EFFECTIVENESS OF AGILE DEVELOPMENT FRAMEWORKS WITH RESPECT TO TESTING

M. AHMAD NAWAZ UL GHANI, M. SHOAIB FAROOQ, AMJAD HUSSAIN ZAHID
Department of Computer Science, University of Management And Technology, Lahore, Pakistan
Email: 115007114021@umt.edu.pk
shoaib.farooq@umt.edu.pk; amjad.zahid@umt.edu.pk

ABSTRACT. Software Development Life Cycle models are used as the basis to design software applications. In these models, there exists some tribulations and to overcome these defects, agile models are presented. The mature software applications of agile development utilize both iterative and incremental style. It can be viewed as a response against conventional procedural activity. It is viewed not as to meet the volatility and varying situations but as the modern business ethics for software development. By examining and exploring the agile models, it has been found that efficiency of agile models can be improved. This research investigates the agile frameworks (DSDM and LSD) in terms of testing. It also examines that the effectiveness of dynamic systems development model (DSDM) from testing perspective. The results show the performance of DSDM agile framework and the estimated efficiency of this framework.

Keywords: SDLC; Software Models; Agile; DSDM; LEAN; LSD

1. Introduction: Software Engineering is the process to analyze clients’ requirements, designing the required structures and testing the end user applications. It also includes creating standards for software development. It provides the basic framework for handling larger and complex systems that are both reliable and productive. Software Engineering is being utilized in every field of life, from nanotechnology to large spaceships, it plays a pivotal role. It deals with complex mathematics and large data structures. A software engineer considers needs of end user then plans and creates new applications.

There are bunch of difficulties which still need to be addressed in the field of Software Engineering. It's difficult to accept the robotizing of everything which will prompt a superior world. As software applications are now supercharging our efficiency, applications are less demanding. Software Engineering is the orderly way to deal with the advancement, operation, support, and accomplishment of software. Software Engineering utilizes an all-around characterized and systematic way to deal with creative software development. This approach is thought to be the best method for delivering great applications. In spite of this elderly approach in software development, there are still some genuine difficulties being faced in Software Engineering [1].

In order to overcome problems and challenges at the beginning, the software development life cycle (SDLC) was introduced. Software Development Life Cycle (SDLC) is a description of the structure for assembling software applications from the beginning to the final stage of the program. In some strategic SDLC, a cost-effective, powerful and high assessment test is carried out [2]. The SDLC technique overall runs with the following stages: Requirements, Analysis, Design, Implementation, Testing, Deployment and Maintenance [3]. Software development lifecycle models have been proposed like Waterfall, Iterative, Agile, etc. The software development life cycle provides the most regular element in the entire structure of the project, regardless of any stages. It's a kind of project plan, which uses various SDLC methods. The basic SDLC methodologies are waterfall, V-shaped, iterative, spiral, bigbang, agile and so many others [4].

In this paper focus on agile development frameworks. In agile development, the products associations which are not being utilizing agile before faces few difficulties attempt to actualizing agile development process. Agile is based on time boxed development, where improvement life-cycles are called Sprints, which are short and bound in expansion. Agile sprints typically take two weeks to a month. Agile development of joining of new elements constantly checked time by time, which implies that code is checked regularly and re-gathered at the meantime. So in this way, software is always showing signs of change. The Agile has been developed deliberately, on center of standards, strategies and offer numerous systems. To
make this examination intense and significant, this research focuses on have two agile framework structures which are lean software development (LSD) and dynamic software development (DSDM), to assess the feasibility of agile advancement systems with difference of testing. Agile strategies are utilized diversely as indicated by the project’s point of view. The primary point of this research is to separate these systems on the base of efficiency, advancement and enhancement, so that a powerful and proficient agile development lifecycle framework selected which work as per the users’ prerequisites.

2. Literature Review: A large number of organizations are realizing that their software development procedures must be in consistency with some type of direction. This is necessary in order to have simplicity in the development process as well as for the creation of specific software in short time, for example, automotive software, robotics, medical devices or financial management frameworks. Abrahamson has done research on the philosophy of the agile models and has compared them from project management point of view. This research has explained various models of agile with software development lifecycle, extending administration systems, and experimental standards [5]. The other notable research on agile strategies productivity was carried out by Shelly. Shelly has talked about the conceivable components and attributes, such as, emphasize on length, documentation, the span of the group and an approach such as iterative/incremental. Besides, much consideration is being given to methods for enhancing the effectiveness of organizations, for instance, by adopting lean standards. This brings up the issue that how one can adopt lean standards for software development inside a controlled domain. By examining and exploring the agile models, it has been found that efficiency of agile models can be improved with respect to testing. This research investigates the agile frameworks in terms of testing [6]. The agile has been developed on core rules, methods and offer many frameworks. To make this research acute and profound, this study discusses two agile frameworks (LSD & DSDM) to estimate the effectiveness of agile development frameworks with respect to testing.

