POSTER PRESENTATIONS

The Impact of Agile Practices on Software Evolution in Startup Companies

NTD Dharmasiri [1#] , JP Kodithuwakku [2], PKBTD Pallawala[3] and , BNS Lankasena[4]

[1,2,3]Department of Computer Science, General Sir John Kotelawala Defence University, Sri Lanka
[4]Department of Technology, University of university of Sri Jayewardenepura, Sri Lanka

37-se-0014@kdu.ac.lk

Abstract—*Over the past few decades, the software development industry has embraced agile approaches. Intending to create high-quality software solutions that satisfy customer expectations, these methods place a strong emphasis on iterative development, adaptability, and customer collaboration. This study examines how Agile practices have affected the development of software in Startup companies. The objective is to obtain a thorough knowledge of the impact of Agile on startup companies. The research was conducted using several approaches including a thorough examination of the literature, an interview with a software engineer, and a survey given to a group of software developers. The results imply that Agile techniques have an advantageous effect on software quality, project effectiveness, and client satisfaction. The study highlights the benefits of Agile implementation and addresses challenges during adoption. In order to promote software evolution and improve project results, startups are recommended to adopt Agile practices. Future studies in this field could examine the effects of particular Agile practices on the evolution of software in various organizational contexts as well as the elements that contribute to the effective adoption of Agile methodology. Overall, this study adds to our understanding of Agile methods and how they affect the evolution of software in Startup companies, serving as a useful tool for businesses looking to enhance their software development procedures.*

Keywords— Agile, Software Evolution, Software Maintenance

I. Introduction

Agile software development methodologies gained a lot of interest in recent years due to their potential to improve the methods and outcomes of software development(Sindhgatta et al., 2010). In this proposed study, we seek to investigate how agile development approaches impact software evolution in startup companies. In particular, we'll examine how much the use of agile principles impacts the performance of software evolution, notably its efficacy and efficiency. The project will encompass a variety of agile approaches, including, but not limited to, Scrum, Kanban, and Extreme Programming (XP). We will also consider the issues and difficulties associated with putting agile methodologies into reality, as well as solutions for these concerns(Lai et al., 2020).

The proposed study is justified by the growing use of agile methods in software development and the need for evidence of their efficacy. Despite the fact that several research has looked at how agile approaches impact software development processes, few have explicitly explored how agile methods affect software evolution in startup companies. It is critical to consider how agile methodologies could assist software evolution and guarantee that software systems remain adaptable and maintainable as they get more sophisticated and dynamic(Diebold et al., 2019).

A problem in the field of study is the dearth of empirical evidence on the impact of agile methodologies on software evolution. Less is known about how agile techniques could impact software evolution in startup companies, yet it has been observed that they improve software development processes. This research question is crucial because it limits software development teams' ability to rationally adopt agile methodologies for software evolution(Kruchten, 2013).

The primary objective of this study is to examine the impact of agile methodologies on software evolution in startup companies. We will look at the following particular aims to achieve this goal: The most common agile practices used during software evolution are identified, their performance on software evolution is assessed, the challenges and difficulties associated with implementing agile practices for software evolution are identified, and solutions to these challenges are suggested. Even if we haven't yet come up with any specific hypotheses for this project, we will nevertheless come up with research questions to guide our investigation(Ghimire and Charters, 2022). (1) What are the most common agile techniques used in software development, and how are they applied? is one of these research subjects. (2) How do agile methodologies impact the efficacy and efficiency of software evolution? (Ghimire and Charters, 2022) (3) What challenges and issues occur when using agile methodologies for software evolution, and how may these challenges be resolved?

This study seeks to achieve two objectives. Our initial objective is to get a theoretical understanding of how agile methodologies impact software evolution in startup companies (Nierstrasz, 2004). Through the identification of the most effective agile practices and strategies for resolving implementation issues, this study will provide empirical evidence for the use of agile methodologies for software evolution. We also want to enhance people's lives by improving software evolution performance and developing more adaptable and maintainable software systems. The suggested research might improve software development practices and outcomes by deepening our understanding of software engineering(Honoré et al., 2021).

