International Journal of Research and Applied Technology

2(2)(2022) 184-197
Journal homepage: https://ojs.unikom.ac.id/index.php/injuratech

DevOps Main Area and Core Capabilities Adopting DevOps in the Last Decade: A Systematic Literature Review

Jefri Zulkarnain*, Rizki Firdaus Mulya, Tika Pratiwi, Widi Pangestuti, Fahma Inti Ilmawati

[12345]Department of Information Technology, Amikom University Yogyakarta, Indonesia

Email: *jefrizulkarnain@students.amikom.ac.id

Abstract. DevOps is a software development and operations collaboration that uses frameworks such as continuous integration, microservices, continuous delivery, and continuous deployment to create an agile software development process. Automation, iteration, and continuous release and development are all DevOps principles. The purpose of this research is to learn about DevOps development over the last decade and how to potentially adopt the DevOps development process. The Systematic Literature Review (SLR) method is used in this study to locate, evaluate, and summarize relevant works published in the public domain between 2012 and 2022. The findings of this review will be used by researchers and practitioners as a source of information about DevOps core capabilities and main areas of DevOps from the last decade of DevOps adoption.

Keywords: DevOps, Area, and Capability.

1. Introduction

Today, DevOps bridges the gap between software development and implementation in large organizations. A major goal of DevOps projects is the use of frameworks such as continuous integration, microservices, continuous delivery, and continuous deployment for agile software development processes. [1] Part of the progress in this area is whether software is delivered to target users using dedicated mobile interfaces or software distribution networks, or delivered over the Internet via servers (i.e., as a service). It is related. In the fast-evolving Internet age, shorter distribution times are made possible by these new tendencies. In both the practitioner literature and the software development industry, formal work on DevOps has received a lot of attention. Companies spend more time and money developing and delivering high-quality software at a faster rate as the software industry becomes more competitive. Among the two continuous procedures intended to help businesses speed up the development and delivery of product features while preserving efficiency are continuous integration (CI) and continuous delivery (CDE). While CDE is worried about the possibility to distribute values to customers

International Journal of Research and Applied Technology

2(2)(2022) 184-197

Journal homepage: https://ojs.unikom.ac.id/index.php/injuratech

fast and effectively by employing automation as much as feasible, CI enables integrating work-in-progress numerous times a day. To enable automatic and consistent program deployment to production or client environments, it is crucial to fully include CDE practice, even though CI is the first step toward doing so (i.e., CD practice). There hasn't been any serious effort put into thoroughly assessing and synthesizing the research on continuous practices. To understand how CI, CDE, and CD practices interact with one another and what actions may be done to make the transition from one to the other successful and smooth, it is necessary to jointly explore these practices, resources, challenges, and activities [1].

This study seeks to understand DevOps development over the last decade by conducting a Systematic Literature Review of DevOps methods, tools, issues, areas, and capabilities.As the market becomes more competitive and there is greater pressure to meet customer needs, entrepreneurs feel a need to adapt to the current trends. They can't keep making customers wait for a program to debut for months or even a year before asking for feedback on how it works. Customers desire a connection that is active so they may provide continual feedback. To meet the demands of the present issues, more businesses must be agile and lean during the life cycle of product creation. For many years, businesses have frequently used process transformations (such as agile methods) in their application development. Nevertheless, assignments are frequently dropped during the whole software development process.

Development moves more quickly than teams will implement new technology. It is frequently argued that the supply line that influences others has the most vulnerability. They must thus address the cycle's flaws at any time. According to the yearly "State of the Art of DevOps" study, the proportion of DevOps teams increased from 19% in 2015 to 22% in 2016 and then to 27% in 2017 [2]. The absence of analytical investigations of DevOps' actual deployment outside of blog posts and surveys is a result of its increasing success. There aren't many case studies that look at DevOps in the context of continuous software development.

