International Journal of Advanced Science and Computer Applications
DOI: 10.47679/ijasca
https://ijasca.org/index.php/ijasca

RESEARCH ARTICLE

An Exploration into Hybrid Agile Development Approach

Siti Sarah Maidin

Abstract

The aim of this paper is to provide a review of the different hybrid agile models. This study raised the question “what are the types of hybrid agile models and their features”? The systematic review was done using Preferred Reporting Items for Systematic Reviews and Meta-Analyses Model for comprehensive searching. Scopus is used as the database for searching articles. A total of 131 papers related to agile and hybrid agile models were retrieved and finally after screening and filtering, only 26 papers were included in the study. This paper probes into the features of agile and hybrid agile models and thus recommends hybrid agile models as one of the best-suited models in software development due to several reasons. Some of the reasons are due to its comprehensiveness in managing a large-scale project with good documentation and developing better methods for business analysis. This paper concludes by providing insight into the different types of hybrid agile models in software development. This paper starts with an Introduction section, followed by a Materials and Method section, continued with a Results and Discussion section, and finally concludes the research in the Conclusion section.

Keyword: Software Development, Agile, Hybrid Agile, Development Approach, Models, Scrum

Introduction

Software development is a complex process that involves several distinct steps. Agile software development methodologies have provided significant benefits to software development methodology over the traditional waterfall. Although in principle each agile approach contains various processes, in practice some of its fundamental concepts are often disregarded [1]. The quicker agile development moves, the more likely it is that resources will be consumed. Thus, some companies go from pure agile development to a hybrid method that draws upon both the waterfall and agile methodologies. Hybrid agile is an approach to software development that incorporates both waterfall and iterative processes. Hybrid agile has been used well by many businesses to oversee massive projects, provide thorough documentation, and improve business analysis [2]. Collaboration with stakeholders to keep the development process on the correct track is one way in which combining a plan-driven development model with an agile approach can boost team productivity. Furthermore, pundak [3] highlighted that hybrid models are being utilized in software development projects due to the necessity of incorporating many approaches, each with its own set of benefits and drawbacks, into a single endeavor. The software engineering team has arrived at the decision to use a hybrid agile model as a result of the varied requirements of a software project, the shortcomings of agile approaches or the downsides of plan-driven development models, and the potential benefits promised by both agile and plan-driven development models. Therefore, this study will be looking into the systematic review of hybrid agile models which can be applied in any software development project. The main aim of the study is to explore different Hybrid Development Approaches. The research questions that need to be answered are What are the hybrid development approaches used by the software developer in the software development process? This section will explain the five main sub-sections, namely PRISMA, resources, the systematic review process, and data abstraction and analysis.

PRISMA

PRISMA or Preferred Reporting Items for Systematic Reviews and Meta-Analyses is a published standard to conduct a systematic review. PRISMA, or the "Preferred Reporting Items for Systematic Reviews and Meta-Analyses," is a quality assessment tool for reporting systematic reviews and meta- analyses that is widely used and supported by evidence. The PRISMA statement includes a checklist of 27 items and a flowchart to aid authors in providing clear and consistent reporting of all relevant details in a meta-analysis or systematic review.

This includes steps like finding relevant studies, determining their quality, extracting relevant data, and synthesizing the findings. In addition, PRISMA places special focus on review reports that analyze randomized trials; these reports can serve as the foundation for providing systematic reviews of other forms of research as well [4]. When conducting a systematic review or meta-analysis, it is important to follow the guidelines laid out in the PRISMA statement to ensure that your findings can be trusted. The PRISMA statement provides a standardized reporting format for systematic reviews and meta-analyses that allows readers to more easily understand the methods and results of these studies, as well as to compare and evaluate the results of these studies.

Several crucial steps are included in the PRISMA statement's guidelines for conducting a systematic review for article selection:

i. Identification of relevant studies: Finding studies that are relevant to your research question requires you to search multiple databases, including PubMed, Embase, and the Cochrane Library, as well as other sources, including conference proceedings and grey literature.
ii. Screening of studies: Studies are then screened for relevance and eligibility using inclusion and exclusion criteria that have already been established. Typically, this is done by a minimum of two reviewers who compare and contrast each study's abstract and full text.

Received 11 May 2023 | Received in revised form 10 Juli 2023 | Accepted 12 August 2023
Available online 14 September 2023

iii. Data extraction: Information on the study's design, population, interventions, outcomes, and results is extracted after all eligible studies have been found. The quality of the studies and the overall results are assessed and synthesized with the help of this data.
iv. Quality assessment: The included studies' quality is evaluated using predefined tools and criteria, such as the Cochrane Risk of Bias tool for randomized controlled trials and the Newcastle-Ottawa Scale for observational studies.
v. Synthesis of results: To answer the research question and draw conclusions, it is necessary to synthesize the data that was extracted. The results of multiple studies may be combined in this way if it makes sense, or a narrative synthesis may be performed instead.

All of these procedures are carried out methodically and meticulously to reduce the possibility of error and guarantee the quality of the final product. Guidelines on how to report each of these procedures are provided in the PRISMA statement, which aims to increase the openness and reproducibility of systematic reviews and meta-analyses.

