Navigating DevOps Implementation Hurdles in the Pakistani Software Industry

Sidra Soomro[1,*], Qasim Arain[1], Erum Yousef Abbasi[2], Mehwish Shaikh[1], Shabana Tanwari[1],

[1]Department of Software Engineering, MUET, Jamshoro, Pakistan
[2]Beijing University of Posts and Telecommunications (BUPT), Beijing, China
[*]Corresponding author: sidrasoomro14@gmail.com

Abstract

DevOps has become extremely essential in the software industry around the world due to the simplicity of the use of software development and deployment. DevOps aims to establish a collaborative partnership between developers and operators with the goal of expediting the development process and producing efficient software that automates the regular delivery of new applications. Currently, the software industry in Pakistan is not embracing DevOps as a source of added value or a competitive advantage. As a result, companies are facing challenges in terms of performance metrics as they strive to survive in this environment. DevOps stands as one of the contemporary software methodologies that contribute to enhancing agility practices within a collaborative culture, ultimately empowering the processes of software delivery and development. The primary goal of the present research was to examine the critical challenges and their impacts on the performance of DevOps Engineers. In relation to the aforementioned variables effects of various demographic variables were also sought. Their performance was operationalized through DORA matrics and Data was analyzed using SPSS and MAXQDA, which were compared later. Results revealed that a higher level of performance is achieved among foreign DevOps as compared to Pakistani DevOps despite facing the same challenges. Recommendations for future research were suggested and implications of the study along with its limitations were also discussed especially since there is a dire need for research to measure the performance of DevOps via using other instruments in order to make more suggestions for Engineers, policymakers, researchers, and others.

Keywords—Challenges, DevOps, DORA matrics, Pakistan

1 Introduction

DevOps has gained prominence in the global software business because of its easy-to-use software development and deployment procedures. The goal of development and operations, or DevOps, is to establish a collaborative partnership between developers and operators in order to construct efficient software that automates the continuous supply of new software and speeds up development. Businesses in Pakistan's software industry continue to perform quite well even if they choose not to implement DevOps because the sector is not currently using it to create value or gain a competitive edge.

One of the latest approaches to software development, dubbed DevOps, enhances the software delivery and construction process by expanding agile methods within a collaborative setting. It has also been noted that every project has different goals for its development and operations teams. The developer's goal is to introduce new features, whereas the operator's is to maintain the software's availability and stability. The project's goals and the cooperation between the development and operations teams are seen as essential. A strategy called DevOps has been put forth to achieve these goals, expand them, and bring the software development and operations teams together (Erich et al., 2017)[1]. It's an organizational strategy meant to promote empathy and collaboration between different divisions. Cutting down on the amount of time be-tween software development and use while sacrificing quality is the ultimate goal. Nevertheless, neither a definition nor an inventory of all the topics covered by DevOps is established.

In software engineering, DevOps has grown in impor-

ISSN: 2523-0379 (Online), ISSN: 1605-8607 (Print)
DOI: https://doi.org/10.52584/QRJ.2102.08

tance for both the commercial and academic domains (Luz et al., 2019)[2]. The global paradigm shift towards the constant usage of software has brought up both opportunities and challenges (Lwakatare et al., 2015), particularly for poor countries like Pakistan. More accurately, it has been difficult to deploy DevOps (Smeds et al., 2015). It necessitates that the com-pany innovate, modernize its technology, appoint staff with specialized training, and create new protocols. Above all, the adoption process is exclusive to the business. It has been shown that adopting DevOps presents hurdles for this undertaking. There are generally three viewpoints regarding the difficulties: Engineers can gain from both, in addition to being highly competent and competent for a DevOps position. But in order to implement continuous delivery, enterprises must know how to change their systems (Abdelke-bir et al., 2017)[3]. Supervisors are eager to evaluate the effectiveness and quality of DevOps implementations across the company (Agrawal and Rawat, 2019)[4]. Managers and engineers alike, or practitioners, must select the appropriate automation toolkit. The last responsibilities given to researchers were to evaluate the condition of Devops as it is currently applied and to teach the coming generation of software engineers about Devops principles and approaches (Bass et al. 2015; Luz et al., 2019)[5]. Several practitioners have provided alternative frameworks, methods, and checklists to expedite DevOps adoption and reduce risks (Smeds et al., 2015)[6]. While major corporations like Amazon and Flickr have had success using these strategies, debates over their practicality and applicability continue.

