Readiness model for DevOps implementation in software organizations

Article *in* Journal of Software Evolution and Process · October 2020

DOI: 10.1002/smr.2323

Received: 14 May 2020 | Revised: 24 August 2020 | Accepted: 19 September 2020
DOI: 10.1002/smr.2323

RESEARCH ARTICLE - EMPIRICAL

Journal of Software: Evolution and Process WILEY

Readiness model for DevOps implementation in software organizations

Saima Rafi[1] | Wu Yu[2] | Muhammad Azeem Akbar[3] | Sajjad Mahmood[4] | Ahmed Alsanad[5] | Abdu Gumaei[5]

[1]School of Computer Science and Technology, Chongqing University of Posts and Telecommunication, Chongqing, China
[2]School of Cyber security and Information Law, Chongqing University of Posts and Telecommunication, Chongqing, China
[3]College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China
[4]Information and Computer Science Department, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia
[5]STC's Artificial Intelligence Chair, Department of Information Systems, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia

Correspondence
Saima Rafi, School of Computer Science and Technology, Chongqing University of Posts and Telecommunication, Chongqing 400065, China.
Email: saeem112@yahoo.com

Wu Yu, School of Cyber security and Information Law, Chongqing University of Posts and Telecommunication, Chongqing, China.
Email: wuyu@cqupt.edu.cm

Ahmed Alsanad, STC's Artificial Intelligence Chair, Department of Information Systems, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia.
Email: aasanad@ksu.edu.sa

Funding information
National Social Science Foundation of China, Grant/Award Number: 17XFX013

Abstract

DevOps is a new software engineering paradigm adopted by various software organizations to develop the quality software within time and budget. The implementation of DevOps practices is critical, and there are no guidelines to assess and improve the DevOps activities in software organizations. Hence, there is a need to develop a readiness model for DevOps (RMDevOps) with an aim to assist the practitioners for implementation of DevOps practices in software firms. To achieve the study objective, we conducted a systematic literature review (SLR) study to identify the critical challenges and associated best practices of DevOps. A total of 18 challenges and 73 best practices were identified from the 69 primary studies. The identified challenges and best practices were further evaluated by conducting a survey with industry practitioners. The RMDevOps was developed based on other well-established models in software engineering domain, for example, software process improvement readiness model (SPIRM) and software outsourcing vendor readiness model (SOVRM). Finally, case studies were conducted with three different organizations with an aim to validate the developed model. The results show that the RMDevOps is effective to assess and improve the DevOps practices in software organizations.

KEYWORDS
best practices, case study, guidelines, readiness model

1 | INTRODUCTION

The technical experts of enterprises are evolving towards continuous delivery (CD) and rapid change to meet customers' needs and deliver the software on time. Software enterprises need to adapt methodologies that can support timely feedback from all stakeholders during the software development life cycle. To meet challenges of the current era, software companies are adopting the DevOps activities to bridge the gap between developers, operators, and deliver continuous services to the customer.[1,2]

DevOps is a set of practices that combine development and operations to support continues delivery of quality software.[2] There are a number of DevOps definitions in the literature. For example, Dyck et al[3] define DevOps as "a software process that emphasize the collaboration within and between the teams involved in software development process." Similarly, Smeds et al[4] defined DevOps as a combination of inter-supported elements, that is, cultural enablers, capabilities, and technology enablers. Capabilities are defined as a main aspect of DevOps, including all capabilities such as "continuous development, continuous planning, continuous integration and testing, continuous deployment and release, continuous feedback and recovery, and continuous optimization and monitoring of infrastructure and service failure handling" without any delay, whereas cultural and technology enablers are responsible for sharing goals, values, and responsibilities in building and testing automation. These process improvement techniques are not same for all companies and departments end upon the nature of the issues they are facing in an organization. To conclude this problem, effective practices are required as a guideline for future processes. DevOps aims to bridge the gap between development and operational teams for continuous deployment and continuous integration.[5]

Capability Maturity Model Integration (CMMI) is an industry de facto process improvement program that provides a set of development best practices to help software organizations understand their current capacities and offer a guideline to achieve their business goals. The CMMI model has five levels (i.e., Levels 1–5; the higher the level more maturity it contains) depending upon standards of process used for software development and maintenance. Furthermore, CMMI has three variations, that is, CMMI for service (CMMI-SVC), CMMI for development (CMMI-DEV), and CMMI for acquisition (CMMI-ACQ), respectively.[5,6] However, CMMI is not designed for DevOps projects.. To date, only few frameworks have been designed for improving different aspects of DevOps, namely, unicorn framework for service improvement strategy,[7] IBM DevOps for performance analysis,[8] DORA platform for DevOps that is used for the assessment of software product delivery value stream,[9] and ontology-based DevOps maturity model to perform DevOps tasks.[10]

However, none of the existing frameworks help an enterprise in addressing critical challenges (CCHs) during implementing of DevOps projects. There is a need of readiness model that could help software practitioners to manage Devops adoption properly. Therefore, the objective of this study is to develop a readiness model that provides guidelines to the practitioners to manage challenges associated with DevOps projects. The identified best practices are mapped against each factor to assist the practitioners with a body of knowledge to resolve certain challenges in DevOps adoption. In this paper, we have identified CCHs, success factors and practices associated with DevOps projects using systematic literature review (SLR). Next, we propose a readiness model DevOps (RMDevOps) by adapting CMMI structure.[11] The initial idea of RMDevOps has already been published EASE2020[12] in which we have identified the success factors of DevOps. In this paper, we extend the work by identifying the challenges and best practices of DevOps. To achieve the objective of the study, we develop the following steps to address the research problems:

Step 1: The aim of this step is to explore the CCHs and the best practices of DevOps implementation reported in the literature by adopting SLR approach.
RQ1: What CCHs of DevOps are indicated in the literature?
RQ2: What are the best practices of DevOps reported in the literature?
Step 2: This step aims to classify the identified best practices against each identified CCH and design the readiness levels of RMDevOps.
RQ3: Are the identified challenges and practices related to industrial practitioners?
RQ4: How can we design a comprehensive general RMDevOps?
Step 3: The objective of this step is to evaluate the proposed RMDevOps in software organizations.
RQ5: How to evaluate the RMDevOps efficiency in implementing the DevOps activities in the software organization?

The rest of the paper is organized as follows: Section 2 presents study background and motivation. Section 3 highlights the selected research methodology. The findings of the study are given in Section 4. Sections 5 and 6 present summary and limitations of the study. Section 7 describes study implications. The conclusion and future directions of the study are presented in Section 8.

2 | LITERATURE REVIEW AND MOTIVATION

2.1 | Overview of DevOps

The DevOps is a new paradigm adopted by software industry to promote close collaboration between development and operational teams. DevOps aims to reduce inconsistencies between development, operations, and release teams.[13] The traditional software development approaches such as waterfall and iterative models do not explicitly focus on these tasks. The work done by Lwakatare et al[14] also shows agreement on a proposed concept; that is, DevOps is a coordination between development and operational teams. On the other hand, some researchers are against restricting DevOps to coordination task because various sources and tasks are involved to fulfill requirements for software development and operations.[15] Therefore, there is a need to make additional inquiries from practitioners about understanding of DevOps concepts to omit challenges and hurdles causing unsatisfactory environment in software organizations while adopting DevOps.

Several scholars have covered different aspects of DevOps. For example, Leite et al[16] conducted a survey study and highlighted the challenging factors of DevOps adoption in software industry. They further indicated that the DevOps provides an environment to work in a close collaboration by both development and operation teams. However, they did not provide any model based on challenging factors of DevOps. Similarly, Rafi et al[17] indicate that the operational and development teams have different tasks to perform during software development process. The operational team demand the new produced feature; they implement it and give the certain feedback to the development team, whereas the development team try to release new versions into production, while operations staff try to resist these changes to maintain software stability. Bolscher and Daneva[18] indicated that the DevOps practices are compulsory to derive organization towards higher performance. This high performance in CD and DevOps is gained by following practices of DevOps in software enterprises, while developing a system. Ambler[19] emphasized that the new and effective practices were required to improve the agility in an organization with the aim to implement the continuous development and release process. Ambler illustrates several factors to adopt new practices of agile collaborated with DevOps to lead software organization towards successful production unit. Continuous development, deployment, and integration of new release are only applicable by practicing DevOps principles.[20–23] Perera et al[1] conducted an interview study with industry experts and summarized that the adoption of DevOps practices could significantly enhance the quality of software projects.

2.2 | Challenges for DevOps

Although there are various challenges while implementing DevOps in an organization, the main challenge inhibiting the adoption of DevOps is "lack of education around DevOps" as pointed out by Kamuto and Langerman[24]; the organizational management resists to change the development approach because of lack of knowledge about DevOps practices. Similarly, there is a conceptual gap between operation and development team; this gap should be merged by close collaboration and continuous development and deployment process.[25] The enterprises mostly fail to have "continuous development environment"[26] because practices like continuous integration and continuous testing that can mitigate certain problems are lagging behind. The developers discourage to coordinate with operational team because of different understanding levels in both teams, which may cause frustration and other problems in an organization.[27,28] One of the key concerns in DevOps adoption is the availability of effective tools.[29]

2.3 | Existing models of DevOps

The literature shows several models developed to guide the industry practitioner for the successful implementation of DevOps activities. For example, unicorn framework[7] developed to overcome the challenges of DevOps by providing microservices via cloud for rapid and continuous release. However, this model is limited to some extent, that is, to provide the continuous release in entire system. Similarly, the DORA platform[9] assessment focuses on the software product delivery value stream but does not provide guideline to the practitioners to resolve risks regarding DevOps adoption. Moreover, the focus of DORA is more on software product delivery rather than improving whole software development process while working in DevOps environment.

Mohamed[30] developed a DevOps maturity model based on CMMI aspects to bridge a gap between traditional organization processes. He assessed his model leveling based on four dimensions, quality, automation, governance, and communication. However, there are some limitations in his model such as there is no mapping between factors of DevOps. Second, he focused on distributed software engineering processes to assess the impact of his model instead of organizations working with DevOps. The maturity model lacks proper guidelines (mapped: challenges, success factors, and practices) at each level to reach certain maturity level. All the marked challenges were covered in this research to make explicit inquiry on these aspects. Furthermore, based on Motorola assessment tool and case study, we evaluate our readiness model RMDevOps. This model will generally support all types of software organizations to improve their potential towards adopting DevOps.

Teixeira[31] also developed a maturity model for DevOps by identifying capabilities and practices from SLR and verified the results by conducting interviews. Teixeira's study is related with our study but there are some limitations in his work. He did not categorize factors as (success factors and challenges) in his work. Second, he only used interviews to develop the model. Furthermore, there are no assessment criteria to check the validity of his proposed model. In this study, we have identified practices and factors having positive and negative impact on DevOps adoption and verified them from industrial practitioners using questionnaire approach. We also map factors and practices to their respected levels. Furthermore, the case study technique is used to validate the effectiveness of model in real industry.

Zarour et al[32] and Badshah et al[8] studied existing maturity models of DevOps, that is, IBM DevOps, Compagemini's DevOps model, Hewlett Packard Enterprise's DevOps maturity model, and ITIL in DevOps. None of the above-mentioned model provides all three aspects (i.e., challenges, success factors and practices) to assess DevOps practices. In our work, we have validated the RMDevOps model with industry using case study approach. Furthermore, we have also received feedback from practitioner about RMDevOps model based on three criteria, that is, easy to use, user satisfaction, and structure of RMDevOps. The feedback from the practitioners indicates that RMDevOps is helpful in improving DevOps practices.

