Jurnal Teknik Informatika Vol. 16 No. 1, April 2023 (45-57)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v16i1.28995

JURNAL TEKNIK INFORMATIKA

*Homepage* : http://journal.uinjkt.ac.id/index.php/ti

# Systematic Literature Review of Trend and Characteristic Agile Model

**Liana Trihardiningsih[1], Ariel Yonatan Alin[2], Maie Istighosah[3], Muhammad Ryandy Ghonim Asgar[4]**

[1,2,3,4]Magister of Informatics Engineering, University of Amikom Yogyakarta
[1.2,3,4]Jl. Ring Road Utara, Ngringin, Condongcatur, Kec. Depok, Kabupaten Sleman, Daerah Istimewa Yogyakarta 55281, Indonesia.

**Article:**
Accepted: April 12, 2023
Revised: April 02, 2023
Issued: April 30, 2023



***Correspondence Address:**
liana.trihardianingsih@students.amikom.ac.id

**ABSTRACT**

Agile is a methodology and engineering approach for software development that encourages change in collaboration through tasks carried out at various stages of the software development life cycle. Scaled Agile Framework, Kanban, Scrum, Lean, Extreme Programming, Crystal, Dynamic System Development Method, and Feature Driven Development are a few of the approaches that go along with agile. Each of these approaches has distinct traits and qualities of its own. Every engineer and researcher needs to be aware of the benefits and characteristics of each method before deciding to use one. In order to assist engineers and researchers who will use one of these methods, this research will analyze it. The method used in this paper is a systematic literature review, which involved at 52 papers published in the previous eight years, from 2018 to 2022. This method is carried out by determining research questions, determining library initiation and selection, determining inclusion and exclusion criteria, and finally performing data extraction. This essay seeks to establish: (i) Study trends on each agile technique from 2018 to 2022 and (ii) Each agile method's characteristics. The results of this literature review indicate that Scrum and Extreme Programming have overtaken other agile methodologies as the most popular agile techniques over the last eight years. Through an analysis of the characteristics of each methodology, namely the development approach, suggested iteration time period, team communication, project size, project documentation, design, workflow approach, project coordinator, role assignment, coding, testing, and the nature of customer interaction, it is found that Scrum and Extreme Programming do have several advantages over other methodologies.

**Keywords:** *Systematic Literature Review, Software Development, Agile, Trend Study, Characteristic.*

ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v16i1.28995

## 1. INTRODUCTION

Every engineer requires a so-called process model when developing software. The software development process is abstracted in the software process model. The stages and order of each process are established by this process model. Process models are constantly evolving with the times, and the Agile Model is one of the process models that has evolved into a container for many other process models.

Throughout the course of a project's lifespan, repetitive and incremental interactions are made as part of the agile conceptual framework for software engineering, which starts with the initial planning phase and continues until the deployment phase. The agile method's major objective is to reduce additional spending on software development while allowing for change without endangering the procedure or causing overwork [1].

It is not surprising that numerous researchers have raised research on agile approaches given the dependability they offer. According to studies by Aftab et al. [2] between Extreme Programming (XP) and the Feature Driven Development (FDD) approach. This study examines a number of indicators, including productivity, actual work effort, and design classes. According to the study's findings, FDD performs poorly when compared to XP in practically every significant statistic. Due to extensive development, challenging projects, and communication issues, FDD is made worse. However, this study claims that the FDD's heavy, inflexible structure and intricacy are its two most important factors.

Additionally, Puttida et al. [3] employing the scrum approach to research. This study uses the scrum method to evaluate team performance. According to the study's findings, the scrum methodology may be used to implement product management and working methods, enabling teams, products, and working conditions to be continuously improved. Additionally, this study explains how several elements, like bravery, commitment, focus, openness, and respect, have an impact on the effectiveness of the scrum technique.

In 2021, research employing the FDD approach will be conducted once more. This Research [4], In order to help the disaster response team conduct assessments in the impacted areas more efficiently and promptly, we must study and design a disaster emergency response assessment system. The investigation's finding is that larger and more significant software projects can use the FDD method. Consequently, it is anticipated that this research will be able to give advice that is pertinent to and in line with Indonesian laws and regulations.

There are several forms of agile methodologies besides those already mentioned, including Scrum Methodology, Scaled Agile Framework, Lean Software Development, Kanban, Extreme Programming (XP), Crystal Methodology, Dynamic Systems Development (DSDM), and Feature Driven Development (FDD). Each methodology has its own guiding principles, life cycle, constituent parts, benefits, and drawbacks. Therefore, before choosing to employ one of these methods, every engineer and researcher needs to be aware of its qualities and properties.

