Critical Challenges of Continuous Integration and Testing (CI/CT) in DevOps: A Systematic Literature Review Protocol with preliminary Results

Anum Bahar
Faculty of Computing
Riphah International University,
Lahore, Pakistan

Muhammad Yaseen
Faculty of Computing
Riphah International University
Lahore, Pakistan

Muhammad Asif Nauman
Faculty of Computing
Riphah International University
Lahore, Pakistan

Amara Parveen
Faculty of Computing
Riphah International University,
Lahore, Pakistan

ABSTRACT
Continuous Integration (CI) and Continuous Testing (CT) are essential components in DevOps for improving software delivery efficiency and quality. However, the implementation of CI and CT in DevOps settings faces considerable challenges that affect scalability, efficiency and dependability. This Systematic Literature Review (SLR) protocol aims to precise and examine the key challenges in adopting CI/CT. The investigation revolves around one primary research questions: recognizing challenges, evaluating their effects, creating an assessment framework, and confirming findings through case studies. The study employs a methodical approach, encompassing review planning, execution, and reporting. An initial screening process evaluated 2,200 research papers, reducing them to 100 pertinent studies that offer valuable insights into the most pressing issues impacting CI/CT. Preliminary findings reveal recurring challenges such as automation issues, integration challenges, test instability, scalability limitations, security weaknesses, and the absence of standardized procedures. These challenges hinder smooth software development and deployment, underscoring the necessity for additional research and enhancements in CI/CT practices. Through a systematic analysis of these challenges, this study contributes to a more comprehensive understanding of CI/CT constraints in DevOps. The results provide a foundation for future research, assisting organizations and researchers in developing more effective strategies to enhance software development processes and optimize CI/CT workflows.

Keywords
DevOps, Continuous Integration, Continuous Testing, Challenges, Best Practices, Systematic literature Review.

1. INTRODUCTION
Continuous Integration (CI) and Continuous Testing (CT) are critical steps in today's development cycles, especially in DevOps. Since companies have aimed to shorten the delivery cycle and improve software quality, CI practices have become crucial. However, working with the CI and Test Automation environment creates several issues that can affect the efficiency of the processes. [1] DevOps is a set of practices that incorporates software development (Dev) and IT operations (Ops) to improve collaboration and the software delivery process. From breaking down traditional silos to creating a culture of continuous integration, continuous testing, continuous delivery, and continuous improvement between dev teams and ops teams. [2] Continuous Integration (CI) facilitates seamless integration of changes for prompt testing and feedback. Concurrent Testing (CT) enhances this by conducting tests simultaneously with development, improving communication and software delivery efficiency. [3]

1.1 DevOps Concept

The first conference of DevOps Days was organized by Patrick Debois in the year 2009 and was held in Ghent Belgium. This meeting later progressed to other countries. DevOps is the strategy that stabilizes the cooperation between the development and operation departments to release the code faster and automatically. It includes all developers involved in the production phase and system engineers, administrators, and security specialists. It makes the work of an organization fast, facilitates the delivery of services to its customers, and enhances competition in a particular market [4]. DevOps breaks down silos between development and operations teams through culture, automation, feedback, and constant improvement for high-quality software in a changing environment. [5][6]. When the development and operations teams are integrated, the organization is better placed to address market demands and customers' feedback most efficiently and effectively [7]. DevOps has four dimensions: collaboration, automation, measurement, and monitoring. DevOps is a continuous improvement to the agile software development process that is a software construction that stresses continuous software deployment and continuous integration with automation technology minimizing delays and enhancing communication, teamwork, and dependability [8]. Moreover, the implementation of DevOps practices improves the rate of deployments and the time taken to process change [9].

1.2 DevOps Life Cycle and Phases

Several development phases make up the DevOps lifecycle to support and enable the consistent deployment of robust software. Some of these phases may include planning and design, coding and development, integration, testing,

deployment, monitoring, and feedback. All of them are linked so they are possible to iterate through and the process may be improved more quickly [10]. The planning and design phase, in this first phase, entails the identification of requirements, architectural design, and modeling of the infrastructure. [11]. The Development Phase: Developers produce code, create new options, and work with versioning tools like Git [12]. Continuous Integration/Continuous Delivery (CI/CD) Phase: Continuous integration, building, testing, and deployment processes provide routine and consistent software updates [13]. Testing and Verification Phase: Automated testing tools are utilized in software testing to monitor software quality and identify defects early in its evolution, enhancing the efficiency and effectiveness of the testing process [11]. Deployment Phase: Applications are tested and delivered to production environments through infrastructure, including code and containerization [12]. Operations and Monitoring Phase: The application is constantly monitored and managed after deployment to enhance its performance and swiftly address problems [14]. Feedback and Improvement phase: This means that information gathered from operations is incorporated into the developmental cycles to facilitate future improvements [15]. Figure 1. Show DevOps Life Cycle.

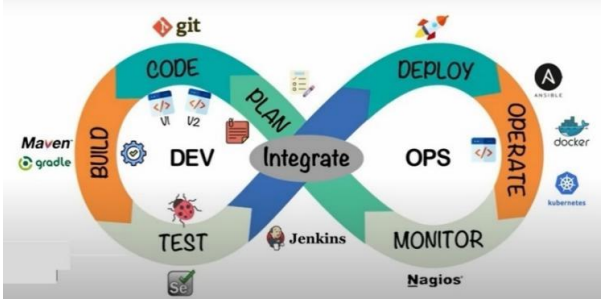

Figure 1. DevOps Life Cycle

1.3 Continuous Integration and Continuous Testing

CI (Continuous Integration) and Testing are considered to be core elements of the DevOps model improving cooperation between the development and operations departments. IT organizations that apply automated integration and testing frameworks are known to deliver value faster while at the same time producing better software. Continuous Integration (CI) is a key value in DevOps, enabling the automation of code integration by combining code contributions from multiple developers. CI is often achieved using tools like Jenkins, which automates build and test checks. CI helps detect defects early, reducing the overall cost and time needed to repair problems in later development stages [16][17]. Continuous testing is essential for real-time validation of code changes and maintaining code quality. Green Continuous Testing optimizes test cycles to minimize carbon footprints and maintains quality [18]. Integrating security testing within CI pipelines (DevSecOps) helps identify vulnerabilities early in the development lifecycle, reducing incident time and enhancing overall security [19]. .