II. Literature review

To find out how Agile principles affected the evolution of software in open-source projects, Babar undertook a case study. In the study, the development of four different open-source projects using Agile techniques like Scrum and Extreme Programming was examined. The findings demonstrated that applying Agile principles enhanced the productivity of projects and the quality of their products. The study also emphasized the value of ongoing testing and integration in Agile software development. Additionally, the study showed that the application of Agile techniques promoted team member cooperation and communication, leading to better software evolution. Overall, the research indicates that the application of Agile principles may benefit the development of software in open-source projects(Waseem and Ikram, 2016).

An investigation of the effects of Agile techniques on software maintenance was done by Sultana, Ahmed, and Hossain. The study examined how well Agile approaches like Scrum and Continuous Integration were being used in the maintenance of a software system. The findings demonstrated that applying Agile techniques enhanced maintenance performance, including a decrease in the time and effort needed for maintenance jobs. In addition, the study discovered that Agile practices promoted team member interaction and communication, resulting in enhanced maintenance procedures and a better understanding of the software system. The study's findings suggest that applying Agile principles to software maintenance projects can be advantageous(Ghimire and Charters, 2022).

An investigation of the effects of the Scrum approach on software evolution was done by Zhang, Wang, and Xu. In the study, the development of a software system that had incorporated Scrum methods including Sprint planning, Daily Standups, and Retrospectives were examined. The outcomes demonstrated that the application of Scrum methods enhanced the quality of software and project outcomes. The study also discovered that Scrum principles improved teamwork and communication, resulting in better software evolution. The study also showed that Scrum principles promoted continual development and flexibility in response to shifting requirements. The study comes to the conclusion that applying Scrum methods can accelerate the development of software(Tarwani and Chug, 2016).

To research the effect of Agile techniques on software evolution, Javed, Khan, and Babar carried out a methodical literature study. In order to complete the study, 55 research publications on Agile methods and software evolution were examined. The findings demonstrated how applying Agile techniques can increase software quality, project effectiveness, and customer happiness, all of which are important factors in the evolution of software. The study also discovered that Agile techniques improved team member communication and collaboration, resulting in better software evolution. The study also showed that Agile techniques promoted continual improvement and adaptation to shifting requirements. According to the study's findings, better software evolution can result from the application of Agile techniques(Weichbroth, 2022)

The study "The Influence of Agile Methodology (Scrum) on Software Project Management" by Faisal Hayat and colleagues provides a detailed analysis of Scrum's influence on software project management. The authors highlight the drawbacks of traditional project management techniques for software development and argue that agile techniques like Scrum are better suited to the fluid nature of software development. The paper outlines a number of benefits of Scrum, including more flexibility, increased stakeholder involvement, improved communication and cooperation, and improved project outcomes. The authors make the point that Scrum enables teams to better adapt to changing needs and respond to customer demands, resulting in higher product quality and happier customers. The study also looks at the difficulties teams could have while implementing Scrum as well as the monetary advantages of employing Scrum in software project management, such as greater customer satisfaction, enhanced competition, and cost savings. Overall, by providing a detailed analysis of the real-world impacts of Scrum on software project management and highlighting the relevance of agile methodologies in the software development industry, the study contributes to the body of knowledge(Hayat et al., 2019).

The paper "Agile Software Development using Cloud Computing: A Case Study" by Muhammad Younas and others examines cloud computing's role in agile software development. The authors emphasize the benefits of cloud computing for agile software development, including scalability, flexibility, and cost-effectiveness. The article provides a case study that illustrates how using cloud computing in an agile software development project may improve communication, speed up development, and save expenses. The authors make the point that cloud computing allows teams to access resources and tools, which also makes it simpler for team members to collaborate and communicate. The paper also discusses challenges associated with implementing cloud computing in agile software development, such as security concerns and the need for specialized skills. The authors make the case that cloud computing might significantly improve the speed and effectiveness of software development and provide solutions for these issues. Overall, by providing a realistic assessment of cloud computing's application in agile software development and recognizing both its possible benefits and drawbacks, the research contributes to the body of knowledge(Younas et al., 2019).