In order to make it simpler for researchers and engineers to decide which technique is best to employ on the project at hand, this study gives information regarding trend studies in agile methods as well as the characteristics of each agile approach. In order to determine whether the method is updated annually, articles from the last eight years, from 2018 to 2022, were gathered for this study.

## 2. LITERATURE REVIEW

An incremental approach to project software development is prioritized by agile software development in project management surpasses the specified phase [5]. This approach places a strong emphasis on three things: (i) ongoing software testing; (ii) collaboration between developers and consumers; and (iii) the human element of software development, which goes beyond methods and tools.

Software development operations can be made better and faster by integrating quality practices and systems that track benefits. This will raise customer satisfaction levels [6]. Agile Software Development enables firms to respond quickly to ongoing changes in business conditions because of this, with an emphasis on flexibility in the face of uncertainty and ongoing change. The most popular Agile Methodology techniques are listed below:

a. Scaled Agile Framework

Dean Leffigwell created the Scaled Agile Framework to modernize agile for large businesses [7]. The success that its customers

Jurnal Teknik Informatika Vol. 16 No. 1, April 2023 (45-57)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v16i1.28995

have had in achieving enterprise-wide success has lately helped the Scaled Agile Framework gain widespread recognition. By integrating SAFe across the entire organization, other Agile methodologies like Scrum, Extreme Programming, test-driven programming, team focus, and project-level practices are improved. The Scaled Agile Framework (SAFe) methodology was created to aid big businesses in producing high-quality products more quickly and affordably while also increasing user happiness [8]. According to [9] SAFe in terms of human resources and project management practices is quite demanding and costly to implement SAFe, companies need to commit the time and resources they have and need for this method to be adopted properly. the individuals involved also need to receive specific attention and training so that this method can work effectively and deliver the best results.

It blends Scrum, Extreme Programming, Kanban, and Lean techniques at the team level. Value stream, program, portfolio, and team levels are available. Built-in quality, transparency, alignment, and program execution are the basic values of SAFe.

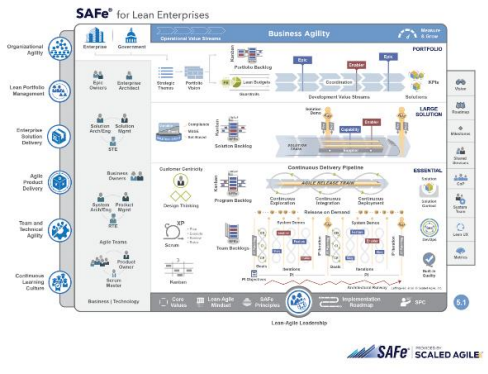


*Figure 1. Workflow of Scale Agile Framework*

b. Kanban

Kanban is a method for displaying a manufacturing system's process. By focusing on the flow of production, Kanban employs queuing theory to regulate and enhance the flow of value. David Anderson, working with the Microsoft software team, introduced Kanban to software engineering for the first time in 2004. Anderson lists these five guiding concepts for Kanban: Create specific process policies, visualize workflows, limit work-in-progress, measure and monitor flow, restrict work-in-progress, use models to find expanding opportunities [10].

Implementing kanban can help businesses satisfy customer demands and coordinate multiple products by reducing overproduction. Kanban is useful for fostering both continuous improvement and overall production support. With Kanban, cycle times are shortened since the focus is on minimizing active development activity and visualizing the flow of development value [11], [12].

In practice, Kanban provides great flexibility and can be adapted very well to projects that have different needs and characteristics, for the success of kanban, a deep understanding of kanban principles and practices is required, as well as the participation of all members[13] .

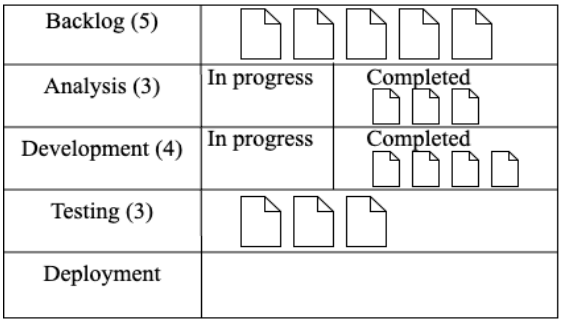


*Figure 2. Workflow of Kanban*

According to research [14], The least prescriptive agile methodology is kanban, which only adds restrictions related to visualizing the workflow and limiting work in progress.

c. Scrum

Scrum is a structured Agile methodology used for solving challenging problems in product development [15]. Scrum is intended to accelerate development, harmonize personal and organizational goals, and improve individual growth and quality. Although it is not always employed, the agile method known as Scrum has grown in popularity recently and is proving effective for software development [16].

