Agile Testing with Scrum-A Survey

Chhavi Malhotra   **Anuradha Chug**
USICT, Guru Gobind Singh Indraprastha University   *USICT,Guru Gobind Singh Indraprastha University*
Sector - 16C Dwarka, Delhi - 110078, India   *Sector - 16C Dwarka, Delhi - 110078, India*

Abstract: Agile testing deals in carrying out work rapidly and successively in small iterations. As against other traditional approaches like waterfall model where testing used to take place very late after coding , this approach is more flexible and practically adaptable as here testing start as early as requirement and planning phase. Scrum attempts to build the work in short iterations where each iteration consists of short time boxes. Automation can serve to be very fruitful at many phases of scrum methodology. It can benefit not just at the time of test execution but at the time of managing the various activities of scrum. It can serve to provide the solution rapidly by increasing reliability, repeatability, comprehensiveness and efficiency which is the soul of agile. This paper reviews several papers on agile, its methodologies including the testing aspects of it and also automation in agile and various techniques to it. Survey concludes that Automation is a viable solution in Agile and its methodologies including Scrum due to its very iterative nature. This paper will help researchers get a view of automation in agile testing, various techniques in which it can be done, scrum incorporated with agile testing in an easy and effective manner.

Keywords: Scrum, Crystal, DSDM, Selenium, RAD, KDT, TDD

I.   Introduction

In agile development, testing is involved throughout the development lifecycle, testing the software throughout the development. Developer is taught of the testing techniques (white box and black box as well). Agile testing experts expect to work in a team that emphasizes cutting to chase. Testers do not have a testing phase; however the developers are engaged in testing tasks performing testing by writing unit tests and testing it with the help of automated tools. Our goal is to provide better quality software, yet following the agile manifesto. With automated unit testing it becomes easier to validate and test the individual build features and also it is important to expect the build daily so that the integration can be done as and when along with the new build arrival [2].

The importance of agile comes from the fact that the product is to be kept ready in released shippable condition, so that it can be delivered whenever it is appropriate [5]. But testing is not only done by developers, testers have an equal role to play and they cannot be left behind as the developers cannot test their own code with precision. The role of tester is more of a quality assurance individual involved in testing from the beginning. However, there may be ambiguity that may take place amongst the testers because the specifications and requirements are rapid and if they are delivered on time it's easier to test but many of the time they arrive much late or are not formally written as its agile environment. More problems may arise if the requirements are there but aren't well written and documented. This paper provides a review on testing methodology carried out in agile over the years covering various agile methods.

Agile methodology has gained popularity starting year 2000 wherein delivery in quick succession, iterativeness and incremental growth gained popularity for any product. Agile methods are gaining pace day by day due to the many advantages that this methodology comes with. Agile values customer over contract. It produces working software at the end of each iteration. It values individuals over processes and methods. Agile is that methodology where change happens rapidly. The transition of testing practices as was in traditional approaches like waterfall approach to agile approaches is taking place rapidly with the increase in rapidness in the product delivery.

II.   Review Process

A review was carried out in order to look into the work being carried out in the field of agile preferably the testing part of it. After reviewing papers on agile process, agile progression down the years, agile testing, Methodologies of agile and automation in agile we were able to answer several questions which are listed in Table 1. Several researchers, industry experts have their work carried forward in agile methodology along with testing in agile which is highlighted below :-

A. Testing with Agile (RQ1)

Testing in agile should be carried out iteratively [4]. She concludes that whether it is reviews in static testing or execution time dynamic testing each and every activity must be practiced iteratively. Testing is the only activity in development process that can be segregated from other agile methods.

Agile projects go through an initiation phase (Iteration 0) where we set the base for the project, a construction phase where we code the software, in end game phase we transform and prepare the system for release and a production phase where we operate the system and support users [6]. During iteration0 we perform initial setup tasks which involve identifying the people who will be on internal testing team, identification and installation of testing tools. Also if the project has a deadline then you identify the date on which the project will enter an End Game. A great amount of testing takes place at construction phase, which means more of testing at unit level. Testing activities do not end up here, testers are still required at later on stages like integration testing.

