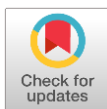


Innovative Computing Review (ICR)

Volume 5 Issue 1, Spring 2024

ISSN(P): 2791-0024, ISSN(E): 2791-0032

Homepage: <https://journals.umt.edu.pk/index.php/ICR>



Article QR



- Title:** DevOps Integration in Agile Software Development: Challenges and Benefits in the Context of Software Industry in Pakistan
- Author (s):** Muhammad Ahmad, Muhammad Adnan*, Anum Bahar, and Amara Parveen
- Affiliation (s):** Riphah School of Computing & Innovation, Faculty of Computing, Riphah International University, Lahore, Pakistan.
- DOI:** <https://doi.org/10.32350/icr.52.02>
- History:** Received: June 19, 2025, Revised: July 23, 2025, Accepted: August 20, 2025, Published: December 29, 2025
- Citation:** M. Ahmad, M. Adnan, A. Bahar, A. Parveen, “DevOps Integration in Agile Software Development: Challenges and Benefits in the Context of Software Industry in Pakistan,” *Innov. Comput. Rev.*, vol. 5, no. 2, pp. 41-56, December. 2025, doi: <https://doi.org/10.32350/icr.52.02>.
- Copyright:** © The Authors
- Licensing:**  This article is open access and is distributed under the terms of [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/)
- Conflict of Interest:** Author(s) declared no conflict of interest



A publication of
School of Systems and Technology
University of Management and Technology, Lahore, Pakistan

DevOps Integration in Agile Software Development: Challenges and Benefits in the Context of Software Industry in Pakistan

Muhammad Ahmad^{1*}, Muhammad Adnan², Anum Bahar³, and Amara Parveen⁴

Riphah School of Computing & Innovation, Faculty of Computing, Riphah International University, Lahore, Pakistan

ABSTRACT DevOps and Agile are two progressive methodologies that are aimed at improving software delivery through cooperation. These methodologies specifically focus on speed in delivering quality solutions. The current research evaluated the prospect of Agile-DevOps implementation in Pakistan's software industry. The study specifically focused on enhancement in the quality of the software and the flexibility of businesses. A survey was also conducted among the working professionals from the Pakistani leading software houses. These software houses included Developers Studio, InvoZone, Cyber Nest, and Design Bub. The purpose of these surveys was to gather insights regarding the real implementation of the solutions proposed in this study. In general, it was revealed that cultural issues, lack of skills, as well as the improper use of tools and processes remain a significant barrier to the integration of Agile-DevOps. The absence of a proper training program implemented in the course of the transition also contributes to the communication breakdown. Nevertheless, the adoption of Agile-DevOps has been proved to be advantageous in that more software(s) are delivered speedily, collaboration is enhanced, and the quality of software is increased with high customer satisfaction. Moreover, the latest trends and practices include Continuous Integration and Deployment (CI/CD), automated testing, Agile-DevOps models, mixed and modern, and cloud in software development in Pakistan. The results obtained revealed that the adoption of Agile-DevOps correlates well with the improvement in software quality of products that underwent structured training for higher successful implementation. Hence, there is a need to undertake targeted interventions of organizational readiness, skills enhancement, as well as leadership to drive change and process management in order to address the barriers highlighted in the study. These recommendations offer a guide that may help the Pakistani software firms to implement the Agile-DevOps model in the best way possible in near future. This may sustain innovation, constant growth, and competitive advantage in the advanced digital world.

INDEX TERMS Agile methodologies, benefits, challenges, DevOps practices

I. INTRODUCTION

The convergence of DevOps and Agile methodologies has become a central theme in the current software development practices. This addresses the growing need for improved collaboration, efficiency, and rapid deployment of high-quality software solutions. With companies going through

the changing digital environment, combining DevOps and Agile practices provide a promising solution to achieving these goals. DevOps builds further on the concepts of Agile, emphasizing a culture of collaboration between development and operation teams as well as using automation, continuous integration, and delivery very heavily. With this synthesis,

*Corresponding Author: muhammad.ahmad.raf@gmail.com

the life cycle of software development is simplified. Moreover, it is more responsive to customer feedback and market changes [1].

Nevertheless, applying DevOps in Agile contexts poses a number of challenges. These include cultural resistance to change, skills shortage, and the fact that it requires significant changes to the organizational processes [2]. The changing nature of Information Technology (IT) functions requires the development of new skills when the old jobs are broadened to involve both development and operational functions [3]. It necessitates a paradigm shift in the culture of the organization, where individual departments give way to a system of collaboration that facilitates a sense of common goals and mutual responsibility [4]. Despite the challenges associated with these changes, the combination of DevOps and Agile approaches to change has significant advantages. Increased cooperation leads to better communication and teamwork and, as a result, to the higher quality of software products and operational efficiency [5]. I

n addition, the integration of the iterative philosophy of Agile and the automation and feedback loop of DevOps allows the teams to release software updates more consistently and regularly in accordance with the dynamic market needs [6]. With companies having to deal with the intricacies of contemporary software development, the notion of the synergy between DevOps and Agile practices is vital in ensuring long-term success and creating an innovation in the sector. Agile software development is a user-centered information systems (IS) and software development technique, and is intended to be constructed incrementally and iteratively. Agile software development methodology is a term that was originally

coined in the year 2001. Agile was created as a process model to counter the problems of the previous approach called ‘the waterfall model of software development’ that was more plan-driven and was called the predictive model. The agile organizations, unlike traditional hierarchy structures, are networked teams with brief learning cycles. The vision of these organizations that relates to the community makes it short and decentralized the decision-making process to immediate teams. Agile strategies are adaptive, not innovative, to control and improve efficiency within the networked knowledge-based economy. Additionally, these are developed and adjusted in the ongoing research effort. The Agile methodology is an extremely efficient process of software development and is based on 4 basic principles. These include individual interactions, working software, customer collaboration, and responding to change. It is scalable in its requirement, project size, and other factors, such as DAD, SAFe, XP, and Scrum that are easy to use and extensive. Agile software development offers several advantages over traditional software development methods, including limited documentation, intensive customer collaboration, rapid delivery, and continuous alignment with business goals or requirements [7]. Since organizations have come to know about the necessity to provide agility and speed to the demands of markets, Agile has been implemented in other fields besides the IT sector in organizations. The first DevOps Days conference was organized in 2009 under the name Patrick Debois, Ghent, Belgium and was also extended to other nations. DevOps is a culture, which encourages the development and operation teams to work together to release code in a timely fashion that is automated to scale-up production.