One area of ongoing literature research concerns the application of DevOps in SMEs or small and medium-sized businesses. A limited number of researchers have focused on the performance of DevOps engineers, so it is necessary to pay attention to their performance in light of these challenges. Several scholars have called attention to these challenges and performance of DevOps, including the Performance Evaluation of Traditional Software Development Techniques and DevOps Automation Cultural Affairs, how evaluation of performance is addressed in DevOps, and many more. This ignorance highlights the degree of uncertainty caused by actual application, and the necessity of comparing the performance of Pakistani DevOps engineers with those of other nations (Leppänen et al., 2015)[7]. Consequently, it is necessary to compare the performance of Pakistani engineers with that of other up-and-coming DevOps engineers.

2 Literature Review

There has been a lot of work in recent years on deployment-related issues, particularly the miscommunication between DevOps and operators. This section highlights the research gap and discusses a few study approaches. Software products can be improved to deliver better results by adopting DevOps practices. The software product quality will increase when adopting DevOps practices that consider the strong relationship between culture, automation, measurement, and sharing, as they enhance quality (Perera et al., 2017)[8]. DevOps consists of practices and cultural values to minimize the barriers between development and operations teams and DevOps adoption involves a tight relationship between agility, automation, collaborative culture, continuous measurement, quality assurance, resilience, sharing, and transparency (Luz et al., 2018)[9]. The importance of comprehending how deployment practices are implemented in the development team is thus increased by using operations that make the software product available and ready for the requester as soon as it is implemented. This can be done by establishing a suitable maturity model and then employing it to evaluate the development team and operation team and their practices (Virtanen et al., 2017)[10].

One of the latest approaches to software development, dubbed Devops, enhances the software delivery and construction process by expanding agile methods within a collaborative setting. It has also been noted that every project has different goals for its development and operations teams. The developer's goal is to introduce new features, whereas the operator's is to maintain the software's availability and stability. The project's goals and the cooperation between the development and operations teams are seen as essential. A strategy called DevOps has been put forth to achieve these goals, expand them, and bring the software development and operations teams together (Erich et al., 2017)[11]. It's an organizational strategy meant to promote empathy and collaboration between different divisions. Cutting down on the amount of time be-tween software development and use while sacrificing quality is the ultimate goal. Nevertheless, neither a definition nor an inventory of all the topics covered by DevOps is established.

Furthermore, earlier practitioner studies focused on the industrial adoption of CI (Hilton et al., 2016)[12] along with efficiency testing (State of Performance Engineering, 2018). TechBeacon's 2018 report, "State of Performance Engineering," has an indirect connection to DevOps because it evaluated

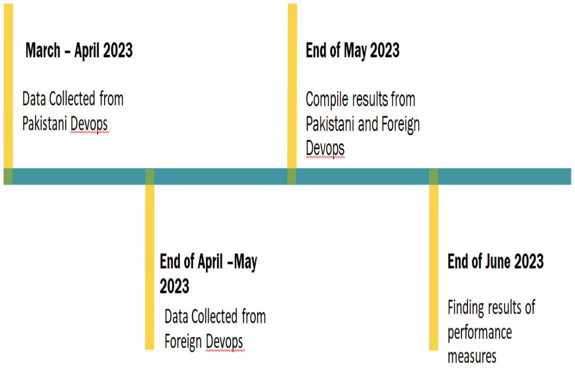

March - April 2023
Data Collected from Pakistani Devops
End of May 2023
Compile results from Pakistani and Foreign Devops
End of April -May 2023
Data Collected from Foreign Devops
End of June 2023
Finding results of performance measures

Fig. 1: Data Collection Timeline

practices for performance engineering across the software development cycle and found that 62% of respondents thought performance engineering was crucial for DevOps. According to Hilton et al. (2016), who investigated the challenges developers encounter while utilizing continuous integration (CI), flaky tests, and the intricacy of CI tools represent significant obstacles to successful DevOps integration.