TABLE 1. REVIEW QUESTIONS

	Review Question	Main Motivation
RQ1	What is Agile Testing?	Identify testing being carried out using agile methodology.
RQ2	How does QA Team transition from Waterfall to Agile?	Identify team progression from traditional approach such as waterfall to more modern approach like agile.
RQ3	What are the various methodologies of Agile?	Identify various methodology and practices of agile used today.
RQ4	What is scrum Process?	Identify which is the widely used practice of agile.
RQ5	How has agile testing progressed down the years?	Identify progression of agile testing down the years.
RQ6	What is agile automated testing?	Identify the areas and field of automation with agile along with widely used tools.
RQ7	How does defect handling takes place in agile?	Identify process of defect management.

Planning activities, documentation of requirement still take place but our focus is on testing activities. During the end game final system testing takes place which will define the effort for release, including system as well as acceptance testing. If testing has already taken place effectively in construction phase, final testing efforts will prove to be straightforward and quick. If any kind of serious testing takes place at End Game phase, then the team may be in trouble as there might not be sufficient time to act on those defects. Majority of testing occurs during construction iteration on agile projects. Testing effort revolves around construction. There are two construction iterations, indicating there is a confirmatory testing performed by the team, and investigative testing efforts performed by independent test team. Confirmatory

Testing focuses on verifying that the system fulfills the intent of the stakeholders as described to the team, whereas investigative testing strives to discover problems that the development team didn't consider. There are two aspects of confirmation testing, agile acceptance testing and developer performing testing, both of which is automated to enable continuous regression throughout the life cycle. The acceptance tests are based on specification. Acceptance testing is a mix of functional testing and acceptance testing as the development team and stakeholders are doing it collaboratively. Testing by developer is a mix of traditional unit testing and traditional integration testing. Testing by developer strives to verify both application code and database schema. Goal here is to look into coding errors, still do coverage if not full path coverage, and also to ensure that system meets the current intent of stakeholders.

There is a significant value to be gained by submitting the system to an independent test team at intervals throughout the lifecycle so that they can verify the quality of work. Agile teams produce working software at the end of each working iteration. A common practice is to produce a working set of software after every construction iteration. Also to produce a working set of software every once a week, regardless of iteration length .The investigative test team's goal should be to ask, "What could go wrong," and to explore potential scenarios that neither the development team nor business stakeholders may have considered. They are attempting to ask the question, "Is this system of any help?", "Does the system fulfill the written user requirement?" The confirmatory testing effort verifies whether the system fulfills the intent, so simply repeating on that work isn't going to add much value.

Performance testing of agile projects is also critical. The requirements are an important part of performance projects.TDD can even be used to verify whether performance requirements have been met[22]. Practically the approach involves a well

defined and structured methodology. The creation of multi skilled teams and implementing performance testing team in development methodology will help in testing non-functional requirements in agile[21].

B. Transition from waterfall to Agile (RQ2)

Going into waterfall to agile transition is not an easy task. When automation is implemented where traditional approaches were the flavor of the day is also challenging. It is not easy to implement automation in an organization that is working solely with waterfall approach and moving into agile testing with it. The obstacle is the embedding of business logic into presentation layer due to which it wasn't possible to start automation before developers would separate business layer from presentation layer [11].

One of the challenges in transitioning to agile approach is converting the mindset of people [16] . The change is difficult to adapt and has to be started from the beginning of development lifecycle. The project tasks are divided into phases to be carried on week-by week basis. In first week analysis and design of the problem. In the second week design, coding and unit testing and in the third week with coding and unit testing. Gradually following the cycle into fourth and fifth week with coding, unit testing and system integration and sixth week with system integration testing. The conclusion drawn by them suggest that the change from traditional to agile approaches is not easy and should be planned. People not adapting to the change should not be included further in the team.

Limitation and challenges of traditional approach and integration of them with agile approach is not easy. Some of the challenges being ambiguity in requirements, lack of customer involvement at various onsite projects, conflicts in development process, people conflicts. There is also a broad comparison done of various development methods of developing software [17].

The transition from traditional approach of testing into agile approach incorporating practices of user stories, scrums and time-boxed iterations [18]. There are many problems highlighted with some of them proposing lack of automated regression tests, shortage of skilled people ,to build automation development environment for various development methods. Scrum has been a useful approach . Without agile Automation will not be possible so automation is the viable solution in order to implement agile.