2. LITERATURE REVIEW

The literature on Continuous Integration and Testing in DevOps highlights several challenges, including the need for a cultural shift to foster collaboration between development and operations teams, and the need to break down traditional silos for efficient Continuous Integration and Testing (CIT) processes, which can obstruct effective communication and integration [20]. Hemon et al. propose a maturity model in which it is assumed that to reach higher level integration, both functions have to be aligned and testing has to be synchronized with code development. A culture of shared responsibility and accountability is fundamental to effective DevOps practices and for that, this alignment is necessary [21]. The automation of testing processes is also a big challenge. Manual testing relies too much on people which can lead to bottlenecks and pipeline delays that can undermine the benefits of rapid deployment [22]. Jenkins is a vital automation tool to streamline workflows, reduce human intervention, and reduce the duration of software delivery cycles [22]. However, the integration of these tools into existing workflows is complex and must be carefully planned and executed [23]. Additionally, the automation process becomes even more complex because of the requirement for comprehensive test distribution and test artifact management [24]. One of the challenges encountered in Continuous Integration and Testing (CI/CT) with DevOps is related to security concerns. The integration of security practices into the CI/CD pipeline, known as DevSecOps, plays a crucial role in ensuring that potential security vulnerabilities are identified and addressed at the earliest stages of the development process [25]. However, that integration requires a culture change where security is prioritized and automated security testing tools must play seamlessly in the CI/CD framework [26]. In the literature, several best practices have been identified to address CIT challenges. The collaboration and shared responsibility between development and operations teams must become a culture. Training and workshops should be done to educate and raise the acceptance of DevOps Principles by all stakeholders [27]. Robust automation frameworks are required to be adopted for automated testing and minimized human intervention to streamline testing processes. Jenkins and GitLab CI/CD are key tools to automate the build and testing processes and speed up software delivery cycles [28]. Robust automation frameworks need to be adopted to streamline testing processes and reduce manual intervention. Automating the build and testing processes with tools like Jenkins or GitLab CI/CD helps speed up the software's delivery cycle [29]. Continuous testing practices with automated testing frameworks bring about higher software quality by assuring that code changes are tested quickly and efficiently [30]. To enhance security, organizations should incorporate security practices into their CI/CD pipelines, adopting a DevSecOps approach. This involves automating security testing and integrating security reviews into the CI/CD pipeline to identify vulnerabilities early and reduce the risk of security breaches. Constante et al. emphasize that the integrity and security of software products depend on integrating security standards into DevOps pipelines [31]. The Continuous Testing challenges are integration issues, insufficient test coverage, and slow feedback loops. Automating tests, continuous integration of codes, etc. are best practiced in Agile to promote efficiency and quality of the software [32]. However, continuous testing comes with high resources and time costs because it is required to continue testing all the time. Regression test selection (RTS) and test case prioritization (TCP), adapted for lightweight analyzes, are used best practices to efficiently manage test suite volumes [33]. Many organizations face resource limitations that hinder the implementation of continuous testing practices. In healthcare, compatibility issues with laboratory data for medication admixtures can compromise patient safety and require a rigorous testing protocol [34][35]

3. RESEARCH QUESTION

The work presented in this paper is based on the following four research questions:

RQ1: What are the possible challenges of critical continuous integration and continuous testing in DevOps?

4. RESEARCH METHODOLOGY

The proposed Systematic Literature Review (SLR) protocol aims to explore and comprehend the challenges associated with Continuous Integration (CI) and Continuous Testing (CT) in DevOps. This methodical approach involves collecting and examining existing research through a well-planned search strategy to identify pertinent studies. The review's objective is to offer a comprehensive understanding of the primary challenge's organizations encounter when implementing CI/CT within DevOps frameworks. This investigation delves into current knowledge and research gaps to emphasize the technical and organizational challenges that complicate CI/CT adoption. The research methodology employs an unstructured search of publications, allowing for a step-by-step review process that enhances the selection and analysis of relevant literature. This adaptable approach facilitates the examination of a broad range of studies across various industries and technologies. Similar methodologies have been utilized by other researchers to ensure a thorough and impartial assessment of CI/CT challenges. The outcomes of this study establish a foundation for future research and assist organizations in enhancing their DevOps practices by addressing common issues in CI/CT implementation [36] [37] [38].

4.1 SLR Data Collection

In this paper, an SLR is conducted to gather the Critical Challenges of Continuous Integration and Testing in DevOps, the step-by-step approach to conduct an SLR is shown in Figure 2.

4.2 Planning Review

Figure 2 shows that the first phase of the SLR is planning the review. In this phase, research questions are defined, data sources, search strings, criteria of literature inclusion and exclusion, and criteria of Quality assessment are determined. Inclusion criteria, exclusion criteria, and QA criteria define the literature selection process [39].

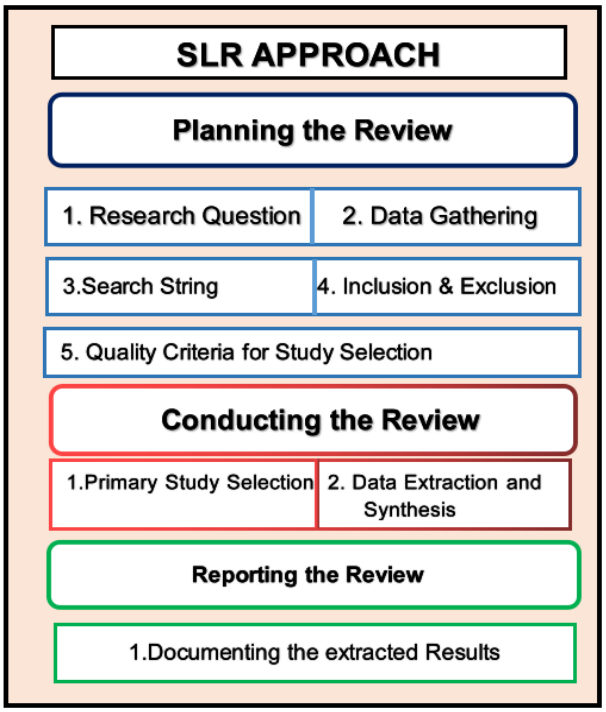

Figure. 2 SLR Approach

4.2.1 Research Question

This systematic literature review (SLR) protocol aims to identify the most important challenges of Continuous Integration (CI) and Testing within DevOps. This study identifies four research questions. The SLR is focus on RQ1, enabling a broad overview of the main issues affecting CI and Testing in DevOps. Such an approach is intended to produce valuable/insights that can maximize the applicability and usefulness of these processes in software development settings.

4.2.2 Data Gathering

This systematic literature review explores the challenges of Continuous Integration (CI) and Testing in the context of DevOps. Various data sources, such as journal articles and conference proceedings, were utilized, with key electronic databases including Wiley, ACM, ScienceDirect, and Google Scholar. Focusing on English publications from 2016 to 2024, the review aims to offer insights into the latest trends in CI and Testing methodologies within DevOps. Below Table 1 shows the Data Source.

4.2.3 Search String

The search string can produce extensive results, yet it remains within a reasonable scope. To address the study questions, relevant keywords, and their synonyms were extracted from the search string. The Boolean operator OR was employed to combine alternative meanings, while AND was used to link key elements. The complete search string is structured as follows below in Table 2.

4.2.4 Inclusion Criteria

Search strings and retrieved papers is limited by using entry criteria limiting them to a number. For final data selection, included. Here are some inclusion criteria.

- Every paper should have a source from a journal, conference, or book chapter.

- Papers on Critical Challenges of Continuous Integration and Testing in DevOps selected.
- Selected papers must analyze challenges which prevent organizations from implementing continuous integration and testing for continuous development and delivery.
- The papers should discuss the relationship between implementing continuous integration/testing in DevOps and strategies to handle such challenges. It is also for them to talk about how they impact software development processes.
- Selected Papers written in the English language.
- Research Papers with less than five pages will be excluded.
- Research Articles that appear in more than one digital library be excluded.
- Selected papers that should be available in full-text

4.2.4 Exclusion Criteria

In this research study, exclusion criteria are applied if the research publications are not relevant of concern in this research study and any literature not used in the data extraction process are irrelevant. Any article that does not meet the quality assessment criteria will be excluded.

Table 1: Data source

Electronic Databases	Wiley (www.wiley.com). ACM Digital Library(http://dl.acm.org). ScienceDirect (www.sciencedirect.com). Springer Link (link.springer.com). Google Scholar (scholar.google.com). IEEE Xplore (http://ieeexplore.ieee.org).
Search items	Journal Paper, Articles, and conference papers with full-text
Language	English
Publication period	from 2011 to 2024
Space	No bound-on space and size of papers

Table 2: Search string used in research

Keywords	**Alternatives**
DevOps	(“DevOps Overview” OR “DevOps Life Cycle” OR “Continuous Integration” OR “Continuous Development” OR “Continuous Delivery” OR “Continuous Testing” OR “Continuous Monitoring” OR “Continuous Feedback” OR “Continuous Monitoring” OR “DevOps Tools” OR “DevSecOps”) **AND**
Challenges	(“Continuous Integration Challenges” OR “Continuous testing Challenges” OR “DevOps Challenges” Continuous Integrations Risks” OR “Continuous Testing Risks” OR “Continuous Integration issues” OR “Continuous Testing barriers”) **AND**

These two tables discuss data sources and search strings.