3 Methodology

The present study aimed to investigate the Identification of critical challenges to adopting DevOps in national and international (Pakistan and others) Software industries in DevOps engineers. The present study is divided into two parts i.e. MLR and main study. In the main study the hypothesis of the present study and predicted model were also tested. The timeline for data collection for this study is shown is Fig. 1.

3.1 Research Design

The present study utilized a correlation research design to conduct the study through the survey method. The participants selected in the study were those experienced in DevOps engineering with one year or more. The study consisted of two phases, which are;

1) MLR (Multivocal Literature Review)
2) Main study

3.2 Instruments

The construct of the present study was operationalized through psychometrically sound instruments. The measure was valid and reliable. Demographics like name, country, gender, and age were also considered in the research.

3.2.1 DORA (matrics)

This 4-item matrics was developed by (Gene Kim and Jez Humble along with Dr. Nicole Forsgren, 2019) to assess the performance of the DevOps teams and find out whether they are "low performers" to "elite performers". It consisted of four matrices of DF (Deployment Frequency), LT (Lead Time for Changes), MTTR (Mean Time to Recovery), and CFR (Change Failure Rate) as a global measure of Performance for DevOps Engineers. The responses on the matrices were rated on the elite to low rate performance.

3.2.2 MLR (Multivocal Literature Review)

Multilingual analysis of the research was done from both gray literature and formal academic publications. The peer-reviewed, published papers in meetings, publications, and seminars make up the formal scholarly literature. Following the recommendations of (Garousi et al., 2019) we have established detailed protocols for MLR in order to complete the literature research and data extraction steps successfully. A summary of the MLR procedure is provided in (Garousi et al., 2019)[13].

3.2.3 Main study

After the selection of matrices and determining their properties, the scale was administered to a small sample of (N=106) DevOps Engineers which included (n=59) men and (n=47) women Engineers in an array to test the proposed objectives of the current research.

3.2.4 Sample of the Main Study

The sample of the main study consisted of DevOps Engineers (N=106) from Pakistan recruited through random sampling as well as the snowball technique in which one person refers the things to another participant for the survey other uses his own contacts and this makes a chain to collect the data for the survey. Informed consent was also taken from all participants before the administration of the matrices. The demographics such as gender, age, and country were also asked.

Table 1 displays the sample's demographic characteristics in terms of frequency and percentage. In Table 1 the frequency of men is higher than women. In the same way, the Proportion of DevOps Engineers from Pakistan is higher than that of other countries. The study comprised four categories of age which are 20-24, 24-28, 28-32, and 32-36 respectively. The proportion of youth is much more than that of elderly in DevOps engineering in Pakistan as well as other countries according to the sample of present research.

TABLE 1: The sample's demographic details (N=106)

Demographics	Frequency	Percentage	Sample=N
Gender			
Male	59	55,7	106
Female	47	44.3	
Country			
Pakistan	70	66.0	
Australia	18	16.0	
India	18	17.0	
Others	1	1.0	106
Age			
20-24	44	41.5	
24-28	56	52.8	
28-32	6	5.7	
32-36	0	0	106

4 Inclusion and Exclusion Criteria

People who were DevOps Engineers for more than one year, experienced or operating in software houses were included in the study. People with other Software Engineers and the elderly were not part of the study.

4.1 Demographic Form

The demographic form was designed to request data that was required for the study, including name, age, gender, and country. The consent of the subjects was obtained before data collection, including a statement outlining the purpose of the study.

5 Procedure

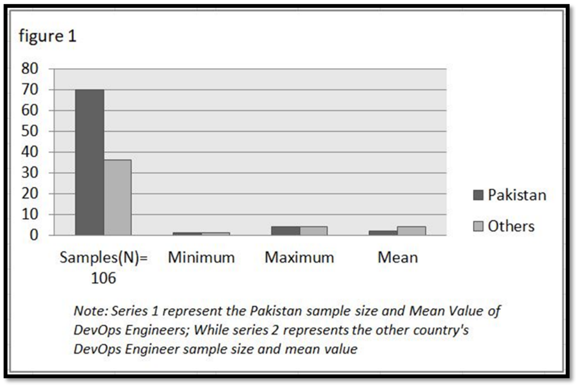
After the selection and approval of the topic, the English version matrices and used scales for the main study. Most participants were selected from Pakistan randomly using the snowball technique and a few of them were selected from foreign countries. After getting informed consent the matrices of the present study with demographic sheet were administered. After obtaining the desired data from the participants, the participants were thanked for their cooperation. The analysis was done through quantitative techniques as well as using SPSS V2 and MAXQDA.