2.4 | Existing readiness models of other software engineering domains

Readiness models and efficient algorithms[33,34] have been developed in various fields of software engineering to provide proper guidelines for improving the development process. Niazi et al[35] developed an implementation maturity model (IMM), concerning to improve the implementation of software development activities in industry. The CCHs were used to design levels of IMM. They conducted case study approach with three different organizations to check the effectiveness of the proposed IMM and summarized that the develop model is an effective tool to improve the implementation of software development activities.

Furthermore, Ali and Khan[33] proposed a readiness model for outsourcing relationship of software enterprise (software outsource partnership management [SOPM]). They identified critical barriers of outsource partnership and mapped the identified barriers into five readiness levels (success, conversion, maturity, readiness, and contract). They used case study technique to test the adaptability and efficiency of their model.

Khan[36] developed a software outsourcing vendor readiness model (SOVRM). To develop the readiness levels of SOVRM, researcher used the critical success factors and critical barrier of outsourcing vendor, identified using SLR and questionnaire survey approaches.

Similarly, Akbar[37] developed a software requirement change management maturity model (SRCMIMM) in context of global software development organizations. The SRCMIMM is based on the CCHs and critical success factors of requirement change management (RCM). Akbar claims that the SRCMIMM assists to assess a certain maturity level of an organization with respect to their requirements change management process.

2.5 | Motivation

DevOps is a new software development paradigm, to move forward towards continuous development and delivery process. Despite the significance of DevOps in the software industry, less attention has been given to develop the guidelines for DevOps implementation. Based on the state-of-the-art literature and to the best of our knowledge, there is no comprehensive general readiness model to assist the software organization with proper guidelines while adopting DevOps practices. This model will help the practitioners to focus on the key areas that need improvements based on the significance of factors in DevOps environment. The key goal of this study is to propose a model, that is, RMDevOps for better implementation of DevOps activities and assist software organizations to resolve DevOps challenges by adopting adequate practices. The proposed RMDevOps will be based on the existing models of other software engineering domain[35,38,39] and on critical challenging factors and best practices of DevOps. To achieve study objective, we have conducted an SLR and questionnaire survey with the aim to explore and identify the CCH and potential best practices of DevOps from industrial perspective. On the basis of the findings, and by following the road map of existing readiness models,[11,36] we have designed the readiness levels of proposed RMDevOps.

3 | RESEARCH DESIGN

For this research, we have used the SLR, questionnaire survey and a case study approach to achieve the study objectives Figure 1. The details are as follows:

Phase 1: The aim of this step is to identify the CCHs and the best practices of DevOps implementation, reported in the literature by adopting SLR approach.

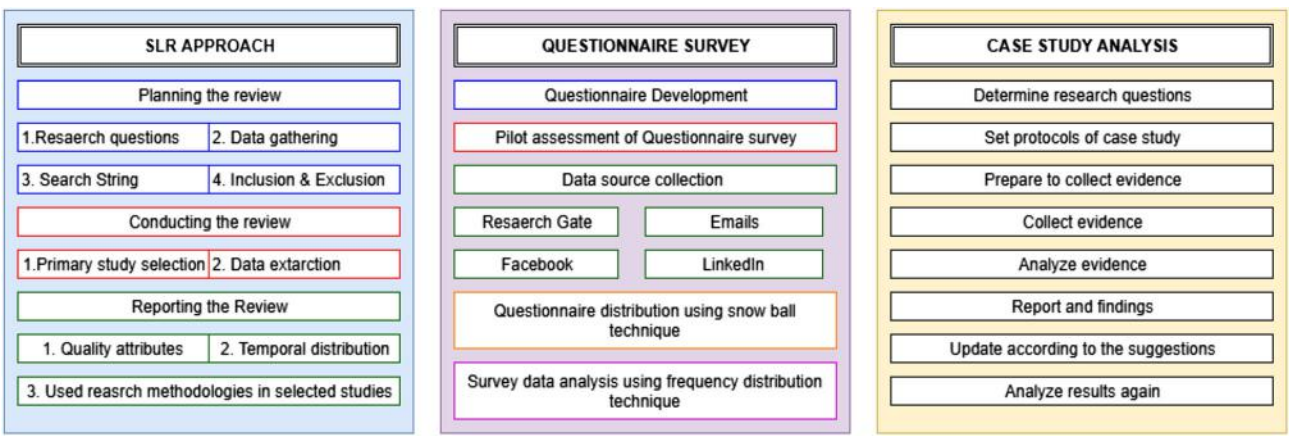


FIGURE 1 Research methodology

RQ1: What CCHs of DevOps are indicated in the literature?
RQ2: What are the best practices of DevOps reported in the literature?
Phase 2: This step aims to classify the identified best practices against each identified CCH from industrial practitioners and to design the readiness levels of RMDevOps.
RQ3: Are the identified challenges and practices related to industrial practitioners?
RQ4: How can we design a comprehensive general RMDevOps?
Phase 3: The objective of this step is to evaluate the RMDevOps in software organizations.
RQ5: How to evaluate the RMDevOps efficiency in implementing the DevOps activities in the software organization?

3.1 | Systematic literature review

We used SLR as a research method to identify the challenges and best practice of DevOps from the available literature and classify the explored literature using inclusion and exclusion criteria.[36] The outcomes from SLR are less biased and more reliable then informal literature review.[37] SLR approach has been adopted for identification in various fields of software engineering.[40–47] Kitchenham and Charters[48] has discussed the three phases to conduct SLR, that is, planning of review, conducting of review, and reporting of review. In this study, we have adopted the same strategy adopted by Kitchenham and Charters[48] to conduct SLR. The phases of SLR are discussed in detail in the following section:

3.1.1 | Phase 1: Planning the review

Research questions

To achieve research objective, we have designed research questions in three steps, in order to address the study objectives (Section 1). Step 1 (RQ1 and RQ2) has been addressed using SLR approach

Data gathering sources

To collect the appropriate data related to study objective, we have selected the six digital libraries (Table 1) based on our personal research experience and by following the suggestion.[45]

Search strings

To develop a research strings, the key term with respect to our search questions and their alternatives were used.[49,50] The search string was constructed considering the population, intervention, outcome of relevance, and experimental design. All the selected keywords and their alternatives were concatenated using the Boolean "OR" and "AND" to formulate the search strings:

- "Population: DevOps adoption in software organizations."
- "Intervention: DevOps challenges and best practices."
- "Outcome of Relevance: List of challenges and best practices."
- "Experimental Design: Empirical studies."

Population: ("DevOps" OR "Dev and Ops" OR "development and operational team" OR "continuous deployment" OR "continuous software engineering using DevOps" OR "continuous integration and continuous delivery".

TABLE 1 List of selected electronic database

URL of electronic databases	**"http://ieeexplore.ieee.org"** **"http://dl.acm.org"** **"www.link.springer.com"** **"www.wiley.com"** **"www.sciencedirect.com"** **"www.scholar.google.com"**
Language used	Only English
Publication forums	Journals, conferences, workshops full text to avoid biasness related to our search

Intervention: “challenges” OR “barriers” OR “hurdles” OR “difficulties” OR “hindrance” OR “inhabited” OR “limitations” OR “impediments”) AND “practices” OR “Tools” OR “methods” OR “techniques” OR “processes” OR “programs” OR “approaches.”

Experimental Design: “case study”, “questionnaire survey”, “interviews”, “empirical investigations”.

The complete search string is designed as “population” AND “intervention” AND “experimental design.”

Inclusion and exclusion criteria

The inclusion (IC) and exclusion (EC) criteria were developed for the initial refinement of collected literature received in the response of executing the search string on the selected electronic databases. The IC and EC were developed, by considering the studies conducted in other software engineering domains.[51–53]

The IC includes the following:

IC-1: Papers selected should be in journal, book chapter, workshops, or conference.
IC-2: Papers that discuss challenges associated with DevOps.
IC-3: Papers that explain well about factors hindered in DevOps adoption in an organization for continues development, deployment, testing, delivery etc.
IC-4: Paper that shows association between DevOps adoption metrics, challenges in implementation of DevOps in the software industry, strategies how to minimize these challenges, and customer satisfaction measures.
IC-5: Paper should be in English language.

The EC includes the following:

EC-1: Papers irrelevant to study objective must be rejected.
EC-2: Papers having no clear explanation about DevOps methodology will not be considered for analysis.
EC-3: Papers that do not explain challenges and practice guidelines to adopt DevOps.
EC-4: In case of duplicated studies, the most complete version will be considered.

Study QE

The quality evaluation (QE), that is, tollgate approach,[54] is performed with data extraction phase to assess the quality of selected research material. The checklist was designed based on instruction given by the previous studies[51–54]; the formulated checklist is shown in Table 2. This process consists of five questions (QE:1–QE:5), the QE score (QES) was allocated to each article as presented in Appendix A. Similar evaluation criteria of QE for selected studies have been performed by Shameem et al and Khan and Keung[52,53] in their studies. By following the same guidelines, we calculated QES of our study. The questions for QE are helpful in measuring the level of article's effectiveness in the particular selected search area. Furthermore, the main objective of performing QE is to confirm that certain findings will make a valuable contribution for an SLR.

3.1.2 | Phase 2: Conducting the review

Criteria of primary study selection

The targeted studies were refined, more by using the tollgate approach as used by Afzal et al.[54] This method consists of five phases as shown in Section 3. In initial stage, 2,113 studies were collected from the selected online data warehouses by using the search strings mentioned in above (Section 3) and also by following inclusion and exclusion criteria in Section 3. The tollgate approach QE criteria (Table 2) shortlisted 69 primary

TABLE 2 Quality evaluation checklist

QES	Quality evaluation criteria
QES-1	Score “1” if an article contains answers of the checklist
QES-2	Score “0.5” if article contains partial answers to the checklist
QES-3	Score “0” if article does not address the checklist
QE questions	**Questions about quality evaluation**
QE:1	“Does the adopted research method address the research questions?”
QE:2	“Does the study discuss any challenge about DevOps?”
QE:3	“Does the study discuss DevOps framework and its adoption guidelines?”
QE:4	“Is the collected data related to DevOps?”
QE:5	“Are the identified findings according to the research questions justification?”

studies, which is 3.27% of total extracted research studies, as shown in Table 3 with all five phases. The shortlisted final selected studies are listed in Appendix A. For more details, see Table 3, and Figure 2 shows the percentage of occurrence of research material in various data warehouses after performing all five steps discussed above.

Data extraction and synthesis

In this step, the data extraction method was defined, which will investigate that the information from the selected studies is related to our research questions. The selected primary studies were reviewed carefully to extract the ideas, strategies/techniques, concepts, contributions, organization types, and findings. The data after extraction were summarized to form a list of challenges and practices involved in DevOps adoption. In this process, the first and third authors were continually involved. The concern of second author was to carefully review all the data from primary selected articles in order to overcome research bias. Through adopting this method, the challenges validity also enhanced, as multiple authors were involved in data verification. After completing the data extraction process, an inter-rater reliability test was performed to eliminate bias. Thus, for this test, we involved three external reviewers from software engineering field. The reviewers selected 15 studies from the initial stage of the tollgate approach and followed steps of SLR process. We calculated nonparametric Kendall's coefficient of concordance (W)[54] to calculate inter-rater reliability test between findings of the reviewers and authors of the study. The value of W is defined as follows: “the value of $W = 0$ represented a complete disagreement and $W = 1$ represented complete agreement.” The result of inter-rater reliability test for 14 selected studies indicated that $W = 0.82$ ($p = 0.004$) calculated by using code in link (https://rdrr.io/cran/DescTools/man/KendallW.html), which shows agreement between the authors and the external reviewers.