C. Agile methodologies (RQ3)

Several methodologies of agile like Extreme programming also called XP in short as coined by Kent Beck is very popular and intuitive approach in agile. It also promotes customer involvement as proposed in agile where customer is utmost priority. It promotes continuous testing, continuous feedback and planning and close teamwork to deliver working software at frequent intervals. In XP customer works with development team also referred to as "User Stories". The development team delivers highest priority tested working software on iteration by iteration basis [2].

Crystal methodology is most lightweight approaches of software development. Crystal is comprised of family of agile methodologies such as crystal clear, crystal yellow, crystal orange whose unique characteristics are governed by team size and project priorities. The crystal family talks about the realization that each methodology addresses set of policies, procedures and practices. Several features of crystal include team-work, communication and simplicity. It also promotes early delivery of software project.

Dynamic system development method (DSDM) dates back in 1994 grew popularity in need to provide an industry standard project delivery framework that was referred to as rapid application development. RAD was very popular in 1990. The RAD approach to software delivery revolved in very unsystematic manner. As a result of which the DSDM consortium was created with a goal of devising and creating a common industrially led framework for rapid software development. DSDM revolves around nine key principles that are primarily based on business needs and values, active user involvement, frequent delivery, integrated testing and stakeholder involvement. Requirements are baseline at a high level early in the project. Requirements are planned and delivered in time boxes also referred to as iterations. Within each iteration, less critical items are included so that if necessary, they can be removed to keep on impacting higher priority requirement on schedule.

With Scrum, projects progress through a series of month long iterations called sprint. There is building of product backlog, sprint backlog. Daily 15 minute meeting is held which is called scrum wherein all the work done during the ongoing iteration is addressed.

Comparison for using XP or scrum with agile depends on factors like which method gives the best coverage for all the developmental activities and agility degree of each method should be assessed[25].

D. Scrum Process (RQ4)

Scrum approach has been developed for managing the systems development process. It's an empirical approach applying the ideas of industrial process control theory to systems development resulting in an approach that reintroduces the idea of flexibility, adaptability and productivity. Scrum concentrate on how the team members should function in order to produce the system flexibly in constantly changing environment. The main idea of scrum is that system variables involve several technical and environmental variables (e.g. Requirements, time frame, and resources) that is likely to change during a process. Scrum is a set of guidelines that govern the development process of a product, from its design stage to its completion. Scrum is implemented with the help of Sprint.

During each sprint, the team creates finished portions of product. Features that go into a sprint come from the *product backlog* decided by product owner, which is a prioritized wish list comprising of user stories which is an ordered list of requirements. Which backlog items go into the sprint (the sprint goals) is determined during the sprint meeting. During this, the Product Owner informs the team of the items in the product backlog that he or she wants completed. The team then determines how much of this they can commit to complete during the next release, and records this in the *release backlog*. User stories are time lined and defined ranging from 8 hours, 4 hours and from hours to days here. Sprint planning meeting is conducted taking the release backlog and making a sprint backlog constituting a *Sprint backlog* team to create estimates. These estimates give an idea of total amount of work involved. Initially the estimates are less accurate but improve from sprint to sprint[24].The sprint backlog is property of the whole development team in which they decide during a sprint; no one is allowed to edit the sprint backlog except for the development team. The sprint goals should not be changed during the sprint. Development is time scheduled such that the sprint must end on time; if requirements are not completed for any reason they are left out and returned to the product backlog. Daily meetings are held called daily scrum after a sprint is completed. Creation of self organizing teams is encouraged by scrum by where co-location of all team members and verbal communication between all team members and disciplines in the project is decided. Scrum can be implemented only through a wide set of tools [8,9,20].

Alternating sprints in scrum process management and development helps in a great deal in managing scrum process. One of the other challenges in implementing scrum is lack of structured requirements. The structured agile refinery approach can serve to be helpful. Daily scrum meetings are an important part of scrum also which should be practiced rigorously.[19]

E. Agile Testing down the years (RQ5)

Meta- analyses prove to be the viable solution to address the issues on Test Driven development and many other approaches in agile. Methods other than extreme programming proves to be a good solution. Various tools have been used in the past by academicians as well as by industry practitioners down the years. Many versions of xUnit have been used for different programming languages. However the focus has been on unit and acceptance testing of code. In the near future other tools providing support in the testing process is being expected [14].

