Empirical Software Engineering (2020) 25:627–677
https://doi.org/10.1007/s10664-019-09763-0

Search. Review. Repeat? An empirical study of threats to replicating SLR searches

Jacob Krüger[1,2] · **Christian Lausberger**[1,3] · **Ivonne von Nostitz-Wallwitz**[1,3] · **Gunter Saake**[1] · **Thomas Leich**[2,3]

Published online: 30 August 2019

Abstract
A systematic literature review (SLR) is an empirical method used to provide an overview of existing knowledge and to aggregate evidence within a domain. For computer science, several threats to the completeness of such reviews have been identified, leading to recommendations and guidelines on how to improve their quality. However, few studies address to what extent researchers can replicate an SLR. To conduct a replication, researchers have to first understand how the set of primary studies has been identified in the original study, and can ideally retrieve the same set when following the reported protocol. In this article, we focus on this initial step of a replication and report a two-fold empirical study: Initially, we performed a tertiary study using a sample of SLRs in computer science and identified what information that is needed to replicate the searches is reported. Based on the results, we conducted a descriptive, multi-case study on digital libraries to investigate to what extent these allow replications. The results reveal two threats to replications of SLRs: First, while researchers have improved the quality of their reports, relevant details are still missing—we refer to a *reporting threat*. Second, we found that some digital libraries are inconsistent in their query results—we refer to a *searching threat*. While researchers conducting a review can only overcome the first threat and the second may not be an issue for all kinds of replications, researchers should be aware of both threats when conducting, reviewing, and building on SLRs.

Keywords Tertiary study · Systematic literature review · Software engineering · Threats to validity · Replication · Digital library

Communicated by: Burak Turhan

Ivonne von Nostitz-Wallwitz has previously published as Ivonne Schröter.

✉ Jacob Krüger
jkrueger@ovgu.de

Extended author information available on the last page of the article.

1 Introduction

In the medical domain, SLRs are an established method to consolidate knowledge of existing empirical studies before conducting a new one (Webster and Watson 2002; Babar and Zhang 2009; Sackett et al. 1997). Over time, other domains, such as social sciences, criminology, and software engineering, have adopted the use of SLRs as a methodology and derived corresponding guidelines (Dybå et al. 2005; Babar and Zhang 2009; Kitchenham et al. 2009, 2007; Webster and Watson 2002). SLRs are a suitable way to identify, compare, and summarize the findings of studies that address a specific problem. By following defined guidelines, researchers can systematically collect papers on such a problem, consolidate the provided information, and identify research opportunities (Kitchenham and Charters 2007; MacDonell et al. 2010; Boell and Cecez-Kecmanovic 2015; vom Brocke et al. 2015).

SLRs as an empirical research method have gained growing interest in the software-engineering research community, indicated by an increasing number of such papers being published (Kitchenham et al. 2009, 2010; da Silva et al. 2011; Zhang and Babar 2013; Budgen et al. 2018a). An SLR aims to provide an unbiased analysis and follows a well-defined methodology (Kitchenham and Charters 2007, 2016) that can be replicated by other researchers and, in an ideal world, would result in the same outcome if no parameters are changed. For instance, Kitchenham et al. (2011) explicitly highlight repeatability as an anticipated benefit of an SLR, but also find that this is hardly achievable. While all steps of an SLR are equally important, the first step of searching relevant papers arguably has significant impact on the outcome, for example, due to important papers being missed (Badampudi et al. 2015; Jalali and Wohlin 2012; Felizardo et al. 2016). A literal replication or repeat[1] (Gómez et al. 2014) of a search may be neither desirable nor possible (Lindsay and Ehrenberg 1993; Riedl 2007; Kitchenham et al. 2011, 2016), for example, considering limitations of digital libraries (Kitchenham and Charters 2007; Shakeel et al. 2018). However, independently of the actually applied replication strategy (Riedl 2007; Gómez et al. 2014), it is important to have the necessary information of the original setup (e.g., search strings) and be aware of technical limitations. In this context, we know of only few studies that are concerned with analyzing threats to searches and their repeatability. Such knowledge is necessary for researchers conducting, reviewing, reading, replicating, and extending an SLR to assess potential threats to the original study, replications, extensions, and comparisons between all of these, with comparisons being recommended for experimental replications (Carver 2010).

Research Goal In this article, we investigate the initial step of searching primary studies for an SLR in computer science, focusing on two research questions:

RQ_1 *Do researchers report information needed to replicate the search of an SLR?*
To replicate an SLR, several pieces of information are necessary. With this research question, we aimed to provide an overview of what information is described in SLRs that allows replicating the initial search process, for example, guidelines, search strategies, and search strings. We report details on the study design in Section 4 and discuss the results in Section 5.

[1] https://www.acm.org/publications/policies/artifact-review-badging

RQ_2 *To what extent can we replicate searches on literature resources?*

The results of our study show that automatic searches in digital libraries are the most common search strategy. Consequently, they are an essential factor when replicating an SLR. We conducted a descriptive multiple-case study (Yin 2018) to assess the extent to which researchers can replicate automatic searches and report the details in Section 6.

By answering these research questions, we empirically analyzed to what extent researchers can replicate the search processes of SLRs. To this end, we applied a two-step methodology: First, we performed an SLR on other SLRs—a *tertiary study* (Kitchenham et al. 2016)—to assess what information researchers report. Second, based on the results of our tertiary study, we designed and conducted a descriptive multiple-case study on digital libraries to evaluate the possibility to replicate automatic searches in these.

Overall, we contribute the following with this article: We provide an overview on information that researchers report in SLRs, namely the guidelines, search strategies, literature resources, and search strings that were used, as well as problems the researchers faced during the conduct. The results indicate multiple threats in reporting SLRs that need to be addressed. While other studies already show that literature resources in computer science do not support the SLR process sufficiently (Zhang and Babar 2010; Kitchenham and Brereton 2013; Shakeel et al. 2018), they do not investigate the consistency of search results. Thus, we analyzed whether we could replicate SLRs that relied on automatic searches, with our results indicating that this seems hardly possible.

Outline The remaining article is structured as follows. In Section 2, we introduce the general concepts that are necessary to understand this article. Within Section 3, we provide an overview of related work, putting our own work into context and providing additional motivation. Afterwards, we report the design and conduct of our tertiary study within Sections 4 and 5, respectively. Based on the results we obtained in this tertiary study, we conducted a descriptive multiple-case study to analyze literature resources, for which we report the details in Section 6. We summarize our findings in Section 7 to conclude this article.

2 Background

In this section, we introduce the concepts that are important to understand this article. To this end, we describe the purpose of an SLR and its phases, different search strategies, and literature resources.

2.1 Systematic Literature Reviews

A SLR is an empirical method to systematically identify papers related to a certain research question to evaluate, consolidate, and interpret the existing evidence (Kitchenham and Charters 2007, 2016). The investigated papers are referred to as primary studies, while the SLR itself is a secondary study. Independent of the domain, SLRs are helpful to identify the state of the art on a research topic and to reveal opportunities for further investigations (Webster and Watson 2002; Boell and Cecez-Kecmanovic 2015; vom Brocke et al. 2015).

In contrast to narrative reviews that summarize the experiences and insights on a topic from an expert's perspective (Collins and Fauser 2005; vom Brocke et al. 2015), SLRs aim to be more systematic to avoid biases (Booth et al. 2012) that may occur, for example, due to identical author risk (Jalali and Wohlin 2012) or "cherry-picking" (Kitchenham et al. 2016). Consequently, SLRs require the researchers to Kitchenham and Charters (2007, 2016):

– Construct a review protocol before the conduct;
– Define a search strategy to identify papers;
– Explain inclusion and exclusion criteria to select papers;
– Define quality criteria to evaluate the papers; and
– Describe the review and evaluation process within the protocol.

Overall, the goal of an SLR is to aggregate existing evidence on a research topic, while mitigating threats of deriving biased results. In particular, the structured protocol enables replications and allows verification of the results by other researchers.

Still, conducting an SLR is a demanding task that poses difficulties to novices and experienced researchers alike (Kitchenham and Charters 2007; Biolchini et al. 2005; Dybå and Dingsøyr 2008; Riaz et al. 2010). To facilitate the conduct of SLRs, guidelines have been proposed to help researchers with defined phases and steps. The guidelines by (Kitchenham and Charters 2007) and their updated version (Kitchenham et al. 2016) are widely adopted for software engineering and many SLRs have been based on them (Kitchenham et al. 2009, 2010), wherefore we also rely on these guidelines. An SLR comprises three phases, as we depict in Fig. 1.

Planning Before starting an SLR, the researchers have to identify its necessity, mainly investigating that the considered topic has not been reviewed recently and that such a review is needed. Then, the planning phase starts with defining the research questions to scope the extent of the SLR. Based on these questions, the conducting researchers write a review protocol, addressing: motivation, research questions, search strategy, selection criteria, quality criteria, data extraction, information consolidation, threats to validity, type of the review, and additional management activities. As a result, they create a detailed plan for the SLR and, in a final step, evaluate it.

Conducting After the planning phase, the researchers execute the steps they defined in the protocol. Thus, they identify relevant papers based on the search strategy and assess these papers for relevance by applying selection and quality criteria. From the resulting relevant papers, the researchers extract and consolidate all important information. As a result, they collect the information that is necessary to answer the previously defined research questions.

Reporting Finally, the conducting researchers use the consolidated information to answer the research questions. To this end, they need to focus on the contribution of the SLR to the investigated domain, new findings, and have to reason about the validity of the results (Shaw 2003). To document the outcome, the researchers write a protocol that comprises the answers to these aspects and the research questions. Finally, the researchers

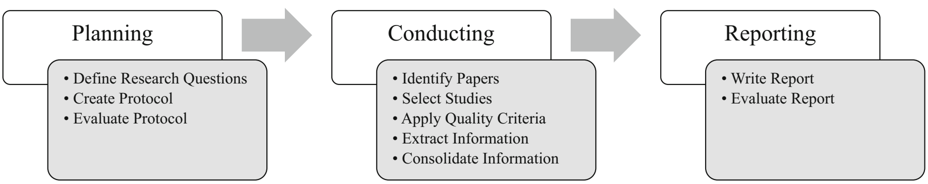


Fig. 1 Phases of an SLR according to Kitchenham et al. (2016)

evaluate the report to ensure that it fulfills all requirements and that they performed the SLR as defined in the planning phase.

2.2 Search Strategies

Identifying relevant papers is an essential step in any SLR. Researchers define this step in the search strategy, which they have to adapt according to the goal of the SLR (vom Brocke et al. 2015; MacDonell et al. 2010). In this article, we differentiate between three core strategies that researchers can use independently or in combination (Kitchenham et al. 2016):

- *Automatic Search*
  For an automatic search, researchers define a search query and execute it on a digital library or database. The advantage of this search strategy is its speed and limited effort to identify a large number of papers. However, automatic searches also include large numbers of irrelevant papers that must be manually filtered (Kitchenham and Charters 2007). In addition, the quality of the results depends on several factors, such as the libraries and databases that the researchers use, the search query and its adaptations, the existence of synonyms, as well as duplicates (Zhang et al. 2011; Jalali and Wohlin 2012; Shakeel et al. 2018).
- *Manual Search*
  A manual search is limited to defined venues, for example, a set of Journals or Conferences. Consequently, researchers manually collect and evaluate each paper from the selected venues to identify whether it is relevant for the SLR. Such a manual search requires less effort than an automatic one, if a limited and appropriate set of venues is selected. Still, each paper must be manually evaluated and inconsistent usage of keywords, poor quality of abstracts, and synonyms can pose problems (Brereton et al. 2007).
- *Snowballing*
  For snowballing, an initial set of relevant papers must exist. Then, researchers can analyze the papers' references (backwards-snowballing) or their citing papers (forwards-snowballing) to identify further papers (Jalali and Wohlin 2012; Wohlin 2014). This process can be performed incrementally, continuing with the newly identified papers of each iteration. Essential for this search method is the initial set of papers that should be based on different communities, publishers, years, authors, and not too small (Wohlin 2014). The advantage of this method is its simplicity, but depending on the number of referencing and citing papers, it can become an extensive analysis compared to an automatic search (Badampudi et al. 2015).

Researchers can use each of these search strategies to identify relevant papers for an SLR. Due to the limitations and disadvantages of each method, several authors and guidelines recommend to combine different search strategies (Zhang and Babar 2010; Jalali and Wohlin 2012; Kitchenham et al. 2016).

2.3 Literature Resources

In the previous section, we referred to digital libraries and literature databases. For simplicity, we summarize these terms and any other collection that provides long-term access to scientific literature and that can be searched for an SLR as *literature resources* in the remaining article. While such literature resources are essential, they have several shortcomings that hamper their usability for SLRs. This includes, for example, different search models and query syntaxes (Brereton et al. 2007; Babar and Zhang 2009; Imtiaz et al. 2013; Falagas et al. 2008; Shakeel et al. 2018), limitations for complex queries (MacDonell et al. 2010; Falagas et al. 2008), varying publisher coverage (Falagas et al. 2008), missing standard keywords (Kitchenham and Brereton 2013), and permanent development, including sudden expiration (Giles 2006; Orduña-Malea et al. 2014; Harzing and Alakangas 2017). In particular, all these issues are rarely communicated or documented, and thus researchers are seldom aware of these issues until they face those (Orduña-Malea et al. 2014). As such problems are known, researchers that conduct an SLR can address them. In contrast, only the providers of literature resources can influence whether these behave consistently. Still, inconsistency is a potential threat that may influence automatic searches for an SLR, which is why we conducted an empirical study on this issue.

3 Motivation and Related Work

The motivation for our study was to investigate the ability of researchers to replicate the search of an SLR. We focused on this first step of conducting an SLR, as it scopes the set of papers, and thus the available evidence. In this section, we discuss related work, highlight the differences to our study, and provide additional motivation.

Kitchenham et al. (2011) highlight that achieving repeatability may not be the goal of an SLR in software engineering, but is an expected characteristic. Similarly, Booth et al. (2012) characterize that an SLR is reproducible, in contrast to other forms of reviews. While literal replications (Yin 2018) are not always intended and are hardly possible, it is still important to understand threats that hamper replications—independent of their purpose and design, for example, literal replications or extensions. Such threats are connected to the reporting of SLRs or technical limitations, as we highlight based on the following related papers.