It introduces the developing stage of the developers, system engineers, administrators, and security specialists. The practice improves speed of an organization, enhances customer service, and competition in the market. Companies are deploying DevOps to improve their software delivery timeline in order to decrease the gap between the functions and development departments after its successful implementation and adoption in software organizations [8]. DevOps model consists of 4 dimensions, that is, collaboration, automation, measurement, and monitoring. DevOps is an agile method of software development that is further discussed in the context of continuous software delivery and integration. Moreover, it is automated to reduce the time to minimal, promotes communication, cooperation, and reliability [9]. Better communication, speed in time to market, responsive, and less disruption is better with the use of DevOps recovery software delivery. It transforms the product development, enhances responsibility, and increases more extensive skillsets and roles [10]. CI/CD) says DevOps = People + Process + Tooling, and it is said to make it more secure and faster. It promotes open mindedness, predictability, training of cross skills, and teamwork. DevOps tools

are essential in uniting different environments and tools to enable the changes to be distributed to different environments. Organizations may have a hard time identifying the appropriate tools and frameworks to automate and integrate different systems [11].

The lifecycle of DevOps is a constant loop of product plan, design, testing, and repair optimization, which includes integration of the development and operations team. It encourages communication and automation, ensures efficiency between planning and delivery, as well as promotes more effectiveness [12].

The methodology of DevOps is simple and its lifecycle includes continuous development, continuous integrations, continuous testing, and continuous monitoring [13]. It assists in the cultivation of continuous improvement culture that results in reduced iterative and update cycles. This not only helps in cutting time to market but it is also easier to identify flaws. Effective implementation should have tools and guidelines of teams [14].

The knowledge of DevOps lifecycle that is subjected to a brief discussion below determines the success of DevOps.

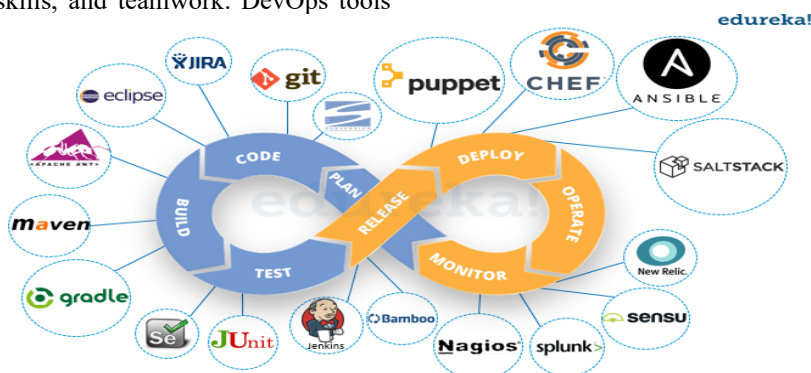


FIGURE 1. DevOps tool chain

Plan: During the planning phase, specific goals are established to be achieved by the

development and operation groups, coordination among the groups is promoted to ensure effective planning of both the development and operation, and risk analysis is conducted to overcome the possible problems and shortcomings of the development process. This approach helps to simplify the management of projects and improve the general concept of decisions.

Code: The code development at this stage involves collaborative code writing, continuous integration, automated testing, and peer code review. This helps provide the quality of the code and elicit defects at early stage of development process. Software developers publish code in bundles, and products, such as Git are useful to version code without difficulty: Automated CI tools are used to compile and execute test cases of the code [15].

Test: During this phase, sustainable testing, such as unit tests, integration tests, and acceptance tests assists in ensuring that the code is of high quality. Additionally, these tests also help recognizing bugs early. Software testing tools, also known as automated testing tools, verify changes to the code that may cause any form of functionality breakage and guarantee code integrity [16].

Deploy: Continuous Deployment (CD) and Infrastructure as Code (IaC) are automated deployment processes at this stage. These are implemented using only tested changes in code in both web application and software production environments. CD is used to test code changes, whereas IaC uses infrastructure configurations in code and provides and manages infrastructure automatically.

Operate: This stage involves monitoring and logging of applications. Application monitoring entails live-time tools that collect data and develop logs to solve problems. Management of incident is

attained through alerts. Thus, response is fast without the need to engage various people and systems. Monitoring and future development as well as operational team improvements regarding incidents are done using feedback loops. This is done to ensure that there is an effective and efficient communication in the system.

Monitoring: In this step, the performance monitoring would involve keeping a daily check of the web application to ensure that it is running optimally. Furthermore, it also entails examining the status of servers among other factors. The monitoring of user experience is concerned with user engagement and satisfaction, whereas data and web application protection is provided by security monitoring [17].

DevOps uses various tools and technologies to have improved communication, integration, and optimization. This enables the incorporation of process and project management which are vital to successful DevOps practices.

Figure 1 suggests the existence of different tools of DevOps and they are discussed as follows: The implementation of DevOps tools in the Agile practice can improve the software development with the help of a better collaboration, addition of automations, and continuous delivery. Nevertheless, software companies face various obstacles to the implementation of these tools, which should be overcome in order to use them in the best possible way.

A. VERSION CONTROL SYSTEMS TOOLS

Git tool is a version control system that enables the collaboration of the developers. Therefore, it is essential to manage the changes in the code. The GitHub/GitLab provides services of Git Repositories,

issues tracking, CI/CD, as well as projects management and allows different members to collaborate.

B. CI/CD TOOLS

Jenkins and Circle CI tools are applied in CI/CD. Jenkin's tool is the open-source automation server of build, deploy, and software development automation. It makes CI/CD workflow configurations simpler as it has its plugins. Circle CI is a CI/CD tool that allows rapid application construction, testing, and deployment as a cloud-based tool [18].

C. CONTAINERIZATION TOOLS

Docker represents a development tool used by developers to build, launch, and operate applications in containers. Whereas, Kubernetes is a management tool used to build applications in containers in order to permit automatic deployments and scaling.

D. INFRASTRUCTURE AS CODE (IAC) TOOLS

Terraform is an open-source application that allows the provision of infrastructure based on declarative configuration language and Ansible applications deployment. It guarantees the conformity of environmental setup [19].

E. MONITORING AND LOGGING TOOLS

Prometheus is a free monitoring system to Kubernetes systems. In the meantime, ELK Stack, which comprises Elasticsearch, Logstash, and Kibana, is applied to provide the analysis and visualization of the real-time logs data concerning the performance of an application and its usage.