In Scrum, there are three roles: Product Owner, Development Team, and Scrum Master [17]. The product owner is in charge of determining the plan as a product backlog. The development team is responsible for implementing the plans from the product backlog. The Scrum Master plays an important role in the entire Scrum process that occurs and ensures that all members of the developer are responsible for the entire process.

Jurnal Teknik Informatika Vol. 16 No. 1, April 2023
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v16i1.28995

Scrum includes benefits like variable sprint requirement selection and no set procedures to follow, and it allows a tailored approach of working on various projects with varying requirements [1]. Despite being widely utilized, scrum still has a few key advantages over other approaches [18].

Scrum has proven to be effective for increasing productivity, transparency and engagement of each team member. Scrum's flexibility allows for good implementation in complex projects and changing environments [19].

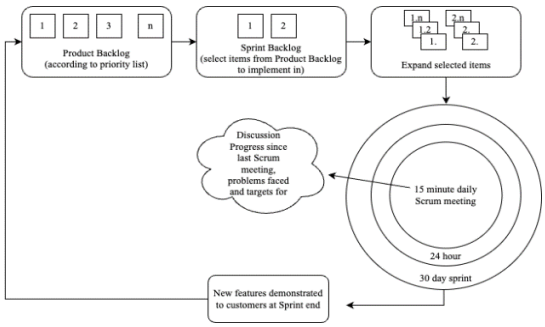


*Figure 3. Workflow of Scrum*

d. Lean

Krafcik (1988) first used the term "lean" to refer to the Toyota Production System's operating procedure. Lean has four main methods: (1) Transparency; (2) Standardization and optimization; (3) Leading by key performance indicators (KPI) (4) Organizational culture [20].

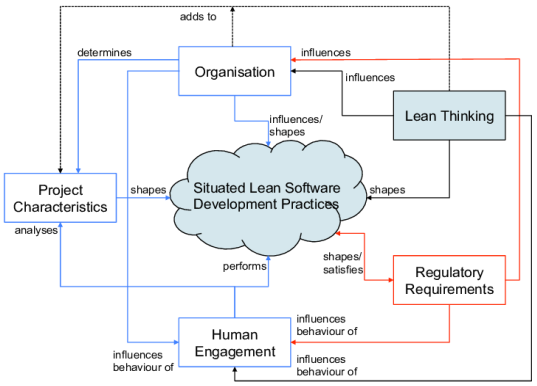


*Figure 4. Conceptual framework for lean software development (simplified).*

In practice, lean has a positive impact on processes. Lean implementation can help increase efficiency, reduce waste, and improve accuracy in process execution. lean focuses on identifying and eliminating waste that occurs in processes thereby increasing productivity and effectiveness. companies/organizations that implement lean practices consistently and continuously have a better chance of achieving higher levels of process quality [21].

**e.** *Extreme Programming*

The most popular and widely adopted Agile methodology is called "Extreme Programming" (XP), which emphasizes the virtues of "simplicity, communication, effective feedback, and courage to face hurdles". This method delivers superior quality products while satisfying customers, made in small teams and team members who value collaboration[22].

With XP, work is intended to begin with a smaller development team and fewer detail artifacts. The development cycle of XP uses an iterative and vertical methodology. The four fundamental XP principles for software development are planning, coding, designing, and testing [12], [23].

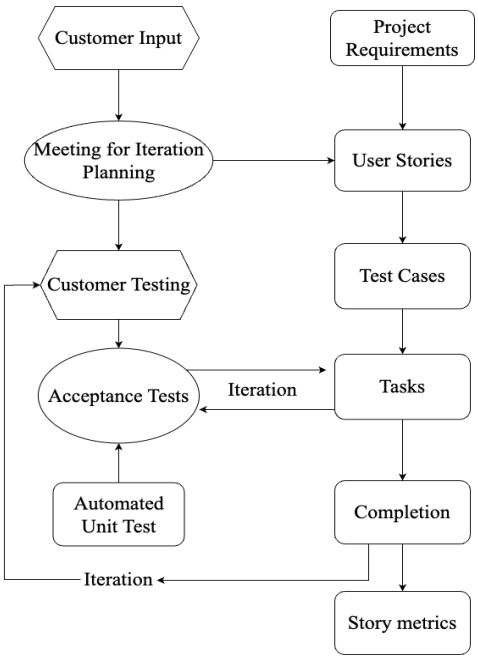


*Figure 5. Workflow of Extreme Programming*

f. Crystal

Cockburn created the system known as Crystal, which places more of an emphasis on people than on processes. The crystal method is essentially a project management philosophy that categorizes projects based on the size of the crew. According to the degree of complexity, the crystal method is

Jurnal Teknik Informatika Vol. 16 No. 1, April 2023 (45-57)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v16i1.28995

split into four labels: white, yellow, orange, and red. The more colors, the deeper and more difficult the problem [12], [24].
The crystal method emphasizes adaptability and flexibility in software development. Crystal can help improve project performance by considering the unique needs of each project. This method emphasizes the importance of communication and collaboration between team members in the project [25].