6 Results and Discussion

The Statistical Product and Service Solution (SPSS 26V) and MAXQDA were used to analyze the study's data. Descriptive statistics were among the statistical methods employed for the analysis. The psychometric characteristics of the study's measurements were calculated. The findings of the current investigation were presented. The results of the primary study and the Multivocal literature review are the two phases that make up the outcome section.

TABLE 2: Deployment frequency where number of samples (N= 106)

Development Frequency	N	Min	Max	Mean	Standard Deviation
Valid N	106	1.00	4.00	2.01	1.037



Note: Series 1 represent the Pakistan sample size and Mean Value of DevOps Engineers; While series 2 represents the other country's DevOps Engineer sample size and mean value

Fig. 2: Comparison of deployment frequency of Pakistan and other country's DevOps

6.1 Deployment Frequency

For the primary application or services you worked on, how often does your organization deploy code to production or release it to the end user?

Table 2 describes the mean value and St. Deviation of the responses given by the respondents. Mean value is always distributed in three categories i.e. 1.00 to 2.33 describes the low category. In the same way, the Moderate value is distributed from 2.34 to 3.66. And higher value is distributed in 3.67 to 5.00. So according to the mean value, all the participants score low in this category. It means they usually do a few deployments in the six months.

Figure 2 depicts the results of the SPSS 23 software. According to it, most of the people do multiple de-ploys per day with a frequency of 46% which is the figure for 49 Devop engineers. While 14 people with a frequency of 13% responded that they do it between once per week or per month. On the other hand, most of the people responded that they perform deployment between once per month and once every 6 months with (f = 34 & % = 36). Similarly, people with (f = 7 & % = 7) answered about their deployment rate that they have done it fewer than once per six months.

TABLE 3: Lead Time for Change

Lead time for Change	N	Min	Max	Mean	ST. Deviation
N value	106	1.00	4.00	2.46	1.296

TABLE 4: Time to restore services

Time to restore services	N	Min	Max	Mean	ST. Deviation
N value	106	1.00	4.00	2.36	1.31

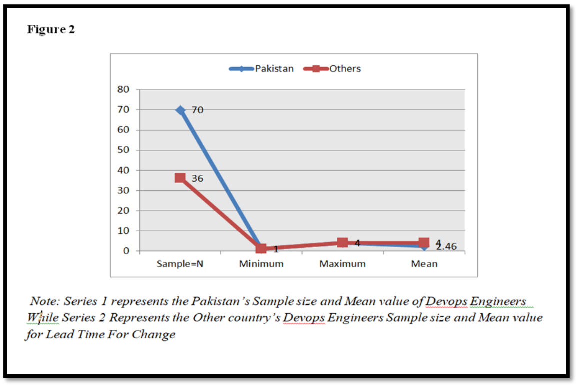


Figure 2

Note: Series 1 represents the Pakistan's Sample size and Mean value of Devops Engineers While Series 2 Represents the Other country's Devops Engineers Sample size and Mean value for Lead Time For Change

Fig. 3: Comparison of Lead time for Change in Pakistan and other country's DevOps

6.2 Lead Time for Change

For the primary application or service you work on, what is your lead time for changes (i.e. how long does it take to go from code committed to code to successfully running in Production)?

Table 3 shows the results for the primary application or service they work on, what is their lead time for changes like how long it takes to go from code committed to code to successfully running in the Production. Most of the participants gave negative answers in response to this question. According to them, it takes more than six months for the primary application or service. Their lead time for changes takes six months or more according to the value of the mean which is 2 approximately. On the other hand, on the basis of the response given by the foreign, they said that they take only one day for the production and require one day for the changes to go from code committed to code to successfully running in Production. They lie in the Elite class according to their responses given in the open-ended Questions. One of the participants from Australia answered "One hour". Others were eye to eye with this statement. In Pakistan, this is done once in 6 months as asserted by most of the participants. This is represented in the form of a graph.