4.2.5 Quality Assessment Criteria

To assess the quality of research papers chosen based on inclusion criteria, a straightforward checklist approach employed. Each selected paper was undergo evaluation, with "yes" or "no" responses recorded for every checklist item. A paper's score is then calculated, and only those achieving 50% or higher incorporated into the review. The scoring criteria, comprising questions labeled QAS-1 through QAS-5, are presented in Table 3. This evaluation method has been widely used by researchers, demonstrating its effectiveness in determining how well an article addresses research questions. By implementing these Quality Assessment (QA) criteria, this study seeks to enhance the quality and objectivity of the review process, minimize bias and ensure that the chosen studies effectively highlight the key challenges associated with continuous integration and testing in DevOps

Table 3: The Quality assessment criteria (QAS) of the selected studies on CI-CT in DevOps challenges

Quality Assessment Score (QAS)	Criteria
QAS. 1	Papers addressing checklist inquiries about Challenges and received a rating of "1."
QAS. 2	Papers that failed to respond to checklist inquiries were given a rating of "0."
Quality Assessment (QA) questions	**Quality assessment (QA) questions checklist**
QA. 1	Does the paper clearly outline the identification of challenges associated with Continuous Integration and Testing?
QA. 2	Are the proposed practices appropriate and relevant to the identified issues?
QA. 3	Are the study's results effectively connected to the discussed Challenges?
QA. 4	Are the Results displayed in a way that's clear and simple to get it?
QA. 5	Study how to validate or implement proposed best practices efficiently.

4.3 Conducting the Review

Conducting the review is the second phase of SLR. The activities performed in this phase include research article selection and data extraction and synthesis.

4.3.1 Primary Study Selection

In this phase, select research articles related to Continuous integration and Testing in DevOps Context. This method was used to select 2200 research articles from the selected collection through search strings. Finally, based on the inclusion, exclusion, and QA criteria set in Phase 1, 100 primary studies were selected.

4.3.2 Data Extraction and Synthesis

A structured approach to the data extraction and synthesis has been adopted to systematically assess the challenge and the gaps in Continuous Integration and Testing (CI/CT) within DevOps. The aim was identifying and analyzing existing literature, extracting information based on this, and categorizing results into main themes impacting CI/CT effectiveness. The data extraction focused on repeating problematic issues, proposed fixes, and areas for research not yet conducted. It was a detailed review of how the studies addressed automation, scalability, security, integration complexity, and testing strategy. The approach to data extraction and synthesis involves collecting important information from the articles selected to respond to these research questions. These data include:

- The date of the Article review
- Publication details (such as the title, author, and complete reference)
- Research Methods
- Location of the Publication analysis
- Publication year
- Challenges in CI/CT in DevOps
- Quality assessment of the publication based on pre-established standards.

The data extraction steps were performed by one researcher to maintain both accuracy and reliable consistency. A second researcher acted as guidance for any problems which occurred in the process. The processing phase created clear responses to research questions about Continuous Integration (CI) and Continuous Testing (CT) within DevOps methodologies through systematic analysis of data. The evaluation process delivered important patterns and essential insights about the mentioned challenges.

4.4 Reporting the Review

The third part of the SLR is Reporting the Review, which focuses on reporting the results from the selected studies. The selection of relevant papers is complete and at this stage, we are designing the SLR protocol. It is this phase, which summarizes the findings, in which we will present important key information such as publication details (titles, author names, references), sample population (company sizes, study locations), challenges involved in Continuous Integration (CI) and Continuous Testing (CT), as well as in coming out with a quality assessment of each publication. This will help us with analysis and as a basis for a more thorough SLR later, about challenges and best practices of Continuous Integration (CI) and Continuous Testing (CT) in DevOps.

The extracted data is structured that it helps understand the motivations, best practices, challenges and gaps of Continuous Integration (CI) and Continuous Testing (CT) in DevOps. This study helps to improve the CI/CT processes by identifying common problems, and exploring how to solve them. Such insights not only close up research gaps, but also provide guidance for automating, securing, and scaling more effectively. It benefits both the academia and the industry professionals in terms of making DevOps practices better for more reliable and efficient development of software.

5. PRELIMINARY RESULT

A total of 100 research papers were selected for this review after applying the predefined inclusion and exclusion criteria. 2,200 papers were systematically analyzed to identify studies most relevant to the research questions, and initially 450 were identified as most promising, to be subsequently filtered to 165 in the literature. The evaluation of each paper followed all rounds as described above, including relevance, research methodology and contribution to understanding challenges of Continuous

Integration (CI) and Continuous Testing (CT) in DevOps. This review includes only high quality and impact studies because the selection process was strict.

The paper distribution according to the different sources and the selection phases is summarized in Table 4. This table depicts the progression of the dataset from broad search to a very specific and refined subset of studies pointing directly at the research objectives. Step by step approach of different phase are also shown in figure 3 below.

Table 4 Final Paper Selected

Paper Source	*PH:*1	*PH:*2	*PH:*3	*PH:*4	*PH:*5	% (*n* = 100)
IEEE Xplore	240	180	90	70	40	60%
Wiley	200	100	60	40	10	5%
ACM Digital Library	250	200	100	30	10	5%
Springer Link	210	150	70	40	5	3%
Science Direct	300	170	80	50	15	5%
Google Scholar	1000	700	300	70	20	22%
Total	2200	1500	700	300	100	100%

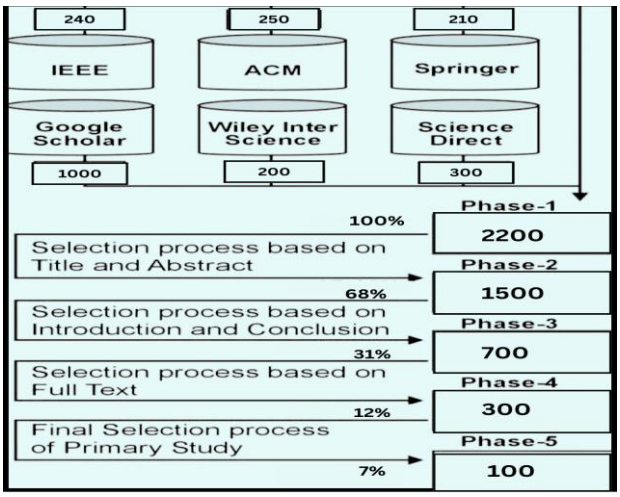

Figure. 3 SLR Approach

100 papers selected are structured to analyze recurring challenges with CI/CT. A detailed challenges table summarizes also available studies with the most reported challenges. The main issues are the automation complexity that can break CI/CT reliability, merging trouble between numerous tools and platforms, useless validation results that lead to inconsistent results, confidentiality vulnerability in automated CI/CT pipelines, protection of the ability to scale up large DevOps environments, as well as the lack of the CI/CT adoption methodologies which are standardize. There is a section with a comprehensive list of the selected papers including the challenges table 5 that summarizes the challenges in the literature in a structured form.

.

