Agile Software Development Methodologies Evaluation Research Analysis: A review

Sadaf Niaz

noorsadaf378@gmail.com
Department of Computer Science & IT
University of Lahore Gujrat campus, Gujrat ,Pakistan
Babar Hayat
baber.hayat@cs.uol.edu.pk
Department of Computer Science & IT
University of Lahore Gujrat campus, Gujrat ,Pakistan

Abstract
In the quickly developing world, the idea of the conventional advancement needs to advance[1]. When applied to the advanced work environment, the conventional strategies give as much damage as they do great. Tragically, an absence of versatility in the customary techniques has prompted an inflexible work structure that is truly not viable with the present business. The Agile technique is, subsequently, a more adequate practice, in view of creating programming at a faster speed, while as yet looking after proficiency. The coordinated programming improvement strategies are concentrated in this paper. As per study results, Agile software development group needs solid client association; great light-footed task the executives' measures; item proprietor expands business esteem conveyed by group and need and draw in partners; great deft designing procedures or practices; and great advancements and advancement apparatuses[2]. This examination has suggestions for positive social change since associations that comprehend the basic components might have the option to improve project the executives' systems and money saving advantages prompting higher effectiveness, productivity, and efficiency hence profiting the board, representatives, and client. This survey paper incorporates various methodologies of Agile and their analysis.
Keywords:
Agile methods; scrum; Extreme programming (XP); DSDM; Crystal; Feature driven development component

1. Introduction

The point of this acquaintance is with depict the implications that are as of late related to "Agile", to give a meaning of the coordinated turn of events. Deft advancement procedure (otherwise called lightweight improvement philosophy) is a product improvement structure that depends on previous steady and iterative improvement guideline. Lithe generally is an iterative,

lightweight and lean programming plan and improvement strategy. The product arrangement movement gets mind boggling because of the need of direct client cooperation in programming displaying to create an answer in an incorporated way. The utilization of Agile Software Development (ASD) strategy in project the board permits programming delivery to clients rapidly by lessening the time between the plan and sending, advancing both incomplete tests and conveyances with more prominent deftness contrasted with exemplary undertaking the executives' techniques. Among the practices embraced[3] in the ASD we discover (I) face to face correspondence (ii) cycle arranging meeting and reviews encouraged without anyone else coordinating cross-utilitarian groups and (iii) ceaseless reconciliation with testing encourages short emphases and deliveries. In the hierarchical point of view, ASD advances coordinated effort between partners of programming projects, with straightforwardness and greater saw esteem by the client, as well as improving the item's quality and diminishing an opportunity to address potential mistakes. Then again, 84% of associations are as yet in a low development light-footed level and 40% of Agile professionals show this strategy as an approach to oversee conveyed groups, while 22% demonstrate it as a technique to lessen costs in programming improvement projects. To tackle this issue, the association needs to make components that partner customers' cravings to the item or administration being created which is one of the objectives of the Design Thinking (DT) approach. The DT or Human Centered Design (HCD) approach acts from the use of the creator's idea to distinguish issues and building attractive answers for clients, that are mechanically plausible and reasonable for the association's business[4]. The standard of spry programming improvement suggests that "at normal stretches, the group considers how to turn out to be more successful, at that point tunes and changes its conduct likewise". In different terms it could be said that nimble technique tends to precisely the difficulties of a flighty, confused business and innovation climate. Lithe strategies

are utilized to accomplish better programming in a more limited timeframe, self-sorting out groups, client cooperation, less documentation and diminished chance to showcase. Dexterous technique incorporates a group of lightweight strategies that incorporate Scrum, Crystal Clear, Extreme Programming (XP), Adaptive Software Development (ASD), Feature Driven Development (FDD), and Dynamic Systems Development Method (DSDM) Crystal, Lean Software Development and so on[5].

Disintegrating the necessities into tiny errands additionally gives the advantage of having the option to handily decide the level of the product that has been finished at any moment on schedule. This permits the venture directors to decide whether the task is staying up with the normal timetable. On the off chance that the task starts to fall behind, the light-footed cycle normally presents the danger quickly which takes into consideration the supervisors to work with the partners to fine a potential danger alleviation.

