iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** XI **Month of publication:** November 2023

DOI: https://doi.org/10.22214/ijraset.2023.56986

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue XI Nov 2023- Available at www.ijraset.com

Unravelling DevOps Agile Methodologies: A Comprehensive Review of Recent Research

Duncan Nyale[1], Shem Mbandu Angolo[2]
[1]*School of Computing and Mathematics, Co-operative University of Kenya, Nairobi, Kenya*
[2]*School of Computing and Mathematics, Co-operative University of Kenya, Nairobi, Kenya*

Abstract: ***This study report offers a thorough analysis of current studies on DevOps Agile techniques. The study summarizes and dissects the major discoveries made in peer-reviewed literature, conference proceedings, and other academic publications. The article examines the most recent DevOps Agile practice trends and advances, as well as how they affect software development, testing, deployment, and operations. The analysis emphasizes the advantages of applying DevOps Agile approaches, including enhanced collaboration, faster time to market, and higher-quality software products. The paper also examines the challenges and potential limitations of DevOps Agile adoption, such as cultural resistance, lack of skills, and integration issues. Finally, the study concludes with recommendations for practical implications for software development teams and organizations seeking to implement DevOps Agile methodologies. The review provides valuable insights and guidance for researchers, practitioners, and decision-makers in the field of software engineering and information technology.***

Keywords**: DevOps, Agile Development, Common Principles, Continuous Integration, Continuous Deployment, Change Failure Rate, Mean Time to Recover, Lead time, Deployment Frequency, Metrics, Legacy Systems, Automation, Toolchain, Collaboration, Compliance, strategy, Infrastructure, Complementary Methodologies, Security, Error Rate***

I. INTRODUCTION

DevOps is a collection of processes that integrates software development (Dev) and IT operations (Ops) to shorten the systems development life cycle and provide features, fixes, and upgrades on a regular basis in tight alignment with business goals. With a strong emphasis on collaboration, flexibility, and customer satisfaction, the agile methodology is an iterative approach to software development that focuses on producing high-quality software fast and continuously.

One of the most well-liked methods in software development is DevOps. The other is Agile methodology. While Agile places an emphasis on flexibility and cooperation, DevOps focuses on removing barriers between development and operations teams. Many firms have now integrated these approaches to develop a DevOps Agile strategy.

Continuous integration and delivery, iterative development, and a focus on customer feedback are some of the main tenets of DevOps Agile. Organizations may establish a software development process that is more successful and efficient by integrating these ideas. The advantages of DevOps Agile techniques have recently been the subject of numerous studies and papers. For instance, a research from the DevOps Institute indicated that companies using DevOps Agile methods experience higher levels of automation, better team communication, and increased customer satisfaction [1]. According to a different Gartner research, 90% of all firms will incorporate DevOps Agile approaches into their software development process by 2023 [2].

DevOps Agile processes are supported by a wide range of tools and technology. These include cloud-based infrastructure providers like Amazon Web Services and Microsoft Azure, collaboration tools like Slack and Microsoft Teams, and continuous integration and delivery tools like Jenkins and GitLab.

Overall, as businesses seek to enhance their software development processes, DevOps Agile techniques have grown in popularity in recent years. Organizations may establish a more streamlined and efficient development process by fusing the agility and cooperation of Agile with the automation and effectiveness of DevOps.

II. METHODOLOGY

This study focused on published literature to perform a thorough evaluation of current research on DevOps Agile techniques. A thorough examination of the literature was conducted using academic databases and search engines. Only papers with empirical proof of the value of DevOps Agile approaches were eligible for inclusion.

Thematic analysis was used to study the chosen publications, which involves finding commonalities and themes in the literature. The themes discovered were then combined to create a thorough analysis of current research on DevOps Agile techniques.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue XI Nov 2023- Available at www.ijraset.com

III. LITERATURE REVIEW

A. *Key Concepts of DevOps Agile Methodologies*

DevOps Agile approaches combine the concepts of DevOps and Agile to produce a software development process that is more effective and efficient. Some of the key concepts of DevOps Agile include:

1) *Continuous Integration and Delivery (CI/CD):* A major DevOps idea called continuous integration and delivery is releasing software in short, regular releases while also continuously integrating code changes into a shared repository. As a result, efficiency is improved and risk is decreased [3].
2) *Infrastructure as Code (IaC):* DevOps also emphasizes the idea of infrastructure management as code, which involves leveraging automation technologies and code to manage infrastructure. By doing so, the chance of human error is decreased and consistency is increased. [4].
3) *Cross-functional Teams:* Agile approaches place a lot of emphasis on cross-functional teams. These teams are made up of persons with various responsibilities and skill sets who cooperate to accomplish a shared objective. This helps to promote collaboration and communication [5].
4) *Iterative Development:* Another essential idea in Agile approaches is iterative development. It entails segmenting a project into smaller, more manageable components and delivering them iteratively. This improves flexibility and lowers risk [6].
5) *Agile Manifesto:* A collection of guidelines for Agile techniques, the Agile Manifesto stresses customer happiness, teamwork, and flexibility. Since its initial introduction in 2001, it has gained a lot of traction in the software development sector [7].

Overall, DevOps Agile approaches offer a framework for businesses to accelerate, enhance, and streamline their software development process while also fostering teamwork and communication.

TABLE 1
BASIC DEVOPS ARCHITECTURE

Layer	Technology	Description
Infrastructure	Virtualization technologies (e.g., VMs, containers)	The layer responsible for creating, configuring, and managing the underlying infrastructure that applications run on.
Configuration Management	Tools like Ansible, Puppet, and Chef	The layer responsible for automating the configuration and management of infrastructure components.
Continuous Integration	Tools like Jenkins, Travis CI, and CircleCI	The layer responsible for automatically building, testing, and integrating code changes into a single shared repository.
Continuous Delivery/Deployment	Tools like Spinnaker, Harness, and Azure DevOps	The layer responsible for automating the release and deployment of software changes into production environments.
Monitoring and Logging	Tools like Prometheus, Grafana, and ELK stack	The layer responsible for collecting, analyzing, and visualizing data about the performance and behavior of applications and infrastructure.

B. *Analysis of the Relationship between DevOps and Agile Methodologies*

The approaches for developing software using Agile and DevOps have become increasingly well-known over time. In contrast to DevOps, which emphasizes cooperation and continuous delivery, Agile approaches place a strong emphasis on flexibility and adaptation. Specifically,

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

1) *Integration of Agile and DevOps:* The fusion of Agile and DevOps techniques is one of the most important developments in software development. While DevOps stresses continuous integration and delivery, Agile approaches put the emphasis on producing usable software in short iterations. These two approaches can be used to enable enterprises to produce and deliver software more quickly while maintaining quality and customer satisfaction. The DevOps Institute published a report that stated that organizations who use both Agile and DevOps jointly achieve more success in their digital transformation efforts [1].
2) *Common Principles:* The approaches of Agile and DevOps are similar in that they both emphasize teamwork, feedback, and continuous improvement. Both techniques put a strong emphasis on giving consumers value and call for close coordination between development and operations teams. Agile and DevOps are said to have a foundation of values and concepts, including cooperation, feedback, and continuous development, according to a research by the Association for Computing Machinery [8].
3) *Agile as a foundation for DevOps:* Agile approaches have been embraced by many firms as the cornerstone for their DevOps deployment. Agile approaches emphasize collaboration, adaptability, and continuous improvement, which provide DevOps a strong basis. Agile is frequently viewed as a prerequisite for DevOps since it provides the required framework for a collaborative and iterative approach to software development, according to a Gartner analysis [9].
4) *Agile and DevOps as Complementary Methodologies:* Despite having differing goals, Agile and DevOps are compatible approaches that can cooperate to produce high-quality software. DevOps stresses continuous delivery and feedback, whereas Agile techniques place emphasis on producing usable software in brief iterations. Agile and DevOps are complementary approaches that can help firms create software at a faster pace while ensuring quality and customer satisfaction, according to a McKinsey analysis [10].

Despite having differing goals, Agile and DevOps are compatible approaches that can cooperate to produce high-quality software. DevOps stresses continuous delivery and feedback, whereas Agile techniques place emphasis on producing usable software in brief iterations. Agile and DevOps are complementary approaches that can help firms create software at a faster pace while ensuring quality and customer satisfaction, according to a McKinsey analysis [10].

C. *DevOps Metrics*

DevOps metrics are measurements that are used to monitor and assess how well DevOps processes are doing within an organization. DevOps metrics are quantitative measurements that offer insight into how well the IT organization is delivering value to the business by enhancing the flow of work from development through operations and customer support, according to the book Accelerate: Building and Scaling High Performing Technology Organizations by [11].