F. PROJECT MANAGEMENT TOOLS

JIRA is a project management web-based application. This enables the Agile project development (Scrum and Kanban) and

assists software development teams in their project planning, tracking, and management. Trello project management software is a visualization tool designed to organize the tasks and create teams [20]. DevOps is a radical organizational evolution since it alters how it engineer's software. However, it is based on the agile practices of software development. By combining the two, exploiting the iterations and discipline that Agile inherit with the collaboration principles of both DevOps, along with an automation-centric approach, it is likely to enable organizations to have a more coherent lifecycle.

Agile methodology has improved the communication between users and developers, whereas DevOps is a team-building approach that isolates silos into teams. Although similar, they do not resemble each other completely. There are claims of superiority of DevOps to Agile methodology. It is essential to grasp the peculiarities of each methodology in order to remove ambiguity. Two decades ago, emphasis was put on the development, implementation, and integrated IT. DevOps with the Agile and other software development methods may help revolutionize the software engineering practices, whereby the organization would have the ability to break the cultural barrier, streamline operations, and motivate teams. This method promotes innovation, reliability, and efficiency, which makes it respond to market crises quickly, and thus respond to the requirements of the high quality. DevOps and Agile practices play a key role in ensuring a continuous process of improvement as well as customer satisfaction. A blend of the DevOps and Agile advantages may drive the success of an organization. The adoption of DevOps culture in Agile environments increases the harmonization of development and

operations so that the teams can collaborate effectively and produce high quality software. Agile practice integrated in DevOps promotes the existence of continuous feedback, which is a significant aspect in enhancing iteratively and

innovating. Agile software development philosophy combined with DevOps practices is emerging as an important initiative that companies should implement to improve their software development competencies.

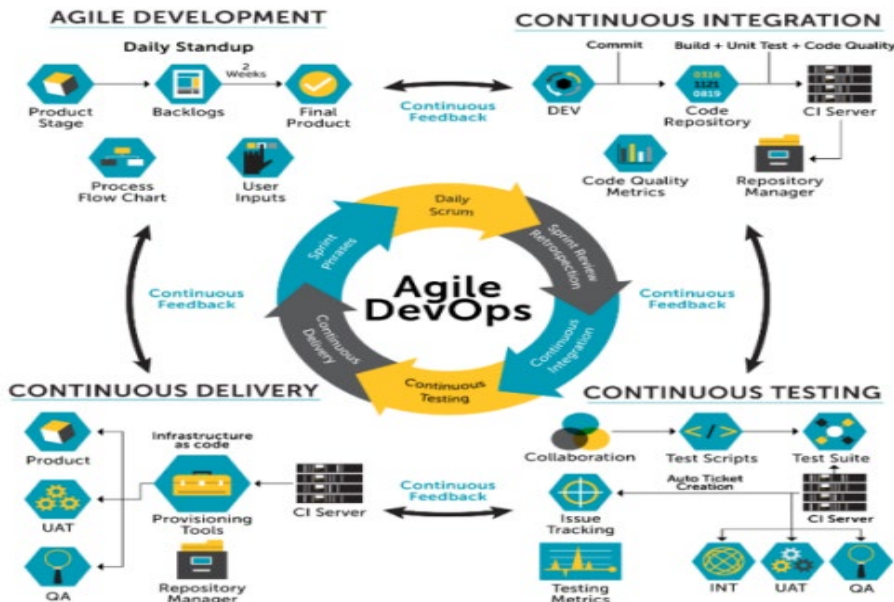


FIGURE 2. DevOps in agile

Figure 2 describes the connection between Agile Development and DevOps. The main concepts of iterations and automation are well explained in various steps of the software development process.

Agile Development: In Agile Development, incremental and iterative development is valued; this is achieved by daily stand-ups, sprint cycles, and frequent user feedback. The teams focus on short development cycles (sprints) of 2-4 weeks and guarantee the evolution of products through the continuous stakeholder feedback. The process starts by defining the stage of the product and backlog, then continues based on the input of the users, refining it to create the final product. Agile allows flexibility of

the teams to respond effectively to the evolving requirements, facilitating a more collaborative and user-centric way of software development.

Continuous Integration (CI): CI is a process of frequently taking a combination of changes in code and testing it automatically in a shared repository. Their changes are submitted by developers and automatically built, tested, and checked with CI systems, such as Jenkins or CircleCI. CI allows finding the problems in the integration in a timely manner, reducing the chance of serious errors during the project. Repository managers and code quality metrics are some of the important tools that help to maintain the integrity and

stability of the application.

Continuous Testing: Continuous testing is a lifecycle process in which, automated testing is performed during the process of development, to guarantee quality and functionality requirements of code changes. It deals with the development of test scripts, automation of issue tickets, and test of the application by integration and user acceptance testing. Developers, testers, and stakeholders should cooperate with each other, and such tools as test suites and issue trackers enable them to do it.

Continuous Delivery (CD): CD in Agile enables the software to be deployed to production on-command, with the help of tools, such as Terraform or Ansible. The CD deploys repository managers and CI servers to run concrete code in a QA or staging apparatus, preceding a clear launch to general utilization. This is in line with the value proposal of Agile, which shows value in small increments that allow releasing it in a low-risk and fast manner.

After reviewing the literature, the study addressed the issues that are the primary to the organizations in order to use DevOps practices in the framework of Agile practices. These included:

- To quantify the gains and the advantages realized in the process of integrating DevOps in Agile designations.
- To collect and analyze information that is industry-wide through surveys, determine trends, and patterns that are common.
- To introduce the strategies of overcoming the obstacles and maximizing benefits in the process of Agile-DevOps integration.

G. RESEARCH QUESTIONS

The current research aimed to address the following questions:

RQ 1. What are the critical issues that organizations experience in integrating DevOps into Agile frameworks?

This question would investigate the most significant challenges that organizations face when implementing the DevOps practices within the already established Agile models. These include the cultural resistance, the absence of collaboration between teams, the complexity of the toolchains, and the goal coordination across the development and operations teams.

RQ 2. How far does DevOps offer any benefits in the form of integration of Agile methodologies in organizations?

This question explores the quantifiable benefits of integrating DevOps with Agile, including shorter delivery time, better collaboration, higher quality of the product and greater operational efficiency, as well as evaluating whether the benefits are achieved in reality.

RQ 3. What is the effective way of applying these Agile and DevOps tools in today's software development?

