

Impact of Agile Methodology on Software Development Life Cycle

Fizzah Sohail¹, Syed Saood Zia¹, Rehan Qureshi¹, Muhammad Naseem¹ and Hira Haider²

¹Software Engineering Department, Sir Syed University of Engineering & Technology, Karachi, 75300, Pakistan

²Computer Engineering Department, Sir Syed University of Engineering & Technology, Karachi, 75300, Pakistan

Corresponding author: Second Author (e-mail: szia@ssuet.edu.pk)

Abstract- Software engineering strategies had been hired for decades to create software products. The most challenging task for the development of software is to select appropriate software development methodologies. Software developers use agile techniques to reduce the problems faced by the traditional waterfall process. These days, the motivation behind most programming associations is to give programming in a fast time, inside financial plan and changing situations that propelled the conveyance of Agile. The strategy that utilizes iterative expansion and prototyping is generally utilized in an assortment of industry ventures as a lightweight advancement method that could fulfil the alterations of necessities. Short cycles are utilized which can be required for proficient item delivery. In the paper, we become mindful of the impacts that the coordinated strategy has programming improvement strategies with prominence to decent inside the authoritative, efficient, and social system. Coordinated methods are not in every case great, they have a few restrictions too. In this paper, we likewise clarify the estimations of Agile, its points of interest, and its weaknesses.

Index Terms-- Agile Methodology, Crystal Method, DSDM, FDD, Lean, Scrum, SDLC, XP.

I. INTRODUCTION

In 2001, the concept of agile methodology explained by the Agile Manifesto, there were twelve principles collected and considered for refining the philosophy captive in the principles by the manifesto's members who build an Agile Alliance or Agile Software Development Alliance which was considered as a non-profitable organization[1]. Following are the twelve principles of agile manifesto are [2][3][4].

- The foremost demand in agile methodology is to fulfil the customer requirement by initially and constant conveyance of valued software.
- The iteration, incrimination, and continuous changing activity even though in the late development and the most important process bind the change for the client's reasonable advantage.
- At the ordinary spans group thinks about how to turn out to be progressively gainful and afterward changes as needs be. The review gatherings guarantee the usage of the exercises mastered during the task into the following cycle [5].
- Businesspersons and originators (developer) work collectively throughout the project daily for weeks to months and preferring a short period, working to deliver software frequently, Construct projects for encouraged individuals. If they need any support so providing environment and maintenance and trusting them till they complete the requirement of the job [6-37].
- Person to person conversation is highly motivated for conveying data or information efficiently and effectively between the development team [38].

- Maintainable development is prioritized by the agile method. The promoters, consumers, and developers preserve a constant bound indeterminately.
- The main degree of growth is the software workable state.
- Enhancement of agility depends on the Constant consideration to quality of techniques and better design.
- Easiness in the skill of increasing the quantity of works which is not done considered vital.
- Self-organized groups arise the best architectures, desires or requirements, and strategies or designs [39].
- To adjust and mold the behaviour of the team for the sake of boosting their effectiveness is conducted at regular intervals.
- Self-sorting out groups, the scrum group has self-sufficiency and obligation to meet the objectives of the run [40].

By the fact that testing is the prominent key of the agile development [36]. Associations that are making software plans are gone up against the inconvenient choice of picking the right programming improvement life cycle (SDLC). Also developers tried to describe the approaches to manage the requirements of the customer [35].

II. BACKGROUND

The software business utilized the software development lifecycle (SDLC) to configuration, create, produce high caliber, solid and practical items. To build up an application, the venture group utilized some system which may incorporate antiques and pre-characterizing explicit expectations [6]. By and large, the groups including the necessity investigator and software

engineers, who catch the client prerequisites and build up the application are geologically circulated and do not have a typical business language to connect proficiently and successfully. Additionally, since the groups are topographically dispersed, the accessibility of the product master can't be guaranteed consistently, and for every single group[7][8].