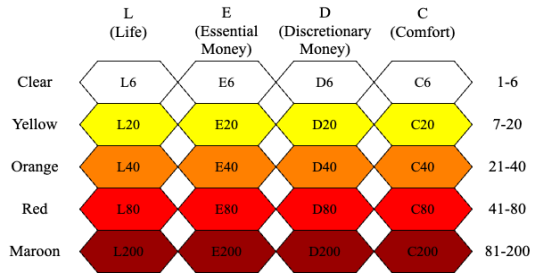


*Figure 5. Workflow of Crystal*

g. Dynamic System Development Method

Stapleton first introduced the Dynamic System Development Method (DSDM) in 1995. It is an iterative and incremental technique that unifies the project and product management life cycles into a single process [26]. Because DSDM focuses on strategic goals while managing time, money, risk, and quality, it has been demonstrated that it may assist deliver outcomes efficiently and rapidly. However, in practice, not all projects—such as those that are real-time or safety-critical—are appropriate for adopting DSDM [27].

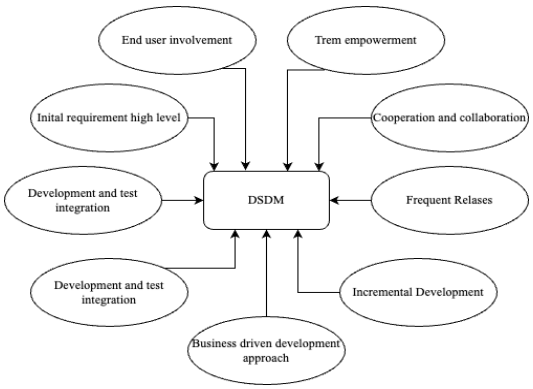


*Figure 6. Workflow of DSDM*

In practice, DSDM may present obstacles and hurdles, such as a lack of understanding and awareness of DSDM, a lack of stakeholder involvement, and difficulties in managing rapid change. For this reason, in practice, DSDM needs strong management support, appropriate training, effective communication between team members and interested parties, and an organizational culture that supports agile practices [28].

h. Feature Driven Development

Peter Coad and colleagues created the concept of feature driven development in 1999, and Palmer and colleagues improved it in 2002 [29]. Large and complicated projects frequently adopt the agile paradigm known as "Feature Driven Development" (FDD) [2]. The five stages of FDD are: developing the overall model; building the feature; planning and designing by feature; and building by feature. FDD, on the other hand, concentrates on the design and development stages before emphasizing quality factors all across the process by meticulously tracking each process's development. Because each FDD process involves a number of activities and responsibilities, it depends on the developer's experience and is strict in its execution, making it only appropriate for usage on medium- to large-scale projects.

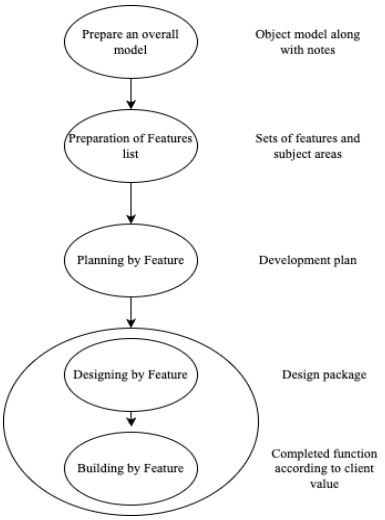


*Figure 7. Workflow FDD*

In practice, FDD takes a structured and systematic approach to identifying, planning and implementing key features. FDD implementation uses object oriented modeling which is key to FDD success. FDD is suitable for projects that have high complexity and require a clearly organized structure. FDD is relatively easy to understand and implement, making it suitable for learning approaches in educational contexts [30].

ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v16i1.28995

## 3. METHODOLOGY

### 3.1 Research Question

A systematic literature review is a task that involves finding, assessing, and interpreting pertinent research utilizing a query connected to intriguing subjects and interesting phenomena to watch [15]. Therefore, it is essential to provide a study topic to guide what will serve as the emphasis for resolving the issues discovered.

The study's research questions are

RQ1. What has been the recent trend in studies utilizing agile techniques in the last 8 years?

RQ2. What are the characteristics of each process model in agile methods??

