

Article

Factor Prioritization for Effectively Implementing DevOps in Software Development Organizations: A SWOT-AHP Approach

Noor Mohammed Noorani ¹, Abu Taha Zamani ², Mamdouh Alenezi ³, Mohammad Shameem ^{4,*} and Priyanka Singh ⁵

¹ Department of Computer Science, UpGrad Education Private Limited, Mumbai 400018, Maharashtra, India

² Department of Computer Science, Northern Border University, Arar 73211, Saudi Arabia

³ Software Engineering and Disruptive Innovation (SEDI), College of Computer and Information Sciences, Prince Sultan University, Riyadh 11586, Saudi Arabia

⁴ Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Guntur 522302, Andhra Pradesh, India

⁵ Department of Computer Science and Engineering, SRM University, Amaravati 522240, Andhra Pradesh, India

* Correspondence: shameem.ism@gmail.com

Abstract: DevOps (development and operations) is a collective and multidisciplinary organizational effort used by many software development organizations to build high-quality software on schedule and within budget. Implementing DevOps is challenging to implement in software organizations. The DevOps literature is far away from providing a guideline for effectively implementing DevOps in software organizations. This study is conducted with the aim to develop a readiness model by investigating the DevOps-related factors that could positively or negatively impact DevOps activities in the software industry. The identified factors are further categorized based on the internal and external aspects of the organization, using the SWOT (strengths, weaknesses, opportunities, threats) framework. This research work is conducted in three different phases: (1) investigating the factors, (2) categorizing the factors using the SWOT framework, and finally, (3) developing an analytic hierarchy process (AHP)-based readiness model of DevOps factors for use in software organizations. The findings would provide a readiness model based on the SWOT framework. The proposed framework could provide a roadmap for organizations in the software development industry to evaluate and improve their implementation approaches to implement a DevOps process.

Keywords: AHP; DevOps; SWOT; factors



Citation: Noorani, N.M.; Zamani, A.T.; Alenezi, M.; Shameem, M.; Singh, P. Factor Prioritization for Effectively Implementing DevOps in Software Development Organizations: A SWOT-AHP Approach. *Axioms* **2022**, *11*, 498. <https://doi.org/10.3390/axioms111100498>

Academic Editors: Palle E.T. Jorgensen and Hsien-Chung Wu

Received: 23 July 2022

Accepted: 14 September 2022

Published: 23 September 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent years, software development has become a complex task due to increasing demand for software products, the competitive market and complex software development activities [1,2]. Competitive and on-demand product delivery emerged as the most challenging factor for software organizations to release a commercially viable product within time and budget [3]. Therefore, it is important for software development companies to address the difficulties related to product quality and timeliness release by enhancing organizational practices and processes. A mature software process can help software development practitioners execute software development activities successfully [4,5]. There are a number of models and techniques that have been formed for the successful administration and management of software development activities [6,7]. Development and operations (DevOps) has been considered one of the most significant processes that aims to enhance and expedite the delivery of business value by facilitating efficient cooperation between development and operations silos [2,7,8].

DevOps refers to a collection of collaborative and diverse efforts utilized to carry out software development tasks [8,9]. The concept of DevOps was initially introduced

to improve and accelerate business delivery by facilitating the effective coordination between development and operation teams [7,8]. According to Leite et al. [9], “DevOps is a multicultural movement within the organization to accelerate the delivery of business use cases by making the collaboration between development (Dev) and Operation (Ops) teams”. Lucy Ellen Lwakatare et al. [3] stated that DevOps is, “a cultural movement by making collaboration between development and operations associates to improve and accelerate the business values”. DevOps was first presented to manage discrepancies between development and operations to respond quickly to client needs [10,11].

According to the State of DevOps Report [5], a majority of software development organizations are heading toward the DevOps process to develop a quality product by capturing the characteristics of continuous deployment and integration. Based on the DevOps Report [5], software development and deployment activities using the DevOps practices are 30 times faster than those using the traditional software development processes.

Different studies have presented different approaches and frameworks for efficiently and effectively implementing continuous integration and deployment practices to improve the coordination between development and operations [12–14]. The DORA (DevOps Research and Assessment) [8] platform designed an assessment tool for the assessment of the software product delivery value stream, and an ontology-based DevOps maturity model. Despite the availability of these frameworks, DevOps practitioners and researchers still face significant issues for effectively managing the automation pipeline between the development and operation silos [6,15,16]. Different studies have focused on investigating challenges and motivators for DevOps implementation, but how these factors are managed is an area that needs to be explored more. Therefore, there is a need for a readiness model to tackle the identified DevOps factors and provide best practices to manage them for further improvements.

Kerzazi and Adams [10] reported that the DevOps developers need to have technological infrastructures for pipeline optimization between Dev and Ops teams. Moreover, they have also discussed an effective implementing and integrating pipeline deployment, as the existing software release is challenging and requires a diverse, skillful force. Various studies have focused on the challenging factors of DevOps implementing practices; however, research on the topic of identifying and discussing the success factors is still limited [2,4,11]. In one of the studies, Khan et al. [4] identified the various challenges for implementing DevOps. They identified 16 challenging elements from the available literature which were empirically validated through the survey method. They classified the identified items into three categories, i.e., people, business and change, based on the process improvement framework. In a recent study, Rafi, Saima et al. [11] conducted a study and developed a readiness model in which they have focused on various challenges. The limitations of this study were that they focused on the motivators; now, we need to focus more on DevOps implementation.

1.1. Problem Statement

The DevOps concepts were adopted from lean development, which is a known approach in agile software development and the manufacturing industry [1,2]. The continuous deployment practice of DevOps enables the development teams to implement the most important product features frequently [17]. Currently, DevOps plays a fundamental role in software organizations that have increased its use to rapidly deploy changes hundreds of times per day, have a fast time to market and carry out relentless experimentation by making the development practices more agile as compared to the processes which do not follow the continuous integration and continuous deployment practices [18–21]. According to Kim et al. [12], software development organizations could easily lose the competition due to delays in the launching of the product. Various studies discussed the significance of DevOps practices in software development. Various studies [4,22–25] have discussed the significance of DevOps culture in software development organizations, but still, the DevOps literature is far from presenting a readiness model for effectively managing the

DevOps activities. Therefore, we need significant and timely research to highlight the most significant areas that could negatively/positively impact the DevOps activities.

1.2. Research Questions

The following research questions are formulated to fill the discussed research gap and focus on the key research goals.

RQ1: What are the factors, as discussed in the literature, related to effectively implementing DevOps in the software industry?

RQ2: How can the investigated DevOps factors be classified based on the SWOT framework?

RQ3: How can the relative importance of the identified factors and their categories be evaluated using the AHP technique?

RQ4: How can factor-based readiness be developed for DevOps implementation in software organizations?

2. Related Work

Currently, an increasing number of software organizations are heading toward DevOps to obtain the advantages in terms of early product release, high project visibility among the team members and minimum documentation for both client and vendor organizations to develop a high-quality product that can satisfy customer requirements. In recent years, a significant focus has been given to identifying various challenges for effectively implementing DevOps, which has encouraged software organizations to deploy DevOps culture. Lwakatare [3] conducted a literature survey and interviews with DevOps practitioners to identify the elements underpinning the phenomenon of DevOps. The four main dimensions of DevOps identified by them were collaboration, automation, measurement, and monitoring. Toivakka, H. [26] identified seven DevOps-related topics, including collaboration culture, automation, measurement, sharing, services, quality assurance and governance. The authors studied six organizations to investigate how DevOps is being implemented in practice. Humble, J.; Kim, G. [12] formulated a theory on DevOps adoption. They discovered a relationship between seven aspects of DevOps adoption: agility, automation, collaborative culture, continuous measurement, quality assurance, resilience, sharing and transparency. Nicole Forsgren et al. [5] conducted a hierarchical cluster analysis using a sample of 7522 DevOps professionals, examining how throughput and stability measures work together, and created three different software delivery performance profiles of development settings. Keren Marieke Heine [27] introduced multivariate regression models that can predict the effectiveness of DevOps implementations in combination with other software development frameworks and practices. The purpose of this study was to give quantitative data to IT managers to help them make decisions on DevOps implementation, other methodologies that complement DevOps and commonly utilized strategies for good project management. Smeds [28] used attribute capabilities, culture and technology to define the DevOps phenomenon.

Regardless of the significance of the problems for effectively implementing DevOps, little empirical research has been witnessed to develop techniques and frameworks that could address the factors faced by the DevOps practitioners [2,29,30]. Research in this field is expected to provide valuable insights into the views of DevOps practitioners for the readiness of their organizations to successfully implement DevOps in software organizations.

3. Research Methodology

The methods or strategies used to find, select, process and evaluate information on a topic in order to provide trustworthy research results are referred to as research methodology [30]. However, the research methods refer to the tools which are used to collect the data, such as interviews or questionnaire surveys. The research methods can either be quantitative, qualitative, or mixed methods [31,32]. The quantitative research is

based on figures or numbers, whereas the qualitative methods are based on the positivism paradigm [32–37].

To answer the research questions as discussed in Section 1.2, we have used both qualitative and quantitative research methods (i.e., mixed research method), which comprised the systematic literature review (SLR), and a survey instrument for implementing the analytic hierarchy process (AHP) analysis [36]. In the mixed research method, both qualitative and quantitative data are concurrently collected in a single study [36]. Gregar [37] reported that a mixed research method facilitates in overcoming the limitations of both the qualitative and quantitative methods. Walker et al. [38] reported that the qualitative and quantitative methods are complementary to each other. The qualitative data can be converted into quantitative data using the coding scheme [4,38,39]. In their empirical studies, Niazi et al. [39] and Khan et al. [34] discussed that the coding scheme converts the qualitative data to quantitative data without impacting the subjectivity or objectivity of the data. Furthermore, Brannen, J. and Coram, T. [36] discussed that the conversion of quantitative data to qualitative data is also possible without impacting objectivity. The use of case study methods is one of the examples in which the qualitative data are analyzed by converting them into quantitative research [38]. Based on a brief overview of both qualitative and quantitative methods, survey-based research can enable the use of well-founded decisions. In this study, we have used a mixed research method for the data collection and analysis. The qualitative data were collected from the SLR, which were converted into quantitative data (i.e., frequency) in order to perform the statistical analysis. The research methodology used in our study is shown in Figure 1.

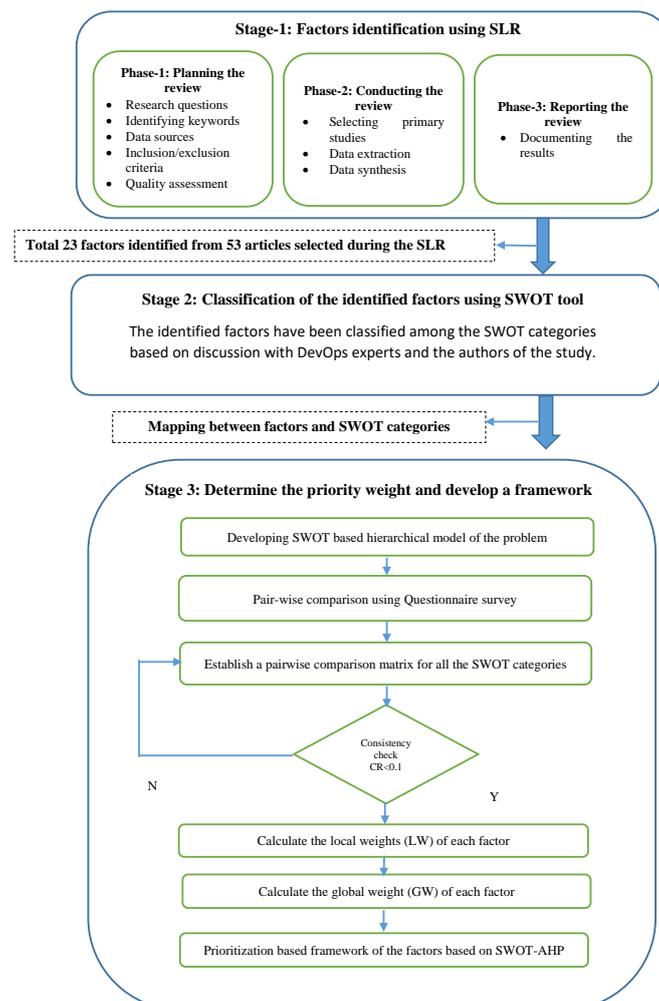


Figure 1. Proposed research methodology.