Methods

Resources

The review utilized the SCOPUS database. Furthermore, [5] has highlighted that researcher could improve their chances of finding useful papers by scanning a reliable database.

The Systematic review process for selecting articles

The systematic review consists of four steps before the articles are selected for the analysis phase. The steps are listed below:

1) Articles identification

The systematic review process comprises three main phases which include the keyword and the searching process. The current research works were able to retrieve a total of 685 articles. In this study, a uniform search of keywords or key terms was defined. The keywords include "Hybrid Software," "Hybrid Development," and "Hybrid Agile," which were the main words of this research. The most relevant papers for the study from different research databases for the previous ten years (10) are listed in Table 1 and the outcome is the assessment from the combination of the research paper title, abstract, and keywords. The database used is Scopus. In total, 131 numbers of articles were returned from the search. After the filter, only 26 articles are accepted for review in this paper. The query used in the search is:

(TITLE-ABS-KEY (hybrid AND agile) AND TITLE-ABS-KEY (practitioner) OR TITLE-ABS-KEY (team) OR TITLE-ABS-KEY (software AND engineer) AND TITLE-ABS-KEY (software AND project) OR TITLE- ABS-KEY (development) OR TITLE-ABS-KEY (project)) AND PUBYEAR > 2012 AND PUBYEAR > 2012

2) Screening

The aim of the screening stage is to remove duplicate articles. The criterion used for the search is the type of article needed to be the journal article only, the language used for the publication is English, the year of publication is between 2013 and 2023 and the subject area is Information Technology and Computer Science. Overall, a total of 63 articles were excluded based on the above criterion. Fig.1 illustrates the screening criterion.

Criterion	Eligibility	Exclusion
Literature Type	Journal Articles	Journals (review), book series, book, chapter in the book, conference proceeding
Language	English	Non-English
Timeline	Between 2010 and 2023	<2010
Subject Area	Articles related to software development methodology and/or software security and/or hybrid agile and articles without hybrid agile framework/model	Articles are not related to software development methodology and/or software security and/or hybrid agile and articles without hybrid agile framework/model

Fig. 1. Example of a figure caption.

3) Eligibility

A total of 32 articles were prepared for the eligibility phase. During this stage, the title, abstracts, and important contents are observed to ensure it fulfills the inclusion criterion. Finally, a total of 32 articles were selected to be analyzed.

4) Data abstraction and analysis

Thematic analysis was used to construct themes and subthemes. First, data was compiled for theme development. In this step, the writers analyzed 26 papers to find answers to their study questions. In phase 2, the writers coded the data to establish meaningful categories. Figure 1 illustrates the flow diagram of the study.

.

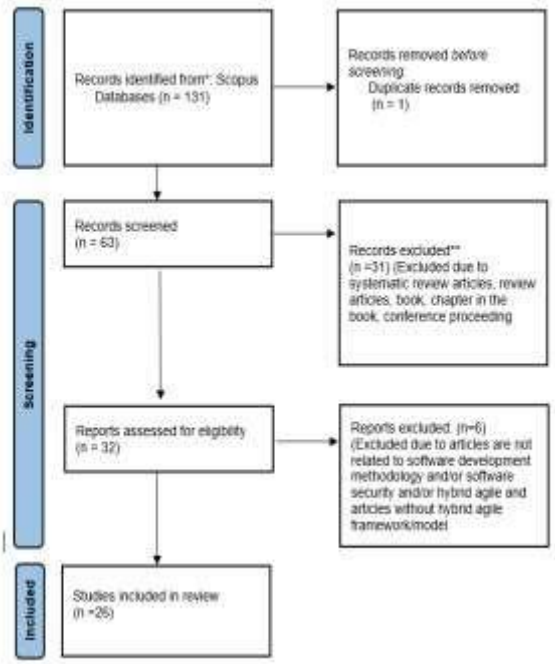

Figure 1 Study Flow Diagram

The process yielded two primary themes which is agile and hybrid agile. The sub-theme under hybrid agile includes Scrumfall, Hybrid Agile by Neelu and Kavitha, Waterscrumfall, Scrumbanfall, Blending Scrum and Waterfall Model and Hybrid approaches.

Results and Discussion

This section will further elaborate on the identified themes earlier which are agile and hybrid agile.

Agile

The term "agile methodology" refers to a set of principles, practices, and processes utilized by teams and organizations in the creation of products and services. It places an emphasis on teamwork, comments from customers, and flexibility. The primary advantage of adopting an agile approach is that it facilitates rapid iterations, which enable teams to rapidly and easily make improvements to products and services without having to start from scratch. A more open and collaborative atmosphere is fostered by the agile methodology, which further aids teamwork. Since customers can provide feedback throughout the development process and have their needs addressed rapidly and effectively, agile methodology can lead to increased customer satisfaction. Agile is a development approach that is both flexible and innovative. It's a strategy for thriving during the chaos and unpredictability that often characterizes software development projects. Solutions for the software project evolve through collaboration between self-organizing cross-functional teams using the appropriate practices for their context [6]. This is made possible by the fact that agile software development places an emphasis on the people doing the work and how they work together. The Agile Manifesto advocates for iterative and evolutionary approaches to software development, which shorten the development cycle and increase the quality of the result. The flexibility to adapt to new circumstances and alter strategies is another perk. However, the agile approach works best with a small team and a flexible setting for developing software [7]. As a lightweight process paradigm, agile places an emphasis on communication and collaboration between all stakeholders. Agile's primary intent is to shorten the duration of development cycles, increase software quality, and decrease overall costs.