Jalali and Wohlin (2012) compare an automatic search to backwards-snowballing, both with one iteration. They identified 26 and 15 unique papers, respectively, as well as 27 common papers. While they could derive similar patterns from both searches, each search individually missed some important papers and would provide an incomplete overview, among other threats to each search strategy. Similarly, Badampudi et al. (2015) compare those two search strategies in the context of systematic mapping studies. They identified 45.9% of the relevant papers using an automatic search, while snowballing revealed 83%. The authors also derived conclusions based on the identified papers and found that the overlap was quite small (5 out of 15 compared conclusions). More importantly, only snowballing revealed conflicting primary studies and seemed to be more reliable (14 out of 15 compared conclusions). Felizardo et al. (2016) applied forwards-snowballing and an automatic search to update an existing SLR. Contrary to the two previous studies, they found that snowballing may reduce the effort, but may miss important papers. Thus, it seems unclear, which search strategy is more useful for SLRs. Finally, comparing the same procedure of combining automatic search and snowballing with two different teams, MacDonell et al. (2010) conclude that SLRs are robust and repeatable, as both teams derived the same findings. These works highlight the impact of missing papers in the search process and the importance of ensuring its consistency.

Considering reporting, some tertiary studies indicate that, despite an increase in quality, important details in reports of SLRs are still missing, preventing full understanding (Kitchenham et al. 2010; Budgen et al. 2018a). These studies already indicate the importance of knowing potential threats to the initial search and its reporting in SLRs to ensure reliable evidence, but also to provide details for scoping replications and extensions. A rather different example is the tertiary study conducted by Budgen et al. (2018b). The authors focus on the importance of industrial case studies for SLRs and highlight that, in order to improve the relevance for practitioners, different reporting strategies are required, such as a specific summary of findings (von Nostitz-Wallwitz et al. 2018).

Other tertiary studies investigate the general development of SLRs in software engineering and are related to our work (Kitchenham et al. 2009, 2010, 2013; da Silva et al. 2011). However, these works are more concerned with statistics about publication venues, authors, and the general quality of the provided information. Some authors, such as Budgen and Brereton (2006), Brereton et al. (2007), Dybå et al. (2007), Babar and Zhang (2009), MacDonell et al. (2010), Imtiaz et al. (2013), and Shakeel et al. (2018), report their experiences, conduct tertiary studies, or interview experts to analyze the application and problems of SLRs. For example, these studies identify some issues of the applied search strategies, search models, APIs, and adaptations of search strings.

As we described, there have been some studies on the repeatability, search strategies, and reporting of SLRs. While some of these works have an overlap with our research questions, none of them is focusing on the details that researchers report on the search processes or the consistency of literature resources. The existing studies show that different searches can have an impact on the results researchers derive from an SLR. Thus, this initial step is important to analyze further, helping us to understand threats, especially to their replication.

4 Tertiary Study: Design

As first step of our study, we conducted an SLR, aiming to answer our first research question and scoping our case studies for the second research question. In Fig. 2, we depict the review protocol process of our SLR that we use to report the steps we performed during the conduct (cf. Fig. 1). In this section, we describe each of these steps in detail. At

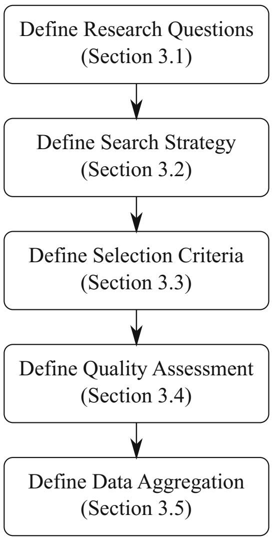


Fig. 2 Review protocol content for SLRs based on Brereton et al. (2007)

the beginning, we refined our first research question to scope our study and describe the resulting sub-questions in Section 4.1. Afterwards, we defined a search strategy to identify suitable papers, as we report in Section 4.2. In Sections 4.3 and 4.4, we describe our selection and quality criteria, respectively. Finally, we present how and what data we extracted and aggregated in Section 4.5.

4.1 Research Questions

The purpose of our review was to investigate whether researchers report the pieces of information that others need to replicate the search of an SLR. Consequently, we performed a secondary study on secondary studies, which Kitchenham et al. (2016) refer to as *tertiary study*. To scope our tertiary study, we refined our first research question, focusing particularly on details about the search. We defined five sub-questions based on previous works and our own experiences (cf. Section 3):

$RQ_{1.1}$ *What guidelines do researchers use to conduct SLRs?*

Several authors have proposed guidelines for conducting SLRs in various domains, for example, Webster and Watson (2002) in information systems or Kitchenham and Charters (2007) in software engineering. With this research question, we aimed to identify what guidelines researchers follow and whether one is dominating in computer science. Conducting an SLR without any guidelines or based on varying (e.g., versions or variants) guidelines may hamper researchers' ability to understand and replicate the process, for example, because information and even phases differ or were added only in later versions (Dresch et al. 2014; Kitchenham et al. 2016).

$RQ_{1.2}$ *What search strategies do researchers employ?*

As we described in Section 2.2, three main strategies and their combinations exist to search for relevant primary studies: *automatic search*, *manual search*, and *snowballing*—each strategy having different pros and cons. Consequently, it is essential that researchers specify the search strategy that was employed in order to support replications. We identified how often researchers have employed which search strategy and in which combinations.

$RQ_{1.3}$ *What literature resources do researchers use to search for relevant papers?*

In addition to the employed search strategies, we were also interested in the literature resources used. As such resources develop over time (Fuhr et al. 2007; Orduña-Malea et al. 2014; Harzing and Alakangas 2017) and are seen as differently important for SLRs (Brereton et al. 2007), we aimed to identify what resources have been commonly considered. This provides an overview for researchers and is—in combination with our descriptive multi-case study—a means to assess validity threats of searches that are performed for SLRs.

$RQ_{1.4}$ *What information on search strings do researchers provide?*

The search string is the essential input for any automatic search. We investigated which details researchers report on the used search string, namely whether they provide a *search query*, only *keywords*, or even *no information* at all. This granularity indicates whether an existing SLR can be replicated, especially if it remains unclear how the researchers constructed a search string or adapted it for different literature resources. We investigated adaptations of search strings within this and the next sub-question.

RQ$_{1.5}$ *What problems do researchers report?*

Finally, we were also interested in problems other researchers experienced while conducting SLRs. We read each paper and gathered reported experiences to summarize challenges and pitfalls. This can help researchers when conducting SLRs, replications or extensions, and indicates research opportunities for future work.

Overall, answering these research questions has four purposes. First, we provide an overview on details reported on the search processes of SLRs that are necessary to replicate these. Second, we show what threats to validity may exist, due to missing information. Third, we derive recommendations for assessing and conducting especially automatic searches for SLRs to ensure that researchers can replicate them and that they are consistent. Fourth, we used the results to scope our descriptive multiple-case study for our second research question (cf. Section 6).

4.2 Search Strategy

To answer our research questions, we conducted an automatic search in Scopus.[2] Scopus indexes papers of different publishers, for instance, ACM, IEEE, Springer, and Elsevier, that are previously assessed by an independent committee. As a result, all papers should have a certain quality, as at least two reviews have been performed: The papers have been reviewed to be accepted at a venue and the venue itself to be included in Scopus. We applied the following search string.

```
TITLE-ABS-KEY(''systematic literature review'' OR
''tertiary study'') AND
(LIMIT-TO(SUBJAREA, ''COMP'') ) AND
(LIMIT-TO(PUBYEAR, 2016) OR LIMIT-TO(PUBYEAR, 2015))
```

Thus, we restricted our results to papers that explicitly mention an SLR or tertiary study in their title, abstract, or keywords. In addition, the papers had to be published and be available in Scopus from 2015 to June 2016 (we conducted our query in July 2016), and address the domain of computer science. We remark that we considered not only software engineering, but computer science as a whole, to provide a broader overview and avoid disagreement on what areas are still part of software engineering research. To avoid this problem, we followed the classification of Scopus that, however, may also not be ideal. Nonetheless, our results mainly comprise papers that are based on the guidelines by Kitchenham and Charters (2007), indicating that software engineering is the major contributor of SLRs in our set of papers.

4.3 Selection Criteria

We excluded any paper that did not fulfill all of the following inclusion criteria:

- The paper is written in English;
- The publication year is 2015 or 2016;
- The paper addresses the domain of computer science; and
- The paper describes the conduct of an SLR.

[2] https://www.scopus.com/home.uri

We focused on the period preceding the date of our search to consider current developments and limited the period to one and a half year, as several studies show an increasing number of SLRs being published in software engineering (Babar and Zhang 2009; Kitchenham et al. 2009, 2010; da Silva et al. 2011; Budgen et al. 2018a). As we analyzed the applied search strategies—not a specific research question—of an SLR, we expected a large number of relevant papers even for this short period. Moreover, we excluded irrelevant papers, especially if they only discuss the results of an SLR, but do not report how the conduct was performed.

4.4 Quality Assessment

Scopus only indexes papers from venues that have been examined by an independent committee. Due to the two review phases for each paper, we assume that all of them have a certain quality. Identical to other researchers (Kitchenham and Brereton 2013), we did not exclude papers based on quality criteria—wherefore we did not define such criteria. We did this because our research questions address the quality of the papers. Thus, excluding them in advance would bias the results, which reflect quality criteria to a certain extent.

4.5 Data Aggregation

We added each paper into a reference management tool, namely *Zotero*.[3] With this tool, we automatically collected the following standard information: *title*, *authors*, *publication year*, and *venue*. To answer our research questions, we first read each abstract to exclude papers not fulfilling our exclusion criteria and, if necessary, each paper completely. In accordance with proposals of other researchers (Brereton et al. 2007; Kitchenham et al. 2009), one of the authors then extracted the relevant data, while the others reviewed the results. These reviews were based on two of the other authors verifying random samples of papers and all authors checking the completeness of the extracted data. We resolved conflicts by first checking the corresponding papers again and, in few cases, had to decide by majority vote. Overall, we extracted the following data:

– The guidelines used for the SLR if the researchers explicitly report to follow and cite them.
– The search strategy (i.e., automatic, manual, snowballing) the researchers employed.
– The literature resources in which the researchers searched for relevant papers.
– The information on search strings (i.e., search string, keywords, none).
– The problems the researchers report about conducting the SLR.

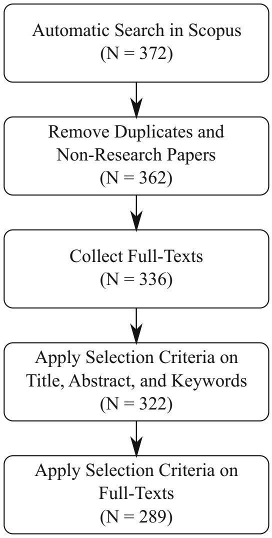
This information allows us to answer our first research question and its sub-questions. To store the corresponding data, we first added tags in the Zotero database. For our analysis, we automatically migrated the data into an SQL database and used queries to consolidate the information for each research question.

5 Tertiary Study: Conduct and Results

In this section, we first describe the conduct of our tertiary study (cf. Fig. 1). We then present and discuss the results we obtained for each sub-question, describe threats to the validity of our SLR, and finally answer our first research question to summarize the results.

[3] https://www.zotero.org/

Fig. 3 Conduct phase of our tertiary study



5.1 Conduct

Based on the protocol we described in the previous section, we executed our automatic search on July 5th 2016. We illustrate the steps of our conduct in Fig. 3. As we show, our automatic search on Scopus resulted in 372 papers that matched our search string. In the next step, we imported all papers into Zotero, removing duplicates and non-research literature, such as, tables of contents and proceeding introductions. For this reason, we removed 10 papers from our initial set. Then, we downloaded all full-text versions we had access to, which was not the case for 26 of the papers. Afterwards, we read the title, abstract, and keywords of each paper and excluded 14 that did not match our defined inclusion criteria (cf. Section 4.3). Finally, we applied our inclusion criteria on the full texts of all remaining 322 papers, of which we excluded 33. Consequently, we identified 289 papers to be relevant for our study, spanning one and a half year of research based on SLRs. We provide a list of all included papers in the Appendix of this article.

5.2 $RQ_{1.1}$: What Guidelines do Researchers Use to Conduct SLRs?

Measurement To answer this research question, we read each paper in detail and identified whether the researchers followed a guideline. We considered this to be true, if both of the following conditions applied:

- The researchers explicitly write that they follow a certain guideline.
- The used guideline is cited in the paper.

As some guidelines have been extended, updated or published in different versions (e.g., by Kitchenham and Charters 2007, 2016), we condensed references to such works of the same (or most) authors to the version that has been published first. In addition, in four papers the researchers state to apply more than one guideline, wherefore we identified more usages of guidelines than investigated papers. Consequently, the numbers in Table 1 add up to more than 100%.

Table 1 Guideline used and referred to in the analyzed SLRs

ID	Guidelines	Usage		Domain
		Total	Relative	
1	Kitchenham and Charters (2007)	192	66.4%	Computer Science
2	Webster and Watson (2002)	9	3.1%	Computer Science
3	Tranfield et al. (2003)	9	3.1%	Economics
4	Biolchini et al. (2005)	4	1.4%	Computer Science
5	Khan et al. (2003)	4	1.4%	Medicine
6	Moher et al. (2009)	3	1.0%	Medicine
7	Okoli and Schabram (2010)	3	1.0%	Computer Science
8	Booth et al. (2012)	2	0.7%	Medicine
9	Petersen et al. (2008)	2	0.7%	Computer Science
10	Levy and Ellis (2006)	1	0.3%	Computer Science
11	Soni and Kodali (2011)	1	0.3%	Economics
12	Galvan and Galvan (2016)	1	0.3%	Social Sciences
13	Wolfswinkel et al. (2013)	1	0.3%	Computer Science
14	Bandara et al. (2011)	1	0.3%	Computer Science
	None	60	20.8%	

Results We provide an overview of all guidelines that we identified in Table 1, comprising the cited or earliest reference, the number and ratio of papers referring to each guideline, and the domain of the guideline. We remark that the domains reflect broader research areas. Thus, instead of assigning (Kitchenham and Charters 2007) to software engineering and Webster and Watson (2002) to information systems, we assigned both to the more general domain of computer science. In our set of papers, researchers most commonly used the guidelines by Kitchenham and Charters (2007) with approximately 66.4% of the analyzed papers referring to these, highlighting the dominance of software engineering in our analysis. Overall, 20.8% of the papers do not mention any guidelines. 14.2% (41) of the papers follow at least one of the other 13 guidelines we identified. In most cases, the researchers applied guidelines that are proposed or adopted for computer science, while some rely on such from economics, medicine, and social sciences.

Discussion In Table 1, we can see that the guidelines proposed by Kitchenham and Charters (2007) are dominating the structure of SLRs in computer science. This is not surprising, as they have been the first to adopt this methodology for software engineering and reason about its suitability for this domain—from which most of the papers we identified originate. The other guidelines we found have been applied due to different and interdisciplinary research topics within the field of computer science, relying on corresponding guidelines. Still, this diversity raises several questions and issues: It seems necessary to compare the structure of the proposed guidelines to identify the extent of commonality and diversity in their processes. Researchers may find it challenging to compare reviews on the same topic if the authors performed and report different steps or if some steps are not explicit (Dresch et al. 2014). Another question is whether this large number of guidelines is necessary and useful. For software engineering, and to the extent of this study computer science, the guidelines of Kitchenham and Charters (2007) seem to dominate and may be used or consolidated with the remaining ones to define the standard in computer science.