### 3.2 Library Initiation and Selection

The method of searching for literature reviews that will be used to respond to the research question is ;

a. Choose keywords. Agile, Scale Agile Framework, Kanban, Scrum, Lean, Xtreme Programming Crystal, and Dynamic System Development Method were the keywords employed in this study's literature search.
b. The publish or perish 8 application was utilized as the data source to get the literature. Additionally, it makes use of a number of websites, including Google Scholar, IEEE Xplore, Springer, Elsevier, ACM Digital Library, and related publications.

### 3.3 Inclusion and Exclusion Criteria

Inclusion criteria are criteria used to identify existing subjects in existing research to later be included in the research to be done [20]. Criteria that are used as a cue in one or more scholarly journal articles, conference papers, and journal articles. The exclusive criteria are the study topics that don't relate to the active research topics. Here are some hints and details found in this literary work:

Inclusion Criteria

- The Scale Agile Framework, Kanban, Scrum, Lean, Xtreme Programming Crystal, and Dynamic System Development Method are the only minimum agile methods that are discussed in the literature.
- The literature consulted was written between 2018 and 2022.
- Literature uses English.

Exclusion criteria

- None of the varieties of agile methodologies listed in inclusion receive much attention in the literature.
- The literature that was used wasn't released between 2018 and 2022.
- Literature does not use English

3.4 Data Extraction

Data extraction is done in order to take into account the restrictions on the types of data that will be examined in a systematic literature review. The following data were extracted for this study:

1. Title
2. Year
3. Type of Publisher
4. Process Model used
5. Characteristics

## 4. RESULT AND DISCUSSION

The important difference that exists between the use of the agile method and the traditional method lies in the adjustment of the criteria for the project to be worked on, where software development with the agile method is carried out by considering various things that suit the needs of the user. In the agile method, software development does not immediately produce a design until the end of the system, but gradually, according to the method you want to use. Each method will make a design to find out the errors and deficiencies of the development process. Developers will also adapt to the topics or ideas that users want and in what areas the software will be applied. as in the study conducted by Aftab et al. [2], who compared productivity, actual work effort, and design class of XP with FDD. The study found that FDD performed poorly when compared to XP in almost every significant statistic due to its heavy and inflexible structure and complexity.

Another study conducted by Puttida et al. [3] using the Scrum approach to evaluate team performance found that the Scrum methodology can be used to implement product management and work methods, making it possible for teams, products, and working conditions to continue to be improved. Therefore, this literature will present an overview of which agile methods are popularly used for software development, along with explaining the criteria for each method based on the results of research that has been conducted by previous researchers.

We offer the findings from the analysis and comprehensive literature review in this part. In addition to addressing the research question posed in section III, the objective is to describe the extracted data. Table 1 displays the findings of data extraction performed on 52 papers acquired over the course of 8 years, from 2018 to 2022.

The purpose of the next section is to provide an answer to the research question posed in part III using the data extraction findings shown in table 1.

*Tabel 1 Extraction Data*

| No | Year | Type of Publisher | Method |
|---|---|---|---|
| 1 | 2022 | Conference | Scaled Agile Framework |
| 2 | 2022 | Conference | Kanban |
| 3 | 2022 | Journal | Scrum |
| 4 | 2022 | Journal | Lean |
| 5 | 2022 | Conference | Extreme Programming |
| 6 | 2022 | Conference | Crystal Method |
| 7 | 2022 | Journal | DSDM |
| 8 | 2022 | Journal | FDD |
| 9 | 2021 | Journal | Scrum |
| 10 | 2021 | Journal | Scrum and XP |
| 11 | 2021 | Journal | Scrumban (Scrum and Kanban) |
| 12 | 2021 | Conference | FDD |
| 13 | 2021 | Journal | Extreme Programming |
| 14 | 2021 | Journal | Extreme Programming |
| 15 | 2021 | Workshop | Scrum |
| 16 | 2021 | Journal | Extreme Programming |
| 17 | 2021 | Journal | Scrum and Kanban |
| 18 | 2021 | Journal | Extreme Programming |
| 19 | 2020 | Journal | Scaled Agile Framework (SAFe) |
| 20 | 2020 | Journal | Scrum |
| 21 | 2020 | Journal | Lean |
| 22 | 2020 | Journal | Scrum |
| 23 | 2020 | Journal | Extreme Programming |
| 24 | 2020 | Conference | Scrum |
| 25 | 2020 | Journal | Extreme Programming |
| 26 | 2020 | Journal | Scrum |
| 27 | 2020 | Journal | Dynamic Systems Development |
| 28 | 2020 | Journal | Lean Software Development |
| 29 | 2019 | Journal | Feature Driven Development |
| 30 | 2019 | Journal | Feature Driven Development |
| 31 | 2019 | Journal | Scrum |
| 32 | 2019 | Journal | Scrum |
| 33 | 2019 | Conference | Scrum |
| 34 | 2019 | Journal | Extreme Programming |
| 35 | 2019 | Workshop | Scrum |
| 36 | 2019 | Conference | Scrum |
| 37 | 2019 | Journal | Extreme Programming |
| 38 | 2019 | Journal | Extreme Programming |
| 39 | 2018 | Journal | Kanban |
| 40 | 2018 | Journal | Scrum |

ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v16i1.28995

*Table 1 continued…*

| No | Year | Type of Publisher | Method |
|---|---|---|---|
| 41 | 2018 | Journal | Scrum |
| 42 | 2018 | Conference | Scaled Agile Framework |
| 43 | 2018 | Conference | Scaled Agile Framework |
| 44 | 2018 | Conference | Lean Software Development |
| 45 | 2018 | Journal | Lean Software Development |
| 46 | 2018 | Conference | Kanban |
| 47 | 2018 | Journal | Kanban |
| 48 | 2018 | Journal | Kanban |
| 49 | 2018 | Journal | Kanban |
| 50 | 2018 | Conference | Extreme Programming |
| 51 | 2018 | Conference | Extreme Programming |
| 52 | 2018 | Conference | Extreme Programming, Crystal Clear, DSDM, FDD |

In this section, we present the results of the analysis and systematic literature review. The aim is to describe the extracted data and to answer the research question described in section III.

RQ1: Trend Studi

Agile study methodology publication trends are seen in Figure 8. Over 52 pieces of literature from the years 2018 to 2022 are cited in the bibliography. As seen in Figure 8, the highest percentage for the use of the agile methodology in trend publications is using scrum, at around 31.4%, followed by extreme programming, at approximately 27.5%. In contrast, the two methods that are most frequently used are the Crystal with a percentage of 2.0%, and DSD with a percentage of 3.9%.

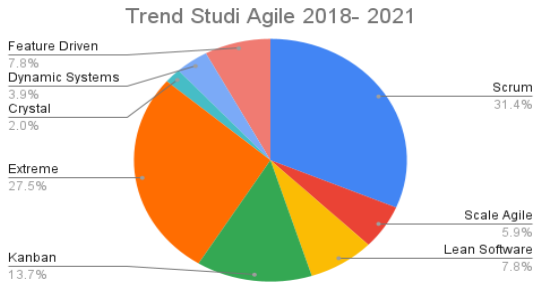


*Figure 8. Trend Studi Agile 2018-2022*

RQ2: Characteristic

This study outlines a number of the traits of each model before tabulating them in accordance with the traits of each process model. Table 2 shows various criteria for each agile method. To make it easier to review the topics to be discussed, we group each character into topic discussions. From the study literature obtained, the criteria will be defined into 13 sub-criteria, which include development approach, recommended iteration time period, team communication, project size, project documentation, design, workflow approach, project coordinator, role assignment, coding, and nature of testing customer interaction: and team collaboration.

Jurnal Teknik Informatika Vol. 16 No. 1, April 2023
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v16i1.28995

*Table 2. Characteristic Agile*

| **Characteristic** | **Scrum** | **Extreme Programming** | **Lean** | **Kanban** | **FDD** | **Scale** | **Crystal** | **DSDM** |
|---|---|---|---|---|---|---|---|---|
| Development approach | Iteractive increments | Iteractive increments | Continual learning & improvement | No obligatory iterations | Itteractive | Build incrementally with fast | Incremental | Itteractive |
| Recommended Iteration time period | Two to four weeks | Time | Does not specify technical practices | Does not specify technical practices | Two days to two weeks | 8 – 12 weeks long | Depending on method family | 80% solution in 20% total time |
| Team Communication | Informal daily stand-up meeting | Informal project | Informal project | Use (theoretical) models to recognize improvement opportunities | Documentation based | Informal project | Informal face to face | Documentation based |
| Project size | All types of project | Smaller project | All types of project | All types of project | More complex project | Bigger project | All types of project | All types of project |
| Project Documentation | Basic | Basic | Basic | Important | Important | Important | Basic | Exist |
| Design | Complex | Simplification of code & accommodation of unexpected changes through refactoring | Simple | Limits the amount of work in progress & ensures waste reduction | Simple | - | Depending on method family | Designed from the ground up by business people |
| Work flow approach | Iterations (sprint) | No Iterations | Continuous improvements | Short Iterations | Iterations are not as well defined | Integrated learning cycles | Short Iterations | Does not specifically exact iteration lengths |
| Project coordinator | Scrum Master | XP Coach | Top Management | Team work | Class Owner | Leaders | Team work | Business People |

ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v16i1.28995

*Table 2 continued…*

| Characteristic | Scrum | Extreme Programming | Lean | Kanban | FDD | Scale | Crystal | DSDM |
|---|---|---|---|---|---|---|---|---|
| Role Assigned | Scrum Master, product owner, and development team. | No prescribed roles | Top Management, employee | No prescribed roles | No prescribed roles | No prescribed roles | No prescribed roles | Project Level - Solution Development Team - Supporting - Levels of Engagement - Fullfilling |
| Coding | No coding Standart | Have a coding standart | - | No Coding standart | Have a coding standart | Have a coding standart | - | - |
| Testing | No formal approach used for testing | Test driven development, including acceptence testing | Implementation process is characterized by continuous improvements | Testing done after implementation of each work product | No formal approach used for testing | - | Automated testing | Testing is carried out throughout the project life cycle |
| Nature of customer interaction: | Not compulsorily on-site | On-site customer interaction | - | Not compulsorily on-site | Client not involved | Customer Centric development process | On-site customer interaction | On-site customer interaction |
| Team Collaboration | Cross functional team | Self organizing teams | - | Team comprises of specialized resources | Individual by class owner | Self organizing teams | Not work well for distributed teams | Does not specifically address team size |

Based on the results of trend studies, Scrum is the most widely used method. This is in line with its characteristics, which are quite easy to apply for both users and developers because it is suitable for all types of projects, requires a relatively short duration of time, there are no coding standards to follow, can be done in a cross-functional team, and the resulting design is complex. Furthermore, there is the XP method, which occupies the second position for the trend of using Agile methods. XP has slightly different characteristics from the Scrum method. Based on its characteristics, XP is more suitable for projects with a small scale but has good risk management due to refactoring to simplify coding and accommodate unexpected design changes so that it has certain standards for coding. In its implementation, XP is carried out by self-organizing teams.

Furthermore, the scale method is the least used Agile method. Based on its characteristics, scale is indeed a method that is only suitable for projects with a large scale and requires a long time (8–12 weeks). While other methods are suitable for all types of projects, such as lean, kanban, crystal, and DSDM, it doesn't take that long to work on one project. Each agile method has its own advantages and disadvantages. So that the selection of Agile methods to develop software can be adjusted to the needs and pay attention to the characteristics of the method to be used,.

## CONCLUSION

The most pertinent studies for each research topic were selected for this systematic literature review by developing and using inclusion and exclusion criteria. Only eight process models—the Scale Agile Framework, Kanban, Scrum, Lean, Xtreme Programming Crystal, and Dynamic System Development Method—are covered in this study, which focuses on agile approaches. Based on 52 literatures discovered between 2018 and 2022, the results suggest that scrum is the most popular approach, with up to 26.4% of usage, while the Scale Agile Framework is the least popular, with 3.9% of usage.

According to a number of criteria, including the development approach, recommended iteration time period, team communication, project size, project documentation, design, work flow approach, project documentation, design, work flow approach, project coordinator, role assigned, coding, testing, and the nature of customer interaction, this study also examines the various characteristics of each process model. However, some attributes cannot be precisely described because of the limits of the many sources used to construct some models.

## REFERENCE

[1] S. Al-saqqa, S. Sawalha, and H. Abdelnabi, “Agile Software Development : Methodologies and Trends,” pp. 246–270.

[2] S. Aftab *et al.*, “Using FDD for Small Project : An Empirical Case Study,” vol. 10, no. 3, pp. 151–159, 2019.

[3] P. Sakulvirikitkul, K. Sintanakul, and J. Srisomphan, “The Design of a Learning Process for Promoting Teamwork using Project-Based Learning and the Concept of Agile Software Development,” vol. 15, no. 3, pp. 207–223.

[4] W. Huda, “Design of emergency response assessment system using Feature-Driven Development approach Design of emergency response assessment system using Feature-Driven Development approach,” pp. 1–7, 2021, doi: 10.1088/1757-899X/1098/3/032072.

[5] I. Freitas, “Describing the design thinking and extreme programming activities during a technology innovation academic workshop,” vol. 17, no. 3, pp. 267–284, 2020, doi: 10.1108/INMR-03-2019-0039.

[6] S. Baweja, M. Consultants, and N. Venugopalan, “Agility in Project Management 1,” vol. IV, no. X, pp. 1–14, 2015.

[7] M. Alqudah and R. Razali, “A Review of Scaling Agile Methods in Large Software Development,” vol. 6, no. 6, pp. 828–837, 2016.

[8] B. Chandra, “Systematic Review of Success Factors for Scaling Agile Methods in Global Software Development Environment : A Client-Vendor Perspective,” pp. 17–24, 2017, doi: 10.1109/APSECW.2017.22.

