

The Agile Methodology

Charul Dewan¹, Rishabh Jain², Ruchi Kohli³

¹Assistant Professor, Northern India Engineering College, Delhi
charularora@gmail.com

²Assistant Professor, Northern India Engineering College, Delhi
jain2986@gmail.com

³TS InfoTech Pvt. Ltd. Gurgaon, Haryana
ruchikohli@gmail.com

Abstract

The technologies are numerous and Software is the one which is most widely used. Some companies have their own customized methodology for developing their software but the majority speaks about two kinds of methodologies: Traditional and Agile methodologies. In this paper, we will discuss some of the aspects of what Agile methodology is, how it can be used to get the best result from a project, how do we get it to work in an organization.

Keywords: Agile, Heavy Weight Approach, Scrum.

I. INTRODUCTION

Software Engineering is the application of a systematic, disciplined, quantifiable approach to the development, operations, and maintenance of software, and the study of these approaches; that is the application of engineering to software. [1]

The Heavyweight methodologies come under traditional software and are also known as Work Down approach. It is based on comprehensive planning, detailed documentation, and expansive design. Whereas Agile methodology follows the rules of the agile manifesto, treating software development as the customer of testing. Agile testing involves testing from the customer perspective as early as possible, testing early and often as code becomes available and stable enough from module/unit level testing. Since working increments of the software is released very often in agile software development there is also a need to test often. This is often done by using automated acceptance testing to minimize the amount of manual labor. Doing only manual testing in agile development would likely result

in either buggy software or slipping schedules because it would most often not be possible to test the whole software manually before every release.

II. ISSUES RELATED DUE TO LACK OF CORRECT PROCESSES FOLLOWED

If proper measures will not have been followed, then outcome will be unexpected and will lead to loss of wealth, lives. Different types of hazards have occurred in various fields and People faced so many problems due to these hazards. A lot of confusion created and that lead to inefficient misunderstandings and worthless chaos. Some of the examples are mentioned below.

- A) Computer system problems at one of Japan's largest banks resulted in a nationwide ATM network of more than 5,600 machines going offline for 24 hours, internet banking services being shut down for three days, delays in salary payments worth \$1.5 billion (£939 million) into the accounts of 620,000 people and a backlog of more than 1 million unprocessed payments worth around 9 billion (£5.64 billion).[2]
- B) An Australian bank began giving out large sums of money from 40 cash machines across one city. Officials at the company said they were operating in stand-by mode, so could not identify the account balances of customers.[3]
- C) People have suffered so many losses in their assets due to the account systems not functioning properly in the Finance Department. Delays in payments and when payments were received, they were not accurate. Error in Fund transfers, interests and

sometimes hacking issues have created a lot of pain to the customers.

- D) Software problems in the automated baggage sorting system of a major airport in February 2008 prevented thousands of passengers from checking baggage for their flights. It was reported that the breakdown occurred during a software upgrade, despite pre-testing of the software. The system continued to have problems in subsequent months.[4]

III. LITERATURE REVIEW

In the past few years, Jokela, T. and Abrahamsson, P [5] have proposed several approaches for extreme programming and mentioned that agile leads to a successful project but the results will be great if mixed with UEX(user Experience). He mentioned that agile teams are made up of generalizing specialists with the ability to work on a wide range of things, instead of narrowly focused specialists which are found on traditional teams. Jeff Patton mentioned that UEX combined with agile will resolve many issues related to software development. Stober Th. And Hansmann, U. [6] have mentioned how to get metrics and estimates in software products using the combination of XP and Scrum. They have mentioned how to simplify Team setup, Dependencies, Tools, and Processes by reducing the complexity of a system. Magnus ThorsteinSletholt [7] mentioned how well do current scientific software Development processes match the practices found in agile development methods, and what are the effects of using agile practices in such processes.

MaaritLaanti, OutiSalo, PekkaAbrahamsson [8] presented how agile methodology is replacing traditional methodology at Nokia and this has been achieved because of increased quality and transparency, and earlier detection of defects. They have mentioned that people working on projects using agile don't want to revert back to old methods like waterfall. But they have not mentioned anything about "Fixed Bid" projects where only traditional method can work. Scott W. Ambler [8] says that it is unethical for teams to do fixed bid projects but the teams can go for agile fixed bid projects. He has not mentioned about how agile fixed bid projects will result into a product or system which will handle "Respond to changes" at any point of time.

