

COMPARISON OF AGILE AND DEVOPS METHODOLOGIES: ANALYSIS OF EFFICIENCY, FLEXIBILITY, AND APPLICATION IN SOFTWARE DEVELOPMENT

Vladimir Radovanović¹, Olja Krčadinac², Jasmina Perišić³, Marina Milovanović⁴,
Željko Stanković⁵

¹*Elektromreža Srbije a.d., Belgrade, Serbia, Kneza Milosa 11, 1112vlada1995@gmail.com, 0009-0001-8363-5903*

²*"Union – Nikola Tesla" University, Faculty of Informatics and Computer science, Belgrade, Serbia, Cara Dusana 42-46, okrcadinac@unionnikolatesla.edu.rs, 0000-0002-6299-371X*

³*"Union – Nikola Tesla" University, Faculty of Informatics and Computer science, Belgrade, Serbia, Cara Dusana 42-46, jperisic@unionnikolatesla.edu.rs, 0000-0001-7412-7870*

⁴*"Union – Nikola Tesla" University, Faculty of Entrepreneurial Business and Real Estate Management, Belgrade, Serbia, Cara Dusana 42-46, mmilovanovic@unionnikolatesla.edu.rs, 0000-0002-3778-0026*

⁵*Pan-European APEIRON University, Banja Luka, B&H, Vojvode Pere Krece 13, stanz@medianis.net, 0000-0002-9893-9088*

Review paper

<https://doi.org/10.7251/JIT2401078R>

UDC: 004.421.2:519.676

Abstract: This paper provides a concise overview of Agile and DevOps methodologies in software engineering. It aims to introduce readers to the fundamental principles of Agile and DevOps, accompanied by brief descriptions and practical examples. The advantages and disadvantages of each methodology are discussed, followed by a comparative analysis highlighting key differences. Understanding these methodologies is crucial in today's IT landscape, as they are commonly employed in various organizations, impacting project management, team collaboration, and product delivery. This paper serves as a valuable resource for individuals seeking a basic understanding of Agile and DevOps methodologies in software engineering.

Keywords: Agile methodology, DevOps methodology, software engineering, comparative analysis, software development

INTRODUCTION

In the realm of software development, Agile and DevOps methodologies stand out as transformative approaches that have revolutionized project management and delivery processes. Agile methodologies, characterized by iterative and collaborative development cycles, have become synonymous with adaptability and responsiveness to changing requirements. On the other hand, DevOps methodologies emphasize the integration of development and operations teams, promoting automation, continuous delivery, and streamlined workflows.

In this study, we delve into the comparison of Agile and DevOps methodologies, focusing on their efficiency, flexibility, and application in software development. Specifically, we hone in on the widely recognized Agile methodologies of SCRUM and Extreme Programming (XP), examining their core principles,

implementation strategies, and real-world impacts. Seventeen researchers convened during a conference to collaboratively formulate a document recognized as the Agile Manifesto (Pacagnella Junior & Da Silva, 2023). The Agile Manifesto as a document for the software development process relies on 4 values and 12 principles. The values are: Individuals and Interaction, Software that Works (Applicable Software), Collaboration with Customers, and Response to Change (Cobb, 2023).

The significance of this comparison lies in the nuanced understanding it offers to organizations seeking to optimize their software development processes. By evaluating the strengths and limitations of Agile and DevOps methodologies, we aim to provide valuable insights that can inform decision-making, enhance project outcomes, and drive innovation in the software engineering domain. There are several

types of agile methodologies, the most important of which are SCRUM, Extreme Programming (XP), Kanban, Feature driven development, Lean software development and others (Tetteh, 2024).

Through a combination of qualitative and quantitative analyses, we dissect the effectiveness of these methodologies, considering factors such as team collaboration, project adaptability, time-to-market, and overall project success rates. Our findings not only contribute to the academic discourse but also offer practical guidance for practitioners and industry leaders navigating the complexities of modern software development.

EVALUATION METHODS FOR AGILE AND DEVOPS Comparison

For the analysis of data collected in this study, we employed a combination of qualitative and quantitative analytical methods. These methods were chosen to provide a comprehensive understanding of the efficiency and flexibility of Agile and DevOps methodologies in software development. One of the primary analytical methods utilized was qualitative text analysis. This approach involved systematically reviewing and coding textual data obtained from relevant literature. Through thematic analysis, we identified key patterns, themes, and insights related to the implementation and impact of Agile and DevOps methodologies. To facilitate the analysis process, we utilized various tools and software packages. For the analysis of qualitative data in our research, we utilized Python, an open-source programming language that offers various modules and libraries for text processing. By leveraging Python, we had the flexibility to tailor analytical processes to our specific research needs, including text preprocessing, pattern identification, and insights generation from our data.