3.1. Stage 1: Systematic Literature Review (SLR)

An SLR is a systematic procedure that uses inclusion and exclusion criteria to study, categorize and assess the available literature in a certain research field [40–42]. Kitchenham [40] reported the SLR process in three main phases, which includes planning, conducting and reporting the review, as shown in Figure 2.

The SLR approach is widely accepted by researchers in different domains [17,18]. For example, Arjumand et al. [42] used the SLR to analyze the impact of software engineers' personality on project performance. Jéssyka et al. [43] followed the SLR to integrate the requirements for engineering and a safety process aspect. Moreover, Khan Siffat and Mohammad [44] conducted the SLR process to identify the intercultural challenges which affect the software team performance. All the authors were involved in all three systematic phases of SLR.

3.1.1. Phase 1: Review Planning

The objectives of this study are to investigate the factors which positively/negatively impact DevOps practices when implementing them in software organizations. This study addresses the research questions discussed in Section 1.

Primary Data Sources

Based on the experience of previous research studies and suggestions given by Khan et al. [18], the digital libraries were considered as the data sources to collect the primary data, as shown in Table 1.

Table 1. Data sources for SLR data collection.

Digital Library	URL
"ACM Digital Library"	" http://dl.acm.org "
"IEEE Explorer"	" http://ieeexplore.ieee.org "
"John Wiley"	" https://onlinelibrary.wiley.com "
"Science Direct"	" https://www.sciencedirect.com "
"Springer Link"	" https://link.springer.com "
"Google Scholar"	" https://scholar.google.com "

Although the above-mentioned digital libraries are different in terms of searching mechanisms, the authors then performed the search strategy based on their searching mechanisms.

Search Strategy

To collect the primary data sources for the SLR, the search strategy was performed in four steps, as suggested by Kitchenham [40]:

(a) **Develop the major keywords from population, intervention and outcomes.**

In the first step, the searching terms were constructed based on the population, intervention, outcomes and experimental design [39,40].

- **Population:** implementation of DevOps in software development.
- **Interventions:** factors which impact DevOps practices positively/negatively.
- **Outcomes:** list out the identified factors.
- **Experimental design:** systematic literature review.

(b) **Find the synonyms and the words having similar meanings to the above-described keywords.**

The academic databases were used to validate the following keywords, and the following synonyms show the potential relevance in the topic [7,9,11]:

DevOps implementation = DevOps, development and operations, Dev and Ops teams, continuous delivery, continuous testing, continuous deployment.

Factors = challenges, issues, barriers, obstacles, risks, success factor, motivators, positive factor.

(c) **Develop Boolean expressions.**

We combined the keywords into search strings using the Boolean “OR” and “AND” operators. The following string was used to search the digital repositories:

(“Difficulties” OR “Challenges” OR “issues” OR “barriers” OR “obstacles” OR “risks”) AND (“Success factors” OR “Motivators” OR “Drivers”) AND (“DevOps” OR “Development and Operation” OR “Continuous deployment” OR “Continuous testing” OR “Continues development and Operation”).

(d) **Verification of Boolean expression using digital libraries.**

The digital repositories were searched using the above strings.

Inclusion Criteria

The primary studies were selected using the following inclusion criteria:

- The paper must be written in English and be available as a full-text article.
- Articles must be reported in journals, conferences, magazines and book chapters.
- Studies must be focused on the challenges/success factors in DevOps implementation.

Exclusion Criteria

- The primary studies were selected using the following exclusion criteria:
- Studies that do not relate to DevOps factors.
- Articles written in languages other than English.
- Graduation project, master thesis and Ph.D. thesis all remain unpublished.
- Civil engineering, for example, is a study that is unrelated to software development.
- Redundant manuscripts.

Quality Assessment Criteria for Study Selection

The criteria were used to judge the quality of the selected articles discussed [40]. Five questions, as mentioned in Table 2, were used to assess how effectively the papers were chosen. To evaluate individual criteria scores, the following point system was used: Yes (Y) = 1 point, Partial (P) = 0.5 points, and No (N) = 0 points. The overall quality score for each selected article was calculated by summing up the values of the five particular criteria. As a result, the total quality score for each selected paper ranged from 0 (extremely poor) to 5 (very good). An article with a quality score greater than 3 was regarded to be of high quality and was included in our SLR study. The list of selected articles along with their quality score are provided in Appendix A.

Table 2. Quality assessment criteria.

Questions for QA	Score
Are there readers able to understand the motive of research?	No = 0, Partial = 0.5, Yes = 1,
Do the findings of the study clearly discusses about the DevOps?	No = 0, Partial = 0.5, Yes = 1,
Does the study discuss any challenge/success factor in the DevOps?	No = 0, Partial = 0.5, Yes = 1,
Are the logical arguments well-presented and justified in the articles?	No = 0, Partial = 0.5, Yes = 1,
Are the results related to the research questions?	No = 0, Partial = 0.5, Yes = 1,

3.1.2. Phase 2: Conducting the Review

Selecting the Primary Data

The entire process of choosing relevant articles was carried out in four steps:

Step 1: After applying the developed search string, as mentioned in Section 3.1.1 (c) a total of 2066 articles were displayed over the selected digital databases.

Step 2: The papers selected in phase 1 were evaluated after reading the titles and abstract. There was a total of 607 papers that were filtered out in phase 2 after applying the inclusion and exclusion criteria, as discussed in Section 3.1.1 (d).

Step 3: After reading the introduction, conclusion and the result sections of the articles selected in step 2, a total of 107 relevant publications were extracted.

Step 4: Based on the reading of the full text of the papers selected in phase 3, the total number of relevant articles was finally selected after applying the quality criteria. In this phase, all 53 articles were finally evaluated based on the quality score. Finally, the selected papers were subjected to a quality evaluation criterion. If the quality score is found to be ≥ 3 , an article was considered as a quality paper. In Table 3, the following four scanning stages are discussed. Appendix A contains a final list of selected articles together with their quality score.

Table 3. Results of selecting articles.

Digital Libraries	1st Phase	2nd Phase	3rd Phase	4th (Final Phase)	Percentage of Final Selected Papers
ACM	95	58	19	02	04
IEEE	396	102	31	21	40
John Wiley	93	51	12	02	04
Science Direct	314	157	14	07	13
Springer	412	98	21	10	19
Google Scholar	756	141	10	11	20
Total	2066	607	107	53	100 %

Data Synthesis

A data synthesis was conducted and a list of factors including challenges/success factors was created from 53 finally selected articles.

3.1.3. Phase 3: Reporting Review Process

Distribution of Final Selected Articles According to Types

There are 20 journal articles, 31 conference papers, 1 master's thesis and 1 book chapter among the 53 finally selected publications (Figure 2).

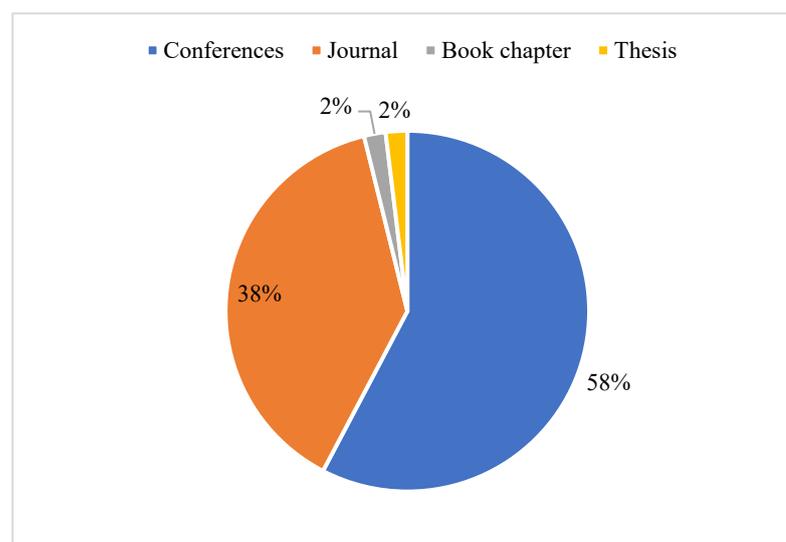


Figure 2. Types of the studies.

Temporal Distribution of the Published Paper

A summary of the primary selected papers along with their publication years are shown in Figure 3. The maximum number of articles was published in the years of 2021–2022, which indicates an increasing interest in the research related to DevOps.

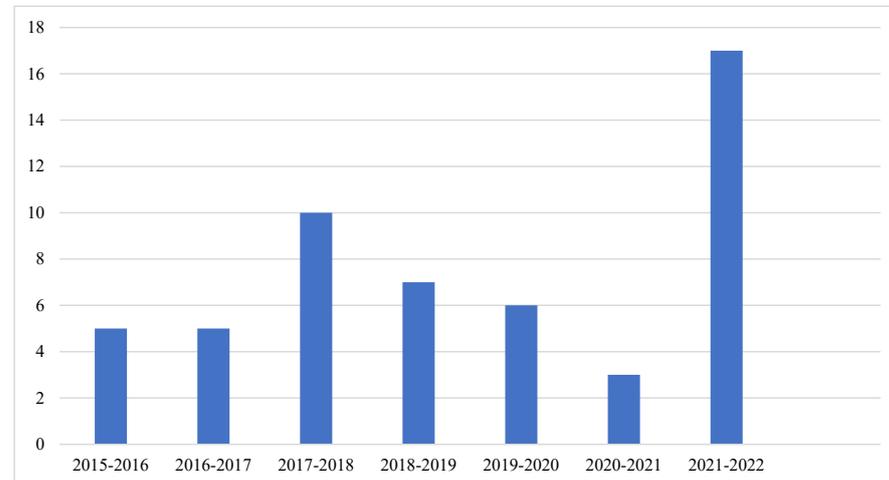


Figure 3. Temporal distribution of selected studies.

4. Findings from SLR

4.1. Findings Obtained from SLR

A total of 53 articles were retrieved after applying explicit inclusion and exclusion criteria. To answer the RQ1, the frequency of identified factors in DevOps implementation projects, along with their percentages, is shown in Table 4.

Table 4. Identified factors in the SLR.