The question is aimed at revealing the best practices, strategies, and frameworks to successfully apply the methods of Agile and DevOps, such as automation, CI/CD, and the promotion of a collaborative culture.

RQ 4. What is the role of organizational culture and leadership in facilitating or preventing the integration of Agile-DevOps?

The question emphasizes as to how organizational culture and leadership can support or impair the process of Agile and DevOps integration with the focus on trust, communication, and a common vision as

the key elements towards successful adoption.

II. LITERATURE REVIEW

In last couple of years, considerable attention has been paid on integrating DevOps with Agile methodologies, particularly in software houses. The literature review in the current study summarizes recent examinations regarding the challenges, rewards, and developments of this integration in software development practices. DevOps brings up the collaboration aspect between development and operations teams which fosters better collaboration, improving software quality, time to market, reducing cost, and risk. Integrating DevOps practices with Agile frameworks allows organizations to leverage CI/CD that supports the delivery of high-quality software in a dynamic environment. In addition to the speeded-up innovation, this synergy helps in market demand adaptability [21]. DevOps in particular is an extension of Agile as it helps to provide better communication between the development and operations departments. It incorporates principles intended to enhance software quality regarding the CAMS (Culture, Automation, Measurement, Sharing) where a major focus is placed upon automation as being fundamental to improving the quality of the software [22]. DevOps, which implied the collaboration of the development team with the operations team, is considered as the advancement of Agile methodologies in software development. It is designed to facilitate working procedures, increase communication, as well as the speed of software releases [23].

DevOps in Agile environments facilitates significant benefits such as; time to market, improved software quality, and customer satisfaction [24]. However, the challenges

of DevOps implementation in software houses cannot be neglected. Organizations experience numerous challenges, such as cultural resistance, skill deficiency, and integration challenges. DevOps adoption makes the process even trickier as there are no definite guidelines and standardized methodologies to adopt it. Another problem encountered by many organizations is the availability of numerous tools and the need for a radical change in the development paradigm [25]. Another advantage, when combining DevOps with Agile, is the utilization of automation which positively impacts cross-team collaboration since more processes become integrated and the time to market is reduced with shorter software delivery cycles [26]. This integration also comes with flexibility, effectiveness, and dependability of service delivery, especially in health facilities [27]. Agile with DevOps makes projects more collaborative, comprehensive in nature, and faster in terms of time to market. The issues include change handling and getting the right cooperation from the team, however, the advantages, such as visibility, lessened risks, and better software quality are enough to make the effort worthwhile [28].

DevOps is an extension of Agile as it improves collaboration, automation, integration, and software delivery. Cultural issues that may need to be addressed include activities, such as cultural adaptation and gradual implementation, while the benefits include: shortening of the time of product release, enhancement of product dependability, customer feedback management, improvement of the quality assurance, as well as security and software development [29]. DevOps improves the speed of software releases, automates tests, and integrates with related departments in software houses [30]. Some of the problem areas associated with DevOps in Agile

include concerns about cultural change and integration. However, on the positive side, the integration leads towards better collaboration, faster deployment, and improved quality in software houses hence, leading towards efficient development and operations [31]. Agile-related DevOps, like the problem, need more practical experience, documentation resources need to be enhanced, and training needs to be more formal.

It is worth mentioning that, although the listed challenges are not strictly related to Agile, they might affect software houses implementing DevOps within Agile frameworks [32].

Combining DevOps practices with Agile methodologies has the potential to greatly improve software development by fostering better teamwork, increasing automation, and enabling continuous delivery. Nevertheless, companies encounter obstacles, such as resistance to change, lack of necessary skills, difficulties in choosing appropriate tools, and the absence of standardized implementation procedures, which prevent them from fully realizing these advantages. These challenges call for a comprehensive analysis of the pros and cons associated with merging Agile and DevOps approaches. Such an examination would provide practical strategies to overcome hurdles, streamline processes, and empower organizations to achieve superior software quality, quicker delivery times, and greater flexibility in ever-changing development landscapes [33].

III. METHODOLOGY

The proposed methodology of DevOps Integration in Agile Software Development: Challenges and Benefits in the Context of Pakistan's Software Industry has two phases:

Phase #1: Based on Literature Review

Phase #2: Based on Questionnaire Survey

This study utilized a questionnaire to explore the challenges and benefits of integrating DevOps with Agile methodologies. Web-based surveys were designed to gather data from at least 100 industry professionals who implement Agile and DevOps practices across various sectors. The survey results highlighted specific areas of difficulty, advantage, and emerging trends in the current combination of these approaches. Quantitative data analysis employs descriptive and inferential statistical methods to validate or disprove conclusions and uncover relationships between examined factors. Additionally, data analysis was conducted by using SPSS. Ethical considerations in the research process included obtaining participant consent, ensuring anonymity, allowing voluntary participation, and safeguarding the data. Researchers utilized online platforms, such as Google Forms or LinkedIn to administer the surveys. This streamlined method involves completing questionnaires to generate recommendations for the convergence of Agile and DevOps practices.

The first part of this study therefore, required conducting an extensive systematic literature review (SLR) that synthesizes published research on Agile and DevOps as well as the challenges, benefits, and trends related to the two methodologies. The main focus of this phase was to highlight research questions and objectives as well as to develop a theoretical framework for the research. Through the analysis of the papers, journals, and case studies, the study investigated the way in which organizations across the globe and more important organizations in the software industry of

Pakistan are adopting DevOps within the ecology of Agile. The implications and enablers section of the literature review also covered the Agile-DevOps integration and its related organizational, technical, and cultural factors, and any best practices could be implemented within Pakistani IT industry.

The category used in the second phase of this study was a structured questionnaire survey that provided primary data collected from at least 100 practicing professionals within the software industry in Pakistan. The targeted groups comprised the software engineers, DevOps specialists, project managers, IT leaders, and the employees of software houses, such as Developers Studio, InvoZone, Cyber Nest, and Design Bub. The questionnaire was administered via online modes, such as Google Forms and LinkedIn account to make it easily accessible by the people from all over the world, leading towards a representation of diversified backgrounds and varying industries.

The purpose of the questionnaire was to look at:

- The several issues/challenges organizations go through while adopting DevOps in relation to Agile practices.
- The benefits of successfully implementing DevOps.
- The role organizational culture and leadership play in transforming to Agile-DevOps.
- The adoption of Agile-DevOps in Pakistan's IT industry: current and emerging trends.

