Agile Lean Development for Waste Management: A Review

Pooja B. Deokate[1], Dr.Vikrant Shaga[2]

[1]Research Scholar, MCA Department, G. S. Mandal's , MIT(E) College, Aurangabad, India
[2]Assistant Professor, MCA Department G. S. Mandal's , MIT(E) College, Aurangabad, India

ABSTRACT

Nowadays, lots of projects get failed due to the low project quality & maturity level. The top-inclined methodology of software development in present scenario is agile methodology. In Agile project management, there are various methods which can be adapted to develop a successful product. One of them is Lean development Methodology. The lean software development and agile methodology have a characteristics i.e. waste elimination and market responsiveness respectively. Waste elimination tends to reduce or remove the non-value added i.e. NVA activities from the process. This paper focuses on Agile Lean software development methodology and to aware about the wastes occurring while developing a product and how it can be overcome or ignored.

Keywords: Lean Development; Reasons for waste; Solution to waste

I. AGILE METHODOLOGIES

In present scenario of software development industries, lots of projects get fails due to the low project maturity level. This situation shows that there is a need to adapt the well defined project management methodology to develop the software. Agile is an iterative and incremental process which allows revisiting to the software development phases and also improves the efficiency. For example: Requirement gathering phase can be revisited again and again unless and until complexities and risk factors can be reduced. The major advantage of using iterative approach is that it actively reduces the risk[1][2]. Top-trending project management methodology of software development process is Agile Project Management methodology. The success factor of this Agile PM methodologies are related to team, training, customers, team, company culture, planning and scheduling. Even Agile Estimation plays an important role for estimating projects and evaluating size of projects using story points is one of the good methods for agile estimation [1] [2]. The old traditional techniques like waterfall, unified process, prototyping model and spiral model are not able to develop software within budget and in given deadline. In general Agile methods are lightweight processes which utilize the short iterative development cycles.(cho,2008) highlighted the differences of connecting agile and traditional software development method and he explained the Scrum framework which is one of the Agile method. The difference between Agile development and traditional software development methods is not code oriented but it is method oriented [3].

The currently using Agile methodologies are: Extreme Programming which is a lightweight methodology focuses on reducing cost, unit testing during code activities, pair programming and frequent releases of working software(Beck,2000). Scrum methodology uses iterative and increment approaches to develop software products using cross function teams (Deemer et al.,2010). In Lean software development, the software should delivers the features needed by customer and removes the waste that do not add value to the customer. Feature-Driven Development (FDD) is composed five sequential processes develop an overall model, build a features list, plan by features, design by feature & build by feature. Dynamic systems development method (DSDM) is an open framework for rapid application development (RAD) & it assumes that project flow should be based on number of resources and time. Adaptive software development (ASD) provides guidance to avoid the failure in software development but doesn't provide excessive guidance. ASD focuses on iterative development increment approach. Crystal methodologies use time-boxed iterations & focuses on communication and collaboration (Abrahamsson et al., 2002). Rational unified process (RUP) is an iterative process for object oriented system development which includes Unified modeling language (UML) diagrams or components [3][4].

This paper is organized as follows. Firstly, we provide an overview of Agile and its methodologies. Secondly, we described the use of lean software development and how waste management is done. Finally, we concluded our work.

II. LEAN SOFTWARE DEVELOPMENT

Lean production have evolved and were successfully implemented from 1950's (Aziz & Hafer,2013). After some years Womack and Jones studied this system & started calling philosophy behind the system i.e. about the lean thinking (Womack & Jones, 2010). Lean thinking is a method which is defined to achieve more with less.(Marhani et al 2013) believe in lean software development is excellent in managing the construction process & to achieve the projects goal by eliminating waste from development(Marhani, Jaapar, Bari & Zawawi, 2013) [5] . Lean software development is collection of operational techniques which mainly focuses on productive use of the resources. In lean development quality, lead time and cost are identified as a market qualifiers and service level is identified as market winner. The lean and agile paradigm distinctly can be different, but can be combined within successfully designed and operated total supply chains through a decoupling point. This term is known as 'leagility'. The lean software development and agile methodology have a characteristics that waste elimination and market responsiveness respectively (Womack and Jones). Waste elimination tends to reduce or remove the non-value added i.e. NVA activities from the supply chain process. The lean software development is suitable when customer demands are stable and variety is low or minimum where agile development is suitable when demands are unpredictable and have high product variety. The waste can be identified in three types: Unobvious waste, less obvious waste and obvious waste. The obvious waste includes inventory, excessive setup times, unneeded processes; unreliable machines and rework etc. The lean is capable to eliminate the obvious waste and it is able to reduce less obvious waste. The lean and agile can be characterized as collection of practices which can be associated with lean which includes equipment management and total preventive maintenance, human resource training and involvement, decision support systems etc [6].