Identified Factors	Frequency (53)	%
Clashes between Dev and Ops mentality	45	85
Lack of microservice architecture understanding	34	64
Automation testing	46	87
Lack of communication strategies	46	87
Too much focus on tools	44	83
Lack of knowledge about DevOps tools	25	47
Lack of team ownership	23	43
Resistance to change	37	70
Lack of metrics monitoring	32	60
Continuous learning	43	81
Lack of expertise in human resources	38	72
Lack of visibility	37	70
Managing multiple environments	34	64
High implementation cost	29	55
Cross-functional team	43	81
Continuous delivery mode	45	85
High security	41	77
Product owner role	39	74
Early project release	45	85
Requirement trackability	38	72
Effective configuration management	33	62
Continuous management support	46	87
Continuous delivery	47	89

4.2. Categorization of the Identified Factors Based on SWOT Matrix

A SWOT (strengths, weaknesses, opportunities, and threats) analysis is a framework, as shown in Figure 4, which is commonly used as a tool for analyzing external and internal environments of the organization to help with decision-making [45]. Moreover, SWOT uses a diagnostic approach to identify key factors determining the success or failure of a plan or product [46]. A SWOT analysis may also help you identify aspects of your business that are holding you back or that your rivals may exploit if you do not defend yourself [47]. The SWOT analysis is frequently used in strategic planning, and it considers every individual aspect impacting the system environment in four SWOT categories [48]. The two categories of strengths and weaknesses for the factors identified evaluate the internal environment of the organization, whereas the opportunities and threats are recognized as evaluating the external environment of the system [48]. The internal and external environments include the variables related to the inside and outside of the system, respectively, which are considered as the most significant factors. Hence, the SWOT analysis employs a diagnostic method to identify significant elements influencing the success or failure of a strategy or system.

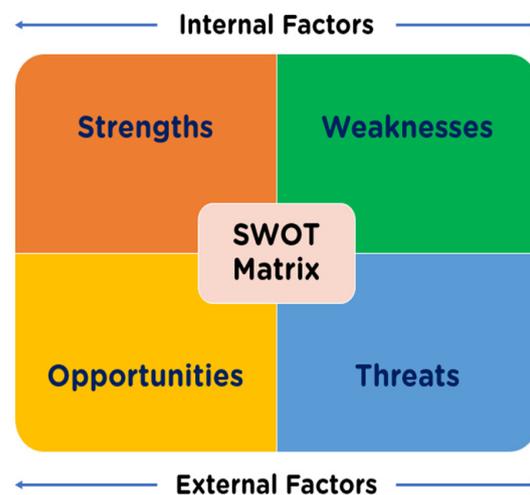


Figure 4. SWOT matrix.

A SWOT analysis comprises four categories: strengths (S), weaknesses (W), opportunities (O), and threats (T), as detailed below (Figure 4). Though the elements and discoveries under these categories may differ from firm to company, a SWOT analysis is incomplete without all of these elements.

(a) **Strengths (S):**

Strengths in the SWOT tool describe that how an organization excels and what the practices are that separates it from the competition in terms of developing products and implementing some strategies for obtaining benefits such as reliable products, a loyal customer base, unique technology, and so on.

(b) **Weaknesses (W):**

Weaknesses refer to factors that prevent an organization from achieving its performance to the best of its level. They are areas in which the company has to improve in order to remain competitive: a lack of budget, higher-than-average turnover, a lack of tools and procedures, or a lack of capital.

(c) **Opportunities (O):**

Opportunities are external variables that might provide a business with a competitive edge. For example, if a company provides a sufficient budget to implement a new technology, a new and better product can be delivered to the customer.

(d) **Threats (T):**

Threats refer to the factors that have the potential to cause harm to the business of an organization. For example, a lack of tools and techniques could be a threat to a software company trying to deliver a quality product to their customers, as the project may fail/be canceled.

SWOT has a wide spectrum of applications for conducting a strategic assessment of the organizations in a variety of domains studied by different researchers. Longhurst, G.J. et al. [49] conducted a case study using a SWOT analysis and developed a framework of prioritization of ecosystem management in the National Park Djerdap, Serbia. Richa et al. [45] used the SLR and SWOT approaches and developed a SWOT-based strategic framework for DevOps implementation in the software development process. Rajvikram et al. [48] used the SWOT framework to develop a strategic model for the evaluation of success factors and challenges for renewable energy development in significant countries including India, China, Iceland, Sweden, and the US.

Each identified factor has been labeled as S, W, O, and T, which stands for “Strengths”, “Weaknesses”, “Opportunities”, “Threats”, respectively. All the authors were involved in the mapping process of all the 23 identified factors among the SWOT categories based on their understanding, as shown in Figure 5.

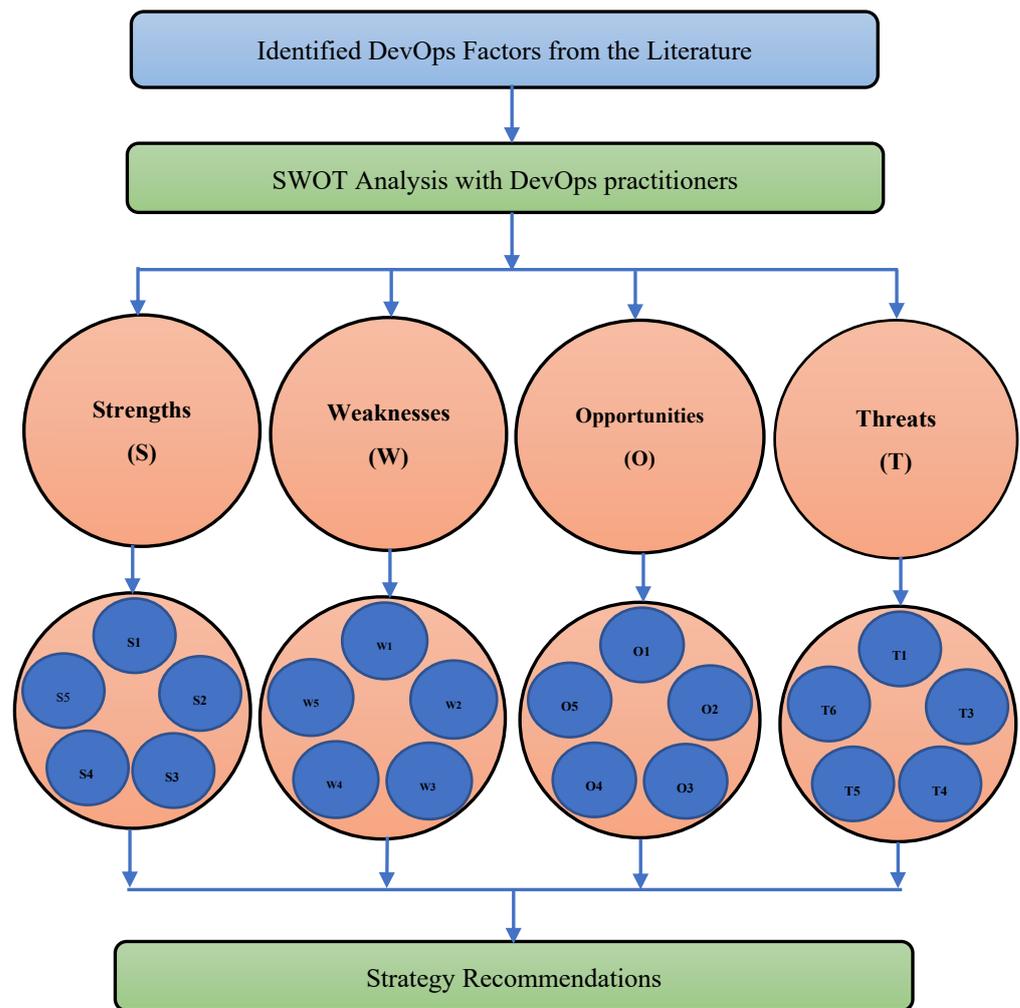


Figure 5. The SWOT structure.

For example: One of the factors, i.e., “Product owner role” was mapped to the “Opportunities” category. In the discussions with the experts, it was found that DevOps extends the role of customer/product owner, unlike the traditional approach in which the customer

has limited opportunities to interact with the development team. In DevOps and agile development, the customer works as an integral part of the development team, which gives an opportunity to effectively and efficiently manage the solution for their business values. Therefore, all the participants agreed to keep the factor “Product owner role” in the opportunities category. Similarly, the same process is followed in the mapping of all 23 identified factors. The SWOT framework used for implementing DevOps practices in software development is presented in Table 5.

Table 5. Categorization of the factors based on the SWOT categories.

Strengths (S)	Weaknesses (W)
Effective configuration management	Lack of team ownership
Early project release	Resistance to change
Automation testing	Lack of metrics monitoring
Continuous delivery mode	Lack of communication strategies
Cross-functional team	Lack of expertise in human resources
High security	Lack of microservice architecture understanding
Managing multiple environments	Clashes between Dev and Ops mentality
Opportunities (O)	Threats (T)
Requirement traceability	Too much focus on tools
Product owner role	High implementation cost
Continuous management support	Lack of knowledge about DevOps tools
Continuous learning	Lack of visibility
Continuous delivery	

4.3. SWOT-AHP Based Framework for DevOps Implementation

The SWOT technique has significant applications in various practical applications, but it is also criticized due to its limitations related to quantifying each item of all SWOT categories, as it becomes significantly difficult to identify which item is more influential for strategic decision-making [47,48]. In another way, the SWOT tool does not show any analytic approach to evaluate the relative significance between the items or the capacity to evaluate the suitability of alternate options on the basis of such elements [49]. Some of the key limitations of only using SWOT tools are discussed below:

- The SWOT analysis uses the environmental elements gathered by the qualitative examination.
- It does not evaluate relative importance between the items of SWOT categories.
- It does not focus on ambiguities raised between the items of a particular SWOT category.
- Increasing the number of factors in the particular category leads to an exponential increase in the number of strategies for decision-making.

Based on the limitations of SWOT as discussed above, it has not been considered as a powerful technique for decision-makers to develop a strategic framework which can help the organization successfully implement the system [50]. The SWOT has been blended with the analytic hierarchy process (AHP) and called SWOT-AHP, which can quantify the significance of individual factors and their respective categories in order to improve the decision-making and make adequate judgments [46]. Therefore, in this study an integrated method, i.e., SWOT-AHP, has been used for developing a strategic framework to manage the DevOps practices in software development industries. However, a number of studies have been published on DevOps, but they have not reported the SWOT structures of the factors. In this study, we have a total of 23 factors that impact DevOps practices implemented in software organizations, as given in Table 6. The AHP technique and its

application for prioritizing the identified factors and their SWOT categories are briefly discussed in the following sections.

4.3.1. Analytic Hierarchy Process (AHP)

The AHP approach is the most widely used multi-criteria decision-making (MCDM) technique. It was initially introduced by Thomas L. Saaty [51]. Since the introduction of the AHP method, various researchers have considered using it in numerous quantitative and qualitative research fields for tackling complicated decision-making challenges [51–58]. The AHP method is, basically, based on the following *three* different phases:

- Create a hierarchical structure of a complex problem, as shown in Figure 6.
- Use pairwise comparisons between the factors and their categories to determine the priority weight of each component and sub-factor.
- Examine the consistency of the decisions.