2. Literature Review

Agile programming advancement has arisen to give a choice to design driven and frequently heavyweight techniques. Deft techniques share regular qualities and standards. Agile strategies esteem people and associations, working programming, client cooperation, and reacting to change over measures, documentation, agreements, and plans. Agile advancement accentuates short improvement cycles, successive conveyances, consistent vis-à-vis correspondence, and learning. Well known Agile advancement strategies incorporate Scrum and Extreme Programming. Lean Software Development (Lean SD) and Kanban share the comparable qualities and standards with Agile techniques.

The discoveries[6] of the eighth Annual State of Agile Development Survey led by Versionone.com in 2013 infer that 92% of respondents accept that spry methodology helps them in overseeing changing client necessities; 87% of them concur that dexterous methodology improves their group's profitability while 70% accept that light-footed programming approach speeds up the product advancement measure. Further, the study demonstrates a reasonable pattern towards the higher appropriation of Scrum-based philosophies with 73% of respondents rehearsing Scrum and Scrum variations for programming improvement.

Specialists demonstrated that IT projects keep on fizzling at a high rate. One investigation, distributed in 2012 by Dr.Dobbs demonstrated that Agile had a 72% achievement rate, contrasted with a 64% achievement utilizing customary philosophies. While better, an 8% improvement is scarcely a transformation. In the present serious business climate, we need to do improve regarding achievement rate.

Fig. 1. Benefits of Agile Methods

Agile is anything but a silver projectile. Activities actually come up short at roughly a similar rate today as in 2001. It shows up little has modified or advanced in this regard. Also, Kropp's coordinated investigation results and contended that concerning ASD techniques what works for one group won't deal with other.

Kerzner contended that some IT projects fall flat since [23]project administrators are not checking the factors of timetable, cost, and degree. Shenhar and Dvir represented that over 60% of IT projects are not finished on schedule and inside financial plan. The Standish Group (2010) composed that 32% of the IT projects analyzed were effective and 68% of the IT projects were not fruitful; the disappointments were expected fizzled and abandoned activities. IT project achievement relies upon different variables: having an IT project chief with successful initiative and dynamic. Generally, project achievement additionally relies upon accomplishing the venture in a given time, financial plan, and degree. Be that as it may, there is a restricted comprehension of the lived encounters of the individuals who experience the accompanying achievement factors: individuals, measure, hierarchical, specialized and innovations and advancement apparatuses. The motivation behind this subjective, phenomenological research study was to comprehend the lived encounters of IT directors with the achievement components of individuals, measure, hierarchical, specialized, and advances and improvement instruments at worldwide work environments situated in the United States[24]. Modern, genuine networks are indistinguishable from multiple points of view. These people group have inward factors included correspondence, project the board, openness of assets, project arrangement, spending portion, necessity and delivery the executives, and change control measure. Nonetheless, restricted data is accessible in regards to progress factors that affect Informational Technology ASDT projects. The extent of the examination was inside the United States and included virtual group experts. The members were drawn from individuals from administrators from a global undertaking the board affiliation. This examination has suggestions for positive social change since associations that comprehend achievement factors that influence its accomplishment may

create techniques to improve project the executives and money saving advantages prompting higher effectiveness, benefit, and efficiency.

Agile software development (ASD) is significant worldview[7], in field of computer programming which has been generally received by the business, and much examination, distributions have directed on light-footed advancement approaches over the previous decade. The customary method to create programming techniques follow the nonexclusive designing worldview of prerequisites, plan, fabricate, and keep up. These philosophies are likewise called cascade based taking from the old-style programming improvement worldview. They are likewise known by numerous different names like arrangement driven, (Boehm and Turner, 2004); documentation driven, heavyweight procedures, and enormous plan forthright, (Boehm, 2002).

