

CHALLENGES & OPPORTUNITIES OF ADOPTING DEVOPS IN THE PMI CONTEXT

Mr. Zakriya Khan, Ms Laiba Ali, Supervisor Saleem Zubair
Superior University: MS batch 2025

Abstract

*Adopting DevOps within the Project Management Institute (PMI) framework presents both significant challenges and substantial opportunities. We examine the cultural, technical, and system-centric DevOps adoption (and implementation) challenges organizations face when trying to integrate PMI realms of organizational/operational context in this paper. These challenges are centered around the culture change to enable better collaboration, fitting in missing pieces surrounding automation and continuous integration skills, and incorporating different tools together seamlessly under one system featuring a mix of agile DevOps methodologies with traditional PMI processes while not alienating security amidst repeat deployments. And despite all the challenges, DevOps adoption contains transformative capabilities. These will help organizations to work better from all directions in some of these ways:
As organizations tackle these challenges and exploit the opportunities, they will be much more successful at delivering all projects to a higher standard. Using Present academic underpinnings and contemporary enterprise DevOps transformation case studies, the current analysis aims to provide a complete view of the commonly understood landscape in terms of adopting DevOps within PMI. Offering pragmatic guidance for those who need to integrate DevOps practices with PMI Project Management, the guide is designed to increase project success and improve innovation around project management.*

I. INTRODUCTION

In the contemporary landscape of software development and IT operations, the adoption of DevOps practices has emerged as a transformative force, fundamentally reshaping how organizations approach the creation, delivery, and management of software. DevOps, a portmanteau of "Development" and "Operations," encapsulates a set of principles and practices aimed at bridging the traditional gap between software development teams and IT operations teams. Developers work with operators to enhance the speed, quality, and efficiency of the development lifecycle by introducing improved collaboration automation and continuous improvement.

For years, the Project Management Institute (PMI) has been a guiding light in what type of practices should be considered best practices within project management. PMI's procedures, such as PMBOK (Project Management Body of Knowledge), provide a structured system for managing projects with distinct processes and lots of documents that focus on phases and milestones within the project. These traditional project management practices have been successfully used in complex projects across various industries, but its application to the fast pace and ever-changing requirement of modern software development often meets limitations.

The incorporation of DevOps into PMI frameworks could offer an attractive solution to the aforementioned constraints. This quite well fits into the need of modern software development which requires rapid iteration and feedback loops, both things that DevOps introduces by means of CI/CD (Continuous Integration / Continuous Delivery) practices as well as automation. These DevOps practices enable organizations to potentially improve cooperation between development and operations teams, reduce work efficiency timeframes, and deliver high-quality software more efficiently.

Moving from traditional PMI to a DevOps-driven approach faces scores of problems. Key challenges to overcome include the necessary culture change associated with adopting DevOps best practices. The typical PMI environment is highly structured as in hierarchical

and maintains a strong distinction of roles often leading to the creation of silos hence hindering seamless integration between developers and operators. DevOps, in contrast, embodies a culture of collective accountability and cooperation. Behaving like this is a big cultural change, and in most organizations that are completely stuck, it takes some doing to really make the culture shift.

Additionally, the tech side of DevOps (automation, CI/CD pipelines and IaC), could be very intimidating for teams that are used to traditional project management frameworks. Taking these new practices and integrating them with your existing systems and tools is an involved process that can really tax resources, so it takes some planning to get right.

Integration challenges - Like how both these powers can span air travel, telecommunications and weather forecasting. Due to the reliance on specific project management tools and systems, traditional PMI frameworks often do not mesh well with modern DevOps toolchains. It can be difficult to get all these systems working together and serve as a barrier to adopting DevOps techniques. On top of this, the increased need for strong security measures in a DevOps environment (due to "constant testing" and continual change) adds another tangle to an already complex web.

The agility and flexibility that DevOps allows are well integrated with today's dynamic business environments. Further, these organizations can be more adaptive to changing requirements and market conditions, ensuring a competitive advantage in an ever-changing landscape. DevOps standard of continuous improvement aligns with PMI emphasis on processes, which can then reduce the waste and inefficiency associated with redundant rework thus yielding improvements in overall project success rates over time. This paper focuses on the cultural, technical and process challenges organizations face when they adopt DevOps in the PMI framework. This study presents a holistic literature review and thematic analysis, to demonstrate the real-world aspects of DevOps adoption and its effects in traditional project management practices. The paper distills current academic research and practitioner examples to provide recommendations that are attuned to the work organizations undergoing such transformative journeys.