F. Automated Agile Testing (RQ6)

Automation is very integral part of agile methodology. As in agile testing starts from early phased there are tools available that build test cases from requirements in the initial phases. Tester's work deals in designing test cases keeping the environment in mind and executing them on the build with the actual data at the time of execution. It can benefit not just at the time of test execution but at the time of managing the various activities of scrum. Automation in Agile testing follows a cycle, understand how testing takes place, identify the tool and technology that can be useful for testers, deliver that solution in less than a week and repeat many times [3]. It's not easy to implement automation in an organization that is working solely with waterfall approach and moving into agile testing with it. The obstacle that poses is the embedding business logic into presentation layer due to which it wasn't possible to start automation before developers would separate business layer from presentation layer. Automation in testing not only provides regression but also helps in carrying out work in iterations.

Automation testing is a viable technique in order to gel with agile framework. There are also some best practices proposed that may serve to be very fruitful with software development process. Automation will help to avoid any mistakes which can otherwise be possible manually [1].

Testing activities appear in a traditional way. It is important to carry out testing proactively from the start as early as possible in development cycle. Test team should be involved in not only detecting defects but also in preventing it. Automation is a solution to provide complete test coverage using scripts that have ability to carry out work iteratively [5].

Automated regression testing in agile is also common these days as the work is not possible manually in short iterations after changes [7]. It is also useful in case of requirement changes as a small change in requirement leads to many tests failing. There are several dimensions to carry out automation consisting with regression. While some focus on adding granularity to system under test., creation of testing approaches comprising of record-playback approach and manual testing approach testing the interface with user interface and internal software interface.

Projects tend to move form hardware to software as going into more depth and realization and providing more flexibility. The approach makes the system more complex but the system is built more from user perspective. Software testing is a key element in Traditional approaches starting form W-model to agile practices including Scrum. In case of Test driven development, test first development is practiced wherein test scenarios are identified and tests are built from requirements and user specification. Keyword driven testing is also followed which is an extension of data driven tests wherein both manual and automation strategy prevails. Fitnesse can be a help in devising a framework for SUT and Test management system. The keyword (KDT) approach extends by defining keywords. KDT approach enables a systematic testing process in order to implement execution of test cases [10].

Test First development (TDD) works on 4 principles. Selection of requirements and building test cases based on those requirements. Implementation of those test cases takes place with their execution, test cases should fail as there is no code available. Cyclic implementation of test cases takes place until successful completion. Refactoring is the next step in which the test cases are executed on the same functionality. The step is followed in a cyclic manner again and again. There are

various levels in testing as well starting form system level testing wherein acceptance tests and system tests are executed by customers, architecture level tests focusing on integration and component level tests for individual components. Building an automation test framework consists of building a Test suite , system under test . Test runner which applies the test cases in test suite to system under test which provides feedback in form of test results [12].

Selenium is an open source tool that incorporates several set of tools in it and can work with agile projects. It has the ability of execution of tests on multiple browsers. It offers selenium IDE which allows the flexibility to handle the selenium script made under different languages to run under different browsers. Selenium grid allows building large test suites to run on multiple environments. Building of functions including integration with continuous integration server also helps. Jenkins can serve to be an integration tool in order to build tests [13].

Automation is an important phase of construction phase in agile testing due to increased need of regression testing [6]. There are a number of different tools and technique which help to do acceptance testing including various requirement elicitation techniques in order to gather requirements. Some of them being use cases, UML diagrams, scenario descriptions etc. However, the tools selection is also critical and important so as its selection. Tools needs to evolve as the needs of teams in agile change[23].

G. Defect management(RQ7)

The process of defect management using a two way approach using a defect tracking system and a backlog tracking system has been proposed [2]. A defect tracking system helps to detect defect, causes, variances while a backlog tracking system which keeps a track of bugs and features in product.TDD, continuous integration tend to get minimum defect .A bug may take a role of story if it is too big to be fixed in the same iteration or if it's a bug for which requirements are not clear.

Defect stories are used at the end of testing to describe the potential problems. It can be estimated and prioritized and put on requirement stack. So, at the end the only requirement type which needs to be working on is defect stories [6].

H. Continuous integration in Agile (RQ6)

Continuous integration became popular with extreme programming. It involves increasing the execution time of integration. It promotes automated source repository when developers start integration. The first step in continuous integration is to put forward continuous chain of events. The first step is to get a latest source code in a compiled format. Unit tests are continued if compilation does not fail. If unit testing does not fail then acceptance tests are executed in test environment set up previously. If acceptance test does not fail the build is published to a specific public location. At last the email is sent to everyone in the team notifying the number of tests run and build version [15].