Software has been part of current society for over fifty years. There are a number of software development procedures that are being used today. Few organizations have their own customized techniques for building up their software but the main issue revolves around two types of methodologies: heavyweight and lightweight. Heavyweight techniques are the traditional approaches to create software, focus on advanced result, detailed documentation and extensive outlining of the design. The lightweight procedures, also called agile modeling, have gained significance among the software designing teams in the last few years. Unlike conventional strategies, agile techniques utilize short iterative cycles, and depend on implicit information instead of documentation [7].

According to Boehm, organizations should precisely develop the best mix of agile and plan-driven techniques that fits their circumstances. Most organizations do not disregard the agile procedures; however for organizations saturated with the traditional frameworks development techniques, practically during the agile software development process there will be a few challenges, since the two software development approaches are grounded in contradicting ideas. Previous studies have shown that changes in the software development process and complex changes in miracle systems differ only in the use of new tools and techniques to replace existing tools and techniques to improve. This change can affect different parts of the organization, including its structure, culture and management practices. Therefore, understanding the phenomenon of the evolution is related to the basic initial steps of coping up with these changes [8].

Agile software development Methodologies are completely composed of practices that have been made by professionals and scholars. For software applications development, this research examines the agile structures as far as testing. By analyzing and investigating the agile models, it has been found that proficiency of agile models can be enhanced by testing. In this research, two different approaches have been discussed for agile structures namely lean software development (LSD) and dynamic software development (DSDM), these techniques can be seen as a cause or a response based on the traditional strategy, which emphasizes the on the defend based approach to ensure that the problem is completely specified, so each question is not surprising that there are ideals and policies. Traditionalists are considered to advocate a wide range of arrangements e.g., Systematic program after thorough reuse, to make productive and humble. By complexity, light-footed procedures address the test of a flighty world by depending on individuals and their imagination as opposed on procedures [9].

A. **Lean Software Development**: Lean Software Development promotes an incremental model which makes strong adaptability, faster than any other development process, creating more predictable quality products.
Agile programmers follow the statement which was written in 2001 [10]. "Agile Manifest" that developers need to pay attention to:

- Personal and interactive processes and tools
- Complete documentation software work
- Driving customer collaboration through customer consultations
- React as expected

This statement is so interesting that some companies in the manufacturing sector, before the "Agile Manifesto" was designed followed a similar trend. In a software building environment, this development model is known as lean development. This is a lower cost model than the traditional development models because of faster production and customers' value-added products. Lean focuses on any reduction in waste production, and maximizing the value of the product. To reduce waste, the lean manufacturing model relies on the fast flow process to produce quality products, precisely supply the needs of customers at the time they want. To achieve these effects in the same software environment, the principles of lean should be followed in the software development process. Historically, the software is using a cascading variation model to plot almost entire design of the system. First of all requirements are collected, and then the entire product development lifecycle is changing according customers mind. Software architect carefully plans and designs, and then discards system changes or problems, inappropriate to the previous plan. The root cause of these problems is a predictive model, where everything is predicted; assuming product according to the beginning of the program. However, opting for the feedback model these problems can be avoided [11]. The model focuses on the evolution of small increments; each increment quickly adds useful features requested by the end customer feedback. This is at the heart of Lean's additional software development. Lean Software Development Experts, Mary and Tom Poppendieck used it in the Toyota plant, and provided a solid foundation for agile programming. They created a set of lean software development principles. Many steps have been implemented in other programs, such as extreme programming [12].

**B. Dynamic Systems Development Method:** The Dynamic Systems Development Method approach reflects the basics of much project management knowledge. DSDM is rooted in the software developer's community, but the software development, through the development and development of the project development of a comprehensive process, DSDM structure is the basis of this problem has changed [13].

It can be implemented the DSDM framework for the agile and traditional development process. To illustrate how DSDM and agile methods, learn the principles associated DSDM critical value associated with agile development process. The following nine principles are essential for the implementation of any DSDM, ignoring a principle that would call into question the concept of the framework, and considerably increase the risk of the project [14]. In contrast, the DSDM project, some steps can be omitted or modified to suit the specific conditions of the configuration item.