2.1 DIFFERENT METHODS OF AGILE SOFTWARE DEVELOPEMENT METHOD

- Extreme programming (Beck 1999, Beck 2000) [8]
- Adaptive Software Development (Highsmith, 2000)
- Scrum (Schwaber and Beedle, 2001)
- Crystal (Cockburn,2001)
- Dynamic System Development Method (DSDM) (Staplaton, 1997)
- Features Driven Development (FDD) (Palmer and Felsing, 2002)

At the point when the Agile Manifesto was distributed in 2001, the light-footed programming advancement has obtained fame. There exist different programming sellers like Adobe, SAP, Microsoft, and numerous others that have carried out various nimble philosophies over the previous years prior. Accordingly, the agile programming advancement shows up these days as a standard improvement approach. This methodology remembering a persistent consideration for master programming engineers inside limited approvals relating to the improvement approach viability. Abrahamsson et al. Study the dexterous strategies, which incorporate helpful (close correspondence with clients), steady (little programming discharges), direct (the elaborate methodologies are easy to comprehend, change and learn), and versatile (the capacity of creating changes inside the last second) programming improvement technique[9]. Conboy improves another regularly referred to definition as indicated by a broad nimbleness examination through other exploration spaces. In light of the point of view of Conboy, spryness comprises of two ideas, which are leanness and adaptability. Deftness incorporates the capacity of changing, yet rather can inspire the capacity of the undertaking group to quickly adjust to a specific change. Furthermore, leanness is characterized as the association of the clear client esteem dependent on quality, straightforwardness and economy.

Table 1. Conventional versus agile software development.

	Conventional Models	Agile Development
Basic assumptions	Developed Systems are built through extensive and meticulous planning, and are fully predictable and specifiable.	Small teams develop high-quality software using the principles of continuous design improvement and testing, and based on rapid feedback and change.
Management style	Control and command	Collaboration and leadership
Control	Process focal	People focal

3. Methodology

3.1 Agile software development

Agile signifies "moving rapidly "or "adaptable", as the significance it likewise work that way. As such, coordinated advancement is a cutting-edge approach which manages assessment, observing, and self-association, fast conveyance of value programming and ways to deal with client needs with organization objectives. Coordinated programming advancement is depicted as iterative and gradual since all stages are returned to for the duration of the existence cycle. Considering[10] the business see, Agility is the capacity of an undertaking to act ahead of time for the progressions occurring in the climate to expand the advantages[11]. Client criticisms are invited by a light-footed endeavor and attempt to fuse those inputs in the item.
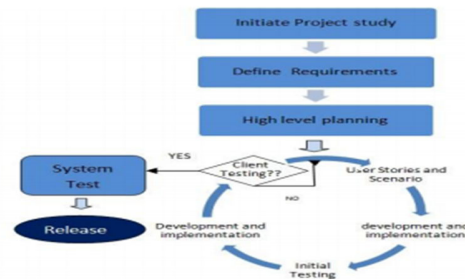

Fig. 2. Working of Agile Methodology

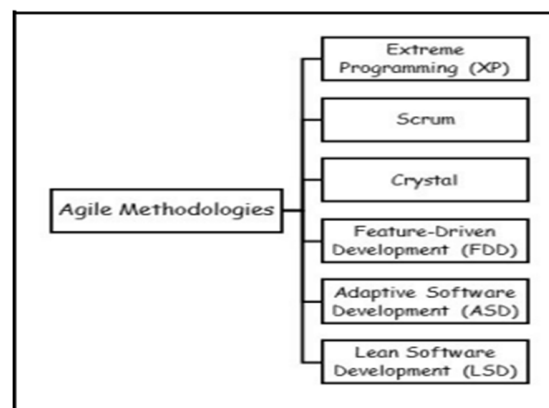Some of the popular agile methodologies are briefly described as under:

Fig. 3. Agile Methodologies

3.2 Extreme Programming

Among the entirety of the light-footed techniques, Extreme Programming (XP) is the most mainstream and all-around reported strategy. XP is essentially founded on these qualities: [12]