III. Analysis

After reviewing the papers it was found, that there has been a gradual increase in papers on agile with scrum methodology over the years. With 2008 wherein papers were the maximum followed by 2011 and 2012. As depicted in Table 2.

TABLE 2. PAPERS SELECTED BY YEAR

Years	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012
Paper	1	2	1	1	2	2	4	2	3	3	4
%	4%	8%	4%	4%	8%	8%	16%	8%	12%	12%	16%

Form Figure 1 and Table 3 it is clear that out of the total papers reviewed there were good amount of papers available for Automated Agile Testing which concluded that it's not easy to implement automation in an organization that is working solely with waterfall approach and moving into agile testing with it [3]. Automation is a solution to provide complete test coverage using scripts that have ability to carry out work iteratively [5], automation is only possible with agile [1]. Automated regression testing in agile is also common these days as the work is not possible manually in short iterations after changes [7]. In case of Test driven development, test first development is practiced wherein test scenarios are identified and tests are built from requirements and user specification[10]. Selection of requirements and building test cases based on those requirements is the main objective [12]. Implementation of those test cases takes place with their execution, test cases should fail as there is no code available. Constant refactoring is available in order to handle this. In automation certain tools are gaining popularity like selenium [13]. Automation is an important phase of construction phase in agile testing due to increased need of regression testing [6]. However, the usage of tools in agile and their selection should be proactive [23]

Next in the line were the papers on transition from waterfall to Agile. Transition from waterfall to agile is complicated and complex process. Implementing automation into agile is not an easy task where business layer is embedded with presentation layer which should be separated first [11]. Changing the mindset of people when transitioning from waterfall to agile is a tough challenge [16]. Project tasks are divided into phases to be carried on week-by week basis. While [17] proposes limitations and challenges of traditional approach and integration of them with agile approach**.** The transition from traditional

approach of testing into agile approach incorporating practices of user stories, scrums and time-boxed iterations [18] also takes place.

Papers on agile testing, agile methodology , scrum process and defect handling and management were next in order in which [2] has proposed many agile approaches including scrum, DSDM and crystal just to name a few. Agile testing is a complete Scrum approach has been developed for managing the systems development process. Scrum is a set of guidelines that govern the development process of a product, from its design stage to its completion. Scrum is implemented with the help of Sprint wherein finished portions of the product are prepared on iterations basis [8, 9, 20].There are certain limitation and challenges of agile in implementing scrum which should be overcome[19].

Meta- analyses prove to be the viable solution to address the issues on Test Driven development and many other approaches in agile. Other tools providing support in the testing process is expected [14].

There were papers on agile methodologies and continuous integration in agile which provided idea of various methodologies of Agile [2] with also a comparison to use scrum or XP in providing agility for project[25].

Complete process of continuous integration which became popular with extreme programming [15] has also been highlighted.
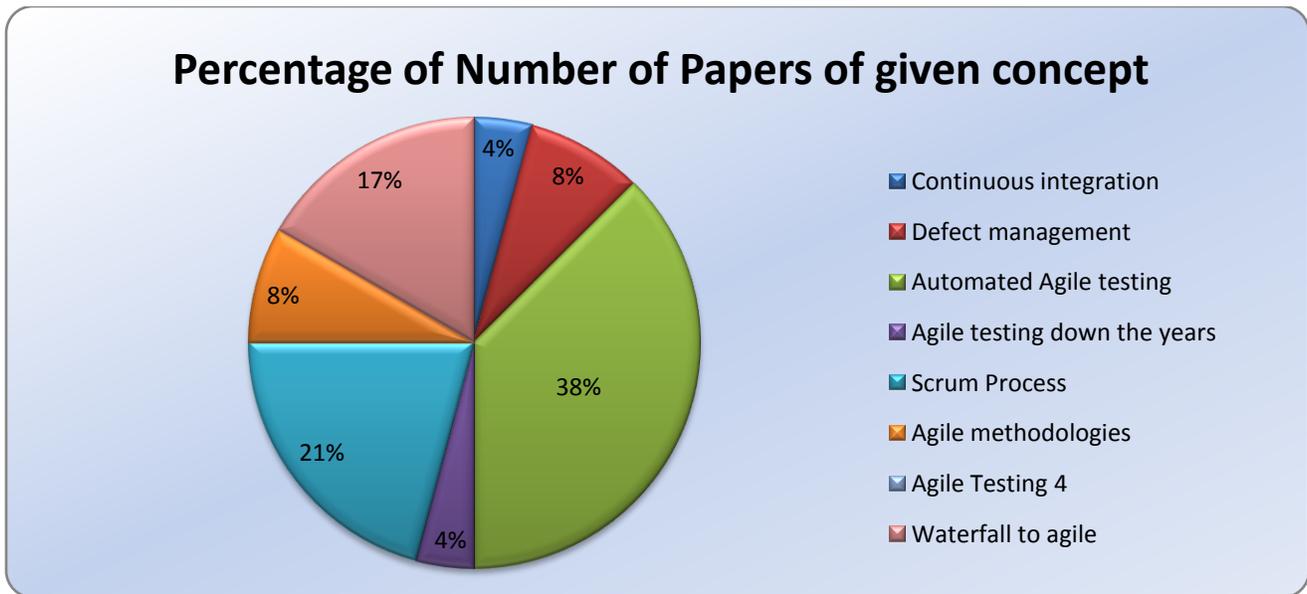