Almeida et al. highlighted the benefits of combining Agile and DevOps methodologies for handling complex customer requirements. Through qualitative analysis of twelve case studies, it identifies twelve benefits, including process automation, improved team communication, and reduced time to market. The research concludes that while Agile and DevOps have distinct goals, their combined adoption offers significant advantages for organizations. This study is crucial for comparing Agile and DevOps technologies

as it systematically analyzes the benefits of their combined adoption, providing insights into their impact on organizational efficiency and competitiveness (Almeida, Simões, & Lopes, Exploring the benefits of combining devops and agile, 2022).

Masud et al. propose integrating DevOps principles into Agile to create a hybrid DevOps Enabled Agile methodology for software development, addressing collaboration gaps and skill requirements. A pilot project demonstrates how this integration improves speed and quality in software development. This is important for comparing Agile and DevOps technologies as it shows their combined benefits in overcoming implementation limitations and enhancing development processes (Al Masud & et al., 2022). Researchers explore how small software companies with limited resources can adopt agile and DevOps frameworks more effectively. Three best practices based on DevOps were identified, which can be integrated within SCRUM, particularly beneficial for small companies. Through a case study in three small companies, these practices showed improvements in customer support and deployment success, highlighting the importance of integrating DevOps approaches into agile methodologies to enhance software development processes and achieve better product quality (Almeida, Simões, & Lopes, Exploring the benefits of combining devops and agile, 2022).

Jayakody in his research (Jayakody & Wijayanayake, 2021) focuses on analyzing the utilization of the DevOps concept in IT projects, with a specific emphasis on challenges and strategies for success. In relation to the paper describing DevOps as a key component for enhancing performance in software development and operations (Langerman & Leung, 2023), the updated information in our research indicates a continuous increase in the significance of DevOps methodologies within the industry. Recent developments in the Agile and DevOps domains affirm the importance of integrating operations and software development to ensure efficiency and adaptability of organizations. Studies have demonstrated that organizations successfully implementing the DevOps approach have accelerated software delivery and ensured continuous delivery of value to customers. Additionally, evidence from the literature suggests that recent advancements in tools and practices within DevOps methodologies have supported the concepts

described in the cited paper, highlighting their role in enhancing organizational performance. For instance, the emergence of new tools for process automation and infrastructure management has enabled teams to more effectively manage software development and delivery, providing further support for the idea of DevOps as a means to improve performance in software development and operations. By analyzing the literature, we noticed that improved internal communication among teams using Agile and DevOps methodologies often results in greater efficiency in software development. This theme supported previous findings in the literature that highlighted that good communication among team members can significantly improve productivity and reduce the risk of errors in software development. The conclusion of our research confirms that the effectiveness of Agile and DevOps methodologies can be directly related to improved communication between teams.

RESULTS

The results obtained from the analytical methods were interpreted in conjunction with theoretical frameworks and existing literature on Agile and DevOps methodologies. To ensure the validity and reliability of our findings, we employed strategies such as triangulation of data sources, peer review, and validation through expert feedback.

In the realm of software development methodologies, Agile and DevOps stand out as two prominent approaches that have revolutionized the industry (Maroukian, 2021). Agile methodologies prioritize iterative development, customer collaboration, and responding to change over rigid planning and documentation (Heeager & Nielsen, 2020). This approach has proven highly effective in delivering value to customers through incremental improvements and rapid feedback loops.

On the other hand, DevOps focuses on collaboration between development and operations teams, emphasizing automation and continuous integration (Srivastav, Allam, & Mustyala, 2023). DevOps aims to streamline the software delivery process, reduce manual errors, and accelerate time-to-market by breaking down silos and fostering a culture of shared responsibility.

Agile excels in fostering agility, adaptability, and customer-centricity, while DevOps enhances efficien-

cy, collaboration, and automation across the entire software development lifecycle.

The table below presents a comparison between Agile and DevOps methodologies across various parameters crucial for software development. The parameters were selected through this research based on their significance in understanding the strengths and characteristics of each methodology.