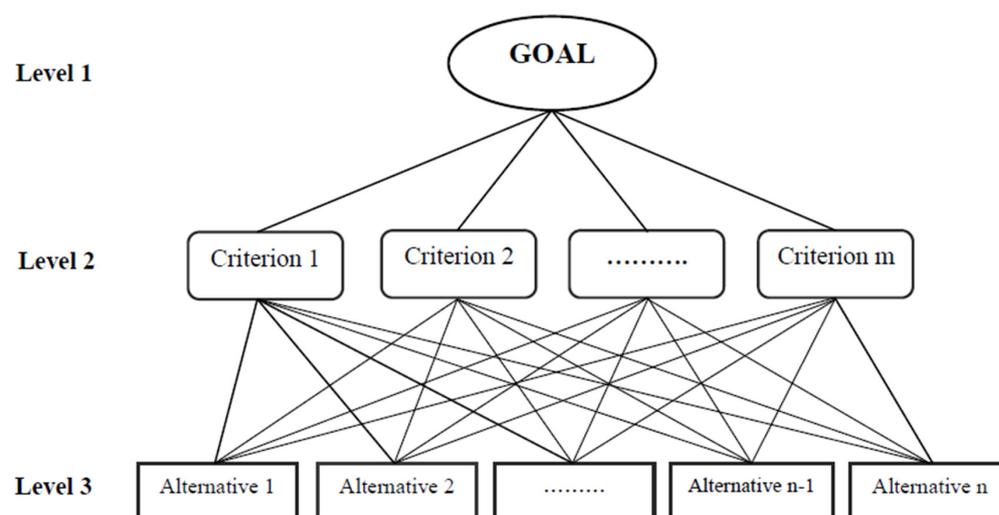


Figure 6. Hierarchical structure of the problem.

In the following steps, all three steps of the AHP method are briefly discussed.

Step 1: Develop the hierarchical tree of a complex decision-making problem. In this stage, we have developed a hierarchical tree of decision-making problems in their associated categories, and their factors [52–54,59,60]. The goal of the study is presented at the top level of the hierarchy. The different SWOT categories of the identified factors are connected at level 2, whereas the factors are connected at the lowest level of the hierarchy tree, as shown in Figure 6.

Stage 2: Determine the priority weight. After constructing the hierarchical structure of the problem, the comparative significance of the criteria within each level was finished at the lower level [18,55,61–64]. The preferences of decision-makers are quantified in the AHP using a standard 9-point AHP scale (as shown in Table 6).

Table 6. Scale for AHP method.

Linguistic Criteria for Importance	Intensity of Importance
“Equally Important (EI)”	1
“Nearly Important (MI)”	3
“Strongly Important (SI)”	5
“Very strongly Important (VSI)”	7
“Absolute Important (AI)”	9
“Intermediate values”	2,4,6,8

The AHP method is based on pairwise comparison matrices for solving the complex MCDM problem. Let us assume we have n different criteria $C_1, C_2, C_3, \dots, C_n$ having the weights $w_1, w_2, w_3, \dots, w_n$, respectively. The results of the pairwise comparison on the n criteria are presented in the $(n \times n)$ evaluation matrix A (as indicated in Equation (2)), where each element represents the weight quotient of the respective criterion, as displayed in the matrix B [4,17,58,59].

$$A = \begin{bmatrix} 1 & a_{12} & \dots & a_{1n} \\ a_{21} & 1 & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & 1 \end{bmatrix}, \text{ Where } a_{ij} = \frac{1}{a_{ji}}, a_{ji} > 0 \tag{1}$$

A normalized matrix (i.e., B) of A is computed to test the consistency matrix of pairwise comparison matrix A .

$$B = \begin{bmatrix} 1 & w_{12} & \dots & w_{1n} \\ w_{21} & 1 & \dots & w_{2n} \\ \dots & \dots & \dots & \dots \\ w_{n1} & w_{n2} & \dots & 1 \end{bmatrix} \tag{2}$$

where w_{ij} can be calculated using Equation (3).

$$w_{ij} = \frac{a_{ij}}{\sum_{i=1}^n a_{ij}} \tag{3}$$

To calculate the relative weight, the sum of values of each row is divided by n using Equation (4).

$$A = \frac{\sum_{i=1}^n w_{ij}}{n} \tag{4}$$

The pairwise comparison matrix, i.e., A , is said to be consistent if $A * W = \lambda_{max} * W$, which is a characteristic equation in the eigenvalue problem [4,17]. It is critical to assume that the greatest eigenvalue exceeds or equals n (i.e., $\lambda_{max} > n$), which indicates that if λ_{max} equals the sum of the column vector, i.e., AW , then matrix A will be strongly consistent [4,17,60,62].

C. Evaluation of consistency criteria of pairwise comparison matrix. The consistency of pairwise comparison matrices must be evaluated. The two matrices, i.e., the consistency ratio (CR) and consistency index (CI), are used to test the consistency. According to Saaty [50], the CI and CR may be determined using Equations (5) and (6), respectively.

$$CI = \frac{(\lambda_{max} - n)}{(n - 1)} \tag{5}$$

where λ_{max} denotes the largest eigenvalue of the pairwise comparison matrix, which can be calculated using the following equation discussed by different researchers:

$$\lambda_{max} = w_1y_1 + w_2y_2 + w_3y_3 + \dots + w_n y_n = \text{largest eigenvalue of matrix of order } n.$$

w: local criteria weight.

y: sum of the columns in a pairwise comparison matrix.

$$CR = \frac{CI}{RI} \tag{6}$$

where RI indicates the value of the randomly generated consistency index for various sizes (i.e., n) of the matrix, as shown in Table 7. The value of CR is accepted up to the value of 0.10. If the evaluated result is not more than 0.10, then the priority vector (weight) of the factor is acceptable, and we can conclude that the matrix is consistent [65,66]. If the CR

value exceeds 0.10, the technique must be evaluated, appraised and adjusted until the CR is acceptable.

Table 7. Relationship between index (size of matrix and random consistency RI).

n	1	2	3	4	5	6	7	8	9	10
RI	0	0	0.58	0.9	1.12	1.24	1.32	1.41	1.45	1.49

Various researchers have used the AHP approach to quantify and rank the elements in their investigations [18,55,56]. For example, Azeem et al. [64] followed the AHP method for prioritizing the requirement-related barriers in the GSD environment and developed an AHP-based framework for effectively handling these factors. Kabra et al. [52] utilized the AHP method to rank the coordination barriers in humanitarian supply chain management. Khan et al. [4] used the AHP process for prioritizing the DevOps challenges for implementing DevOps in software organizations. We briefly addressed the AHP technique in Section 4.3.1 to prioritize the categories (i.e., SWOT) and their related variables for successfully implementing DevOps.

4.3.2. Application of AHP for Prioritizing the Factors and Their SWOT Categories

The AHP is used in this section to prioritize the variables and their respective SWOT categories for implementing DevOps. Initially, the factors were investigated in the literature using the SLR process, and they were further categorized based on the SWOT framework, as briefly discussed and listed in Table 2. In the following, we have discussed all the steps of the AHP approach for prioritizing the identified factors:

Step 1. The goal of the study is to implement DevOps in the software industry by prioritizing the factors; Table 5 includes the identified factors and their SWOT categories.

Step 2. Create a hierarchical structure of the problem using the identified factors and their SWOT categories.

The AHP process begins with creating a hierarchical structure of the research problem, which is presented in Figure 7. The hierarchical model is developed on three levels: the goal of the study (level 0), SWOT categories (level 1: criteria), DevOps factors (level 2: sub-criteria).

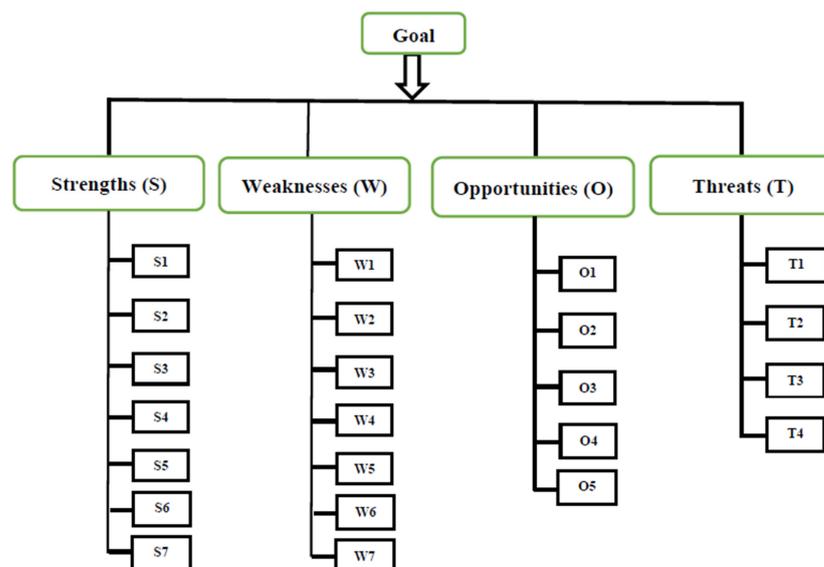


Figure 7. Hierarchical structure of the factors.

Step 3. Pairwise comparison of explored factors and their SWOT categories.

We have developed a second survey instrument (provided in Appendix B) to evaluate the relative significance between the identified factors and their SWOT categories [4,56,58,63,64]. The

AHP questionnaire was shared among the 30 DevOps experts working in DevOps development, and this list of experts includes DevOps developers, testers and researchers. These DevOps experts were approached through different social media groups that operate on Facebook, ResearchGate and LinkedIn. Table 8 shows the pairwise comparisons between the SWOT categories of the DevOps factors. The normalized matrix for calculating the weight for SWOT categories are shown in Table 9. The pairwise comparisons of DevOps factors, i.e., “Strengths”, “Weaknesses”, “Opportunities”, and “Threats”, are provided in Tables 10–13, respectively. The pairwise comparison matrix is synthesized to calculate the local weight (LW) for each category of factor. The calculation of LW is as follows:

- Compute the sum of each column in the pairwise comparison matrix.
- Divide each matrix element by its appropriate column sum.
- The priority weight is determined by taking a row-by-row average.

Table 8. Pairwise comparison between the SWOT categories of the factors.

Categories	S	W	O	T
S	1	2	2	2
W	0.5	1	0.5	0.5
O	0.5	2	1	2
T	0.5	2	0.5	1
Column Sum	2.5	7	4	5.5

Table 9. Normalized matrix for calculating weight.

Categories	S	W	O	T	Average (Weight)
S	0.4	0.29	0.5	0.36	0.388
W	0.2	0.14	0.125	0.09	0.139
O	0.2	0.29	0.25	0.36	0.275
T	0.2	0.29	0.125	0.18	0.199

Table 10. Pairwise comparison for “Strengths” category.

	S1	S2	S3	S4	S5	S6	S7	Local Weight
S1	1	1	0.142857	0.5	3	1	0.5	0.124
S2	1	1	0.5	0.5	1	1	0.5	0.104
S3	2	2	1	1	3	1	0.25	0.163
S4	2	2	1	1	1	0.5	0.2	0.125
S5	0.111111	1	0.333333	1	1	1	0.142857	0.076
S6	1	1	1	2	0.142857	1	0.111111	0.103
S7	2	2	4	4	2	2	1	0.305
Column Sum	9.11	10	7.98	10	11.14	7.5	2.7	$\Sigma = 1.000$

$\lambda_{max} = 7.16, CI = 0.027, CR = 0.020 < 0.10$

Table 11. Pairwise matrix for “Weaknesses” category.

