A Study of Software Testing Process in Agile Development

Neetu Anand[1], Kartikeya Vij[2]

[1, 2] Maharaja Surajmal Institute, New Delhi

Abstract- Software testing is the most important process to verify the quality of a product. Software testing in Agile development is very complex and controversial issue in literature and industry. Different people have different views about software testing in Agile methods, because most of Agile methods do not focus much on software testing activities. Agile strongly focus on the close customer collaboration, short iterations and frequent deliveries. But when it comes to software testing, then it is challenging, as Agile do not include many destructive testing practices, which are normally required for a quality product. This paper covers the area of software testing process in Agile development. Agile development processes could be more beneficial and refined by adding testing practices and for this purpose; we proposed a concept of an independent integrated software testing team. This research also identifies the practices of Agile development in industry and the critical issues in industry while practicing Agile development. The issues of automated and manual testing, good practices in automation, and how to manage independent testing teams in Agile development are also highly lightened. This paper focus on every aspect of software testing process in Agile development. This research is based on literature reviews and an industrial survey.

Keywords- Software testing, Agile development process, Quality assurance.

I. INTRODUCTION

"Software Testing is the process of executing a program or system with the intent of finding errors or, it involves any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results". Practices of software testing activities give confidence to the companies that the software meets the requirements and will perform all the required tasks. Testing is an essential activity in software engineering that is used to validate whether the system behaves as intended and identifies the malfunctions. It is widely used in industry for quality assurance, as it provides a realistic feedback of the behaviour of the system. Nowadays the software systems are being used to perform critical tasks where the margin of error is really low. So, these systems should be error free and of high quality. Software testing is an important process that can help to achieve software quality assurance. Companies are spending huge amount of money on testing activities. Research shows that more than 50% of the total cost of development is devoted to software testing. Currently Agile is one of the highly practiced methodologies. [3] According to Agile adoption rate survey, Feb 2008, by Scott W. Ambler; 69% organizations are using one or more Agile projects, and Agile success rates: 82% for co-located teams, 72% for non-co-located, 60% for significantly distributed. Agile is an evolutionary approach to software development which is performed in a highly collaborative manner by self-organizing teams that produces high quality software in a cost effective and timely way which also meets the changing needs of its stakeholders. Agile development approach believes in the involvement and frequent communication between the developer team and stakeholders, and regular delivery of functionality.[4] According to Agile development, people are more important than processes and tools; and the customer must be involved in the entire process.[5] Most of Agile methods do not focus much on testing.

II. TYPES OF TESTING

- **Automated Testing-**Automated testing can be defined as a testing in which no human intervention is involved. Test automation depends upon the importance of scenario, if the scenario is not so important then manual is suitable. And also, another reason to automate the test is: when there are hundreds of lines of code to test or there is repetition then automated testing is more suitable to save the time. The process of Automated testing is shown in fig.1
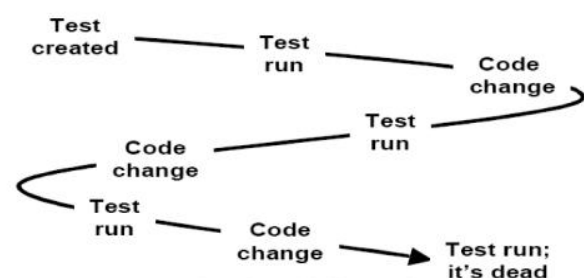

Fig1. Process of Automated Testing

There are some pros and cons of automated testing.

Advantages: -

- Automation is best to run repeatedly tests.
- It gives the ability to main stream scenario and run automation against code that frequently changes to catch regressions in a timely manner.
- Automated tests can be run at the same time on different machines, whereas the manual tests would have to be run sequentially.
- Cost is low when long term testing.

Disadvantages: -

- The cost of automation is more, especially when you writing the tests or configuring the automate framework.
- The visual reference cannot be automated e.g. if the font colour or size can't be defined via code then it is manual test.
- If the tool has limitations then those tests are manual.
- The big con is that it does not find a new bug.
- The cost of test automation is high when its short-term testing.