Figure 3 depicts the results for Lead time for change provided by the SPSS 23 software. According to it, most of the people take less than one hour with (f = 36 &% = 34), while 22 people with a frequency of 24% responded that they do it between one day and one week. On the other hand, most of the people responded that they take time for changes between one month and six months with (f = 7 & % = 7). Similarly, people with (f = 36 & % = 39) answered that their deployment take almost more than six months time for changes. Hence, it showed that most of the people are suffering from the challenge of time taken for changes.

6.3 Time to restore service

For the primary application or service you work on, how long does it generally take to restore service, when a service incident or defect that impacts usurers occurs (e.g., unplanned outage or service impairment)?

Table 4 displays the results for the primary application or service the respondents worked on, and depicts the length of time it takes to restore service when a service incident or defect that impacts users occurs (e.g. unplanned outage or service impairment). Overall results from the respondents showed that all the participants responded negatively to the answer to this question. According to the mean value of 1.5, they scored low in answer to the given statement. So, according to the matrix scoring pattern, they have a low-performance score on this statement. It takes almost more than six months in order to restore defects. On the other hand, those who belong to foreign countries like Australia and India, take just one hour or less to fix the defects that have a huge impact on the users in the case of an unplanned outage or service impairment. It means they belong to the elite class category of DevOps Engineers according to the matrices. But, on the basis of Pakistani DevOps Engineers, they scored low on this section as mentioned in the above paragraph. It's graphically represented.

Figure 4 depicts the results for the time to restore services provided by the SPSS 23 software. According to it, most people take less than one hour with (% = 42) to restore services when blocked by any error or any other hurdle, while people with a frequency of 10% responded that they do it between one day and one week. On the other hand, most of the people responded that they take time to restore the services between one month and six months with (%= 16). Similarly, people with (%= 31) answered that their de-ployment takes

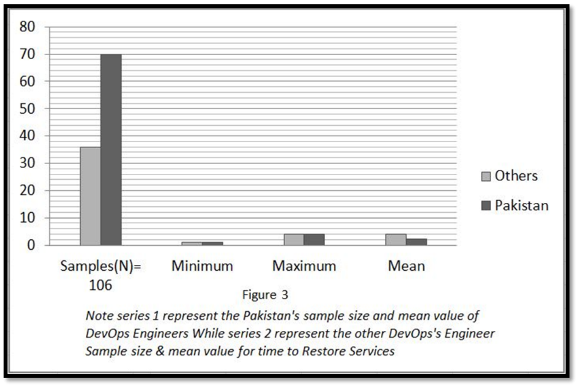


Fig. 4: Comparison of Time to restore services of Pakistan and other countries DevOps

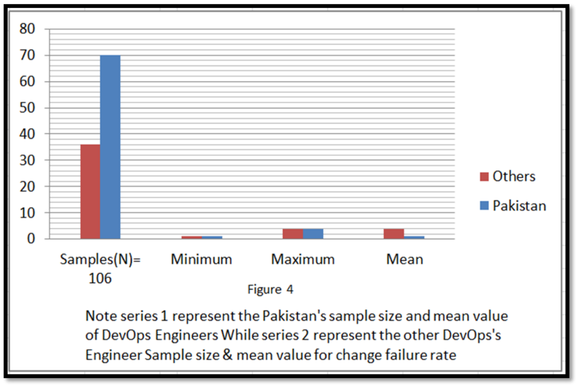


Fig. 5: Comparison of change failure rate of Pakistan and other country's DevOps

TABLE 5: Change Failure Rate

Change failure rate	N	Min	Max	Mean	ST. Deviation
N value	106	1.00	4.00	1.18	.582

almost more than six months time for changes. Hence, it showed that most of the people are suffering from the challenge of time to restore services.

6.4 Change failure rate

For the primary application or service you work on, what percentage of changes to production or released to users result in degraded services (e.g. lead to service impairment or service outage) and subsequently require remediation (e.g. require a hotfix, rollback, fix forward, patch)?