TABLE 3 Tollgate approach results

Electronic databases	P1	P2	P3	P4	P5	Percentage ($N = 69$)
ACM Digital Library	250	190	70	21	7	10
IEEE Xplore	243	194	120	74	10	15
Wiley Inter Science	178	83	43	26	7	10
SpringerLink	203	107	39	20	6	8
ScienceDirect	250	201	123	31	10	15
Google Scholar	989	643	242	74	29	42
Total	2,113	1,418	637	246	69	100

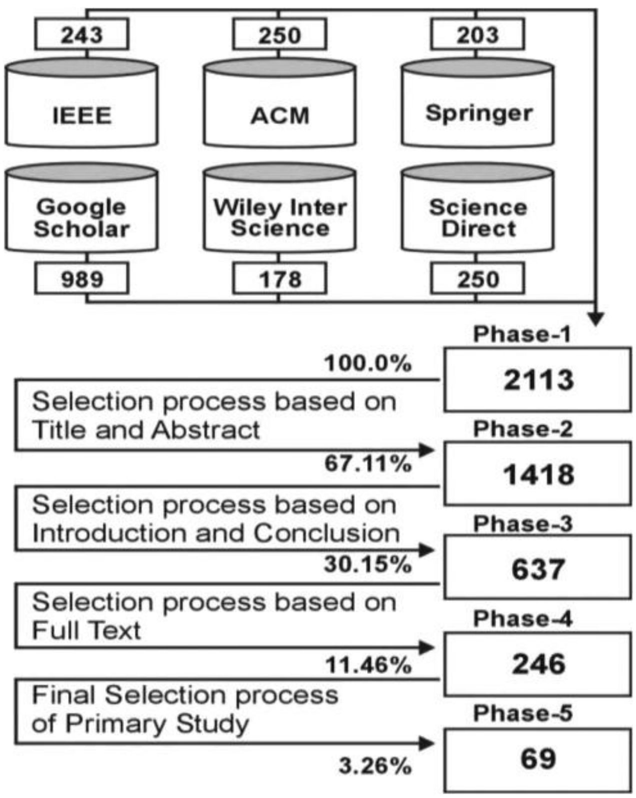


FIGURE 2 Phases of “tollgate approach”

3.1.3 | Phase 3: Reporting the review

Quality attributes

The QE was evaluated based on five questions (Section 3). The selected papers were listed, along with QES evaluation in Appendix A. The overall score of QE questions specifies that 68% of selected studies scored ≥70% or above, which proved the effectiveness of selected primary studies to answer the research questions. Furthermore, we considered 40% QES as a threshold for primary studies, as shown in Appendix A.

Temporal distribution of selected primary articles

A summary of selected primary articles also showcases that the publication year is from 2013 to 2020. Out of 69 primary studies, only 1% and 3% of work were done during the years 2013 and 2014 because of a new area of research. More work has been done in this filed during the year 2019 (39%), which indicates the increasing trend of publication related to DevOps in software organizations. The year 2020 is still preceding stage. Publication years with research methods are shown in Figure 3.

Used research methodology in selected articles

The selected articles consist of 11 (16%) questionnaire survey (QS), 13 (19%) case studies (CS), 11 (16%) of grounded theories (GTs), 8 (11.5%) content analysis (CA), 14 (20%) action research (AR), and 12 (18%) mixed methods (MMs) as elaborated in Figure 4. The results illustrate that the most common methods used were (action research 20%) and (case study 19%). As case study is the most significant technique approach for evaluation of models,[35,37] we applied the same approach to measure the effectiveness of our proposed model.

3.2 | Identified CCHs and best practices

After final selection of challenges and best practices from SLR, the basic ideas, finding, concepts, themes, recommendations etc. were recorded. A total of 18 challenges and 73 best practices were identified from 69 primary studies.

By following the methodology of Akbar et al[55] and coding method of grounded theory approach,[56] we mapped and identified challenges and practices with levels of RMDevOps. The four main steps are as follows: (a) code, (b) subcategories, (c) categories, and (d) framework/theory were performed carefully. All investigated best practices were mapped against the identified challenges. The identified challenging factors and their related best practice are classified into five readiness levels of RMDevOps, based on the understanding of challenges, practices, and the experience of mapping team.

The first three authors of this study were involved in mapping process. The fourth author of this study involved to verify the mapping process. We further conducted inter-rater reliability test to remove researcher's biasness. To perform this task, we invited three experts (researchers) from different empirical software engineering research group (City University of Hong Kong and Nanjing University, China) and visiting their profile on Research Gate. They perform all mapping steps and categories the investigated challenges and practices based on their understanding. By

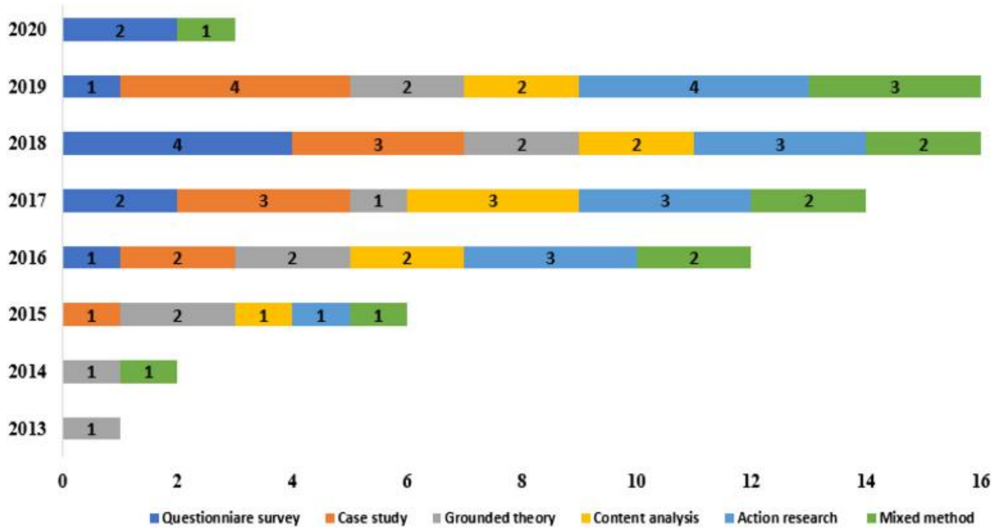
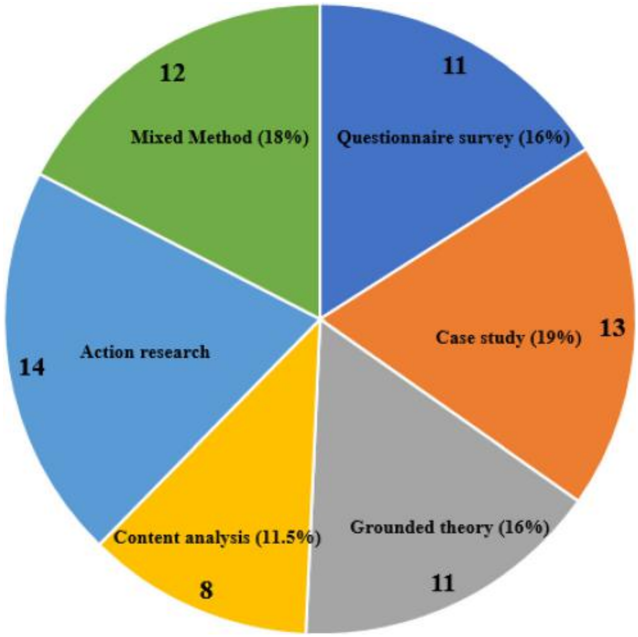


FIGURE 3 Selected articles publication year with research method

FIGURE 4 Used research methods in other studies



considering the mapping outcomes of both authors and external experts, we calculated the nonparametric Kendall's coefficient of concordance (W).[54] The results (W = 0.92, p = 0.006) show the significant positive agreement between the mapping of both teams. This indicates that the mapping process is unprejudiced and categorization of factors is consistent.

3.3 | Questionnaire survey

The questionnaire approach is the most effective way to gather responses from the targeted population. We have adopted questionnaire survey approach to verify the factors and their mapping practices from the industrial participants, having knowledge about DevOps.

3.3.1 | Questionnaire development

We have developed a questionnaire survey by using platform of Google Form (docs.google.com/forms). The survey questionnaire consist of three sections: (i) bibliography, (ii) detailed closed-ended questions that contain the list of practices and challenges of DevOps, and (iii) open-ended questions in which we ask respondents about the additional challenges.

3.3.2 | Pilot assessment of questionnaire survey

After questionnaire development, we have conducted a pilot assessment of questionnaire from industry experts. The three external experts were involved to perform this task, that is, two industrial experts Virtual Force Pakistan and one academic expert from Chongqing University China. The survey questionnaire was modified according to the recommendations of experts; for example, the organization name and respondent name should be optional and use tabular form to answer the questions. The final survey questionnaire sample is given in link (https://tinyurl.com/yafmsx7e).

3.3.3 | Data source

The targeted population is essential to collect data.[55] Based on the objective of this study to target the population, we have used professional Emails, ResearchGate, and LinkedIn profiles. The snowball technique is used to spread the questionnaire survey to the target population. This technique has been used by other researchers in different fields of engineering.[52,53,55] We collect data from 83 respondents in 1-month time duration. After checking the responses manually, we came out with five incomplete responses and decided with team not to use them for future data analysis process. The detail of demographic data is discussed in Appendix D.

3.3.4 | Survey data analysis

We have adopted frequency analysis approach to analyze survey data. This is an effective approach to compare the respondents' opinions between variables and group of variables used in other software engineering fields as well.[55,57] The table of frequency analysis is given in Appendix B and Section 2. Only three practices P35 “Shift left strategy for flexible architecture 47%,” P45 “Ability to encompass multiple technology over multiple domains 48%,” and P58 “End-to-end application delivery processes to check value streams 49%” score less than 50%, which is the threshold value. However, all the challenges score more than 50%, and CCH13 “Lack of flexibility due to rigid Industrial constraints, i.e., 90%” frequency is the highest ranked challenge according to industry practitioners. This proves the mapping scheme and significance of factors and practices in DevOps implementation. The challenges are represented by “CCH” and practice by “P” as shown in Appendix B.

3.4 | Case study analysis

To measure the efficiency of RMDevOps, we have applied a case study approach, as the case study is an effective technique for evaluation.[14,50,58] Three case studies were conducted in different software firms to evaluate RMDevOps (see Section 4).

For follow-up, we also organized a feedback session to evaluate the RMDevOps related to following aspects:

a. Ease of use: assessment of model with respect to ease of use in an organization.
b. Satisfaction level of user: evaluate the satisfaction level of users with proposed model after implementation.
c. Structure of RMDevOps: evaluate the efficiency of key components of RMDevOps.