Table 5: List of initial identified challenges

Challenges
➢ CI/CT Long Wait Times for Builds/Releases [1] [35]
➢ Queue Growth Due to Resource Limitations in CI/CT [2] [3] [4]
➢ Time-Consuming Testing [11] [15] [17]
➢ Automated Flaky Tests [5][80][99]
➢ User Interface Testing Problems[6][70]

➢ Lack of Standard Tools for System Integration Testing [10][50][85]
➢ Inadequate Test Coverage[26][48][70]
➢ Complexity in Automation[20][65][86]
➢ Issues in code Integration tools[44][56][79]
➢ Security Risks in CI/CT Pipelines[10][41][72]
➢ Lack of Standardized Practices[26][60][82]
➢ Integration Bottlenecks[31][45][52]
➢ Ambiguous Test Results Ineffective communication of outcomes leads to misunderstanding.[47][62][93]
➢ Limited Skilled Resources Insufficient qualified staff obstruct successful CI/CT integration.[27][77][100]

6. CONCLUSION AND FUTURE WORK

The Systematic Literature Review (SLR) protocol was developed in order to investigate the challenges associated with Continuous Integration (CI) and Continuous Testing (CT) and to write this paper. In order to identify challenges to the successful application of CI/CT, the samples of previous research are analyzed and sorted using a defined and organized process. Lack of standard tools, issues in code integration, integration issues, intricate test automation, and security threats are among the issues. These are a few of the challenges that obstruct the quality, productivity, and adoption of DevOps processes, making it difficult for businesses to achieve smooth software development. The main goals of this study, which is still in its early phases, are to specify the research methodology, choose relevant literature, and precise important knowledge gaps. The following stage entails a thorough and methodical assessment, during which information is acquired, examined, and combined to offer more profound understanding of the issues that have been identified. Enhancing software quality, streamlining DevOps processes, and strengthening CI/CT frameworks all depend on an understanding of these problems. This study offers useful insights for experts in academia and industry by defining a precise framework for evaluating CI/CT difficulties. The results help in improving DevOps techniques, creating better ways to get past challenges, and promoting ongoing enhancements to software development and deployment procedures.

7. REFERENCES

[1]. None Suprit Pattanayak, N. P. Murthy, and N. A. Mehra, “Integrating AI into DevOps pipelines: Continuous integration, continuous delivery, and automation in infrastructural management: Projections for future,” International Journal of Science and Research Archive, vol. 13, no. 1, pp. 2244–2256, Oct. 2024, doi: https://doi.org/10.30574/ijsra.2024.13.1.1838.

[2] D. Pandey and N. Parmar, “Unleashing the Power of DevOps: Revolutionizing Software Development Practices,” International Journal of Scientific Research in Computer Science Engineering and Information Technology, vol. 10, no. 2, pp. 767–772, Apr. 2024, doi: https://doi.org/10.32628/cseit24102121.

[3] J. M. Ali, “DevOps and continuous integration/continuous deployment (CI/CD) automation,”

Adv. Eng. Innov., vol. 4, no. 1, pp. 38–42, Nov. 2023, doi: 10.54254/2977-3903/4/2023031.

[4] A. Nalamwar and P. Sirsat, “An Empirical Study on DevOps,” vol. 3, no. 11, 2019.

[5] B. Erdenebat, B. Bud, T. Batsuren, and T. Kozsik, “Multi-Project Multi-Environment Approach—An Enhancement to Existing DevOps and Continuous Integration and Continuous Deployment Tools,” Computers, vol. 12, no. 12, p. 254, Dec. 2023, doi: https://doi.org/10.3390/computers12120254.

[6] A. Qumer Gill, A. Loumish, I. Riyat, and S. Han, “DevOps for information management systems,” VINE Journal of Information and Knowledge Management Systems, vol. 48, no. 1, pp. 122–139, Feb. 2018, doi: https://doi.org/10.1108/vjikms-02-2017-0007.

[7] X. Zhou et al., “Revisit security in the era of DevOps: An evidence-based inquiry into DevSecOps industry,” IET Software, vol. 17, no. 4, pp. 435–454, Aug. 2023, doi: https://doi.org/10.1049/sfw2.12132.

[8] Harshali Rohit Kadaskar, “UNLEASHING THE POWER OF DEVOPS IN SOFTWARE DEVELOPMENT,” Int. J. Sci. Res. Mod. Sci. Technol., vol. 3, no. 3, pp. 01–07, Mar. 2024, doi: 10.59828/ijsrmst.v3i3.185.

[9] L. Prates, J. Faustino, M. Silva, and R. Pereira, “DevSecOps Metrics,” Information Systems: Research, Development, Applications, Education, pp. 77–90, 2019, doi: https://doi.org/10.1007/978-3-030-29608-7_7.

[10] 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.

[11] A. Alnafessah, A. U. Gias, R. Wang, L. Zhu, G. Casale, and A. Filieri, “Quality-Aware DevOps Research: Where Do We Stand?,” IEEE Access, vol. 9, pp. 44476–44489, 2021, doi: https://doi.org/10.1109/access.2021.3064867.

[12] D. Srivastava, M. Verma, Shashank Sheshar, and M. Gupta, “DevOps Tools: Silver Bullet for Software Industry,” Studies in computational intelligence, pp. 105–118, Dec. 2022, doi: https://doi.org/10.1007/978-981-19-6290-5_6.

[13] M. Gokarna, “DevOps phases across Software Development Lifecycle.” institute of Electrical Electronics Engineers, Jan. 06, 2021. doi: 10.36227/techrxiv.13207796.v2.

[14] Chellamalla Mamatha and S C V S L S Ravi Kiran, “Implementation of DevOps Architecture in the project development and deployment with help of tools,” International Journal of Scientific Research in Computer Science and Engineering, vol. 6, no. 2, pp. 87–95, Apr. 2018, doi: https://doi.org/10.26438/ijsrcse/v6i2.8795.

[15] A. Sen, "DevOps, DevSecOps, AIOPS- Paradigms to IT Operations," springer singapore, 2020, pp. 211–221. doi: 10.1007/978-981-15-7804-5_16.

[16] E. Naresh, S. V. N. Murthy, N. Sreenivasa, S. Merikapudi, and C. R. Rakhi Krishna, "Continuous Integration, Testing Deployment and Delivery in Devops," 2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS), pp. 1–4, Apr. 2024, doi: https://doi.org/10.1109/ickecs61492.2024.10616918.

[17] P. G. Gowda, Stanley, and J. E. Joyce, "DevOps Dynamics: Tools Driving Continuous Integration and Deployment," pp. 1–7, Jun. 2024, doi: https://doi.org/10.1109/iciteics61368.2024.10624986.

[18] K. Nayak, S. Route, M. Sundararajan, A. Jain, and Shashidhar R, "Sustainable Continuous Testing in DevOps Pipeline," May 2024, doi: https://doi.org/10.1109/incccs60947.2024.10593566.

[19] C. Feio, N. Santos, N. Escravana, and B. Pacheco, "An Empirical Study of DevSecOps Focused on Continuous Security Testing," doi: https://doi.org/10.1109/EuroSPW61312.2024.00074.

[20] "Compliance and Audit Challenges in DevOps: A Security Perspective," International Research Journal of Modernization in Engineering Technology and Science, Oct. 2023, doi: https://doi.org/10.56726/irjmets45309.

[21] A. Hemon, B. Lyonnet, F. Rowe, and B. Fitzgerald, "Conceptualizing the Transition from Agile to DevOps: A Maturity Model for a Smarter IS Function," inria.hal.science, Jun. 25, 2018.

[22] B. Erdenebat, B. Bud, T. Batsuren, and T. Kozsik, "Multi-Project Multi-Environment Approach—An Enhancement to Existing DevOps and Continuous Integration and Continuous Deployment Tools," Computers, vol. 12, no. 12, p. 254, Dec. 2023, doi: https://doi.org/10.3390/computers12120254.

[23] J. Zulkarnain, R. F. Mulya, T. Pratiwi, W. Pangestuti, and F. I. Ilmawati, "DevOps Main Area and Core Capabilities Adopting DevOps in the Last Decade: A Systematic Literature Review," International Journal of Research and Applied Technology (INJURATECH), vol. 2, no. 2, pp. 184–197, Nov. 2022, doi: https://doi.org/10.34010/injuratech.v2i2.8364.

[24] D. A. Meedeniya, I. D., and I. Perera, "Traceability Establishment and Visualization of Software Artefacts in DevOps Practice: A Survey," International Journal of Advanced Computer Science and Applications, vol. 10, no. 7, 2019, doi: https://doi.org/10.14569/ijacsa.2019.0100711.