Agile allows teams to respond quickly to changes in customer needs, keeps teams focused on the customer, and fosters close collaboration between developers and stakeholders. It also enables teams to stay focused on delivering high-value features that matter most to the customer. It helps teams be more efficient because they can respond to customer feedback more quickly, reducing the need to plan features far in advance and the risk of wasted effort. Furthermore, agile assists teams in reducing waste by focusing on quickly producing the most valuable features and eliminating features that are not valuable to the customer. Agile encourages iterative and evolutionary approaches to development, which shorten the time required to both create and deliver a product. The flexibility to adapt to new circumstances and alter strategies is another perk. Agile software development emphasizes four principles which are: 1) people and interactions over processes and tools; 2) working software over thorough documentation; 3) customer collaboration over contract negotiation, and 4) responding to change over following a plan [8]. Task prioritization is essential in the agile methodology because it is built on a manifesto that promotes a project environment that values adaptation, teamwork, self-organization, rapid delivery, and client focus [9]. When compared to other software development methodologies, the agile software development process stands out because of the incremental nature of its software releases, the speed with which its cycles are completed, and the emphasis placed on tight collaboration between clients and developers [10].

Hybrid Agile

Hybrid agile is the practice of combining two different software development methodologies—a plan-driven development model and an agile process model—into one. Improved project quality [11], on-time software delivery to end users [11], and more precise resource workload estimation [12] are just a few of the many benefits of adopting a hybrid agile approach. Furthermore, the needs of the user, not the process, are prioritized in hybrid agile, and the product is constantly refined. Software development uses hybrid

models because different techniques have distinct benefits [13]. Hybrid agile blends plan-driven and agile approaches [12]. The combination creates a detailed system [11]. Lom, Pribyl [13] claimed an agile project's product owner (PO) identifies user needs and expectations. The solution supplier and client are hybrid agile project product owners (PO-C). Both product owners must collaborate to ensure a project runs well and all stakeholders have the same focus and purpose. Hybrid agile uses many processes. Combines Scrum with Waterfall. ScrumFall produces high-quality, dependable products [11]. Imani, Nakano [14] suggested hybrid agile can scale to big project teams and improve project success rates, notably cost-benefit ratio likelihood. Many companies utilize hybrid agile to manage huge projects, generate documentation, and increase business analysis [13][14]. Hybrid agile features include speedy output, easy-to- change requirements, and modest user participation. According to the study done by [15], 76,8% of organizations implemented hybrid methods. Organizations utilize multiple frameworks, practices, and methods to provide efficient and effective development processes. It is for the continuous produce software products that meet stakeholders' requests and market needs [15]. The usage of hybrid agile depends on the types of process models that are combined. Hybrid Agile is an excellent option for software developers that want to reap the benefits of Agile but also wish to keep a more structured approach because it combines the best of both worlds. Additionally, it makes it possible to exercise a greater degree of control and accountability, which is particularly helpful when working with larger teams or projects that call for a greater degree of quality assurance. When it comes to projects that have high stakes, limited budgets, and limited amounts of time, Hybrid Agile is the solution that excels. Hybrid Agile is able to assist in ensuring that the final product exceeds expectations by utilizing an iterative methodology that places a primary emphasis on the delivery of value. To summarize, some of the benefits of using hybrid agile are which can boost productivity, resulting in better products, and customers satisfaction includes:

(a) Increased flexibility and adaptability: A Hybrid Agile approach can offer increased flexibility and adaptability to changing circumstances and requirements because it draws from multiple Agile methodologies.
(b) Business goal achievement: A Hybrid Agile approach can better align with an organization's overall business strategy and goals because it incorporates elements of traditional project management methodologies.
(c) Better stakeholder engagement: An advantage of adopting a Hybrid Agile methodology is the increased stakeholder engagement it fosters through increased participation in the process.
(d) Risk management: Risks and dependencies can be better managed with a Hybrid Agile approach, which combines elements of traditional project management methods with the flexibility of Agile methodology to produce results that are more consistent with the project owner's expectations.
(e) Improved scalability: An advantage of the Hybrid Agile methodology is its adaptability to projects of varying sizes and complexity. This is because the methodology can be scaled up or down as needed.