>Case study analysis technique could be used in different fields to measure the findings, quality of products, and is also helpful to investigate the weak areas of the development process.[50,59] Various existing studies of the other software engineering domains also used the same analysis approach.[46,57,60–66]

4 | READINESS MODEL FOR DevOps

The RMDevOps is based on a concept of SOVRM,[36] Rise RM,[67] CMMI,[11] and software process improvement readiness model (SPIRM).[61] There are various levels in this model, which were designed by the aforementioned literature study. The mention readiness models[36,61,67] used influence factors (i.e., success factors and challenges) to design the readiness levels of RMDevOps.

Based on existing readiness models of other software engineering fields, we have used CCHs to design the readiness levels of proposed model. After plotting the CCHs in five levels of RMDevOps, we map the investigated best practices with every CCH, which are significant to address them effectively and efficiently. We have conducted a questionnaire survey with DevOps experts to check the reliability of the mapping process before conducting the case study. The phases used to design the levels of RMDevOps are shown in Figure 5.

4.1 | Structure of RMDevOps

The challenges (RQ1) and practices (RQ2) are identified during SLR and are validated by survey (RQ3) to provide us the basis for developing components of proposed RMDevOps. The identified challenges were mapped by following the concept of CMMI,[11] Rise RM,[67] and SOVRM.[36] Further, we verified the mapping of factors and practices from survey participants to remove biasness in research. The structure of RMDevOps consists of the following components:

- Levels of RMDevOps with mapped challenges
- Components of RMDevOps
- Practices to map with challenges

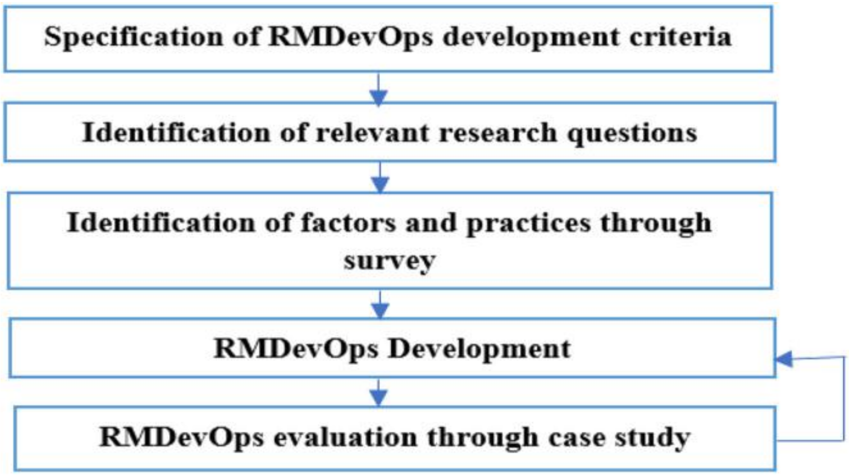


FIGURE 5 Readiness model for DevOps (RMDevOps) development flow process

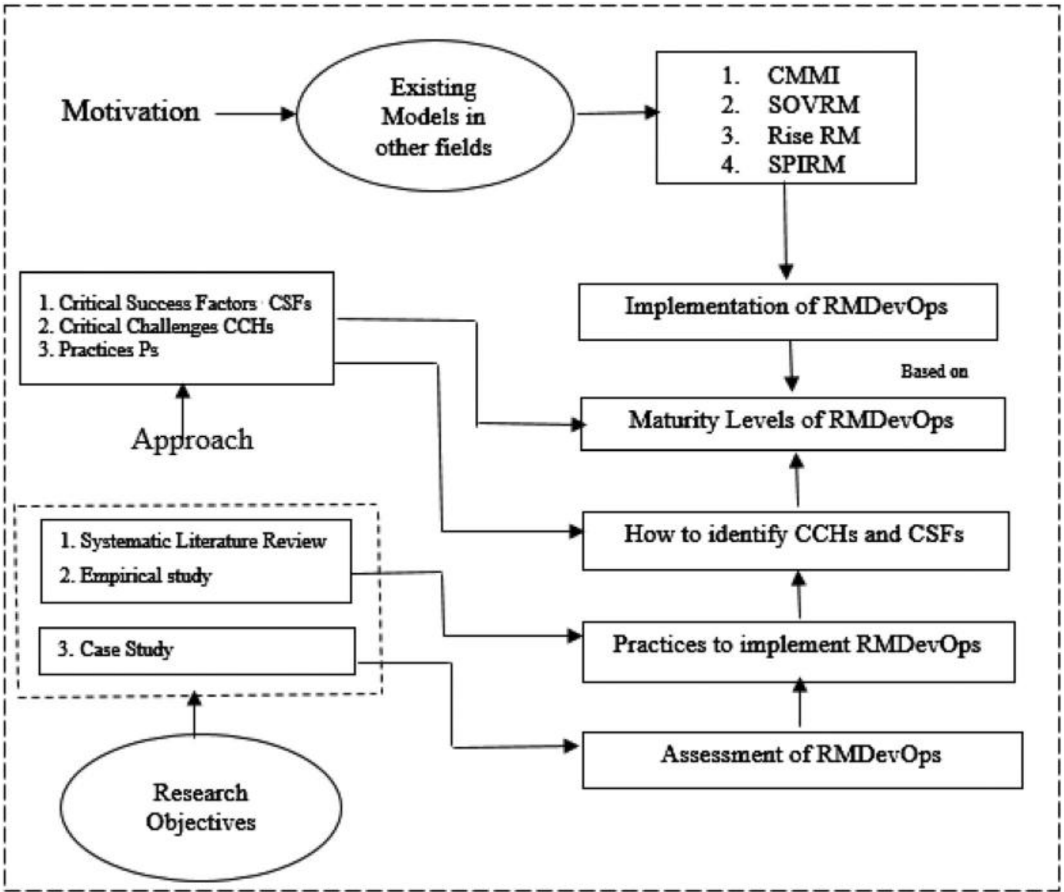


FIGURE 6 Technology roadmap: Relationship between RMDevOps components. CHH, critical challenge; CMMI, Capability Maturity Model Integration; RMDevOps, readiness model for DevOps; SPIRM, software process improvement readiness model; SOVRM, software outsourcing vendor readiness model

The RMDevOps is designed based on the investigated CCHs and best practices of DevOps. The proposed model assists the software firms to address the challenges practitioners faced while deploying DevOps activities. Figure 6 demonstrates the relationship between levels challenges and components of DevOps.

4.1.1 | Maturity levels of RMDevOps with mapped challenges

Each maturity level of RMDevOps consists of various challenges and practices. To reach a certain level of RMDevOps, software enterprise should satisfy all the challenges and practices mapped to that particular level. The readiness levels are developed with a comparison to the maturity levels of CMMI.[11] The defined maturity levels of the proposed model (RQ4) are presented in Figure 7 and briefly discussed in the subsequent sections:

Level 1: Initial

In this level, challenges depend on individual iterative and past experiences of an enterprise.[68] The findings from previous experiences were utilized to produce better quality production. Even there is sometimes reuse of past practices such as design patterns to avoid various risks.[67] Therefore, no formal process was defined in this level, just to avoid resistance, to adopt DevOps concept. Some classification is done in this level, for building and implementing knowledge, cohesive teamwork, and continuous practice and planning to measure communication across certain standards.

Level 2: Managed

This level plans to use several practices, for implementation of new concepts in an organization. The basic techniques to be used are identified by the organization and evaluated by policies defined during organization build.[67] The implementation of DevOps requires certain approaches and commitments from the organization, in the sense of participating in continuous development process in software industry. So considering the results from SLR, this level used various practices for DevOps management regarding issues such as awareness about DevOps, flexibility issues because of rigid industrial constraints, effective communication, resources accountability, and disintermediation of roles within teams. At Level 2, organization is capable to evolve a systematic approach to produce product.[69] The procedures to define main scope and variabilities are specified, in the future, which will in-line with integration approach.

Level 3: Defined

The main concern of Level 3 is to address all challenges identified during DevOps integration management, to plan several practices that are standardized and deployed in the whole enterprise.[69] Proper measures about uncertainties such as integrated tools and practices, heterogeneity in structure, infrastructure maintainability, use of immature automated tools, and threat modeling are covered to avoid risk in management, though using defined strategy, reusable artifacts are stored for automated artifacts on-demand troubleshooting.

Level 4: Qualitatively managed

Process and activities are standardized and integrated with software development cycle for proper deployment.[68,69] For this standardization, the focus of this level is to qualitatively analyze product quality goals and plans.[67] Functions are analyzed to evaluate legacy software, deep-seated company culture, and DevOps artifacts reusability to avoid tools boundless as they are valuable attributes for DevOps standard.

Level 5: Optimization

In Level 5, organization will explore and analyze new strategies of business planning based on customer feedback and strategic suggestions.[67] Predictable architecture refactoring, balancing skills, and high management strategies are involved to move one step forward, to satisfy business demands and customer needs in short interval of time.

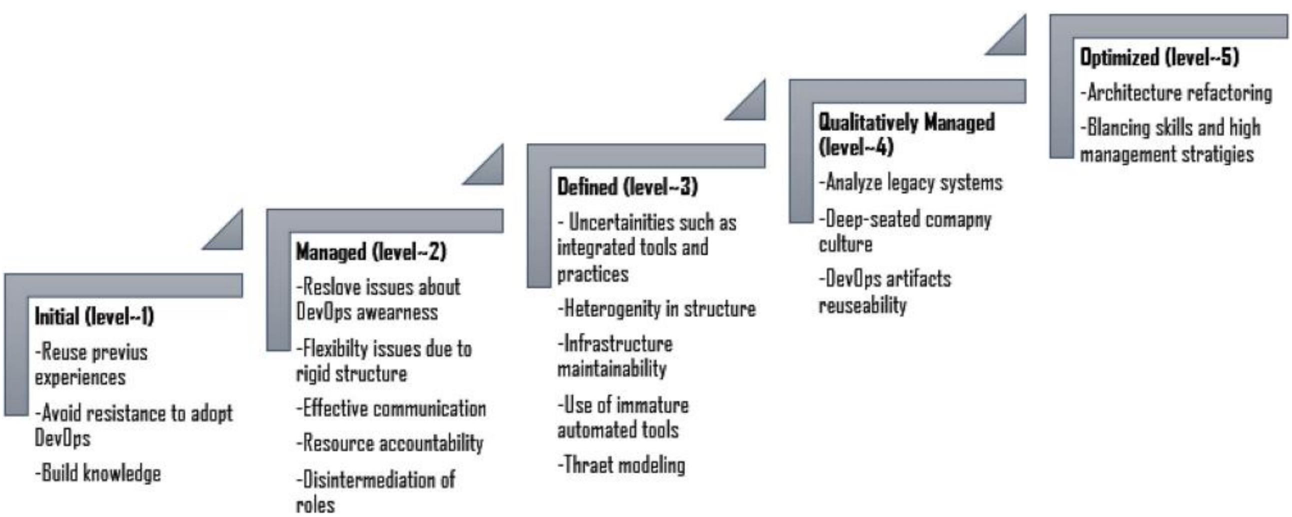


FIGURE 7 Levels and components of readiness model for DevOps (RMDevOps)

4.1.2 | Components of RMDevOps

The SPI model[35] was designed by categorizing the critical success factors and challenges into five levels of software process. Ali and Khan use the critical success factors are used to design the readiness level of SOPM.[33] This research followed the above-mentioned concepts to develop readiness levels of RMDevOps, which supports DevOps activities implementation in an organization with best practices. The detail about these practices is available (Appendix B, section 1). The levels of RMDevOps were derived from CMMI.[11] Eighteen CCHs were identified and categorized into five readiness levels of RMDevOps. The description of all levels and their respective challenges are shown in Table 4. To provide the guidelines for addressing the challenging factors, the best practices were mapped against each success factor based on the concepts gathered from SLR and survey respondents.

