



Characterizing DevOps Culture: A Systematic Literature Review

Mary Sánchez-Gordón  and Ricardo Colomo-Palacios  

Østfold University College, 1757 Halden, Norway
{mary.sanchez-gordon, ricardo.colomo-palacios}@hiof.no

Abstract. Time and quality pressures are affecting software process in all its stages. One of the proposed solutions to these pressures is DevOps. DevOps is aimed to increase the frequency, quality and speed of deploying software from development into production by means of new organizational structures and processes with a high degree of automation. Several authors underlined the fact, that beyond the tool chain, DevOps is a culture shift. However, to date the characterization of DevOps culture remains unclear. In this paper, authors tackle this problem by means of a Systematic Literature Review. Results provide a deeper understanding of the phenomena from human factor's perspective.

Keywords: DevOps · Culture · Empathy · Systematic Literature Review
Human factors

1 Introduction

For software makers, one way to gain a sustainable competitive advantage is to deliver products and new features to customers considerably faster than before, if not near to real-time [1]. In this scenario, continuous software engineering (CSE) is a new approach aiming at establishing strong connections between software engineering activities in order to accelerate and increase the efficiency of the software process [2]. CSE can be related to DevOps [2, 3], which has risen to the fore as a prominent trend in the software engineering community and attracted growing attention from researchers in the last years, especially since 2014 [3]. For instance, Bosch [4] provides an overview of the adoption of CSE practices at large companies producing software-intensive systems. More recently, Fitzgerald and Stol [5] have proposed a roadmap and agenda for CSE. The findings in [6] have confirmed that DevOps is as an evolution of agile software development and is informed by a lean principles background. However, according to [1], research on continuous deployment is still in its infancy, despite the industrial relevance of the topic. In support of that, the results of a recent systematic mapping study [3] emphasizes both continuous practices and the term DevOps are vaguely defined and loosely used in the software engineering community. Likewise, other mapping [7] points out that there is no standard definition for DevOps. By reviewing the published literature, one can see that DevOps efficiently integrates development, delivery, and operations, thus facilitating a lean, fluid connection of these traditionally separated silos [8]. DevOps integrates also any technique aiming to

decrease the time between changing a system and transferring that change to the production environment, including continuous deployment but also practices like continuous monitoring [9]. But the most common interpretation is that DevOps is about culture [3]: DevOps means a culture shift toward collaboration between development, quality assurance, and operations [8], or DevOps is about aligning the incentives of everybody involved in delivering software [10] where its success is based on four principles:

- Culture. Joint responsibility for the delivery of high quality software.
- Automation. Automation in all development and operation steps towards rapid delivery and feedback from users.
- Measurement. All process must be quantified to understand delivery capability and setting goals to improve the process.
- Sharing. It is crucial the sharing of knowledge enabled by tools.

Furthermore, a novel perspective is the notion of DevOps as a superset of values, principles, methods, practices—including continuous practices—and tools [3]. This proposal is based on the point of view several forefront figures of the movement [10–12], but is focused on what one might consider a “meta definition” of the concept. In this scenario, it is not surprising that human aspects are taken in account by DevOps because software is a product of human activities that incorporates our problem solving capabilities, cognitive aspects, and social interaction [13]. In other words, software is intensive in human capital [14, 15]. Indeed, in DevOps, tools are important but people are an integral part of any human-designed complex system [16, 17]. Therefore, how we grow DevOps culture and practices in our organizations needs more attention [16]. In fact, [8] highlights that a key lesson for companies which embraced DevOps was not to underestimate the needed culture shift. In support of that view, a more recent study [18] reveals that DevOps is more a cultural shift for IT than a process or tools shift.

In the light of that, a key question, which will facilitate the understanding of the current status of research and address further investigation, is “How the scientific literature is characterizing DevOps Culture?”. To the best of authors’ knowledge, there are not published secondary studies about this topic. This paper is aimed to bridge this gap by conducting a systematic literature review (SRL) on the cultural side of DevOps.

The structure of the paper is as follows. The remainder of this section analyzes the works related with our proposal. Section 2 presents the design of this SRL. In Sect. 3 reports on the results of the SRL. Finally, Sect. 4 summarizes a conclusion and future research.