This work thus deepens our comprehension of appropriate ways to incorporate DevOps with PMI contexts, acknowledging the challenges (as well as possible key points for transformation) that are related to this integration. The results provide information that enables organizations to implement DevOps successfully, supporting them in delivering projects and moving code faster.

II. LITERATURE REVIEW

The literature on DevOps adoption in PMI contexts was shown to be fraught with challenges and opportunities. The literature suggests that some of the key barriers to facilitating PE and (perhaps more so) integration of practices are cultural in nature. Springer has research on it, citing how difficult overcoming the institutional inertia against collaborative and automated workflows will prove to be. It also points out integration challenges, noting that bringing new DevOps tools with incumbent systems is fraught with difficulty as per a few studies.

Another major barrier is the disadvantage for skills and training gap. SpringerLink says quite simply that there is often not the expertise in advanced DevOps practices so a lot of investment ends up going into training and developing your staff. Further, accelerated delivery at this pace could be met by a software quality disaster of apocalyptic dimensions (IEEE).

However, DevOps also has some disadvantages. For instance, it promotes teamwork among development and operations teams leading to improved workflow. Moreover, efficient automation of testing and deployment minimizes human errors. Furthermore, employing DevOps also accelerates the time to get products onto markets while favoring flexibility that allows businesses to respond quickly to market dynamics.

Cultural Resistance

Research Findings: One of the major problems that are depicted in the literature is cultural resistance. In many organizations, project management practices are deeply embedded in their culture and this often runs contrary to what DevOps stands for. It can take various forms such as hesitancy to embrace new tools and processes, doubt on the benefits of DevOps, and a general dislike for change.

Example Case Studies: Case studies from organizations that have attempted to implement DevOps highlight these cultural challenges. For instance, a study by SpringerLink describes a financial services company that faced significant pushback from its operations team, which was accustomed to a rigid, hierarchical structure. The development team, on the other hand, was more open to the agile methodologies that DevOps promotes. This cultural divide created friction and slowed down the adoption process.

Strategies for Overcoming Cultural Resistance

To overcome cultural resistance, organizations can adopt several strategies. These include:

1. **Leadership Commitment:** Senior leadership must demonstrate a strong commitment to the DevOps transformation, actively promoting its benefits and leading by example.
2. **Cross-Functional Training:** Providing training programs that bring together development and operations teams can foster mutual understanding and collaboration.
3. **Gradual Implementation:** Introducing DevOps practices gradually rather than attempting a wholesale transformation can help ease the transition and reduce resistance.

Integration Issues

Research Findings: Integration issues are another major challenge identified in the literature. Organizations often struggle to integrate new DevOps tools with their existing project management and IT infrastructure. This challenge is particularly acute in organizations with legacy systems that are not designed for the level of automation and continuous delivery that DevOps requires.

Example Case Studies: A study published in IEEE Xplore discusses a large healthcare provider that faced significant integration challenges when attempting to implement DevOps. The organization's existing IT infrastructure was built on a monolithic architecture, making it difficult to integrate new DevOps tools and practices. The integration process was further complicated by the need to comply with strict regulatory requirements.

Strategies for Addressing Integration Issues

To address integration issues, organizations can consider the following strategies:

1. **Tool Compatibility:** Selecting DevOps tools that are compatible with existing systems can reduce integration challenges.
2. **Incremental Integration:** Integrating DevOps tools incrementally, starting with non-critical systems, can help identify and resolve issues before they impact core operations.
3. **API-Driven Integration:** Leveraging APIs to facilitate communication between DevOps tools and existing systems can simplify the integration process.

Skills and Training Gaps

Research Findings: The literature consistently highlights skills and training gaps as significant barriers to DevOps adoption. PMI-trained professionals often lack the expertise needed to implement and manage DevOps practices, particularly in areas such as automation, CI/CD pipelines, and IaC.

Example Case Studies: A case study from SpringerLink discusses a technology company that struggled with skills gaps during its DevOps implementation. The company's project managers and IT staff were well-versed in traditional PMI methodologies but lacked the

technical skills required for DevOps. This gap necessitated extensive training and hiring of new talent with DevOps expertise.