There was a project called HELENA (Hybrid and development approach in software systems development), which investigates the reality that organizations are using Hybrid methods. It is an international exploratory research project with multiple stages. According to [16], who are doing part of the HELENA project, most of the organizations use hybrid approaches goal is enhancing the frequency of delivery to clients, and the flexibility and adaptability of the process to respond to change. HELENA project aim is to maximize productivity. In another part of HELENA's project, [17] found that different methods, practices, and frameworks have been used in combination with hybrid methods. It is happening to the organization of all industry sectors and sizes. They also discovered that the practices have restricted dependencies to the methods in hybrid. Hybrid inherited the concepts of combination. For instance, hybrid agile is inherited based on the combination of agile and non-agile techniques and methods. The pro and cons of each technique and method are used to overcome the strength and weaknesses of each other. This is where the uniqueness of hybrid-agile is highlighted.

Scrumfall

Scrumfall is a fusion of the Scrum methodology for software development and the Waterfall approach. Scrum and Waterfall can be thought of as a hybrid of Agile and a more traditional waterfall paradigm of software development. Both the plan- driven and agile approaches have advantages that make them more suitable for project sizes and types. In contrast to Agile, which is better suited to small systems and requires specialist agile workers throughout the project, plan-driven models are better suited to large, stable systems and require experienced personnel at the outset of the project. Practitioners have found that ScrumFall works well for both large, critical systems and geographically dispersed, huge teams that consist of a mix of seasoned veterans and fresh faces. ScrumFall has also been proven to be efficient in terms of time, money, and the bottom line. Soon, we hope to use this model across other organizations to test its efficacy and refine it [18]. Figure 2 depicts a typical Scrumfall in which a software project begins with the Waterfall model for the planning, estimation, and high-level design, and then Scrum takes over development, which includes testing and delivery of modules based on sprint. Finally, integration testing and system testing occur in the Waterfall model, and the product is released and documented.

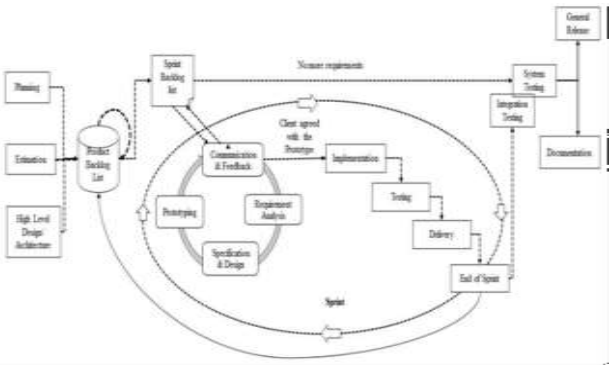

Figure 2 Scrumfall model [7]

Hybrid Agile by Neelu and Kavitha

Neelu and Kavitha [20], in their paper titled "Estimation of software quality parameters for hybrid agile process model," presented the "Hybrid Agile" process model for software development. Hybrid Agile is short for "Hybrid Agile software development. According to the findings of their research, Hybrid Agile is a methodology that combines Agile with more traditional approaches, such as Waterfall. It is intended to provide a more well-rounded strategy for the development of software in order to compensate for some of the deficiencies that are inherent in the pure Agile or Waterfall methodologies.

Neelu and Kavitha [20] present a methodology for estimating software quality parameters applicable to the Hybrid Agile process model within the context of their research paper. They present a set of metrics and methods that can be used to evaluate the quality of software that was created using the Hybrid Agile model. In addition to this, the paper offers a summary of the Hybrid Agile process model and explains how it is distinct from both traditional Agile and Waterfall approaches.

It is essential to keep in mind that this is a novel and specialized method; additional study, as well as application in the real world, are required to achieve a comprehensive understanding of the advantages and disadvantages of this approach. Hybrid agile models such as those presented by Neelu and Kavitha [20] are best suited to software development projects and depicted in Figure 3.

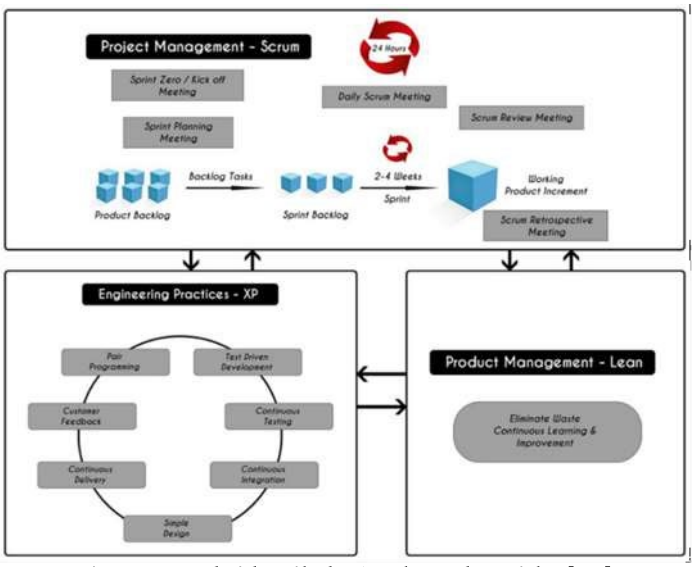

Figure 3: Hybrid Agile by Neelu and Kavitha [20]