III. REASON FOR THE WASTE

Agile software development practices seek to eliminate the waste which occurs during software development. To do this, first step necessary to see the waste. To aid software development managers in their quest to find the waste, there are seven wastes of software development given below figure.

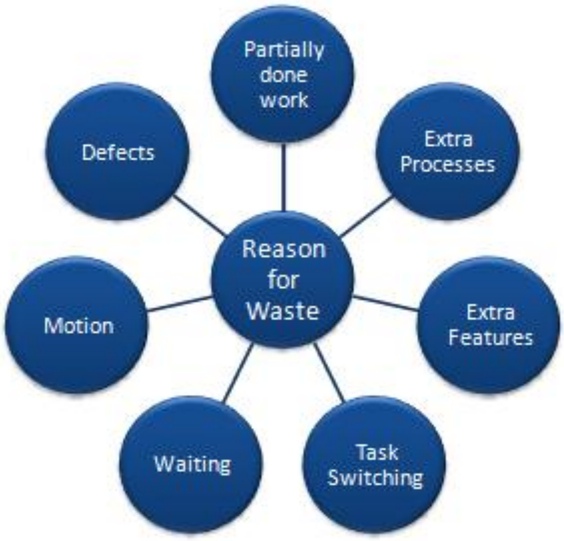

Fig 1: The seven wastes of software Development

A. Partially Done Work

Partially Done software development become an obsolete & it gets in the way of other development they might need to be done. But the problem with the partially done software is that you might have no idea whether or not the partially developed software eventually work. You may have a code which is may be unit tested. But until the software is integrated with the rest of the environment, you don't really know what problem might occur, and until the software is in production you don't really know if it will solve the all business related problem or not. In software development investment are sometimes capitalized, and depreciation starts when software goes into production. what will happen if system never makes it into production?

B. Extra Processes

In software development process paperwork consumes resources. It slows down the response time, It can become obsolete and add no value. So, is it necessary to do paperwork?

C. Extra Features

Developer might like to add a new technical capability into the software just to know how it works. This may harmless, but on the other hand, it is serious waste. As every bit of code in the software system has to be compiled, integrated & tested every time and maintained for rest of the life of the system. Every bit of code increases the complexity & it may become a potential failure point.

D. Task Switching

Assigning the people to multiple project creates the source of waste. Belonging to multiple team, usually causes interruption & more task switching. This task switching time is a waste in development process of software.

E. Waiting

One of the biggest issue in software development process is waiting for things to happen. Delays are common waste in software development. Delay can be done in starting project, staffing ,reviews & approvals and in testing and deployment. Delay keeps the customer from realizing value as fast as possible.

F. Motion

When developer has a question like, how much motion does it take to find out the answer? Can the developer find out the results of test without walking into hall? Development is an activity which requires great concentration & focus, so walking down to hall takes more time than you might think. It will probably take developers several time to reestablish focus to get the question answered.

G. Defects

The amount of waste caused by defect is determined by the defect impact & time it goes undetected. A critical defect which is detected in some minutes is not big source of waste. But, a minor defect which is not discovered for week is a much bigger waste.[7]

IV. SOLUTION TO ELIMINATE WASTES

A. Minimize the partially done work

Partially done software development work can carries huge amount of financial risks. Minimizing this partially done software development work is the risk reduction & as well as it becomes a effective waste reduction strategy.

B. Reduce the extra Processes

Lots of paperwork or documentation during software development may increase the waste in software development process. If you want to produce the paperwork which adds some value to the product, you must follow the three rules : Keep it short, keep it high level and do it off line. Designing solution design document is possibly a good solution to minimize the extra documentation during development as it provides a complete summary of project which is easily understandable to all stakeholders of the project.