4.1.3 | Practices for each CCHs

In order to address the challenging factors of RMDevOps, we have identified 73 best practices using SLR. All the investigated best practices were mapped against the challenging factors of each readiness level based on authors understanding and concepts gathered from SLR. The mapping categories are further investigated through questionnaire survey to remove vagueness. The numbers of mapped best practices for each challenge are presented in Table 4, and the detailed practices that correspond to the challenges are given (Appendix B, section 1).

4.2 | RMDevOps assessment method

Motorola assessment tool[70] was adopted to evaluate the efficiency of RMDevOps. Using the components of the Motorola assessment tool, the degree of implementation of each best practice of every readiness level of RMDevOps was determined. Several existing studies also adopted the same assessment tool such as CMMI[11] and SOVRM[36] publications. Three different assessment aspects of Motorola assessment tools include[70]:

a. Approach: This aspect deals with support management and commitment of organization to utilize practices for concerned challenge.
b. Deployment: This aspect deals with scalability and consistency of practice deployment while observing challenge critically.
c. Results: This aspect focuses on positive outcomes in terms of the effect scale of the project.

TABLE 4 RMDevOps model with identified challenges

Sr no	Levels	# challenges	#practices
1	Initial	Resistance to adopt DevOps (CCH1)	6
2	Managed	Lack of flexibility due to rigid industrial constraints (CCH2)	5
		Lack of awareness about DevOps (CCH3)	4
		Lack of effective communication channel (CCH4)	4
		Resources accountability issues (CCH5)	6
		Risk of disintermediation of roles within teams (CCH6)	5
3	Defined	Lack of integrated testing tools and practices (CCH7)	8
		Lack of feedback and bugs prioritization (CCH8)	12
		Heterogeneity in development and operational structure (CCH9)	6
		Lack of continuous deployment infrastructure maintainability (CCH10)	9
		Use of immature automated deployment tools (CCH11)	6
		Problem of threat modeling scalability (CCH12)	12
4	Qualitatively managed	Lack of analytic frameworks of DevOps for business needs and technical QoS (CCH13)	5
		Deep-seated company culture and legacy software (CCH14)	8
		DevOps artifacts are bounded to certain tools (CCH15)	7
5	Optimized	Lack of strategic suggestions from leadership (CCH16)	4
		Lack of skilled staff for new technological stack (CCH17)	4
		Lack of explicit support for continuous architecture refactoring and optimization (CCH18)	7

Abbreviations: CCH, critical challenge; RMDevOps; readiness model for DevOps.

d. Likert scale (0, 2, 4, 6, 8, and 10) was used for each aspect which can be determined by given criteria in Appendix D.

The following steps of the Motorola tool have been adopted for RMDevOps assessment.

Step 1: Case study participants determine the score for three dimensions of the Motorola instrument for each CCH.
Step 2: The calculated scores of each challenge are added together and are divided by three. The final scores rounded to the nearest whole number.
Step 3: Repeat the Step 2 for each challenge. Add all of the calculated scores for the final results separately for each CCHs.
Step 4: If the determined score for a CCH is ≥7, this indicated that the best practices for that CCH are implemented effectively. If the score is less than 7, then the CCH practices considered weak.[70]
Step 5: To achieve a certain readiness level of RMDevOps, the organization should address all the CCHs of that level. For example, if an organization wants to attain maturity Level 4, it must address all the CCHs and practice of Level 4. By considering all aspects of Motorola assessment tool, an example is provided in Table 5.

4.2.1 | Assessment of RMDevOps

To fulfill the requirements of RQ5, we considered a case study approach to evaluate the efficiency of proposed RMDevOps using the Motorola assessment tool. The case study approach is an effective method for assessment and can provide adequate information from real world industry experience.[60] To conduct the case study, we have followed the guidelines of Runeson et al[59] and followed the studies in the domain of software engineering.[36,67] As RMDevOps was developed for the implementation in the real world, we have conducted case studies with three software organization that is two rounds. The main objective of real-world assessment is to check significance of RMDevOps for the implementation of DevOps activities in the software industry.

4.2.2 | RMDevOps assessment criteria

We use following approach to assess the efficiency of RMDevOps:

a. Easy to use: The focus of this criterion is to measure efficiency of how easily experts can understand implementation of RMDevOps.
b. User satisfaction: The focus of this criteria is to assess the user satisfaction level with RMDevOps.
c. Structure of RMDevOps: The objective this criterion is to identify efficiency of the key components of RMDevOps, which is based on the CCHs and their associated best practices.

RMDevOps assessment criteria are based on literature and have been adopted by many researchers.[71,72] The above-mentioned assessment criteria were used to identify factors that need further improvements in RMDevOps. These criteria will help to improve the efficiency of RMDevOps.

TABLE 5 Example of Motorola assessment tool

Sr.	Practices	"Approach 0, 2, 4, 6, 8, 10"	"Deployment 0, 2, 4, 6, 8, 10"	"Result 0, 2, 4, 6, 8, 10"	"Average"
P1-CCH1	Building DevOps knowledge	8	8	8	8
P2-CCH1	Cohesive team work to fill gap during isolation changes	8	8	8	8
P3-CCH1	Promoting team mindset and expertise	8	8	8	8
P4-CCH1	Implementing knowledge of DevOps practically	6	6	6	6
P5-CCH1	Continuous practice and planning to avoid resistance	8	8	8	8
P6-CCH1	Adjust objectives streamline to measure communication across silos	4	4	4	4
Average score					42
Final score = 42/6					7

5 | RMDevOps ASSESSMENT USING CASE STUDIES

We conducted three case studies in different software enterprise that are using DevOps practices. To conduct the case studies, we have randomly selected the organizations by visiting their profiles on LinkedIn and their organizational website.[36,67] On initial stage, we contacted the organization via email and contact number available on their websites. We write an invitation letter to invite the organizations for conducting the case study. The sample invitation letter is in link (https://tinyurl.com/y5hmhoqa). Finally, three organizations agreed to conduct a case study for the assessment of RMDevOps. The selected organizations are willing to share rich information about the adopted DevOps practices. We sent a guideline document to each participant of case study, which covers a brief description about RMDevOps and assessment procedure. The sample of guideline document can be found in a link (https://tinyurl.com/y2b46juh). The selected organizations assessed DevOps attributes of their enterprise against readiness levels of RMDevOps. From each organization, the contact participant communicated through email and Skype call to get a thorough understanding of the RMDevOps. It was also decided that each participant will consult with other colleagues during the assessment process to get the representative outcomes of RMDevOps.

Moreover, a feedback session was conducted with the case study participants to discuss the outcomes and efficiency of RMDevOps (Section 5). A questionnaire was designed to get the feedback from participants that can be found via link (https://tinyurl.com/y32xzjhu). The questionnaire consists of three parts: Part 1 is about demographic information about their organization, Part 2 contains evaluation criteria of RMDevOps as discussed in (Section 4), and in Part 3, list of all identified challenges and their corresponding best practices were enlisted for participants review and suggestions.

5.1 | Profiles of selected organizations participated in first round

The selected organizations tagged as A and B due to privacy issues original names are not disclosed. The demographic detail of organizations participated in case study is presented in Appendix D.

5.2 | Evaluation outcomes of case studies

A case study provides real-world perspective that suits our requirements to evaluate RMDevOps. Initially, we selected two organizations to conduct case study and communicated with each participant in both organizations with the aim to introduce the concept of RMDevOps. The summarized outcomes of both Organizations A and B are shown in Table 6. We just interacted with one responsible member of a team to collect the response of whole team as a single opinion. However, it was decided that each participant will consult each other colleagues during the assessment process in order to get the representative outcomes of RMDevOps. Moreover, each participant was asked to do assessment based on experiences of previous projects. Because of time and quality concerns, we requested the participants to complete a questionnaire at work sites. All the documents were provided to the case study participants via Email. The case study result renders that Organization A is fully satisfied with the structure of RMDevOps, but participants from Organization B suggest some modifications.

5.2.1 | Assessment outcomes of Organization A

The assessment results presented in Table 6 renders that Organization A is at initial level; the initial level consists of only one challenging factors (i.e., resistance to adopt DevOps), which contain score 7. In addition, we noted that only one challenge (i.e., lack of awareness about DevOps) of second level (managed) contain score 9. However, for achieving Level 2, Organization A should need to address all the challenges stated at Level 2. The overall result shows that the DevOps practices are very weak in Organization A. For the successful implementation of DevOps practices, Organization A should consider the best practices (Appendix B) that are suggested to address each challenge of readiness level.

5.2.2 | Assessment outcomes of Organization B

The results presented in Table 6 that Organization B is also at the initial level as only one challenge; that is, resource accountability issue scored “5,” if managed properly, will help organization to attain managed level successfully. However, issue of resources can be managed by following practices suggested in Appendix B. Therefore, for the satisfactory execution of DevOps practices, the organization needs to implement suggested best practices.

TABLE 6 Case study results of Organizations A and B

RMDevOps levels			Organization A		Organization B	
No	Levels	Critical challenges (CCHs)	Score	Status	Score	Status
1	Initial	Resistance to adopt DevOps (CCH1)	7	Implemented	7	Implemented
2	Managed	Lack of flexibility because of rigid industrial constraints (CCH2)	8	Implemented	7	Implemented
		Lack of awareness about DevOps (CCH3)	8	Implemented	9	Implemented
		Lack of effective communication channel (CCH4)	9	Implemented	8	Implemented
		Resources accountability issues (CCH5)	6	Not implemented	5	Not implemented
		Risk of disintermediation of roles within teams (CCH6)	5	Not implemented	7	Implemented
3	Defined	Lack of integrated testing tools and practices (CCH7)	1	Not implemented	0	Not implemented
		Lack of feedback and bugs prioritization (CCH8)	2	Not implemented	1	Not implemented
		Heterogeneity in development and operational structure (CCH9)	1	Not implemented	3	Not implemented
		Lack of continuous deployment infrastructure maintainability (CCH10)	1	Not implemented	1	Not implemented
		Use of immature automated deployment tools (CCH11)	1	Not implemented	0	Not implemented
		Problem of threat modeling scalability (CCH12)	3	Not implemented	1	Not implemented
4	Qualitatively managed	Lack of analytic frameworks of DevOps for business needs and technical QoS (CCH13)	1	Not implemented	2	Not implemented
		Deep-seated company culture and legacy software (CCH14)	4	Not implemented	6	Not implemented
		DevOps artifacts are bounded to certain tools (CCH15)	1	Not implemented	1	Not implemented
5	Optimized	Lack of strategic suggestions from leadership (CCH16)	1	Not implemented	2	Not implemented
		Lack of skilled staff for new technological stack (CCH17)	2	Not implemented	3	Not implemented
		Lack of explicit support for continuous architecture refactoring and optimization (CCH18)	0	Not implemented	3	Not implemented

Feedback summary

A feedback secession was also conducted with case study participants with the aim to assess RMDevOps from three different aspects, that is, ease of use, user satisfaction, and structure of RMDevOps. Both companies fill the feedback form that is provided to evaluate RMDevOps from three different aspects (ease of use, user satisfaction, and structure of RMDevOps). A sample of the feedback form is given in the link (https://tinyurl.com/y32xzjhu). These aspects were evaluated through qualitative measurement. Furthermore, in the feedback questionnaire, we also mention some important questions to get the suggestion to improve the proposed model. Each participant was asked to evaluate the RMDevOps efficiency against below mentioned criteria.