	W1	W2	W3	W4	W5	W6	W7	Local Weight
W1	1	0.5	0.333333	0.111111	2	2	0.142857	0.093
W2	2	1	0.5	0.333333	0.5	3	0.142857	0.111
W3	3	2	1	2	2	2	0.111111	0.18
W4	0.142857	3	0.5	1	1	0.166667	0.142857	0.097
W5	1	2	0.111111	1	1	1	0.2	0.098
W6	3	0.142857	1	2	1	1	0.142857	0.131

$\lambda_{max} = 7.25, CI = 0.041, CR = 0.031 < 0.10$

Table 12. Pairwise comparison for “Opportunities” category.

	O1	O2	O3	O4	O5	Local Weight
O1	1	0.2	0.2	0.333333	3	0.124
O2	0.2	1	0.111111	0.142857	0.333333	0.046
O3	5	6	1	3	5	0.547
O4	3	5	0.333333	1	0.2	0.228
O5	0.333	0.2	0.142857	0.2	1	0.056
Column Sum	9.53	12.4	1.79	4.68	9.53	$\Sigma = 1.000$

$\lambda_{max} = 4.331, CI = 0.082, CR = 0.074 < 0.10$

Table 13. Pairwise comparison matrix for “Threats” category.

	T1	T2	T3	T4	Local Weight
T1	1	3	2	0.2	0.252
T2	0.2	1	0.5	0.5	0.1
T3	2	2	1	2	0.371
T4	0.5	2	3	1	0.28
Column Sum	3.7	8	6.5	3.7	$\Sigma = 1.000$

$\lambda_{max} = 4.247, CI = 0.082, CR = 0.091 < 0.10$

Step 4. Calculating the local priority weight of factors and ensuring the consistency of pairwise comparison matrices.

It was important to test the degree of consistency of each pairwise matrix. Therefore, we calculated the largest eigenvalue (λ_{max}) for each pairwise comparison matrix for the SWOT categories as follows:

$$\lambda_{max} = (2.5) * (0.39) + (7) * (0.14) + (4) * (0.28) + (5.5) * (0.20) = 4.175$$

Step 5. Local ranking of each factor (ranking of the factors in their respective categories). The local rank (LR) of each factor (Table 13) was calculated using the factor’s local weight in the specific category. The element with the highest local weight was considered the most imperative factor in the corresponding category.

Step 6. Calculating the global weight (GW) for each factor (overall ranking). The global AHP weight for each identified factor was computed to determine their overall rank and relative relevance. Global rank (GR) highlights the importance of each factor for DevOps implementation that is evaluated by the product for a specific component’s local weight and category weight. The rank of a given factor rises as the global weight rises; therefore, the top rank factor has the largest global weight value. The global weight (GW) and global rankings (GR) are provided in Table 14.

Table 14. Details of global and local weights of the factors and their SWOT categories.

	Category Weight	Factors	Local Weight (LW)	Local Rank (LR)	Global Weight (GW)	Global Rank (GR)
Strengths	0.388	S1	0.124	4	0.0481	9
		S2	0.104	5	0.04	11
		S3	0.163	2	0.063	4.5
		S4	0.125	3	0.049	8
		S5	0.076	7	0.029	13
		S6	0.103	6	0.04	11
		S7	0.305	1	0.118	2

Table 14. Cont.

	Category Weight	Factors	Local Weight (LW)	Local Rank (LR)	Global Weight (GW)	Global Rank (GR)
Weaknesses	0.139	W1	0.093	7	0.013	20.5
		W2	0.111	4	0.015	17.5
		W3	0.18	2	0.025	14
		W4	0.097	6	0.013	20.5
		W5	0.098	5	0.014	18.5
		W6	0.131	3	0.018	16
		W7	0.291	1	0.04	11
Opportunities	0.275	O1	0.124	3	0.034	12
		O2	0.046	5	0.013	20.5
		O3	0.547	1	0.15	1
		O4	0.228	2	0.063	4.5
		O5	0.056	4	0.015	17.5
Threats	0.199	T1	0.252	3	0.05	7
		T2	0.1	4	0.02	15
		T3	0.371	1	0.074	3
		T4	0.28	2	0.056	6

5. Results and Discussions

This study shed light on the various factors that positively/negatively impact the implementation practices of DevOps. Moreover, it provides a readiness model for implementing DevOps cultures in software development organizations, which provides the body of knowledge for both academicians and researchers of DevOps.

RQ1: Factors for DevOps Implementation

To address RQ1, a total of 23 factors were investigated which need to be managed to effectively implement DevOps practices in software development organizations. The reported factors recommended for the DevOps practitioners are the primary areas the DevOps teams need to address for effective and efficient implementation. Moreover, the success factors identified in the literature have been validated using the questionnaire survey technique with experts of DevOps. The analysis of survey responses indicated that the identified factors from the literature also exist in the real-world practice of DevOps.

RQ2: Categorizations of Identified Factors

In order to categorize the identified factors based on their types, all identified 23 factors were categorized into four categories based on the SWOT model. The details about the factors and their categories are discussed in Table 14. The significance of this categorization provides a broad understanding of the factors based on their nature. The factors of the categories of weaknesses and threats negatively impact DevOps practices, which could be managed based on their priorities assigned by the AHP process. On the other hand, the factors of strengths and opportunities positively impact DevOps practices, which indicates that these areas need to be more focused for effective implementation.

RQ3: Significance of Identified Factors and their Categories

The relative importance of individual factors and their categories is evaluated using the AHP technique, which is briefly discussed step by step. We pairwise compared each factor and the categories of the factors. Each success factor and its categories were prioritized based on the weight calculated from the pairwise comparison matrices. The outcomes of the AHP analysis show that technology is the most significant category of the factors, which is followed by culture, process and people. Table 14 shows the ranking of success factors and their categories along with global and local weights.

RQ4: Prioritization-based Taxonomical model for DevOps Implementation

Based on the findings of the AHP analysis, the taxonomy of the factors for implementing DevOps in software development was developed, as shown in Figure 8. In this study, the framework developed was based on the identified 23 factors that mapped into four categories, i.e., “Strengths”, “Weaknesses”, “Opportunities” and “Threats”. The developed taxonomy of the factors is shown in Table 13, which indicates that “Strengths” (CW) is the highest-ranked category of the identified factors, suggesting that the DevOps environment needs to manage various aspects in every release and be committed in an automated fashion, enabling the rapid building, testing and deployment of every project between the development and operations teams to successfully be implemented through DevOps in the software organizations. This result is consistent with the results of the study conducted by Rafi, S. et al. [12]. Table 13 presents opportunities as the second most significant category of factors that needs to be addressed by the DevOps practitioners. The mentioned factors in the “Opportunities” category focuses on different areas, i.e., continuous management support, requirement traceability, continuous delivery and the role of product owner. Among all the identified factors of the opportunities category, the taxonomy also presents the significance of each factor in the overall DevOps implementation project. The factor, “O3: Continuous management support”, is the most significant factor that needs to be focused on by the practitioners. The majority of DevOps participants strongly believe management should be strongly committed toward DevOps implementation and that they must provide sufficient funds and support the organization with the required tools so that the DevOps development activities can be implemented effectively and efficiently. Similarly, the AHP respondents primarily focused on the “S7: Managing multiple environments”, “T3: Lack of knowledge about DevOps tools”, “S3: Automation testing”, “T4: Lack of visibility”, “T1: Too much focus on tools”, “S1: Effective configuration management”, “S2: Early project release” and ‘O2: “Requirement traceability”’.

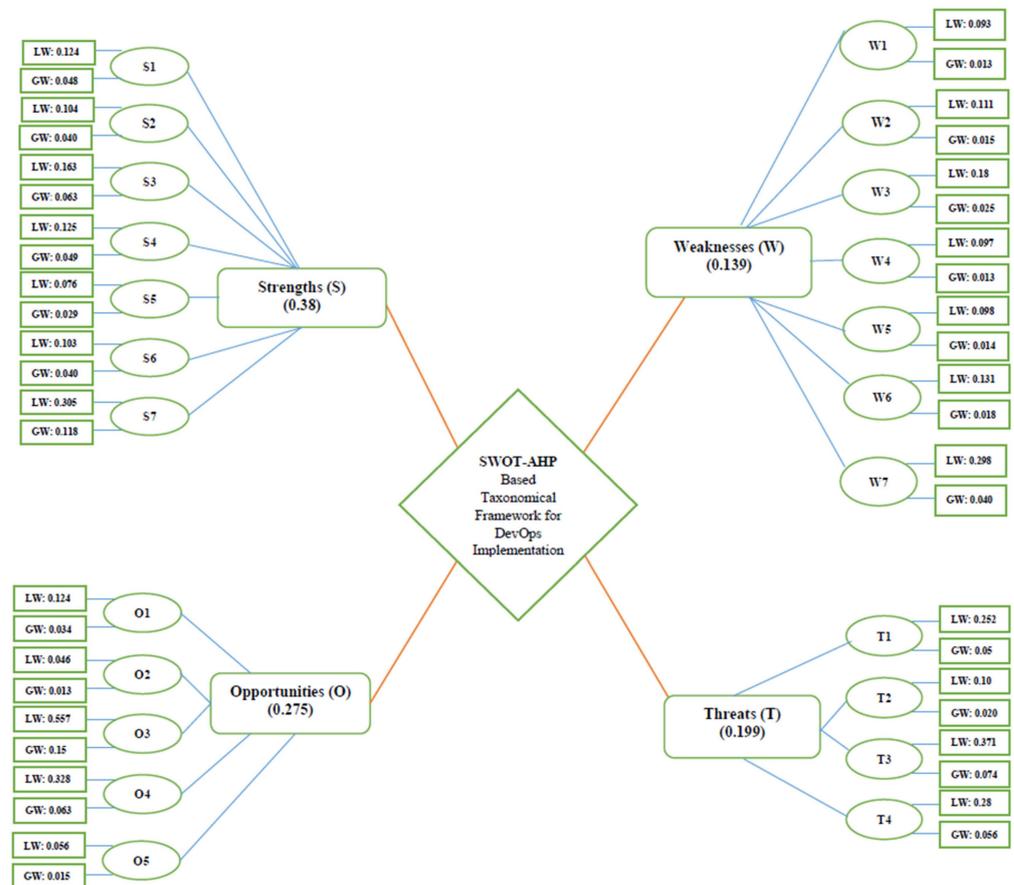


Figure 8. Taxonomy-based framework of the identified factors and their SWOT categories.

6. Limitations

In this thesis, we have followed a mixed research design using an SLR in order to design the priority-based framework for implementing DevOps practices. Using the SLR protocol, the factors (challenges, success factors) were identified. A sample of 53 primary studies were selected to extract data related to the factors. Due to the large number of research papers on DevOps development, it is possible that we could have missed some of the related research articles. However, it was not a systematic omission like other SLR studies [17,18,55,57]. With the AHP study, a survey questionnaire was used as an instrument to collect responses from the DevOps practitioners for conducting pairwise comparisons between the factors and their SWOT categories. It was developed based on the factors identified from the literature. However, this study was limited in verifying the perceptions and experiences of the survey participants. Due to limited resources and a low response rate in the survey, a sample size of 30 DevOps practitioners may not be justifiable and strong enough to validate the relative significance between the factors. However, based on the other existing empirical studies [18,37,57,58,67], our sample size is adequately sufficient to justify the AHP implementation.

7. Implications

This study provides a thorough overview regarding DevOps implementation in software organizations and developed a SWOT-AHP based framework. This study brings attention to various implications for both researchers and industry practitioners. The available literature reported 23 dimensions that significantly impact (i.e., positively/negatively) the DevOps activities and helped us develop a framework using the SWOT-AHP-based framework, which provides a knowledge base for both industry practitioners and researchers. The outcomes of the literature and empirical study enhance the knowledge of DevOps researchers, and the taxonomy of factors contributes to the industry by providing a robust framework that provides a roadmap to the software industry for implementing DevOps methods. Moreover, the factors' taxonomy enhances the understanding of DevOps practitioners when considering the most significant dimensions before implementing DevOps practices in software organizations.