1.1 Related Works

This SLR focuses on DevOps culture. Before performing this study, an initial study was conducted to identify the existing secondary studies related to the topic. In order to obtain the maximum information about this topic, we searched the following two major online search academic article search engines: Scopus and Google Scholar. Given that, both of them cover all major publisher venues—e.g. Elsevier, Springer, ACM and IEEE—, they were estimated as enough for this initial purpose.

The searches were conducted in May 2018 using the search string (“*Culture*” AND “*DevOps*”) AND (“*Systematic review*” OR “*Systematic literature review*” OR “*Systematic mapping*” OR “*Mapping study*” OR “*Multivocal review*” OR “*Multivocal literature review*”). When the searches were performed, 19 results were found in Scopus while Google Scholar showed 221 results. However, most of them were not actually a secondary study. After reviewing the literature on secondary studies for similar research objectives, it can be identified that there is no previously published search on the topic.

2 Research Methodology

This study was carried out following Kitchenham and Charters guidelines on Systematic Literature Review (SLR) [19]. In what follows, an overview of this SLR is presented.

2.1 Planning

In this stage, a SLR protocol was adapted to define the plan for the review. The protocol comprises research background, research questions, search strategy, study selection criteria and procedures, data extraction, and data synthesis strategies to make sure that the study is undertaken as planned and reduce the possibility of researcher bias. In this review protocol, the whole study timetable was not decided from the beginning, but rather the actual timetable of the study and results produced were recorded as the study progressed.

Objectives and Research Questions. To get an explicit view of the current definition of DevOps culture, this SLR is conducted with the following specific objectives in mind. The objectives of this study are threefold. First, we would like to understand the attributes that define DevOps culture. Second, authors would like to investigate and find out the emotional phenomenon behind DevOps culture. Finally, we would like to see if there is a growing interest in the field or not.

In order to achieve these goals, the research objectives were translated into specific research question as follows:

1. What are the documented attributes of DevOps culture?
2. What emotional phenomenon could be experienced by people in the DevOps culture? and
3. What is the trend of studies related to DevOps culture in the scientific literature?

The keywords used to find an answer to the research questions were two: “DevOps” and “Culture”. Therefore, the search string was “*Culture*” AND “*DevOps*”.

Search Strategy and Search Process. The search strategy includes search resources and search process. Each one of them is detailed as follows:

Search Resources. In order to find the scientific literature available about DevOps culture, the search was performed on five electronic databases: (i) ACM Digital Library

(ii) IEEE Xplore Digital Library, (iii) ScienceDirect, (iv) Wiley Online Library and (v) SpringerLink.

Search Process. The overall search process is depicted in Fig. 1 and is explained in what follows.

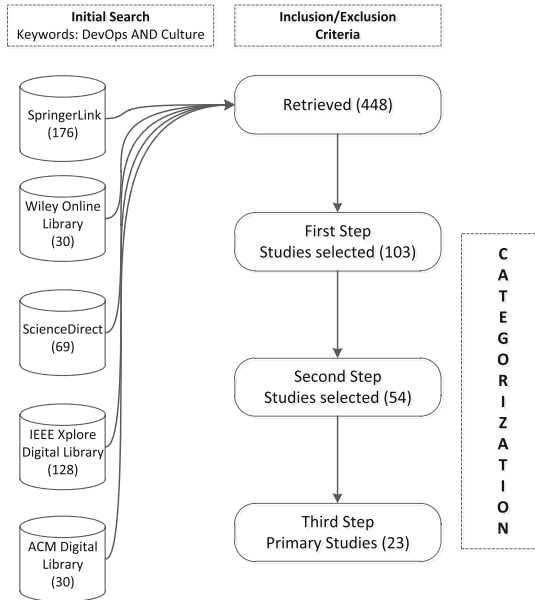


Fig. 1. Search process description

First step, the search string was applied in May 2018, returning 448 papers (in total). By manual inspection of abstract and the keywords in text context, the irrelevant studies were removed and a set of 103 unique papers remained. If multiple studies with the same title by the same author(s) were found, the most recent one was included and the rest were excluded. Moreover, only studies written in English language and electronically available were included.