- Communication

- Simplicity

- Feedback

- Courage

The XP is as of now executed in various little and medium programming tasks and it has been effectively adjusted commonly. Most XP projects additionally join practices and standards from other dexterous philosophies; however Extreme Programming makes an exceptionally valuable beginning stage for contemplating nimble approaches. XP address the greater part of the coordinated guideline as a job of the improvement interaction. The iterative interaction and quick advancement are the fundamental and the extreme assignments in XP. Besides, the XP requires collaboration between the client, the board and advancement group. The high correspondence among client and engineers can be an exceptionally basic piece of spry procedure. This will decrease the time needed to build up the framework on the grounds that the client is exceptionally near the improvement group. Any additional data that is needed from the client can be gathered effectively and quicker. This will decrease the general season of the improvement cycle. Then again, this will expand the level of client's acknowledgment of the work. The XP developers support a powerful fixation on a product coding measure as opposed to a documentation or plans. Furthermore, the product quality is viewed as the essential fixation where the quality should be enduringly checked dependent on mechanized tests. Moreover, XP software engineers keep up straightforward plan and stay away from attributes that are overmanned. The XP configuration follows a careful rule, which is known as the "Keep It Simple" (KIS) standard. The XP technique [21] energizes refactoring — an advancement practice that depends on rebuilding of the product execution that could improve the product quality, for example its construction or comprehensibility, without having the product usefulness changed. The point is to expand the product long haul viability and extensibility. In a XP interaction, client is consistently accessible to determine and focus on the client necessity. The client is part or some portion of the advancement group. Client imparts the necessity to different individuals from the group.
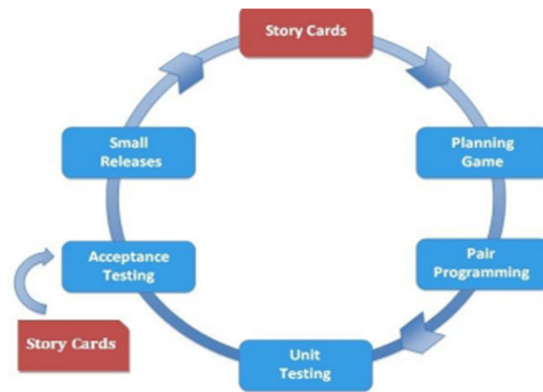

Fig. 4. XP Process

3.3 Scrum

Scrum is a product item advancement and the board system, established by Jeff Sutherland and Ken Schwaber in 1993. Scrum is a lightweight system which empowers capacity of group to address complex versatile difficulties for improvement and conveyance of high worth items by improving coordinated effort, inventiveness and profitability. Its standard dependent on the way that little groups working cross practically creates great outcomes[22]. The above all else advantage of utilizing scrum is that it gives speedy and compelling reaction to showcase needs and requests. As the work is done in nonstop additions and emphasis so it is not difficult to fuse quick changes. It is cost saving and changes are reversible. Quality is primary significant factor in scrum. Scrum makes a point to give as great as could be expected. Client is included at each phase of Scrum so generally the items created utilizing scrum are client fulfilled.
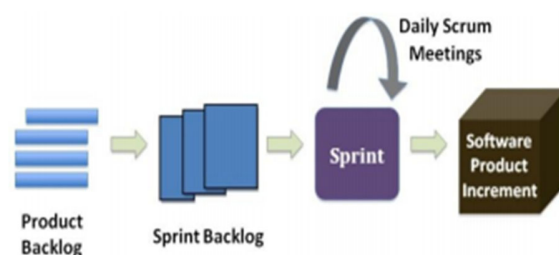

Fig. 5. Scrum Process

3.4 Crystal

The expression "crystal" is taken from the highlights identifying with the geographical gems alongside their own hardness, shape and shading. A Crystal family is really a gathering of deft cycles, which are demonstrated to act successfully through various undertaking types. The point here is to allow the light-footed colleagues to pick the part

that has a place with the gem family and is the most proper for their current circumstance and venture. Strategy plan standards. Capacity to choose the most reasonable technique dependent on project size . It is most versatile ways to deal with programming advancement. This strategy involves group of nimble procedures and having scaling as indicated by group size, framework criticality , various tones signify weight like completely clear ,gem yellow , precious stone orange , gem orange web , gem Red , gem jewel. This family is extraordinarily intended to meet the project's interesting trademark.
[13]All Crystal techniques have 3 Priorities and 3 Properties Priorities: 1. Wellbeing 2. Productivity 3. Livability Properties: 1. Successive Delivery 2. Intelligent Improvement 3. Close Communication

In Crystal Clear the primary jobs requiring separate people are (Cockburn 2002a): support, senior creator software engineer, architect developer and client. These jobs exemplify different sub-jobs. For instance, the planner developer comprises of business class architect, developer, programming documenter and unit analyzer (Cockburn 1998). The sub-jobs that can be appointed to different jobs are organizer, business master and prerequisites finder (Cockburn 2002a). The business master addresses a business see in the undertaking, having information about the particular business setting. He ought to have the option to deal with the field-tested strategy, focusing on what is steady and what is changing (Cockburn 1998).