8. Conclusions and Future Directions

Over the years, software development organizations have consistently and constantly adopted software development processes to develop commercially viable, quality products that meet the customer satisfaction level [4,67,68]. Presently, the organizations are following the DevOps framework for efficiently and effectively developing software products that can satisfy clients' requirements by integrating both development and operations silos under the single umbrella. Implementing DevOps in software development practices is not a straightforward approach due to various challenges that could hinder the DevOps activities [66,67]. The aim of the mentioned integration is to shorten the software development life cycle with continuous deployment. The increasing trend of implementing DevOps in software development motivated us to investigate the factors that positively/negatively impact DevOps implementation. Factors present the key areas that need to be considered when scaling the software development activities in the DevOps domain. A total of 23 factors were identified from the literature and classified across four categories of the SWOT framework (i.e., strengths, weaknesses, opportunities, threats) based on the software process improvement manifesto given in [54,68]. Furthermore, we conducted a survey (SWOT-AHP) to prioritize the identified factors and their respective categories based on their pairwise comparison weights.

The prioritization taxonomy of the factors was developed based on the local and global weights of each category and their respective factors. The taxonomy portrays (Figure 8) the significance of each category and the identified factors. Organizations willing to adopt DevOps practices could use the given taxonomy as a guide to consider the most significant factors and categories.

In the future, we plan to extend this study by identifying management strategies and practices for the success factors given in the developed taxonomy. These practices could be used by the software development organizations as guidelines to effectively implement the identified factors. Additionally, the findings of this study are planned to be considered as a building block of the maturity model that could evaluate the organizational DevOps capabilities and provide strategies for further improvements. The outcomes of this research study based on the prioritization of the factors for effectively implementing DevOps could be the first step for developing the maturity model.

Author Contributions: Conceptualization, data curation, investigation, methodology, validating, formal analysis, N.M.N. and M.S.; reviewing & editing, validating, A.T.Z.; reviewing, editing, formal analysis, M.A.; formal analysis, reviewing the article, P.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data is provided in the manuscript.

Acknowledgments: The authors would like to acknowledge the support of Prince Sultan University for paying the article processing charge (APC) of this publication.

Conflicts of Interest: No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work.

Appendix A

Second list of the SLR literature.

S. No.	Reference
PS1	Perera, P., Bandara, M. and Perera, I., 2016, September. Evaluating the impact of DevOps practice in Sri Lankan software development organizations. In <i>2016 Sixteenth International Conference on Advances in ICT for Emerging Regions (ICTer)</i> (pp. 281–287). IEEE.
PS2	Kamuto, M.B. and Langerman, J.J., 2017, May. Factors inhibiting the adoption of DevOps in large organisations: South African context. In <i>2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)</i> (pp. 48–51). IEEE.
PS3	McCarthy, M.A., Herger, L.M., Khan, S.M. and Belgodere, B.M., 2015, June. Composable DevOps: automated ontology-based DevOps maturity analysis. In <i>2015 IEEE international conference on services computing</i> (pp. 600–607). IEEE.
PS4	Waseem, M. and Liang, P., 2017, December. Microservices architecture in DevOps. In <i>2017 24th Asia-Pacific Software Engineering Conference Workshops (APSECW)</i> (pp. 13–14). IEEE.
PS5	Valani, A., 2018, September. Rethinking secure DevOps threat modeling: The need for a dual velocity approach. In <i>2018 IEEE Cybersecurity Development (SecDev)</i> (pp. 136–136). IEEE.
PS6	Dyck, A., Penners, R. and Lichter, H., 2015, May. Towards definitions for release engineering and DevOps. In <i>2015 IEEE/ACM 3rd International Workshop on Release Engineering</i> (p. 3). IEEE.
PS7	Virmani, M., 2015, May. Understanding DevOps & bridging the gap from continuous integration to continuous delivery. In <i>Fifth international conference on the innovative computing technology (intech 2015)</i> (pp. 78–82). IEEE.
PS8	Michener, J.R. and Clager, A.T., 2016, June. Mitigating an oxymoron: Compliance in a devops environments. In <i>2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC)</i> (Vol. 1, pp. 396–398). IEEE.
PS9	Bass, L., 2017. The software architect and DevOps. <i>IEEE Software</i> , 35(1), pp. 8–10.
PS10	Trihinas, D., Tryfonos, A., Dikaiakos, M.D. and Pallis, G., 2018. Devops as a service: Pushing the boundaries of microservice adoption. <i>IEEE Internet Computing</i> , 22(3), pp. 65–71.
PS11	Artac, M., Borovssak, T., Di Nitto, E., Guerriero, M. and Tamburri, D.A., 2017, May. DevOps: introducing infrastructure-as-code. In <i>2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C)</i> (pp. 497–498). IEEE.
PS12	Perera, P., Silva, R. and Perera, I., 2017, September. Improve software quality through practicing DevOps. In <i>2017 Seventeenth International Conference on Advances in ICT for Emerging Regions (ICTer)</i> (pp. 1–6). IEEE.

S. No.	Reference
PS13	Rong, G., Zhang, H. and Shao, D., 2016, May. CMMI guided process improvement for DevOps projects: an exploratory case study. In <i>Proceedings of the International Conference on Software and Systems Process</i> (pp. 76–85).
PS14	Domínguez-Acosta, M.F. and García-Mireles, G.A., 2021, October. Identifying Activities for Enhancing Software Quality in DevOps Settings. In <i>2021 10th International Conference On Software Process Improvement (CIMPS)</i> (pp. 84–89). IEEE.
PS15	Diel, E., Marczak, S. and Cruzes, D.S., 2016, August. Communication challenges and strategies in distributed DevOps. In <i>2016 IEEE 11th International Conference on Global Software Engineering (ICGSE)</i> (pp. 24–28). IEEE.
PS16	Rajkumar, M., Pole, A.K., Adige, V.S. and Mahanta, P., 2016, April. DevOps culture and its impact on cloud delivery and software development. In <i>2016 International Conference on Advances in computing, communication, & automation (ICACCA)(Spring)</i> (pp. 1–6). IEEE.
PS17	Marijan, D., Liaaen, M. and Sen, S., 2018, July. DevOps improvements for reduced cycle times with integrated test optimizations for continuous integration. In <i>2018 IEEE 42nd annual computer software and applications conference (COMPSAC)</i> (Vol. 1, pp. 22–27). IEEE.
PS18	Colomo-Palacios, R., Fernandes, E., Soto-Acosta, P. and Larrucea, X., 2018. A case analysis of enabling continuous software deployment through knowledge management. <i>International Journal of Information Management</i> , 40, pp. 186–189.
PS19	Laukkarinen, T., Kuusinen, K. and Mikkonen, T., 2018. Regulated software meets DevOps. <i>Information and Software Technology</i> , 97, pp. 176–178.
PS20	Plant, O.H., van Hillegersberg, J. and Aldea, A., 2022. Rethinking IT governance: Designing a framework for mitigating risk and fostering internal control in a DevOps environment. <i>International Journal of Accounting Information Systems</i> , p. 100560.
PS21	Toivakka, H., Granlund, T., Poranen, T. and Zhang, Z., 2021, November. Towards RegOps: A DevOps Pipeline for Medical Device Software. In <i>International Conference on Product-Focused Software Process Improvement</i> (pp. 290–306). Springer, Cham.
PS22	Bobbert, Y. and Chtepen, M., 2021. Problems of CI/CD and DevOps on Security Compliance. In <i>Strategic Approaches to Digital Platform Security Assurance</i> (pp. 256–285). IGI Global.
PS23	Rafi, S., Akbar, M.A., Yu, W., Alsanad, A., Gumaedi, A. and Sarwar, M.U., 2022. Exploration of DevOps testing process capabilities: An ISM and fuzzy TOPSIS analysis. <i>Applied Soft Computing</i> , 116, p. 108377.
PS24	Pérez-Sánchez, J., Ros, J.N. and Gea, J.M.C.D., 2021, March. DevOps Certification in IT Industry: Preliminary Findings. In <i>World Conference on Information Systems and Technologies</i> (pp. 473–479). Springer, Cham.
PS25	Rafi, S., Akbar, M.A., AlSanad, A.A., AlSuwaidan, L., Abdulaziz AL-ALShaikh, H. and AlSagari, H.S., 2022. Decision-Making Taxonomy of DevOps Success Factors Using Preference Ranking Organization Method of Enrichment Evaluation. <i>Mathematical Problems in Engineering</i> , 2022.
PS26	Faustino, J., Adriano, D., Amaro, R., Pereira, R. and da Silva, M.M., 2022. DevOps benefits: A systematic literature review. <i>Software: Practice and Experience</i> .
PS27	Lima, J.A.P. and Vergilio, S.R., 2020. Test Case Prioritization in Continuous Integration environments: A systematic mapping study. <i>Information and Software Technology</i> , 121, p. 106268.
PS28	Gupta, V., Kapur, P.K. and Kumar, D., 2017. Modeling and measuring attributes influencing DevOps implementation in an enterprise using structural equation modeling. <i>Information and software technology</i> , 92, pp. 75–91.
PS29	Battina, D.S., 2021. AI and DevOps in Information Technology and Its Future in the United States. <i>INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)</i> , ISSN, pp. 2320–2882.
PS30	Hermawan, A. and Manik, L.P., 2021. The Effect of DevOps Implementation on Teamwork Quality in Software Development. <i>Journal of Information Systems Engineering and Business Intelligence</i> , 7(1), pp. 84–90.
PS31	Fitzgerald, B. and Stol, K.J., 2017. Continuous software engineering: A roadmap and agenda. <i>Journal of Systems and Software</i> , 123, pp. 176–189.
PS32	Elazhary, O., Werner, C., Li, Z.S., Lowlind, D., Ernst, N.A. and Storey, M.A., 2021. Uncovering the benefits and challenges of continuous integration practices. <i>IEEE Transactions on Software Engineering</i> .