Figure 1. Number of Papers of given concept

IV.   Conclusion

After reviewing papers on agile methodologies, agile testing, progress in agile down the years and scrum process. It was evident from the papers that automation is a viable solution in Agile. The reason being that agile approach which is rapid and involves continuous integration cannot be solved manually. Automation which is reliable, repeatable, comprehensive and programmable fits to the need of agile testing and various methodologies of agile.

It is also evident that scrum methodology which requires the use of tools can be solved with automation since the methodology involves making of product backlog, sprint backlog, release backlog and carries work on iteration basis. As it also involves tracking the progress with the help of burn down charts it will be very tedious to carry that work with manual methods. There are scrum management tools available that can serve to be an ideal solution in managing activities of scrum including progress tracking, reviewing and planning activities.

We can also make out that industrially scrum methodology of agile is very popular as much of the work is done by industry practitioners. There is however, requirement of more research papers into this area particularly in the automation part of scrum and testing incorporated with it.

TABLE 3. NUMBER OF PAPERS IN REVIEW

Concepts / Techniques	Number of Papers of Concepts / Techniques	Author(s)

Agile Testing	4	1. Leanne Howard[4]
		2. Scott Ambler[6]
		3. Chih-Wei Ho, Michael J. Johnson, Laurie Williams1,E. MichaelMaximilien[22]
		4. Jamie Dobson[21]
Transition from Waterfall to Agile	4	1. Megan Sumrell[11]
		2. Kalpana Sureshchandra, Jagadish Shrinivasavadhani[16]
		3. Woi Hin, Kee[17]
		4. Susan D. Shaye[18]
Agile methodologies	2	1. Shrini Kulkarni [2]
		2. Joao M. Fernandes and Mauro Almeida[25]
Scrum Process	5	1. Ken Schwaber and Jeff Sutherland[8]
		2. Pete Deemer, Gabrielle Benefield, Craig Larman[9]
		3. V. Mahnic[24]
		4. Kevin Vlaanderen, Slinger Jansen , Sjaak Brinkkemper , Erik Jaspers[19]
		5. Ken Schawaber[20]
Agile Testing down the years	1	1. Theodore D. Hellmann, Abhishek Sharma, Jennifer Ferreira
Automated Agile Testing	9	1. Harish R Madhu B K Lokesha V[1]
		2. James Bach[3]
		3. Eliane Collins, Arilo Dias-Neto, Vicente F. de Lucena Jr.[5]
		4. ScottAmbler[6]
		5. Gerard Meszaros[7]
		6. Reinhard Hametner , Dietmar Winkler Alois Zoitl[10]
		7. Reinhard Hametner, Dietmar Winkler, Thomas Östreicher, Stefan Biffl, Alois Zoitl[12]
		8. Rosnisa Abdull Razak **,** Mimos Berhad, Fairul Rizal Fahrurazi[13]
		9. Fabrizio Cannizzo, Gabriela Marcionetti, Paul Moser[23]
Defect Management	2	1. Shrini Kulkarni[2]
		2. Scott Ambler[6]
Continuous integration	1	1. Sean Stolberg[15]

REFERENCES

[1] Harish R Madhu B K Lokesha V "A Sophisticated Study on Best Practices of Agile Software Testing" International Journal of Electronics Communication and Computer Engineering Volume 3, Issue (1) NCRTCST, ISSN 2249 – 071X ,2012

[2] Shrini Kulkarni "Agile Development/Testing – Is history repeating itself?" 5th International Software Testing conference February 21- February 22, 2005 Hyderabad, India.