Second step, all 103 papers were reviewed based on full text, and then were classified into two types:

- Relevant papers: if the paper satisfies the two inclusion criteria (explained in what follows).
- Excluded papers: other papers, which are not relevant to the topic.

A paper is kept in this study if it satisfies one of the two criteria:

- The paper is explicitly related to the DevOps and reveals some cultural aspect.
- The paper is relevant to software engineering research.

Some of the reasons for elimination were:

- The short versions of studies (with less than 4 pages).
- Book chapters. As it is generally difficult to determine how robust their findings are and if they have been subjected to peer review. However, chapters from books that are compiled as scientific articles or conference proceedings were included in this SLR.

This list was reviewed in order to check for inconsistencies. When there was doubt or disagreement about the classification of a paper, it was included in the relevant group, leaving the chance of discarding the paper during the next phase when the full texts of the papers were studied again. As a result, 54 papers were classified as relevant.

Third step, each full paper was retrieved and read to verify its inclusion or exclusion. But this time, we attempt to identify the cultural aspect in the results or discussion. The reason for exclusion or inclusion in this third phase was documented. The result of this step was that 23 papers were classified as relevant.

Fourth step, in order to check the consistency of the inclusion/exclusion decisions, a test-retest approach and re-evaluation of a random sample of the primary studies was made. However, there is a risk that some papers have been missed. Therefore, this study cannot guarantee completeness, but it can still be trusted to give a good overview of the relevant literature on DevOps culture.

2.2 Data Extraction

The data extracted from each paper was documented in a spreadsheet and kept in a reference manager. The bibliographic details for all the 23 primary studies are available in appendix A. In this paper, the primary studies are referred in the form of [S01],..., [S23] and these labels are the same as in the appendix. After selection of the primary studies, the following data was extracted: (i) Source (journal or conference), (ii) Title, (iii) Authors, (iv) Publication year (v) Classification according to a set of categories (see Table 1), (vii) Summary of the research. Based on (at least) the title, abstract and introduction of each study, a set of initial categories was created and assigned to them. When the assignment of studies to categories could not be clearly determined in this way, more about the study was considered. This process was inspired in open coding, memoing and constant comparison techniques proposed by Ground Theory. Thus, the categories were emerged and they were updated or clarified during the classification process as necessary. Moreover, both the categories and the assignment of studies to the categories were further refined. That means that an attribute generalization and iterative refinement was done. We used a spread sheet to record this process and a whiteboard and post-it notes to get a visual representation of the categories. As a result, a characterization of DevOps culture was built. Though we did not a-priori develop a categorization scheme for this research, we were broadly interested in: (i) Collaboration (ii) Sharing knowledge, and (iii) Communication.

Table 1. Characterization of DevOps culture.

ID	Attribute	Frequency	%	Primary studies
1	Communication	22	14,10	[S02]–[S23]
2	Collaboration	19	12,18	[S01]–[S05],[S07],[S09],[S10],[S12],[S13],[S15]–[S23]
3	Feedback (Continuous and immediate)	17	10,90	[S02]–[S10],[S12],[S15]–[S20],[S23]
4	Responsibility (personal/mutual)	17	10,90	[S01],[S02],[S04],[S05],[S07],[S09],[S12]–[S16],[S18]–[S23]
6	Sharing knowledge	15	9,62	[S01],[S02],[S06],[S07],[S10],[S12],[S13],[S16]–[S23]
5	Improvement cycle	15	9,62	[S03]–[S05],[S08],[S09],[S12],[S15]–[S23]
7	Transparency	12	7,69	[S01],[S02],[S05],[S07],[S15]–[S20],[S22],[S23]
8	Commitment and agreement	9	5,77	[S01],[S05],[S08],[S09],[S14],[S16],[S17],[S20],[S23]
9	New personnel and ideas	8	5,13	[S03],[S05],[S06],[S07],[S08],[S16],[S18],[S22]
10	Leadership	7	4,49	[S05],[S06],[S13],[S16],[S19],[S20],[S21]
11	Blameless	6	3,85	[S05]–[S07],[S12],[S13],[S16]
12	Experimentation	5	3,21	[S01],[S07],[S16],[S18],[S19]
13	Trust	4	2,56	[S05],[S18],[S21],[S22]
	Total	156	100,00	