Ease of use. In this step, the case study participants were requested to assess the model with respect to their ease of use. The outcomes presented in Table 7 show that both the organizations (A and B) give positive feedback and are satisfied with evaluation outcomes of RMDevOps, as the proposed RMDevOps is easy to adopt and implement.

User satisfaction. User satisfaction level was evaluated and the results (Table 7) presented that both organizations show full satisfaction that the RMDevOps could be useful for the implementation of DevOps. They were agreed to deploy RMDevOps in software industry, which proves that experts considered RMDevOps as adequate model for all types of industry. They encouraged that RMDevOps will help organizations in the areas which need further improvements regarding DevOps activities.

TABLE 7 Feedback of Organizations A and B

	Organization perception (*N* = 2)							
	Positive			Negative			Neutral	
	SA	A	%	SD	D	%	N	%
User satisfaction								
RMDevOps is generic in nature and could be installed in different global software development organizations.	0	2	100	0	0	0	0	0
RMDevOps could help an on organization to understand weak and strong areas of process improvement	0	2	100	0	0	0	0	0
The use of RMDevOps could improve the software process of our organization.	0	2	100	0	0	0	0	0
I will prefer to adopt RMDevOps in my organization.	0	2	100	0	0	0	0	0
I am confident with the outcomes of RMDevOps.	0	2	100	0	0	0	0	0
The software tool of RMDevOps could facilitate the experts to implement DevOps.	0	2	100	0	0	0	0	0
RMDevOps is adequate model for software development organizations.	0	2	100	0	0	0	0	0
Ease of learning								
Representation of RMDevOps is simple.	0	2	100	0	0	0	0	0
RMDevOps need little DevOps knowledge to understand.	0	2	100	0	0	0	0	0
It is easy to understand the practices of critical challenges	0	2	100	0	0	0	0	0
It is easy to understand the Motorola assessment technique.	0	2	100	0	0	0	0	0
It is easy to implement RMDevOps for process improvement activities.	0	2	100	0	0	0	0	0
It is easy to recognize the maturity levels of RMDevOps along their critical challenges and practices.	0	2	100	0	0	0	0	0
RMDevOps needs training to fully understand.	0	2	100	0	0	0	0	0
Structure of RMDevOps								
1. The key components of RMDevOps are self-exploratory and no need of further explanation to be used effectively.	0	2	100	0	0	0	0	0
2. The RMDevOps components are practical and could be used in an organization.	0	2	100	0	0	0	0	0
3. The implementation of RMDevOps could assist an organization to identify issues relating to DevOps.	0	2	100	0	0	0	0	0
4. The five maturity levels of RMDevOps are enough to assess the DevOps maturity level of an organization.	0	1	50	0	0	0	1	50
5. Would you like to suggest further improvements or suggestions for DevOps?	0	2	100	0	0	0	0	0
6. Are there any additional components for RMDevOps and also give the reasons."	0	2	100	0	0	0	0	0

Where *N* = 2 represents results of Organizations A and B as a whole

Structure of RMDevOps. In third step, structure aspect was evaluated as shown in Table 7. Both organizations agreed that the structure of RMDevOps is clear and suitably arranged. The distribution of various levels on identified challenges and practices causes no confusion, while implementing this model organizations can check their maturity level for DevOps adoption.

We received some criticism only from Organization B. This organization reported some suggestions regarding levels of RMDevOps that we should modify RMDevOps levels to six instead of five. A criticism behind this was that the documentation must be improved as an organization is adopting new terminology and should be treated in separate level instead of merging it with previous practices as documentation plays an important role for building structure of an organization.

Another suggestion by Organization B is to shift one practice CCH10-P34 (lean requirements at scale) shown in (Appendix B) to CCH5, which will help organization to eliminate waste at the initial level if certain strategies and tools are not accountable for the new environment. In addition, some suggestions were also given to improve questionnaire for future. We utilized these suggestions in the next section where we discussed RMDevOps after modification.

5.3 | Modification of RMDevOps

Considering the results in first round of case studies, we further made some modifications in our model. The modifications were made to make RMDevOps highly usable for the software industry. We updated new level in start, that is, informal level for reuse of practices depending upon individual initiative, while dealing with documents and design code separately in initial level, to aid in implementation of new documentation and components.

Secondly, as mentioned in Section 5, we moved practice CCH14-P34 (lean requirements at scale) to CCH5 (resource accountability issues). Modifications were updated in RMDevOps as shown in Figure 8 and Table 8.

5.3.1 | After modification case study evaluation (second round)

We conducted additional case studies using Organizations B and C. We targeted one more Organization C to verify our changes and to remove biases while considering same organizations again. The profile of Organization B and new selected software Organization C is mentioned in Appendix D. The outcomes of case studies are discussed above in section.

TABLE 8 Level and components of readiness model for DevOps (RMDevOps) after modification

RMDevOps levels		Critical challenges (CCHs)	Practices
1	Informal	Nil	—
2	Initial	Resistance to adopt DevOps (CCH1)	6
3	Managed	Lack of flexibility because of rigid industrial constraints (CCH2)	5
		Lack of awareness about DevOps (CCH3)	4
		Lack of effective communication channel (CCH4)	4
		Resources accountability issues (CCH5)	7
		Risk of disintermediation of roles within teams (CCH6)	5
4	Defined	Lack of integrated testing tools and practices (CCH7)	8
		Lack of feedback and bugs prioritization (CCH8)	12
		Heterogeneity in development and operational structure (CCH9)	6
		Lack of continuous deployment infrastructure maintainability (CCH10)	9
		Use of immature automated deployment tools (CCH11)	5
		Problem of threat modeling scalability (CCH12)	12
5	Qualitatively managed	Lack of analytic frameworks of DevOps for business needs and technical QoS (CCH13)	5
		Deep-seated company culture and legacy software (CCH14)	8
		DevOps artifacts are bounded to certain tools (CCH15)	7
6	Optimizing	Lack of strategic suggestions from leadership (CCH16)	4
		Lack of skilled staff for new technological stack (CCH17)	4
		Lack of explicit support for continuous architecture refactoring and optimization (CCH18)	7

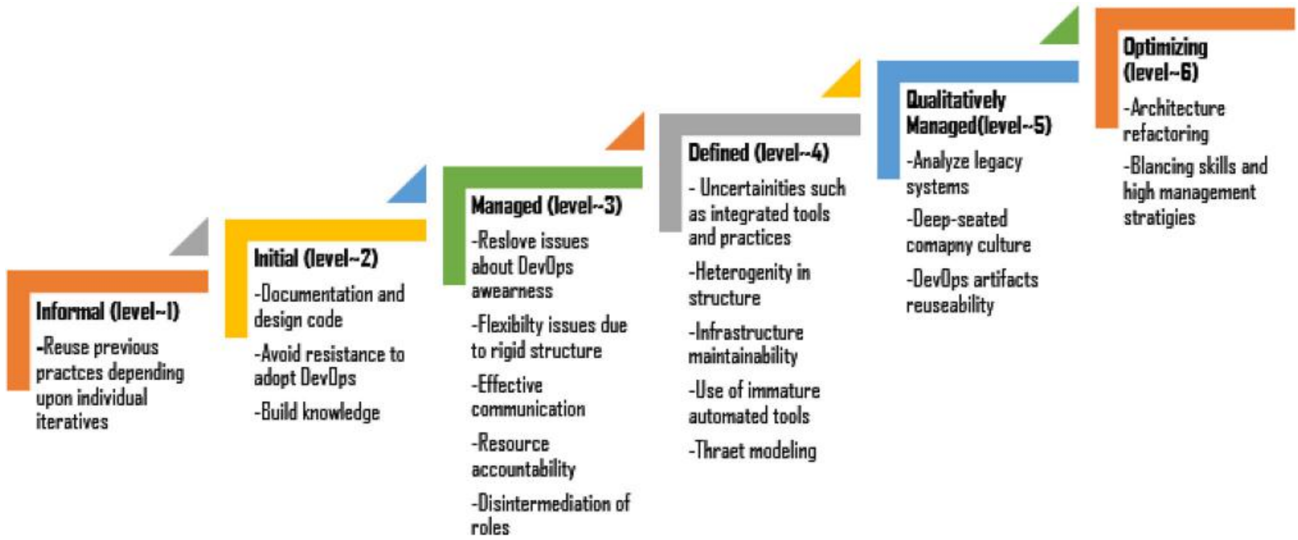


FIGURE 8 Readiness model for DevOps (RMDevOps) model after modification

5.3.2 | Additional case study in Organization B

The main purpose of conducting the second case study in Organization B is to verify the improvement in the RMDevOps model. We compare the outcomes of the first and second rounds of case study to assess the user satisfaction level with respect to the modified RMDevOps model. As most of suggestions were given by Organization B, so their point of view is essential in the second round of case study. Table 9 shows the

TABLE 9 Case study results of Organizations B and C after modification

RMDevOps levels			Organization B		New Organization C	
No	Levels	Critical challenges (CCHs)	Previous score	New score	Score	Status
1	Informal	Nil				
2	Initial	Resistance to adopt DevOps (CCH1)	8	8	8	Implemented
3	Managed	Lack of flexibility due to rigid industrial constraints (CCH2)	7	7	7	Implemented
		Lack of awareness about DevOps (CCH3)	8	8	8	Implemented
		Lack of effective communication channel (CCH4)	6	7	6	Not implemented
		Resources accountability issues (CCH5)	6	6	6	Not implemented
		Risk of disintermediation of roles within teams (CCH6)	7	7	7	Implemented
4	Defined	Lack of integrated testing tools and practices (CCH7)	4	4	4	Not implemented
		Lack of feedback and bugs prioritization (CCH8)	5	5	5	Not implemented
		Heterogeneity in development and operational structure (CCH9)	6	7	6	Not implemented
		Lack of continuous deployment infrastructure maintainability (CCH10)	3	3	3	Not implemented
		Use of immature automated deployment tools (CCH11)	4	4	4	Not implemented
		Problem of threat modeling scalability (CCH12)	5	5	5	Not implemented
5	Qualitatively managed	Lack of analytic frameworks of DevOps for business needs and technical QoS (CCH13)	2	2	2	Not implemented
		Deep-seated company culture and legacy software (CCH14)	3	3	3	Not implemented
		DevOps artifacts are bounded to certain tools (CCH15)	4	4	4	Not implemented
6	Optimizing	Lack of strategic suggestions from leadership (CCH16)	2	3	2	Not implemented
		Lack of skilled staff for new technological stack (CCH17)	5	5	5	Not implemented
		Lack of explicit support for continuous architecture refactoring and optimization (CCH18)	1	1	1	Not implemented

comparative outcomes of Organization B that shows some improvements in Organization B for the implementation of DevOps practices. The respondents from Organization B appreciated and were satisfied with the updated structure of RMDevOps. In feedback session, they did not add any further comments and suggestions, however agreed upon three basic aspects of assessment criteria.