Another problem arises, as approximately one fifth of all papers we analyzed are not reporting whether a specific guideline was used or not. This is a decrease compared to the findings of Kitchenham et al. (2010), who identified that less than half of the investigated SLRs report them. Nonetheless, such SLRs hamper the understandability, do not allow the reader to gain background information, and may complicate replications. In summary, while the awareness for guidelines seems to increase, there are still many papers that do not follow or report them.

5.3 $RQ_{1.2}$: What Search Strategies do Researchers Employ?

Measurement Again, we read each paper to identify the employed search strategy. In this case, we analyzed the mentioned searches, but also descriptions of the search process—especially because snowballing is a kind of manual search and researchers may not clearly differentiate these strategies. We identified for each paper whether the researchers employed an automatic, manual or snowballing search (cf. Section 2.2). As before, papers can rely on more than one method, as is also recommended, for example, by Zhang and Babar (2010).

Results We display the numbers of employed search strategies as `UpSet` (Lex et al. 2014) plot in Fig. 4. The bottom left part illustrates how many papers employed each strategy in total. In the remaining bottom part, `UpSet` plots display interaction sets (combinations) and the bars above show how often each interaction set has been used. Thus, `UpSet` plots are an equivalent representation to Venn diagrams, but are more useful for large sets or if scales cannot be represented properly.

In Fig. 4, we can see that the sole automatic search is used the most by far (165 times) and also in almost all combinations (278), except for 11 SLRs. Manual search and snowballing have been employed almost equally often with 78 and 69 papers relying on them,

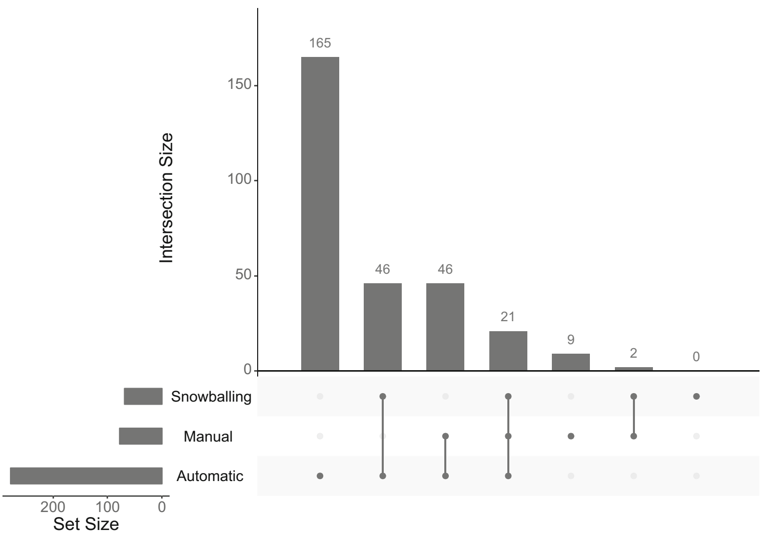


Fig. 4 Profile of search strategies employed in the SLRs we analyzed

respectively. However, they mostly occur in combination with automatic searches. A purely manual search was employed only nine times in our set of papers, while a sole snowballing strategy was never used. Combinations of these two methods are also rare, appearing only twice.

Discussion It is positive that every SLR that we investigated clearly defines the search strategy employed. Of these, almost all SLRs employed automatic searches, arguably because they allow to easily cover many venues. Still, in contrast to manual searches and snowballing, researchers have to provide additional information and must rely on literature resources. In particular, they cannot control the search mechanisms and evolution of these resources—potentially threatening replications of the search process. Also, the researchers have to adapt searches for different literature resources that each cover a subset of the available papers in computer science. This may threaten replications of the search process, due to flaws during the adaptation of search strings and inconsistencies within the literature resources.

5.4 $RQ_{1.3}$: What Literature Resources do Researchers Use to Search for Relevant Papers?

Measurement As aforementioned, researchers most often employed automatic searches in the SLRs that we analyzed. We identified what literature resources have been used by the 278 papers that relied on such an automatic search. Again, we assigned different versions (e.g., CiteSeer and CiteSeerX) to a single literature resource.

Results We display all literature resources that researchers used for automatic searches in our set of papers in Fig. 5. To improve readability, we omit 43 literature resources that appear fewer than five times and have been used in 58 papers. Overall, mostly IEEE Xplore, ACM Digital Library, Science Direct, and Springer Link have been used—representing four major publishers for computer science. With Google Scholar, Scopus, and Web of Science follow literature resources that index papers of different publishers. The researchers used interdisciplinary search indexes and publishers less frequently and we often omit these in Fig. 5. We remark that Science Direct[4] and Elsevier[5] denote apparently different search engines, where the first includes only peer-reviewed publications, while the second can also include other resources, such as websites.

Discussion While every paper mentions the literature resources that were searched, the results also indicate multiple threats to the validity of SLRs conducted with automatic searches. First, while including a broader set of literature resources allows for a more complete overview, it requires adaptations to the search strategy. For example, some literature resources allow searching only on a subset of fields, have a different syntax, and limit the number of results. Thus, to ensure the completeness and possibility of a replication, researchers have to provide the details about adaptations to the search query.

Second, as other findings also indicate (Zhang and Babar 2010; Budgen et al. 2018a), many SLRs rely on the literature resources of the four major publishers in computer science. For our set of papers, we see the same dominance of IEEE Xplore, ACM Digital Library,

[4] https://www.sciencedirect.com/
[5] https://www.elsevier.com/search

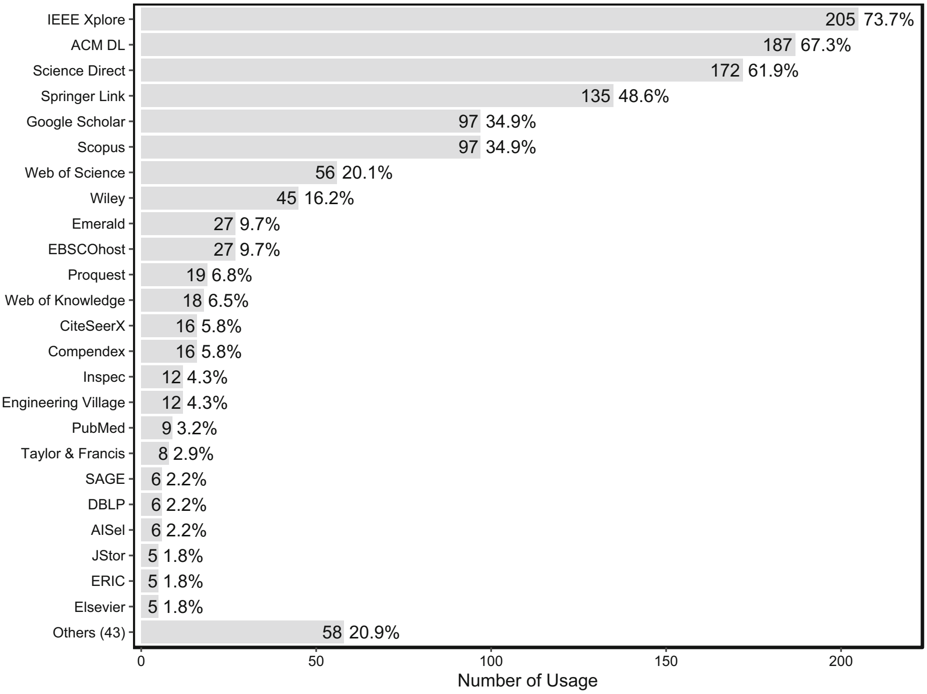


Fig. 5 Distribution of literature resources that were used in the SLRs we analyzed.

Science Direct and Springer Link. While this may ensure a certain quality, it can also prevent a complete overview on a research topic. As a result, such SLRs may face a selection bias and should be extended with snowballing or broader literature resources, such as, Scopus, Google Scholar, or DBLP.

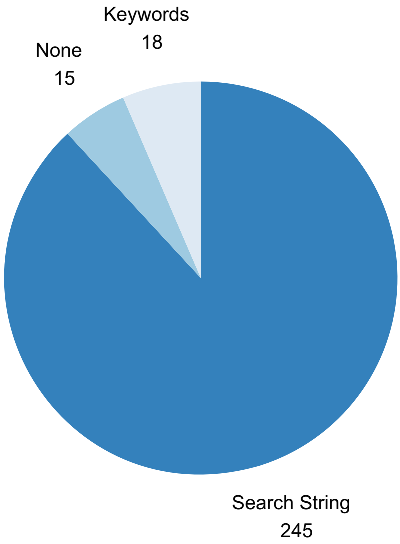
Finally, researchers depend on the interfaces provided by the literature resources. We already described that they can heavily differ and also evolve over time (cf. Section 2.3), potentially preventing replications. In particular, the question arises if the most used literature resources are consistent in the results that can be obtained. As an inconsistent behavior of literature resources would prevent any reliable SLR and its replication, we investigated this issue further with our descriptive multi-case study (cf. Section 6).

5.5 $RQ_{1.4}$: What Information on Search Strings do Researchers Provide?

Measurement To perform an automatic search, a corresponding search string is necessary. For our analysis, we differentiated three levels of information: *None*, *keywords*, and *search string*. We analyzed each paper to identify whether the researchers report a complete search string—including the keywords and their operators (cf. Section 4.2)—or only keywords. Researchers can only perform proper replications of searches if the full search string is described. Otherwise, it is unclear how the original researchers connected keywords or what they even searched for if they provide no information at all.

Results We show the overall outcome of our analysis in Fig. 6. As we can see, in 15 SLRs the researchers do not report any information at all and in 18 other SLRs the researchers

Fig. 6 Reported details about search strings for automated searches



do report only the keywords. Consequently, most researchers report the complete search string they used during their automatic search (245; 88.1%). These search strings mostly contain the simple boolean operators AND and OR, brackets for prioritizing, and enclosures for exact matches. Rarely, other operators—which are also not supported by all literature resources (Shakeel et al. 2018)—are used, for instance, to limit the search to certain parts of the document and wild-cards.

Discussion It is positive that most researchers report the full search string they used during their SLR. Only this way, they can mitigate threats to the replication of the searches they employed in an SLR. Researchers in all roles—conductors, reviewers, and readers—should be aware that a fully described search string is an essential requirement. Even better would be to report the search string for each literature resource used, which can take a lot of space. We rarely found papers that provided these adaptations, neither in the papers nor as an appendix. To ensure replications, it is essential to make the adaptations also available, for instance, as an online appendix that could also comprise other artifacts, such as used tools, results, and statistical data.

5.6 $RQ_{1.5}$: What Problems do Researchers Report?

Measurement This question is not directly concerned with replicating the search of an SLR. Instead, we aimed to consolidate the experiences of researchers that conducted SLRs—with particular interest on those that are connected to the search process. For this purpose, we read each paper carefully and identified all problems or pitfall the researchers reported.

Results Only few researchers mentioned challenges of conducting an SLR. Considering automatic searches, we identified the following problems:

– 35 papers report that it is challenging to adapt search strings to different literature resources, for various of the issues we described.

– Google Scholar allows a maximum of 256 characters (Ilyas and Khan 2015) and does not offer extended filters (Mbiydzenyuy 2015).
– There are several problems and inconsistencies regarding the different literature resources, such as, ACM Digital Library, IEEE Xplore, Science Direct, and Springer Link. For example, the lengths of search strings and options to search within certain parts of the available papers vary and are limited (Soni and Kodali 2011; Abdelmaboud et al. 2015; Afzal et al. 2016).

Overall, we found only specific problems related to the automatic search strategy and literature resources in computer science. Regarding manual searches and snowballing, some researchers mention that they take a lot of effort, but describe no concrete problems.

Discussion The results show that there are several limitations concerning automatic searches, as has also been discussed by other researchers (cf. Section 3). Because these limitations are related to technical problems, researchers cannot resolve them on their own. Instead, we must report such problems to the providers who can address them. However, manual searches and snowballing can at least mitigate the identified problems. The low number of papers reporting such problems seems surprising, considering that we included 278 automatic searches and how often the researchers used the same literature resources.

5.7 Threats to Validity

Construct Validity The construct validity represents how well the research questions can be understood and addressed with the used methodology—preventing biases of assumptions or design (Kitchenham et al. 2016). Our research questions require no previous knowledge about SLR methods. We did not limit our analysis to any specific guidelines, but derived independent, general-purpose questions. Thus, we argue that the construct validity of our tertiary study is not threatened.

Internal Validity Considering internal validity, we only used a single literature resource and a limited period. Furthermore, we excluded any paper that is not related to computer science and its interdisciplinary topics. As a result, we have certainly excluded papers that report an SLR. However, Scopus includes papers of most publishers with a certain quality and we explicitly focused on more recent years to also cope with temporal improvements. Overall, we argue that we are able to answer our research questions and scope our multi-case study based on the findings without threatening their validity.

A factor that we cannot influence is the indexing of literature resources and authors. Thus, some papers may be indexed incorrectly or some SLRs may be named differently (e.g., "systematic review"), which would both mean that we did not include these papers. Similarly, some of the included papers may actually report systematic mapping studies or even other studies, despite explicitly stating to apply an SLR. However, we investigated all papers in detail, which is why most of such papers have been excluded; and for those still included, the search processes are often identical or similar. Moreover, we included 289 papers and carefully checked the results to avoid selection biases. This way, we aimed to mitigate these threats to the internal validity.

Quite related to our study is one more external threat: Digital libraries may not be updated in time, but add Conference proceedings or Journal volumes only after some time. Thus, it may be that we did not find all papers for the covered period of time. Consequently, replications of our study may comprise a larger set of papers. Nonetheless, we included 289 papers and as there is some overlap in their characteristics (e.g., the literature resources used) to other studies, we argue that the results are reliable.

External Validity The results of our tertiary study cannot be generalized to all existing SLRs. However, this is mainly due to our goal of capturing the more recent development of SLRs—similar to the works, for example, by Kitchenham et al. (2010) or da Silva et al. (2011). Due to the ongoing evolution and advancement of SLR methods and literature resources, our results are hard to generalize. However, we present important insights into the current practice and in this context can help to raise the awareness for threats to the search processes used for SLRs.

Conclusion Validity Considering the results, we argue that they can be replicated by other researchers. We described each step and how we extracted the data to derive the results, which multiple authors controlled. As a result, we argue that the conclusions we derived are not threatened. However, the results can change depending on the considered time period, future developments, and the literature resources included. In particular, the literature resources will be the subject of our following descriptive multi-case study.