- **Manual Testing-** Manual testing is vice versa of automated testing, manual testing is a testing in which human interventions are involved. In this testing, test engineers test the code their self. For manual testing a good knowledge of software tester is required and a software tester possess a certain set of qualities i.e. patient, observant, speculative, creative, innovative, open-minded, resourceful, un-opinionated, and skilful.The steps used in Manual testing is shown in Fig2.
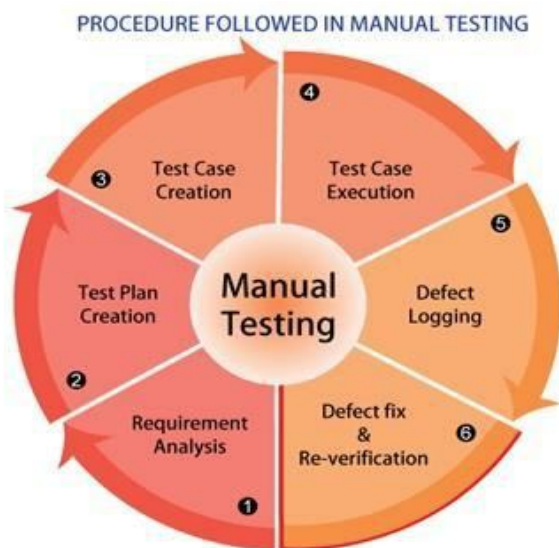

Fig.2. Steps used in Manual Testing

Having all these qualities a software tester can find more bugs, but complete testing is not possible.

Advantages: -

- If a test case runs only twice then it should be manual to save the cost.
- Ad-hoc or random testing is possible for tester.
- More bugs can be found via ad-hoc technique as compare to automation.
- There are no limitations in manual testing.
- You can find new bugs using manual testing.

Disadvantages: -

- Manual testing is more time consuming.
- After every new build code tester must rerun all required tests again, this will be a   huge at the end.
- If testing is long term then manual test is time consuming as well as higher cost.
- To report the first bug to programmer, manual testing takes less time than automated.

- **Manual and Automated Testing in Industry**

We analysed that most of the companies are using both of manual testing and automated testing. Some of them are using automated tools for that and relying more on the automation for catching bugs during the sprints. Company A is using the Bamboo build/test server from Atlassian for continuous integration. They also manually test the product before each official release. [1,2]The customer teams also verify that the updates work on the customer's handsets before sending them new versions. They do have a regression testing before the final release to assure that no more errors are present in the system. Company B is not using any automated tools and according to them, they are relying more on manual testing. They handout prototype builds to a group of internal or external testers. As bugs occur, they report them; the developers try to fix most of them right away, and if there are some harder ones that are difficult to handle by that time, then they are put on the list for next sprint. New feature requests from stakeholders are taken care of directly, or put on the list in the product backlog. Company C is using automated and manual testing for quality assurance. They are using tools like JUnit, for automated testing. They use it while developer is testing his/her own work before sending it to the team lead. The team lead then review his/her work and sends to the testing department for more detailed manual testing. Company D is practicing both automated and manual testing is done, load and stress testing is done using custom made tools

specific to the product and also by using Mercury load testing tools. Functional testing is normally done manually.Company E is using both manual and automated testing techniques. All tests, manual and automated, are documented and stored. The automated tests are stored in test suites and run both as regression tests and functional tests. But the emphasis is on automated tests.

From the above study we analysed that most of the companies in industry are utilizing both of testing ways like: manual and automated testing. The only difference is the use of automation tools. The sample companies were the well-known and bigger companies in the industry. When asked about why to use both testing ways? They answered that they cannot rely on one kind of testing, because automated testing is not enough to get a quality product and if they do only manual testing then it can take a lot of time and effort. In the result of that the project can go out of time and budget. So, that is why they use automated testing to test the application as soon as possible but still there remain some complex problems that need to be addressed through manual testing. During the sprint the developer writes test cases to test his/her own code.

III. SEPARATE TESTING TEAM

The focus of Agile software development is on the interactions between the individuals, close collaboration of customer, short development cycles and the frequent deliveries of the working piece of software. If we see it from the perspective of software testing, then it has some challenging things, and Agile methods have some lacking aspects if we compare it with traditional way of testing. There are some really important aspects of software testing that are overviewed in Agile development, and these practices are the fundamental to a successful quality assurance. The Agile working says that, if the developer and customer have a close coordination with one another then there is no need for a separate testing team.

Basically, the processes of Agile software development are based on iterative and incremental development, which uses development cycles which are short and often time-boxed. The satisfaction of the customer is at the highest priority in Agile development, by providing him/her with short and regular deliveries. [3,4]As the fundamentals of Agile development basis on close interaction of the customer and quick responding to change and less emphasis is given to the processes, tools, documentation, and following a plan. These activities are the traditional way of getting quality product and among the best quality assurance and testing practices. Quality assurance is really an important and a crucial task of most efforts of software development and

testing is one of them. In the traditional way of development, testing has a separate phase at the end of the development, and that includes checking the system for bugs and also integration of different parts of the system. Sometimes this can take a long time and often project goes out of time frame. Defiantly in Agile this process should be different but quality is such an important factor and we cannot sacrifice quality for time constraint. Because in Agile time is a crucial factor, and there is no such time for proper planning for a phase and for a process like testing there require a proper planning. Most of the Agile methodologies do not say anything about the testing. Most of them include unit testing, automation testing and acceptance testing at the end of the process. But still only a few methods give any information more than unit and integration testing. So, there is need to introduce some practices for software testing of the product in Agile development.