Agile techniques are the collective occurrence of considerable procedures for improvement that depends on the iterative approach and quick development of the product. There are fundamental strategies for advanced correspondence that lies in four different characteristics such as flexibly arranged, unforeseen advancement of transformation, steady, and versatile reactions made for changing or doing any amendment if required [8][9]. The significance of the agile methodology to accept the measures of the product in nature of "light anyway sufficient" all encourage person to person dealing to motivate the communication and sharing of knowledge to accomplish the product, that's why it is also named as lightweight methodology due to its logical undertaking for improving little in the product if necessary [10].

The strategy behind the agile technique is the progression capturing to gather the way out to complete the product and should start with direct and obvious estimations to fulfill the final need and thereafter avail the iterative approach to lengthen the life expectancy of the unforeseen advancement of the product. By following the continuous tools likewise, organizing, planning, coding, and testing in each phase of development will refine the product. Thusly, the prominent part is when the product shaped its actual face on the prescribed requirement given by the customer will turn as the last phase of the product development.

Agile method emphasis working on small box-unit along arranging each small activity in sequential steps with continuous check-up on each box-unit, it covers a short time period almost a month to accomplish the target, is known as Iteration. Each prominence integrates a collective work for the progression includes smart decision, assessment of essentials, planning, coding, testing of each unit, and verified testing. Each box-unit alerts the developing team to adjust the changes steadily if needed from the client-side. Most of the light-footed utilization uses a legitimate step by step eye to eye correspondence among associates. There are brief documents that associates with the previous day work, today's work plan, the obstacles they faced previously, and then the resolutions made to overcome those hindrances with collaborate meet-up between the team and the product master.

On the other hand, the agile methods identify the difficulties faced in the unusual, disordered business and to introduce change in condition in certain circumstances. In a short time frame, brief report, sorting the team, coordination with the client are the tactics to be utilized in accomplishing the product. Agile methodology consists a cluster of techniques that is applied respectively under the consideration of the requirement of the client, it consolidates the methods as stated here; Scrum, Crystal Clear, Extreme Programming (XP), Adaptive Software Development (ASD), Feature Driven Development (FDD), and Dynamic Systems Development Method (DSDM) Crystal, Lean Software Development, Kanban, these are considered as "light-

weighted" techniques. It also addresses the issues that arose in the development stage, and jumbled business with unpredictable outcomes, agile intends to achieve the progression of improvise a version of the product in a short time, less documentation, client composed exertion, and diminished chances to publicize.

III. LITERATURE REVIEW

Various reviews and overviews have been collected that show the Agile procedure's pervasiveness reliant on typical for requirements, little or huge association, and experience of the task group. Coordinated techniques have demonstrated their adequacy and are changing the product business. Quality Assurance is not responsible for to check upon the quality only as the significant reason behind is to assure the customers and to develop a product without any delay [41]. A portion of the audit discoveries are introduced here.

F. Almeida received an efficient writing audit to portray and combine the primary element of difficulties in-movement from waterfall to agile atmospheres. The methodology which is used is reasonable to distinguish, fundamentally assess, and coordinate the discoveries of all applicable and great individual investigations tending to at least one research question. Change (Agile) is a mind-boggling process that includes individuals, associations, and procedures. The presentation of lithe procedures in an association carries with it various advantages getting from the qualities of these practices; however, there is a protection from its reception for an enormous scope by the managers, engineers, and developers. Likewise, it clarified the four dimensions of difficulties inside the setting of this work[2]. The three papers from different databases were gathered, after a survey of the considerable number of papers in the survey. It is noticed that agile philosophy has beneficial outcomes in IT anticipates where necessities are dynamic. Agile standards are confined to enterprises as well as be applied in scholastics and are proposed to be applied in each venture independent of the size of the undertaking where prerequisites are adaptable[9].

The primary objective in the venture the board is to accomplish the undertaking objective inside the given constraints (opportunity, period of time, value, and execution plan). The traditional approach accepting the executives was about the fruition of an undertaking inside time and spending plan. The agile methodology is a superior choice for the undertaking where necessities are dynamic. The agile technique chips away at the iterative and gradual turn of events[9].