Strategies for Bridging Skills and Training Gaps

To bridge skills and training gaps, organizations can adopt the following strategies:

1. **Targeted Training Programs:** Developing targeted training programs that focus on key DevOps skills can help upskill existing staff.
2. **Hiring Expertise:** Bringing in external experts with DevOps experience can provide valuable guidance and accelerate the adoption process.
3. **Mentorship and Collaboration:** Encouraging mentorship and collaboration between experienced DevOps practitioners and PMI-trained professionals can facilitate knowledge transfer and skill development.

Quality Assurance Challenges

Research Findings: Maintaining software quality while accelerating delivery is a complex challenge in DevOps adoption. The continuous nature of DevOps, with its emphasis on rapid iterations and frequent deployments, can strain traditional quality assurance processes.

Example Case Studies: An analysis of a software development organization that encountered problems with quality assurance throughout its DevOps transition is published in IEEE Xplore. Due to the inadequacy of the company's current QA procedures, software faults, and operational problems increased as a result of the rapid pace of DevOps.

Strategies for Ensuring Quality Assurance

The following tactics might be taken into consideration by enterprises to guarantee quality assurance in a DevOps environment:

1. **Automated Testing:** Implementing automated testing at various stages of the development pipeline can help identify and resolve issues quickly.
2. **Continuous Monitoring:** Establishing continuous monitoring practices can provide real-time insights into system performance and detect anomalies early.
3. **Integrated QA Teams:** Integrating QA teams into the development and operations processes can ensure that quality is a shared responsibility and is addressed throughout the development lifecycle.

III. METHODOLOGY

This study employs a rigorous qualitative research methodology to comprehensively examine the challenges and opportunities associated with adopting DevOps within Project Management Institute (PMI) contexts. The methodology comprises a detailed systematic literature review (SLR) to synthesize and analyze insights from recent academic research on the topic.

Systematic Literature Review (SLR):

- **Definition and Purpose:** A Systematic Literature Review (SLR) is a methodical and structured approach to reviewing existing research literature. It aims to identify, evaluate, and synthesize relevant studies to provide a comprehensive understanding of a specific topic. In this study, the SLR is employed to gather and analyze scholarly articles that discuss DevOps adoption within the PMI framework.
- **Search Strategy:** To ensure a comprehensive review, a systematic search was conducted across several reputable academic databases. These included IEEE Xplore, SpringerLink, and Elsevier, which are known for their extensive collections of research papers in technology and project management. The search strategy involved using a combination of relevant keywords and phrases such as "DevOps challenges,"

"DevOps adoption," "PMI contexts," and "DevOps benefits."

- **Inclusion and Exclusion Criteria:** The selection criteria for including studies in the review were based on relevance, quality, and applicability to the PMI context. Studies were included if they:
 - Focused on DevOps practices and their integration into project management frameworks.
 - Provided empirical evidence or theoretical insights relevant to PMI environments.
 - Were published in peer-reviewed journals or reputable conference proceedings.

Research was disregarded if it:

- Did not particularly discuss DevOps within the PMI framework
- were either not published in peer-reviewed journals or did not follow rigorous methodology.
- were out-of-date or didn't fulfill the inclusion requirements for quality standards.

Data Extraction:

- **Data Collection:**Data were methodically gathered from the chosen publications in order to gather pertinent details regarding the potential and difficulties associated with the implementation of DevOps. This required determining the main themes pertaining to integration problems, skills shortages, cultural opposition, automation's advantages, and other pertinent elements.
- **Data Organization:** Based on recurrent themes and subjects, classifications were created from the retrieved data. During this process, each study's findings, methodology, and contributions to our understanding of DevOps in PMI contexts were highlighted in-depth descriptions and annotations.

Thematic Analysis:

- **Coding and Categorization:** Patterns in the retrieved data were found and examined using thematic analysis. This required categorizing the data according to important themes like automation, collaboration enhancements, skills gaps, cultural opposition, and integration difficulties. The process of classifying qualitative data in order to find important themes and patterns is known as coding.
- **Pattern Identification:** Through thematic analysis, common patterns and trends across the literature were identified. This analysis helped in understanding how different studies approached the topic and highlighted consensus or variations in findings. It also provided insights into recurring challenges and potential benefits associated with DevOps adoption.