3.5 Feature Driven Development

The FDD is exceptionally versatile programming advancement measure that is profoundly and short iterative, accentuation quality at all means, conveys continuous, unmistakable working outcomes at all means so it resembles by customer, directors and engineers.
FDD [14]technique is one of the AGILE strategies for programming advancement. FDD is an iterative and steady strategy dependent on isolating the product into various highlights (models), and afterward assembles each model independently. The advancement interaction for each model (highlight) comprises of five exercises: create general model, form include list, plan for highlight, plan for highlight and work by highlight. FDD is utilized for huge ventures, since it very well may be separated into some little size undertakings, which increment the likelihood to finish the task effectively. This likewise gives the administration the achievability to change the group while the undertaking running is without influencing the task time plan and the general quality. Then again, the documentation task is more intricate, and the overhead of the main software engineer is high, since he goes about as a facilitator, coach and lead originator.
The FDD is exceptionally versatile programming advancement measure that is profoundly and short iterative,

accentuation quality at all means, conveys continuous, unmistakable working outcomes at all means so it resembles by customer, directors and engineers.
FDD technique is one of the AGILE strategies for programming advancement. FDD is an iterative and steady strategy dependent on isolating the product into various highlights (models), and afterward assembles each model independently. The advancement interaction for each model (highlight) comprises of five exercises: create general model, form include list, plan for highlight, plan for highlight and work by highlight. FDD is utilized for huge ventures, since it very well may be separated into some little size undertakings, which increment the likelihood to finish the task effectively. This likewise gives the administration the achievability to change the group while the undertaking running is without influencing the task time plan and the general quality. Then again, the documentation task is more intricate, and the overhead of the main software engineer is high, since he goes about as a facilitator, coach and lead originator.

FDD comprises of five successive cycles during which the planning and working of the framework is completed (Figure 8). The iterative piece of the FDD measures (Design and Build) upholds lithe advancement with speedy transformations to late changes in prerequisites and business needs. Commonly, a cycle of a component includes a one-to-three-week time of work for the group[15]
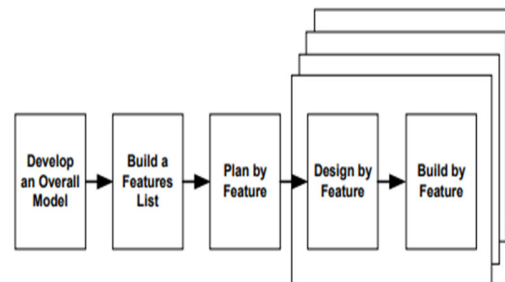

Fig. 6. FDD process design

3.6 Dunamic Systems Development Method (DSDM)

DSDM is a non-benefit and non-restrictive system for RAD improvement, kept up by the DSDM Consortium[16]. This is coordinated improvement structure which is utilized as advancement technique. This is system which exemplifies information about the current task. It depends on nine key rules that principally rotate around business needs, dynamic client association, engaged groups, regular conveyance, coordinated testing and partner cooperation. The basic thought behind DSDM is that as opposed to fixing the measure of usefulness in an item, and afterward changing time and assets to arrive at that usefulness, it is liked to fix time and assets, and afterward change the measure of usefulness in like manner. DSDM comprises of five stages: attainability study, business study, practical

model cycle, plan and assemble emphasis, and execution (Figure7). The initial two stages are successive, and done just a single time. The three last stages, during which the genuine advancement work is done, are iterative and gradual. DSDM approaches cycles as timeboxes. A timebox goes on for a predefined timeframe, and the emphasis needs to end inside the timebox. The time took into consideration every emphasis to take is arranged heretofore, alongside the outcomes the cycle is ensured to create. In DSDM, an ordinary timebox term is from a couple of days to half a month.
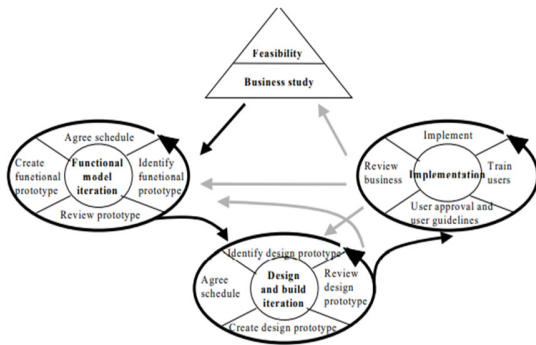