The DSDM project team usually consists of one or two components, the second team can be responsible for the development teams for testing the product. Studies have shown that the team of five experts is ideal in a team, and 6 persons team is highly recommended. If a team can provide more jobs, structure of a multi-team DSDM project size project of 150 people is made. Several projects have found that any single feature set of test and development teams are useful, with the reason and the project is the most important quality of each component. Because the test is a destructive job, it usually produces more efficient task by team [15].

The DSDM development process is divided into seven phases, some of which may be implemented in specific projects that can be omitted. Each stage has several key tasks, that can be changed to include more tasks (if any), and may require other methods for developing DSDM. In particular, DSDM does not provide the necessary technical tasks that allow DSDM in different situations and projects, because this can be achieved through the framework of each organization. These three steps are designed to be iterative, which means that they must be performed in each increment. DSDM assign specific results to each task and each stage. However, the specification of the results is sufficient for the general use of DSDM projects and commercial projects. Here the different phases of dynamic system development methods are discussed [16].

To identify important factors, management must start before the inspection and DSDM process,
the practice is critical to the success of the project. DSDM Alliance has experienced the 10 most important factors for its members [17]:

- Before you start the DSDM concept acceptance
- The decision of the user and developer in the development team.
- Senior management with the aim of ensuring the constructive engagement of end users.
- Incremental delivery.
- Developers have easy access to end users.
- Team stability.
- Development team experience
- The size of the development team.
- Good business relations
- Development of technology

About this list it is interesting to note that many factors are even the responsibility of senior management. This fact shows DSDM requires quite advanced corporate culture as successful as long term. Some factors of success may not even apply to projects, namely "business relationship support" which usually does not exist in open source projects.

3. Research Methodology: It is necessary to follow the appropriate research method to confirm the strength and value of our work. The method of research is a scheme for solving systemic problems as a scientific research. In essence, we are going to be describing our work, explaining and predicting the phenomena according to specific problem statement. Agile Software development techniques have been a topic of great interest among the software development companies. Both LEAN and DSDM are being used at the moment. Both of these have their own pros and cons. But DSDM seems to be ahead of Lean in some aspects. Both of these techniques are very much important in software development, but no comparative analysis has been done in the recent past. This research will help software development companies to get an idea about which technique will be feasible for their company. On the basis of information gathered through sources like research papers, conference papers and magazines, three hypotheses were developed. The hypotheses mainly dealt with examining which agile technique is better than the other.

The research strategy used in this research is Quantitative. Quantitative study is objective, means that reality does not change in any situation. The target audience consists of Software development companies. The participants of the research were software developers who have a hand on experience on both LEAN and DSDM Strategies. The research instrument used in this study is Questionnaire. The data gathered has been analyzed using SPSS. T tests were mainly applied to test the hypothesis.

In order to analyze or determine specifications of required scientific problematic statement, this research needs to gather information from different sources. For this research the information gained is to make accurate, complete, well descriptive and functional.

The information collected is based on the questionnaire filled by the experts and professionals. In this research, all the data collection is based on questionnaire which is authenticated and frequent with the problematic specifications. The opinions of several people are collected via questionnaire to make this research consistent. The responses were satisfying because questionnaire about the research was quite understandable. The questions were according to problem statement are relevant to the format and respondents. To carry out this research, three hypotheses were developed. To investigate and evaluate the hypothesis, the questionnaire link was shared on social media groups and also forwarded to numerous software houses. The hypotheses were tested through SPSS on the basis of which the hypothesis were approved or disapproved. One sample T-test was used to conduct analysis.

The confidence interval was taken to be 95%. The value of alpha was taken to be 0.05. The hypothesis was tested using one Sample T- Test and the results were analyzed on the basis of significance value.

A. Hypothesis# 1: “DSDM is more efficient and effective agile development framework than LSD”.

Following are the results of one sample T Test conducted in SPSS:

| TABLE I. ONE SAMPLE STATISTICS | 58 |
It can be seen from the above Table I., that the mean value of responses (2.8762) is less than the test value (3) which means that the responses are more inclined towards the disagree side. But when we see the significance value it comes out to be p=0.243 which is greater than alpha=0.05 and shows that the difference between the mean and test value is not significant which means that the stated hypothesis is supported.

This hypothesis is based on the extensive literature review of existing work on frameworks of Agile Software development. This hypothesis basically focuses on comparing the effectiveness of DSDM over Lean. The hypothesis is tested by questions which address the efficiency, effectiveness, reliability, understanding, scheduling and resource management, quick delivery, shortcomings and usability.