For the data analysis, both descriptive and inferential statistical approaches were employed with the help of SPSS software

to measure correlation between Agile-DevOps practices and software quality. Various ethical issues were upheld here which included; consent of the individuals, anonymity, as well as use of voluntary participants and security of data.

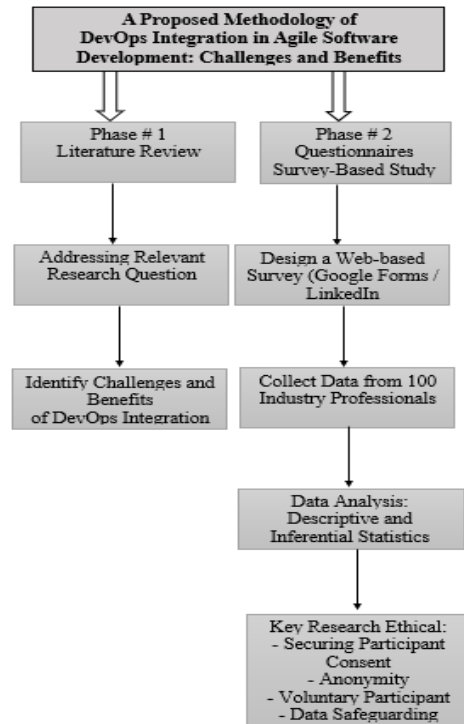


FIGURE 3. Proposed approach for DevOps integration in agile software development: challenges and benefits

It can be seen that these two-phased research works provide a vast and detailed understanding of Agile-DevOps integration in the software industry to cater for challenges in Pakistan. Moreover, these also provide the much-needed strength in problems and solutions derived from the data. The review shall come up with relevant recommendations for the implementation of Agile—DevOps in organizations that wish to improve

efficiency, teamwork, and software quality. Figure 3 shows the proposed methodology of DevOps integration in Agile software development.

A. DATA ANALYSIS AND RESEARCH ETHICS

Quantitative data analysis, encompassing both descriptive and inferential statistical approaches, was performed on the gathered survey responses. This analytical process validates the research findings, explores correlations between various integration-influencing factors, and establishes a solid foundation for conclusions. Ethical protocols are rigorously maintained throughout the study, including securing informed consent, guaranteeing anonymity, ensuring voluntary participation, and implementing stringent data protection measures. The survey distribution utilized online tools, such as Google Forms or SurveyMonkey, facilitating an efficient data collection process while safeguarding participant confidentiality.

III. RESULTS AND DISCUSSION

The study responded positively to the set research questions by using data collected from 100 professionals from the software industry of Pakistan including Developers Studio, InvoZone, CyberNest, and DesignBub. The study provided an analysis of the difficulties, advantages, and approaches to the enactment of Agile-DevOps integration, and the influence of organizational culture in the Pakistani software industry. This research identified the challenges and key factors that affect Agile-DevOps adoption in Pakistan's industry while stressing the role of culture and leadership for technical teams. According to the research, Agile and DevOps provide massive benefits in software process implementation. However, the integration of the two

methodologies is full of profound challenges. The challenges are resistance to change, lack of skills amongst the workers, and misfit of the tools and processes therefore hitch the progress in the process of change, leading to decreased effectiveness. However, in the case of organizations that implement DevOps within an Agile environment, this not only enhances lending, but also increases connectivity, the pace of creating new products and services, and the quality of the software created. Hence, there is a need to come up with a sound framework when implementing DevOps. It looks at the areas that states that integration is not only a technology and automation affair but a cultural and leadership also affair. It is, therefore, witnessed that enterprises committed to learning, skill-building, and process optimization reap better outcomes in Agile-DevOps journey in the long run. This research also highlights that CI/CD pipelines, automation testing, and cloud solutions are crucial factors that support Agile-DevOps integration. This shows that Pakistan is progressing in its software development culture. The study overall offered a guideline for organizations interested in improving software delivery development with the objective of maintaining competitiveness in the global context.

A. CHALLENGES IN INTEGRATING AGILE AND DEVOPS (RQ1)

In response to the RQ1, the study highlighted the main problems that organizations encounter in adopting DevOps in Agile frameworks. The top barrier (65%) identified was cultural resistance. This is because most organizations' cultural practices are a hindrance to DevOps, requiring a different work flow of automation and decentralized collaboration. Additionally, skills deficit at

48% presents a challenge to implementing CI/CD. This is because most IT specialists in organizations do not have a satisfactory level of experience in CI/CD, cloud, and the use of automation tools. Tool and process misalignment: 42% of the organizations indicate that this aspect leads to difficulties in optimizing integration between Agile methodologies and DevOps automation

within the teams. This study underlined the requirement for training interventions, process integration approaches, and organizational development campaigns to address these issues in the Pakistani software industry. Figure 4 presents the results obtained in percentages for the challenges of DevOps integration.

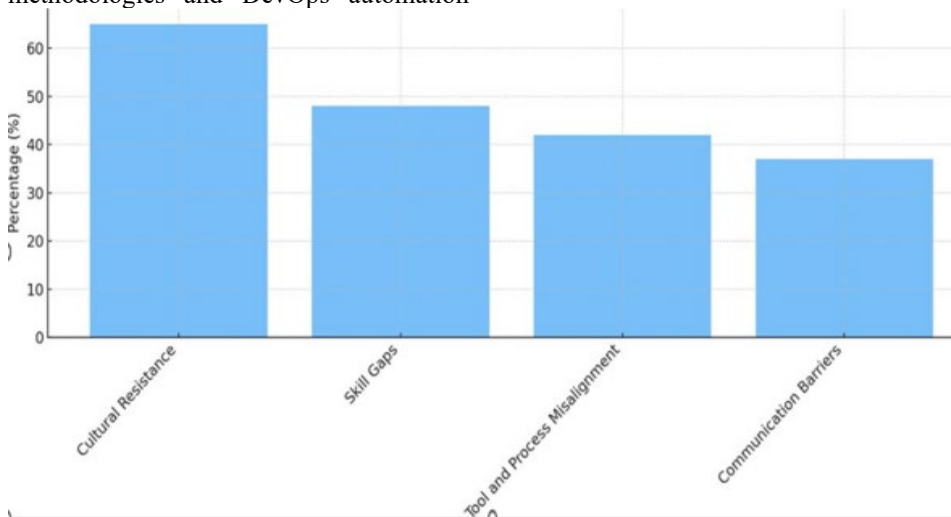


FIGURE 4. Obtained results in percentages for the challenges in agile-DevOps integration