Fig. 7. DSDM Process

The *DSDM* group size fluctuates somewhere in the range of two and six, and there might be numerous groups in a venture. *The* base of two people included comes from the way that each group must have at any rate one client and one designer. [17] *T*he limit of six is a worth found by and by. *DSDM* has been applied in little and enormous ventures the same. *The* precondition for utilizing it in enormous frameworks, as expressed prior, is that the framework can be parted into segments that can be created in little groups. *Dsdm* likewise advances a few jobs that others have not thought of. *Such* client (or client related) jobs are envoy, visionary and consultant. *These* client jobs address distinctive client perspectives

Analysis of different agile methods based on characteristics: Table .1

Parameters	XP(Extreme programing)	Scrum	Crystal	FDD(Feature driven development)	DSDM(Dynamic system development method)
Process ownership	Teamwork	Scrum master	Teamwork	Teamwork	Teamwork
Design	Simple	Complex	Simple	Complex	Simple
Product ownership	Group responsible	Itself responsible for product	Group responsible	Group responsible	Group responsible
Development process	Short releases metaphor, simple testing ,refactoring	Scrum team's product backlog Sprint overview	Integrative method Staging User viewing	Domain object modeling developing by feature individual itself Regular buckles	Active user involvement ,use for frequent product delivery suitable for those projects having tight schedule
Development speed	Very fast	Very fast	Fast	Very fast	Very fast
Special features	Small teams (usually 10-15 members)	15 min daily meeting	It allows team to select a suitable method	Predefine development standers	Contain dynamic development prototyping methods

Additional features	Flexible ,simple designs Use coding standard	Flexible Adoptable	Fast Collaborative	Visibility of results	Flexible Fast collaborative

4. Conclusion

In this review paper we describe our examination of agile methodology   and their various strategies or procedures which execute coordinated. Agile software development gives a powerful answer for the difficulties being as of now looked by the product business including always expanding programming intricacy, dynamic client prerequisites, low spending plans and tight timetables. The abrupt expansion in appropriation of Agile techniques across the product business demonstrates their handiness and viability. [18]Agile cycles guarantee regular collaboration among designers and clients, other than offering esteem expansion and improved profit from venture, alongside speedy responsiveness to changing software requests. With our experimental investigation, we expect to add to the all-around existing assortment of information about the relative examination of dexterous approaches. Understanding the distinctions among different techniques improves the dynamic with respect to determination of the most appropriate procedure in a reasonable manner. Agile developments are acquiring prevalence in industry. In the previous few years research on dexterous programming advancement recommends that deft strategies are powerful and appropriate for some circumstances and conditions. This paper surveys probably the most recent work on spry programming advancement. . Each Agile technique comprises of its own arrangement of determined works on including a wide range of fixations. A portion of the light-footed techniques that are now in presence address nimble standards in various ways for various kind of perspectives. The five lithe strategies include: Feature Driven Development (FDD), [19]Extreme Programming (XP),   Crystal, SCRUM and Dynamic Software Development Method (DSDM). These techniques share a portion of the practices and highlights yet at the same time address an interesting method of building up the framework Agile strategies, like different techniques, enjoy a few benefits and weaknesses. The quick turn of events, consumer loyalty, mistake recuperation in beginning phases, momentary delivery, high correspondence among partners and the effortlessness are the best benefits of lithe strategy. These rules make a progress ahead in the advancement

strategies and cover a few issues that emerge with customary techniques. The most widely recognized impediment of the coordinated technique[20] is that the light-footed strategy turns out to be less compelling when we use it to grow enormous scope projects. Also, this strategy isn't appropriate for the basic life frameworks that cannot acknowledge the straightforwardness as job.

References

[1] Pekka Abrahamsson, Kieran Conboy, and Xiaofeng Wang. 'lots done, more to do': the current state of agile systems development research, 2009.

[2] Pekka Abrahamsson, Outi Salo, Jussi Ronkainen, and Juhani Warsta. Agile software development methods: Review and analysis. *arXiv preprint arXiv:1709.08439*, 2017.