2.3 Study Quality Assessment

In this study, each paper was assessed for quality at the same time as the data extraction process was performed. This process provided information about author and source, as well as the minimum information required to establish credibility. 22 of the 23 selected studies satisfied the quality questionnaire: (i) Does the paper introduce any aspect of culture? (ii) Is there a clear statement of the aims of the research? (iii) Does the paper provide relevant data related the research topics?, (iv) How adequately is the research results documented? (v) Does the paper allow answering the research questions?. The remaining paper [S14] was kept although it was identified as an expert opinion because it is focused explicitly on the topic and allowed us to answer the second question.

3 Results

From the initial set of 448 publications (see Fig. 1), 23 studies were identified as contributing to DevOps culture. This section presents an overview of this topic according to the research questions.

3.1 What are the Attributes of DevOps Culture Confessed to?

Bearing in mind that this study is focused on culture as a human factor in SE processes and particularly DevOps, a full review of the 54 publications in the second step was done. At this stage, the findings revealed that some actually do not address — much less discuss the meaning of — culture at all. That is the reason why we attempt to identify the cultural aspect into the sections of results and discussion of each paper during the third step of this SLR (23 publications). The final classification scheme was developed after applying the process described in Sect. 2.2. Table 1 lists the attributes, Columns 1 and 2 are self-explanatory. Column 3 denotes the number of publications related to the attribute while Column 4 denotes the percentage of average weighted by attribute. Finally, Column 5 indicates the list of primary studies related to the attribute.

To summarize, 13 attributes were identified in the primary studies. 7 of the attributes are up to 75% out of the total. These attributes are: (i) Communication, (ii) Collaboration, (iii) Feedback (Continuous and immediate), (iv) Responsibility (personal/mutual), (v) Improvement cycle, (vi) Sharing Knowledge, and (vii) Transparency. However, taking into account the number of primary studies (23) it seems that there is a consensus of more than 70% of them in which DevOps culture is primarily seen as Collaboration, Communication, Feedback and Responsibility.

3.2 What Emotional Phenomenon Could Be Experienced by People in the DevOps Culture?

This is not an easy question to answer because there is not one standard emotion word hierarchy [20]. Even more, according to [21], Kleinginna et al. reported more than 90 definitions have been produced for this term, and no consensus in the literature has been reached. Therefore, from a comprehensive literature review of this topic in SE, we focus on the Parrott's emotion framework which was previously chosen to conduct an exploratory analysis of emotions in software artifacts [22]. Table 2 shows as this framework classifies human emotions into a tree structure with three levels. Each level refines the granularity of the previous level, making abstract emotions more concrete. Taking into account that structure, the statements associated to the attributes during the data extraction were read again to identify the emotions of practitioners. Furthermore, although, that structure allowed us to understand these emotions at different levels during the characterization process we chose the use of think-aloud as a strategy to enhance the ability to think critically. In this way, eventually, the answers were “*compassion*” and “*empathy*”. The first clue was found in the article “*Containers Will Not Fix Your Broken Culture (and Other Hard Truths)*” [S14] which points out that “*We have to live it [DevOps]; change for the better is a choice we make every day through our actions of listening empathetically and acting compassionately*”. However, such an idea was not clear at the beginning. During the third step of this review, the idea was growing as the same time that the empirical evidence related to the attributes was identified. At the end, we built a schema of attributes (for DevOps culture) where “*empathy*” seemed to fit well. Thus, empathy “*dissolves the barriers between self and other*” [23] as DevOps dissolves the barriers between developers and

operators. This is certainly accepted by practitioners and researchers as part of the essence of DevOps despite that the term DevOps is vaguely defined.

Table 2. Parrott’s emotion framework.