The evidence is extracted out from the research of a logical report about agile programming advancement approaches, performing efficient and fast as compare to traditional philosophies likewise in huge, appropriated ventures. Improvements are noticed in terms of quality, attributes, and the customer's perspective on the completed outcome, although agile methodologies think about essential changes even late in the endeavor. Simultaneously, gathering better correspondence and collaboration in the gathering because of following the agile practices, findings in updated relations among associates, and to improved specialist satisfaction measurements [12].

There are currently fewer holes between corporate administration and proprietorship due to having a VP of

Development and Quality Assurance; this, alongside the expansion of an Agile (scrum) procedure, made the procedure considerably more compelling and effective [43]. Reviews, accumulations, and stand-up strategies utilized by the scrum techniques made the ventures progressively effective[10]. The heightening procedure is calmer and simpler to achieve. The scrum procedure improved the SDLC (Software Development Life Cycle) procedure and it is basically "self-recuperating."

IV. AGILE METHODOLOGIES AND ITS TECHNIQUES

Agile software development methodologies are a collection of development techniques or methods that allow software improvement using numerous forms of iterative development techniques (see Fig. 1). The most widely used agile practices are quickly depicted as under [11][12]:

A. Scrum:

Scrum is one of the widely used technique, it is an agile procedure normally utilized for creating, delivering, and maintaining complex products[13]. It allows us to deliver business values in a short period. It is a simple framework that helps group collaboration on complex tasks. Although it is far most often used by software development groups, scrum can essentially be beneficial to any group that is working towards a common task [42]. In precise, scrum is a collection of conferences, roles, and equipment that work collectively to help groups to higher shape and manipulate their workload. For less or group of ten fellows, the scrum technique is designed, who divided the respective work into objectives that may be accomplished within time-boxed repetitions, customarily not more than a month or weeks referred to as sprints. The Scrum Crew focuses on the development in prescribes time-box such as 15-minute everyday conferences referred to as daily scrums. The experts of software quality keen to learn to collaborate the team in an efficient manner to produce workability [44]. The technique poses a Product Owner, Scrum Team, Scrum Master. The *Owner of the product* is the main stakeholder i.e. responsible for understanding business goals and prioritizes work to form a product backlog. The responsibility to manage the workflow of the squad lies in the hand of Scrum Master. The *Scrum Team* is a group of members with different skills[14][15].

B. Crystal Methodologies:

Crystal methods are an agile methodology. The developer of this methodology was Alistair Cockburn. People and their interaction instead of tools or process have been focused on this methodology. The Crystal agile framework is constructed on two core beliefs[16]:

- Teams can discover methods on their own to enhance and optimize their workflows.
- Every task is precise and constantly converting, which is why that project group is satisfactory appropriate to determine how it will tackle the work

In Crystal Methodology organizations develop and utilize many techniques such as their enterprise goals call for. Small groups and initiatives can use this method that is not existence-critical.

According to Cockburn, it should view product development as a sport that needs to stimulate all people to interact, turn out to be creative, and bring awesome ideas. He says that instead of focusing on questions like "is our model correct?" we should be seeking out answers to questions like "Is our product meeting the consumer's wishes? Or "Do we have our desires aligned as a team?"

C. Dynamic Software Development Method(DSDM):

DSDM has been developed to deal with common issues faced by projects like late shipping, price overruns, or the very last deliverable not being matched for a motive[17]. For building prior functionality, the iterative and incremental advancement approach is used that is known as DSDM. Its incremental environment supports in attaining customer remarks as the product grows. A single procedure carried out to combine project management and product development related activities in DSDM [18][19]. This Method has eight standards that direct the software team, represent a manner of working, and create a mind-set to handover products on time, within budget. Each standards need to be considered by the development team, as ignoring any of the standards can increase the risk of challenge failure.