[25] "Compliance and Audit Challenges in DevOps: A Security Perspective," International Research Journal of Modernization in Engineering Technology and Science, Oct. 2023, doi: https://doi.org/10.56726/irjmets45309.

[26] Dinesh Reddy Chittibala, "Web Application Firewalls (WAF) Integration in DevOps Practices: A Scholarly Exploration of Security, Automation and Continuous Protection," pp. 1–5, Dec. 2022, doi: https://doi.org/10.47363/jaicc/2022(1)210.

[27] F. M. A. Erich, C. Amrit, and M. Daneva, "A qualitative study of DevOps usage in practice," Journal of Software: Evolution and Process, vol. 29, no. 6, p. e1885, Jun. 2017, doi: https://doi.org/10.1002/smr.1885.

[29] C. Vassallo, S. Proksch, H. C. Gall, and Massimiliano Di Penta, "Automated Reporting of Anti-Patterns and Decay in Continuous Integration," May 2019, doi: https://doi.org/10.1109/icse.2019.00028.

[30] Y. Li, J. Wang, Q. Wang, and J. Hu, "A Class-level Test Selection Approach Toward Full Coverage For Continuous Integration," Proceedings/Proceedings of the ... International Conference on Software Engineering and Knowledge Engineering, vol. 2019, pp. 49–54, Jul. 2019, doi: https://doi.org/10.18293/seke2019-011.

[31] F. Moyón, R. Soares, M. Pinto-Albuquerque, D. Mendez, and K. Beckers, "Integration of Security Standards in DevOps Pipelines: An Industry Case Study," Product-Focused Software Process Improvement, pp. 434–452, 2020, doi: https://doi.org/10.1007/978-3-030-64148-1_27.

[32] Amit, Bhanushali. (2023). 2. Challenges and Solutions in Implementing Continuous Integration and Continuous Testing for Agile Quality Assurance. International journal of science and research, doi: 10.21275/sr231021114758

[33] G. Rothermel, "Improving regression testing in continuous integration development environments (keynote)," Oct. 2018, doi: https://doi.org/10.1145/3278186.3281454.

[34] J. Sosnowski, J. Baker, O. Arnold, M. Silva, and M. West, "Reflections on 10 years of Operating a Computer-based Testing Facility: Lessons Learned, Best Practices," 2024 ASEE Annual Conference & Exposition, Jun. 2024, doi: https://doi.org/10.18260/1-2--47930.

[35] M. Cheok, N. Michael, and R. Wojnar, "Compatibility of medication admixtures in continuous subcutaneous infusions: prioritizing laboratory testing for common combinations," International Journal of Pharmacy Practice, vol. 32, no. 5, Aug. 2024, doi: https://doi.org/10.1093/ijpp/riae039.

[36] M. Yaseen, Samad Baseer, and S. Sherin, "Critical challenges for requirement implementation in context of global software development: A systematic literature review," Dec. 2015, doi: https://doi.org/10.1109/icosst.2015.7396413.

[37] M. Yaseen, S. Ali, Abullah, and N. Ullah, "An Improved Framework for Requirement Implementation in the context of Global Software Development: A Systematic Literature Review Protocol," International Journal of Database Theory and Application, vol. 9, no. 6, pp. 161–170, Jun. 2016, doi: https://doi.org/10.14257/ijdta.2016.9.6.16.

[38] M. Yaseen, "Exploratory study of existing research on software requirements prioritization: A systematic literature review," Journal of Software Evolution and Process, vol. 36, no. 6, Sep. 2023, doi: https://doi.org/10.1002/smr.2613.

[39].M. A. Akbar, A. Alsanad, S. Mahmood, A. A. Alsanad, and A. Gumaei, "A Systematic Study to Improve the Requirements Engineering Process in the Domain of Global Software Development," IEEE Access, vol. 8, pp. 53374–53393, 2020, doi: https://doi.org/10.1109/access.2020.2979468.

8. APPENDIX

[1] Vidroha Debroy, S. Miller, and L. Brimble, "Building lean continuous integration and delivery pipelines by applying DevOps principles: a case study at Varidesk," Foundations of Software Engineering, Oct. 2018, doi: https://doi.org/10.1145/3236024.3275528.

[2] N. Azad, "Understanding DevOps critical success factors and organizational practices," IEEE Xplore, May 01, 2022. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9808814

[3] A. K. Patel, "Streamlining Development: Best Practices for Salesforce DevOps and Continuous Integration," Journal of Mathematical & Computer Applications, pp. 1–4, Feb. 2024, doi: https://doi.org/10.47363/jmca/2024(3)e110.

[4] Kavita Dhakad, "Adopting Continuous Integration Practices to Achieve Quality in DevOps," International Journal of Advanced Research in Science, Communication and Technology, pp. 101–119, Feb. 2023, doi: https://doi.org/10.48175/ijarsct-8368.

[5] K. Nayak, S. Route, M. Sundararajan, A. Jain, and Shashidhar R, "Sustainable Continuous Testing in DevOps Pipeline," May 2024, doi: https://doi.org/10.1109/incccs60947.2024.10593566.

[6] Bipin Balu Shinde, "Automated Testing in DevOps: Strategies and Tools," International Journal of Advanced Research in Science, Communication and Technology, pp. 550–554, Jun. 2024, doi: https://doi.org/10.48175/ijarsct-19074.

[7] Josh Mahmood Ali, "DevOps and continuous integration/continuous deployment (CI/CD) automation," Advances in engineering innovation, vol. 4, no. 1, pp. 38–42, Nov. 2023, doi: https://doi.org/10.54254/2977-3903/4/2023031.

[8] S. Banala, "DevOps Essentials: Key Practices for Continuous Integration and Continuous Delivery," International Numeric Journal of Machine Learning and Robots, vol. 8, no. 8, pp. 1–14, Jan. 2024, Available: https://injmr.com/index.php/fewfewf/article/view/83

[9] J. A. Tsapa, "Integrating DevOps with Agile and other Software Development Methodologies," vol. 3, no. 3, p. 5, Jul. 2022, Available: https://www.researchgate.net/publication/382852396_Integrating_DevOps_with_Agile_and_other_Software_Development_Methodolog

[10] R. Eramo, Gian Luca Scoccia, M. Nolletti, A. Celi, and M. Autili, "An Empirical Study on the Role of Devops in the Development of Mobile Applications," Jan. 2024, doi: https://doi.org/10.2139/ssrn.4719199.

[11] M. Senapathi, J. Buchan, and H. Osman, "DevOps Capabilities, Practices, and Challenges," Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018 - EASE'18, 2018, doi: https://doi.org/10.1145/3210459.3210465.

[12] M. Shahin, A. Rezaei Nasab, and M. Ali Babar, "A qualitative study of architectural design issues in DevOps," Journal of Software: Evolution and Process, Sep. 2021, doi: https://doi.org/10.1002/smr.2379.

[13] M. Fernandes, S. Ferino, A. Fernandes, U. Kulesza, E. Aranha, and C. Treude, "DevOps education," Proceedings of the ACM/IEEE 44th International Conference on Software Engineering: Software Engineering Education and Training, May 2022, doi: https://doi.org/10.1145/3510456.3514152.

[14] K. Kuusinen et al., "A Large Agile Organization on Its Journey Towards DevOps," IEEE Xplore, 2018. https://ieeexplore.ieee.org/abstract/document/8498186 (accessed May 10, 2021).

[15] N. A. Jo□ã, o Faustino, R. Pereira, B. Alturas, and M. M. D. Silva, "Agile information technology service management with DevOps: an incident management case study," International Journal of Agile Systems and Management, vol. 13, no. 4, p. 339, 2020, doi: https://doi.org/10.1504/ijasm.2020.112331.