DevOps is a viable strategy, but companies still need to change their culture and philosophy. Several key principles support this idea: (1) Automation: Organizations need to automate processes, especially workflows, learn new code and configure infrastructure to reduce redundant effort and rework. (2) Repeat: Timeboxed sprints require writing small chunks of code to enable releases and subreleases, increasing the frequency and speed of deployments. (3) Continuous Release and Delivery: Continuous testing often helps you learn from mistakes and adapt to feedback to be more effective, save costs, and reduce implementation time. Here, silos between development, IT operations, and quality assurance are broken down by bringing teams together, fostering collaboration, and breaking down silos [3].

Figure 1 depicts one of the key DevOps principles: continuous release and development. The benefits of continuous release and deployment are numerous. Continuous deployment makes it possible to get early user and customer feedback. It permits regular and trustworthy releases, which raises client satisfaction [4].

Despite establishing DevOps as an organization that is fraught with difficulties, Camuto and Langerman argue that the biggest barrier to DevOps adoption is a lack of education in the field. As a result, organizational management is unable to alter the development methodology. Similar conceptual gaps exist between the development and operation teams, these gaps should be closed by tight cooperation and continual development and deployment processes. The most business struggle to create a "continuous development environment" because techniques like continuous integration and continuous testing, which can help to solve some issues, are lacking. Due to the varying levels of understanding between the two teams, the

International Journal of Research and Applied Technology

2(2)(2022) 184-197

Journal homepage: https://ojs.unikom.ac.id/index.php/injuratech

developers discourage working together because it could lead to resentment and other issues inside the organization. The availability of efficient tools is one of the main issues with DevOps adoption [5].

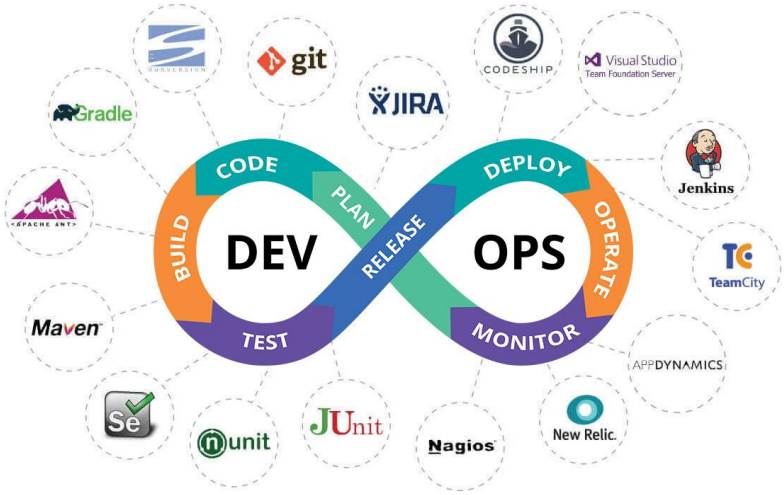

Figure 1. Key principles of DevOps

2. Research Methodology

This study refers to Kitchenhams for SLR [6], which is supplemented by Webster and Watson's centric approach [6], which includes the following steps:

Planning: The process of conducting a systematic review and developing a review protocol (i.e. plan) that defines the basic review procedure.

Conducting: Obtaining a study that will be the subject of an opinion using a previously constructed review protocol.

Reporting: The process of concluding a systematic review, which includes the review's opinion and disseminating the results to those who are interested.

2.1 Protocol Review

Protocol This study conducts an article review by conducting a literature review to identify the dataset to find out the proposed literature. To put it another way, we are looking for potentially different attributes or characteristics, so the enumeration used is as follows:

Search String.

For DevOps capabilities. DevOps AND (Capability OR Capabilities OR Practice)

For DevOps areas. DevOps AND (Area, Principles, View, Dimensions, and Perspective)

2.2 Data Sources

We collected from several sources related to the questions that we have previously described, this search we took from several sources:

- IJCRT

International Journal of Research and Applied Technology

2(2)(2022) 184-197

Journal homepage: https://ojs.unikom.ac.id/index.php/injuratech

- Research Gate
- Science Direct
- Springer
- IEEE
- Google Scholar

2.3 Research Question

A comprehensive review of the literature was used in this study, with the goal of determining the potential state of DevOps development over the last decade. The SLR method, which is used to gather experience from a variety of different studies in sequence, is the tool used to support evidence in other searches. Any comparisons in the DevOps question are also included, as different results reveal varying amounts of useful information. The following hypotheses are tested to answer three research questions as a method of conducting research and limiting the scope of study

RQ1: What are the main DevOps areas?