The responses of people on the above mentioned dimensions were in favor of DSDM. This means that people found DSDM to be more reliable, understandable, effective, easy to use and free from errors as compared to LEAN. This is also evident from the above results of T-Test. So based on this information we can say that DSDM is a more effective agile software development framework than LEAN.

B. **Hypothesis# 2**: “Project development by using the testing technique of DSDM have higher success rate as compared to those of LSD”.

Following are the results of one sample T Test conducted in SPSS:

### SPSS results for the hypothesis 1.

It can be seen from the above Table I., that the mean value of responses (2.8762) is less than the test value (3) which means that the responses are more inclined towards the disagree side. But when we see the significance value it comes out to be p=0.243 which is greater than alpha=0.05 and shows that the difference between the mean and test value is not significant which means that the stated hypothesis is supported.

This hypothesis is based on the extensive literature review of existing work on frameworks of Agile Software development. This hypothesis basically focuses on comparing the effectiveness of DSDM over Lean. The hypothesis is tested by questions which address the efficiency, effectiveness, reliability, understanding, scheduling and resource management, quick delivery, shortcomings and usability.

The responses of people on the above mentioned dimensions were in favor of DSDM. This means that people found DSDM to be more reliable, understandable, effective, easy to use and free from errors as compared to LEAN. This is also evident from the above results of T-Test. So based on this information we can say that DSDM is a more effective agile software development framework than LEAN.

### SPSS results for the hypothesis 2.

It can be seen from the above Table I., that the mean value of responses (2.8762) is less than the test value (3) which means that the responses are more inclined towards the disagree side. But when we see the significance value it comes out to be p=0.243 which is greater than alpha=0.05 and shows that the difference between the mean and test value is not significant which means that the stated hypothesis is supported.

This hypothesis is based on the extensive literature review of existing work on frameworks of Agile Software development. This hypothesis basically focuses on comparing the effectiveness of DSDM over Lean. The hypothesis is tested by questions which address the efficiency, effectiveness, reliability, understanding, scheduling and resource management, quick delivery, shortcomings and usability.

The responses of people on the above mentioned dimensions were in favor of DSDM. This means that people found DSDM to be more reliable, understandable, effective, easy to use and free from errors as compared to LEAN. This is also evident from the above results of T-Test. So based on this information we can say that DSDM is a more effective agile software development framework than LEAN.

B. **Hypothesis# 2**: “Project development by using the testing technique of DSDM have higher success rate as compared to those of LSD”.

Following are the results of one sample T Test conducted in SPSS:
It can be seen from the table that the Mean (3.6990) of the sample is greater than the test value (3). Since the significance value p=0.000 is less than alpha=0.05, this means that the difference between the mean value and test value is significant, which means that the stated hypothesis is supported. This implies that a project developed by DSDM is more successful than software developed by Lean.

This hypothesis is solely associated with the project development techniques. Results of these techniques are compared on the bases of the success rate of both DSDM and LSD techniques in this hypothesis. It has been examined and analyzed by the questions which are addressing the output, flexibility, production and complexity.

The respondents had shown their interest in the favor of DSDM because they think that it is more flexible, adjusting and goal oriented than LEAN. One-Sample T-Test has been applied on the above mentioned responses which prove that DSDM comparatively have higher success rate than LEAN.

C. **Hypothesis #3**: “DSDM have a more scrutinized process of software development as compared to LEAN”.

Following are the results of one sample T Test conducted in SPSS:

| N  | Mean | Std. Deviation | Std. Error Mean |
|----|------|----------------|-----------------|
| H3 | 103  | 3.728          | .83071          |
|    |      |                | .08185          |

| T  | Df  | Sig. (2-tailed) | Mean Difference | 95% Confidence Interval of the Difference |
|----|-----|-----------------|-----------------|-----------------------------------------|
|    |     |                 |                 | Lower | Upper |
| H3 | 102 | .000            | .72816          | .5658 | .8905 |

It can be seen from the table that the Mean (3.7282) of the sample is greater than the test value (3). Since the significance value p=0.000 is less than alpha=0.05, this means that the difference between the mean value and test value is significant, which means that the stated hypothesis is supported. This implies that a project developed by DSDM is more successful than software developed by Lean.

This hypothesis is the result of in depth research of work practiced in the field of DSDM and LEAN with respect to scrutinized process flow. DSDM and LEAN both are compared to know that which one of them is more error free. This hypothesis is venerated by the questions which are focusing on the flow of error free process of software development.
The responses of the questions related to hypothesis three (H3) are supporting the DSDM as compared to LEAN. Ttest applied on the responses, whose results have proven that according to respondents, DSDM is more scrutinized process of software development as compared to LEAN.