[16] N. Jamal, "A Review Paper on: Benefits and Challenges of DevOps Developer and Modern Era," Scandic Journal Of Advanced Research And Reviews, vol. 2, no. 2, pp. 068–080, May 2022, doi: https://doi.org/10.55966/sjarr.2022.2.2.0035.

[17] A. Miller, R. Giachetti, and D. Van Bossuyt, "Challenges of Adopting DevOps for Combat Systems Development Environment," Defense Acquisition Research Journal, vol. 29, no. 99, pp. 22–48, Jan. 2022, doi: https://doi.org/10.22594/dau.21-870.29.01.

[18] N. Azad, "Developing a Critical Success Factor Model for DevOps," Lecture notes in computer science, pp. 662–668, Jan. 2022, doi: https://doi.org/10.1007/978-3-031-21388-5_53.

[19] Sara and Denise Hideko Goya, "Major Challenges of Systems-of-Systems with Cloud and DevOps – A Financial Experience Report," May 2019, doi: https://doi.org/10.1109/sesos/wdes.2019.00010.

[20] R. Gupta, M. Venkatachalapathy, and F. K. Jeberla, "Challenges in Adopting Continuous Delivery and DevOps in a Globally Distributed Product Team: A Case Study of a Healthcare Organization," 2019 ACM/IEEE 14th International Conference on Global Software Engineering (ICGSE), 2019, doi: https://doi.org/10.1109/ICGSE.2019.00020.

[21] A. Bijwe and P. Shankar, "Challenges of Adopting DevOps Culture on the Internet of Things Applications - A Solution Model," IEEE Xplore, Oct. 01, 2022. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9988182 (accessed Mar. 13, 2023).

[22] E. Diel, S. Marczak, and D. S. Cruzes, "Communication Challenges and Strategies in Distributed DevOps," 2016 IEEE 11th International Conference on Global Software Engineering (ICGSE), Aug. 2016, doi: https://doi.org/10.1109/icgse.2016.28.

[23] M. Chen et al., "Critical Challenges and Solutions for an Ultra-Large-Scale Enterprise DevOps Platform," 2022 7th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA), Apr. 2022, doi: https://doi.org/10.1109/icccbda55098.2022.9778937.

[24] B. Jambunathan and D. Y. Kalpana, "Design of devops solution for managing multi cloud distributed environment," International Journal of Engineering & Technology, vol. 7, no. 3.3, p. 637, Jun. 2018, doi: https://doi.org/10.14419/ijet.v7i2.33.14854.

[25] A. Trigo, J. Varajão, and L. Sousa, "DevOps adoption: Insights from a large European Telco," Cogent Engineering, vol. 9, no. 1, Jun. 2022, doi: https://doi.org/10.1080/23311916.2022.2083474.

[26] I. A. Mohammed, "An Empirical Study Of The Importance Of Devops Strategies And Approaches In Information Management Systems," Ssrn.com, Mar. 2015. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4686579 (accessed Nov. 09, 2024).

[27] Journal of Software Engineering & Intelligent Systems, "DEVOPS APPROACH AND LEAN THINKING IN AGILE SOFTWARE DEVELOPMENT: OPPORTUNITIES, ADVANTAGES, AND CHALLENGES," Journal of Software Engineering & Intelligent Systems, 2020, doi: https://doi.org/102085432/45317716/35378599/s200_journal_of_software_engineering.

[28] L. E. Lwakatare, I. Crnkovic, and J. Bosch, "DevOps for AI – Challenges in Development of AI-enabled Applications," IEEE Xplore, Sep. 01, 2020. https://ieeexplore.ieee.org/document/9238323 (accessed Oct. 29, 2021).

[29] S. imran Abbas and M. Singh, "DevOps for Edge Computing: Challenges and Solutions," Jun. 2023, doi: https://doi.org/10.1109/icpcsn58827.2023.00213.

[30] M. Stoica and Alexandru Ionuț Nițu, "Efficiency and Cost-Effectiveness in Agile DevOps with Cloud Computing," Proceedings of the ... International Conference on Business Excellence, vol. 18, no. 1, pp. 3543–3556, Jun. 2024, doi: https://doi.org/10.2478/picbe-2024-0287.

[31] Akalanka Nayanajith and Ruwan Wickramarachchi, "Exploring the challenges affecting the successful adoption of DevOps practices in Sri Lanka," vol. 10, pp. 1–6, Apr. 2024, doi: https://doi.org/10.1109/scse61872.2024.10550479.

[32] M. O. Khan, "Fast Delivery, Continuously Build, Testing and Deployment with DevOps Pipeline Techniques on Cloud," Indian Journal of Science and Technology, vol. 13, no. 5, pp. 552–575, Feb. 2020, doi: https://doi.org/10.17485/ijst/2020/v13i05/148983.

[33] Q. Mi, Q. Bao, and L. Cui, "Identifying Topics and Trends in DevOps: A Study of Stack Overflow Posts," pp. 402–409, Sep. 2023, doi: https://doi.org/10.1109/seaa60479.2023.00067.

[34] P. Batra and A. Jatain, "Hybrid model for evaluation of quality aware DevOps," International Journal of Applied Science and Engineering, vol. 18, no. 5, pp. 1–11, 2021, doi: https://doi.org/10.6703/ijase.202106_18(2).013.

[35] Z. Ahmed and Shoba. C. Francis, "Integrating Security with DevSecOps: Techniques and Challenges," IEEE Xplore, Nov. 01, 2019. https://ieeexplore.ieee.org/document/9105789 (accessed Jul. 12, 2022).

[36] T. Rangnau, R. v. Buijtenen, F. Fransen, and F. Turkmen, "Continuous Security Testing: A Case Study on Integrating Dynamic Security Testing Tools in CI/CD Pipelines," 2020 IEEE 24th International Enterprise Distributed Object Computing Conference (EDOC), Oct. 2020, doi: https://doi.org/10.1109/edoc49727.2020.00026.

[37] J. De Kock and J. Ophoff, "Critical Success Factors for Integrating Security into a DevOps Environment." Available: https://ifip.byu.edu/00000188-e1b8-d3db-afbf-e3bd83ff0000/drw-2023-paper-17

[38] S. S. Sravan, C. Sai Ganesh, K. V. D. Kiran, T. Aakash Chandra, K. Aparna, and T. Vignesh, "Significant Challenges to espouse DevOps Culture in Software Organisations By AWS : A methodical Review," IEEE Xplore, Mar. 01, 2023. https://ieeexplore.ieee.org/abstract/document/10113021/ (accessed May 30, 2023).

[39] P. Abrahamsson et al., "Towards a Secure DevOps Approach for Cyber-Physical Systems," International Journal of Systems and Software Security and Protection, vol. 11, no. 2, pp. 38–57, Jul. 2020, doi: https://doi.org/10.4018/ijsssp.2020070103.

[40] None Harshali Rohit Kadaskar, "UNLEASHING THE POWER OF DEVOPS IN SOFTWARE DEVELOPMENT," International Journal of Scientific Research in Modern Science and Technology, vol. 3, no. 3, pp. 01–07, Mar. 2024, doi: https://doi.org/10.59828/ijsrmst.v3i3.185.

[41] A. O. Mansour and M. R. Jameel Qureshi, "Proposal To Cope Change Resistance Using DevOps," International Journal of Computer Science and Mobile Computing, vol. 9, no. 9, pp. 43–49, Sep. 2020, doi: https://doi.org/10.47760/ijcsmc.2020.v09i09.005.

[42] Sarthak Srivastav, K. Allam, and Anirudh Mustyala, "Software Automation Enhancement through the Implementation of DevOps," International Journal of Research Publication and Reviews, vol. 4, no. 6, pp. 2050–2054, Jun. 2023, doi: https://doi.org/10.55248/gengpi.4.623.45947.