Scrum, eXtreme Programming, and Lean are all examples of agile software process frameworks and techniques [21][22]. The approaches are blended to capitalise on their complementary qualities. Scrum is utilised for project management, XP for the engineering process (which emphasises coding and testing) and Lean for the ongoing education and development of software. According to Githens [23], Lean software development has the potential to improve morale by allowing the development team to take responsibility for making decisions that directly affect the project's progress and reducing wasteful activities, expenses, and delays. However, Extreme Programming is a community- driven set of practices that aims to speed up the delivery of high-quality software while also allowing for its adaptation to the dynamic needs of businesses. XP covers all aspects of software development for a small team, from planning to testing to coding to deploying to maintaining [24]. Figure 3 depicts a complete software development life cycle using a combination of Scrum, XP, and Lean software development and XP [20]. In contrast, eXtreme Programming lacks planning and performs poorly on medium- and large-scale projects.

Waterscrumfall model

The Agile methodology and the traditional Waterfall method are both components of the hybrid project management model known as Waterscrumfall. Waterscrumfall combines elements from both of these methodologies. The term "Waterscrumfall" was coined to describe a situation in which an Agile approach to project management is utilized, but there are also components of the Waterfall model in play at the same time. The development of the product follows an iterative process that is centered on making incrementally smaller improvements. This is how the Agile methodology works. This makes it possible to quickly adapt to changes in the requirements or the needs of the customer. In contrast, the Waterfall method is an approach to development that follows a linear and sequential progression, where each phase of the process must be finished before moving on to the next.

The Waterscrumfall model integrates Agile development practices into the more conventional Waterfall framework in an effort to strike a balance between the two approaches. For instance, the overall project timeline and milestones may adhere to a sequential Waterfall methodology, but the development team may make use of agile techniques such as Scrum to manage their work and ensure that they remain focused on providing value to the customer throughout each iteration. This hybrid strategy can be helpful for companies that need to keep some level of control and predictability in the timelines of their projects, while at the same time benefiting from the adaptability and flexibility of Agile methodologies. Hybrid agile methodologies are used in the majority of Agile implementations, [25]. The fact that Agile adoption has been practitioner-driven has contributed to the popularity of the hybrid model by convincing groups to focus on what they can influence most: themselves. But because the development team has little say in other departments' practices, such as business analysis and release management, they can only adopt Scrum at the team level [25]. Figure 4 depicts the overlap between Waterfall and Scrum during the development phase of a software project, with Waterfall used toward the end.

.

Scrumbanfall

Scrumbanfall is a cost-efficient process methodology that aims to improve agile software development by combining elements of Scrum and Kanban.

Scrum is a popular agile framework for managing software development projects, while Kanban is a visual method for managing work in a just-in-time manner. By combining the strengths of both approaches, [27] propose that Scrumbanfall can help organizations achieve greater efficiency and cost savings in their software development projects. Scumbanfall is a software development methodology that combines elements of the Scrum, Kanban, and Waterfall models [26][27]. the Scrumbanfall paradigm was proposed for use in Software Engineering Management [28]. Scrumbanfall is a model that combines the best features of Scrum, Kanban, and Waterfall into a single framework, giving software development organizations a significant advantage over those using only one of these approaches. External stakeholders' active participation in documenting project requirements and analyses; planning, estimating, and trackers presents a number of difficulties.

Blending Scrum and Waterfall model

A hybrid approach to the development of software is one that combines elements of different software development methodologies, such as Scrum and the Waterfall model. The Waterfall model is a more traditional, sequential approach to software development that involves distinct phases of planning, analysis, design, implementation, testing, and deployment. The Scrum methodology is an Agile approach that emphasizes collaboration, flexibility, and continuous improvement. In contrast, the Waterfall model is an Agile approach that emphasizes continuous improvement.

In a model that combines Scrum and Waterfall, the positive aspects of both methodologies are combined to produce a workflow that is both more efficient and more effective. Scrum, for instance, is a framework that can be used to manage the iterative development and delivery of software features, whereas the Waterfall model can be used to structure the testing phase of the project. Both models are examples of methodologies that can be used. This can help to ensure that a high level of quality and reliability is maintained in the software, while at the same time allowing for the agility and flexibility that is offered by Scrum.

It is essential to keep in mind that combining the Scrum model and the Waterfall model can be difficult because the two methodologies take distinct approaches to the planning, development, and deployment of their respective products. It is possible that careful consideration and adaptation are required in order to guarantee that the benefits of both strategies are realized, as well as that the project is completed on time and without exceeding its budget.

Figure 4 shows the findings of combining Scrum and Waterfall methods by Singhto and Phakdee [1]. This case study analyses the benefits of combining the Scrum and Waterfall methodologies for the creation of a suite of Tailor- made Software as a Service (SaaS) to aid Small and Medium-Sized Enterprises (SMEs) in the eastern region of Thailand in taking control of their operational procedures.

The development process is broken down into five stages: planning, analysis, design, development, and maintenance. Planning and analysis in Singhto and Phakdee [29] are executed utilizing the Waterfall model. Scrum is used throughout the development phase, while Waterfall is used during the design and maintenance phases.