[3] Malek Al-Zewairi, Mariam Biltawi, Wael Etaiwi, Adnan Shaout, et al. Agile software development methodologies: survey of surveys. *Journal of Computer and Communications*, 5(05):74, 2017.

[4] Hiva Alahyari, Richard Berntsson Svensson, and Tony Gorschek. A study of value in agile software development organizations. *Journal of Systems and Software*, 125:271–288, 2017.

[5] Eman A Altameem. Impact of agile methodology on software development. *Computer and Information Science*, 8(2):9, 2015.

[6] Feras A Batarseh and Avelino J Gonzalez. Predicting failures in agile software development through data analytics. *Software Quality Journal*, 26(1):49–66, 2018.

[7] Paul Clarke, Rory V O'Connor, and Murat Yilmaz. In search of the origins and enduring impact of agile software development. In *Proceedings of the 2018 International Conference on Software and System Process*, pages 142– 146, 2018.

[8] Saru Dhir, Deepak Kumar, and VB Singh. Success and failure factors that impact on project implementation using agile software development methodology. In *Software engineering*, pages 647–654. Springer, 2019.

[9] Meghann L Drury-Grogan, Kieran Conboy, and Tom Acton. Examining decision characteristics & challenges for agile software development. *Journal of Systems and Software*, 131:248–265, 2017.

[10] Imran Ghani, Dayang Norhayati Abang Jawawi, Naghmeh Niknejad, Murad Khan, and Seung Ryul Jeong. A survey of agile transition models. In *Emerging Innovations in Agile Software Development*, pages 141–164. IGI Global, 2016.

[11] Saikat Gochhait, Shariq Aziz Butt, Tauseef Jamal, and Arshad Ali. Cloud enhances agile software development. In *Cloud Computing Applications and Techniques for E-Commerce*, pages 28–49. IGI Global, 2020.

[12] Amani Mahdi Mohammed Hamed and Hisham Abushama. Popular agile approaches in software development: Review and analysis. In *2013 International Conference on Computing, Electrical and Electronic Engineering (Icceee)*, pages 160–166. IEEE, 2013.

[13] Rashina Hoda, Norsaremah Salleh, and John Grundy. The rise and evolution of agile software development. *IEEE software*, 35(5):58–63, 2018.

[14] Kiran Jammalamadaka and V Rama Krishna. Agile software development and challenges. *International Journal of Research in Engineering and Technology*, 2(08):125–129, 2013.

[15] Anupama Kaushik. A literature review on agile software development. *IJARCCE*, 5(9), 2016.

[16] Eetu Kupiainen, Mika V Mˆantylˆa, and Juha Itkonen. Using metrics in agile and lean software development–a systematic literature review of industrial studies. *Information and Software Technology*, 62:143–163, 2015.

[17] Antje Lehmann-Benz. Contracting models for agile projects. 2019.

[18] K Madadipouy. An examination and evaluation of agile methodologies for systems development. *Australas. J. Comput. Sci*, 2(1):1–17, 2015.

[19] Dan Schilling Nguyen. Success factors that influence agile software development project success. *American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS)*, 17(1):171–222, 2016.

[20] M Rizwan Jameel Qureshi. Agile software development methodology for medium and large projects. *IET software*, 6(4):358–363, 2012.

[21] Carlos Tam, Eduardo J´oia da Costa Moura, Tiago Oliveira, and Jo˜ao Varaj˜ao. The factors influencing the success of on-going agile software development projects. *International Journal of Project Management*, 38(3):165–176, 2020.

[22] Sonia Thakur and Amandeep Kaur. Role of agile methodology in software development. *International Journal of Computer Science and Mobile Computing*, 2(10):86–90, 2013.

[23] Anna Zaitsev, Uri Gal, and Barney Tan. Coordination artifacts in agile software development. *Information and Organization*, 30(2):100288, 2020.

[24] Halimah Badioze Zaman, Alan F Smeaton, Timothy K Shih, Sergio Velastin, Tada Terutoshi, Nazlena Mohamad Ali, and Mohammad Nazir Ahmad. *Advances in Visual Informatics: 6th International Visual Informatics Conference, IVIC 2019, Bangi, Malaysia, November 19–21, 2019, Proceedings*, volume 11870. Springer Nature, 2019.