[43] Chellamalla Mamatha and S C V S L S Ravi Kiran, "Implementation of DevOps Architecture in the project development and deployment with help of tools," International Journal of Scientific Research in Computer Science and Engineering, vol. 6, no. 2, pp. 87–95, Apr. 2018, doi: https://doi.org/10.26438/ijsrcse/v6i2.8795.

[44] A. Wiedemann, "IT Governance Mechanisms for DevOps Teams - How Incumbent Companies Achieve Competitive Advantages," Hawaii International Conference on System Sciences, 2018. https://www.semanticscholar.org/paper/IT-Governance-Mechanisms-for-DevOps-Teams-How-Wiedemann/e335d41173a9b4906e425c089c8ed3746bbefd6b (accessed Nov. 09, 2024).

[45] H. Fahmy, S. Samir, and M. Nasr, "IMPROVING INTEGRITY, SECURITY, AND ACCURACY DURING DEVOPS PROCESS," vol. 101, no. 13, 2023, Accessed: Nov. 09, 2024. [Online]. Available: http://www.jatit.org/volumes/Vol101No13/12Vol101No13.pdf

[46] S. Ranjan, "Empowering DevOps with Infrastructure as Code :Trends, Tools and Techniques," INTERANTIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT, vol. 08, no. 06, pp. 1–5, Jun. 2024, doi: https://doi.org/10.55041/ijsrem35407.

[47] N. Vemuri, "Integrating Deep Learning with DevOps for Enhanced Predictive Maintenance in the Manufacturing Industry," Tuijin Jishu/Journal of Propulsion Technology, vol. 43, no. 4, pp. 315–322, Jul. 2023, doi: https://doi.org/10.52783/tjjpt.v43.i4.5041.

[48] L. Leite, C. Rocha, F. Kon, D. Milojicic, and P. Meirelles, "A Survey of DevOps Concepts and Challenges," ACM

Computing Surveys, vol. 52, no. 6, pp. 1–35, Jan. 2020, doi: https://doi.org/10.1145/3359981.

[49] Z. Wang, M. Shi, and C. Li, "An Intelligent DevOps Platform Research and Design Based on Machine Learning," Dec. 2020, doi: https://doi.org/10.1109/cbd51900.2020.00017.

[50] A. M. Salih, S. M. Syed-Mohamad, P. Keikhosrokiani, and N. H. Samsudin, "Adopting DevOps practices: an enhanced unified theory of acceptance and use of technology framework," International Journal of Electrical and Computer Engineering (IJECE), vol. 13, no. 6, pp. 6701–6717, Dec. 2023, doi: https://doi.org/10.11591/ijece.v13i6.pp6701-6717.

[51] Dinesh Reddy Chittibala, "Web Application Firewalls (WAF) Integration in DevOps Practices: A Scholarly Exploration of Security, Automation and Continuous Protection," pp. 1–5, Dec. 2022, doi: https://doi.org/10.47363/jaicc/2022(1)210.

[52] M. S. Sushmitha, "Setting Up CICD Pipeline For Web Development Project in Cloud," International Journal for Research in Engineering Application & Management (IJREAM), vol. 05, pp. 2454–9150, 2019, doi: https://doi.org/10.35291/2454-9150.2019.0091.

[53] J. Media and Budidarma, "DevOps Implementation with Enterprise On-Premise Infrastructure," vol. 7, no. 1, 2023, doi: https://doi.org/10.30865/mib.v7i1.5500.

[54] C. Feio, N. Santos, N. Escravana, and B. Pacheco, "An Empirical Study of DevSecOps Focused on Continuous Security Testing," doi:https://doi.org/10.1109/EuroSPW61312.2024.00074

[55] A. Bhanushali, "Challenges and solutions in implementing continuous integration and continuous testing for agile quality assurance," International journal of science and research, vol. 12, no. 10, pp. 1626–1644, Oct. 2023, doi: https://doi.org/10.21275/sr231021114758.

[56] A. Ahmad, F. Gomes, Eduard Paul Enoiu, K. Sandahl, and O. Leifler, "An Industrial Study on the Challenges and Effects of Diversity-Based Testing in Continuous Integration," 2023 IEEE 23rd International Conference on Software Quality, Reliability, and Security (QRS), pp. 337–347, Oct. 2023, doi: https://doi.org/10.1109/QRS60937.2023.00041.

[57] S. Demeyer, B. Verhaeghe, A. Etien, N. Anquetil, and S. Ducasse, "Evaluating the efficiency of continuous testing during test-driven development," IEEE Xplore, Mar. 01, 2018. https://ieeexplore.ieee.org/abstract/document/8327152 (accessed Dec. 16, 2021).

[58] A. Debbiche, Mikael Dienér, and Richard Berntsson Svensson, "Challenges When Adopting Continuous Integration: A Case Study," Lecture Notes in Computer Science, pp. 17–32, Dec. 2014, doi: https://doi.org/10.1007/978-3-319-13835-0_2.

[59] T. Gamblin and D. S. Katz, "Overcoming Challenges to Continuous Integration in HPC," Computing in Science & Engineering, vol. 24, no. 6, pp. 54–59, Nov. 2022, doi: https://doi.org/10.1109/mcse.2023.3263458.

[60] O. Elazhary, C. Werner, Z. S. Li, D. Lowlind, N. A. Ernst, and M.-A. Storey, "Uncovering the Benefits and Challenges of Continuous Integration Practices," IEEE Transactions on Software Engineering, pp. 1–1, 2021, doi: https://doi.org/10.1109/tse.2021.3064953.

[61] F. Zampetti, V. Nardone, and M. Di Penta, "Problems and solutions in applying continuous integration and delivery to 20 open-source cyber-physical systems," Proceedings of the 19th International Conference on Mining Software Repositories, May 2022, doi: https://doi.org/10.1145/3524842.3527948.

[62] C. Vassallo, "Enabling Continuous Improvement of a Continuous Integration Process," IEEE Xplore, Nov. 01, 2019. https://ieeexplore.ieee.org/document/8952505

[63] R. Jongeling, J. Carlson, and A. Cicchetti, "Impediments to Introducing Continuous Integration for Model-Based Development in Industry," DiVA (Mälardalen University College), Aug. 2019, doi: https://doi.org/10.1109/seaa.2019.00071.

[64] G. Pinto, F. Castor, R. Bonifacio, and M. Rebouças, "Work practices and challenges in continuous integration: A survey with Travis CI users," Software: Practice and Experience, vol. 48, no. 12, pp. 2223–2236, Sep. 2018, doi: https://doi.org/10.1002/spe.2637.

[65] Z. Xie and M. Li, "Cutting the Software Building Efforts in Continuous Integration by Semi-Supervised Online AUC Optimization," Twenty-Seventh International Joint Conference on Artificial Intelligence {IJCAI-18}, pp. 2875–2881, Jul. 2018, doi: https://doi.org/10.24963/ijcai.2018/399.

[66] K. Nayak, S. Route, M. Sundararajan, A. Jain, and Shashidhar R, "Sustainable Continuous Testing in DevOps Pipeline," May 2024, doi: https://doi.org/10.1109/incccs60947.2024.10593566.

[67] T. Rangnau, R. v. Buijtenen, F. Fransen, and F. Turkmen, "Continuous Security Testing: A Case Study on Integrating Dynamic Security Testing Tools in CI/CD Pipelines," 2020 IEEE 24th International Enterprise Distributed Object Computing Conference (EDOC), Oct. 2020, doi: https://doi.org/10.1109/edoc49727.2020.00026.

[68] S. Nithin, H. S. J, and Parvathy S, "Enhancing the Automotive Software Test Environment Using Continuous Integration and Validation Pipeline," Dec. 2023, doi: https://doi.org/10.1109/i-pact58649.2023.10434454.