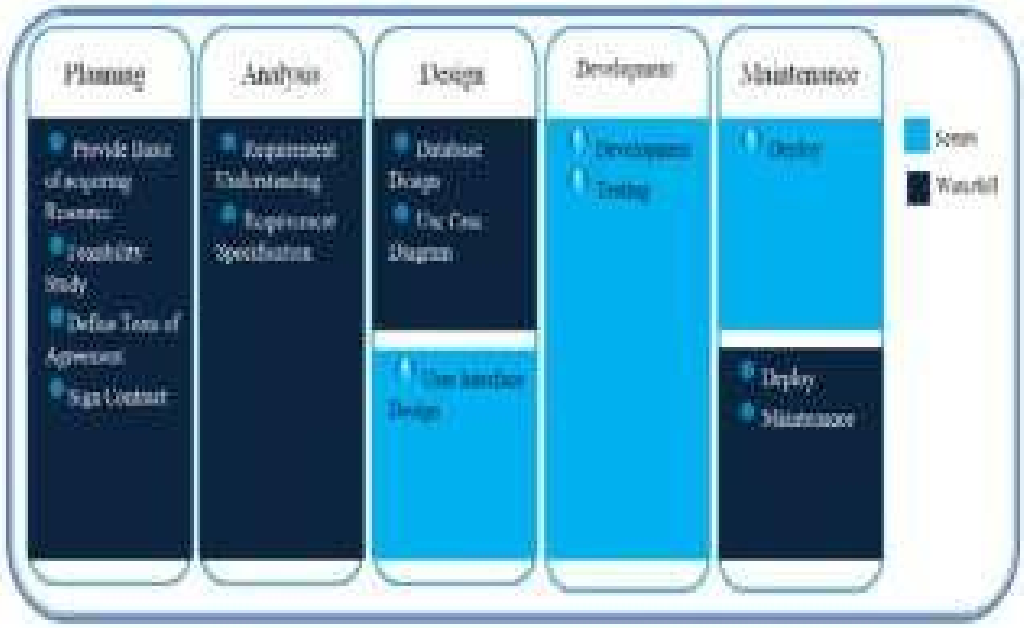

Figure 4: Blending Scrum and Waterfall Model [1]

Hybrid approaches

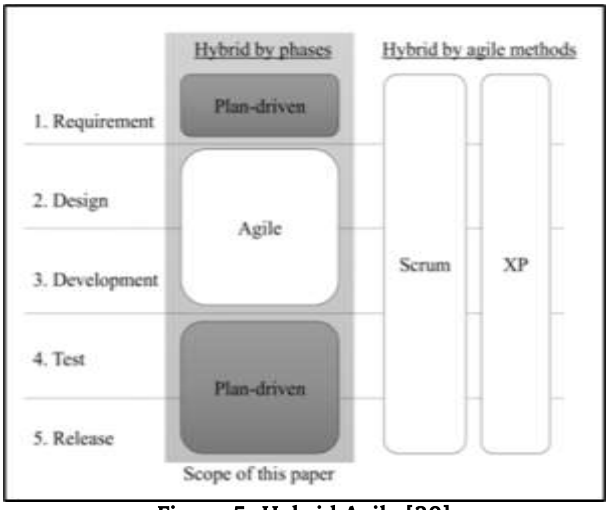

Figure 5: Hybrid Agile [30]

Anantatmula have written a paper titled "Does a hybrid approach of agile and plan-driven methods work better for IT system development projects?" This paper was published by T. Imani. The authors of this publication have conducted research into the application of hybrid approaches in the development of information technology systems. These approaches combine aspects of both Agile and plan-driven methodologies. [14] pointed out that despite the fact that Agile methodologies have become increasingly popular in recent years due to their capacity to respond rapidly to changing requirements, it can be difficult to put these methodologies into practice in actual situations. On the other hand, plan-driven methodologies provide a method of development that is more organized and predictable, but they can be less adaptable and responsive to changes.

Imani, Nakano, & Anantatmula [14] believe that a solution to these problems can be found through the utilization of a hybrid strategy that takes the most beneficial aspects of both Agile and plan-driven methodologies. By combining aspects of Agile and plan-driven methodologies, businesses can reap the benefits of Agile's adaptability and flexibility while still retaining the level of structure and predictability that is provided by plan-driven methodologies. [14] present the findings of a case study that compares the use of a hybrid approach to traditional Agile and plan-driven methodologies. This comparison offers insights into the advantages and disadvantages of utilizing hybrid approaches in the development of information technology systems.

A quantitative and qualitative study by [14] demonstrated the superiority of hybrid approaches over the more conventional plan-driven and agile approaches. Both the scalability of the hybrid strategy for projects with high levels of requirement uncertainties and the improvement of project success rates, especially in terms of cost, have been quantitatively proved in this study. Quantitative research tested two hypotheses about the effects of hybrid agile on businesses, while the qualitative study employed a case study to gather information about two such companies. According to the results of the case study, hybrid agile can be successfully applied to a short-term project with significant requirement uncertainty. As an added bonus, the project was finished on schedule and under budget thanks to the agile iterative development and plan-driven test phase.