5.3.3 | Case study in Organization C

We conducted same case study approach with Organization C after modification of RMDevOps. For conducting the case study with Organization C, we have followed the same procedure discussed in Section 4. The feedback form was sent to the Organization C via Email and their responses were collected within 20 days. The results in Table 9 shows that Organization C is also at initial level, but some challenges of Level 3 (managed) were effectively implemented. The demographic detail of Organization C is given in Appendix D.

The evaluation results of RMDevOps from Organization C are shown in Table 10. The respondents from Organization C strongly agreed upon the ease of use of RMDevOps and agreed with structure of the model, including levels and mapping of practices. In addition, give positive comments to use RMDevOps for indicating the readiness level of DevOps adoption in an organization. We did not find any suggestion or comment form Organization C for the improvement of RMDevOps.

TABLE 10 Feedback of Organization C

	Organization perception (*N* = 1)							
	Positive			Negative			Neutral	
	SA	A	%	SD	D	%	N	%
Ease of learning								
1. Representation of RMDevOps is simple.	0	1	100	0	0	0	0	0
2. RMDevOps need little DevOps knowledge to understand.	0	1	100	0	0	0	0	0
3. It is easy to understand the practices of critical challenges	0	1	100	0	0	0	0	0
4. It is easy to understand the Motorola assessment technique.	1	0	100	0	0	0	0	0
5. It is easy to implement RMDevOps for process improvement activities.	0	1	100	0	0	0	0	0
6. It is easy to recognize the maturity levels of RMDevOps along their critical challenges and practices.	0	1	100	0	0	0	0	0
7. RMDevOps needs training to fully understand.	0	1	100	0	0	0	0	0
User satisfaction								
1. RMDevOps is generic in nature and could be installed in different global software development organizations.	0	1	100	0	0	0	0	0
2. RMDevOps could help an on organization to understand weak and strong areas of process improvement.	0	1	100	0	0	0	0	0
3. The use of RMDevOps could improve the software process of our organization.	0	1	100	0	0	0	0	0
4. I will prefer to use RMDevOps in my organization.	0	1	100	0	0	0	0	0
5. I am confident with the results of RMDevOps.	1	0	100	0	0	0	0	0
6. The software tool of RMDevOps could facilitate the experts to implement DevOps.	0	1	100	0	0	0	0	0
7. RMDevOps is adequate model for software development organizations.	0	1	100	0	0	0	0	0
Structure of RMDevOps								
1. The key components of RMDevOps are self-exploratory and no need of further explanation to be used effectively.	0	1	100	0	0	0	0	0
2. The RMDevOps components are practical and could be used in industry.	0	1	100	0	0	0	0	0
3. The implementation of RMDevOps could assist an organization to identify issues relating to DevOps.	0	1	100	0	0	0	0	0
4. The five maturity levels of RMDevOps are enough to evaluate the DevOps maturity level of an organization.	0	1	100	0	0	0	0	0
5. Would you like to suggest further improvements or suggestions for DevOps?	0	1	100	0	0	0	0	0
6. Are there any additional components for RMDevOps and also give the reasons.”	0	1	100	0	0	0	0	0
Where *N* = 1 represents organization results as a whole								

6 | DISCUSSION AND THREAT AND VALIDITY

6.1 | Study findings

The aim of this paper is to develop a robust RMDevOps for the implementation of DevOps activities in software industry. The results of this study provide the guidelines to both practitioners and researchers for the successful execution of DevOps activities in their organization. Using RMDevOps, the software companies can improve their DevOps practices by addressing the CCHs and by implementing the suggested best practices.

a. RQ1 (CCHs of DevOps)

We have identified 18 CCHs of DevOps, reported in existing literature using SLR approach. The investigated challenges are briefly discussed in Section 3. These challenges are presenting the key areas of DevOps that needs to be fixed for the success and progression of DevOps practices.

b. RQ2 (best practices of DevOps)

A total of 73 best practices of DevOps were identified from the literature using the step by step guidelines of SLR approach. The investigated best practices (using SLR) were further mapped against each identified CCH. The identified best practices are significant to implement for the success and progression of DevOps practice in software industry.

c. c.RQ3 (validation of factors and practices)

We have developed a questionnaire survey to validate the identified challenges and practices and to use this mapping scheme to construct a comprehensive RMDevOps. The mapping scheme is given in Appendix B.

d. R4 (design a robust model)

The robust model was designed based on the concept of other readiness models in the field of software engineering.[36,61,67] The Motorola assessment tool was used for the assessment of RMDevOps. The same technique has been used by other researches for model assessment.[33,35,37] Each level of RMDevOps consists of challenges and best practices explored using SLR. All the components of RMDevOps are briefly discussed in Section 4. To attain a certain level of RMDevOps, the organizations should satisfy all the challenges and practices mapped to that level.

e. RQ5 (efficiency of RMDevOps)

To measure the efficiency of RMDevOps, we have conducted case studies with three different organizations (Section 4). The results from case studies show that RMDevOps is an effective model for software organizations to successfully implement the DevOps practices.

6.2 | Threats and validity

This study has some threats and validations regarding the outcomes of SLR that was utilized to develop the RMDevOps. While selecting primary studies and extracting the useful data, subjective decisions may be made. A reason behind making subjective decisions is that some primary studies do not have a clear description about DevOps challenges and practices. Therefore, to resolve these threats, we involved three external researchers to review the primary studies and used inter-rater reliability test.[54] The researchers were allowed to undertake an iterative selection process and can extract data comprehensively.

One more limitation was that we have covered only data from electronic databases, in which there might be some other materials, which have been missed. Studies that were published since this research work was in process might be missed. However, we believe that our findings cover all the related material published in literature articles. With respect to threat about questionnaire design, we validate it by consulting practitioners in pilot stage (to assess questionnaire) and made changes according to their suggestions before final questionnaire sample of (survey and case study questionnaires). To identify the status of 73 practices, we attached feedback form for further suggestions from organizations participating in the case study analysis.

The case study technique was used to validate the proposed model RMDevOps considering three organizations[71,72] with different expertise and size, having external validity. Thus, generalizing the outcomes on the RMDevOps into other organizations needs proper care and consideration.

7 | STUDY IMPLICATIONS

This study highlighted various factors and practices based on the stat of the art literature. The study conducted explored the factors that have negative influence on DevOps environment. The study provides the body of knowledge to the industrial DevOps experts and academic researchers to concentrate on the most significant challenges of DevOps in software organizations. The investigated challenges assist the experts to build strategies and tools to overcome these concerns of DevOps, which is significant for the development and progression of DevOps in software industry.

Furthermore, the model RMDevOps was proposed based on other models[33,37] by mapping all the factors and practices to each level of model. This model will help practitioners to improve their organization's standards and protocols by addressing the CCHs using best practices.

8 | CONCLUSION AND FUTURE WORK

The importance of continuous development process in software industry motivated us to design a readiness model for the successful implementation of DevOps activities. In this study, we have proposed a readiness model, that is, RMDevOps. Using RMDevOps organization can control challenges hindering in DevOps adoption in order to maintain continuous deployment environment. The RMDevOps was structured based on empirical results and existing models. The identified practices and CCHs were mapped with different levels of model. In order to assess our model, we have conducted a case study in two rounds before and after modifications involving three software enterprises. The outcomes from first round case study come out with some modifications and suggestions to improve RMDevOps. The changes include the addition of one new level in start for reuses of previous practices and keep documents and design code separate, to aid in implementation of new documentation and components. Thus, RMDevOps has now six levels instead of five. Each level contains various challenges to be overcome with defined practices. These changes can enhance the capability and usability of RMDevOps in the software industry. The assessment of model has been done by using the Motorola assessment tool.[70]

Furthermore, we gathered suggestions from experts for RMDevOps model based on three criteria that are easy to use, user satisfactory, and structure of RMDevOps. The objective of gathering these suggestions is to improve RMDevOps for further analysis.

In future, we planned to extend RMDevOps to measure security requirements and data assessment categories in DevOps in collaboration with several software industries. Moreover, we planned to extend RMDevOps with specific characteristics suited for various new technologies, data science, and security. In addition, we will work on DevOps data quality assessment to modify RMDevOps, as better quality assessment model that will help organizations to improve DevOps implementation in various aspects.

ACKNOWLEDGMENTS

This research was supported by National Social Science Foundation of China under grant 17XFX013. The authors are grateful to the Deanship of Scientific Research, King Saud University for supporting through Vice Deanship of Scientific Research Chairs.

ORCID

Saima Rafi https://orcid.org/0000-0002-9807-6235
Muhammad Azeem Akbar https://orcid.org/0000-0002-6880-4991
Sajjad Mahmood https://orcid.org/0000-0001-5786-5118
Abdu Gumaei https://orcid.org/0000-0001-8512-9687

REFERENCES

1. Perera P, Silva R, Perera I. *Improve software quality through practicing DevOps, 2017 Seventeenth International Conference on Advances in ICT for Emerging Regions (ICTer), Colombo*, 2017:1-6. https://doi.org/10.1109/ICTER.2017.8257807
2. Lwakatare LE, Kuvaja P, Oivo M. Relationship of DevOps to agile, lean and continuous deployment. *International Conference on Product-Focused Software Process Improvement, Springer, Cham*. 2016:399-415.
3. Dyck A, Penners R, Lichter H. Towards definitions for release engineering and DevOps. *2015 IEEE/ACM 3rd International Workshop on Release Engineering*. IEEE 2015:3.
4. Smeds J, Nybom K, Porres I. Devops: a definition and perceived adoption impediments. *Agile Processes, in Software Engineering, and Extreme Programming, Springer*. 2015:166-177.