Synthesis and Interpretation:

- **Integration of Findings:** The results of the thematic analysis were combined to create a comprehensive picture of the state of DevOps adoption in PMI settings. This includes a summary of the main obstacles that businesses must overcome, such as cultural resistance and integration problems, as well as the advantages that come with implementing DevOps, like improved cooperation and more automation.
- **Practical Recommendations:** Based on the synthesis, practical recommendations were developed to guide organizations in addressing the identified challenges and leveraging the opportunities presented by DevOps. These recommendations aimed to provide actionable insights for practitioners and decision-makers involved in project management and software development.
- **Implications for Future Research:** The study also highlighted gaps and areas for further research. It provided a foundation for future investigations into specific

aspects of DevOps adoption in PMI contexts, such as the effectiveness of various integration strategies or the impact of cultural change initiatives on successful DevOps implementation

Quality Assurance:

- **Rigorous Review:** A high degree of rigor was used in the review process in order to guarantee the validity and trustworthiness of the results. This involved confirming the quality and relevance of the included research, cross-checking the results, and making sure that data extraction was consistent.
- **Peer Review and Feedback:** Peer review was used to get input from subject-matter experts on the methodology and findings. This assisted in fine-tuning the study and guaranteeing that the findings reached were substantiated by sufficient data.

IV. RESULTS

The results section presents key findings derived from the systematic literature review (SLR) on the challenges and opportunities associated with adopting DevOps within PMI contexts. This section summarizes the main themes identified during the analysis of the literature and highlights the insights gained from recent research.

Challenges Identified:

- **Cultural Resistance:** One of the primary challenges identified is cultural resistance within organizations. Established norms and traditional practices often create a barrier to adopting DevOps, which requires a shift towards more collaborative and agile ways of working. Organizations with entrenched hierarchical structures and siloed departments face significant obstacles in transitioning to a DevOps culture that emphasizes cross-functional teams and open communication.
- **Integration Issues:** Integrating DevOps tools and practices with existing project management systems poses considerable difficulties. Many organizations rely on traditional project management tools that are not designed to work seamlessly with DevOps automation tools. This integration challenge includes aligning DevOps practices with existing workflows, ensuring data consistency across systems, and managing the complexity of combining disparate tools.
- **Skills Gaps:** A significant skills gap is evident, particularly in advanced DevOps practices. Many professionals trained in traditional PMI methodologies may lack the expertise required for automation, continuous integration/continuous deployment (CI/CD) pipelines, and infrastructure as code (IaC). This gap necessitates substantial investment in training and development to equip team members with the necessary skills to implement and manage DevOps practices effectively.
- **Quality Assurance:** Maintaining high software quality while accelerating delivery through DevOps is another challenge. The rapid pace of deployments inherent in DevOps can lead to quality issues if not managed properly. Ensuring that automated testing and quality assurance processes are robust enough to handle frequent changes is crucial to prevent defects and maintain software reliability.

Opportunities Identified:

- **Enhanced Collaboration:** Teams working on development and operations can collaborate more effectively thanks to DevOps. DevOps fosters more effective communication and coordination by dismantling organizational silos and promoting cross-functional teamwork. This results in more cohesive project teams and improved alignment with company objectives.
- **Increased Automation:** One of the main advantages of DevOps is automation, which boosts productivity and lowers human mistake rates. Teams can concentrate on more strategic operations by streamlining procedures and automating repetitive chores like monitoring, testing, and deployment. More software releases with greater

dependability and a quicker time to market are further benefits of this increasing automation.

- **Faster Time-to-Market:** By enabling quicker release cycles, DevOps principles help firms react faster to customer wants and market demands. The time needed to provide new features and improvements to the market is shortened by the frequent updates and changes made possible by continuous integration and continuous delivery (CI/CD) techniques.
- **Improved Adaptability:** The iterative nature of DevOps supports greater adaptability to changing requirements and market conditions. Organizations can more easily adjust their development and deployment processes to accommodate new insights, emerging trends, and evolving customer needs, providing a competitive advantage in a dynamic business environment.

V. DISCUSSION

The results of the literature research are interpreted in the discussion part, which also examines the potential and problems that firms adopting DevOps within PMI contexts may face. A more thorough examination of the results and suggestions for assisting companies in navigating the challenges of adopting DevOps are given in this section.