D. Feature Driven Development (FDD):

This method of agile is a customer-centric was introduced by Jeff De Luca and Peter Code. This approach is used to maintain complex project management. To acquire the best result, developers map out what features they can be growing, ruin complicated requests into a chain of smaller function units, and after which create a plan for how to complete every aim on time. In FDD modifications are categorized into two levels. At the first level, minor adjustments are dealt with by growing function lists, at the same time important adjustments are handled at the domain modeling phase[20]. FDD iteratively performed activities in five ways. In the initial stage, the complete model is developed by the architect. In the second segment, a feature list is developed. This segment will discover a list of capabilities for the entire set of structures. After acquiring a tough and fast of capabilities, the project manager will then, especially: plan each feature. In the fourth stage, the layout of the features will be developed. After the layout is progressed, the completed feature is delivered to the legitimate build for delivery to the consumer.

E. Kanban:

Kanban visualizes each approach and the actual work passing through that procedure. It is a visible device for handling work as it movements through a procedure. The motive of Kanban is to check the issues in the project in your method and resolve them so work can go with the flow in an efficient manner. [21] It is a manner that assists the team members to work collectively more effectively that helps for the completion of the product with optimal time by using the Kanban board that enhances the flow of work. This technique has a series of columns that constitute the diverse states apiece the object can exist for the duration of the development procedure. As work progresses

through the development lifecycle, the cards circulate from one state to the alternative, until they finish within the final column.[21] It has some principles which make it easy to fulfill that increase customer trust and satisfaction. [22] Those principles are as follows:

- Determine the workflow for the development of the product.
- System naming Virtual Kanban is use to restrict the progression of work to some extend (WIP).
- Manage and organize the flow of work.
- Create the policies in the favor of management explicitly.
- Collaborative improvement implemented along with the usage of scientific methods and models.

F. Lean Software Development:

The iterative technique which optimizing the development time, reduces the cost, reducing waste, and maximize customer values is termed as Lean Software Development [20]. This Method removes and ignores the unwanted requirement with the help of asking the client to select the most effective characteristics and prioritize those characteristics, after which the work of which is done in small batches will be supplied. This technique has seven main principles[23]:

- Removing Leftover
- Strengthening Learning
- Determining as Late as Conceivable
- Conveying as Fast as Conceivable
- Endowing the Squad
- Establishing Integrity In
- observing the Entire

G. Extreme programming (XP):

One of the agile method is a light-weight iterative technique. This approach aims to improve the quality of the products. XP methodology attempts to simplify the entire approach of software development by way of taking the maximum simple and instantly ahead technique, rather than enforcing complicated functionalities. It encourages high consumer involvement, speedy feedback loops, continuous testing and evaluating, continuous making plans, working software programs are supplied by the team at frequent durations, typically each 1 to 3 weeks. [45] XP has five stages that are performed iteratively. The process starts with planning and ends when the final product is developed and handed over to the client. In the planning stage, the project is divided into iterations, user stories are written and release making plan is done[24][25]. The main goal of the designing phase is to create a simple design because simplicity is one of the main features of a quality-based product. After designing, the developer writes the simple code and if there is some inflation, fixes the problem. The fourth stage delivers the product after testing all codes and approved by the testing team from the error clearance. In the last stage, the team gets comments from the client. He is the only individual who estimates the very last and intermediate products. Essentially, XP works best among small experienced developers and teams[26].

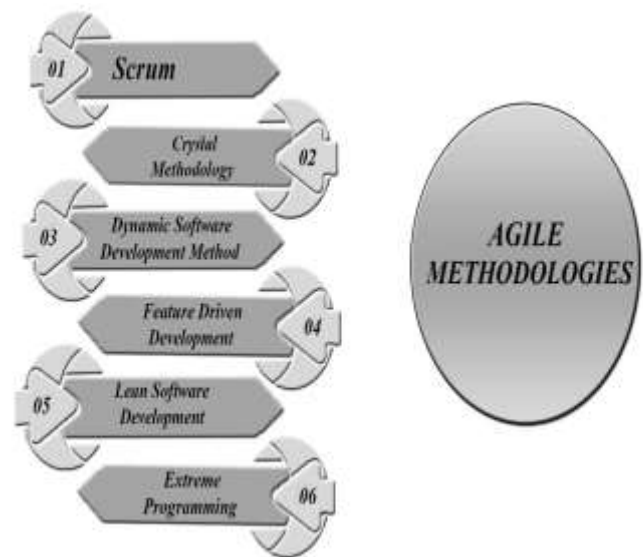


FIGURE 1: Techniques in Agile Process.