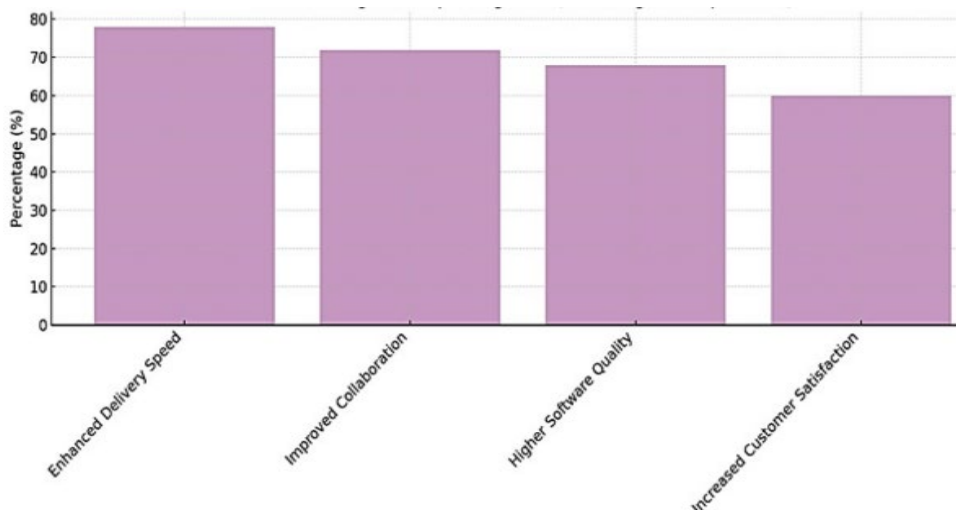


FIGURE 5. Obtained results in percentages for the benefits of agile-DevOps integration

B. BENEFITS OF AGILE-DEVOPS ADOPTION (RQ2)

Regarding the RQ2, the current research identified numerous advantages of Agile-DevOps implementation. Faster software delivery (78%) is followed by other advantages, helping organization's software development cycles get released and more stable at a faster rate. Collaboration (72%) improves working relations and fosters cohesion between the development, testing, and the IT operations department. In addition, with regards to the software quality improvement, 68 percent of the respondents reported this. It was carried out through automated testing, continuous monitoring, and early detection of defects, showing how DevOps improves and optimizes the reliability. These advantages play an important role for Pakistani software industry. This is because domestic firms are trying to compete in the international market and improve the quality of software produced for international buyers. The results obtained are in percentages as shown in Figure 5 under the benefits and organizations that apply Agile-Devops Methodologies.

C. EFFECTIVE IMPLEMENTATION STRATEGIES FOR AGILE-DEVOPS (RQ3)

Answering the RQ3, the study reviewed and identified the best practices or effective implementation strategies for the application of Agile-DevOps in contemporary software development. The most preferred approach priorities by the respondents were CI/CD (80 percent) to develop and release automations, testing (75 percent) that reduces the time of feedback loop mechanism, and makes software more reliable, and Agile DevOps (58 percent) a combination of more flexible Agile and automated DevOps approach. Furthermore, DevOps as a service (55%) utilized cloud-based solutions for its scalability as well as cost effectiveness of tools and integration of other tools. In software industry of Pakistan, these strategies are gradually being implemented in order to maximize its efficiency and to compete with the global standards of software development. Figure 6 depicts the obtained results for the trends in Agile-DevOps practices.

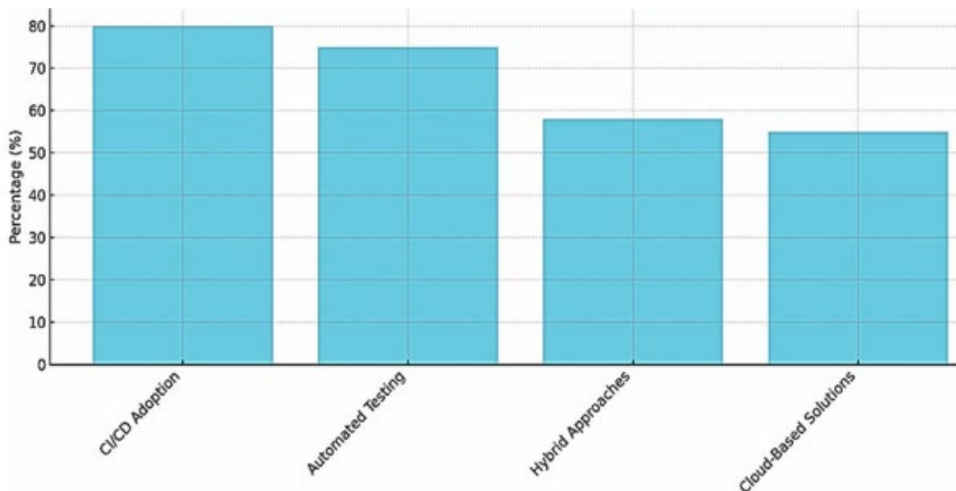


Figure 6. Obtained results for the trends in agile-DevOps practices

D. LEADERSHIP AND ORGANIZATIONAL CULTURE'S CONTRIBUTIONS TO AGILE-DEVOPS SUCCESS (RQ4)

Concerning RQ4, it establishes that indigenous factors, such as organizational culture and leadership are critical in Agile-DevOps implementation. It is essential to identify and promote the leadership-driven interventions to address certain essential issues including lack of collaboration, cultural change, and the lack of effective knowledge sharing for establishing Agile-DevOps integration. It highlighted the lack of adoption, and supports the proposition that even if organizations are willing to embark on cultural changes there is still a critical need for leadership commitment.

The study used SPSS for a correlation analysis to examine the relationship between Agile-DevOps practice implementation and software quality outcomes. Statistical analysis confirmed a strong positive correlation between Agile-DevOps adoption and software quality ($r = 0.78$), and a significant relationship ($p < 0.05$) between training initiatives and successful integration. Survey participants stressed the importance of nurturing a collaborative culture, investing in skill development, and implementing automated tools to overcome challenges and fully realize the benefits of Agile-DevOps integration. Table 1 presents the survey results collected from 100 professionals from different software houses. It offers an overall abstract of the difficulties, advantages, and patterns that have been recognized during the analysis. Moreover, it also presents the category-wise compiled results for the challenges and benefits of DevOps integration in Agile Software development.