RQ2: What are the core capabilities of DevOps?

2.4 Conducting the Review

The review is carried out in the second stage of the SLR methodology according to the review protocol mentioned previously and analyzes the data sources that have been summarized. The authors of the [7] guidelines recommend that five data sources be considered when conducting a systematic review of the literature during the software planning process. We are looking for data sources that mention important potential studies. In this study, a review of criteria with string searches presented in the table found 92 papers without duplicates in the last decade 2012-2022. After that, read the abstract to get aspects relevant to the documentation question. Then from this car, produced 42 papers relevant to our research.

The number of paper findings is shown in Table 1. As demonstrated, the search aims to find all systematics that meet the aspects of the question that DevOps has already mentioned.

Table 1. Number of paper findings

Data Sources	Search Result	Final Selection
IJCRT	4	3
Research Gate	4	3

International Journal of Research and Applied Technology

2(2)(2022) 184-197

Journal homepage: https://ojs.unikom.ac.id/index.php/injuratech

Data Sources	Search Result	Final Selection
Science Direct	7	3
Springer	11	4
IEEE	22	9
Google Scholar	44	20
Total	**92**	**42**

3. Results and Discussion

We have identified 42 relevant references to answer the existing research questions in this section, which is the final step of the SLR methodology. We identified three main topics where there are areas and capabilities. The findings are analyzed to increase understanding of DevOps, which is a prelude to adopting DevOps.

3.1 DevOps Area RQ1

Within that section, we will answer the first Research Questions, which are: What are the key DevOps areas? Furthermore, it presents findings from a literature review that aims to identify the areas most frequently used in DevOps over the last decade. Table 2 presents the main findings regarding the DevOps dimension. Since there are no standards for categorization and related processes in DevOps [8], the author will proceed to detail the most frequent areas in DevOps practice.

The results obtained after analyzing table 2 and only Automation, Technology, People, Culture, and Service that exceed 10 findings, the authors decide to describe areas A1 to A5. The description of each area will be explained from several reviews of previous findings.

3.1.1 Automation

It can be expected that manual work will be automated and the productivity of existing operations will increase. For example, automation streamlines cloud applications and improves the efficiency of information technology departments [9]. As we all know, DevOps has impacted software quality and automation is the number one factor in improving software quality [10]. In addition, there is automation that takes advantage of the potential for repeated and successful use on multiple servers for server configuration and management [11].

International Journal of Research and Applied Technology

2(2)(2022) 184-197

Journal homepage: https://ojs.unikom.ac.id/index.php/injuratech

3.1.2 Technology

The organization's technology and cultural behavior should influence the relationship with the organization or department in the overall strategic alignment of business and IT [12]. Support for some technology enablers such as B. Introducing automation pipelines and cross-functional organizational structures are critical to achieving the expected benefits of DevOps [13].

3.1.3 People

Although DevOps is a pragmatic approach, it still requires a shift in organizational mindset and culture [3]. DevOps encompasses the entire lifecycle/process of delivering a solution or service and necessitates organizational and cultural change [14]. Case studies, for example, were conducted at three different organizations to validate the developed model. The findings indicate that RMDevOps is useful for evaluating and improving DevOps practices in software organizations [5].

3.1.4 Culture

DevOps culture is a step for developers in producing good products for customers [15]. Feedback from customers can be integrated into applications faster in a DevOps culture which keeps customers more engaged and satisfied [16]. Furthermore, DevOps is a cultural and organizational change that is incorporated into the environment by focusing on what is required rather than a pump-and-dump strategy and delivering long-term value [17].