V. ADVANTAGES OF AGILE SOFTWARE DEVELOPMENT

As of late, there has been a noteworthy move from rigid development (RD) toward agile (see Fig. 2). Notwithstanding, it has likewise been recognized that agile procedures are hardly ever followed in their unadulterated structure[27]. Hybrid procedures as mixes of RD and agile practices develop. What's more, agile reception has been accounted for to bring about the advantages[28][29].

Manage Change More Effectively: By making littler cycles, the group can concentrate on offering some incentive without expecting to get all the prerequisites in advance. Toward the finish of every cycle, the group will survey the accumulation of highlights and reorganize where they ought to put the time in the following run[30].

Improved Customer Engagement: Agile necessitates that the customers are fundamentally engaged with the improvement procedure. The advancement group look and focus on the clients to coordinate to let them know about the following running and the audit work item during survey meetings. The nonstop association diminishes the disarray among the customer needs and the designers will give to the customers.

Motivation on the Highest Significances First: In the improvement condition of a product, it is required to settle on numerous choices, and it is difficult to keep everything straight. Your excess turns into your definitive, organized plan for the day that could be easily understood and carried out under the shade of transparency to the team and the customer side [46].

Enlarged Productivity: Agile utilizes your assets, permitting them to begin quicker and stay profitable all through. Emphasize the work into pieces; there is consistently an achievement and cut off time. Engineers are constantly centred on refactoring and pushing ahead. They are not motivated in sitting inactive, hanging tight for the sake of work during the disclosure and configuration stage.

Feedback of the client: The work as split into small block to keep the flow of the product development, it is feasible to give input previously, during, and after everyone. Due to this, the joint effort gives visit chances to guarantee that the team is on track to accomplishing the built-up company objectives.

Extremely Concerted Atmosphere: Deft qualities people, cooperation, and client joint effort. All colleagues have purchase wholly. The achievement of the venture depends on recognizing and streamlining every individual's topic mastery.

Transparency: Agile allows imagining and developing personally acquainted with the item from beginning to consummation. Although viewing the application development, also allow to give criticism as it advances.

Accurateness: Following a run, the group will be aware of their speed. This takes into consideration enhanced arranging. About the future runs, a guide will be filled out for what they will have the option to achieve [47].

Vital Qualities First: Agile permits to organize, to concentrate, and to innovate on the things that give the greatest effect on the company to offer some benefit quicker.

Agreement on Clear Explanation: The two words "prepared" and "done" have a great impact on the organizations. Setting the parameters for such definitions in the agile world clearly. "Done" is the word that means completely tried, else prepare to imply the testing by the customer. Vibrant definitions give engineers greater responsibility, and everybody agrees on objectives[31][49].

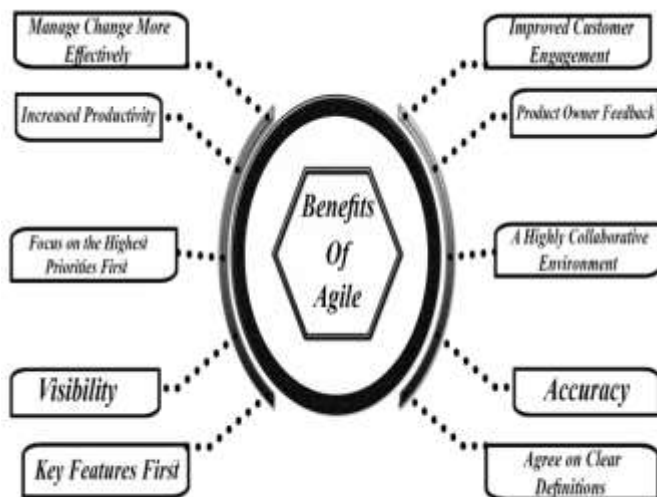


FIGURE 2: Benefits of Agile Methodology.