TABLE I

CATEGORY-WISE OBTAINED RESULTS IN PERCENTAGES FOR DEVOPS INTEGRATION IN AGILE SOFTWARE DEVELOPMENT: CHALLENGES AND BENEFITS

Category	%
Cultural Resistance	65
Skill Gaps	48
Tool and Process Misalignment	42
Communication Barriers	37
Enhanced Delivery Speed	78
Improved Collaboration	72
Higher Software Quality	68
Increased Customer Satisfaction	60
CI/CD Adoption	80
Automated Testing	75
Hybrid Approaches	58
Cloud-Based Solutions	55

The questionnaire that was arranged on 100 similar software houses was productive in terms of identifying the relationship between the Agile and the DevOps methodologies. Major issues including cultural resistance and skills deficits were highlighted, while major advantages including the delivery duration and communication flow improvement were noted. The new direction of the software development processes can be illustrated by changes, which are CI/CD use, e.g., and automated testing. Such insights provide a rationale for specific interventions, training, and leadership guidance to overcome the identified difficulties and optimize the opportunities of Agile-DevOps coupling.

Therefore, the results provided valuable data on the use of Agile and DevOps in Pakistani software industry, advantages and difficulties experienced by organizations during their integration, and their possible results. RQ 1 centered on the challenges, cultural resistance, lack of skills and

mismatched tools/processes emerge as some of the largest challenges and this once again goes on to justify that, DevOps is not just about technology, but cultural change. Currently, most of the software firms in Pakistan are still operating with the conventional structure of IT firms where the top-down approach centralizes decision-making, thus preventing the implementation of DevOps methodologies. To respond to such challenges, for instance, the organizations need to invest in targeted trainings, leadership-driven change solutions, and enhanced tool incorporation methods.

According to RQ2, the benefits of Agile-DevOps reveal the positive changes that can be witnessed from this integration. Thus, by applying DevOps in Agile settings, the organizations are able to achieve shorter time-to-market, better collaboration, and improved quality of the software that is crucial for organizations to remain competitive in today's rapidly evolving IT environment. However, as seen, there remain issues that affect the ability of organizations to standardize their DevOps practices fully; this is in the area of structures and best practices. This means that organizations should pay attention to standard platforms of DevOps adoption plan combined with automated testing and CI/CD pipeline as well as to integrate Agile and DevOps framework. The study also emphasized that the management system and leadership are key factors that determine the success of Agile-DevOps. Leadership support is always required to build such an environment that is team-oriented and centered around innovation cooperation. In the absence of top management support, you would get workers accustomed to the old method of project development and operation of organizations. This necessitates the call for change of management

processes, leadership development, and rewards that would define and encourage programs that can be brought under the Agile-DevOps initiative. It also reveals that Agile-DevOps integration is a process that is implemented continuously. It is not a change that occurs once. An organization has to constantly assess and optimize its strategies, to stay on par with such advanced trends as cloud, AI and AI-based automation, and security-focused DevOps or DevSecOps. Furthermore, the organizations in the context of Pakistan have to ensure their Agile-DevOps alignment to improve their operations with the international market competitors, mainly in terms of agility, efficiency, or innovations.

E. CONCLUSION

The study concluded that the combination of Agile methodology with DevOps practices is one of the biggest updates in software development that attempts to foster structures in order to enhance joint work, the speed of software delivery, and quality. However, culture, lack of skills, tools not properly aligned, and communication are the barriers to implementation. So, this study will reduced delivery time, better team working, better quality of software and customer satisfaction.

The research proposed future research directions that included: 'longitudinal research should be conducted that assesses Agile and DevOps adoption over time'. Industry-specific strategies, tools' effectiveness, cultural adaptation models, effects of training on Agile, and quantitative measures need to be taken in order to improve the synergy between Agile and DevOps.

Author Contribution

Muhammad Ahmad: conceptualization, methodology, investigation, data curation,

formal analysis, writing – original draft. **Muhammad Adnan:** supervision, validation, methodology, writing – review & editing, project administration. **Anum Bahar:** investigation, data curation, formal analysis, validation, writing – review & editing. **Amara Parveen:** investigation, data curation, writing – review & editing.

Conflict of Interest

The authors of the manuscript have no financial or non-financial conflict of interest in the subject matter or materials discussed in this manuscript.

Data Availability Statement

Data supporting the findings of this study will be made available by the corresponding author upon request.

Funding Details

No funding has been received for this research.

Generative AI Disclosure Statement

The authors did not use any type of generative artificial intelligence software for this research.

REFERENCES

- [1] D. Nyale, “Unravelling DevOps agile methodologies: A comprehensive review of recent research,” *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 11, no. 11, pp. 2012–2021, Nov. 2023, doi: <https://doi.org/10.22214/ijraset.2023.56986>.
- [2] P. Craciun, “Why startups outpace multinationals in leveraging DevOps,” *Proc. Int. Conf. Bus. Excellence*, vol. 18, no. 1, pp. 3421–3429, Mar. 2024, doi: <https://doi.org/10.2478/picbe-2024-0277>.
- [3] A. Hemon, B. Lyonnet, F. Rowe, and B. Fitzgerald, “Conceptualizing the transition from Agile to DevOps: A maturity model,” in A. Elbanna, Y. K. Dwivedi, D. Bunker, and D. Wastell, eds. *Smart Working, Living and Organising*, 2018, pp. 209–223, doi: https://doi.org/10.1007/978-3-030-04315-5_15.
- [4] A. Wiedemann, M. Wiesche, H. Gewalt, and H. Krcmar, “Understanding how DevOps aligns development and operations: A tripartite model of intra-IT alignment,” *Eur. J. Inf. Syst.*, vol. 29, no. 5, pp. 458–473, Sep. 2020, doi: <https://doi.org/10.1080/0960085X.2020.1782277>.
- [5] A. Omonije, “Agile methodology: A comprehensive impact on modern business operations,” *Int. J. Sci. Res.*, vol. 13, no. 2, pp. 132–138, Feb. 2024, doi: <https://doi.org/10.21275/sr24130104148>.
- [6] S. M. R. A. Masud, M. Masnun, A. Sultana, F. Ahmed, and N. Begum, “DevOps enabled agile: Combining agile and DevOps methodologies for software development,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 11, Nov. 2022, doi: <https://doi.org/10.14569/ijacsa.2022.0131131>.
- [7] J. W. Beard *et al.*, “Agile development: The promise, the reality, the opportunity,” *Proc. CEUR Workshop*, 2024.
- [8] A. Nalamwar and P. Sirsat, “An empirical study on DevOps,” *J. Artif. Intell. Gen. Sci.*, vol. 3, no. 11, 2019.
- [9] N. Azad, “Understanding DevOps critical success factors and organizational practices,” in *Proc. 5th Int. Workshop Softw.-Intensive Bus.*, Pittsburgh, PA, USA, 2022, pp. 1–10, doi: <https://doi.org/10.1145/3524614.3528627>.