These metrics may include indicators like change failure rate (CFR), mean time to recover (MTTR), and lead time for modifications, among others. Organizations may spot improvement opportunities and enhance their DevOps procedures to boost effectiveness and productivity by monitoring and analyzing these indicators.

TABLE 2
COMMON DEVOPS METRICS

Metric	Description	Key benefits
Lead time	The time it takes to go from code commit to release	Helps identify bottlenecks and inefficiencies in the process
Deployment frequency	The number of deployments per unit of time (e.g., per day)	Indicates how often new features and fixes are released
Mean time to recovery (MTTR)	The average time it takes to recover from a production incident	Helps measure the effectiveness of incident response and identify areas for improvement
Change failure rate (CFR)	The percentage of deployments that result in failures	Indicates the stability and reliability of the software
Availability	The percentage of time that the system is available and functional	Helps measure the reliability of the software and identify areas for improvement
Error rate	The number of errors per unit of time (e.g., per hour)	Helps identify and track the occurrence of errors over time
Customer satisfaction	The level of satisfaction among customers or end-users	Provides feedback on the quality and usability of the software
Team productivity	The amount of work completed by the team over time	Helps identify areas for improvement and optimize team performance

D. Benefits of DevOps Agile Methodologies

In recent years, DevOps and Agile techniques have grown in popularity, and they provide several advantages to businesses wishing to enhance their software development processes, including:

1) *Faster time to Market:* Organizations can provide software more quickly thanks to DevOps and Agile processes, which helps them launch their products more quickly. A DORA study found that DevOps approaches enable high-performing firms to deliver code 208 times more frequently than their low-performing competitors [11].
2) *Improved Collaboration:* Teams such as developers, operations, and quality assurance teams can work together more effectively inside an organization thanks to DevOps and Agile approaches. Better communication and quicker problem-solving result from this teamwork [12].
3) *Continuous Delivery:* Organizations can continuously release software updates and enhancements thanks to DevOps and Agile processes, helping them to remain competitive and satisfy changing client demands. [13].
4) *Increased Efficiency:* Organizations can reduce waste and streamline their operations with the use of DevOps and Agile approaches, which boosts productivity and efficiency [8].
5) *Higher Quality Software:* The software development process is prioritized by testing and quality assurance using DevOps and Agile approaches, which results in higher-quality software with fewer flaws [12].
6) *Improved Customer Satisfaction:* Organizations utilizing DevOps and Agile approaches are able to increase customer satisfaction and loyalty by providing software more quickly and with higher quality [8].

E. Challenges of DevOps Agile Methodologies

1) The organizational cultural shift needed to apply DevOps Agile techniques is one of the biggest hurdles. If there is a history of silos between organizational teams, it may be challenging to implement DevOps because it needs a collaborative approach [11].
2) The requirement for continuous integration and delivery (CI/CD), which might be complicated by the complexity of contemporary software systems, is another difficulty [14]. This necessitates a substantial investment in automation and tooling, which can be difficult for businesses with minimal funding.
3) Fast feedback loops are necessary but can be challenging to implement when working with large, complicated systems [15]. To guarantee that input is collected and taken into consideration in a timely manner, this calls for meticulous planning and coordination.
4) The management of DevOps Agile techniques' security and compliance is another [14]. DevOps can speed up the supply of software, but this can also increase the risk of security flaws and regulatory violations. To maintain security and compliance throughout the software development lifecycle, organizations must have strong processes in place.
5) Finally, firms employing DevOps and Agile approaches may encounter difficulties due to a lack of metrics and measurement. It can be challenging to monitor progress and find areas for development without the right measures. According to a study by Forrester, DevOps teams must measure performance based on business outcomes, not simply technical indicators [16]. To make sure they are meeting their objectives and providing value to their clients, organizations must set up clear metrics and measurement frameworks.

F. Organizational Limitations to Implementing DevOps Agile Methodologies

Organizations may encounter a number of restrictions when implementing DevOps Agile approaches, such as:

1) *Lack of Skills and Expertise:* Lack of qualified personnel who can oversee the new procedures and technologies is one of the biggest problems firms encounter while introducing DevOps Agile methodology. This includes people with knowledge of continuous integration and deployment, containerization, and automated testing. In order to properly apply DevOps practices, 63% of firms cited a shortage of competent workers, according to a Puppet poll [17].
2) *Resistance to Change:* When implementing a new methodology, businesses may encounter resistance to change. Employees could be hesitant to adopt new working practices and give up their old habits. A CA Technologies study found that 50% of respondents mentioned cultural resistance to change as a major obstacle to the adoption of DevOps [18].
3) *Legacy Systems and Processes:* Due to the compatibility issues between older technology and more modern DevOps tools and processes, organizations with legacy systems and processes may find it difficult to apply DevOps Agile methodology. 63% of firms stated in the DevOps Institute's Upskilling survey that legacy systems and procedures are a significant obstacle to the adoption of DevOps [1].