VI. LIMITATIONS AND RESTRICTIONS OF AGILE METHODOLOGY

It is a fact about the world that things come with shortcomings and limitations (see Fig. 3). On the bases of literature, the

following are the limitations and shortcomings of agile methodologies[32][33].

- Agile methodology prominence the development against design or strategy. It emphasizes on the procedures to attain requirements and emerging code and do not favor the design production [48].
- Due to high testing techniques applied to the product result time consuming and low coverage of testing.
- High management and communication required by project managers.
- It is not favorable for large scaled projects, for example, several repetitions are desirable for completing the preferred operations.
- Small and single features are to be taken under observation and tested which results in much time devoted to small parts of the product.
- For lean projects, the opportunity amount to the agile process or methods is high for the inevitable production on a large scale project.
- The agile methodology demands strong and heavy teammates for a successful application that increases the management overhead and a great responsibility on the project manager's shoulders to remain vigilant in a dynamic perspective[34].

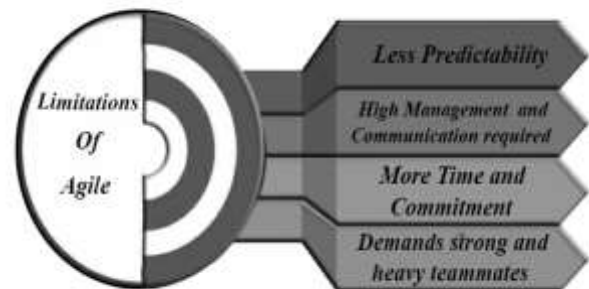


FIGURE 3: Limitations of Agile Methodology.

VII. CONCLUSION

Numerous aspects can stimulate project development in an agile frame. Implementing agile methods for development has an affirmative control on the quality and the production both respectively. Henceforth, the crew of development and the client has become mollified with the enactment in the development processes. Products that involve agile methods accomplished and focus on the customer requirement by releasing small and frequent rapid iteration to avoid rework concept. Though the stability in terms of improvements are still needed in the software development process, amplified adaptive capacity, earlier error uncovering, less testing time, amplified communication. Diverse methods involve diverse variations that are applied to the organization and development principles of software.