Table 5 shows the results for the primary application or service respondents worked on, their percentage of changes to production or released to users result in degraded services (e.g. lead to service impairment or service outage) and subsequently require remediation (e.g. require a hotfix, rollback, fix forward, patch). The results showed again negativity for all the participants. According to the mean value of 1.5, it can be anticipated that all the participants performed poorly at their software houses. Hence, according to the matrices, all the participants present a percentage range from 16 to 30% for the production that leads to degraded services. On the other hand, those participants were part of this study from Australia and India, they belong to the elite class according to the matrices. Their percentage ranges from 0 to 15%. They can do remediation for degraded services in the lowest possible time with a frequency of 0 to 15%, while there is a huge difference between them & Pakistani Participants. They answered in those manners that represent the negative value of frequency ranging from 16% to 30%.

Figure 5 depicts the results change failure rate provided by the SPSS 23 software. According to it, the graph showed that most of the people show the same statements for change of failure rate in the case of Pakistan DevOps Engineers.

7 Suggestions for Future Research

Based on the aforementioned parameters, the following recommendations are made for additional study.

1) To maximize the study's external validity, the sample in a prospective study should be diversified in terms of categories. It ought to extend throughout the district and various Pakistani regions as well as other nations.
2) Other performance measuring matrices should be used to assess the performance of Devops along with exploring new challenges that they face.
3) Longitudinal and experimental research designs should be used to assess the causal effects.
4) It is very hard to manage numerous tests on an Engineers sample in one session. Studies using a multi-method approach should table for numerous sessions of consideration so that they can professionally apply numerous methods to a sample.

8 Discussion

In order to better understand the DevOps problems and the performance of DevOps engineers in Pakistan and other countries, as well as the efficacy of DevOps engineers for their operations in Pakistan and other countries, the current study developed and tested a model of DevOps engineer performance and compared

Pakistani DevOps engineers in various countries using this model, which also took demographic factors like gender, age, and country into account but the other factors such as organizational culture, infrastructure, and resource availability are not part of this study which may be considered for future work to exploit more performance of Devops in Pakistan as well as other countries.

The two stages that comprise the outcome section are the primary study's findings and the Multivocal literature review. The current study's findings demonstrated that a variety of factors influence DevOps's success. The deployment rate is one of them. The findings indicate that Pakistani engineers deploy at a lower rate than foreign engineers. Lead time for change is an additional element. It indicates the amount of time needed to run a productive production. Our findings show that, in comparison to engineers in other nations, Pakistani DevOps Engineers require more time to run the system successfully. Similarly, Pakistani Engineers experienced more effects than others in terms of change failure rate and time to restore services. The survey evaluated performance engineering practices throughout the software development life cycle, and it found that 62% of participants agreed that performance engineering is important for DevOps.

8.1 Implications of the present study

To the best of the researcher's knowledge, this is a significant study investigating the influence of Devops Engineer performance on Devops operations in Pakistan. It contributes to the substantial body of research and might inspire further investigation to gain a more thorough and accurate understanding of the complex cause-and-effect interactions between the study's factors. In my opinion, addressing each of these constructs thoroughly is necessary if we are to properly address the performance of our DevOps Engineers. DevOps Engineers must also do correct procedures in order to continue achieving high-performance levels. Readers unfamiliar with structural equation modeling can interpret results that corroborate our hypothesized causal linkages. Since statistics cannot verify causation, these results do not give sufficient evidence to infer the links. Strong theoretical foundations, experimental and longitudinal evidence, and a wealth of sources confirming the same relationships reported in this work are required to support our capacity to draw significant conclusions. Our study aims to stimulate validation research in this area.

9 Conclusion

The analysis conducted via SPSS and MAXQDA software reveals that performance measures in overseas nations, particularly India and Australia, consistently rank higher on standard matrix scales, signifying elite and high excellence. Conversely, Pakistani DevOps engineers demonstrated lower scores across all matrices, contrasting with the higher scores achieved by their global counterparts, despite facing a similar challenge.