Analysis of Challenges:

- **Addressing Cultural Resistance:** Overcoming cultural resistance is essential for successful DevOps adoption. Organizations should focus on fostering a culture that embraces change and collaboration. This involves leadership commitment to driving cultural transformation, promoting open communication, and providing support for teams as they transition to new ways of working. Change management strategies, including training programs and team-building activities, can facilitate smoother cultural shifts.
- **Navigating Integration Issues:** To address integration challenges, organizations should prioritize selecting DevOps tools that are compatible with their existing systems. Investing in integration solutions or developing custom interfaces to connect disparate tools can help create seamless workflows. Additionally, involving stakeholders from both development and operations teams in the integration process ensures that the tools meet the needs of all users and align with project requirements.
- **Bridging Skills Gaps:** Organizations must invest in training and upskilling to bridge the skills gap associated with DevOps. This includes providing educational resources, workshops, and hands-on training to develop expertise in DevOps practices. Hiring experienced DevOps professionals or consultants can also provide valuable guidance and accelerate the adoption process.
- **Ensuring Quality Assurance:** Maintaining software quality amidst rapid deployments requires robust automated testing and quality assurance processes. Organizations should implement comprehensive testing strategies, including unit tests, integration tests, and performance tests, to identify and address issues early in the development cycle. Continuous monitoring and feedback mechanisms are also crucial for detecting and resolving quality issues promptly.

Leveraging Opportunities:

- **Enhancing Collaboration:** Organizations should leverage the collaboration benefits of DevOps by promoting cross-functional team structures and encouraging regular communication between development, operations, and other stakeholders. Implementing collaborative tools and practices, such as regular stand-up meetings and shared project management platforms, can further enhance team cohesion and productivity.

- **Maximizing Automation:** To fully capitalize on the automation opportunities offered by DevOps, organizations should identify repetitive tasks and processes that can be automated. Developing and maintaining automation pipelines for testing, deployment, and monitoring can lead to significant improvements in efficiency and reliability. Regularly reviewing and updating automation practices ensures that they remain effective and aligned with evolving project needs.
- **Accelerating Time-to-Market:** Organizations can benefit from the faster time-to-market enabled by DevOps by focusing on continuous improvement and iterative development. Adopting agile practices alongside DevOps can further enhance responsiveness to market changes and customer feedback. Prioritizing feature releases based on customer needs and market trends helps ensure that products remain competitive and relevant.
- **Enhancing Adaptability:** The adaptability provided by DevOps supports organizations in staying competitive in a rapidly changing environment. By embracing iterative development and continuous feedback, organizations can quickly adapt to new opportunities and challenges. Implementing flexible development processes and maintaining a proactive approach to market changes contribute to long-term success and resilience.

In general, harnessing the possibilities that arise to improve organizational performance and overcoming obstacles with focused initiatives are necessary for the successful adoption of DevOps in PMI contexts. Organizations can benefit from DevOps by concentrating on skills development, quality assurance, efficient integration, and cultural transformation. These initiatives can lead to higher automation, improved cooperation, quicker time-to-market, and increased adaptability.

VI. CONCLUSION

The integration of DevOps within the Project Management Institute (PMI) framework marks a significant transformation in how organizations approach software development and project management. This study has examined the complex landscape of DevOps adoption, revealing both substantial challenges and promising opportunities.

A primary challenge identified is cultural resistance. Organizations steeped in traditional PMI methodologies often face difficulties shifting from established, siloed structures to the collaborative, agile practices central to DevOps. Overcoming this resistance requires substantial effort in cultural change management. Additionally, integration issues pose a significant hurdle. Aligning new DevOps tools and practices with existing PMI systems can be complex, necessitating careful planning and execution to ensure seamless workflows. Skills gaps also present a notable barrier; the lack of expertise in advanced DevOps practices such as automation, continuous integration, and infrastructure as code necessitates significant investment in training and development. Furthermore, maintaining high software quality amidst rapid deployment cycles remains challenging, highlighting the need for effective automated testing and quality assurance processes.

Despite these challenges, the opportunities provided by DevOps are considerable. Enhanced collaboration between development and operations teams is a key benefit, fostering a more integrated and efficient workflow. Increased automation of repetitive tasks leads to greater operational efficiency and fewer manual errors, enabling teams to focus on more strategic activities. DevOps also facilitates faster time-to-market, allowing organizations to respond swiftly to market demands and customer needs, thereby gaining a competitive edge. Moreover, the iterative nature of DevOps improves organizational adaptability, enabling companies to adjust to changing requirements and market conditions more effectively.