Primary emotions	Secondary emotions	Tertiary emotions
Love	Affection	Adoration, Sentimentality, Liking, Compassion, Caring, ...
	Lust	Desire, Passion, Infatuation
	Longing	
Joy	Cheerfulness	Amusement, Enjoyment, Happiness, Satisfaction, ...
	Zest	Enthusiasm, Zeal, Excitement, Thrill, Exhilaration
	Contentment	Pleasure
	Optimism	Eagerness, Hope
	Pride	Triumph
	Enthrallment	Enthrallment, Rapture
Surprise	Surprise	Amazement, Astonishment
Anger	Irritability	Aggravation, Agitation, Annoyance, Grumpy,. . .
	Exasperation	Frustration
	Rage	Outrage, Fury, Hostility, Bitter, Hatred, Dislike,. . .
	Disgust	Revulsion, Contempt, Loathing
	Envy	Jealousy
	Torment	
Sadness	Suffering	Agony, Anguish, Hurt
	Sadness	Depression, Despair, Unhappy, Grief, Melancholy,. . .
	Disappointment	Dismay, Displeasure
	Shame	Guilt, Regret, Remorse
	Neglect	Embarrassment, Humiliation, Insecurity, Insult,. . .
	Sympathy	Pity, Sympathy
Fear	Horror	Alarm, Shock, Fright, Horror, Panic, Hysteria,. . .
	Nervousness	Suspense, Uneasiness, Worry, Distress, Dread,. . .

Figure 2 depicts our schema, however, it is a little different of the list of attributes in Table 1. As one can see, the attribute “*New personnel and ideas*” was divided in order to facilitate the understanding. The two new attributes are “*Hiring*” and “*New ideas*”, the first one describes the attribute “*New personnel*” and the second one “*New ideas*” allows us to think not only in new personnel.

This scheme is consistent with the findings of a previous empirical study in software development about collective empathy [24]. According to [24], collective empathy prevents team dissolution by facilitating the development of bonds among team members, as well as creating and affirming a sense of groupness.

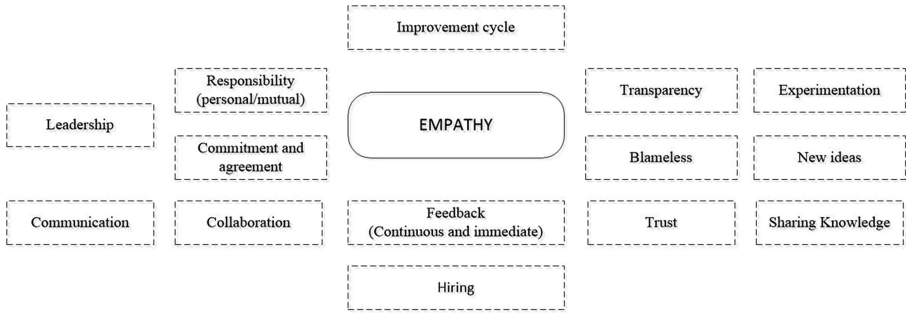


Fig. 2. Characterizing DevOps culture

3.3 What is the Trend of Studies Related to DevOps Culture in the Scientific Literature?

Figure 3 presents the number of publications over time per source. A quick look at the compiled data shows that the research field of Culture DevOps is slowly growing. Moreover, IEEE Xplore Digital Library is the source that more primary studies (13) has provided.

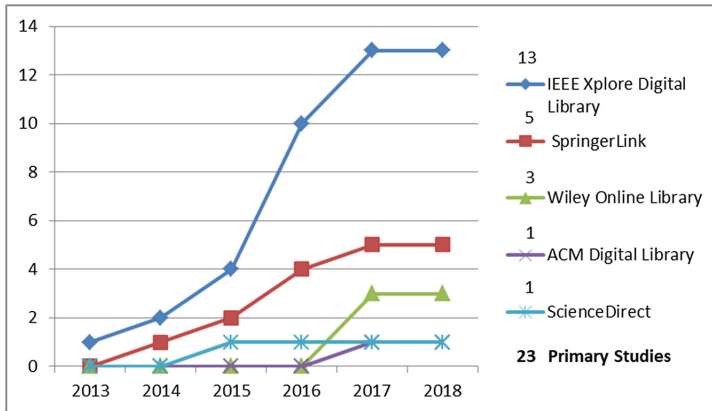


Fig. 3. Number of publications over time per source.