IV TRADITIONAL SOFTWARE TESTING VS AGILE TESTING

Agile testing was different in many ways from 'traditional' software testing. The biggest difference is that on an agile project, the entire development team takes responsibility for quality. This means the whole team is responsible for all software testing tasks, including acceptance test automation. When software testers and programmers work together, the approaches to test automation can be pretty creative. Software Testing is integrated into software development. There are discussion session in which all project members participate and discuss all the related queries and queries are answered within a couple of minutes which would otherwise take one or two days. Hours of time could be saved by overhearing a conversation in the team room. Traditional methodology cannot cope up with frequent changes but agile methodology can.

A Process followed in Heavyweight

A requirements document is received for review. After that one get a requirements document that is considered base lined or signed-off .These requirements are analyzed to create test conditions and test cases .Then one write his own test procedures and then wait for a piece of software to miraculously appear in the test environment. After this, one start executing his tests .Then begin re-executing some of these tests as we now start iterating through new builds which are released to fix bugs or they may even include new functionality .Then one reach the acceptable risk and the software is released.

The above mentioned distinguished features of various Agile technologies along with the key aspect that it responds to change at any point of time makes it successful and limits Heavyweight Methodology. There are various parameters on which agile scores above the heavyweight and the key points are:

Features	Agile Methods	Heavyweight Methods
Approach	Adaptive	Predictive
Success Measurement	Business Value	Conformation to plan
Project size	Small	Large
Management Style	Decentralized	Autocratic
Perspective to Change	Change Adaptability	Change Sustainability
Culture	Leadership-Collaboration	Command-Control
Documentation	Low	Heavy
Emphasis	People-Oriented	Process-Oriented
Cycles	Numerous	Limited
Domain	Unpredictable	Predictable
Upfront Planning	Minimal	Comprehensive
Return on Investment	Early in Project	End of Project
Team Size	Small/Creative	Large

B Process followed in Agile Methodology

In Agile methodology, these basic principles are being followed.

- (a) Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale. [9]
- (b) Business people and developers must work together daily throughout the project.
- (c) Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage
- (d) Business people and developers must work together daily throughout the project
- (e) Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done
- (f) Working software is the primary measure of progress
- (g) Continuous attention to technical excellence and good design enhances agility
- (h) At Regular Intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.
- (i) Experiment shows that by comparing the qualities of output pages with those of the original downloaded pages. As the number of iterations increases better results are obtained still with reasonable execution time. The small size of pages limits chances of improving the page qualities and reducing execution time at a specified number of iterations. It should be noted that the results depends on the preparation methods of constituting pages under test.

V Conclusion and Future Work

In future, I plan to combine various methodologies of agile with heavyweight and see how different techniques can be applied in it. I want to develop an object oriented agile and heavyweight framework focusing on advantages of both the technologies.

VI. REFERENCES

- [1].SWEBOK executive editors, Alain Abran, James W. Moore; Guide to the Software Engineering Body of Knowledge - 2004 Version. IEEE Computer Society. pp. 1–1. ISBN 0-7695-2330-7.
- [2].<http://www.businesscomputingworld.co.uk/top-10-software-failures-of-2011>
- [3].<http://www.businesscomputingworld.co.uk/top-10-software-failures-of-2011>
- [4].www.sereferences.com/software-failurelist.php
- [5].Jokela, T. and Abrahamsson, P. (2004). Usability Assessment of an Extreme Programming Project: Close Co-operation with the Customer Does Not Equal Good Usability. Product Focused Software Process Improvement: 5th International Conference, PROFES 2004 Proceedings, PP 393-407. Berlin: Springer-Verlag.
- [6].<http://www.springer.com/978-3-540-70830-8> “Agile Software Development-Best Practices for Large Software Development Projects”
- [7].MagnuThorsteinSletholt “A literature review of agile practices and their effects in scientific software development”. Proceedings of the 4th International Workshop on Software Engineering for Computational Science and Engineering ACM New York
- [8].MaaritLaanti, OutiSalo, PekkaAbrahamsson Agile methods rapidly replacing traditional methods at Nokia: A survey of opinions on agile transformation “www.sciencedirect.com Original Research article Information and Software Technology, Volume 53, Issue 3, March 2011, Pages 276-290
- [9].<http://www.agilejournal.com>
- [10]. “www.extremeprogramming.org”