5.8 Summary

In the following, we summarize our findings to answer our first research question: *Do researchers report information needed to replicate the search of an SLR?* Considering the results of all sub-questions, we found that often much of the information needed to replicate searches is provided. Nonetheless, there are discrepancies in the details researchers describe and the applied methodologies. We identified the following threats:

1. Approximately 21% of the SLRs we analyzed did not use or do not report on using a guideline. Consequently, it is unclear whether the used methodology is appropriate. This may also hamper other researchers in replicating an SLR, as guidelines can help to better understand and scope the process.
2. As many guidelines exist, the question arises to what extent they cover the same criteria and details. Differences in the ways the conduct of an SLR is reported may prevent its replication. To address this point, a comparative analysis of existing guidelines seems necessary to identify variations and consolidate them.
3. Most SLRs solely rely on an automatic search. However, it is questionable if this search strategy covers all relevant papers (Zhang and Babar 2010; Jalali and Wohlin 2012; Badampudi et al. 2015). Another potential threat is that literature resources may be inconsistent, a threat that we further addressed with our descriptive multi-case study (cf. Section 6). Comparable to previous studies, we identified that the same literature resources are often used in computer science.
4. While almost all SLRs report the full search string, we found only few that also report necessary adaptations for different resources. Consequently, other researchers cannot verify that the used search strings were properly adapted. This threatens the completeness of the results and hampers replications. We are not aware of a previous study analyzing the reported search strings, wherefore we cannot argue about the temporal development in this regard.

In summary, we conclude that much of the necessary information for replicating the search of an SLR is reported. However, the granularity of details is questionable and many aspects of the search processes are unclear—meaning that replications and extensions have to build on some assumptions. We refer to this as the *reporting threat*, which researchers can overcome by reporting more detailed information.

6 Descriptive Multi-Case Study

Assuming that researchers have provided all information to replicate the search of an SLR, there still remains a potential source of threats: the literature resources that are used. Investigations show that such resources have varying characteristics, for example, in their search models, query design, indexing, and truncation of results (Shakeel et al. 2018; Babar and Zhang 2009; Bailey et al. 2007; Brereton et al. 2007). Thus, it can be challenging for researchers to adapt their search string to each literature resource, as was highlighted in 35 papers we analyzed in our tertiary study. However, a different factor to consider is whether the resources are deterministic, and thus consistently provide the same results for the same queries at different points in time. In order to investigate this issue, we conducted a descriptive multi-case study (Yin 2018) to answer our second research question:

RQ_2 *To what extent can we replicate searches on literature resources?*

Within this section, we describe the setup and conduct of our descriptive multi-case study, report and discuss the results, and analyze threats to validity.

6.1 Setup and Conduct

We focus on two **characteristics** of literature resources that are essential to allow replications: interface consistency and temporal consistency. During our tertiary study, we rarely found information on when exactly a search was performed, whether it was repeated, or through what interface the search was applied. However, if the literature resources do not behave consistently, these factors can have significant impact on the starting set of papers, which may bias the results. For example, we may retrieve varying results depending on the day on which we searched and whether we used the web interface or a tool with a crawler. We investigated these two characteristics as follows:

1. **Interface Consistency.** The results of our tertiary study indicate that researchers use different tools to conduct their searches. We compared the results of applying the same search strings on each selected literature resource using (1) their web interface (*manual*) and (2) through available *APIs*. Thus, we analyzed whether these two interfaces provide the same results, as both may be used by researchers and tools. As ACM did not provide an API when we started our studies, we did not include its digital library in this part of our analysis.
2. **Temporal Consistency.** With this characteristic, we investigated whether repeating a manual or an API search yields the same results. To this end, we compared the papers that we found through each resource and interface at different days. This allowed us to investigate if a replication at any point in time yields consistent results.

To analyze these two characteristics, we designed the following setup for our descriptive multi-case study based on the results of our tertiary study.

Automatic Search We aimed to analyze automatic searches, as most of the identified SLRs relied solely on this search strategy or used it in combinations. Overall, only 11 SLRs did not include any automatic search. As aforementioned, we considered automatic searches that may rely on web interfaces or APIs.

Most Prominent Resources We selected the most used literature resources in computer science, according to our results and existing studies (Budgen et al. 2018a; Kitchenham et al. 2016; Shakeel et al. 2018). Furthermore, we limited our selection to the most accessible ones (Shakeel et al. 2018), meaning those that have no access restrictions for searches (e.g., excluding Scopus and Web of Science) and that allow to download the results as collections (e.g., excluding Google Scholar). These criteria allowed us to automate parts of the conduct by automatically querying search strings for APIs and comparing the results. Finally, we included *IEEE Xplore* (IEEE), *ACM Digital Library* (ACM), *Science Direct*, and *Springer Link* (Springer).

Search Strings We aimed to cover different research areas in computer science and varying search-string designs. For this purpose, we created five search strings (abstracted):

String 1 `program comprehension OR debugging OR maintenance AND`
`study OR participants AND`
`code smells AND`
`integrated development environment OR programming language OR programming paradigm OR tool`

String 2 `tool support AND`
`keywords:` `software product line`

String 3 `security issue OR security issues AND`
`keywords:` `internet of things OR iot`

String 4 `function call OR graph database AND`
`keywords:` `intrusion detection system OR intrusion detection systems OR network security`

String 5 **`keywords:`** `type systems`

These strings cover different research areas that we are knowledgeable in. We combined the keywords between operators to enforce searching for exact string matches (e.g., program comprehension becomes "program comprehension") to limit the number of results. Then, we tested whether the queries retrieved varying numbers of papers and followed the restriction of most literature resources that return a maximum of around 2,000 results for a single query (Shakeel et al. 2018). To customize these abstract search strings, we implemented a tool that automatically adapted the strings to each resource and its API. Moreover, our tool automatically performed the search through the APIs. We remark that we searched always with the exact same search string, meaning that even if we may have had a fault in an adaptation, the papers identified should still be the same.

Technical Limitations Due to technical limitations of existing literature resources (Shakeel et al. 2018), we searched only in full texts and keywords—which we indicate with the corresponding prefixes and which most of the selected resources support for manual and API searches. However, there are further limitations that do not allow us to conduct the same searches on all resources. First, ACM does not provide an API, wherefore we cannot perform our API case studies. Second, Springer only allows full-text searches, which is why we can only employ the first search string on this literature resource.

Analysis Besides the fully automatic search performed by our tool, we also manually performed the automatic searches on the web interfaces. This was done by the second author, who executed the searches and inserted the exported results into the implemented tool. Afterwards, we automatically checked for duplicates to verify interface and temporal consistency, using digital object identifiers, if available, or the remaining bibliographic data, otherwise. Finally, we manually checked the results and verified all differences that we identified.

Time Frame We started with the fully automated analysis of our tool, to check if it behaves correctly. This process started on November 15th 2016 and we used the results to check for consistency of the APIs. We started 10 days later with the manual search on November 25th 2016 (with two additional days of testing on the ACM Digital Library), from which point forward we used all results for assessing the resources' consistencies. After this point in time, we performed all searches once a day until December 23th 2016.

6.2 Results and Discussion

In the following, we distinguish between the results of our descriptive multi-case study that are concerned with *interface consistency* and *temporal consistency*. For each, we first report the results and afterwards discuss the implications.

Interface Consistency We show the results for the interface consistency of IEEE, Springer, and Science Direct in Figs. 7, 8, and 9, respectively. To better illustrate the data, we split it up for each search string that we employed and we separately display the numbers of returned papers for each resource. The gray bars show the numbers of papers that we identified through both, manual and API search, on each day, represented by the number at the bottom of each bar. Thus, the gray bars indicate the consistency of both searches compared to each other.

We further show the numbers of papers that we identified solely with one of the two searches, but not the other. The second row of numbers from the bottom corresponds to the API search, illustrated with the circles (∘) on top or above the gray bars. Identically, the row of numbers on top represents papers found solely through the manual search and aligns to the dots (·) in the plots. Consequently, if the dots are in the middle of the circles, and thus on top of the gray bars, there are no discrepancies between the searches, which indicates consistency of both interfaces compared to each other. In such situations, the second and third row of numbers that indicate the unique papers for each interface are zero.

Comparing the different search strings, we can see that they all return different numbers of papers. As aforementioned, we intended to have varying sample sizes to broaden our testing setup. We remark that increasing and afterwards constant numbers indicate that new papers have been added—which we expected for a period of a month.

For IEEE (cf. Fig. 7), we can see that the results of our searches vary for several days of our observation. We can see that on some days the API search is incomplete compared to the manual search, returning even fewer papers than the days before and afterwards. Seeing the overall numbers of papers, the results indicate that this seems to partly align to days at which papers are added into this resource. However, for String 3, a mismatch of one to four papers persists throughout our whole observation period. We see a similar behavior for Springer (cf. Fig. 8), but here it seems to be consistent. While IEEE is sometimes varying, we never obtained the same papers for Springer when comparing manual and API searches. More precisely, both searches return distinct, yet relevant sets of papers. In addition, there seems

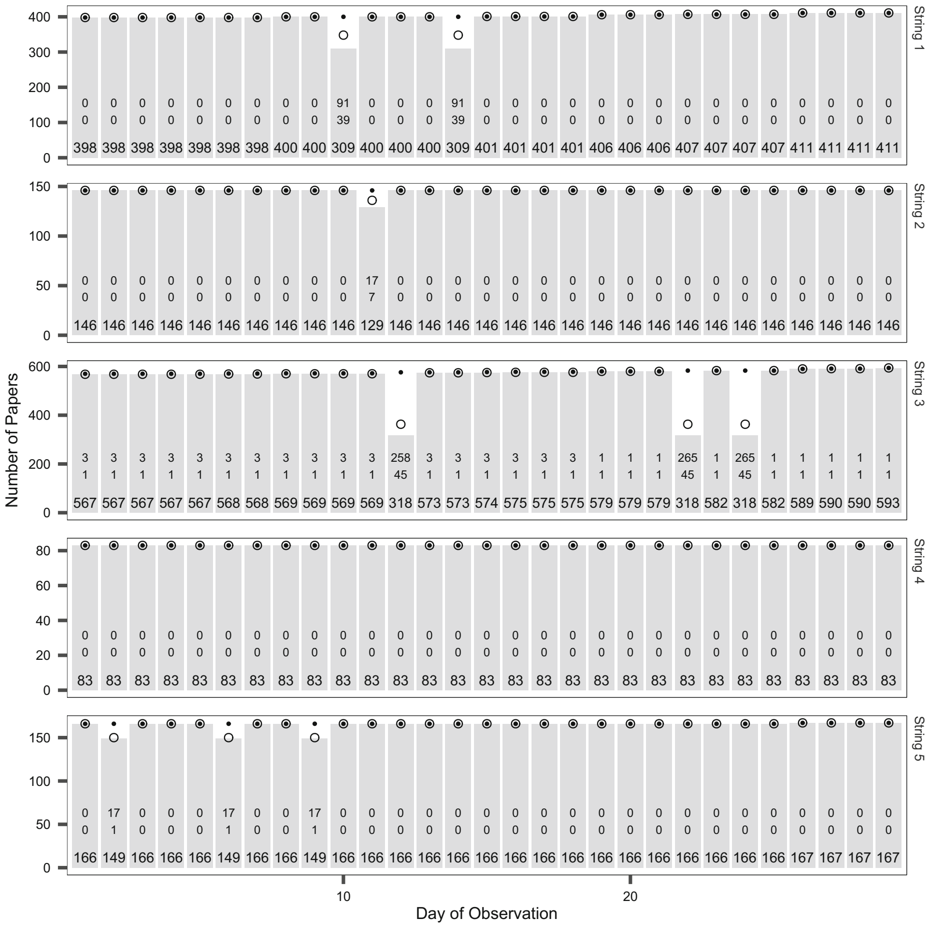


Fig. 7 Interface consistency of IEEE. From the bottom: The first row of numbers represents the papers identified by both searches (manual and API) and corresponds to the gray bars. The second row represents the papers identified only by the API search and corresponds to the circles (∘). The third row represents the papers identified only by the manual search and corresponds to the dots (·)

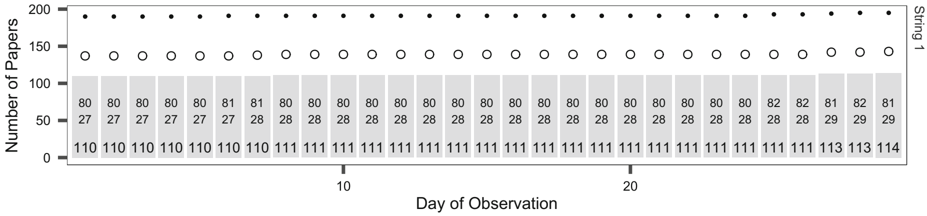


Fig. 8 Interface consistency of Springer. From the bottom: The first row of numbers represents the papers identified by both searches (manual and API) and corresponds to the gray bars. The second row represents the papers identified only by the API search and corresponds to the circles (∘). The third row represents the papers identified only by the manual search and corresponds to the dots (·)

to be an update delay (see also Fig. 10), as we sometimes found new papers at different days for each search. For example, at day six, we found one more paper with the manual search that was also found by the API search at day eight—at which the number of papers found by both searches increased by one, while the number for the manual search (top row) decreased by one. In contrast, for Science Direct (cf. Fig. 9), we can see that all searches always returned consistent results. This is the behavior that is necessary to ensure that we can replicate a search and obtain the same results.