- [10] H. R. Kadaskar, "Unleashing the power of DevOps in software development," *Int. J. Sci. Res. Mod. Sci. Technol.*, vol. 3, no. 3, pp. 1–7, 2024, doi: <https://doi.org/10.59828/ijrsmst.v3i3.185>.
- [11] M. Moez et al., "Comprehensive analysis of DevOps: Integration, automation, collaboration, and continuous delivery," *Bull. Bus. Econ.*, vol. 13, no. 1, 2024, doi: <https://doi.org/10.61506/01.00253>.
- [12] M. Gokarna, "DevOps phases across software development lifecycle," *IEEE*, 2021, doi: <https://doi.org/10.36227/techrxiv.13207796.v2>.
- [13] L. Gren and P. Ralph, "What makes effective leadership in Agile software development teams?" in *Proc. 44th Int. Conf. Softw. Eng. (ICSE)*, Pittsburgh, PA, USA, 2022, pp. 1–10, doi: <https://doi.org/10.1145/3510003.3510100>.
- [14] J. O. Onyarin, "A complete guide to DevOps best practices," *Zenodo*, vol. 20, no. 2, pp. 1–6, 2022, doi: <https://doi.org/10.5281/zenodo.6376787>.
- [15] J. A. Tsapa, "Integrating DevOps with Agile and other software development methodologies," *Open Access J.*, vol. 3, no. 3, 2022.
- [16] C. Mamatha and S. C. V. S. L. S. R. Kiran, "Implementation of DevOps architecture in project development and deployment," *Int. J. Sci. Res. Comput. Sci. Eng.*, vol. 6, no. 2, pp. 87–95, 2018, doi: <https://doi.org/10.26438/ijrsce/v6i2.8795>.
- [17] M. Choras et al., "Measuring and improving Agile processes in a small-size software development company," *IEEE Access*, vol. 8, pp. 78452–78466, Apr. 2020, doi: <https://doi.org/10.1109/ACCESS.2020.2990117>.
- [18] C. Ebert, G. Gallardo, J. Hernantes, and N. Serrano, "DevOps," *IEEE Softw.*, vol. 33, no. 3, pp. 94–100, May 2016, doi: <https://doi.org/10.1109/MS.2016.68>.
- [19] J. Shah, D. Dubaria, and J. Widhalm, "A survey of DevOps tools for networking," in *Proc. IEEE UEMCON*, New York, NY, USA, 2018, pp. 185–188, doi: <https://doi.org/10.1109/UEMCON.2018.8796814>.
- [20] J. G. Süß, S. Swift, and E. Escott, "Using DevOps toolchains in agile model-driven engineering," *arXiv*, 2021, doi: <https://doi.org/10.48550/arXiv.2111.11607>.
- [21] U. Hamza, S. M. Syed-Mohamad, and N. L. Abdullah, "Exploring the benefits, challenges, and guidelines of DevOps adoption," *J. Math. Sci. Informatics*, vol. 4, no. 2, 2024, doi: <https://doi.org/10.46754/jmsi.2024.10.007>.
- [22] M. Gall and F. Pigni, "Taking DevOps mainstream: A critical review and conceptual framework," *Eur. J. Inf. Syst.*, vol. 31, no. 5, pp. 548–567, Sep. 2022, doi: <https://doi.org/10.1080/0960085X.2021.1997100>.
- [23] F. E. Aouni et al., "A systematic literature review on Agile, Cloud, and DevOps integration," *Inf. Softw. Technol.*, vol. 177, 2024, Art. no.

- 107569, doi: <https://doi.org/10.1016/j.infsof.2024.107569>.
- [24] F. Almeida, J. Simões, and S. Lopes, "Exploring the benefits of combining DevOps and Agile," *Future Internet*, vol. 14, no. 2, p. 63, Feb. 2022, doi: <https://doi.org/10.3390/fi14020063>.
- [25] O. A. Akinola, O. Oyerinde, and A. Akinola, "Implementation of DevOps in healthcare systems," *J. Artif. Intell. Gen. Sci.*, vol. 2, no. 1, pp. 166–170, 2024, doi: <https://doi.org/10.60087/jaigs.v2i1.p170>.
- [26] A. Pandey and K. N. Parmar, "Unleashing the power of DevOps: Revolutionizing software development practices," *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, 2024, doi: <https://doi.org/10.32628/cseit24102121>.
- [27] M. Gokarna and R. Singh, "DevOps: A historical review and future works," *arXiv*, 2020, doi: <https://doi.org/10.48550/arXiv.2012.06145>.
- [28] L. Riungu-Kalliosaari *et al.*, "DevOps adoption benefits and challenges in practice: A case study," in *Proc. Int. Conf. Softw. Eng.*, 2016, pp. 1–10, doi: https://doi.org/10.1007/978-3-319-49094-6_44.
- [29] C. Wang and C. Liu, "Adopting DevOps in agile: Challenges and solutions," 2018. [Online]. Available: <http://www.diva-portal.org/smash/record.jsf?pid=diva2:1228684&dswid=5071>.
- [30] E. J. Giraud, "A mixed-method study of DevOps challenges," *Inf. Softw. Technol.*, vol. 161, Art. no. 107244, Sep. 2023, doi: <https://doi.org/10.1016/j.infsof.2023.107244>.
- [31] S. M. Mohammad, "DevOps automation and agile methodology," *J. Softw. Eng.*, vol. 5, no. 3, 2017.
- [32] S. Alsaqqa, S. Sawalha, and H. Abdel-Nabi, "Agile software development: Methodologies and trends," *Int. J. Interact. Mob. Technol.*, vol. 14, no. 11, pp. 246–270, 2020, doi: <https://doi.org/10.3991/ijim.v14i11.13269>.
- [33] R. S. Ramesh, "DevOps and agile computing," in *Proc. IEEE Int. Conf. Comput., Commun. Syst. (INCCCS)*, 2024, doi: <https://doi.org/10.1109/INCCCS60947.2024>.