REFERENCES

- [1] D. Talby, O. Hazzan, Y. Dubinsky, and A. Keren, "Reflections on reflection in agile software development," in *AGILE 2006 (AGILE'06)*, 2006, pp. 11–pp.
- [2] F. Almeida, "Challenges in migration from waterfall to agile environments," *World J. Comput. Appl. Technol.*, vol. 5, no. 3, pp. 39–49, 2017.
- [3] S. Misra and M. Omorodion, "Survey on agile metrics and their inter-relationship with other traditional development metrics," *ACM SIGSOFT Softw. Eng. Notes*, vol. 36, no. 6, pp. 1–3, 2011.
- [4] S. Balaji and M. S. Murugaiyan, "Waterfall vs. V-Model vs. Agile: A comparative study on SDLC," *Int. J. Inf. Technol. Bus. Manag.*, vol. 2, no. 1, pp. 26–30, 2012.
- [5] M. Fowler, J. Highsmith, and others, "The agile manifesto," *Softw. Dev.*, vol. 9, no. 8, pp. 28–35, 2001.
- [6] M. A. Subih et al., "Comparison of Agile Method and Scrum Method with Software Quality Affecting Factors," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 5, 2019.
- [7] S. Ghaisas, "Recommendation system for agile software development." Google Patents, 2016.
- [8] R. C. Martin, *Agile software development: principles, patterns, and practices*. Prentice Hall, 2002.
- [9] R. S. Malik, S. S. Ahmad, and M. T. H. Hussain, "A Review of Agile Methodology in IT Projects," Available SSRN 3351064, 2019.
- [10] J. J. Scarpino and R. G. Chicone, "The Quality Of Agile-Transforming A Software Development Company's Process: A Follow-Up Case Study," *Issues Inf. Syst.*, vol. 15, no. 2, 2014.
- [11] M. Huo, J. Verner, L. Zhu, and M. A. Babar, "Software quality and agile methods," in *Proceedings of the 28th Annual International Computer Software and Applications Conference*, 2004. *COMPSAC 2004.*, 2004, pp. 520–525.
- [12] P. Abrahamsson, O. Salo, J. Ronkainen, and J. Warsta, "Agile software development methods: Review and analysis," *arXiv Prepr. arXiv1709.08439*, 2017.
- [13] K. Schwaber and J. Sutherland, "The scrum guide-the definitive guide to scrum: The rules of the game," *SCRUM. org*, Jul-2013, 2013.
- [14] M. Kuhrmann et al., "Hybrid software and system development in practice: waterfall, scrum, and beyond," in *Proceedings of the 2017 International Conference on Software and System Process*, 2017, pp. 30–39.
- [15] K. Schwaber and M. Beedle, *Agile software development with Scrum*, vol. 1. Prentice Hall Upper Saddle River, 2002.
- [16] K. N. Rao, G. K. Naidu, and P. Chakka, "A study of the Agile software development methods, applicability and implications in industry," *Int. J. Softw. Eng. its Appl.*, vol. 5, no. 2, pp. 35–45, 2011.
- [17] A. Sani, A. Firdaus, S. R. Jeong, and I. Ghani, "A review on software development security engineering using dynamic system method (DSDM)," *Int. J. Comput. Appl.*, vol. 69, no. 25, 2013.
- [18] F. Anwer, S. Aftab, U. Waheed, and S. S. Muhammad, "Agile software development models tdd, fdd, dsdm, and crystal methods: A survey," *Int. J. Multidiscip. Sci. Eng.*, vol. 8, no. 2, pp. 1–10, 2017.
- [19] B. J. J. Voigt, M. Glinz, and D.-I. C. Seybold, "Dynamic system development method," *Dep. Inf. Technol. Univ. Zurich, Zurich*, 2004.
- [20] U. ISMAIL, S. QADRI, and M. FAHAD, "Requirement Elicitation for Open Source Software By using SCRUM and Feature Driven Development," *Int. J. Nat. Eng. Sci.*, vol. 9, no. 1, 2015.
- [21] H. T. Ingason, E. Gestsson, and H. I. Jonasson, "The project kanban wall: combining kanban and scrum for coordinating software projects," *PM World J.*, vol. 2, no. 8, pp. 1–23, 2013.
- [22] N. Kirovska and S. Koceski, "Usage of Kanban methodology at software development teams," *J. Appl. Econ. Bus.*, vol. 3, no. 3, pp. 25–34, 2015.
- [23] O. Cawley, X. Wang, and I. Richardson, "Lean/agile software development methodologies in regulated environments--state of the art," in *International Conference on Lean Enterprise Software and Systems*, 2010, pp. 31–36.
- [24] G. S. Matharu, A. Mishra, H. Singh, and P. Upadhyay, "Empirical study of agile software development methodologies: A comparative analysis," *ACM SIGSOFT Softw. Eng. Notes*, vol. 40, no. 1, pp. 1–6, 2015.