3.1.5 Service

Microservice is one of the architectural frameworks used as microservices in the development of more modern cloud software. With microservices, business processes can be easily automated and any necessary changes in business processes can be implemented by modifying the associated microservices [18]. A DevOps environment was also created to develop microservice-based applications such as Predictive Car Maintenance (PCM) applications using project templates and shared pipeline configurations [19].Table 2. DevOps areas literature review

Table 2. DevOps areas literature review

ID	Area	References	# of References
A1	Automation	[18][20][13][11][21][22][23][4][24][25][26][27][28][17][10][15][29][30][16][9][10][31][32][3]	24

International Journal of Research and Applied Technology

2(2)(2022) 184-197

Journal homepage: https://ojs.unikom.ac.id/index.php/injuratech

ID	Area	References	# of References
A2	Technology	[33][34][35][13][19][21][22][22][23][36][14][4][24][28][12][37][38][10][5][39][40][9][10][31][32]	24
A3	People	[18][34][13][19][23][14][28][12][17][38][41][15][5][29][39][42][30][16][31][32][3]	21
A4	Culture	[35][14][26][27][12][17][41][10][15][5][16][3]	12
A5	Services	[18][33][19][21][36][14][25][39][42][40][32]	11
A6	Measurement	[43][26][28][37][44][38][10][5][42]	9
A7	Process	[18][34][45][25][27][41][15][29]	8
A8	Sharing	[26][10]	2

3.2 DevOps Capability RQ2

In this section, we identify 39 references related to capability DevOps and found 16 capability DevOps (C1-C16) which are detailed in table 3. We define the capability pool based on the previous literature study related to capability DevOps, literature study has been conducted by [6], [46], [47], [48], [49], and [50]. Capability DevOps is a key activity in software and service engineering that entails planning, development, testing, and deployment. These activities are carried out on a continuous basis while paying attention to feedback from other activities [47].

A capability is defined differently from one study to another based on the concept of capability but has the same meaning [6]). The author has grouped the capabilities in table 3 and analyzed them in table 3 it was found that in C1 and C2 there is a very large gap, we also found between C5 and C6 there is a gap but not too far but after C6 to C16 the trend of the

International Journal of Research and Applied Technology

2(2)(2022) 184-197
Journal homepage: https://ojs.unikom.ac.id/index.php/injuratech

emergence of capability continues to decrease. Based on this, we define C1 - C4 capability as the core capability of the 39 references that we have reviewed.

Table 3. DevOps capability literature review

ID	Capability	Reference	# of References
C1	Continuous Integration	[18][20][34][35][45][19][21][23][36][14][4][43][26][27][28][12][17][44][10][15][5][29][30][16][9][32][3]	27
C2	Continuous Deployment	[18][34][35][45][21][22][14][4][24][43][41][5][16][9][32][3]	16
C3	Test automation	[18][20][34][45][13][24][43][25][28][10][9][10][31][32][3]	15
C4	Continuous delivery	[20][13][21][14][43][26][28][41][15][5][29][16][10][3]	14
C5	Continuous planning	[34][35][45][21][22][14][4][26][17][41][32]	11
C6	Monitoring automation	[18][20][34][13][27][28][37][17][30]	10
C7	Continuous Testing	[11][19][21][22][4][44][15][29][16]	9
C8	Infrastructure as code	[18][13][11][25][37][15][42][16][3]	9
C9	Prototyping application	[20][11][19][24][25][44][39][42]	8

International Journal of Research and Applied Technology

2(2)(2022) 184-197

Journal homepage: https://ojs.unikom.ac.id/index.php/injuratech

ID	Capability	Reference	# of References
C10	Continuous Monitoring	[21][22][23][4][40][9][31]	7
C11	Automated deployment	[25][27][12][17][31]	5
C12	ContinuousFeedback	[35][22][4][39][32]	5
C13	Feedback Loops between Dev and Ops	[45][13][43]	3
C14	Stakeholder Participation	[34][24][16]	3
C15	Change Management	[4]	1
C16	Process Standardisation	[12]	1

4. Conclusion

In this study, SLR was conducted to find out the development of DevOps in the last decade by conducting a Systematic Literature Review of the area, and the DevOps capabilities that were the determinants and contributions at the time of implementation. A total of 42 related references have been identified in answering research questions 1 and 2. RQ1, areas have been obtained which include automation, technology, people, culture, and service, and have been defined. RQ2, capabilities have been identified which include Continuous Integration, Continuous Deployment, Test automation, and Continuous delivery. Based on this, the main objective of this study to answer the research question has been achieved. In this study we certainly have limitations, we cannot possibly avoid bias due to the lack of references we get from electronic databases, limiting certain languages certainly cannot make us present strong conclusions about area and capability. Future research will further deepen the findings of the identified areas and capabilities and expand the topics on concepts, practices, tools, benefits, and challenges in adopting DevOps.