Figure 5 shows the model developed by [14]. Hybrid approaches can be broken down into two distinct categories, which Imana et al. [30] call (1) hybrid by phases and (2) hybrid by agile methodologies. Both the classic plan-driven method and the more flexible agile methods are used in the hybrid-by-phases approach. The hybrid approach to agile development makes use of a combination of several agile practices, such as Scrum and XP, as well as a plan-driven estimating tool. The two kinds of models have the same five stages: requirement, design, development, testing, and project release.

Wysocki and Orłowski [32] reported that ScrumFall consists of three phases. The initial phase, the development phase, and the final phase. The initial phase includes requirements analysis and planning and is based on the Waterfall model. Scrum is used in the development phase, which includes the design, development and implementation phases. The integration and testing phases are carried out in the final completion phase [32]. Figure 6 illustrates the Scrum waterfall approach waterfall model diagram and the associated activities.

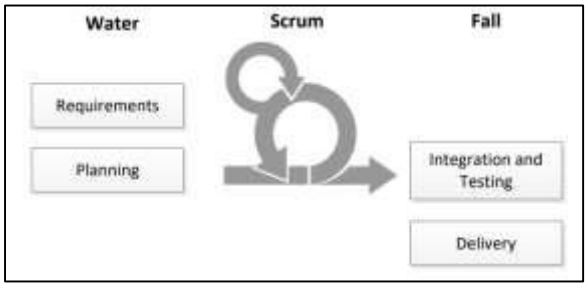

Figure 6: Water Scum Fall as Software Process [32]

Conclusions and Recommendations

Based on the review, it can be concluded that no single methodology can be used for all projects; all agile and non- agile methodologies must be adapted and integrated in order to successfully complete a variety of tasks [30]. Although Scrum has its benefits, it is not the best choice for software development without being combined with another process model, such as the plan-

driven development model Waterfall. The main focus is on how the preferred model would assist the software development team in achieving the project objectives regardless of the project size nor the software development team's preferences. Compared to plan-driven techniques, hybrid agile improve the cost- benefit ratio and is also influenced by the project requirement [31]. Therefore, it can be concluded that hybrid agile is the best solution for the software development team in achieving its project objectives, accelerating the development process, and increasing the team's productivity.

Acknowledgements

This work is supported by the Ministry of Education (MOE) through Fundamental Research Grant Scheme (FRGS/1/2019/ICT01/UIAM/03/2).

References

[1] D. Ionita, C. Van Der Velden, H. K. Ikkink, E. Neven, M. Daneva, and M. Kuipers, "Towards Risk-Driven Security Requirements Management in Agile Software Development," Inf. Syst. Eng. Responsible Inf. Syst. CAiSE 2019, vol. 350, pp. 133–144, 2019.

[2] T.Imani, M. Nakano, and V. Anantatmula, Does a hybrid approach of agile and plan-driven methods work better for IT system development projects. International journal of engineering research and applications, vol. 1, no. 2, pp.3, 2017.

[3] M. Špundak, Mixed agile/traditional project management methodology– reality or illusion? Procedia-Social and Behavioral Sciences, vol. 119, pp. 939-948, 2014.

[4] D. Moher, A. Liberati, J. Tetzlaff, and D. G. Altman, "Preferred Reporting Items for Systematic Reviews and Meta- Analyses: The PRISMA Statement," PLoS Medicine, vol. 6, no. 7, p. e1000097, Jul. 2009, doi: 10.1371/journal.pmed.1000097.

[5] P. Younger, "Using Google Scholar to conduct a literature search," Nursing Standard, vol. 24, no. 45, pp. 40–46, Jul. 2010, doi: 10.7748/ns.24.45.40.s51.

[6] D. Strode, T. Dingsøyr, and Y. Lindsjorn, "A teamwork effectiveness model for agile software development," Empirical Software Engineering, vol. 27, no. 2, Mar. 2022, doi: 10.1007/s10664-021-10115-0.

[7] S. Rahim, A. E. Chowdhury, D. Nandi, and M. Rahman, "ScrumFall: A Hybrid Software Process Model," International Journal of Information Technology and Computer Science, vol. 10, no. 12, pp. 41–48, Dec. 2018, doi: 10.5815/ijitcs.2018.12.06.

[8] Alliances, A., 12 Principles Behind the Agile Manifesto. 2019. [Accessed: 28-Sep-2022].

[9] K. Elghariani, N. Kama, N. Firdaus, and N. Azaliah, "Implicit Thinking Knowledge Injection Framework for Agile Requirements Engineering," International Journal of Advanced Computer Science and Applications, vol. 9, no. 11, 2018, doi: 10.14569/ijacsa.2018.091121.

[10] P. Abrahamsson et al., Agile software development methods: Review and analysis. arXiv preprint arXiv:1709.08439, 2017.