- [25] J. Erickson, K. Lyytinen, and K. Siau, "Agile modeling, agile software development, and extreme programming: the state of research," *J. Database Manag.*, vol. 16, no. 4, pp. 88–100, 2005.
- [26] K. Conboy and B. Fitzgerald, "Method and developer characteristics for effective agile method tailoring: A study of XP expert opinion," *ACM Trans. Softw. Eng. Methodol.*, vol. 20, no. 1, pp. 1–30, 2010.
- [27] K. Dullemond, B. van Gameren, and R. van Solingen, "How technological support can enable advantages of agile software development in a GSE setting," in *2009 Fourth IEEE International Conference on Global Software Engineering*, 2009, pp. 143–152.
- [28] A. Solinski and K. Petersen, "Prioritizing agile benefits and limitations in relation to practice usage," *Softw. Qual. J.*, vol. 24, no. 2, pp. 447–482, 2016.
- [29] E. Bjamason, K. Wnuk, and B. Regnell, "A case study on benefits and side-effects of agile practices in large-scale requirements engineering," in *Proceedings of the 1st Workshop on Agile Requirements Engineering*, 2011, pp. 1–5.
- [30] M. Paasivaara and C. Lassenius, "Could global software development benefit from agile methods?," in *2006 IEEE International Conference on Global Software Engineering (ICGSE'06)*, 2006, pp. 109–113.
- [31] K. Korhonen, "Evaluating the impact of an agile transformation: a longitudinal case study in a distributed context," *Softw. Qual. J.*, vol. 21, no. 4, pp. 599–624, 2013.
- [32] P. Aggarwal and R. M. Chandani, "Agile Methodology Influence on SDLC (Software Development Life Cycle)," *Stud. Indian Place Names*, vol. 40, no. 50, pp. 4579–4589, 2020.
- [33] D. Turk, R. France, and B. Rumpe, "Limitations of agile software processes," in *Proceedings of the Third International Conference on eXtreme Programming and Agile Processes in Software Engineering*, 2002, pp. 43–46.
- [34] F. Kamei, G. Pinto, B. Cartaxo, and A. Vasconcelos, "On the benefits/limitations of agile software development: an interview study with Brazilian companies," in *Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering*, 2017, pp. 154–159.
- [35] P. Sfetsos, *Agile Software Development Quality Assurance*, United states: Information science reference (an imprint of idea Group Inc.), 2007.
- [36] Lisa Crispin, Janet Gregory, Mike Cohn, Brian Marick, *Agile Testing: A Practical Guide for Testers and Agile Teams*, Addison-Wesley Professional, 2009.
- [37] T. Linz, *Testing in Scrum: A Guide for Software Quality Assurance in the Agile World*, Rocky Nook, 2014.
- [38] A. Bash, *Agile Quality Assurance: Deliver Quality Software-Providing Great Business Value*, BOOKBABY, 2017.
- [39] Sondra Ashmore Ph.D., Kristin Runyan, *Introduction to Agile Methods*, Addison Wesley, 2014.
- [40] L. Adkin, *Coaching Agile Teams: A Companion for ScrumMasters, Agile Coaches, and Project Managers in Transition*, 2010: Addison-Wesley Signature Series.
- [41] J. Sutherland, *Scrum: The Art of Doing Twice the Work in Half the Time*, Random House Business, 2014.
- [42] Ken Schwaber, Mike Beedle, *Agile Software Development with Scrum*, Prentice Hal, 2002.
- [43] E. Ries, *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*, United state: Crown Business, 2011.
- [44] M. Cohn, *Agile Estimating and Planning*, Pearson, 2005.
- [45] James Shore, Chromatic, *The Art of Agile Development*, 2008: O'Reilly Media Inc.
- [46] K. Beck, *Extreme Programming Explained*, Addison-Wesley Professional., 2000.
- [47] Diana Larsen, Esther Derby, and Ken Schwaber, *Agile Retrospectives: Making Good Teams Great*, Pragmatic Bookshelf., 2006.
- [48] Rodríguez, Pilar, Mika Mäntylä, Markku Oivo, Lucy Ellen Lwakatare, Pertti Seppänen, and Pasi Kuvaja. "Advances in using agile and lean processes for software development." In *Advances in Computers*, vol. 113, pp. 135-224. Elsevier, 2019.
- [49] Francis, David L. "6 Step One-Orientating: What is known about achieving requisite agility?" In *Exploiting Agility for Advantage*, pp. 80-126. De Gruyter, 2020.