International Journal of Research and Applied Technology

2(2)(2022) 184-197

Journal homepage: https://ojs.unikom.ac.id/index.php/injuratech

References

[1] Ståhl, D., & Bosch, J. (2013). Modeling continuous integration practice differences in industry software development. In *Accelerating Digital Transformation* (pp. 23-49). Springer, Cham.

[2] Lee, D. E., Lim, T. K., & Arditi, D. (2009). Automated stochastic quality function deployment system for measuring the quality performance of design/build contractors. *Automation in Construction*, **18**(3), 348-356.

[3] Yarlagadda, R. T. (2018). Understanding DevOps & bridging the gap from continuous integration to continuous delivery. *Understanding DevOps & Bridging the Gap from Continuous Integration to Continuous Delivery', International Journal of Emerging Technologies and Innovative Research (www. jetir. org), ISSN*, 2349-5162.

[4] Gokarna, M. (2021). DevOps phases across Software Development Lifecycle.

[5] Rafi, S., Yu, W., Akbar, M. A., Mahmood, S., Alsanad, A., & Gumaei, A. (2021). Readiness model for DevOps implementation in software organizations. *Journal of Software: Evolution and Process*, **33**(4), e2323.

[6] Teixeira, D., Pereira, R., Henriques, T. A., Silva, M., & Faustino, J. (2020). A systematic literature review on DevOps capabilities and areas. *International Journal of Human Capital and Information Technology Professionals (IJHCITP)*, **11**(3), 1-22.

[7] Gasparaite, M., Naudziunaite, K., & Ragaisis, S. (2020). Systematic literature review of devops models, vol. 1266 CCIS.

[8] Silva, M. A., Faustino, J. P., Pereira, R., & Mira da Silva, M. (2018). Productivity gains of DevOps adoption in an IT team: a case study.

[9] Mohammad, S. M. (2018). Streamlining DevOps automation for Cloud applications. *International Journal of Creative Research Thoughts (IJCRT), ISSN*, 2320-2882.

[10] Mohammad, S. M. (2018). Improve software quality through practicing devops automation. *Sikender Mohsienuddin Mohammad," IMPROVE SOFTWARE QUALITY THROUGH PRACTICING DEVOPS AUTOMATION", International Journal of Creative Research Thoughts (IJCRT), ISSN*, 2320-2882.

[11] Khumaidi, A. (2021). IMPLEMENTATION OF DEVOPS METHOD FOR AUTOMATION OF SERVER MANAGEMENT USING ANSIBLE. *Jurnal Transformatika*, **18**(2), 199-209.

[12] Hart, M., & Burke, J. (2020). AN EXPLORATORY STUDY ON THE DEVOPS IT ALIGNMENT MODEL. *Interdisciplinary Journal of Information, Knowledge & Management*, *15*.

[13] Senapathi, M., Buchan, J., & Osman, H. (2018, June). DevOps capabilities, practices, and challenges: Insights from a case study. In *Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018* (pp. 57-67).

[14] Demchenko, Y., Zhao, Z., Surbiryala, J., Koulouzis, S., Shi, Z., Liao, X., & Gordiyenko, J. (2019, September). Teaching DevOps and cloud based software engineering in university curricula. In *2019 15th International Conference on eScience (eScience)* (pp. 548-552). IEEE.

[15] Priyadarsini, K., Raj, E. F. I., Begum, A. Y., & Shanmugasundaram, V. (2020). Comparing DevOps procedures from the context of a systems engineer. *Materials Today: Proceedings*.

[16] Soni, M. (2015, November). End to end automation on cloud with build pipeline: the case for DevOps in insurance industry, continuous integration, continuous testing, and

continuous delivery. In *2015 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)* (pp. 85-89). IEEE.