S. No.	Reference
PS33	Chen, L., 2017. Continuous delivery: overcoming adoption challenges. <i>Journal of Systems and Software</i> , 128, pp. 72–86.
PS34	Lwakatare, L.E., Kuvaja, P. and Oivo, M., 2015, May. Dimensions of devops. In <i>International conference on agile software development</i> (pp. 212–217). Springer, Cham.
PS35	Nagarajan, A.D. and Overbeek, S.J., 2018, October. A DevOps implementation framework for large agile-based financial organizations. In <i>OTM Confederated International Conferences "On the Move to Meaningful Internet Systems"</i> (pp. 172–188). Springer, Cham.
PS36	Bheri, S. and Vummenthala, S., 2019. An Introduction to the DevOps Tool Related Challenges.
PS37	Elberzhager, F., Arif, T., Naab, M., Süß, I. and Koban, S., 2017, January. From agile development to devops: going towards faster releases at high quality—experiences from an industrial context. In <i>International conference on software quality</i> (pp. 33–44). Springer, Cham.
PS38	Lwakatare, L.E., Kilamo, T., Karvonen, T., Sauvola, T., Heikkilä, V., Itkonen, J., Kuvaja, P., Mikkonen, T., Oivo, M. and Lassenius, C., 2019. DevOps in practice: A multiple case study of five companies. <i>Information and Software Technology</i> , 114, pp. 217–230.
PS39	Forsgren, N., Tremblay, M.C., VanderMeer, D. and Humble, J., 2017, May. DORA platform: DevOps assessment and benchmarking. In <i>International Conference on Design Science Research in Information System and Technology</i> (pp. 436–440). Springer, Cham.
PS40	Dinner, A., 2020. <i>Factors that Influence the Synergy between Development and IT Operations in a DevOps Environment</i> (Master's thesis, Faculty of Commerce).
PS41	Bruel, J.M., Mazzara, M. and Meyer, B., 2019. Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment. <i>France, Cham: Springer International Publishing</i> .
PS42	Wettinger, J., Andrikopoulos, V. and Leymann, F., 2015, October. Enabling DevOps collaboration and continuous delivery using diverse application environments. In <i>OTM Confederated International Conferences "On the Move to Meaningful Internet Systems"</i> (pp. 348–358). Springer, Cham.
PS43	Düllmann, T.F., Paule, C. and van Hoorn, A., 2018, May. Exploiting devops practices for dependable and secure continuous delivery pipelines. In <i>2018 IEEE/ACM 4th International Workshop on Rapid Continuous Software Engineering (RCOSE)</i> (pp. 27–30). IEEE.
PS44	Capizzi, A., Distefano, S. and Mazzara, M., 2019, May. From devops to devdataops: Data management in devops processes. In <i>International Workshop on Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment</i> (pp. 52–62). Springer, Cham.
PS45	Joby, P.P., 2019. Exploring devops: challenges and benefits. <i>Journal of Information Technology</i> , 1(01), pp. 27–37.
PS46	Saito, H., Lee, H.C.C. and Wu, C.Y., 2019. <i>DevOps with Kubernetes: accelerating software delivery with container orchestrators</i> . Packt Publishing Ltd.
PS47	Poniszewska-Marañda, A. and Czechowska, E., 2021. Kubernetes cluster for automating software production environment. <i>Sensors</i> , 21(5), p. 1910.
PS48	Díaz, J., López-Fernández, D., Pérez, J. and González-Prieto, Á., 2021. Why are many businesses instilling a DevOps culture into their organization?. <i>Empirical Software Engineering</i> , 26(2), pp. 1–50.
PS49	López-Fernández, D., Diaz, J., Garcia-Martin, J., Pérez, J. and Gonzalez-Prieto, A., 2021. DevOps Team Structures: Characterization and Implications. <i>IEEE Transactions on Software Engineering</i> .
PS50	Amaro, R.M.D., Pereira, R. and da Silva, M.M., 2022. Capabilities and Practices in DevOps: A Multivocal Literature Review. <i>IEEE Transactions on Software Engineering</i> .
PS51	Shameem, M., 2022. A Systematic Literature Review of Challenges Factors for Implementing DevOps Practices in Software Development Organizations: A Development and Operation Teams Perspective. <i>Evolving Software Processes: Trends and Future Directions</i> , pp. 187–199.
PS52	Mishra, A. and Otaiwi, Z., 2020. DevOps and software quality: A systematic mapping. <i>Computer Science Review</i> , 38, p. 100308.
PS53	Benjamin, J. and Mathew, J., 2021, February. Enhancing the efficiency of continuous integration environment in DevOps. In <i>IOP Conference Series: Materials Science and Engineering</i> (Vol. 1085, No. 1, p. 012025). IOP Publishing.

Quality score of each selected article.

Reference	QA-1	QA-2	QA-3	QA-4	QA-5	Total
PS1	0.5	1	1	0.5	1	4
PS2	0.5	1	0.5	0.5	1	3.5
PS3	0.5	1	1	0.5	1	4
PS4	1	1	1	0.5	1	4.5
PS5	1	1	0.5	0.5	0.5	3.5
PS6	1	1	1	0.5	1	4.5
PS7	1	1	0.5	0.5	1	4
PS8	1	1	1	0.5	1	4.5
PS9	1	1	1	0.5	1	4.5
PS10	1	1	1	0.5	1	4.5
PS11	1	1	1	1	1	5
PS12	1	1	1	1	1	5
PS13	1	0.5	1	0.5	1	4
PS14	1	1	1	0.5	0.5	4
PS15	1	1	1	0.5	0.5	4
PS16	1	1	1	0.5	0	3.5
PS17	1	1	1	0.5	0.5	4
PS18	1	1	1	0.5	1	4.5
PS19	1	1	1	1	1	5
PS20	1	1	1	1	1	5
PS21	1	0.5	1	0.5	1	4
PS22	1	0.5	1	0.5	0.5	3.5
PS23	1	1	1	1	1	5
PS24	0.5	1	1	0.5	0.5	3.5
PS25	1	1	1	0.5	1	4.5
PS26	1	1	1	1	1	5
PS27	1	1	1	1	1	5
PS28	1	1	1	1	1	5
PS29	0.5	1	0.5	1	0.5	3.5
PS30	1	1	1	0.5	0.5	4
PS31	1	1	1	0.5	0.5	4
PS32	1	1	1	1	1	5
PS33	1	1	1	1	1	5
PS34	0.5	1	0.5	1	0	3
PS35	0.5	1	0.5	1	0	3
PS36	1	1	1	0.5	0	3.5
PS37	0.5	1	0.5	1	0	3
PS38	1	1	1	1	1	5
PS39	1	1	1	1	1	5
PS40	1	1	1	1	1	5
PS41	0.5	1	1	1	0	3.5
PS42	0.5	1	1	1	0	3.5
PS43	0.5	1	1	0.5	0.5	3.5
PS44	0.5	1	1	0.5	0.5	3.5

References

1. Stahl, D.; Martensson, T.; Bosch, J. Continuous practices and devops: Beyond the buzz, what does it all mean? In Proceedings of the 2017 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA), Vienna, Austria, 30 August–1 September 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 440–448.
2. Akbar, M.A.; Mahmood, S.; Shafiq, M.; AlSanad, A.; AlSanad, A.A.-A.; Gumaiei, A. Identification and prioritization of DevOps success factors using fuzzy-AHP approach. *Soft Comput.* **2020**, *1*, 1–25. [\[CrossRef\]](#)
3. Lwakatare, L.E.; Kuvaja, P.; Oivo, M. An exploratory study of devops extending the dimensions of devops with practices. In Proceedings of the ICSEA 2016: The Eleventh International Conference on Software Engineering Advances, Rome, Italy, 21–25 August 2016; Volume 104, p. 2016.
4. Khan, A.A.; Shameem, M. Multicriteria decision-making taxonomy for DevOps challenging factors using analytical hierarchy process. *J. Softw. Evol. Process* **2020**, *32*, e2263. [\[CrossRef\]](#)
5. Forsgren, N.; Smith, D.; Humble, J.; Frazelle, J. *2019 Accelerate State of Devops Report*; Google: Mountain View, CA, USA, 2019.
6. Ravichandran, A.; Taylor, K.; Waterhouse, P. *Devops for Digital Leaders: Reignite Business with a Modern Devops-Enabled Software Factory*; Springer Nature: Berlin, Germany, 2016; p. 173.
7. Riungu-Kalliosaari, L.; Mäkinen, S.; Lwakatare, L.E.; Tiihonen, J.; Männistö, T. November. DevOps adoption benefits and challenges in practice: A case study. In *International Conference on Product-Focused Software Process Improvement*; Springer: Cham, Switzerland, 2016; pp. 590–597.
8. Forsgren, N.; Tremblay, M.C.; VanderMeer, D.; Humble, J. DORA platform: DevOps assessment and benchmarking. In *International Conference on Design Science Research in Information System and Technology*; Springer: Cham, Switzerland, 2017; pp. 436–440.
9. Leite, L.; Rocha, C.; Kon, F.; Milojicic, D.; Meirelles, P. A Survey of DevOps Concepts and Challenges. *ACM Comput. Surv.* **2019**, *52*, 1–35. [\[CrossRef\]](#)
10. Kerzazi, N.; Adams, B. Who needs release and devops engineers, and why? In Proceedings of the International Workshop on Continuous Software Evolution and Delivery, Austin, TX, USA, 14–15 May 2016; pp. 77–83.
11. Rafi, S.; Yu, W.; Akbar, M.A.; Mahmood, S.; Alsanad, A.; Gumaiei, A. Readiness model for DevOps implementation in software organizations. *J. Softw. Evol. Process* **2021**, *33*, e2323. [\[CrossRef\]](#)
12. Kim, G.; Humble, J.; Debois, P.; Willis, J.; Forsgren, N. *The DevOps Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations*; IT Revolution: Melbourne, Australia, 2021.
13. Gillies, A. *Software Quality: Theory and Management*. 2011. Available online: <https://lulu.com> (accessed on 20 July 2022).
14. Tumorin, R.; Mazzara, M.; Kassab, M.; Succi, G.; Lee, J. Quality attributes in practice: Contemporary data. In *Agent and Multi-Agent Systems: Technology and Applications*; Springer: Cham, Switzerland, 2016; pp. 281–290.
15. Bazzana, G.; Andersen, O.; Jokela, T. ISO 9126 and ISO 9000: Friends or Foes? In Proceedings of the 1993 Software Engineering Standards Symposium, Brighton, UK, 30 August–3 September 1993; IEEE: Piscataway, NJ, USA, 1993; pp. 79–88.
16. Chung, L.; Nixon, B.A.; Yu, E. Using quality requirements to systematically develop quality software. In Proceedings of the Fourth International Conference on Software Quality, Basel, Switzerland, 17–20 October 1994.
17. Perera, P.; Bandara, M.; Perera, I. Evaluating the impact of DevOps practice in Sri Lankan software development organizations. In Proceedings of the 2016 Sixteenth International Conference on Advances in ICT for Emerging Regions (ICTer), Negombo, Sri Lanka, 1–3 September 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 281–287.
18. Shameem, M.; Kumar, C.; Chandra, B.; Khan, A.A. Systematic review of success factors for scaling agile methods in global software development environment: A client-vendor perspective. In Proceedings of the 2017 24th Asia-Pacific Software Engineering Conference Workshops (APSECW), Nanjing, China, 4–8 December 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 17–24.
19. Akbar, M.A.; Khan, A.A.; Huang, Z. Multicriteria decision making taxonomy of code recommendation system challenges: A fuzzy-AHP analysis. *Inf. Technol. Manag.* **2022**, *1*–17. [\[CrossRef\]](#)
20. Kamuto, M.B.; Langerman, J.J. Factors inhibiting the adoption of DevOps in large organisations: South African context. In Proceedings of the 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, India, 19–20 May 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 48–51.
21. McCarthy, M.A.; Herger, L.M.; Khan, S.M.; Belgodere, B.M. Composable DevOps: Automated ontology-based DevOps maturity analysis. In Proceedings of the 2015 IEEE International Conference on Services Computing, New York, NY, USA, 27 June–2 July 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 600–607.
22. Waseem, M.; Liang, P. Microservices architecture in DevOps. In Proceedings of the 2017 24th Asia-Pacific Software Engineering Conference Workshops (APSECW), Nanjing, China, 4–8 December 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 13–14.
23. Trihinas, D.; Tryfonos, A.; Dikaiakos, M.D.; Pallis, G. DevOps as a Service: Pushing the Boundaries of Microservice Adoption. *IEEE Internet Comput.* **2018**, *22*, 65–71. [\[CrossRef\]](#)
24. Rong, G.; Zhang, H.; Shao, D. CMMI guided process improvement for DevOps projects: An exploratory case study. In Proceedings of the International Conference on Software and Systems Process, Pittsburgh, PA, USA, 19–20 May 2022; pp. 76–85.
25. Colomo-Palacios, R.; Fernandes, E.; Soto-Acosta, P.; Larrucea, X. A case analysis of enabling continuous software deployment through knowledge management. *Int. J. Inf. Manag.* **2018**, *40*, 186–189. [\[CrossRef\]](#)
26. Toivakka, H.; Granlund, T.; Poranen, T.; Zhang, Z. Towards RegOps: A DevOps Pipeline for Medical Device Software. In *International Conference on Product-Focused Software Process Improvement*; Springer: Cham, Switzerland, 2021; pp. 290–306.