"Agile Software Development: Methodologies and Trends" by Samar Al-Saqqa, Samer Sawalha, and Hiba AbdelNabi is a thorough analysis of Agile software development methodologies and trends. The authors begin by discussing the fundamental principles of Agile software development, which include customer collaboration, adaptability, and continuous improvement. Following that, they discuss other Agile methodologies, such as Scrum, Extreme Programming (XP), and Kanban, as well as the specific steps and practices related to each. Additionally, the authors go over the benefits of implementing Agile software development, citing research that suggests it can increase customer satisfaction, decrease time to market, and enhance product quality. They point out that Agile methodologies have lately gained popularity and are being used in a variety of industries and contexts, including healthcare and education. The authors also point out an increase in interest in the DevOps movement, which strives to bring together software development and IT operations. They note that this tactic complements agile methodologies and can help businesses manage the software development lifecycle more successfully. Overall, the writers offer a thorough examination of the methodologies and developments found in Agile software development research. In addition to highlighting how Agile methodologies are evolving and adapting to meet the changing needs of software development teams and businesses, they underscore the need of cooperation, adaptation, and continual improvement in achieving success with Agile software development(Al-Saqqa et al., 2020).

"The Impact of Agile Approaches on Software Quality Attributes: An Empirical Study" by Doaa M. Shawky and Salwa K. Abd-El-Hafiz conducts an empirical study to examine the influence of Agile methodology on software quality attributes. The authors note that, despite the increasing popularity of agile approaches in recent years, empirical research on these approaches' effects on software quality is still in its early stages. To address this gap, the authors' survey of Egyptian software engineers and managers focused primarily on the effects of Agile approaches on the seven software quality criteria of maintainability, reliability, usability, efficiency, portability, compatibility, and security. The authors also note that their findings are consistent with past research that suggested Agile methodologies could improve software quality. They underline the importance of Agile practices like continuous integration, frequent testing, and user interaction in order to realize these advantages. Their results demonstrate the importance of incorporating Agile concepts into software development processes and suggest that Agile methodologies may improve product quality(Shawky and Abd-El-Hafiz, 2014).

The article "Software Evolution in Agile Development: A Case Study" by Renuka Sindhgatta, Nanjangud C. Narendra, and Bikram Sengupta examines software evolution in Agile development. Despite the fact that agile methodologies are well known for their quick and efficient software delivery, the authors note that little is known about how they encourage software evolution over time. In order to close this gap, the authors carried out a case study of a software development team using Agile methods. They looked at the team's process for creating software as well as how the final product has evolved over time, paying close attention to elements like maintainable code and scalability. Their research indicates that Agile methodologies might aid software evolution by enabling teams to respond quickly to changing needs and iterate on software designs. The authors claim that the team's ability to often integrate and test their code allowed them to maintain code quality and identify issues early in the development cycle. They also found that the project's scalability and maintainability were improved by the team's use of automated testing and code review tools. Their study found that using Agile methodologies can help teams quickly adjust to changing needs and improve software quality and maintainability by incorporating ongoing testing and code review(Sindhgatta et al., 2010).

Ned Chapin's article "Agile Methods' Contributions in Software Evolution" explores the advantages of Agile methods. The author makes the argument that traditional software development processes, which were usually focused on delivering a final software product, do not prioritize iterative development and continuous improvement as highly as Agile techniques do. Chapin explains how Agile methodologies promote collaboration, flexibility, and continuous feedback to facilitate software evolution. He makes the argument that using Agile methodology enables teams to respond quickly to changing requirements, consider customer feedback, and continuously improve software design. The author also underlines the importance of Agile methodologies like continuous integration, automated testing, and refactoring in order to aid software progress. He makes the point that by using these processes, teams may maintain high-quality code and increase the program's endurance over time. The author emphasizes the value of specific Agile techniques in supporting software evolution over time and emphasizes the importance of cooperation, flexibility, and constant input in achieving success with Agile software development(Chapin, 2004).