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

4) *Lack of Communication and Collaboration:* High levels of team cooperation and communication are necessary for DevOps Agile techniques. The DevOps process can become fragmented if teams are not collaborating well, which can cause delays, mistakes, and a lack of advancement. A DORA poll found that 37% of participants identified poor teamwork and communication as a major obstacle to the implementation of DevOps[19].
5) *Security and Compliance:* When deploying DevOps Agile Methodologies, firms must take security and compliance seriously. Due to the high level of automation and integration required by these approaches, there is an increased chance of security lapses and non-compliance. In a Puppet poll, 38% of participants identified security and compliance as a significant roadblock to DevOps adoption [17].

IV. ANALYSIS AND FINDINGS

A. *Analysis*

Delivering high-quality software to consumers quickly and effectively is a key component of both DevOps and Agile approaches. However, they employ various strategies to achieve this objective.

In contrast to Agile, which emphasizes iterative and incremental development, customer satisfaction, and teamwork, DevOps places a strong emphasis on the integration of development and operations teams as well as automation and continuous delivery and deployment.

While Agile values people and interactions, ongoing customer involvement and input, cultural shifts, and cooperation, DevOps emphasizes infrastructure as code, monitoring, and feedback loops. The table below provides examples of these elements.

TABLE 3
COMPARATIVE ANALYSIS OF DEVOPS AND AGILE METHODOLOGY

DevOps	Agile
Integration of development and operations teams	Emphasis on iterative and incremental development
Focuses on continuous delivery and deployment	Prioritizes customer satisfaction and delivering working software
Emphasizes automation and tooling	Relies on face-to-face communication and collaboration
Shifts left in the software development lifecycle	Values individuals and interactions over processes and tools
Promotes cultural changes and collaboration	Prioritizes adaptability and responding to change
Measures success by business outcomes and value delivery	Measures success by meeting user stories and sprint goals
Encourages monitoring and feedback loops	Requires a product backlog and a sprint backlog
Advocates for infrastructure as code	Encourages constant customer feedback and involvement
Supports cross-functional teams	Employs a scrum master to facilitate agile processes
Embraces continuous improvement	Allows for flexibility and changing requirements

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

TABLE 4
SUMMARY OF KEY CONCEPTS OF DEVOPS AGILE METHODOLOGIES

Key Concept	Description
Continuous Integration	The practice of continuously integrating code changes into a shared repository, which is verified by automated tests.
Continuous Delivery	The practice of continuously delivering code changes to production-like environments for testing and verification.
Continuous Deployment	The practice of automatically deploying code changes to production environments after they pass automated testing.
Infrastructure as Code	The practice of managing infrastructure using code, allowing for more efficient and consistent infrastructure deployment and management.
Agile Development	A software development methodology that emphasizes iterative development, continuous feedback, and customer collaboration.
Scrum	An Agile framework for managing and completing complex projects that involves time-boxed iterations, daily stand-up meetings, and other collaborative practices.
Kanban	An Agile framework that focuses on visualizing work, limiting work in progress, and continuous delivery.
Lean	A software development approach that emphasizes reducing waste, optimizing processes, and continuous improvement.
Collaboration	A key aspect of DevOps and Agile methodologies that emphasizes cross-functional team collaboration, communication, and sharing of knowledge and resources.
Automation	A core principle of DevOps that involves automating as many processes as possible, reducing manual work and increasing efficiency.