[11] S. Bhalerao and M. Ingle, "Hybrid agile project management practices: Novel approach for improving project management", IJRAR- International Journal of Research and Analytical Reviews (IJRAR), vol. 7, no.1, pp. 723-727.

[12] F. S. Pires, C. A. Pahins and P. Fonseca, Improving Workload Estimation In Inexperienced Teams With Hybrid Agile Approach, https://doi.org/10.5121/csit.2020.101426.

[13] J.I. Janjua, A. Ali, and M.U. Chaudhry, OPENTCQ: Towards change management in hybrid agile model, Proceedings of the 7th International Conference on Computing Communication and Networking Technologies, 2016. DOI: https://doi.org/10.1145/2967878.2967920.

[14] T.Imani, M. Nakano, and V. Anantatmula, Does a hybrid approach of agile and plan-driven methods work better for IT system development projects. International journal of engineering research and applications, vol. 1, no. 2, pp.3, 2017.

[15] J. Klünder et al., "HELENA Study: Reasons for Combining Agile and Traditional Software Development Approaches in German Companies," Springer Int. Publ., vol. 10611, pp. 428–434, 2017.

[16] P. Tell et al., "What are Hybrid Development Methods Made Of? An Evidence-Based Characterization," in 2019 IEEE/ACM International Conference on Software and System Processes (ICSSP), 2019, pp. 105– 114.

[17] T. Hayata and J. Han, "A Hybrid Model for IT Project with Scrum," in Service Operations, Logistics, and Informatics (SOLI), 2011, pp. 285– 290.

[18] S. Rahim, A. E. Chowdhury, D. Nandi, and M. Rahman, "ScrumFall: A Hybrid Software Process Model," International Journal of Information Technology and Computer Science, vol. 10, no. 12, pp. 41–48, Dec. 2018, doi: 10.5815/ijitcs.2018.12.06.

[19] "What is Scrum?, Scrum.org," Scrum.org. https://www.scrum.org/resources/what-is-scrum [Accessed: 30-Sep- 2022].

[20] L. Neelu and D. Kavitha, "Estimation of software quality parameters for hybrid agile process model," SN Applied Sciences, vol. 3, no. 3, Feb. 2021, doi: 10.1007/s42452-021-04305-0.

[21] W. Zayat and O. Senvar, "Framework Study for Agile Software Development Via Scrum and Kanban," International Journal of Innovation and Technology Management, vol. 17, no. 04, p. 2030002, Jun. 2020, doi: 10.1142/s0219877020300025.

.

[22] K. Bhavsar, Dr. V. Shah, and Dr. S. Gopalan, “Scrum: An Agile Process Reengineering In Software Engineering,” International Journal of Innovative Technology and Exploring Engineering, vol. 9, no. 3, pp. 840– 848, Jan. 2020, doi: 10.35940/ijitee.c8545.019320.

[23] G. Githens, “Product Lifecycle Management: Driving the Next Generation of Lean Thinking by Michael Grieves,” Journal of Product Innovation Management, vol. 24, no. 3, pp. 278–280, May 2007, doi: 10.1111/j.1540-5885.2007.00250_2.x.

[24] K. Beck and M. Fowler, Planning extreme programming. 2001: Addison- Wesley Professional.

[25] D. West et al., "Water-scrum-fall is the reality of agile for most organisations today", Forrester Research, vol. 26., pp. 1-17, 2011.

[26] K. Bhavsar*, Dr. V. Shah, and Dr. S. Gopalan, “Scrumbanfall: An Agile Integration of Scrum and Kanban with Waterfall in Software Engineering,” International Journal of Innovative Technology and Exploring Engineering, vol. 9, no. 4, pp. 2075–2084, Feb. 2020, doi: 10.35940/ijitee.d1437.029420.

[27] E.E. Hassanein and Hassanien S.A., "Cost Efficient Scrum Process Methodology to Improve Agile Software Development", International Journal of Computer Science and Information Security (IJCSIS), vol. 18, no. 4, 2020.

[28] M. Fowler and J. Highsmith, "The agile manifesto", Software development, vol. 9, no. 8, pp. 28-35, 2001.

[29] W. Singhto, and N. Phakdee, "Adopting a combination of Scrum and Waterfall methodologies in developing Tailor-made SaaS products for Thai Service and manufacturing SMEs." in 2016 International Computer Science and Engineering Conference (ICSEC), IEEE, 2016.

[30] A. Mahanti, "Challenges in Enterprise Adoption of Agile Methods - A Survey", Journal of Computing and Information Technology, vol. 14, pp. 197-206, 2006. DOI: 10.2498/cit.2006.03.03.

[31] M. Kuhrmann, P. Diebold, J. Munch and P. Tell, Hybrid software development approaches in practice: a European perspective, IEEE Software, vol. 36, no. 4, pp. 20-31, 2018, doi: 10.1109/MS.2018.110161245.

[32] W. Wysocki and C. Orłowski, "A multi-agent model for planning hybrid software processes," Procedia computer science, vol. 159, pp. 1688-1697, 2019, doi: https://doi.org/10.1016/j.procs.2019.09.339.

.