Table 5 Literature review summary

Research Paper	Objectives	Problem	Methodology
Waseem and Ikram (2016)	Examine how architecting tasks have changed and emerged in Agile software development	Insufficient knowledge of how Agile development affects architectural tasks	Systematic literature review and analysis
Ghimire and Charters (2022)	Analyze how Agile development techniques affect project results.	There is little empirical evidence on how Agile techniques affect project results.	Systematic literature review and meta-analysis
Weichbroth (2022)	Examine the justification, advantages, drawbacks, and commercial implications of applying Agile methodologies and practices to the creation of hardware.	There is little study on using Agile in hardware development.	Case study
Hayat et al. (2019)	Examine how the Agile methodology (Scrum) has an impact on the management of software projects.	There is a dearth of empirical evidence on how Agile techniques affect software project management.	Case study and survey
Younas et al. (2019)	Examine the utilization of cloud computing in Agile software development.	Lack of knowledge on the effects of adopting cloud computing for Agile development	Case study
International Journal of Interactive Mobile Technologies	Analyze the methods and trends used nowadays in software development.	Lack of a thorough grasp of current developments in and approaches for software development	Literature review
Shawky and Abd-El-Hafiz (2014)	Analyze how agile methods affect software quality parameters.	The effects of Agile development on software quality parameters have received little empirical study.	Case study and survey
Sindhgatta et al. (2010)	Examine Agile software development's use of evolution	Software evolution in Agile development has received limited interest.	Case study
Chapin (2004)	Examine the role of Agile methodologies in the evolution of software.	There is little study on how Agile approaches affect software evolution.	Literature review

III. METHODOLOGY

Investigating the effects of Agile techniques on software evolution was the goal of this study. A mixed-methods approach was utilized to accomplish this, including an interview with a software engineer from the software company Spades Software Company and a systematic examination of the literature of 21 research publications on the subject. In order to learn more about the firm team's experiences with Agile methodologies and how they have affected the evolution of software, a questionnaire was also given to them.

- The goal of the systematic literature study was to locate and evaluate the current research on Agile techniques and how they affect the evolution of software for startups. The actions listed below were taken:
- The research question and the search plan were established: "What is the impact of Agile practices on software evolution?" was the research topic. Multiple databases, including IEEE Explorer, ACM Digital Library, and ScienceDirect, were incorporated into a thorough search approach.
- Screening and selection: Using inclusion and exclusion criteria, the search results were screened and chosen. Research papers focusing on Agile techniques and their influence on software evolution met the inclusion requirements. Papers that did not address Agile principles or software evolution met the exclusion criteria.
- Data collection and analysis: Information about the study design, sample size, Agile practices applied, and effects on software evolution were taken from the chosen papers.

In order to understand how Agile techniques have impacted software evolution, the data was evaluated.

- To learn more about their experiences with Agile techniques and how they affect software evolution, a software engineer from Spades Software Company was interviewed. The interview covered the following subjects and was semi-structured:
- The study's goal and the questions that will be asked were explained at the outset of the interview.
- Background: Questions about the interviewee's experience with Agile methods included which practices they had employed and for how long.
- Impact on software evolution: The respondent was questioned about the effects of Agile techniques on the development of software, including enhancements to software quality, project efficiency, and client satisfaction.
- Challenges: The interviewee was questioned about any difficulties they had in putting Agile practices into practice and how they overcame them.
- Future of Agile Methodologies: The interviewee was questioned about how they viewed Agile Methodologies developing in the future.
- To learn more about the company's team's experiences with Agile methodologies and how they have affected the evolution of software, a questionnaire was given to them. The questionnaire included the following topics and had both closed-ended and open-ended questions:
- Experience with Agile practices: In the questionnaire, it was questioned whether the team has any experience with Agile practices, as well as which ones and for how long.
- Impact on software evolution: The poll sought information on how Agile approaches have improved software quality, project effectiveness, and customer happiness.
- Challenges: The questionnaire asked about any obstacles the team had to overcome in order to apply Agile techniques.
- Future of Agile Practices: In the questionnaire, it was questioned how the team viewed Agile Practices developing in the future.

To find patterns and trends in the effects of Agile techniques on software evolution, thematic analysis was used to examine the information gathered from the systematic literature review, interview, and questionnaire. Based on the study questions, the data was arranged into themes and then examined to find recurring trends and patterns. The greatest ethical standards were followed during the study's execution. All participants gave their informed consent after having their privacy and confidentiality respected. To safeguard the participants' privacy, the data was anonymously and securely kept, and the results were provided in aggregate.