Table 1. Comparison of Agile and DevOps Methodologies: Key Parameters

Parameter	Agile	DevOps
Customer Focus	High	High
Iterations	Regular	Continuous
Process Automation	Partial	High
Team Collaboration	Centralized	Integrated
Delivery Speed	Fast	Very Fast
Software Quality	High	Very High
Development-Operations Integration	Partial	Full
Flexibility in Changes	High	Very High
Security Integration	Low	High
Scalability	Moderate	High
Resource Efficiency	Moderate	High

Examining the parameters in the table 1 comparing Agile and DevOps methodologies in software development, we observe significant differences and similarities between these two approaches. For instance, while both methodologies are highly focused on user needs and software quality, DevOps methodology stands out for its capability of continuous software delivery and high degree of process automation. On the other hand, Agile methodology emphasizes regular iterations and centralized team collaboration. Although there is partial integration between development and operations in Agile methodology, DevOps aims for full integration to facilitate continuous software delivery and maintenance.

To quantify the terms *fast* and *high*, we established clear criteria reflecting specific aspects of delivery speed and software quality. For instance, delivery speed was defined as the average time required to deliver a software product or the frequency of software releases within a given timeframe. Fast defined as an average release cycle of 1-2 weeks, while *Very Fast* could defined as an average release cycle of less than 1 week. As for software quality, we quantified

it through various metrics, including the number of bugs or defects identified in the software, customer satisfaction ratings, or adherence to predefined quality standards.

Through this analysis, we conclude that both methodologies have their strengths and weaknesses, but they can complement each other when used together. Integrating DevOps principles within Agile methodology can contribute to more efficient team collaboration, faster software delivery, and improved product quality. Our study suggests that organizations should consider a hybrid approach combining the best aspects of Agile and DevOps methodologies to achieve optimal results in software development. This may involve implementing DevOps principles such as process automation, continuous testing, and security integration within the Agile development framework to support rapid and high-quality software development.

The parameters included in the table were carefully selected based on their relevance to the comparison between Agile and DevOps methodologies in software development. Each parameter was chosen to highlight key characteristics of both methodologies and facilitate a clear comparison between them. For instance, parameters such as "Customer Focus" and "Software Quality" were selected to emphasize the importance of meeting customer needs and ensuring high-quality software products. Conversely, parameters like "Process Automation" and "Delivery Speed" were chosen to underscore the differing approaches between Agile and DevOps methodologies.

The selection of these parameters was guided by the research focus and objectives, aiming to provide meaningful insights that support the study's conclusions and contribute to a comprehensive understanding of the differences between Agile and DevOps methodologies in the context of software development.

DISCUSSION

Our comparative analysis between Agile and DevOps methodologies reveals significant differences and similarities across key parameters. While both methodologies prioritize customer focus, iteration, and software quality, DevOps stands out with its emphasis on process automation, delivery speed, and integration between development and operations

teams. Agile, on the other hand, excels in team collaboration and iterative improvements. The findings have important implications for software development practices. Organizations using Agile, DevOps, or a combination of both can benefit from understanding the strengths and challenges of each methodology. Integrating DevOps principles into Agile methodology can lead to enhanced team efficiency and streamlined development processes.

Our research offers practical insights into the implementation of Agile and DevOps methodologies. For example, integrating automation tools and continuous integration practices can improve delivery speed and quality. Furthermore, fostering a culture of collaboration and continuous improvement is essential for success in both methodologies.

While our study provides valuable insights, it is important to acknowledge its limitations. Future research could explore the long-term impact of integrating Agile and DevOps, as well as delve deeper into specific industry contexts or organizational sizes to uncover more nuanced findings. Based on our results, we recommend organizations consider a hybrid approach that combines the strengths of Agile and DevOps. This could involve adopting DevOps principles such as automation, continuous integration, and collaboration within Agile development cycles. Additionally, investing in training and tools to support this integration can lead to improved software delivery and customer satisfaction (Faustino, Adriano, Amaro, Pereira, & da Silva, 2022). Case studies are valuable for gaining deeper insights into specific implementations of Agile and DevOps methodologies in real-world settings. However, it is important to acknowledge that the limited number of case studies may restrict the generalizability of the results to broader populations of organizations. Our study relied on a restricted number of case studies, which may impact the broader applicability across different industries or project sizes.