3.4 Limitations of Results

The limitation to academic search engines represents the state-of-the art of academic DevOps research. Therefore, future research should focus on the gap between professional research and the academic research on the topic, maybe using a multivocal literature review. The inclusion of English-only papers might mean that relevant studies in other languages are missed out, but this study is focused on the academic field and English is the most common language on this field. Another major limitation is possible

selection bias, but the protocol is a way to reduce this threat. Finally, it is worth noting that the categorization was also reviewed by another researcher in order to minimize the threat's risk of do that in a wrong way.

4 Conclusions and Future Work

In spite of the literature presents an increasingly interest on DevOps, a comprehensive systematic review about DevOps culture does not exist. Even more, the definition of DevOps remains unclear in the scientific literature despite the previous efforts, such as [3, 7], made in this direction. Therefore rather than define “DevOps culture”, we prefer to characterize it in order to understand its current status and address further research.

This review reveals that the soft side of DevOps is not always confessed among practitioners and researchers but it is always presented in software development [17]. It seems that culture is a term that everyone thinks they understand and it has become a powerful aspect of identity. In fact, culture is very related to human factors [25]. As a result of the characterization process, we identified 13 attributes. The most frequently attributes in the 23 primary studies were 7: (i) Communication, (ii) Collaboration, (iii) Feedback (Continuous and immediate), (iv) Responsibility (personal/mutual), (v) Improvement cycle, (vi) Sharing Knowledge, and (vii) Transparency. However, there is a relatively scarce number of primary studies related to this topic, although it is slowly growing in the scientific literature. Therefore, there is a need for empirical research.

Another aspect in the soft side of DevOps is the emotional phenomenon experienced by people. At the end of this review, empathy seems to be behind DevOps culture because, as already mentioned, “*dissolves the barriers between self and other*” [23] as DevOps dissolves the barriers between developers and operators. Bearing in mind that idea, we also built a scheme which is consistent with the findings of a previous empirical study in software development about collective empathy [24]. However, further research is needed in order to validate and enhance the schema and study the phenomenon itself.

Appendix A: Primary Studies

- [S01] de Bayser, M., Azevedo, L. G., & Cerqueira, R. (2015). ResearchOps: The case for DevOps in scientific applications. In *Integrated Network Management (IM), 2015 IFIP/IEEE International Symposium on* (pp. 1398–1404). IEEE.
- [S02] Ebert, C., Gallardo, G., Hernantes, J., & Serrano, N. (2016). DevOps. *IEEE Software*, 33(3), 94–100. <https://doi.org/10.1109/MS.2016.68>
- [S03] Soni, M. (2015). End to end automation on cloud with build pipeline: the case for DevOps in insurance industry, continuous integration, continuous testing, and continuous delivery. In *Cloud Computing in Emerging Markets (CCEM), 2015 IEEE International Conference on* (pp. 85–89). IEEE.