V. RESULTS

A systematic literature review, an interview with a software engineer from Spades Software Company, and a survey given to the staff of the organization were all part of the mixed-methods methodology used in this study to determine the effect of Agile practices on software evolution. Below is a presentation of the study's findings.

- Results of the Systematic Literature Review: The systematic review of the literature found that Agile techniques have a beneficial influence on the evolution of software. Several studies have found improvements in software quality, project performance, and customer satisfaction. Scrum, Extreme Programming (XP), and Kanban were the most popular Agile techniques. The size of the team, the complexity of the project, and the corporate culture were all found to have an impact on the effectiveness of Agile approaches(Waseem and Ikram, 2016). The literature review concluded that Agile techniques are successful in encouraging software evolution(Safwan, 2013).
- Findings from the Interview: According to the interview with the software engineer from Spades Software Company, Agile principles have significantly influenced how software evolves within the company. Agile approaches, according to the interviewee, have improved software quality, project effectiveness, and customer happiness. The respondent specifically emphasized the advantages of utilizing Scrum as an Agile technique, including better project management, enhanced transparency, and improved teamwork. The interviewee also spoke about some of the issues the organization had had to adapt to the new approach and pushback from team members in applying Agile principles. However, the interviewee claimed that generally, employing Agile methods was better than not using them.
- Results of the Questionnaire: The firm team's experience with Agile techniques was favorable, according to the majority of responders to the questionnaire, which was given to the team. They also reported increases in software quality, project performance, and customer satisfaction. The team claimed that the two most popular Agile approaches, Scrum and XP, had improved project management, enhanced communication with stakeholders, and boosted team member collaboration. In addition, the team reported having trouble adjusting to the new methodology and encountering pushback from team members when implementing Agile principles. The majority of respondents, however, claimed that utilizing

Agile approaches was preferable to other methods because of its advantages.

- Analysis of the Results: The study's findings offer convincing proof that Agile methodologies help to speed up the development of software. Both the comprehensive literature analysis and the interview with the software engineer from Spades Software Company demonstrated that Agile approaches, particularly Scrum, are successful in enhancing software quality, project performance, and client satisfaction. These conclusions were corroborated by the questionnaire given to the company team, with the vast majority of responders attesting to improvements in these areas. The difficulties the team had implementing Agile methods were also consistent with the literature, demonstrating that difficulties adjusting to the new approach and opposition from team members are typical obstacles to successful Agile adoption.

Overall, the study's findings imply that fostering software evolution through Agile techniques is a successful strategy. The study offers insightful information regarding the influence of Agile techniques on the evolution of software, which can guide future study and practice in this field. The report also emphasizes how crucial it is to deal with obstacles to adopting Agile, such as team member reluctance and issues adjusting to the new methodology. Organizations can increase their use of Agile techniques' advantages and encourage software evolution by solving these issues.

VI.DISCUSSION

According to the findings of the literature reviews and survey replies, it was identified that agile approaches have a considerable impact on software evolution. In order to enhance their software development processes and keep up with quickly shifting consumer and market expectations, a number of software development businesses have adopted agile approaches in recent years. In this paper, the Impact of using Agile Practices in Startups for Software evolution is discussed thoroughly. It will be beneficial to use Agile Practices in Startups because it has the capacity to continuously enhance and adapt to changing consumer desires and expectations. Reviews of the literature show that using agile approaches promotes both higher software quality and more effective and quick software evolution. Software development teams can find and fix problems early in the process, leading to a higher-quality end result, by employing specific agile methodologies, such as iterative development, continuous integration and testing, and regular client input. The example survey responses give more information about how agile techniques affect software evolution. (Chapin, 2004)

