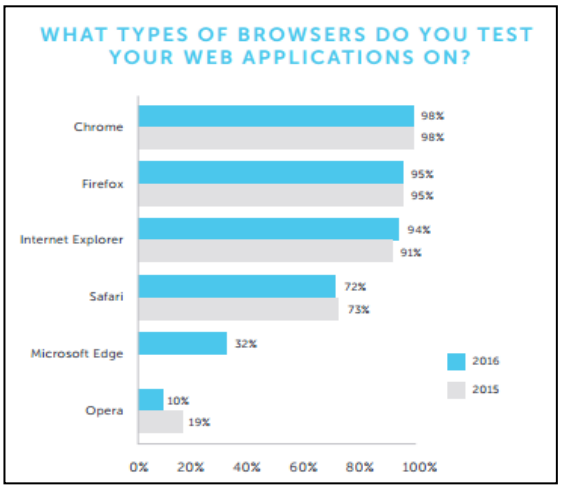

Fig.9 Testing for Microsoft browsers

9) Use of Cloud for Testing remains steady

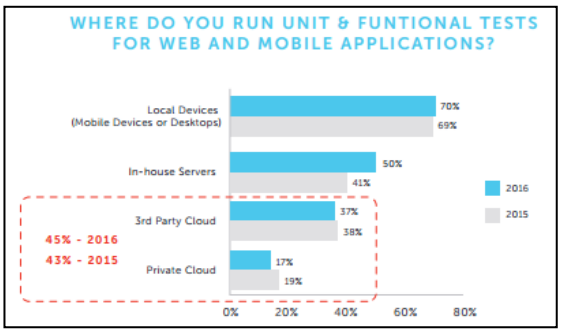

Fig.10 Use of Cloud for Testing

The survey results show the increased adoption of agile development and the amplified importance of online testing environment.

Conclusions

The Product Backlog, Sprint Backlog, Daily Scrum meeting and potential shippable product increment are the key factors to be considered in scrum framework. While developing a web based applications the software professionals should follow all the twelve principles set in agile manifesto and also a quality facilitator role has to be created in order for the successful elimination of the mentioned risks associated in development process. The survey conducted by Sauce Labs show that the success embraces at the enterprise level of agility in web application development.

References

Bowen, Janine Anthony (2011). Cloud Computing: Issues in Data Privacy/Security and Commercial Considerations. *Computer and Internet Lawyer Trade Journal,* 28 (8), 8.

Coram, M., and Bohner, S. (2005), The Impact of Agile Methods on Software Project Management, In *Proceedings of the 12th IEEE International Conference and Workshops*

on the Engineering of Computer-Based Systems (ECBS), April 2005, pp. 363-370.

Nerur, S., Mahapatra, R., and Mangalaraj, G. (2005), Challenges of migrating to Agile methodologies, *Communications of the ACM*, Vol. 48, No. 5, May 2005.

https://saucelabs.com/resources/white-papers/sauce-labs-state-of-testing-report-2016.pdf

Mc Breen (2003), P: Quality Assurance and Testing in Agile Projects. *Mc Breen consulting*, 2003, http://www.mcbreen.ab.ca/talks/CAMUG.pdf

Baker, Steven W. (2005) Formalizing Agility: An Agile Organization's Journey toward CMMI Accreditation. *Agile 2005 Proceedings*, IEEE Press, 2005.

Ming Huo, June Verner, Liming Zhu and Mohammad Ali Babar (2004), Software Quality and Agile Methods, *COMPSAC '04*, IEEE 2004.

Craig Larman, Victor R (2003), Basili. Iterative and Incremental Development: A Brief History, 0018-9162/03/© 2003 IEEE

Leffingwell, Dean (2007), Scaling Software Agility: Best Practices for Large Enterprises, *Addison-Wesley Professional*.

Veerapaneni Esther Jyothi, K. Nageswara Rao (2012) Effective Implementation of Agile Practices, *International Journal on Computer Science and Engineering,* Vol. 4 No. 01 Page No. 87

Veerapaneni Esther Jyothi, K. Nageswara Rao (2010) Effective implementation of agile practices – A collaborative and innovative framework, *CiiT International Journal of Software Engineering and Technology*, September2010

E. Arisholm, Gallis, H.E., Dybå, T., Sjøberg, D., (2007) Evaluating Pair Programming with Respect to System Complexity and Programmer Expertise, *IEEE Transactions in Software Engineering*, pp. 65-86.

Alleman, Glen (2005), Agile Program Management: Moving from Principles to Practice. Agile Product & Project Management, Vol. 6 No. 9, Cutter Consortium, September, 2005.

Veerapaneni Esther Jyothi, Kaitepalli Srikanth, K. Nageswara Rao (2012) Effective Implementation of Agile Practices – Object Oriented Metrics Tool To improve Software Quality, International Journal of Software Engineering & Applications, Vol.3, No.4, July 2012.

Anderson, David J (2003), and Eli Schragenheim, Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Prentice Hall, 2003.

Poppendieck, Mary and Tom Poppendieck (2003), Lean Software Development: An Agile Toolkit, Addison Wesley Professional, 2003.

Ming Huo, June Verner, Liming Zhu, Mohammad Ali Babar (2004), Software Quality and Agile Methods, COMPSAC '04, IEEE 2004.

Veerapaneni Esther Jyothi, K. Nageswara Rao (2014) Effective implementation of agile practices – In Collaboration with Cloud Computing, International Journal of Current Engineering and Technology, Vol.4, No.3 (June 2014).

Salem, A. M. (2010), A Model for Enhancing Requirements Traceability and Analysis. International Journal of Advanced Computer Science and Applications - IJACSA, 1(5), 14-21.