- [S04] Lwakatare, L. E., Kuvaja, P., & Oivo, M. (2015). Dimensions of devops. In *International Conference on Agile Software Development* (pp. 212–217). Springer.
- [S05] Farroha, B. S., & Farroha, D. L. (2014). A framework for managing mission needs, compliance, and trust in the DevOps environment. In *Military Communications Conference (MILCOM), 2014 IEEE* (pp. 288–293). IEEE.
- [S06] Kamuto, M. B., & Langerman, J. J. (2017). Factors inhibiting the adoption of DevOps in large organisations: South African context. In *Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2017 2nd IEEE International Conference on* (pp. 48–51). IEEE.
- [S07] Feitelson, D. G., Frachtenberg, E., & Beck, K. L. (2013). Development and deployment at facebook. *IEEE Internet Computing*, 17(4), 8–17.
- [S08] Furfaro, A., Gallo, T., Garro, A., Sacca, D., & Tundis, A. (2016). ResDevOps: a software engineering framework for achieving long-lasting complex systems. In *Requirements Engineering Conference (RE), 2016 IEEE 24th International* (pp. 246–255). IEEE.
- [S09] Punjabi, R., & Bajaj, R. (2016). User stories to user reality: A DevOps approach for the cloud. In *Recent Trends in Electronics, Information & Communication Technology (RTEICT), IEEE International Conference on* (pp. 658–662). IEEE.
- [S10] Diel, E., Marczak, S., & Cruzes, D. S. (2016). Communication Challenges and Strategies in Distributed DevOps. In *Global Software Engineering (ICGSE), 2016 IEEE 11th International Conference on* (pp. 24–28). IEEE.
- [S11] Camacho, C. R., Marczak, S., & Cruzes, D. S. (2016). Agile team members perceptions on non-functional testing: influencing factors from an empirical study. In *Availability, Reliability and Security (ARES), 2016 11th International Conference on* (pp. 582–589). IEEE.
- [S12] Perera, P., Silva, R., & Perera, I. (2017). Improve software quality through practicing DevOps. In *Advances in ICT for Emerging Regions (ICTer), 2017 Seventeenth International Conference on* (pp. 1–6). IEEE.
- [S13] Hussain, W., Clear, T., & MacDonell, S. (2017). Emerging trends for global DevOps: a New Zealand perspective. In *Proceedings of the 12th International Conference on Global Software Engineering* (pp. 21–30). IEEE Press.
- [S14] Kromhout, B. (2017). Containers Will Not Fix Your Broken Culture (and Other Hard Truths). *Queue*, 15(6), 50:46–50:56. <https://doi.org/10.1145/3178368.3185224>
- [S15] Gupta, V., Kapur, P. K., & Kumar, D. (2017). Modeling and measuring attributes influencing DevOps implementation in an enterprise using structural equation modeling. *Information and Software Technology*, 92, 75–91.
- [S16] Colomo-Palacios, R., Fernandes, E., Soto-Acosta, P., & Larrucea, X. (2017). A case analysis of enabling continuous software deployment through knowledge management. *International Journal of Information Management*.
- [S17] Hussaini, S. W. (2015). A Systemic Approach to Re-inforce Development and Operations Functions in Delivering an Organizational Program. *Procedia Computer Science*, 61, 261–266. <https://doi.org/10.1016/j.procs.2015.09.209>

- [S18] Erich, F. M. A., Amrit, C., & Daneva, M. (2017). A qualitative study of DevOps usage in practice. *Journal of Software: Evolution and Process*, 29(6).
- [S19] Riungu-Kalliosaari, L., Mäkinen, S., Lwakatare, L. E., Tiihonen, J., & Männistö, T. (2016). DevOps adoption benefits and challenges in practice: a case study. In *International Conference on Product-Focused Software Process Improvement* (pp. 590–597). Springer.
- [S20] Lwakatare, L. E., Karvonen, T., Sauvola, T., Kuvaja, P., Olsson, H. H., Bosch, J., & Oivo, M. (2016). Towards DevOps in the embedded systems domain: Why is it so hard? In *System Sciences (HICSS), 2016 49th Hawaii International Conference on* (pp. 5437–5446). IEEE.
- [S21] Nybom, K., Smeds, J., & Porres, I. (2016). On the Impact of Mixing Responsibilities Between Devs and Ops. In *International Conference on Agile Software Development* (pp. 131–143). Springer.
- [S22] Erich, F., Amrit, C., & Daneva, M. (2014). A mapping study on cooperation between information system development and operations. In *International Conference on Product-Focused Software Process Improvement* (pp. 277–280). Springer.
- [S23] Myrbakken, H. avarð, & Colomo-Palacios, R. (2017). DevSecOps: A Multivocal Literature Review. In *International Conference on Software Process Improvement and Capability Determination* (pp. 17–29). Springer