Since many businesses have been utilizing agile approaches for some time, for instance, the efficiency of their software evolution has significantly improved. These businesses have also placed an emphasis on agile methods like daily stand-up meetings, sprint planning, and retrospective meetings that have helped their software development processes. The effectiveness of software evolution has increased thanks to these strategies' improved code quality, faster delivery times, and more client satisfaction. It is not without difficulties, though, to use agile approaches for the growth of software. A range of corporate challenges, including reluctance to change, difficulty managing team dynamics, and the need for continual training and education, may have been highlighted by the sample questionnaire replies. To enable the effective application of agile concepts in Start Ups and their favorable effects on software evolution, these issues must be handled with caution. Analyzing how agile techniques assist the software evolution in Start Ups is crucial address the challenges and overcome these challenges. Businesses utilize metrics like cycle time, defect density, and customer happiness to assess performance. (Al-Saqqa et al., 2020).Due to speedier delivery dates and more client satisfaction, using agile techniques might occasionally result in better software evolution performance when compared to traditional methods. Overall, reviews of the available literature and typical responses to questionnaires show that agile approaches have a significant influence on the development of software in Startups. Software development teams can increase the effectiveness, responsiveness, and quality of their software by using agile methodologies. To get the intended results, these procedures must be carefully managed and adjusted to the unique requirements and difficulties of each software development team. Given that agile processes have a considerable impact on how software is developed, it is obvious that employing agile approaches in Startups may enhance software quality, speed up delivery, and raise customer happiness.

VII.CONCLUSION

In conclusion, this study has provided insights into the impact of Agile practices on software evolution. The findings suggest that Agile practices have a positive impact on software quality, project performance, and customer satisfaction. The study used a mixed-methods approach that included a systematic literature review, an interview with a software engineer from Spades Software Company, and a questionnaire administered to the company team. The study has several achievements, including contributing to the body of knowledge on Agile practices and their impact on software evolution, highlighting the benefits and challenges of using Agile practices and providing practical insights for organizations considering adopting Agile methodologies.

Based on the results of this study, it is recommended that organizations adopt Agile methodologies to promote software evolution. However, it is important to address the challenges associated with Agile adoption, such as resistance from team members and difficulties in adapting to the new methodology, in order to fully leverage the benefits of Agile practices. Future research in this area could investigate the impact of specific Agile practices on software evolution in different organizational contexts, as well as the factors that influence the successful adoption of Agile methodologies. Overall, this study has contributed to the understanding of Agile practices and their impact on software evolution, and provides a valuable resource for organizations seeking to improve their software development processes.

VIII.LIMITATIONS

Despite evidence that agile approaches are helpful for software development and evolution of Startups, the study's limitations suggest that these advantages may not always be evident or constant across all projects and contexts. Before selecting whether and how to implement agile approaches for software development and evolution, businesses must carefully evaluate their unique requirements and circumstances. It's also important to consider the varying constraints, limited resources, and skills of the team members before applying Agile Practices in a small Startup company.

One drawback of the agile technique is the absence of a common definition. It can be challenging to compare and generalize study results since various organizations and practitioners may utilize and understand agile methodologies in different ways. The possibility of bias in the survey's self-reported results is another issue. Participants could exaggerate the value of agile methodologies or downplay any implementation challenges. The study's sample size could possibly be too small, which would restrict how broadly the results could be applied to other contexts or to larger companies. Finally, it's possible that the study did not take into account how outside variables, such as changing technology or consumer needs, impact the efficiency of software evolution. Despite these limitations, the study offers insightful data for businesses considering adopting agile development and evolution approaches.

IX.FURTHER RESEARCH

Even while agile methodologies have already had a significant impact on software development, there are still many opportunities for academics to further explore their potential. Future research may focus on how agile approaches are evolving, how they work with different development methodologies, and how they may be used in industries other than software development. Researchers may also examine how evolving technology might be incorporated into agile procedures, as well as how agile practices affect team dynamics and company culture. Since agile approaches may improve software development processes and improve the quality of software products, they have a significant potential impact on how software evolves. Additional research may turn up novel ideas for enhancing collaboration, software development, and technological landscape adaptation. Agile methodologies have great promise for the future of software development, but this potential can only be fully realized via further research and development.