[69] A. Ahmad, F. Gomes, Eduard Paul Enoiu, K. Sandahl, and O. Leifler, "An Industrial Study on the Challenges and Effects of Diversity-Based Testing in Continuous Integration," 2023 IEEE 23rd International Conference on Software Quality, Reliability, and Security (QRS), pp. 337–347, Oct. 2023, doi: https://doi.org/10.1109/QRS60937.2023.00041.

[70] Eva Hadler Vihovde and Q. Meng, "Test-Driven Development: Ensuring Code Quality in Integration with CI/CD," Jan. 2024, doi: https://doi.org/10.1109/icmss61211.2024.00009.

[71] A. Deshpande, S. V. Veenadevi, and Suresh Aleti, "Test Automation and Continuous Integration using Jenkins for Smart Card OS," Jul. 2021, doi: https://doi.org/10.1109/icccnt51525.2021.9580021.

[72] L. Huang, H. Liu, Y. Liu, Y. Shang, and Z. Li, "A Generative Adversarial Imitation Learning Method for Continuous Integration Testing," Mar. 2024, doi: https://doi.org/10.1109/ainit61980.2024.10581812.

[73] P. K. Sinha, P. Nand, S. S. Chauhan, and S. Tiwari, "Study of Failures in Continuous Integration Environment," 2024 International Conference on Electrical Electronics and Computing Technologies (ICEECT), pp. 1–6, Aug. 2024, doi: https://doi.org/10.1109/iceect61758.2024.10739077.

[74] M. A. Mascheroni, E. Irrazábal, and G. Rossi, "Continuous Testing Improvement Model," IEEE Xplore, May 01, 2021. https://ieeexplore.ieee.org/document/9462968 (accessed Apr. 16, 2023).

[75] E. Naresh, S. V. N. Murthy, N. Sreenivasa, S. Merikapudi, and C. R. Rakhi Krishna, "Continuous Integration, Testing Deployment and Delivery in Devops," 2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS), pp. 1–4, Apr. 2024, doi: https://doi.org/10.1109/ickecs61492.2024.10616918.

[76] P. G. Gowda, Stanley, and J. E. Joyce, "DevOps Dynamics: Tools Driving Continuous Integration and Deployment," pp. 1–7, Jun. 2024, doi: https://doi.org/10.1109/iciteics61368.2024.10624986.

[78] C. Feio, N. Santos, N. Escravana, and B. Pacheco, "An Empirical Study of DevSecOps Focused on Continuous Security Testing," doi: https://doi.org/10.1109/EuroSPW61312.2024.00074.

[79]"Compliance and Audit Challenges in DevOps: A Security Perspective," International Research Journal of Modernization in Engineering Technology and Science, Oct. 2023, doi: https://doi.org/10.56726/irjmets45309.

[80] A. Hemon, B. Lyonnet, F. Rowe, and B. Fitzgerald, "Conceptualizing the Transition from Agile to DevOps: A Maturity Model for a Smarter IS Function," inria.hal.science, Jun. 25, 2018.

[81] D. Eswararaj, K. L. Rao, and B. R. Sekhar, "DevOps Implementation: Essential Tools, Best Practices, and Solutions to Overcome Challenges for Seamless Development and Operations Integration - Archive Paper Guardians," Articleproms.com, Oct. 2024.

[82] Liang, P., Wu, Y., Xu, Z., Xiao, S., & Yuan, J. (2024). Enhancing Security in DevOps by Integrating Artificial Intelligence and Machine Learning. Journal of Theory and Practice of Engineering Science, 4(02), 31-37.

[83] Sharma, P. (2024). DevSecOps Integration-Security in the Software Delivery Pipeline: Exploring the integration of security practices into the software delivery pipeline to ensure secure software development practices. Australian Journal of Machine Learning Research & Applications, 4(1), 46-54.

[84] Cholli, N. G. (2024). Integrating Agile and DevOps: A Comprehensive Review of DevOps-Enabled Agile in Modern Software Development.

[85] Mbuta, I. D. A. Software testing optimization and continuous integration with MS AZURE DevOps.

[86]M. Moeez et al., "Comprehensive Analysis of DevOps: Integration, Automation, Collaboration, and Continuous Delivery," Bulletin of Business and Economics (BBE), vol. 13, no. 1, Mar. 2024, doi: https://doi.org/10.61506/01.00253.

[87] Paul, C. Best Practices for Implementing Test-Driven Development (TDD) and Behavior-Driven Development (BDD) in Continuous Deployment Environments.

[88] Md Sarazul Ali and D. Puri, "Optimizing DevOps Methodologies with the Integration of Artificial Intelligence," Mar. 2024, doi: https://doi.org/10.1109/inocon60754.2024.10511490.

[89] M. Saleem, F. Saleem, F. Arif, A. B. Tariq, M. A. Riaz, and N. Abbas, "Evaluating Software Quality in DevOps Practices in Pakistan: A Survey Approach," vol. 34, pp. 183–189, Nov. 2023, doi: https://doi.org/10.1109/icet59753.2023.10374716.

[90] "Learning-to-Rank vs Ranking-to-Learn: Strategies for Regression Testing in Continuous Integration," Ieee.org, 2020.

[91] P. G. Gowda, Stanley, and J. E. Joyce, "DevOps Dynamics: Tools Driving Continuous Integration and Deployment," pp. 1–7, Jun. 2024, doi: https://doi.org/10.1109/iciteics61368.2024.10624986.

[92] K. Charan, Nadipineni Karthikeya, Pujala Bhogeswara Narasimharao, S. A. Devi, Vaka Abhilash, and S. Srinivas, "Effective Code Testing Strategies in DevOps: A Comprehensive Study of Techniques and Tools for Ensuring Code Quality and Reliability," vol. 67, pp. 302–309, Jul. 2023, doi: https://doi.org/10.1109/icesc57686.2023.10193275.

[93] E. A. Da, J. A. Prado, R. C. Silva, and Sílvia Regina Vergílio, "Machine Learning Regression Techniques for Test Case Prioritization in Continuous Integration Environment," 2022 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER), Mar. 2022, doi: https://doi.org/10.1109/saner53432.2022.00034.

[94] F. Leinen, D. Elsner, A. Pretschner, A. Stahlbauer, M. Sailer, and Elmar Jürgens, "Cost of Flaky Tests in Continuous Integration: An Industrial Case Study," pp. 329–340, May 2024, doi: https://doi.org/10.1109/icst60714.2024.00037.

[95] Pandya, K. (2024). Automated Software Compliance Using Smart Contracts and Large Language Models in Continuous Integration and Continuous Deployment With DevSecOps (Master's thesis, Arizona State University).

[96] Pando, B., Silva, A., & Dávila, A. (2024). DevOps Adoption: A Tertiary Study. In New Perspectives in Software Engineering (pp. 39-52). Cham: Springer Nature Switzerland.

[97] C. Feio, N. Santos, N. Escravana, and B. Pacheco, "An Empirical Study of DevSecOps Focused on Continuous Security Testing," doi: https://doi.org/10.1109/EuroSPW61312.2024.00074

[98] Ayyash, M. A. I. A. (2024). Implementing Agile and DevOps at Scale: Identifying Best Frameworks, Practices, and Success Factors (Doctoral dissertation, Al-Quds University).

[99] N. Azad and Sami Hyrynsalmi, "Navigating Challenges and Pitfalls in DevOps-Driven Digital Transformation," Lecture notes in computer science, pp. 115–128, Jan. 2024, doi: https://doi.org/10.1007/978-3-031-72234-9_10.

[100] A. Gupta, "A Case Study: DevOps Transformation in Mobile Application Products: Journey, Challenges, and Solutions," Journal of Artificial Intelligence, Machine Learning and Data Science, vol. 1, no. 1, pp. 655–660, Mar. 2023, doi: https://doi.org/10.51219/jaimld/amit-gupta/167.