[17] Katal, A., Bajoria, V., & Dahiya, S. (2019, March). DevOps: Bridging the gap between Development and Operations. In *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)* (pp. 1-7). IEEE.

[18] Sen, A., & Skrobot, I. (2021). Implementation of DevOps paradigm to deployment and provisioning of microservices. *Issues in Information Systems*, **22**(1), 136-148.

[19] Throner, S., Hütter, H., Sänger, N., Schneider, M., Hanselmann, S., Petrovic, P., & Abeck, S. (2021, August). An Advanced DevOps Environment for Microservice-based Applications. In *2021 IEEE International Conference on Service-Oriented System Engineering (SOSE)* (pp. 134-143). IEEE.

[20] Arulkumar, V., & Lathamanju, R. (2019). Start to finish automation achieve on cloud with build channel: By DevOps method. *Procedia Computer Science*, **165**, 399-405.

[21] Subramanya, R., Sierla, S., & Vyatkin, V. (2022). From DevOps to MLOps: Overview and Application to Electricity Market Forecasting. *Applied Sciences*, **12**(19), 9851.

[22] Reine De Reanzi, S., & Ranjit Jeba Thangaiah, P. (2021). A survey on software test automation return on investment, in organizations predominantly from Bengaluru, India. *International Journal of Engineering Business Management*, *13*, 18479790211062044.

[23] Cois, C. A., Yankel, J., & Connell, A. (2014, October). Modern DevOps: Optimizing software development through effective system interactions. In *2014 IEEE international professional communication conference (IPCC)* (pp. 1-7). IEEE.

[24] P. Maragathavalli, "Automation Pipeline and Build Infrastructure using DevOps," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 8, no. 11, pp. 882–886, Nov. 2020, doi: 10.22214/ijraset.2020.32105.

[25] A. Amrit, PJ Akhil, Pranjal, R. Raj N, and BS Shylaja, "MICROSERVICES EVOLVING DEVOPS PIPELINES," *Int. Res. J. Mod. Eng. Technol. Sci.*, Sep. 2022, doi: 10.56726/irjmets29808.

[26] Hemon, A., Fitzgerald, B., Lyonnet, B., & Rowe, F. (2019). Innovative practices for knowledge sharing in large-scale DevOps. *IEEE Software*, *37*(3), 30-37.

[27] Lwakatare, L. E., Karvonen, T., Sauvola, T., Kuvaja, P., Olsson, H. H., Bosch, J., & Oivo, M. (2016, January). Towards DevOps in the embedded systems domain: Why is it so hard?. In *2016 49th hawaii international conference on system sciences (hicss)* (pp. 5437-5446). IEEE.

[28] Kiran and S. Raju, "Performance Analysis of Automation Monitoring System shifting from DevOps to DevSecOps," *Int. J. Emerg. Trends Eng. Res.*, vol. 8, no. 9, pp. 5128–5134, Sep. 2020, doi: 10.30534/ijeter/2020/40892020.

[29] Süß, J. G., Swift, S., & Escott, E. (2022). Using DevOps toolchains in Agile model-driven engineering. *Software and Systems Modeling*, 1-16.

[30] Schaefer, A., Reichenbach, M., & Fey, D. (2013). Continuous integration and automation for DevOps. In *IAENG Transactions on Engineering Technologies* (pp. 345-358). Springer, Dordrecht.

[31] Surya, L. (2017). AI in information technology and its future in the United States. *International Journal of Creative Research Thoughts (IJCRT), ISSN*, 2320-2882.

[32] Yarlagadda, R. T. (2017). Implementation of DevOps in healthcare systems. *Implementation of DevOps in Healthcare Systems', International Journal of Emerging Technologies and Innovative Research (www. jetir. org), ISSN*, 2349-5162.

[33] Casale, G., Chesta, C., Deussen, P., Di Nitto, E., Gouvas, P., Koussouris, S., ... & Zhao, Z. (2016). Current and future challenges of software engineering for services and applications. *Procedia computer science*, **97**, 34-42.

[34] Sen, A., Falter, S., & Mayer, N. (2022). Using DevOps paradigm to deploy web applications. *Issues in Information Systems*, **23**(4).