TABLE 5
SUMMARY OF BENEFITS AND CHALLENGES OF DEVOPS AGILE METHODOLOGIES

Benefits	Challenges
Faster time-to-market for software products	Lack of skills and expertise
Increased collaboration and communication	Resistance to change
Improved quality and reliability of software	Legacy systems and processes
More frequent and predictable releases	Lack of communication and collaboration between teams
Enhanced customer satisfaction	Security and compliance concerns
Greater agility and flexibility	Integration and automation challenges
Continuous feedback loop for improvement	Need for cultural change and new ways of working

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

TABLE 6
SUMMARY OF ORGANIZATIONAL CHALLENGES TO IMPLEMENTING DEVOPS AGILE METHODOLOGIES

Limitation	Description
Siloed Teams	Teams within an organization often work in isolation, with limited communication and collaboration between them. This can make it difficult to implement a DevOps approach, which emphasizes collaboration and cross-functional teams.
Resistance to Change	Implementing DevOps Agile methodologies often requires significant changes in an organization's culture, processes, and tools. Some team members and leaders may be resistant to these changes, which can slow down or even prevent adoption.
Lack of Automation	DevOps relies heavily on automation to streamline processes and reduce errors. However, some organizations may lack the necessary automation tools and expertise, making it difficult to implement a fully automated pipeline.
Legacy Systems	Many organizations have legacy systems that are difficult to integrate with modern DevOps tools and practices. These systems may require significant time and resources to update or replace, which can be a barrier to adopting DevOps Agile methodologies.
Limited Metrics	DevOps emphasizes the use of metrics and data-driven decision making to improve processes and outcomes. However, some organizations may not have the necessary metrics in place or may not know how to use them effectively, which can hinder their ability to implement DevOps effectively

B. Findings

1) DevOps and Agile Methodologies

After careful consideration of how these two approaches relate to one another and how they may be used to enhance software delivery. This study came to some important conclusions:

a) To increase software delivery speed and team communication, several firms are integrating DevOps and Agile approaches. According to a poll by [17], 95% of respondents had already embraced or planned to use Agile methods, compared to 84% who had previously accepted or planned to adopt DevOps approaches. Additionally, the research revealed that firms that blended DevOps and Agile had higher deployment frequencies and quicker time to market.

b) There are several similarities between DevOps and Agile, such as a focus on teamwork, cooperation, and client input. Both techniques highlight the value of automation and the need to eliminate team silos. According to [20], 75% of businesses that have adopted DevOps also employed Agile techniques.

c) DevOps and Agile share certain commonalities, but there are also some significant distinctions. DevOps focuses on the entire software delivery process, including deployment and operations, whereas Agile focuses on iterative development and providing value to the client. According to a study by [21], Agile was more prevalent in smaller businesses, whereas DevOps was more prevalent in large corporations.

d) DevOps and Agile need major organizational and cultural transformation, making their integration difficult. According to a survey by [22], cultural opposition, a lack of collaboration, and outdated technology are the top obstacles to successfully combining DevOps and Agile.

In brief, DevOps and Agile are two techniques that can work together to help organizations develop software more quickly and improve teamwork. However, incorporating these approaches can be difficult and necessitates considerable organizational and cultural adjustments.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

2) DevOps and Organizational Performance

The results of studies on the relationship between DevOps methods and organizational performance were as follows:

a) According to [23, 24], the performance of software delivery in software enterprises was favorably correlated with the use of DevOps principles. The studies also discovered that the most effective DevOps strategies for enhancing software delivery efficiency were continuous integration, continuous delivery, and monitoring and logging.

b) In a different investigation, [25] examined the effect of DevOps methods on organizational agility and discovered a favorable correlation between the two. The research also revealed that continuous integration and delivery were the most effective strategies for enhancing organizational agility.

c) [26] looked at the connection between organizational performance and DevOps methods in Finnish software firms. According to the study, the most significant DevOps approaches were continuous integration and continuous delivery, which were favorably associated to both software delivery performance and organizational performance.

d) Another study was conducted by [27] to examine how DevOps speeds up company innovation. The study discovered that the adoption of DevOps methods was positively associated to innovation performance, with continuous integration and continuous delivery being the most significant practices.

According to these research, DevOps approaches may enhance organizational performance in terms of software delivery, organizational agility, and creativity. The techniques with the greatest influence on achieving these advantages seem to be continuous integration and delivery.