In light of these limitations, we recommend further research that includes a larger number of case studies or encompasses different industries and project sizes. A larger sample of case studies would provide a broader perspective and contribute to a better understanding of the effects of Agile and DevOps methodologies in various contexts. For instance, future research could explore how industry-specific

characteristics or project sizes influence the success of implementing these methodologies, providing additional insights for organizations planning their implementation.

In the context of growing cybersecurity threats, it is important to consider how Agile and DevOps methodologies can adapt to enhance security and respond to threats (Friman, 2024). This is particularly crucial given the increasing digitization of businesses and the sophistication of attacks that organizations may face. One key way Agile and DevOps methodologies can adapt to strengthen security is by integrating security testing in the early stages of development. Rather than treating security as an afterthought, which can lead to vulnerabilities being discovered late in the development cycle when remediation is more costly and complex, security testing can be integrated into all stages of the development process. This allows teams to identify and address security flaws as early as possible, minimizing the risk of serious security incidents. Additionally, adopting principles of continuous security can be crucial for enhancing security in Agile and DevOps environments. This involves continuously monitoring, assessing, and improving security practices and policies throughout the development cycle. This ensures that security requirements are continually taken into account and implemented, rather than treated as a one-time activity. Furthermore, implementing DevSecOps practices can be an effective way to integrate security considerations throughout the entire development cycle. DevSecOps promotes collaboration between software development, operations, and security teams, enabling them to work together to identify, address, and prevent security threats at all stages of software development and delivery.

Through these approaches, Agile and DevOps methodologies can be effectively adapted to ensure stronger security and resilience against growing cybersecurity threats, providing organizations with the ability to develop and deliver software that is secure and resilient to security challenges.

CONCLUSION

This research provides a thorough examination and comparison of Agile and DevOps methodologies in software development. Our findings indicate that Agile methodologies excel in iterative improvements,

team collaboration, and customer focus, while DevOps methodologies stand out with their emphasis on automation, continuous integration, and delivery speed. Integrating DevOps principles into Agile frameworks can enhance the efficiency and effectiveness of software development teams, leading to faster delivery times and improved product quality.

Our study highlights the importance of considering the specific needs and contexts of organizations when choosing between Agile and DevOps methodologies. A hybrid approach that combines the best practices of both methodologies can offer significant benefits, including increased flexibility, improved collaboration between teams, and enhanced software development processes.

This research contributes to the existing body of knowledge on Agile and DevOps methodologies and provides practical recommendations for organizations looking to optimize their software development processes. By embracing a holistic approach that leverages the strengths of both Agile and DevOps, organizations can achieve greater success in delivering high-quality software products that meet the evolving needs of their customers.

REFERENCES

- [1] Al Masud, S., & et al. (2022). DevOps Enabled Agile: Combining Agile and DevOps Methodologies for Software Development. *International Journal of Advanced Computer Science and Applications*, 13(11).
- [2] Almeida, F., Simões, J., & Lopes, S. (2022). Exploring the benefits of combining devops and agile. *Future Internet* 14.2, 63.
- [3] Almeida, F., Simões, J., & Lopes, S. (2022). Exploring the benefits of combining devops and agile. *Future Internet*, 14(2), 63.
- [4] Cobb, C. (2023). *The project manager's guide to mastering Agile: Principles and practices for an adaptive approach*. John Wiley & Sons.
- [5] Faustino, J., Adriano, D., Amaro, R., Pereira, R., & da Silva, M. (2022). DevOps benefits: A systematic literature review. *Software: Practice and Experience*, 52(9), 1905-1926.
- [6] Friman, O. (2024). *Agile and DevSecOps Oriented Vulnerability Detection and Mitigation on Public Cloud*. Aalto University, School of Science.
- [7] Heeger, L., & Nielsen, P. (2020). Meshing agile and plan-driven development in safety-critical software: a case study. *Empirical Software Engineering*, 25(2), 1035-1062.
- [8] Jayakody, J., & Wijayanayake, W. (2021). Challenges for adopting DevOps in information technology projects. In *2021 International Research Conference on Smart Computing and Systems Engineering (SCSE)*, (Vol. 4, pp. 203-210).
- [9] Langerman, J., & Leung, W. (2023). The effect of outsourcing and insourcing on Agile and DevOps. *Journal of Information Technology Teaching Cases*.