4. Conclusion: The overall intention of this research is to estimate the effectiveness of the agile development frameworks (LSD & DSDM) thorough research, based on the responses of industry experts. This research also describes some of the issues and their solutions available for agile development frameworks. The goal of this research is to find a software development procedure, which is the most suitable out of the two agile development techniques. The results show the validity and effectiveness of DSDM agile model with respect to testing. This study will be useful in several aspects of software development lifecycle and help improve skills of software engineers for developing software according to the needs of end users.

This research investigates problem statement in different aspects to identify the clear differences between the two agile development techniques. The main task is to estimate the effectiveness of lean and dynamic systems development methodologies. In the concluding remarks, this research identifies that the dynamic systems development is effective than lean software development in all the perspectives. DSDM does not compromise on quality and delivers on time and eliminates waste, caters any late changes in development process and giving output compatible to the needs of the user. DSDM collaborates the whole procedural activity and provide information continuously and clearly during the development procedure as compared to lean. DSDM is an incremental process which means that all activities are sequential. DSDM and LSD both employ different testing techniques such as timebox testing, MoSCoW prioritization technique and flow of error free process of software development. DSDM show their effectiveness with respect to testing in the whole study.

The crux of this study is that institutes need to increase the quality of the quantitative and agile software development frameworks. Agile software development methodology, such as DSDM, is extensively being used in the industry and deserves further attention. We see that in this area there is a research backlog. It is necessary to create an agile software development framework particularly for dynamic systems development (DSDM) for future research.

REFERENCES

[1] Fitzgerald, B., & Stol, K. J. (2014, June). Continuous software engineering and beyond: trends and challenges. In Proceedings of the 1st International Workshop on Rapal Continuous Software Engineering (pp. 1-9). ACM.

[2] GUPTA, R. S., & LAXMI, V. (2017). Software Development Life Cycle (SDLC) Implementation in Information Technology & Management. International Journal of Recent Advances in Information Technology & Management (Online ISSN: 2581-3609), 1(1).

[3] Kumar, N., Zadgaonkar, A. S., & Shukla, A. (2013). Evolving a new software development life cycle model SDLC-2013 with client satisfaction. International Journal of Soft Computing and Engineering (IJSCCE), 3(1), 2231-2307.

[4] Johnston, Simon K., and Martin P. Nally. "Representing models in systems development lifecycle (SDLC) tools using a network of internet resources." U.S. Patent No. 9,122,422. 1 Sep. 2015. https://patents.google.com/patent/US20090187573, accessed 04 January 2017

[5] Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. (2017). Agile software development methods: Review and analysis. arXiv preprint arXiv:1709.08439.

[6] Shelly. “Comparative Analysis of Different Agile Methodologies.” International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online) Vol. 3, Issue 1, pp: (199-203), accessed 09 February 2017

[7] Stoica, M., Mircia, M., & Ghilic-Micu, B. (2013). Software Development: Agile vs. Traditional. Informatica Economica, 17(4).

[8] Dingsøyr, T., & Moe, N. B. (2013). Research challenges in large-scale agile software development. ACM SIGSOFT Software Engineering Notes, 38(5), 38-39.

[9] Kruchten, P. (2013). Contextualizing agile software development. Journal of Software: Evolution and Process, 25(4), 351-361.
[10] Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Kern, J. (2001). The agile manifesto.

[11] Bolman, L. G., & Deal, T. E. (2017). Reframing organizations: Artistry, choice, and leadership. John Wiley & Sons.

[12] Poppendieck, M. (2011). Principles of lean thinking. IT Management Select, 18, 1-7.

[13] Sani, A., Firdaus, A., Jeong, S. R., & Ghani, I. (2013). A review on software development security engineering using dynamic system method (DSDM). International Journal of Computer Applications, 69(25).

[14] Mead, N. R., Viswanathan, V., & Padmanabhan, D. (2008, July). Incorporating security requirements engineering into the dynamic systems development method. In Computer Software and Applications, 2008. COMPSAC'08. 32nd Annual IEEE International (pp. 949-954). IEEE.

[15] Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Kern, J. (2001). Manifesto for agile software development.

[16] Abrahamsson, P., Warsta, J., Siponen, M. T., & Ronkainen, J. (2003, May). New directions on agile methods: a comparative analysis. In Software Engineering, 2003. Proceedings. 25th International Conference on (pp. 244-254). IEEE.

[17] Coyle, S., & Conboy, K. (2009, May). A study of risk management in DSDM. In International Conference on Agile Processes and Extreme Programming in Software Engineering (pp. 142-148). Springer, Berlin, Heidelberg.