Organizations aiming to adopt DevOps within a PMI context must navigate these challenges with strategic planning. Addressing cultural resistance, investing in integration solutions,

bridging skills gaps, and maintaining rigorous quality assurance processes are essential for successful implementation. Concurrently, leveraging the benefits of enhanced collaboration, increased automation, faster time-to-market, and improved adaptability can lead to significant improvements in project management and overall operational performance. Future research should explore the impact of specific DevOps tools and practices, analyze successful case studies, and develop frameworks to guide effective adoption. In conclusion, adopting DevOps within PMI frameworks presents both challenges and opportunities, but with a thoughtful approach, organizations can achieve enhanced project management outcomes and maintain a competitive advantage in a dynamic marketplace.

REFERENCES

- [1] Fowler, M. Continuous integration. <https://www.martinfowler.com/articles/continuousIntegration.html>. Last accessed 27 Nov 2020.
- [2] Lwakatare, L. E., Kuvaja, P., & Oivo, M. (2015). Dimensions of DevOps. In *International Conference on Agile Software Development* (pp. 212–217). Springer, Helsinki, Finland. https://doi.org/10.1007/978-3-319-18612-2_19.
- [3] Debois, P. Agile infrastructure & operations. <http://www.jedi.be/presentations/agile-infrastructure-agile-2008.pdf>. Last accessed 27 June 2020.
- [4] Parnin, C., Helms, E., Atlee, C., Boughton, H., Ghattas, M., Glover, A., Holman, J., Micco, J., Murphy, B., Savor, T., Stumm, M., Whitaker, S., & Williams, L. (2017). The Top 10 adages in continuous deployment. *IEEE Software*, 34, 86–95. <https://doi.org/10.1109/MS.2017.86>
- [5] Arvanitou, E. M., Ampatzoglou, A., Bibi, S., Chatzigeorgiou, A., & Deligiannis, I. (2022). Applying and researching DevOps: A tertiary study. *IEEE Access.*, 10, 61585–61600. <https://doi.org/10.1109/ACCESS.2022.3171803>
- [6] Stillwell, M., & Coutinho, J. G. F. (2015). A DevOps approach to integration of software components in an EU research project. In *Proceedings of the 1st International Workshop on Quality-Aware DevOps* (pp. 1–6). ACM, New York, NY, USA. <https://doi.org/10.1145/2804371.2804372>.
- [7] Ebert, C., Gallardo, G., Hernantes, J., & Serrano, N. (2016). DevOps. *IEEE Software*, 33, 94–100. <https://doi.org/10.1109/MS.2016.68>
- [8] Senapathi, M., Buchan, J., & Osman, H. (2018). DevOps capabilities, practices, and challenges. In *Proceedings of the 22nd International conference on evaluation and assessment in software engineering 2018* (pp. 57–67). ACM, New York, NY, USA. <https://doi.org/10.1145/3210459.3210465>.
- [9] Al-Zahrani, S., & Fakieh, B. (2020). How DevOps practices support digital transformation. *International Journal of Advanced Trends in Computer Science and Engineering*. 9, 2780–2788. <https://doi.org/10.30534/ijatcse/2020/46932020>.
- [10] Mishra, A., & Otaiwi, Z. (2020). DevOps and software quality: A systematic mapping. <https://doi.org/10.1016/j.cosrev.2020.100308>.
- [11] Erich, F., Amrit, C., Daneva, M. (2014). Report: DevOps literature review. <https://doi.org/10.13140/2.1.5125.1201>.
- [12] Rütz, M. (2019). Devops: A systematic literature review. IT Management Seminar Paper Summer Term 2019.
- [13] Guerrero, J., Zúñiga, K., Certuche, C., & Pardo, C. (2020). A systematic mapping study about DevOps. *Journal de Ciencia e Ingeniería*. 12, 48–62. <https://doi.org/10.46571/JCI.2020.1.5>.
- [14] Strieker, J., Volchk7ov Supervisor Martin Svensson Karlskrona, S. (2022). Broad devOps adoption and organizational performance of the software firm. <http://urn.kb.se/resolve?urn=urn:nbn:se:bth-24285>.
- [15] Khan, M. U., Sherin, S., Iqbal, M. Z., & Zahid, R. (2019). Landscaping systematic mapping studies in software engineering: A tertiary study.