- [10] Maroukian, K. (2021). Towards Practice and Principle Adoption through Continuous DevOps Leadership. *J. Softw.*, 16(1), 1-13.
- [11] Pacagnella Junior, A., & Da Silva, V. (2023). 20 Years of the Agile Manifesto: A Literature Review on Agile Project Management. *Management and Production Engineering Review*.
- [12] Srivastav, S., Allam, K., & Mustyala, A. (2023). Software Automation Enhancement through the Implementation of DevOps. *International Journal of Research Publication and Reviews*, 4(6), 2050-2054.
- [13] Tetteh, S. (2024). Empirical Study of Agile Software Development Methodologies: A Comparative Analysis. *Asian Journal of Research in Computer Science*, 17(5), 30-42.

Received: April 10, 2024

Accepted: April 20, 2024

ABOUT THE AUTHORS



Vladimir Radovanović is a Lead Information Security Engineer at Joint Stock Company Elektromreža Srbije. He earned his bachelor degree in Computer Science from University Union Nikola Tesla Beograd. He is currently enrolled in a Master's program in Faculty of Informatics and Computer Science at "Union - Nikola Tesla" University. Before Joint Stock Company Elektromreža Srbije, he was teacher in Gymnasium „Uroš Predić“ Pančevo, where he taught IT subjects. His research focuses on cyber security, information security and teaching. He is author of few research papers.



Olja Krčadinac (Latinović, maiden name) is assistant professor at "Union - Nikola Tesla" University - Faculty of Informatics and Computer Science. She earned her Ph.D. in biometric field from University of Belgrade - Faculty of Organizational science, where she conducted groundbreaking research on speaker recognition. In addition to her teaching responsibilities, Olja has authored numerous impactful publications in peer-reviewed journals, contributing valuable insights to the scientific community. Her research focuses on biometric, sensors, IoT and AI, addressing critical issues in AI and making significant contributions to the academic community.



Jasmina Perišić graduated and mastered at the Faculty of Organizational Sciences of the University of Belgrade. She defended her doctoral dissertation in the elective field of Electronic Business at the Faculty of Organizational Sciences of the University of Belgrade. Since 2013, she has been employed as an associate professor at the Union "Nikola Tesla" University in Belgrade (Faculty of Informatics and Computing, Faculty of International Politics and Security, Faculty of Economics and Finance, Faculty of Entrepreneurship, Business and Real Estate Management). She has published ten papers in

journals of international importance on the SCI list, twelve papers in journals of international and national importance that are not on the SCI list, over seventy papers at international and domestic conferences, one monograph of national importance and six papers in thematic collections of national importance.



Marina Milovanović graduated (2000) and mastered (2005) at the Faculty of Mathematics, University of Belgrade. She received her doctorate at the Faculty of Entrepreneurial Business of Union University (2008) and at the Faculty of Science and Mathematics at the University of Kragujevac (2014) as a Doctor of Science (Economic and Mathematical Sciences). She works as a full-time professor at "Union - Nikola Tesla" University. Currently, her research is focused on performance of the financial mathematics, statistics, teaching mathematics and informatics. Throughout her career, she has authored numerous impactful publications in peer-reviewed journals, contributing valuable insights to the scientific community, projects etc.



Željko Stanković received his higher education in Cleveland, Ohio, USA, where he graduated in 1981. The topic of the thesis was "Reversible sound in halls". He defended his master's thesis ("Learning control system (LMS) based on ADL SCORM specifications") in 2006 at the University of Novi Sad, Faculty of Science, Department of Informatics. He defended his doctoral dissertation (Laser perception of defined objects and encapsulation of control and logic elements for an autonomous robotic teaching tool) at Singidunum University, Belgrade, in 2010. He has been programming since 1984, creating programs for his first Commodore 64 computer. He works as a full-time professor at "Union - Nikola Tesla" University. Robotics and bioengineering have been a field of work and interest for many years. He is the holder of the patent right for the teaching tool CD ROBI.

FOR CITATION

Vladimir Radovanović, Olja Krčadinac, Jasmina Perišić, Marina Milovanović, Željko Stanković, Comparison of Agile and Devops Methodologies: Analysis of Efficiency, Flexibility, and Application in Software Development, *JITA – Journal of Information Technology and Applications*, Banja Luka, Pan-Europien University APEIRON, Banja Luka, Republika Srpska, Bosna i Hercegovina, JITA 14(2024)1:78-83, (UDC: 004.421.2:519.676), (DOI: 10.7251/JIT2401078R, Volume 14, Number 1, Banja Luka, June (1-88), ISSN 2232-9625 (print), ISSN 2233-0194 (online), UDC 004