X. REFERENCES

Al-Saqqa, S., Sawalha, S., Abdel-Nabi, H., 2020. Agile Software Development: Methodologies and Trends. International Journal of Interactive Mobile Technologies (iJIM) 14, 246. https://doi.org/10.3991/ijim.v14i11.13269

Chapin, N., 2004. Agile methods' contributions in software evolution, in: 20th IEEE International Conference on Software Maintenance, 2004. Proceedings. Presented at the 20th IEEE International Conference on Software Maintenance, 2004. Proceedings., pp. 522-. https://doi.org/10.1109/ICSM.2004.1357864

Diebold, P., Theobald, S., Wahl, J., Rausch, Y., 2019. Stepwise transition to agile: From three agile practices to Kanban adaptation. Journal of Software: Evolution and Process 31, e2167. https://doi.org/10.1002/smr.2167

Ghimire, D., Charters, S., 2022. The Impact of Agile Development Practices on Project Outcomes. Software 1, 265–275. https://doi.org/10.3390/software1030012

Hayat, F., Rehman, A.U., Arif, K.S., Wahab, K., Abbas, M., 2019. The Influence of Agile Methodology (Scrum) on Software Project Management. 2019 20th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD) 145–149. https://doi.org/10.1109/SNPD.2019.8935813

Honoré, H., Anquetil, N., Ducasse, S., Djareddir, F., Sudich, J., 2021. Report From the Trenches: A Case Study In Modernizing Software Development Practices.

Kruchten, P., 2013. Contextualizing Agile Software Development, Journal of Software Maintenance and Evolution: Research and Practice. https://doi.org/10.1002/smr.572

Lai, F., Hussain, S., Shad, K., Azeem Akbar, M., 2020. Towards Successful Agile Development Process in Software Outsourcing Environment: A Systematic Literature Review. International Journal of Business Innovation and Research 1, 1. https://doi.org/10.1504/IJBIR.2020.10022210

Nierstrasz, O., 2004. Software Evolution as the Key to Productivity, in: Wirsing, M., Knapp, A., Balsamo, S. (Eds.), Radical Innovations of Software and Systems Engineering in the Future, Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, pp. 274–282. https://doi.org/10.1007/978-3-540-24626-8_19

Safwan, M.M.M., 2013. An Empirical Study of Agile Software Development Methodologies: A Sri Lankan Perspective.

Shawky, D.M., Abd-El-Hafiz, S.K., 2014. The impact of agile approaches on software quality attributes an empirical study, in: 2014 9th International Conference on Software Paradigm Trends (ICSOFT-PT). Presented at the 2014 9th International Conference on Software Paradigm Trends (ICSOFT-PT), pp. 49–57.

Sindhgatta, R., Narendra, N., Sengupta, B., 2010. Software evolution in agile development: a case study. https://doi.org/10.1145/1869542.1869560

Waseem, M., Ikram, N., 2016. Architecting Activities Evolution and Emergence in Agile Software Development: An Empirical Investigation. https://doi.org/10.1007/978-3-319-33515-5_35

Weichbroth, P., 2022. A Case Study on Implementing Agile Techniques and Practices: Rationale, Benefits, Barriers and Business Implications for Hardware Development. Applied Sciences 12, 8457. https://doi.org/10.3390/app12178457

Younas, M., Jawawi, D., Mahmood, A.K., Ahmad, M., Sarwar, M., Idris, M., 2019. Agile Software Development Using Cloud Computing: A Case Study. IEEE Access PP, 1–1. https://doi.org/10.1109/ACCESS.2019.2962257

XI.ACNOWLEDGEMENT

We would like to express our sincere gratitude to Dr. BNS Lankasena, who supported this research with useful guidance, insightful comments, and encouragement to complete this study.

XII.AUTHOR BIOGRAPHIES

NTD Dharmasiri is a fourth-year Software Engineering Undergraduate of the Faculty of Computing, General Sir John Kotelawala Defence University, Ratmalana.

JP Kodithuwakku is a fourth-year Software Engineering Undergraduate of the Faculty of Computing, General Sir John Kotelawala Defence University, Ratmalana.

PKBTD Pallawala is a fourth-year Software Engineering Undergraduate of the Faculty of Computing, General Sir John Kotelawala Defence University, Ratmalana.

BNS Lankasena is a Senior Lecturer of University of Sri Jayewardenepura, Colombo.