27. Heine, K.M. Predicting DevOps Effectiveness in Information Technology (IT) Projects. Ph.D. Thesis, The George Washington University, Washington, DC, USA, 2022.
28. Smeds, J.; Nybom, K.; Porres, I. DevOps: A definition and perceived adoption impediments. In *International Conference on Agile Software Development*; Springer: Cham, Switzerland, 2015; pp. 166–177.
29. Luz, W.P.; Pinto, G.; Bonifácio, R. Adopting DevOps in the real world: A theory, a model, and a case study. *J. Syst. Softw.* **2019**, *157*, 110384. [CrossRef]
30. Humble, J.; Kim, G. *Accelerate: The Science of Lean Software and Devops: Building and Scaling High Performing Technology Organizations*; IT Revolution: Melbourne, Australia, 2018.
31. Bite, D.; Janmere, L. Social Research Methods. Available online: https://lais.llu.lv/pls/pub/!pub_switcher.main?au=G&page=course_description_pub/GSOC5046/2/1 (accessed on 20 July 2022).
32. Bryman, A. *Social Research Methods*; Oxford University Press: Oxford, UK, 2016.
33. Bobbert, Y.; Chtepen, M. Problems of CI/CD and DevOps on Security Compliance. In *Strategic Approaches to Digital Platform Security Assurance*; IGI Global: Hershey, PA, USA, 2021; pp. 256–285.
34. Khan, A.A.; Keung, J.; Niazi, M.; Hussain, S.; Ahmad, A. Systematic literature review and empirical investigation of barriers to process improvement in global software development: Client–vendor perspective. *Inf. Softw. Technol.* **2017**, *87*, 180–205. [CrossRef]
35. Watson, R. Quantitative research. *Nurs. Stand.* **2015**, *29*, 44. [CrossRef]
36. Brannen, J.; Coram, T. (Eds.) *Mixing Methods: Qualitative and Quantitative Research*; Aldershot: Avebury, UK, 1992; Volume 5.
37. Gregar, J. *Research Design (Qualitative, Quantitative and Mixed Methods Approaches)*; SAGE: Thousand Oaks, CA, USA, 1994; p. 228.
38. Walker, R.J.; Briand, L.C.; Notkin, D.; Seaman, C.B.; Tichy, W.F. Panel: Empirical validation—what, why, when, and how. In *Proceedings of the 25th International Conference on Software Engineering*, Portland, OR, USA, 3–10 May 2003; IEEE Computer Society: Washington, DC, USA, 2003; p. 721.
39. Niazi, M.; Mahmood, S.; Alshayeb, M.; Riaz, M.R.; Faisal, K.; Cerpa, N.; Khan, S.U.; Richardson, I. Challenges of project management in global software development: A client-vendor analysis. *Inf. Softw. Technol.* **2016**, *80*, 1–19. [CrossRef]
40. Kitchenham, B. *Procedures for Performing Systematic Reviews*; Keele University: Keele, UK, 2004; Volume 33, pp. 1–26.
41. Keele, S. *Guidelines for Performing Systematic Literature Reviews in Software Engineering*; Technical Report, Ver. 2.3 EBSE Technical Report; EBSE: Goyang-si, Korea, 2007.
42. Soomro, A.B.; Salleh, N.; Mendes, E.; Grundy, J.; Burch, G.; Nordin, A. The effect of software engineers’ personality traits on team climate and performance: A Systematic Literature Review. *Inf. Softw. Technol.* **2016**, *73*, 52–65. [CrossRef]
43. Vilela, J.; Castro, J.; Martins, L.E.G.; Gorschek, T. Integration between requirements engineering and safety analysis: A systematic literature review. *J. Syst. Softw.* **2017**, *125*, 68–92. [CrossRef]
44. Khan, S.U.; Azeem, M.I. Intercultural challenges in offshore software development outsourcing relationships: An exploratory study using a systematic literature review. *IET Softw.* **2014**, *8*, 161–173. [CrossRef]
45. Sinha, R.; Shameem, M.; Kumar, C. SWOT: Strength, weaknesses, opportunities, and threats for scaling agile methods in global software development. In *Proceedings of the 13th Innovations in Software Engineering Conference on Formerly Known as India Software Engineering Conference*, Jabalpur, India, 27 February 2020; pp. 1–10.
46. Helms, M.M.; Nixon, J. Exploring SWOT analysis—where are we now? A review of academic research from the last decade. *J. Strategy Manag.* **2010**, *3*, 215–251. [CrossRef]
47. Lee, J.; Kim, I.; Kim, H.; Kang, J. SWOT-AHP analysis of the Korean satellite and space industry: Strategy recommendations for development. *Technol. Forecast. Soc. Change* **2020**, *164*, 120515. [CrossRef]
48. Elavarasan, R.M.; Afridhis, S.; Vijayaraghavan, R.R.; Subramaniam, U.; Nurunnabi, M. SWOT analysis: A framework for comprehensive evaluation of drivers and barriers for renewable energy development in significant countries. *Energy Rep.* **2020**, *6*, 1838–1864. [CrossRef]
49. Longhurst, G.J.; Stone, D.M.; Duloher, K.; Scully, D.; Campbell, T.; Smith, C.F. Strength, weakness, opportunity, threat (SWOT) analysis of the adaptations to anatomical education in the United Kingdom and Republic of Ireland in response to the COVID-19 pandemic. *Anat. Sci. Educ.* **2020**, *13*, 301–311. [CrossRef]
50. Veličkowska, I. Implementation of a SWOT-AHP methodology for strategic development of a district heating plant in fuzzy environment. *Strateg. Manag.* **2022**, *27*, 43–56. [CrossRef]
51. Saaty, T.L. What is the analytic hierarchy process? In *Mathematical Models for Decision Support*; Springer: Berlin/Heidelberg, Germany, 1988; pp. 109–121.
52. Kabra, G.; Ramesh, A.; Arshinder, K. Identification and prioritization of coordination barriers in humanitarian supply chain management. *Int. J. Disaster Risk Reduct.* **2015**, *13*, 128–138. [CrossRef]
53. Albayrak, E.; Erensal, Y.C. Using analytic hierarchy process (AHP) to improve human performance: An application of multiple criteria decision making problem. *J. Intell. Manuf.* **2004**, *15*, 491–503. [CrossRef]
54. Bozbura, F.; Beskese, A.; Kahraman, C. Prioritization of human capital measurement indicators using fuzzy AHP. *Expert Syst. Appl.* **2007**, *32*, 1100–1112. [CrossRef]
55. Barbosa, P.I.; Szklo, A.; Gurgel, A. Sugarcane ethanol companies in Brazil: Growth challenges and strategy perspectives using Delphi and SWOT-AHP methods. *Biomass Bioenergy* **2022**, *158*, 106368. [CrossRef]

56. Akbar, M.A.; Nasrullah; Shameem, M.; Ahmad, J.; Maqbool, A.; Abbas, K. Investigation of Project Administration related challenging factors of Requirements Change Management in global software development: A systematic literature review. In Proceedings of the 2018 International Conference on Computing, Electronic and Electrical Engineering (ICE Cube), Quetta, Pakistan, 12–13 November 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–7. [\[CrossRef\]](#)
57. Solangi, Y.A.; Longsheng, C.; Shah, S.A.A.; AlSanad, A.; Ahmad, M.; Akbar, M.A.; Gumaei, A.; Ali, S. Analyzing Renewable Energy Sources of a Developing Country for Sustainable Development: An Integrated Fuzzy Based-Decision Methodology. *Processes* **2020**, *8*, 825. [\[CrossRef\]](#)
58. Akbar, M.A.; Naveed, W.; Mahmood, S.; Alsanad, A.A.; Alsanad, A.; Gumaei, A.; Mateen, A. Prioritization Based Taxonomy of DevOps Challenges Using Fuzzy AHP Analysis. *IEEE Access* **2020**, *8*, 202487–202507. [\[CrossRef\]](#)
59. Kamal, T.; Zhang, Q.; Akbar, M.A. Toward successful agile requirements change management process in global software development: A client–vendor analysis. *IET Softw.* **2020**, *14*, 265–274. [\[CrossRef\]](#)
60. Khan, A.A.; Shameem, M.; Nadeem, M.; Akbar, M.A. Agile trends in Chinese global software development industry: Fuzzy AHP based conceptual mapping. *Appl. Soft Comput.* **2021**, *102*, 107090. [\[CrossRef\]](#)
61. Shameem, M.; Khan, A.A.; Hasan, M.G.; Akbar, M.A. Analytic hierarchy process based prioritisation and taxonomy of success factors for scaling agile methods in global software development. *IET Softw.* **2020**, *14*, 389–401.
62. Shameem, M.; Kumar, C.; Chandra, B. A proposed framework for effective software team performance: A mapping study between the team members' personality and team climate. In Proceedings of the 2017 International Conference on Computing, Communication and Automation (ICCCA), Greater Noida, India, 5–6 May 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 912–917.
63. Kieu, P.T.; Nguyen, V.T.; Nguyen, V.T.; Ho, T.P. A spherical fuzzy analytic hierarchy process (SF-AHP) and combined compromise solution (CoCoSo) algorithm in distribution center location selection: A case study in agricultural supply chain. *Axioms* **2021**, *10*, 53. [\[CrossRef\]](#)
64. Rafi, S.; Akbar, M.A.; Manzoor, A. DevOps Business Model: Work from Home Environment. In Proceedings of the International Conference on Evaluation and Assessment in Software Engineering 2022, Gothenburg, Sweden, 13–15 June 2022; pp. 408–412.
65. Rafi, S.; Akbar, M.A.; Yu, W.; Alsanad, A.; Gumaei, A.; Sarwar, M.U. Exploration of DevOps testing process capabilities: An ISM and fuzzy TOPSIS analysis. *Appl. Soft Comput.* **2022**, *116*, 108377.
66. Rafi, S.; Akbar, M.A.; AlSanad, A.A.; AlSuwaidan, L.; AL-ALShaikh, H.A.; AlSagri, H.S. Decision-making taxonomy of devops success factors using preference ranking organization method of enrichment evaluation. *Math. Probl. Eng.* **2022**, *2022*, 2600160.
67. Zarour, M.; Alhammad, N.; Alenzi, M.; Alsarayrah, K. Devops Process Model Adoption in Saudi Arabia: An Empirical Study. *Jordanian J. Comput. Inf. Technol.* **2020**, *6*, 234–246. [\[CrossRef\]](#)
68. Zarour, M.; Alhammad, N.; Alenezi, M.; Alsarayrah, K. A research on DevOps maturity models. *Int. J. Recent Technol. Eng.* **2019**, *8*, 4854–4862. [\[CrossRef\]](#)