[35] M. S. Namose and S. D. Patil, "Standard DevOps Pipeline," *Int. J. Comput. Sci. Eng.*, vol. 7, no. 3, pp. 641–646, Mar. 2019, doi: 10.26438/ijcse/v7i3.641646.

[36] Cukier, D. (2013, October). DevOps patterns to scale web applications using cloud services. In *Proceedings of the 2013 companion publication for conference on Systems, programming, & applications: software for humanity* (pp. 143-152).

[37] Adochiei, F. C., Ciucu, R., Adochiei, I. R., Argatu, F. C., Larco, C., & Casian, M. (2019). A Data Analysis platform for high altitude bio monitoring using DEVOPS principles and time series machine learning. *Journal of Advances in Technology and Engineering Research*, **5**(1), 18-24.

[38] Moore, J., Kortuem, G., Smith, A., Chowdhury, N., Cavero, J., & Gooch, D. (2016, May). DevOps for the urban IoT. In *Proceedings of the Second International Conference on IoT in Urban Space* (pp. 78-81).

[39] Magoutis, K., Papoulas, C., Papaioannou, A., Karniavoura, F., Akestoridis, D. G., Parotsidis, N., ... & Stephanidis, C. (2015). Design and implementation of a social networking platform for cloud deployment specialists. *Journal of Internet Services and Applications*, **6**(1), 1-26.

[40] John, W., Meirosu, C., Pechenot, B., Sköldström, P., Kreuger, P., & Steinert, R. (2015, September). Scalable software defined monitoring for service provider devops. In *2015 Fourth European Workshop on Software Defined Networks* (pp. 61-66). IEEE.

[41] Mohamed, S. I. (2015). DevOps shifting software engineering strategy Value based perspective. *IOSR Journal of Computer Engineering Ver. IV. https://doi. org/10.9790/0661-17245157*.

[42] Shi, Z., Ivankovic, V., Farshidi, S., Surbiryala, J., Zhou, H., & Zhao, Z. (2022). AWESOME: an auction and witness enhanced SLA model for decentralized cloud marketplaces. *Journal of Cloud Computing*, **11**(1), 1-25.

[43] Z. Shoieb, L. Abdel-Hamid, and M. A. Abdel-Fattah, "Enhancing Software Deployment Release Time Using DevOps Pipelines," *Int. J. Sci. Eng. Res.*, vol. 11, no. 3, 2020, [Online]. Available: http://www.ijser.org

[44] Mathieson, J. T., Mazzuchi, T., & Sarkani, S. (2020). The systems engineering DevOps lemniscate and model-based system operations. *IEEE Systems Journal*, **15**(3), 3980-3991.

[45] Babar, Z., Lapouchnian, A., & Yu, E. (2015, November). Modeling DevOps deployment choices using process architecture design dimensions. In *IFIP Working Conference on The Practice of Enterprise Modeling* (pp. 322-337). Springer, Cham.

[46] Teixeira, D., Pereira, R., Henriques, T., Silva, M. M. D., Faustino, J., & Silva, M. (2020). A maturity model for DevOps. *International Journal of Agile Systems and Management*, **13**(4), 464-511.

[47] Harzl, A., & Slany, W. (2015, May). Combining Kanban and FOSS: Can it work?. In *Agile Processes in Software Engineering and Extreme Programming: 16th International Conference, XP 2015, Helsinki, Finland, May 25-29, 2015, Proceedings* (Vol. 212, p. 352). Springer.

[48] Bou Ghantous, G., & Gill, A. (2017). DevOps: Concepts, practices, tools, benefits and challenges. *PACIS2017*.

[49] R. Jabbari, N. Bin Ali, K. Petersen, and B. Tanveer, "What is DevOps? A systematic mapping study on definitions and practices," *ACM Int. Conf. Proceeding Ser.*, vol. 24-May-201, 2016, doi: 10.1145/2962695.2962707.

[50] Stahl, D., Martensson, T., & Bosch, J. (2017). Continuous practices and DevOps: beyond the buzz, what does it all mean? In Software Engineering and Advanced Applications (SEAA), 2017 43rd Euromicro Conference (pp. 440-448). IEEE.