Discussion The results of our descriptive multi-case study are concerning. They indicate that researchers cannot literally replicate the searches of many SLRs at any point in time. Worse, such searches are potentially incomplete if they rely solely on a search through the web interface or an API search. As most SLRs report that they only used automatic searches (without additional snowballing or other extensions, cf. Section 5.3) and do not report that

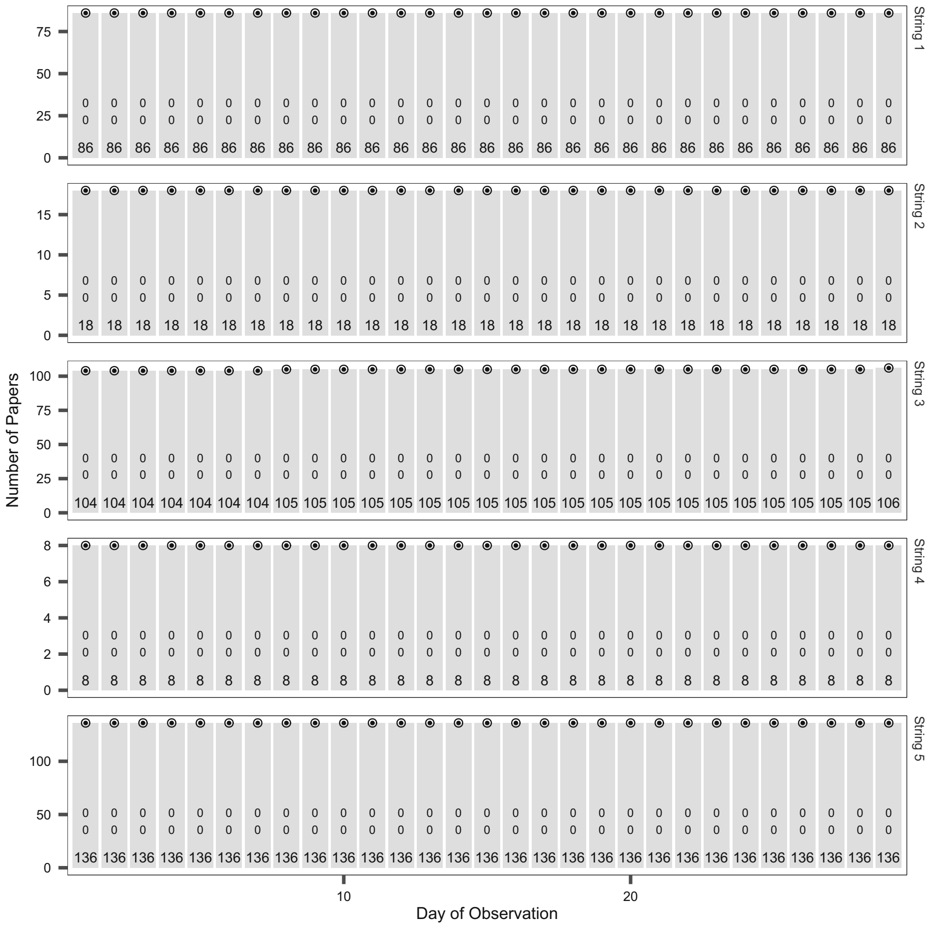


Fig. 9 Interface consistency of Science Direct. From the bottom: The first row of numbers represents the papers identified by both searches (manual and API) and corresponds to the gray bars. The second row represents the papers identified only by the API search and corresponds to the circles (○). The third row represents the papers identified only by the manual search and corresponds to the dots (·)

they used both interfaces (or which of these), they may have missed some papers in the initial set. Thus, also some relevant papers may be missing form the included results and hide important evidence (Badampudi et al. 2015; Jalali and Wohlin 2012).

Moreover, without knowing the exact tools and search interfaces that the researchers used, it is not possible to replicate an SLR. For example, some tools may rely on an API, while others crawl the web interfaces of the literature resources. Thus, replicating a search is only possible if the same version of the same tool is used. Considering that we barely found this information during our tertiary study, this is a significant threat for replications of the search processes of SLRs.

Temporal Consistency In Fig. 10, we show the results for the temporal consistency of the returned papers. We display for each literature resource and search string that we could

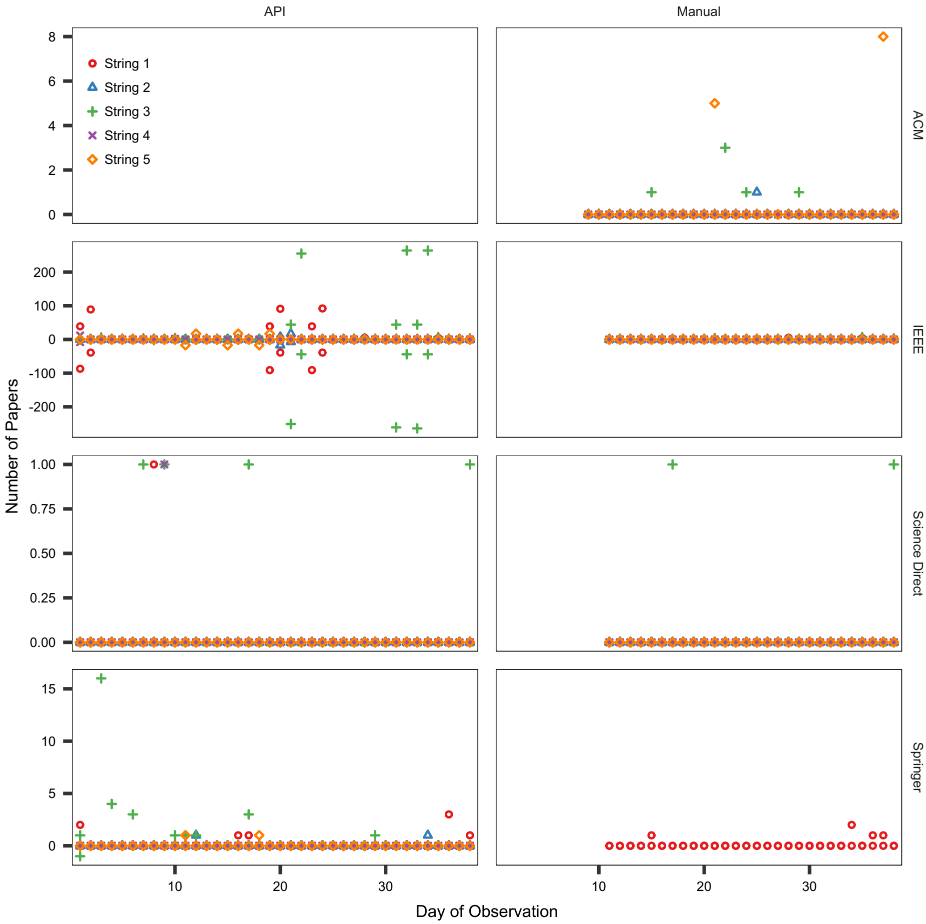


Fig. 10 Consistency of API (left) and manual searches (right) for each literature resource and search string. ACM does not provide an API, while Springer does not support the same features in both scenarios. Marks that are above zero represent added papers, while marks below zero represent removed papers. We show two marks for each observation to consider that papers may have been added and removed at the same day

employ, how many papers have been added or removed at each day of our observation. To this end, we show for each string two marks at each day, with those above zero representing additions and those below zero representing deletions. If only one, additions or deletions, or neither occurred, one mark is always at zero. We remark again that ACM does not provide an API, while we could only employ the first search string for Springer.

We can see that for the manual search of all resources only additions appeared during our whole observation. This is the expected behavior, as papers are added to literature resources after publication, and thus made available to researchers. In contrast, the API searches of IEEE and Springer (once) also returned fewer papers at some point in time. For IEEE, we can see a certain pattern: The same number of papers that was removed at one day was added again the day after. For instance, at day 33, 44 papers were added, while 264 were removed for String 3. On the next day, the same papers were removed and added again, respectively. Consequently, there is no difference in the results for days 32 and 34, besides added papers. The same is true for the other occasions and while this pattern is common, it is not always aligned to an addition of papers (cf. Fig. 7). It is unclear why these discrepancies occur.

Discussion Again, the results we obtained for temporal consistency are concerning. While the manual search of each literature resource seems to be consistent (as far as our case studies are concerned), the API searches vary heavily. The most extreme case is IEEE, for which we found large changes for some days of our observation. This means that any search conducted with the API at a single day in this resource can be biased. If the search is not checked at multiple days, researchers may include fewer or more papers. Moreover, researchers who replicate such an SLR have to be lucky to obtain the same papers.

6.3 Threats to Validity

Internal Validity The selected search strings yield different sets of papers and may be subject to fewer or more changes, depending on the topic and publication activity. Consequently, this could be a threat to the internal validity, as we derived our findings based on only five strings. However, these strings cover different topics and intentionally return varying numbers of papers. Thus, we argue that they are representative and show several problems of the analyzed literature resources. This remains the case, even though we could not employ all search strings on all literature resources.

External Validity Overall, we only considered four literature resources. This means that we cannot conclude in general how reliable literature resources in software engineering (or computer science) are. However, we focused on established resources and the identified issues should be investigated and tested for all of them. While our selection of literature resources means that we cannot directly transfer the results, our findings are still alarming and should be a warning sign when using other literature resources.

Conclusion Validity Considering the conclusions we derived, we reported all details of our descriptive multi-case study and carefully analyzed the results. Thus, we argue that we were able to derive meaningful results that are reliable based on our findings. Still, one issue in our study may be that we wrongly identified papers as overlapping or unique, for example, due to missing or wrong data (e.g., missing DOI). To mitigate this threat, we partly verified the results manually to ensure that the matches of papers have been correct.

6.4 Summary

Overall, we can summarize our findings to answer our second research question: *To what extent can we replicate searches on literature resources?* Based on the results we obtained with our descriptive multi-case study, we found that replicating searches conducted in SLRs is problematic. Manual as well as API searches of literature resources can be biased to some extent and only Science Direct and ACM (for their web interfaces) fulfill our consistency requirements. In particular, we found the following factors that can prevent a reliable replication of searches reported in SLRs:

1. Only Science Direct seems to be interface consistent, while neither IEEE nor Springer are. We cannot conclude meaningful results for ACM, due to the missing API. In contrast, the manual search of all analyzed libraries seems to be temporally consistent. Considering the problems of available APIs, we argue that using them or tools that are based on these threatens the ability of researchers to replicate the search process of SLRs. Moreover, additional information needs to be reported, in particular, on repeating and verifying searches on several days.
2. The API searches partly allow different options to define search strings compared to the web interfaces. As the researchers who conducted the SLRs we analyzed report that they used different tools (most are not reporting any information in this regard), this discrepancy threatens replications.

We considered four of the most widely-used literature resources in computer science and conclude that it is hardly possible to replicate automatic searches of SLRs. This is mostly due to misbehavior of some of the resources, which results in some papers potentially missing in the conducted SLRs. We refer to this situation as *searching threat*, which researchers themselves cannot fully overcome, but address by providing additional information, repeating their searches, and combining search strategies. Arguably, researchers could rely only on those resources that are consistent, which should mean they are deterministic. However, we found problems in established resources that should not be ignored and it seems necessary for the providers to check and prove that their resources work properly.

7 Conclusions

In this article, we reported the details of an extensive tertiary study and a descriptive multi-case study. Within both, we have been concerned with the ability to replicate the searches of SLRs. While our results show that most researchers report the information that is requested by common guidelines and should *ideally* be enough to replicate an SLR, we identified several issues that are connected to **reporting (R)** and **searching (S)** threats, as discussed in Sections 5 and 6, respectively. We also identified one **additional (A)** threat that we experienced during our study:

- R_1 Not all SLRs report that they used a guideline and it is unclear, what differences between guidelines have what impact.
- R_2 Many papers may omit information that is relevant for replications, which can be due to unawareness of existing issues or space limitations.
- R_3 Researchers cannot employ search strings the same way to all literature resources, wherefore all adapted versions should be reported.

- R_4 Many researchers seem to be unaware of searching and reporting threats to SLRs, which they should describe.
- S_1 A sole automatic API search seems unreliable, wherefore researchers should include a larger set of resources (including summarizing ones, such as DBLP or Google Scholar) and also an additional manual and/or snowballing phase.
- S_2 While the development of SLR tools is an important research direction (Hassler et al. 2016), it seems necessary to first analyze the problems of literature resources further to implement repeatable search processes.
- S/R_1 Literature resources have several limitations concerning the consistency of their API searches. Researchers have to be aware of these issues and have to report exactly how they searched for papers. Thus, we recommend to employ API searches at multiple days to check the consistency of the returned results.
- A/S_1 We found that using search strings with different encodings, for instance, manual input and copied from different PDF readers, can break the searches of literature resources. In particular, we tested this for the four resources we analyzed in our descriptive multi-case study. Overall, IEEE as well as Springer are vulnerable to this issue: Different encodings in the search string may yield varying results with only a small overlap, while all returned papers are relevant. Thus, both resources provide incomplete results if not all encodings are tested.

Overall, we identified several reporting and searching threats that challenge the replication of the search process of an SLR. While some of the reporting threats, for instance, missing details on the search string (R_2) and its adaptations (R_3), already pose significant problems in verifying and replicating a search strategy, they can become even worse in combination with searching threats. For example, if we only know of a single search string for an automated search (R_3), but not how the researchers employed it (S/R_1), several methodological variations are possible. In this case, it would be necessary that we adapt the search string ourselves to perform a replication, without being able to check whether the original study has been correct in this regard. Even then, we do not know whether we should use the web interface, an API, or a certain tool to execute the queries. Thus, the threats to validity and replications of SLRs may multiply, due to the dependencies between these threats.

Particularly important is researchers' awareness for such threats, when we put our findings into the context of related work (cf. Section 3). Several researchers highlighted the importance of replications, but our findings suggest that these are hampered significantly. Multiple, partly contradicting, studies investigated to what extent a different set of papers and search strategies may result in biased conclusions. Due to the different results, it may be best to summarize that missing papers can result in biases, and thus the ability to understand and replicate searches is important. Similar to us, other studies suggest an improvement in the details that are reported for an SLR. Still, we see that some of the threats we identified may be known by experts, but rarely by novices, and that additional information is necessary to assess to what extent we can address these threats.

Considering our findings, we can answer the question of our title, whether we can replicate the searches of SLRs, with no. Overcoming the identified threats is not only in the hands of researches, but also publishers and maintainers of literature resources. Still, researchers who conduct, review, read, and replicate SLRs should be aware of the identified threats. Researchers can address these threats to some extent by providing additional information and extending their searches.

In future work, we see the need for more empirical studies on literature resources and their issues. For example, it would be helpful to explain the patterns we found for IEEE. In the same direction, interviews and communication with developers and maintainers of literature resources could help to understand reasons for the inconsistencies we found. Based on this, we could derive more precise recommendations for researchers on how to search in specific resources or may be able to provide an interface that allows for consistent searching. Moreover, developing tools to support and improve SLRs is a major issue that has to be addressed. We remark that we did not perform a sensitivity analysis in this article, meaning that we did not investigate to what extent the discrepancies we identified would actually change the outcome of an SLR, which is important future work. Finally, while SLRs are a valuable research method, we as computer scientists seem to have few to no reliable search opportunities or interfaces and important questions are, whether we should rely on them at this point in time and why we seem to have these problems?

Acknowledgments This research has been supported by the German Research Foundation (DFG) project EXPLANT grants LE 3382/2-1, LE 3382/2-3, SA 465/49-1, and SA 465/49-3.