TABLE 7
IMPACT OF DEVOPS ON ORGANIZATIONAL PERFORMANCE

Factor	Description	Impact of DevOps
Speed	The ability of an organization to deliver software faster.	DevOps improves speed by reducing development cycles, automating processes, and enabling continuous delivery, leading to faster time-to-market and improved competitiveness.
Quality	The reliability and effectiveness of software releases.	DevOps improves quality by integrating testing and quality assurance into the development process, using automation to catch errors early, and providing feedback loops for continuous improvement.
Collaboration	The level of communication and teamwork within an organization.	DevOps encourages collaboration by breaking down silos between development and operations teams, promoting a shared responsibility for software delivery, and enabling cross-functional teams to work together more efficiently.
Innovation	The ability of an organization to create and deploy new ideas and technologies.	DevOps fosters innovation by enabling rapid experimentation, providing faster feedback on new features, and encouraging a culture of continuous improvement and learning.
Scalability	The ability of an organization to handle increased workload and growth.	DevOps improves scalability by automating infrastructure provisioning and deployment, allowing for easy scaling up or down of resources, and enabling the organization to respond quickly to changing business needs.
Security	The level of protection against unauthorized access and data breaches.	DevOps enhances security by integrating security checks into the development process, making security testing a part of every release, and enabling continuous monitoring and remediation of vulnerabilities.
Customer Satisfaction	The level of satisfaction of customers with the organization's products or services.	DevOps improves customer satisfaction by providing faster and more reliable software releases, ensuring that customer feedback is incorporated into the development process, and enabling the organization to respond quickly to customer needs and feedback.

V. RECOMMENDATIONS

For businesses aiming to boost performance, DevOps approaches have a lot to offer. Organizations can increase their productivity, cut costs, and boost customer satisfaction by implementing a DevOps strategy, creating a collaborative culture, automating processes, supporting continuous integration and delivery, utilizing a DevOps toolchain, investing in training and development, and measuring success and continuously improving. DevOps may assist firms in achieving their objectives and staying one step ahead of the competition with the proper methodology. Organizations looking to adopt DevOps practices to enhance their organizational performance should:

A. Develop a DevOps strategy

Creating a concise strategy is the first step in implementing DevOps practices. The objectives of the DevOps initiative, the project's scope, and the resources needed to carry it out should all be specified in this strategy. A strategy for determining success and pinpointing areas that need improvement should be part of it as well. All stakeholders, including IT staff, business leaders, and customers, should be informed of the strategy.

B. Build a Collaborative Culture

Collaboration between the development and operations teams is the foundation of DevOps. Organizations must create a culture that encourages cooperation, communication, and trust if they are to succeed with DevOps. This entails dismantling team silos and promoting cross-functional collaboration. It also entails encouraging an experimental mindset in which failure is seen as a chance to grow and learn.

C. Automate Processes

One of the main elements of DevOps is automation. Organizations may decrease errors, increase productivity, and hasten the delivery of software by automating activities like testing, deployment, and monitoring. Additionally, automation gives IT personnel more time to work on higher-level projects like innovation and strategic planning.

D. Embrace Continuous Integration and Delivery

The DevOps principles of continuous integration (CI) and continuous delivery (CD) allow for the quicker and more dependable supply of software. While CD involves automating the deployment of code changes to production, CI entails regularly integrating code changes into a shared repository. Organizations may lower the risk of errors and accelerate the time to market for new features and products by using CI/CD.

E. Adopt a DevOps Toolchain

Organizations require a collection of technologies that facilitate automation, collaboration, and continuous delivery in order to deploy DevOps methods successfully. These technologies ought to be included in a toolchain that facilitates seamless team collaboration and communication. Jenkins, GitLab, Docker, and Kubernetes are examples of well-known DevOps tools.

F. Invest in Training and Development

The field of DevOps is complicated and developing quickly. Organizations must spend money on IT staff training and development if they want to succeed with DevOps. This involves giving IT professionals access to training programs, conferences, and other materials so they can keep current on the newest DevOps methods and tools.

G. Measure Success and Continuously Improve

Finally, businesses need to evaluate the impact of their DevOps activities and pinpoint areas for improvement. Setting specific performance criteria is required for this, including deployment frequency, lead time, and mean time to recover (MTTR). Regular process reviews and possibilities for optimization and improvement are also part of it.

VI. CONCLUSION

This study report offers a thorough analysis of current studies on DevOps Agile techniques. The results demonstrate both the advantages and difficulties of applying these approaches in software development initiatives. Faster time to market, better software quality, and more interaction between the development and operations teams are all advantages.