IV. WHY DO WE NEED SEPARATE TESTING TEAM?

In software development there are different roles but in this research, we focused on the role of testers. The testers play a key role in development; developers may lack some skills in development which testers provide. But in Agile development the common mistake is to see testers as junior developers. A good tester has many distinguishing features that make difference with developer. By understanding only this difference, project managers can make a good and successful team. Which projects may not need independent test staff? The answer depends on the size and context of the project, the risks, the development methodology, the skill and experience of the developers, and other factors. For instance, if the project is a short-term, small, low risk project, with highly experienced programmers utilizing thorough unit testing or test-first development, then test engineers may not be required for the project to succeed. It would be frustrated for testers which realize that testing is an easy task. Testing requires patience and flexibility. To understand the testing, a person should know the details and understand the big picture of project. The skills and attitude of tester and developer is always different, although a developer can test well or a tester can write a decent code; but the skills that the job requires are always different and opposite. The famous quote about testing is: "You make it, we break it".

V. COMMUNICATION MODEL FOR EFFECTIVE WORKING OF INDEPENDENT SOFTWARE TESTING TEAM IN AGILE DEVELOPMENT

To introduce a separate testing team, the important thing is to identify, that how the project management can

handle the communication and interactions of software tester with developers and customers. Communication and integration are the soul of Agile development, so we suggested that the testing teams should work very closely with other teams. Assigning the roles and responsibilities of the software testers and developers with respect to Agile development is also of important concern.

We have proposed a communication model for the effective working of an independent testing team. This model will help the practitioners to know that how they can manage the communication and interaction of software testers with the developers and customers in an organization. It will be beneficial for the organizations to follow the same process for managing their testing teams in a better way.Fig.3 show a communication model for testing team.
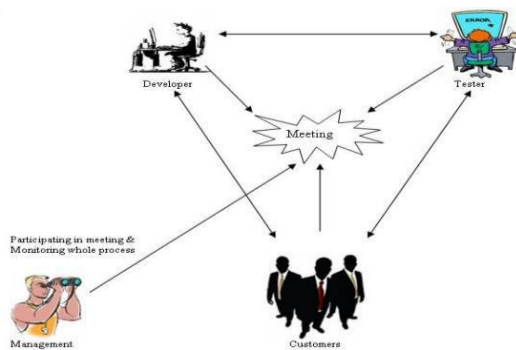

Fig3.Communication Model

VI. CONCLUSION

After this study, we think that Agile development processes should be accompanied with a separate but integrated testing team. The roles and responsibilities of software testers and developers should be assigned. In the industry every company has its own independent software testing team, sometimes if it's company's own product then they also call external testers to test the application. Those external testers can be some expert users, or a sample test group. The testers should work very closely with software developers and another project team. Testers should be part of every project meeting, so that they can know that what the intentions of the project team are. The idea of not having a testing team can be useful in a small organization with a small project size, but in bigger organizations, it is hard to achieve quality product without a separate testing team. We concluded that there should be a combination of both manual and automated testing. Automated testing is effective but still there is need for manual testing. Automated testing can be done by the developer during the sprint and before tester can test the application before the final release. Pair testing is a useful activity and need to be practiced in Agile projects. There

should be proper time slot for test planning in Agile processes. Management plays an important role in Agile methodologies. We feel that as for as the project team is concerned, the team management can conduct some training sessions to educate them and train them. This can help them while working within integrated teams. There is also need to educate the customer to tell him/her that why you are using Agile development, because sometimes the customer hesitates to work with an Agile team, because he/she is busy and cannot be continuously on site or in contact.

REFERENCES

[1] Object Mentor, "Why Use Agile",URL:http://www.objectmentor.com/omSolutions/ Agile_why.html

[2] Rick Hower, "Software QA and Testing", URL: http://www.softwareqatest.com

[3] Brian Marick, "When Should a Test Be Automated?" Testing Foundations,URL: www.testing.com

[4] Ken Arneson, "Automates testing in Agile environment", Checkpoint Technologies, Inc. URL: www.tampabayqa.com/ Images/Agile.ppt

[5] Serena, "An introduction to Agile software development", Serena Software Inc., June 2007, URL: www.serena.com/docs/repository/solutions/intro-to-Agile-dev