Appendix

Authors	Title	Year
Cruz-Hinojosa, N. J.; Gutiérrez-De-Mesa, J.A.	Literature review of the situation research faces in the application of ITIL in Small and Medium Enterprises	2016
Khosravi, P.; Rezvani, A.; Wiewiora, A.	The impact of technology on older adults' social isolation	2016
Dikert, K.; Paasivaara, M.; Lassenius, C.	Challenges and success factors for large-scale agile transformations: A systematic literature review	2016
Jafari Navimipour, N.; Charband, Y.	Knowledge sharing mechanisms and techniques in project teams: Literature review, classification, and current trends	2016
Milani, A. S.; Navimipour, N. J.	Load balancing mechanisms and techniques in the cloud environments: Systematic literature review and future trends	2016
Idri, A.; Hosni, M.; Abran, A.	Systematic literature review of ensemble effort estimation	2016
Zare, M.; Pahl, C.; Rahnama, H.; Nilashi, M.; Mardani, A.; Ibrahim, O.; Ahmadi, H.	Multi-criteria decision making approach in E-learning: A systematic review and classification	2016
Costa, E.; Soares, A. L.; De Sousa, J. P.	Information, knowledge and collaboration management in the internationalisation of SMEs: A systematic literature review	2016
Melendez, K.; Dávila, A.; Pessoa, M.	Information technology service management models applied to medium and small organizations: A systematic literature review	2016

Authors	Title	Year
Soltani, Z.; Navimipour, N. J.	Customer relationship management mechanisms: A systematic review of the state of the art literature and recommendations for future research	2016
Garousi, V.; Mäntylä, M. V.	When and what to automate in software testing? A multi-vocal literature review	2016
Sánchez Guinea, A.; Nain, G.; Le Traon, Y.	A systematic review on the engineering of software for ubiquitous systems	2016
Tarhan, A.; Turetken, O.; Reijers, H. A.	Business process maturity models: A systematic literature review	2016
Ali, S.; Khan, S. U.	Software outsourcing partnership model: An evaluation framework for vendor organizations	2016
Martins, L. E. G.; Gorschek, T.	Requirements engineering for safety-critical systems: A systematic literature review	2016
Balaid, A.; Abd Rozan, M. Z.; Hikmi, S. N.; Memon, J.	Knowledge maps: A systematic literature review and directions for future research	2016
Bissi, W.; Serra Seca Neto, A. G.; Emer, M. C. F. P.	The effects of test driven development on internal quality, external quality and productivity: A systematic review	2016
McGill, M. M.; Decker, A.; Settle, A.	Undergraduate students' perceptions of the impact of pre-college computing activities on choices of major	2016
de França, B. B. N.; Travassos, G. H.	Experimentation with dynamic simulation models in software engineering: planning and reporting guidelines	2016
Sajid, A.; Abbas, H.	Data privacy in cloud-assisted healthcare systems: State of the art and future challenges	2016
Souza, D. M.; Felizardo, K. R.; Barbosa, E. F.	A systematic literature review of assessment tools for programming assignments	2016
Anuar, U.; Ahmad, S.; Emran, N. A.	A simplified systematic literature review: Improving Software Requirements Specification quality with boilerplates	2016
Gheni, A. Y.; Jabar, M. A.; Jusoh, Y. Y.; Mohd Ali, N.; Abdullah, R. H.	Factors for communication technologies selection within virtual software teams	2016
Jia, J.; Zhang, P.; Capretz, L. F.	Environmental factors influencing individual decisionmaking behavior in software projects: A systematic literature review	2016

Authors	Title	Year
Muccini, H.; Sharaf, M.; Weyns, D.	Self-adaptation for cyber-physical systems: A systematic literature review	2016
Yaseen, M.; Baseer, S.; Ali, S.; Khan, S. U.; Abdullah	Requirement implementation model (RIM) in the context of global software development	2016
Batool, D. -. -.; Molta, Y. H.; Sarwar, A.; Abbasi, M. A.; Jabeen, J.	Software architecture and requirements: A systematic literature review	2016
Luh, R.; Marschalek, S.; Kaiser, M.; Janicke, H.; Schrittwieser, S.	Semantics-aware detection of targeted attacks: a survey	2016
Spanos, G.; Angelis, L.	The impact of information security events to the stock market: A systematic literature review	2016
Soomro, A. B.; Salleh, N.; Mendes, E.; Grundy, J.; Burch, G.; Nordin, A.	The effect of software engineers' personality traits on team climate and performance: A Systematic Literature Review	2016
Tahir, T.; Rasool, G.; Gencel, C.	A systematic literature review on software measurement programs	2016
Farwick, M.; Schweda, C. M.; Breu, R.; Hanschke, I.	A situational method for semi-automated Enterprise Architecture Documentation	2016
Schatz, D.; Bashroush, R.	Economic valuation for information security investment: A systematic literature review	2016
Kekgathetse, M. B.; Letsholo, K. J.	A survey on database synchronization algorithms for mobile device	2016
Lu, Q.; Zhu, L.; Zhang, H.; Wu, D.; Li, Z.; Xu, X.	MapReduce job optimization: A mapping study	2016
Aris, H.	Sustainable solvers participation in non-profit mobile crowdsourcing initiatives: A review of successful applications	2016
Marçal, I.; Garcia, R. E.; Eler, D. M.; Junior, C. O.; Correia, R. C. M.	Techniques for the identification of crosscutting concerns: A systematic literature review	2016
Da Silva, G. C.; De Figueiredo Carneiro, G.	Software process improvement in small and medium enterprises: A systematic literature review	2016
Li, Z.; Zhang, H.; O'Brien, L.; Jiang, S.; Zhou, Y.; Kihl, M.; Ranjan, R.	Spot pricing in the cloud ecosystem: A comparative investigation	2016
Ali, O.; Soar, J.; Yong, J.	An investigation of the challenges and issues influencing the adoption of cloud computing in Australian regional municipal governments	2016

Authors	Title	Year
Qu, M.; Yu, S.; Chen, D.; Chu, J.; Tian, B.	State-of-the-art of design, evaluation, and operation methodologies in product service systems	2016
Soomro, Z. A.; Shah, M. H.; Ahmed, J.	Information security management needs more holistic approach: A literature review	2016
Hassan, M.; Jabar, M. A.; Sidi, F.; Jusoh, Y. Y.; Hassan, S.	Enterprise resource planning adoption lifecycle: A systematic literature review	2016
Hakami, Y. A. A.; Hussei, A. R. C. B.; Adenuga, K. I.	Preliminary model for computer based assessment acceptance in developing countries	2016
Rasti, Z.; Darajeh, A.; Khayami, R.; Sanatnama, H.	Systematic literature review in the area of Enterprise architecture during past 10 years	2016
Oliveira, J. B.; Lima, R. S.; Montevechi, J. A. B.	Perspectives and relationships in supply chain simulation: A systematic literature review	2016
Torrecilla-Salinas, C. J.; Sedeño, J.; Escalona, M. J.; Mejías , M.	Agile, web engineering and capability maturity model integration: A systematic literature review	2016
Gasparic, M.; Janes, A.	What recommendation systems for software engineering recommend: A systematic literature review	2016
Boyle, E. A.; Hainey, T.; Connolly, T. M.; Gray, G.; Earp, J.; Ott, M.; Lim, T.; Ninaus, M.; Ribeiro, C.; Pereira, J.	An update to the systematic literature review of empirical evidence of the impacts and outcomes of computer games and serious games	2016
Sobernig, S.; Hoisl, B.; Strembeck, M.	Extracting reusable design decisions for UML-based domain-specific languages: A multi-method study	2016
Chen, T.	Technology-supported peer feedback in ESL/EFL writing classes: a research synthesis	2016
Al-Zubidy, A.; Carver, J. C.; Heckman, S.; Sherriff, M.	A (updated) review of empiricism at the SIGCSE technical symposium	2016
Decker, A.; McGill, M. M.; Settle, A.	Towards a common framework for evaluating computing outreach activities	2016
Charband, Y.; Jafari Navimipour, N.	Online knowledge sharing mechanisms: A systematic review of the state of the art literature and recommendations for future research	2016
Smuts, H.; Kotze, P.; Van Der Merwe, A.; Loock, M.	Threats and opportunities for information systems outsourcing	2016

Authors	Title	Year
Banaeianjahromi, N.; Smolander, K.	What do we know about the role of enterprise architecture in enterprise integration? A systematic mapping study	2016
Effing, R.; Spil, T. A. M.	The social strategy cone: Towards a framework for evaluating social media strategies	2016
Elbattah, M.; Roushdy, M.; Aref, M.; Salem, A. -. M.	Large-scale ontology storage and query using graph database-oriented approach: The case of Freebase	2016
Yaseen, M.; Baseer, S.; Sherin, S.	Critical challenges for requirement implementation in context of global software development: A systematic literature review	2016
Szvetits, M.; Zdun, U.	Systematic literature review of the objectives, techniques, kinds, and architectures of models at runtime	2016
Thuan, N. H.; Antunes, P.; Johnstone, D.	Factors influencing the decision to crowdsource: A systematic literature review	2016
Najafabadi, M. K.; Mahrin, M. N.	A systematic literature review on the state of research and practice of collaborative filtering technique and implicit feedback	2016
Caron, X.; Bosua, R.; Maynard, S. B.; Ahmad, A.	The internet of things (IoT) and its impact on individual privacy: An Australian perspective	2016
Isern, D.; Moreno, A.	A systematic literature review of agents applied in healthcare	2016
Hosseinzadeh, S.; Hyrynsalmi, S.; Conti, M.; Leppänen, V.	Security and privacy in cloud computing via obfuscation and diversification: A survey	2016
Ngadiman, N.; Sulaiman, S.; Wan Kadir, W. M. N.	A systematic literature review on attractiveness and learnability factors in Web applications	2016
Alasbali, N.; Benatallah, B.	Open source as an innovative approach in computer science education A systematic review of advantages and challenges	2016
Mubasher, M. M.; Syed Waqar Ul Qounain, J.	Systematic literature review of vehicular traffic flow simulators	2016
Benslimane, Y.; Yang, Z.; Bahli, B.	Key topics in cloud computing security: A systematic literature review	2016
Inzunza, S.; Juárez-Ramírez, R.; Ramírez-Noriega, A.	User and context information in context-aware recommender systems: A systematic literature review	2016

Authors	Title	Year
Bergström, E.; ÃĚhlfeldt, R. -.	Information classification enablers	2016
Afzal, W.; Alone, S.; Glocksien, K.; Torkar, R.	Software test process improvement approaches: A systematic literature review and an industrial case study	2016
Lawal, M. A.; Sultan, A. B. M.; Shakiru, A. O.	Systematic literature review on SQL injection attack	2016
Nabavi, A.; Taghavi-Fard, M. T.; Hanafizadeh, P.; Taghva, M. R.	Information technology continuance intention: A systematic literature review	2016
Carvalho, J. V.; Rocha, Á.; Abreu, A.	Information systems and technologies maturity models for healthcare: A systematic literature review	2016
Shozi, N. A.; Mtsweni, J.	Big data privacy and security: A systematic analysis of current and future Challenges	2016
Grund, C. K.; Meier, M. C.	Towards game-based management decision support: Using serious games to improve the decision process	2016
Yaman, S. G.; Sauvola, T.; Riungu-Kalliosaari, L.; Hokkanen, L.; Kuvaja, P.; Oivo, M.; Männistö, T.	Customer involvement in continuous deployment: A systematic literature review	2016
Vahida, S.; Lehouxb, N.; De Santa-Eulaliac, L. A.; D'Amoursb, S.; Frayret, J. -.; Venkatadrie, U.	Supply chain modelling frameworks for forest products industry: A systematic literature review	2016
Zander, S.; Kolbe, L. M.	Supplier relationship management in information systems research: A literature review	2016
Aulkemeier, F.; Schramm, M.; Iacob, M. -.; van Hillegersberg, J.	A service-oriented e-commerce reference architecture	2016
Sepúlveda, S.; Cravero, A.; Cachero, C.	Requirements modeling languages for software product lines: A systematic literature review	2016
Alzoubi, Y. I.; Gill, A. Q.; Al-Ani, A.	Empirical studies of geographically distributed agile development communication challenges: A systematic review	2016
Lee, J.	Drivers and consequences in transforming work practices	2016
Febrero, F.; Calero, C.; Moraga, M. Á.	Software reliability modeling based on ISO/IEC SQuaRE	2016
Chauhan, M. A.; Babar, M. A.; Benatallah, B.	Architecting cloud-enabled systems: A systematic survey of challenges and solutions	2016
Haselberger, D.	A literature-based framework of performance-related leadership interactions in ICT project teams	2016

Authors	Title	Year
Vichitvanichphong, S.; Talaei-Khoei, A.; Kerr, D.; Ghapanchi, A. H.; Scott-Parker, B.	Good old gamers, good drivers: Results from a correlational experiment among older drivers	2016
Cano, C.; Melgar, A.; Dávila, A.; Pessoa, M.	Comparison of software process models. A systematic literature review	2015
Mohamad Hazawawi, N. A.; Othman, M. F. I.; Emran, M. H.; Zakaria, M. H.; Che Pee, N.; Othman, M. A.	A structured literature review on applied neuroscience in Information Systems (neuroIS)	2015
Al-Arabiat, D.; Wan Ahmad, W. F.; Sarlan, A.	Review on critical factors of adopting cloud mobile learning	2015
Ahmad, M.; Rahim, L. A.; Arshad, N. I.	Modelling educational games through multi-domain framework in light of game play	2015
Piran, F. A. S.; Lacerda, D. P.; Antunes, J. A. V.; Viero, C. F.; Dresch, A.	Modularization strategy: Analysis of published articles on production and operations management (1999 to 2013)	2015
Soleymani, S. A.; Abdullah, A. H.; Hassan, W. H.; Anisi, M. H.; Goudarzi, S.; Rezazadeh Baee, M. A.; Mandala, S.	Trust management in vehicular ad hoc network: a systematic review	2015
Nishida, A. K.; Braga, J. C.	Systematic review of literature: Educational games about electric energy consumption	2015
Brito, J. A.; Amorim, R.; De Sousa Monteiro, B.; Gomes, A. S.; De Melo Filho, I. J.	Effectiveness of practices with sensors in engaging in meaningful learning in higher education: Extending a framework of ubiquitous learning	2015
Ly, L. T.; Maggi, F. M.; Montali, M.; Rinderle-Ma, S.; Van Der Aalst, W. M. P.	Compliance monitoring in business processes: Functionalities, application, and tool-support	2015
Steinhueser, M.; Richter, A.; Smolnik, S.	How to bridge the boundary? Determinants of inter-organizational social software usage	2015
Mater, W.; Ibrahim, R.	Factors supporting teamwork communication in clinical pathways: Systematic literature review	2015
Tosun, A.; Bener, A. B.; Akbarinasaji, S.	A systematic literature review on the applications of Bayesian networks to predict software quality	2015
Hashim, S. N.; Abdullah, R.; Ibrahim, H.	Collaborative knowledge management system strategic planning (CKMS2P): A systematic literature review	2015
Mahmud, F.; Aris, H.	State of mobile crowdsourcing applications: A review	2015