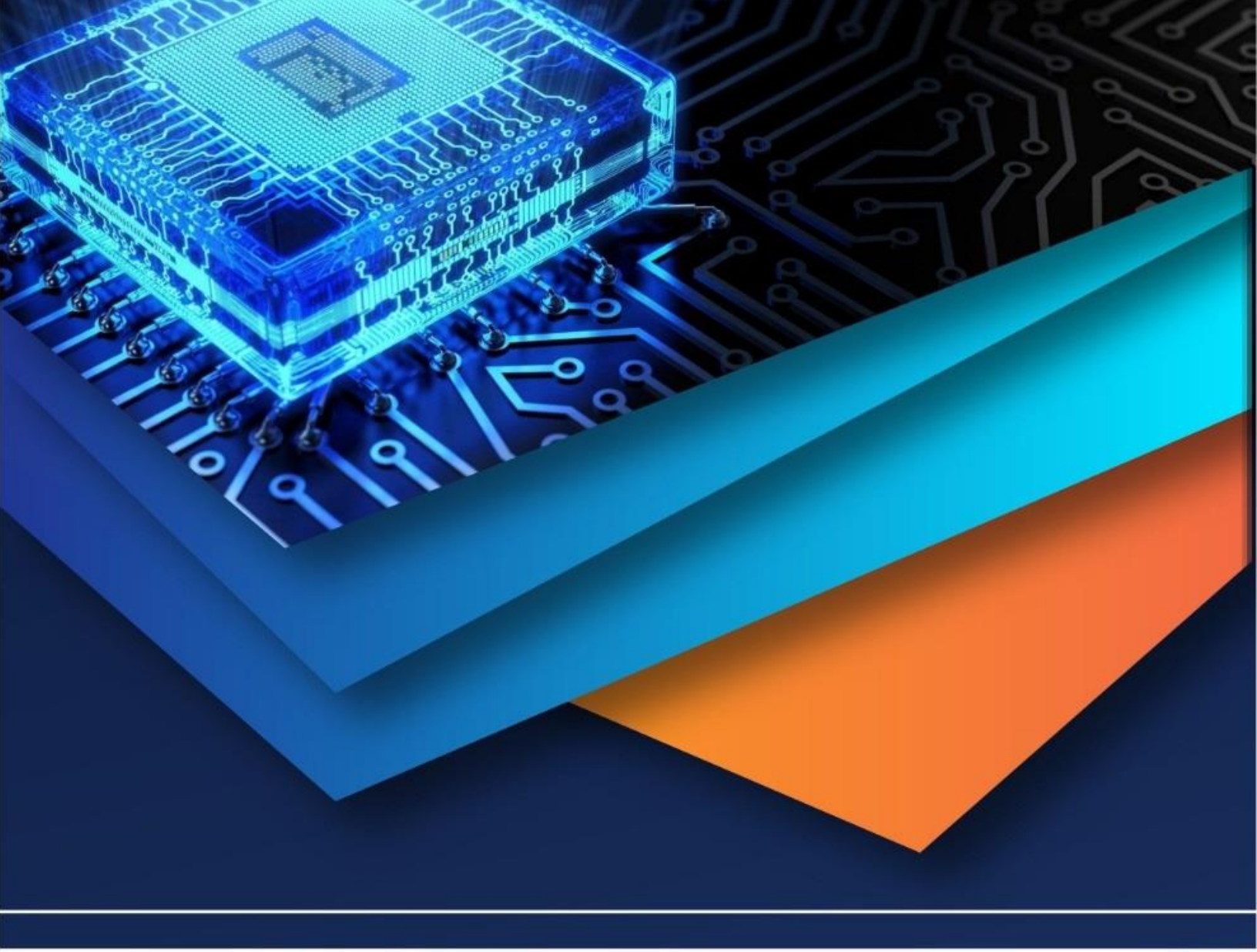
©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

The difficulties, however, include the inability to assess the implementation's success and cultural resistance. The best practices for DevOps Agile methodology adoption are also covered in the paper, including automation, strong leadership support, and continuous integration and delivery. Overall, this study offers insightful information about the current state of DevOps Agile techniques and the elements that make them successful. It acts as a manual for businesses and professionals interested in using these approaches.