References

[1] A.A. Khan, J. Keung, M. Niazi, S. Hussain, and A. Ahmad, ”Systematic literature review and empirical investigation of barriers to process improvement in global software development: client–vendor perspective,” Inf. Softw. Technol., vol. 87, pp. 180–205, 2017.

[2] A.A. Khan, J. Keung, S. Hussain, M. Niazi, and S. Kieffer, ”Systematic literature study for dimensional classification of success factors affecting process improvement in global software development: client–vendor perspective,” IET Softw., vol. 12, no. 4, pp. 333–344, 2018.

[3] A. Virtanen, K. Kuusinen, M. Leppänen, A. Luoto, T. Kilamo, and T. Mikkonen, ”On Continuous Deployment Maturity in Customer Projects,” in Proc. of the Symposium on Applied Computing (SAC '17), pp. 1205–1212, 2017. Available: [Online]. https://doi.org/10.1145/3019612.3019777.

[4] A. Katal, V. Bajoria, and S. Dahiya, ”DevOps: Bridging the gap between development and operations,” in Proc. 3rd Int. Conf. Comput. Methodolog. Commun. (ICCMC), Mar. 2019, pp. 1–7.

[5] A. A. Khan, J. Keung, M. Niazi, S. Hussain, and M. Shameem, ”GSEPIM: A roadmap for software process assessment and improvement in the domain of global software development,” J. Softw.: Evolution and Process, vol. 31, no. 1, e1988, 2019.

[6] G. McGraw, S. Migues, and J. West, ”Building Security In Maturity Model (BSIMM) Version 6,” 2015.

[7] F. Erich, C. Amrit, and M. Daneva, ”Cooperation between information system development and operations: a literature review,” in Proc. of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, 2015.

[8] P. Perera, M. Bandara, and I. Perera, ”Evaluating the Impact of DevOps Practice in Sri Lankan Software Development Organizations,” in Proc. of the 16th International Conference on Advances in ICT for Emerging Regions (ICTer), 2017, pp. 281–287.

[9] W. P. Luz, G. Pinto, and R. Bonifácio, ”Building a collaborative culture: A grounded theory of well succeeded DevOps adoption in practice,” in Proc. 12th ACM/IEEE Int. Symp. Empirical Softw. Eng. Meas., Oct. 2018, pp. 1–10.

[10] A. Virtanen, ”Transitioning towards continuous development within an established business organization,” Aalto Univ., Espoo, Finland, Tech. Rep., 2017.

[11] F. Erich, C. Amrit, and M. Daneva, ”Cooperation between information system development and operations: a literature review,” in Proc. of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, 2017.

[12] M. Hilton, T. Tunnell, K. Huang, D. Marinov, and D. Dig, ”Usage, costs, and benefits of continuous integration in open-source projects,” in Proc. of the 31st IEEE/ACM International Conference on Automated Software Engineering (ASE 2016), 2016, pp. 426–437.

[13] V. Garousi, M. Felderer, and M. V. Mantylä, ”Guidelines for including grey literature and conducting multivocal literature reviews in software engineering,” Inf. Softw. Technol., vol. 106, pp. 101–121, 2019.

[14] P. Narang and P. Mittal, ”Performance Assessment of Traditional Software Development Methodologies and DevOps Automation Culture,” Eng. Technol. Appl. Sci. Res., vol. 12, no. 6, pp. 9726–9731, Dec. 2022.

[15] M. A. Akbar et al., ”A fuzzy analytical hierarchy process to prioritize the success factors of requirement change management in global software development,” J. Softw.: Evolution and Process, vol. 33, no. 2, e2292, 2021.

[16] G. Garbi, E. Incerto, and M. Tribastone, ”P: A Development Framework for Predicting Performance of Microservices by Design,” in 2023 IEEE 16th International Conference on Cloud Computing (CLOUD), 2023, pp. 979–8-3503-0481-7, pp. 178–188.

[17] Y. Liu and Y. Zhou, ”The Challenges and Mitigation Strategies of Using DevOps during Software Development,” 2017.

[18] M. Hüttermann, ”Introducing DevOps,” SpringerLink, 2012. [Online]. Available: https://link.springer.com/chapter/10.1007/978-1-4302-4570-4_2. [Accessed: 06-Jun-2021].