Authors	Title	Year
Clear, T.	'Follow the moon' development: Writing a systematic literature review on Global Software Engineering Education	2015
Carroll, C.; Falessi, D.; Forney, V.; Frances, A.; Izurieta, C.; Seaman, C.	A mapping study of software causal factors for improving maintenance	2015
Martinez-Fernandez, S.; Medeiros Dos Santos, P. S.; Ayala, C. P.; Franch, X.; Travassos, G. H.	Aggregating empirical evidence about the benefits and drawbacks of software reference architectures	2015
Holm, H.; Sommestad, T.; Bengtsson, J.	Requirements engineering: The quest for the dependent variable	2015
Mbiydzenyuy, G.	Arrival times with hours of service regulations for truck drivers-tracks and gaps from current research	2015
Abdalla, G.; Damasceno, C. D. N.; Guessi, M.; Oquendo, F.; Nakagawa, E. Y.	A systematic literature review on knowledge representation approaches for systems-of-systems	2015
Neto, C. R. L.; Cardoso, M. P. S.; Chavez, C. V. F. G.; Almeida, E. S. D.	Initial evidence for understanding the relationship between product line architecture and software architecture recovery	2015
Hassan, M. M.; Afzal, W.; Blom, M.; Lindstrom, B.; Andler, S. F.; Eldh, S.	Testability and software robustness: A systematic literature review	2015
Dutra, A. C. S.; Prikladnicki, R.; Franca, C.	What do we know about high performance teams in software engineering? Results from a systematic literature review	2015
Santana, C.; Queiroz, F.; Vasconcelos, A.; Gusmao, C.	Software process improvement in agile software development: A systematic literature review	2015
Svensson, R. B.; Taghavianfar, M.; Gren, L.	Creativity techniques for more creative requirements: Theory vs. practice	2015
Oueslati, H.; Rahman, M. M.; Othmane, L. B.	Literature review of the challenges of developing secure software using the agile approach	2015
Decker, A.; McGill, M. M.; Settle, A.	Computing outreach literature review	2015
Afolabi, A. O.; Toivanen, P.; Haataja, K.; Mykkanen, J.	Systematic literature review on empirical results and practical implementations of healthcare recommender systems: Lessons learned and a novel proposal	2015
Eikey, E. V.; Reddy, M. C.; Kuziemsky, C. E.	Examining the role of collaboration in studies of health information technologies in biomedical informatics: A systematic review of 25 years of research	2015

Authors	Title	Year
Inayat, I.; Salim, S. S.; Marczak, S.; Daneva, M.; Shamshirband, S.	A systematic literature review on agile requirements engineering practices and challenges	2015
Ermilov, T.; Khalili, A.; Auer, S.	Ubiquitous semantic applications: A systematic literature review	2015
Chagas, A.; Santos, M.; Santana, C.; Vasconcelos, A.	The impact of human factors on agile projects	2015
Abia, M.; Brown, I.	A polymorphic model of information systems success: An outcome of a literature-based grounded theory analysis	2015
Rashid, M.; Anwar, M. W.; Khan, A. M.	Identification of trends for model based development of embedded systems	2015
Salvatori, L.; Marcantoni, F.	Social commerce: A literature review	2015
Mahmood, S.; Niazi, M.; Hussain, A.	Identifying the challenges for managing component-based development in global software development: Preliminary results	2015
Rodriguez, D.; Dolado, J.; Tuya, J.	Bayesian concepts in software testing: An initial review	2015
Jiang, S.; Zhang, H.; Gao, C.; Shao, D.; Rong, G.	Process simulation for software engineering education	2015
Gao, C.; Zhang, H.; Jiang, S.	Constructing hybrid software process simulation models	2015
Hu, Y.; Liu, K.; Zhang, X.; Su, L.; Ngai, E. W. T.; Liu, M.	Application of evolutionary computation for rule discovery in stock algorithmic trading: A literature review	2015
Parkkila, J.; Ikonen, J.; Porras, J.	Where is the research on connecting game worlds? A systematic mapping study	2015
Calderón, A.; Ruiz, M.	A systematic literature review on serious games evaluation: An application to software project management	2015
Hang, F.; Zhao, L.	Supporting end-user service composition: a systematic review of current activities and tools	2015
Van den Berghe, A.; Scandariato, R.; Yskout, K.; Joosen, W.	Design notations for secure software: a systematic literature review	2015
Bianchi, T.; Santos, D. S.; Felizardo, K. R.	Quality attributes of systems-of-systems: A systematic literature review	2015
Ilyas, M.; Khan, S. U.	Software integration in global software development: Success factors for GSD vendors	2015
Weerakkody, V.; Irani, Z.; Lee, H.; Osman, I.; Hindi, N.	E-government implementation: A bird's eye view of issues relating to costs, opportunities, benefits and risks	2015

Authors	Title	Year
Lacerda, T. C.; Nunes, J. V.; Von Wangenheim, C. G.	Usability heuristics for mobile phone applications: A literature review	2015
Ovcjak, B.; Hericko, M.; Polancic, G.	Factors impacting the acceptance of mobile data services - A systematic literature review	2015
Clear, T.; Daniels, M.; Beecham, S.; McDermott, R.; Barr, J.; Oudshoorn, M.; Savickaite, A.; Noll, J.	Challenges and recommendations for the design and conduct of global software engineering courses: A systematic review	2015
Alam, K. A.; Ahmad, R.; Akhunzada, A.; Nasir, M. H. N. M.; Khan, S. U.	Impact analysis and change propagation in service-oriented enterprises: A systematic review	2015
Börjesson, P.; Barendregt, W.; Eriksson, E.; Torgersson, O.	Designing technology for and with developmentally diverse children - A systematic literature review	2015
Woznowski, P.; Kaleshi, D.; Oikonomou, G.; Craddock, I.	Classification and suitability of sensing technologies for activity recognition	2015
Jabangwe, R.; Börstler, J.; Smite, D.; Wohlin, C.	Empirical evidence on the link between object-oriented measures and external quality attributes: A systematic literature review	2015
Sangupamba Mwilu, O.; Comyn-Wattiau, I.; Prat, N.	Design science research contribution to business intelligence in the cloud - A systematic literature review	2015
Kilubi, I.	Strategic technology partnering: A framework extension	2015
Hasan, M. M.	ICTD systems development: Analysis of requirements elicitation approaches	2015
Moghaddam, F. A.; Lago, P.; Grosso, P.	Energy-efficient networking solutions in cloud-based environments: A systematic literature review	2015
Siavashi, F.; Truscan, D.	Environment modeling in model-based testing: Concepts, prospects and research challenges: A systematic literature review	2015
Mahmood, S.; Anwer, S.; Niazi, M.; Alshayeb, M.; Richardson, I.	Identifying the factors that influence task allocation in global software development: Preliminary results	2015
Zhou, Y.; Zhang, H.; Huang, X.; Yang, S.; Babar, M. A.; Tang, H.	Quality assessment of systematic reviews in software engineering: A tertiary study	2015
Liu, G.; Rong, G.; Zhang, H.; Shan, Q.	The adoption of capture-recapture in software engineering: A systematic literature review	2015

Authors	Title	Year
Guessi, M.; Neto, V. V. G.; Bianchi, T.; Felizardo, K. R.; Oquendo, F.; Nakagawa, E. Y.	A systematic literature review on the description of software architectures for systems of systems	2015
Hainey, T.; Connolly, T. M.; Chaudy, Y.; Boyle, E.; Beeby, R.; Soflano, M.	Assessment integration in serious games	2015
Hussain, A.; Abubakar, H. I.; Hashim, N. B.	Evaluating mobile banking application: Usability dimensions and measurements	2015
Steinmacher, I.; Conte, T. U.; Gerosa, M. A.; Redmiles, D. F.	Social barriers faced by newcomers placing their first contribution in open source software projects	2015
Myllärniemi, V.; Savolainen, J.; Raatikainen, M.; Männistö, T.	Performance variability in software product lines: Proposing theories from a case study	2015
Dermeval, D.; Vilela, J.; Bittencourt, I. I.; Castro, J.; Isotani, S.; Brito, P.; Silva, A.	Applications of ontologies in requirements engineering: a systematic review of the literature	2015
Misbhauddin, M.; Alshayeb, M.	UML model refactoring: a systematic literature review	2015
Benslimane, Y.; Plaisent, M.; Bernard, P.; Bahli, B.	Key challenges and opportunities in cloud computing and implications on service requirements: Evidence from a systematic literature review	2015
Dobreva, M.; Angelova, G.; Agre, G.	Bridging the gap between digital libraries and e-learning	2015
Danciu, A.; Kroß, J.; Brunnert, A.; Willnecker, F.; Vögele, C.; Kapadia, A.; Krcmar, H.	Landscaping performance research at the ICPE and its predecessors: A systematic literature review	2015
Akour, M.; Alazzam, I.; Hanandeh, F.; Akour, I.	A systematic literature review to classify pre and post test suite reduction techniques	2015
Hydara, I.; Sultan, A. B. M.; Zulzalil, H.; Admodisastro, N.	Current state of research on cross-site scripting (XSS) - A systematic literature review	2015
Rashid, M.; Anwar, M. W.; Khan, A. M.	Toward the tools selection in model based system engineering for embedded systems - A systematic literature review	2015
Campanelli, A. S.; Parreiras, F. S.	Agile methods tailoring - A systematic literature review	2015
Sheikh Ibrahim, M.; Salleh, N.; Misra, S.	Empirical studies of cloud computing in education: A systematic literature review	2015
Lewis, G.; Lago, P.	Architectural tactics for cyber-foraging: Results of a systematic literature review	2015

Authors	Title	Year
Supulniece, I.; Berzisa, S.; Polaka, I.; Grabis, J.; Meiers, E.; Ozolins, E.	Source code driven decomposition of object-oriented legacy systems a systemic literature review and research outlook	2015
Dutra, A. C. S.; Prikladnicki, R.; Conte, T.	What are the main characteristics of high performance teams for software development?	2015
Soares, M. S.; Maia, M. A.; Silva, R. F. G.	Performance evaluation of aspect-oriented programming weavers	2015
de Almeida, M. M. K.; Marins, F. A. S.; Salgado, A. M. P.; Santos, F. C. A.; da Silva, S. L.	Mitigation of the bullwhip effect considering trust and collaboration in supply chain management: a literature review	2015
Bipat, S.; Sneller, L.	A theoretical examination of constructs underlying information technology capability	2015
Meis, R.; Wirtz, R.; Heisel, M.	A taxonomy of requirements for the privacy goal transparency	2015
Mirza, U. M.; Arslan, M. A.; Cedersjo, G.; Sulaman, S. M.; Janneck, J. W.	Mapping and scheduling of dataflow graphs - A systematic map	2015
De Aguiar Beninca, R.; Huzita, E. H. M.; Galdamez, E. V. C.; Leal, G. C. L.; Balancieri, R.; Massago, Y.	Knowledge management practices in GSD: A systematic literature review update	2015
Shaikh, M.; Salleh, N.; Marziana, L.	Social networks event mining: A systematic literature review	2015
Tosi, D.; Morasca, S.	Supporting the semi-automatic semantic annotation of web services: A systematic literature review	2015
Subbaraj, R.; Venkatraman, N.	A systematic literature review on ontology based context management system	2015
Rouhani, B. D.; Mahrin, M. N.; Nikpay, F.; Ahmad, R. B.; Nikfard, P.	A systematic literature review on Enterprise Architecture Implementation Methodologies	2015
Kupiainen, E.; Mäntylä , M. V.; Itkonen, J.	Using metrics in Agile and Lean software development - A systematic literature review of industrial studies	2015
Ha, I.	Technologies and research trends in wireless body area networks for healthcare: A systematic literature review	2015
Bakar, N. H.; Kasirun, Z. M.; Salleh, N.	Feature extraction approaches from natural language requirements for reuse in software product lines: A systematic literature review	2015
Ibukun, E.; Daramola, O.	A systematic literature review of mobile cloud computing	2015

Authors	Title	Year
Rizvi, B.; Bagheri, E.; Gasevic, D.	A systematic review of distributed Agile software engineering	2015
Fuentes, C.; Gerea, C.; Herskovic, V.; Marques, M.; Rodríguez, I.; Rossel, P. O.	User interfaces for self-reporting emotions: A systematic literature review	2015
Rosmadi, N. A.; Ahmad, S.; Abdullah, N.	The relevance of software requirement defect management to improve requirements and product quality: A systematic literature review	2015
Sa'Don, N. F.; Alias, R. A.; Ohshima, N.	Nascent research trends in MOOCs in higher educational institutions: A systematic literature review	2015
Riegel, N.; Doerr, J.	A systematic literature review of requirements prioritization criteria	2015
Ming, T. M.; Jabar, M. A.; Sidi, F.; Wei, K. T.	A systematic literature review of computer ethics issues	2015
Procaccianti, G.; Lago, P.; Bevini, S.	A systematic literature review on energy efficiency in cloud software architectures	2015
Malinen, S.	Understanding user participation in online communities: A systematic literature review of empirical studies	2015
Väänänen-Vainio-Mattila, K.; Olsson, T.; Häkkilä, J.	Towards deeper understanding of user experience with ubiquitous computing systems: Systematic literature review and design framework	2015
Zarour, M.; Abran, A.; Desharnais, J. -.; Alarifi, A.	An investigation into the best practices for the successful design and implementation of lightweight software process assessment methods: A systematic literature review	2015
Da Silva Estácio, B. J.; Prikladnicki, R.	Distributed pair programming: A systematic literature review	2015
Gonçalves , R. Q.; Von Wangenheim, C. G.	How to teach the usage of project management tools in computer courses: A systematic literature review	2015
Lenberg, P.; Feldt, R.; Wallgren, L. G.	Behavioral software engineering: A definition and systematic literature review	2015
Ahmed, M. M.; Letchmunan, S.	A systematic literature review on challenges in service oriented software engineering	2015
Noraini, C. P.; Bokolo, A. J.; Haizan Nor, R. N.; Azmi Murad, M. A.	Risk assessment of it governance: A systematic literature review	2015
Vacari, I.; Prikladnicki, R.	Adopting agile methods in the public sector: A systematic literature review	2015
Razavian, M.; Lago, P.	A systematic literature review on SOA migration	2015