References

1. Rodríguez, P., et al.: Continuous deployment of software intensive products and services: a systematic mapping study. *J. Syst. Softw.* **123**, 263–291 (2017)
2. Ameller, D., Farré, C., Franch, X., Valerio, D., Cassarino, A.: Towards continuous software release planning. In: 2017 IEEE 24th International Conference on Software Analysis, Evolution and Reengineering (SANER), pp. 402–406 (2017)
3. Stahl, D., Martensson, T., Bosch, J.: Continuous practices and DevOps: beyond the buzz, what does it all mean?. In: 2017 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA), pp. 440–448. IEEE (2017)
4. Bosch, J.: Continuous software engineering: an introduction. In: Bosch, J. (ed.) *Continuous Software Engineering*, pp. 3–13. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11283-1_1
5. Fitzgerald, B., Stol, K.-J.: Continuous software engineering: a roadmap and agenda. *J. Syst. Softw.* **123**, 176–189 (2017)
6. Lwakatare, L.E., Kuvaja, P., Oivo, M.: Relationship of DevOps to agile, lean and continuous deployment. In: Abrahamsson, P., Jedlitschka, A., Nguyen Duc, A., Felderer, M., Amasaki, S., Mikkonen, T. (eds.) *PROFES 2016. LNCS*, vol. 10027, pp. 399–415. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-49094-6_27
7. Jabbari, R., bin Ali, N., Petersen, K., Tanveer, B.: What is DevOps?: a systematic mapping study on definitions and practices. In: *Proceedings of the Scientific Workshop Proceedings of XP2016*, p. 12. ACM (2016)
8. Ebert, C., Gallardo, G., Hernantes, J., Serrano, N.: DevOps. *IEEE Softw.* **33**, 94–100 (2016)

9. Balalaie, A., Heydarnoori, A., Jamshidi, P.: Microservices architecture enables DevOps: migration to a cloud-native architecture. *IEEE Softw.* **33**, 42–52 (2016)
10. Humble, J., Molesky, J.: Why enterprises must adopt DevOps to enable continuous delivery. *Cut. IT J.* **24**, 6–12 (2011)
11. Debois, P.: DevOps Areas - Codifying DevOps practices. <http://www.jedi.be/blog/2012/05/12/codifying-devops-area-practices/>
12. Mueller, E.: What Is DevOps? (2010). <https://theagileadmin.com/what-is-devops/>
13. Capretz, L.F.: Bringing the human factor to software engineering. *IEEE Softw.* **31**, 104 (2014)
14. Colomo-Palacios, R., Gomez-Berbis, J.M., Garcia-Crespo, A., Puebla-Sanchez, I.: Social Global Repository: using semantics and social web in software projects. *Int. J. Knowl. Learn.* **4**, 452–464 (2008)
15. Colomo-Palacios, R., Casado-Lumbreras, C., Soto-Acosta, P., Misra, S., García-Peñalvo, F. J.: Analyzing human resource management practices within the GSD context. *J. Glob. Inf. Technol. Manag.* **15**, 30–54 (2012)
16. Kromhout, B.: Containers will not fix your broken culture (and other hard truths). *Queue* **15**, 50:46–50:56 (2017)
17. Capretz, L.F., Ahmed, F., da Silva, F.Q.B.: Soft sides of software. *Inf. Softw. Technol.* **92**, 92–94 (2017)
18. 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.* **40**, 186–189 (2018)
19. Kitchenham, B., Charters, S.: Guidelines for performing Systematic Literature Reviews in Software Engineering. School of Computer Science and Mathematics, Keele University, Keele (2007)
20. Sailunaz, K., Dhaliwal, M., Rokne, J., Alhaji, R.: Emotion detection from text and speech: a survey. *Soc. Netw. Anal. Min.* **8**(1), 28 (2018)
21. Graziotin, D., Wang, X., Abrahamsson, P.: Understanding the affect of developers: theoretical background and guidelines for psychoempirical software engineering. In: Presented at the 7th International Workshop on Social Software Engineering, pp. 25–32. ACM (2015)
22. Murgia, A., Tourani, P., Adams, B., Ortu, M.: Do developers feel emotions? An exploratory analysis of emotions in software artifacts. In: Proceedings of the 11th Working Conference on Mining Software Repositories, pp. 262–271. ACM, New York (2014)
23. Pavlovich, K., Krahnke, K.: Empathy, connectedness and organisation. *J. Bus. Ethics* **105**, 131–137 (2012)
24. Akgün, A.E., Keskin, H., Cebecioglu, A.Y., Dogan, D.: Antecedents and consequences of collective empathy in software development project teams. *Inf. Manag.* **52**, 247–259 (2015)
25. Sánchez-Gordón, M.-L.: Getting the best out of people in small software companies: ISO/IEC 29110 and ISO 10018 standards. *Int. J. Inf. Technol. Syst. Approach* **10**, 45–60 (2017)