5. Rong G, Zhang H, Shao D. CMMI guided process improvement for DevOps projects: an exploratory case study. *Proceedings of the International Conference on Software and Systems Proces, ACM*. 2016:76-85.
6. P. Team. *CMMI for development, version 1.3, improving processes for developing better products and services*. no. CMU/SEI-2010-TR-033. Software Engineering Institute. 2010:8-9.
7. Trihinas D, Tryfonos A, Dikaiakos MD, Pallis G. Devops as a service: pushing the boundaries of microservice adoption. *IEEE Internet Comput*. Jun. 2018;22(3):65-71.
8. Badshah S, Khan AA, Khan B. Towards process improvement in DevOps: a systematic literature review. *Proceedings of the Evaluation and Assessment in Software Engineering*. 2020:427-433.
9. Forsgren N, Tremblay MC, VanderMeer D, Humble J. DORA platform: DevOps assessment and benchmarking. *International Conference on Design Science Research in Information System and Technology, Springer, Cham*. 2018:436-440.
10. McCarthy MA, Herger LM, Khan SM, Belgodere BM. Composable DevOps: automated ontology-based DevOps maturity analysis. *2015 IEEE International Conference on Services Computing, IEEE*. 2018:600-607.
11. Chrissis MB, Konrad M, Shrum S. CMMI for development: guidelines for process integration and product improvement. Pearson Education 2011.
12. Rafi S, Yu W, Akbar MA. RMDevOps: a road map for improvement in DevOps activities in context of software organizations. In Proceedings of the Evaluation and Assessment in Software Engineering. 2020:413-418.
13. Debois P. Agile infrastructure and operations: how infra-gile are you? *Agile 2008 Conference*. IEEE; 2008:202-207.
14. Lwakatare LE, Kilamo T, Karvonen T, et al. DevOps in practice: a multiple case study of five companies. *Inf Softw Technol*. 2019;114:217-230.
15. Khan AA, Shameem M. Multicriteria decision-making taxonomy for DevOps challenging factors using analytical hierarchy process. *J Softw-Evol Proc*. 2020;32(10):11-13, e2263.
16. Leite L, Rocha C, Kon F, Milojicic D, Meirelles P. A survey of DevOps concepts and challenges. *ACM Comput Surv (CSUR)*. 2019;52(6):1-35.
17. Woods E. Operational: the forgotten architectural view. *IEEE Softw*. 2016;33(3):20-23.
18. Bolscher R, Daneva M. Designing software architecture to support continuous delivery and DevOps: a systematic literature review. *ICSOFT*. 2019: 27-39.
19. Ambler SW. Disciplined agile delivery and collaborative DevOps. *Cutter IT J*. 2011;24(12):18.
20. Beigi-Mohammadi N, Litoiu M, Emami-Taba M, et al. A DevOps framework for quality-driven self-protection in web software systems. *Proceedings of the 28th Annual International Conference on Computer Science and Software Engineering, IBM Corp*. 2018:270-274.
21. Lwakatare LE, Kuvaja P, Oivo M. Dimensions of devops. *International Conference on Agile Software Development, Springer, Cham*. 2015:212-217.
22. Agarwal SG, Choudhury T. Continuous and integrated software development using DevOps. *2018 International Conference on Advances in Computing and Communication Engineering (ICACCE) IEEE*; 2018: 290-293.
23. Humble J, Farley D. *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation (Adobe Reader)*. Boston, MA: Pearson Education; 2010.
24. Kamuto MB, Langerman JJ. Factors inhibiting the adoption of DevOps in large organisations: South African context. In 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), IEEE. 2017:48-51.
25. Waseem M, Liang P. Microservices architecture in DevOps. In 2017 24th Asia-Pacific Software Engineering Conference Workshops (APSECW), IEEE. 2017:13-14.
26. Kuusinen K, Balakumar V, Jepsen SC, et al. A large agile organization on its journey towards DevOps. *2018 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), IEEE*. 2018:60-63.
27. Mansfield-Devine S. DevOps: finding room for security. *Netw Sec*. 2018;2018(7):15-20.
28. Elberzhager F, Arif T, Naab M, Süß I, Koban S. From agile development to devops: going towards faster releases at high quality—experiences from an industrial context. *International Conference on Software Quality, Springer, Cham*. 2017:33-44.
29. Riungu-Kalliosaari L, Mäkinen S, Lwakatare LE, Tiihonen J, Männistö T. DevOps adoption benefits and challenges in practice: a case study. *International Conference on Product-Focused Software Process Improvement, Springer, Cham*. 2016:590-597.
30. Mohamed SI. DevOps shifting software engineering strategy-value based perspective. *Int J Comput Eng*. 2015;17(2):51-57.
31. Teixeira DS. *Maturity model for DevOps* (doctoral dissertation). 2019. https://repositorio.iscte-iul.pt/bitstream/10071/20297/1/Master_Daniel_Simoes_Teixeira.pdf
32. Zarour M, Alhammad N, Alenezi M, Alsarayrah K. *A research on DevOps maturity models*. no. 2019;3:4854-4862.
33. Ali S, Khan SU. Software outsourcing partnership model: an eval-uation framework for vendor organizations. *J Syst Softw*. 2016;117:402-425.
34. Lei D, Tang J, Li Z, Wu Y. Using low-rank approximations to speed up kernel logistic regression algorithm. *IEEE Access*. 2019;7:84242-84252.
35. Niazi M, Wilson D, Zowghi D. Organisational readiness and software process improvement. In: *Product-Focused Software Process Improvement (Lecture Notes in Computer Science)*. Vol.4589 Berlin, Germany: Springer-Verlag; 2007:96-107.
36. Khan SU. Software outsourcing vendors' readiness model (SOVRM). Ph.D. dissertation, School Comput. Math., Keele Univ., Keele, U.K: 2011.
37. Akbar MA. SRCMIMM: managing requirements change activities in global software development: student research abstract. *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing, ACM*; 2019:1633-1636.
38. Rahman AU, Williams L. Software security in devops: synthesizing practitioners' perceptions and practices. *2016 IEEE/ACM International Workshop on Continuous Software Evolution and Delivery (CSED), IEEE*. 2016:70-76.
39. Rockart JF. Chief executives de ne their own data needs. *Harv Bus Rev*. 1979;57(2):81-93.
40. Shahin M, Babar MA, Zhu L. The intersection of continuous deployment and architecting process: practitioners' perspectives. *Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, ACM*. 2016:44-44.
41. Shahin M, Babar MA, Zahedi M, Zhu L. Beyond continuous delivery: an empirical investigation of continuous deployment challenges. *Proceedings of the 11th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, IEEE Press*. 2017:111-120.
42. Brereton OP, Kitchenham BA. The scope of EPIC case studies. *EPIC technical Report EPIC*. 2007:1-10.
43. Khan SU, Niazi M, Rashid A. Factors influencing clients in the selection of offshore software outsourcing vendors: an exploratory study using a systematic literature review. *J Syst Softw*. 2011;84(4):686-699.
44. Turner M, Kitchenham B, Brereton P, Charters S, Budgen D. Does the technology acceptance model predict actual use? A systematic literature review. *Inf Softw Technol*. 2010;52(5):463-479.

45. Chen L, Babar MA, Zhang H. Towards an evidence-based understanding of electronic data sources. In *14th International Conference on Evaluation and Assessment in Software Engineering (EASE)*. 2010:1-4.
46. Akbar MA, Sang J, Khan AA, et al. Success factors influencing requirements change management process in global software development. *J Comput Lang*. 2019;51:112-130.
47. Khan SU, Niazi M, Ahmad R. Critical success factors for off-shore software development outsourcing vendors: a systematic literature review. *Proc. IEEE Int. Conf. Global Softw. Eng. (ICGSE)*. 2009:207-216.
48. Kitchenham B, Charters S. Guidelines for performing systematic literature reviews in software engineering. Keel Univ., Keele, U.K., Tech. Rep. EBSE-2007-01. 2007.
49. Zhang H, Babar MA, Tell P. Identifying relevant studies in software engineering. *Inf Softw Technol*. 2011;53(6):625-637.
50. Afzal W, Alone S, Glocksien K, Torkar R. Software test process improvement approaches: a systematic literature review and an industrial case study. *J Syst Software*. 2016;111:1-33.
51. Khan A, Keung J, Niazi M, Hussain S, Ahmad A. Systematic literature review and empirical investigation of barriers for software process improvement in global software development: client-vendor perspective. *Inf Softw Technol*. 2017;87:180-205.
52. Shameem M, Kumar C, Chandra B, Khan AA. Systematic review of success factors for scaling agile methods in global software development environment: a client-vendor perspective. *Software Engineering Conference Workshops (APSECW), IEEE*: 2017:17-24.
53. Khan AA, Keung J. Systematic review of success factors and barriers for software process improvement in global software development. *IET Softw*. 2016;10(5):125-135.
54. Afzal W, Torkar R, Feldt R. A systematic review of search-based testing for non-functional system properties. *Inf Softw Technol*. 2009;51:957-976.
55. Akbar MA, Sang J, Khan AA, Shafiq M. Towards the guidelines for requirements change management in global software development: client-vendor perspective. *IEEE Access*. 2019;7:76985-77007.
56. Corbin J, Strauss A. *Basics of Qualitative Research: Grounded Theory Procedures and Techniques*. Beverley Hills, IA: Sage; 1990.
57. Niazi MK. A framework for assisting the design of effective implemen-tation strategies for software process improvement. Ph.D. dissertation, Faculty Inf. Technol., Univ. Technol. Sydney, Ultimo, NSW, Australia; 2004.
58. Senapathi M, Buchan J, Osman H. DevOps capabilities, practices, and challenges: insights from a case study. *Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018, ACM*. 2018:57-67.
59. Runeson P, Host M, Rainer A, Regnell B. *Case Study Research in Software Engineering: Guidelines and Examples*. Hoboken: John Wiley & Sons; 2012.
60. Yin RK. *Case Study Research: Design and Methods*. Newbury Park, CA, USA: Sage; 2013.
61. Niazi M, Wilson D, Zowghi D. A maturity model for the imple-mentation of software process improvement: an empirical study. *J Syst Softw*. 2005;74:155-172.
62. Mateen A, Amir H. Enhancement in the effectiveness of requirement change management model for global software development. arXiv preprint arXiv:1605.00770. 2016.
63. Ali S, Iqbal N, Hafeez Y. Towards requirement change management for global software development using case base reasoning. *Mehran Univ Res J Eng Technol*. 2018;37(3):639-652.
64. Fenton N, Bieman J. *Software Metrics: A Rigorous and Practical Approach*. 9th ed. USA: CRC press; 2014.
65. Hussain S, Ehsan N, Nauman S. A strategic framework for requirements change in technical projects: case study of a R & D project. *2010 3rd SecurityIEEE International Conference on Computer Science and Information Technology (ICCSIT)*. 2010;5:354-358.
66. Hussain W. Reflections on requirements change management in global software development: a multiple case study. *In Global Software Engineering Workshops (ICGSEW), 2016 IEEE 11th International Conference on*. 2016:77-79.
67. Garcia VC. RiSE reference model for software reuse adoption in Brazilian companies. From web site http://ivanmachado.com.br/research/rise/thesis/files/2010_ViniciusGarcia_phd.pdf accessed on 25/03/2016. 2010.
68. Spoelstra W. Reusing software assets in agile development organizations—a management tool. from web site http://essay.utwente.nl/59917/1/MA_thesis_W_Spoelstra.pdf accessed on 30/03/2015. 2010.
69. Younoussi S, Roudies O. A new reuse capability and maturity model: an overview. *Proceedings of the 2018 International Conference on Software Engineering and Information Management, ACM*. 2018:26-31.
70. Daskalantonakis MK. Achieving higher SEI levels. *IEEE Softw*. 1994;11(4):17-24.
71. Khan A, Keung JW, Abdullah-Al-Wadud M. SPIIMM: toward a model for software process improvement implementation and management in global software development. *IEEE Access*. 2017;5:13720-13741.
72. Mufti Y, Niazi M, Alshayeb M, Mahmood S. A readiness model for security requirements engineering. *IEEE Access*. 2018;6:28611-28631.

How to cite this article: Rafi S, Yu W, Akbar MA, Mahmood S, Alsanad A, Gumaei A. Readiness model for DevOps implementation in software organizations. *J Softw Evol Proc*. 2020;e2323. https://doi.org/10.1002/smr.2323

APPENDIX A.

A.1 | SELECTED PRIMARY STUDIES AND QES

https://tinyurl.com/y2pfjqst

APPENDIX B.

B.1 | SECTION 1: MAPPING OF IDENTIFIED CHALLENGES AND PRACTICES

https://tinyurl.com/y69u6ss5

APPENDIX C.: SECTION 2: RESPONSE OF SURVEY PARTICIPANTS

https://tinyurl.com/yy936f2c

APPENDIX D.

D.1 | COMPONENTS OF MOTOROLA ASSESSMENT TOOL

https://tinyurl.com/y24tjbd2

D.2 | DEMOGRAPHIC DATA ANALYSIS OF SURVEY PARTICIPANTS

https://tinyurl.com/y5kvtfnl