[9] P. Ciancarini, “Issues in the Adoption of the Scaled Agile Framework,” pp. 175–184, 2022, doi:

10.1145/3510457.3513028.

[10] S. Beecham, T. Clear, R. Lal, and J. Noll, "The Journal of Systems & Software Do scaling agile frameworks address global software development risks ? An empirical study," *J. Syst. Softw.*, vol. 171, p. 110823, 2021, doi: 10.1016/j.jss.2020.110823.

[11] C. Hofmann, S. Lauber, B. Haefner, and G. Lanza, "ScienceDirect ScienceDirect ScienceDirect ScienceDirect Development of an agile development method based on Kanban for Development of an agile development method based on Kanban for and Conference an introduction framework distributed part-time teams and an introduction framework Costing models for capacity optimization in Industry between used capacity operational efficiency," *Procedia Manuf.*, vol. 23, no. 2017, pp. 45–50, 2018, doi: 10.1016/j.promfg.2018.03.159.

[12] D. Sunner, "Agile : Adapting to need of the hour," pp. 130–135, 2016.

[13] M. S. G. Necmetting Ozkan, Sevval Bal, Tugba Gurgen Erdogan, "Scrum , Kanban or a Mix of Both ? A Systematic Literature Review," 2022, doi: 10.15439/2022F143.

[14] M. Tanner, "T HE U SE OF K ANBAN TO A LLEVIATE C OLLABORATION AND C OMMUNICATION C HALLENGES OF G LOBAL," vol. 14, 2017.

[15] S. Raza and U. Waheed, "Managing Change in Agile Software Development," *2018 IEEE 21st Int. Multi-Topic Conf.*, pp. 1–8, 2018.

[16] A. Srivastava, "SCRUM Model for Agile Methodology," pp. 864–869, 2017.

[17] T. Karabiyik, A. Jaiswal, and P. Thomas, "Understanding the Interactions between the Scrum Master and the Development Team : A Game-Theoretic Approach," no. 1, 2020.

[18] P. Ralph, "Science of Computer Programming The two paradigms of software development research," *Sci. Comput. Program.*, vol. 156, pp. 68–89, 2018, doi: 10.1016/j.scico.2018.01.002.

[19] L. A. Garcia, "Tailoring the Scrum framework for software development : Literature mapping and feature-based support," vol. 146, no. June, pp. 1–7, 2022.

[20] P. Stief, J. Dantan, A. Etienne, and A. Siadat, "ScienceDirect ScienceDirect ScienceDirect Digitalization of Lean Product Development Organization Digitalization of a a Development Organization A new methodology to analyze the functional and physical architecture of Germany family identification existing products for an assembly oriented product," *Procedia CIRP*, vol. 91, pp. 764–769, 2020, doi: 10.1016/j.procir.2020.02.232.

[21] A. A. Alzoubi *et al.*, "THE IMPACT OF LEAN PRACTICES AND AGILE PRACTICES ON," vol. 1, no. 1, pp. 100–117, 2022.

[22] M. D. Purvi Shanke; Shruti Mathur, Tasneem Bano Rehman, "Review of an Agile Software Development Methodology with SCRUM & Extreme," pp. 1–11, 2022, doi: 10.1109/CCET56606.2022.10080640.

[23] A. Singh, "Implementation of Requirement Engineering in Extreme Programing and SCRUM," vol. 8, no. 5, pp. 621–625, 2017.

[24] Y. A. Harb and C. Noteboom, "Evaluating Project Characteristics for Selecting the Best-fit Agile Software Development Methodology : A Teaching Case Evaluating Project Characteristics for Selecting the Best-fit," vol. 1, no. 1, 2015.

[25] T. R. S. Rafia Imran, "Mapping of Agile Processes into Project Management Knowledge Areas and Processes," 2022, doi: 10.1109/ICBATS54253.2022.9759013.

[26] J. Stapleton, "DSDM : Dynamic Systems Development Method," 1995.

[27] A. Tarhan and S. G. Yilmaz, "Systematic analyses and comparison of development performance and product quality of Incremental Process and Agile Process," *Inf. Softw. Technol.*, vol. 56, no. 5, pp. 477–494, 2014, doi: 10.1016/j.infsof.2013.12.002.

[28] A. O. Jarikre, Y. K. Sharma, and A. K. John, "Analysing the Obstacles in Agile Software Development Approach: A Review," vol. 5, 2022.

[29] C. Nigam and P. A. Rana, "Analytics of Agile Methodologies : An Empirical Review," pp. 1707–1710, 2016.
[30] V. Hurbungs and S. D. Nagowah, "FDD , Crystal , DSDM — An Educational," pp. 1–12, 2022, doi: https://doi.org/10.1007/978-981-19-3517-6_18.