C. Avoid to add extra code & features

Developers have the practice of keeping some code in comment for future references but this extra code may affect the entire execution process. If code is not necessary and not needed now, putting it into a system is a waste. Avoid to add extra code and features which are not required for long time .

D. Avoid frequent task switching

The fastest way to develop and complete two project that uses same resources is to do them one at a time and second after finishing first one.

E. Delay decision as late as possible

A fundamental lean principle is to delay decisions until the last possible moment by which you can make most informed decision possible. It is the best way to deal with the uncertainty.

F. Team should work in a single workroom

Agile software development generally recommends that a team work in a single workroom where every has access to developers, testers and customer which gives motion to the development.

G. Detect the defect as early as they occur

The way to reduce these waste is to find and detect defect as soon as they occur and test immediately, integrate and release to production as soon as possible.[7]

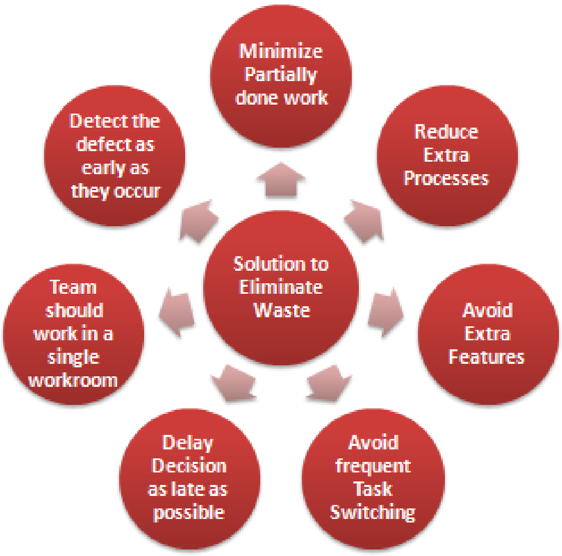
Minimize Partially done work
Reduce Extra Processes
Avoid Extra Features
Avoid frequent Task Switching
Delay Decision as late as possible
Team should work in a single workroom
Detect the defect as early as they occur
Solution to Eliminate Waste

Fig 2: Solution to eliminate the wastes

CONCLUSION

Lean software development process is widely used across the world. As it removes the waste in the software development, it is accepted by the world for rapid development. If you focus on the above solutions given to avoid waste in the system can be removed easily and your software development process takes a motion and can become productive. In this paper we have suggested lean software development as a best approach to reduce the waste in the product and to detect the defect as early as possible. As soon waste is detected, the priority should be given to eliminate the waste and its causes. Causes can be partially done work, extra process, extra code and features, and many others etc. This method will be useful for new start-up organizations or new entrepreneurs to deal with new projects to avoid unnecessary delay in early processes.

REFERENCES

[1] Arturs Rasnacisa & Solvita Berzisaa, Method for Adaptation and Implementation of Agile Project Management Methodology, Procedia Computer Science 104(2017), pp.43-50

[2] Vikrant V.Shaga and Laxmikant K. Shinde , Agile project Estimation: An Imperative Approach for Small Software Industries,International Journal of Computer ScienceTrends and Technology, Vol-5, Issue-1 (2017) , pp.54-57

[3] Shariq Aziz Butt ,Study of agile methodology with the cloud, Pacific Science Review B: Humanities and Social Sciences 2 (2016), pp.22-28

[4] Amadeu Silveira Campanelli , Fernando Silva Parreiras, Agile methods tailoring – A systematic literature review,The Journal of Systems and Software 110 (2015), pp.85-100

[5] Afshin Jalali Sohia, Marcel Hertogha, Marian Bosch-Rekveldta, Rianne Blomb, Does lean & agile project management help coping with project complexity?, Procedia - Social and Behavioral Sciences 226 (2016), pp.252 – 259

[6] Hassan Soltana and Sherif Mostafa ,Lean and agile performance framework for manufacturing enterprises, Procedia Manufacturing 2 (2015) pp.476 – 484

[7] Mary Poppendieck, Tom Poppendieck ,Addison wesley – Lean Software development : An Agile Toolkit .