REFERENCES

[1] DevOps Institute. (2023). Upskilling IT 2023 Report: Embracing Upskilling and Reskilling Within IT. https://www.devopsinstitute.com/upskilling-it/
[2] Gartner. (2020). DevOps and Agile Are Key to Digital Transformation. https://www.gartner.com/en/documents/3975927
[3] Fowler, M. (2006). Continuous Integration. Retrieved from https://martinfowler.com/articles/continuousIntegration.html
[4] Hashicorp. (2018). What is Infrastructure as Code (IaC)? Retrieved from https://www.hashicorp.com/resources/what-is-infrastructure-as-code
[5] Agile Alliance. (2018). Cross-functional Teams. Retrieved from https://www.agilealliance.org/yes-cross-functional-teams-but-real-ones/
[6] Iqbal, M. (2021). What is iterative development? Retrieved from https://www.scrum.org/resources/blog/what-iterative-development
[7] Agile Alliance. (2001). Agile Manifesto. Retrieved from https://www.agilealliance.org/agile101/the-agile-manifesto/
[8] Almeida, Fernando & Simoes, Jorge & Ferreira Lopes, Sergio Francisco. (2022). Exploring the Benefits of Combining DevOps and Agile. Future Internet. 14. 63. 10.3390/fi14020063.
[9] Gartner. (2021). Market Guide for DevOps Value Stream Delivery Platforms. https://www.gartner.com/en/documents/3991050
[10] McKinsey & Company. (2020). Agile, reliable, secure, compliant IT: Fulfilling the promise of DevSecOps. https://www.mckinsey.com/business-functions/mckinsey-digital/our-insights/devops-the-engine-for-digital-transformation
[11] Forsgren, N., Humble, J., & Kim, G. (2018). Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations. In Amazon (1st edition). IT Revolution Press. https://www.amazon.com/Accelerate-Software-Performing-Technology-Organizations/dp/1942788339
[12] Kaya, I. (2018). Agile vs. DevOps: What's the Difference? Retrieved from https://www.cmswire.com/information-management/agile-vs-devops-whats-the-difference/
[13] Sematext. (2023). 10 Key Benefits DevOps Brings to How a Company Works. Sematext. https://sematext.com/blog/devops-benefits/
[14] Pardo, César & Guerrero, Jonathan & Zúñiga, Karen & Certuche Díaz, Samuel. (2020). A systematic mapping study about DevOps. Journal de Ciencia e Ingeniería.
[15] Gwangwadza, Ashley & Hanslo, Ridewaan. (2022). Factors that Contribute to the Success of a Software Organisation's DevOps Environment: A Systematic Review. 10.48550/arXiv.2211.04101.
[16] Forrester. (2020). Forrester Research on DevOps Quality Metrics that Matter. Forrester Research. https://be.tricentis.com/mediaassets/pdf/Forrester_DevOps_Metrics_2020-1.pdf
[17] Puppet (2020). State of DevOps Report. Puppet by Perforce. https://puppet.com/resources/report/2020-state-of-devops-report/
[18] CA Technologies, DevOps in 2020: https://docs.broadcom.com/doc/trends-in-value-based-devops
[19] DORA. (2023). 2023 Accelerate State of DevOps Report. DevOps Research and Assessment. https://cloud.google.com/devops/state-of-devops-2020
[20] CollabNet VersionOne. (2021). 15th Annual State of Agile Report. https://info.digital.ai/rs/981-LQX-968/images/SOA15.pdf
[21] Gartner. (2020). The Secret to DevOps Success. Gartner. https://www.gartner.com/smarterwithgartner/the-secret-to-devops-success
[22] Forrester. (2022). The State of Agile in Europe 2022. https://www.forrester.com/report/state-of-agile-in-europe-2022/RES178224?ref_search=0_1696496910589/
[23] Huang, Huang. (2021). DevSecOps: Exploring Practices of Realizing Continuous Security in DevOps (in Chinese). Journal of Software. 32. 3014-3035. 10.13328/j.cnki.jos.006276.
[24] Srivastava, Divya & Verma, Madhushi & Sheshar, Shashank & Gupta, Madhuri. (2022). DevOps Tools: Silver Bullet for Software Industry. 10.1007/978-981-19-6290-5_6.
[25] Strieker, J., Volchkov, S., Martin, S., Karlskrona, S., & Sweden. (2022). Broad DevOps Adoption and Organizational Performance of the Software Firm. https://www.diva-portal.org/smash/get/diva2:1736723/FULLTEXT02
[26] Laihonen, P. (2018). Adoption of DevOps Practices in the Finnish Software Industry: an Empirical Study. https://aaltodoc.aalto.fi/bitstream/handle/123456789/34400/master_Laihonen_Paul_2018.pdf?sequence=1&isAllowed=y
[27] Successive Cloud. (2021). What Is DevOps? How Does DevOps Accelerate Business Innovation? Successive Cloud. https://successive.cloud/what-is-devops-and-how-devops-accelerate-business-innovation/

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)