[3] James Bach "Agile Test Automation" ,2003

[4] Leanne Howard *"Agile-Why the fear?" Planit:* Sydney http://agiletesting.com.au/,2009

[5] Eliane Collins, Arilo Dias-Neto, Vicente F. de Lucena Jr. "Strategies for Agile Software Testing Automation: An Industrial Experience" 2012 IEEE 36th International Conference on Computer Software and Applications Workshops

[6] ScottAmbler *"Agile Testing Strategies" www.ambysoft.com/* scottAmbler.html,2002

[7] Gerard Meszaros "Agile Regression Testing Using Record & Playback", ACM 1-58113-751-6/03/0010, 2003

[8] Ken Schwaber and Jeff Sutherland "Scrum: Developed and sustained", 2008

[9] Pete Deemer, Gabrielle Benefield, Craig Larman "The Scrum Primer" Version 1.2 ,2010

[10] Reinhard Hametner , Dietmar Winkler Alois Zoitl "Agile Testing Concepts Based on Keyword-driven Testing for Industrial Automation Systems" IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society 25-28 Oct. 2012

[11] Megan Sumrell "From Waterfall to Agile – How does a QA Team Transition?", IEEE Computer Society, Agile Conference(AGILE), 2007

[12] Reinhard Hametner, Dietmar Winkler, Thomas Östreicher, Stefan Biffl, Alois Zoitl "The Adaptation of Test-Driven Software Processes to Industrial Automation Engineering" 2010 8th IEEE International Conference ,ISBN: 978-1-4244-7298-7 ,Page(s): 921 - 927

[13] Rosnisa Abdull Razak , Mimos Berhad, Fairul Rizal Fahrurazi "Agile Testing with Selenium" 2011 5th Malaysian Conference in Software Engineering (MySEC)

[14] Theodore D. Hellmann, Abhishek Sharma, Jennifer Ferreira "Agile Testing: Past, Present, and Future- Charting a Systematic Map of Testing in Agile Software Development" 2012 Agile Conference, ISBN: 978-1-4673-2622-3, Page(s): 55 – 63

[15] Sean Stolberg "Enabling Agile Testing Through Continuous Integration" 2009 Agile Conference , ISBN: 978-0-7695-3768-9 ,Page(s): 369 - 374

[16] Kalpana Sureshchandra, Jagadish Shrinivasavadhani "Moving from Waterfall to Agile", Agile 2008 conference

[17] Woi Hin, Kee "Future Implementation and Integration of Agile Methods in Software Development and Testing", 1-4244-0674-9/06 ©2006 IEEE

[18] Susan D. Shaye "Transitioning a Team to Agile Test Methods", 2008 Agile conference.

[19] Kevin Vlaanderen, Slinger Jansen , Sjaak Brinkkemper , Erik Jaspers "The agile requirements refinery: Applying SCRUM principles to software product management" Information and Software Technology 53 (2011) 58–70

[20] Ken Schawaber "Agile Project management with Scrum" Microsoft Press,2004

[21] Jamie Dobson "Performance Testing on an Agile Project" IEEE Computer Society AGILE 2007

[22] Chih-Wei Ho, Michael J. Johnson, Laurie Williams1,E. Michael Maximilien "On Agile Performance Requirements Specification and Testing" Proceedings of AGILE 2006 Conference ,IEEE Computer Society,2006

[23] Fabrizio Cannizzo, Gabriela Marcionetti, Paul Moser "Evolution of the Tools and Practices of a Large Distributed Agile Team" Agile 2008 Conference, IEEE Computer Society 978-0-7695-3321-6/08

[24] V. Mahnic "A Case Study on Agile Estimating and Planning using Scrum" ELECTRONICS AND ELECTRICAL ENGINEERING, ISSN 1392 – 1215,2011. No. 5(111)

[25] Joao M. Fernandes and Mauro Almeida "A Technique to Classify and Compare Agile Methods", Springer-Verlag Berlin Heidelberg ,XP 2010, LNBIP 48, pp. 385–386, 2010