Authors	Title	Year
Ayora, C.; Torres, V.; Weber, B.; Reichert, M.; Pelechano, V.	VIVACE: A framework for the systematic evaluation of variability support in process-aware information systems	2015
Abdellatif, T. M.; Capretz, L. F.; Ho, D.	Software analytics to software practice: A systematic literature review	2015
Tüzün, E.; Tekinerdogan, B.; Kalender, M. E.; Bilgen, S.	Empirical evaluation of a decision support model for adopting software product line engineering	2015
Al Dallal, J.	Identifying refactoring opportunities in object-oriented code: A systematic literature review	2015
Rochimah, S.; Arifiani, S.; Insanittaqwa, V. F.	Non-source code refactoring: A systematic literature review	2015
Sharafi, Z.; Soh, Z.; Guéhéneuc, Y. -.	A systematic literature review on the usage of eye-tracking in software engineering	2015
Dos Santos Rocha, R.; Degrossi, L. C.; De Albuquerque, J. P.	A systematic literature review of geospatial web service composition	2015
Ulziit, B.; Warraich, Z. A.; Gencel, C.; Petersen, K.	A conceptual framework of challenges and solutions for managing global software maintenance	2015
Santos, J. A. M.; Santos, A. R.; De Mendonça, M. G.	Investigating bias in the search phase of Software Engineering secondary studies	2015
Nakic, J.; Granic, A.; Glavinic, V.	Anatomy of student models in adaptive learning systems: A systematic literature review of individual differences from 2001 to 2013	2015
Chuene, D.; Mtsweni, J.	The adoption of crowdsourcing platforms in South Africa	2015
Kocbek, M.; Jošt, G.; Hericko, M.; Polancic, G.	Business process model and notation: The current state of affairs	2015
Ghanbarzadeh, R.; Ghapanchi, A. H.; Blumenstein, M.	Characteristics of research on the application of three-dimensional immersive virtual worlds in health	2015
Al-Wosabi, A. A. A.; Shukur, Z.	Software tampering detection in embedded systems - A systematic literature review	2015
Hyrynsalmi, S.; Seppänen, M.; Nokkala, T.; Suominen, A.; Järvi, A.	Wealthy, healthy and/or happy — What does 'Ecosystem Health' stand for?	2015
Cohn-Muroy, D.; Pow-Sang, J. A.	Can user stories and use cases be used in combination in a same project? A systematic review	2015
Niazi, M.; Mahmood, S.; Alshayeb, M.; Hroub, A.	Empirical investigation of the challenges of the existing tools used in global software development projects	2015

Authors	Title	Year
Wagner, I.; Eckhoff, D.	Privacy assessment in vehicular networks using simulation	2015
Franco-Bedoya, O.; Ameller, D.; Costal, D.; Franch, X.	Measuring the quality of open source software ecosystems using QuESo	2015
Eze, B.; Peyton, L.	Systematic literature review on the anonymization of high dimensional streaming datasets for health data sharing	2015
Khosravi, P.; Ghapanchi, A. H.; Blumenstein, M.	Investigating various technologies applied to assist seniors	2015
El Ouirdi, A.; El Ouirdi, M.; Segers, J.; Henderickx, E.	Employees' use of social media technologies: A methodological and thematic review	2015
Jamil, N. B. C. E.; Ishak, I. B.; Sidi, F.; Affendey, L. S.; Mamat, A.	A systematic review on the profiling of digital news portal for big data veracity	2015
Anthony, B.; Pa, N. C.	A review on tools of risk mitigation for information technology management	2015
Abade, A.; Ferrari, F.; Lucrédio, D.	Testing M2T transformations: A systematic literature review	2015
Miramontes, J.; Muñoz, M.; Calvo-Manzano, J. A.; Corona, B.	Establishing the state of the art of frameworks, methods and methodologies focused on lightening software process: A systematic literature review	2015
Sadowska, M.	An approach to assessing the quality of business process models expressed in BPMN	2015
Friedrich, T.	Analyzing the factors that influence consumers' adoption of social commerce - A literature review	2015
Leyh, C.; Thomschke, J.	Critical success factors for implementing supply chain management systems - The perspective of selected German enterprises: Full paper	2015
Chatfield, A. T.; Reddick, C. G.; Al-Zubaidi, W. H. A.	Capability challenges in transforming government through open and big data: Tales of two cities	2015
Maranhão, R.; Marinho, M.; De Moura, H.	Narrowing impact factors for innovative software project management	2015
Medeiros, J.; Vasconcelos, A.; Silva, C.	Integration of agile practices: An approach to improve the quality of software specifications	2015
Dutra, A. C. S.; Prikladnicki, R.; Conte, T.	Characteristics of high performance software development teams	2015

Authors	Title	Year
Heaton, D.; Carver, J. C.	Claims about the use of software engineering practices in science: A systematic literature review	2015
Moreno-Montes De Oca, I.; Snoeck, M.; Reijers, H. A.; Rodríguez-Morffi, A.	A systematic literature review of studies on business process modeling quality	2015
Bano, M.; Zowghi, D.	A systematic review on the relationship between user involvement and system success	2015
Steinmacher, I.; Graciotto Silva, M. A.; Gerosa, M. A.; Redmiles, D. F.	A systematic literature review on the barriers faced by newcomers to open source software projects	2015
Nguyen-Duc, A.; Cruzes, D. S.; Conradi, R.	The impact of global dispersion on coordination, team performance and software quality-A systematic literature review	2015
Lee, T.; Ghapanchi, A. H.; Talaei-Khoei, A.; Ray, P.	Strategic information system planning in healthcare organizations	2015
Xu, Y.; Liu, Y.; Zheng, J.	To enlighten hidden facts in the code: A review of software visualization metaphors	2015
Holm, H.; Karresand, M.; Vidström, A.; Westring, E.	A survey of industrial control system testbeds	2015
Younoussi, S.; Roudies, O.	All about software reusability: A systematic literature review	2015
Pa, N. C.; Anthony, B.	A review on decision making of risk mitigation for software management	2015
Santhosh Kumar, R.; Prabu, M.; Venkatesh, P.; Vijaya Rani, S.	An automatic solar panel based street lighting system: A systematic literature review	2015
Méndez-Porras, A.; Quesada-López, C.; Jenkins, M.	Automated testing of mobile applications: A systematic map and review	2015
De Magalhães , C. V. C.; Da Silva, F. Q. B.; Santos, R. E. S.; Suassuna, M.	Investigations about replication of empirical studies in software engineering: A systematic mapping study	2015
Horvath, R.; Nedbal, D.; Stieninger, M.	A literature review on challenges and effects of software defined networking	2015
Saide; Mahendrawathi, E. R.	Knowledge management support for enterprise resource planning implementation	2015
Eyers, D. R.; Potter, A. T.	E-commerce channels for additive manufacturing: An exploratory study	2015
Abdelmabou, A.; Jawawi, D. N. A.; Ghani, I.; Elsafi, A.	A comparative evaluation of cloud migration optimization approaches: A systematic literature review	2015

Authors	Title	Year
Tüzün, E.; Tekinerdogan, B.	Analyzing impact of experience curve on ROI in the software product line adoption process	2015
Díaz, O.; Arellano, C.	The augmented web: Rationales, opportunities, and challenges on browser-side transcoding	2015
Petersen, K.; Vakkalanka, S.; Kuzniarz, L.	Guidelines for conducting systematic mapping studies in software engineering: An update	2015
Svahnberg, M.; Gorschek, T.; Nguyen, T. T. L.; Nguyen, M.	Uni-REPM: A framework for requirements engineering process assessment	2015
Seufert, S.; Stanoevska-Slabeva, K.; Müller, S.; Scheffler, N.	The design of personal learning environments (PLE) with scope on information literacy in high school	2015
Niazi, M.	A comparative study of software process improvement implementation success factors	2015
Ampatzoglou, A.; Ampatzoglou, A.; Chatzigeorgiou, A.; Avgeriou, P.	The financial aspect of managing technical debt: A systematic literature review	2015
Imam, A. A.; Basri, S.; Ahmad, R.	Data synchronization between mobile devices and server-side databases: A systematic literature review	2015
Cremona, L.; Ravarini, A.; Sutanto, J.	KMS in a cluster of firms: The role of a digital platform	2015
Bellisario, A.; Appolloni, A.; Ranalli, F.	Reviewing strategy matters to gain an understanding of balanced scorecard's possible benefits within lean production contexts: A management control perspective	2015
De Mello, R. M.; Travassos, G. H.	Characterizing sampling frames in software engineering surveys	2015
Akhigbe, O.; Amyot, D.; Richards, G.	Information technology artifacts in the regulatory compliance of business processes: A meta-analysis	2015
Lema, L.; Calvo-Manzano, J. -.; Colomo-Palacios, R.; Arcilla, M.	ITIL in small to medium-sized enterprises software companies: Towards an implementation sequence	2015
Ravarini, A.; Cremona, L.	Digital platoms as knowledge artifacts for clusters of SMEs	2015
Akkineni, H.; Lakshmi, P. V. S.; Babu, B. V.	Online crowds opinion-mining it to analyze current trend: A review	2015
Jaramillo Franco, A.	Requirements elicitation approaches: A systematic review	2015

Authors	Title	Year
Blignaut, A. S.	Infinite possibilities for using eye-tracking for mobile serious games in order to improve user learning experiences	2015
Theocharis, G.; Kuhrmann, M.; Münch, J.; Diebold, P.	Is water-scrum-fall reality? On the use of agile and traditional development practices	2015
Abedin, B.; Babar, A.; Abbasi, A.	Characterization of the use of social media in natural disasters: A systematic review	2015
Morente-Molinera, J. A.; Pérez, I. J.; Ureña , M. R.; Herrera-Viedma, E.	On multi-granular fuzzy linguistic modeling in group decision making problems: A systematic review and future trends	2015
Schenkel, M.; Caniëls, M. C. J.; Krikke, H.; Van Der Laan, E.	Understanding value creation in closed loop supply chains - Past findings and future directions	2015
Elbattah, M.; Molloy, O.	Towards improving modeling and simulation of clinical pathways: Lessons learned and future insights	2015
Nawi, H. S. A.; Ibrahim, O.; Shukor, N. S. A.; Omar, S. F.; Ishak, I. S.; Rahman, A. A.	Understanding sustainability: An exploration of the is literature	2015
Matalonga, S.; Rodrigues, F.; Travassos, G. H.	Matching context aware software testing design techniques to ISO/IEC/IEEE 29119	2015
Koç, H.	Methods in designing and developing capabilities: A systematic mapping study	2015
Rahim, N. H. A.; Hamid, S.; Kiah, L. M.; Shamshirband, S.; Furnell, S.	A systematic review of approaches to assessing cybersecurity awareness	2015
Pelone, F.; Kringos, D. S.; Romaniello, A.; Archibugi, M.; Salsiri, C.; Ricciardi, W.	Primary care efficiency measurement using data envelopment analysis: a systematic review	2015
Azarm-Daigle, M.; Kuziemsky, C.; Peyton, L.	A review of cross organizational healthcare data sharing	2015
Shawky, D.; Badawi, A.; Said, T.; Hozayin, R.	Affordances of computer-supported collaborative learning platforms: A systematic review	2015
Cruz, S.; Da Silva, F. Q. B.; Capretz, L. F.	Forty years of research on personality in software engineering: A mapping study	2015
Nye, B. D.	Intelligent tutoring systems by and for the developing world: A review of trends and approaches for educational technology in a global context	2015

Authors	Title	Year
Carroll, N.; Helfert, M.	Service capabilities within open innovation: Revisiting the applicability of capability maturity models	2015
Malhotra, R.	A systematic review of machine learning techniques for software fault prediction	2015
Selleri Silva, F.; Soares, F. S. F.; Peres, A. L.; Azevedo, I. M. D.; Vasconcelos, A. P. L. F.; Kamei, F. K.; Meira, S. R. D. L.	Using CMMI together with agile software development: A systematic review	2015
Tiwari, S.; Gupta, A.	A systematic literature review of use case specifications research	2015
Panahifar, F.; Heavey, C.; Byrne, P. J.; Fazlollahtabar, H.	A framework for Collaborative Planning, Forecasting and Replenishment (CPFR): State of the Art	2015
Idri, A.; Amazal, F. A.; Abran, A.	Analogy-based software development effort estimation: A systematic mapping and review	2015
Nguyen, P. H.; Kramer, M.; Klein, J.; Traon, Y. L.	An extensive systematic review on the Model-Driven Development of secure systems	2015
Brhel, M.; Meth, H.; Maedche, A.; Werder, K.	Exploring principles of user-centered agile software development: A literature review	2015
Mohanarajah, S.; Jabar, M. A.	An improved adaptive and dynamic hybrid agile methodology to enhance software project success deliveries	2015

References

Abdelmaboud A, Jawawi DN, Ghani I, Elsafi A (2015) A comparative evaluation of cloud migration optimization approaches: a systematic literature review. J Theor Appl Inf Technol 79(3):395–414

Afzal W, Alone S, Glocksien K, Torkar R (2016) Software test process improvement approaches: a systematic literature review and an industrial case study. J Syst Softw 111:1–33. https://doi.org/10.1016/j.jss.2015.08.048

Babar MA, Zhang H (2009) Systematic Literature Reviews in Software Engineering: Preliminary Results from Interviews with Researchers. In: International Symposium on Empirical Software Engineering and Measurement. IEEE, pp 346–355. https://doi.org/10.1109/ESEM.2009.5314235

Badampudi D, Wohlin C, Petersen K (2015) Experiences from Using Snowballing and Database Searches in Systematic Literature Studies. In: International Conference on Evaluation and Assessment in Software Engineering. ACM, pp 17:1–17:10. https://doi.org/10.1145/2745802.2745818

Bailey J, Zhang C, Budgen D, Turner M, Charters S (2007) Search Engine Overlaps: Do They Agree or Disagree? In: International Workshop on Realising Evidence-Based Software Engineering. IEEE, pp 2–2. https://doi.org/10.1109/REBSE.2007.4

Bandara W, Miskon S, Fielt E (2011) A systematic Tool-Supported method for conducting literature reviews in information systems. In: European Conference on Information Systems

Biolchini J, Mian PG, Natali ACC, Travassos GH (2005) Systematic Review in Software Engineering. Technical Report. RT - ES 679 / 05, Federal University of Rio de Janeiro

Boell SK, Cecez-Kecmanovic D (2015) Debating Systematic Literature Reviews (SLR) and their Ramifications for IS: A Rejoinder to Mike Chiasson, Briony Oates, Ulrike Schultze, and Richard Watson. J Inf Technol 30(2):188–193. https://doi.org/10.1057/jit.2015.15

Booth A, Sutton A, Papaioannou D (2012) Systematic Approaches to a Successful Literature Review. Sage, Newbury Park

Brereton OP, Kitchenham BA, Budgen D, Turner M, Khalil M (2007) Lessons from applying the systematic literature review process within the software engineering domain. J Syst Softw 80(4):571–583. https://doi.org/10.1016/j.jss.2006.07.009

Budgen D, Brereton OP (2006) Performing Systematic Literature Reviews in Software Engineering. In: International Conference on Software Engineering. ACM, pp 1051–1052. https://doi.org/10.1145/1134285.1134500

Budgen D, Brereton OP, Drummond S, Williams N (2018a) Reporting systematic reviews: Some lessons from a tertiary study. Inf Softw Technol 95:62–74. https://doi.org/10.1016/j.infsof.2017.10.017

Budgen D, Brereton OP, Williams N, Drummond S (2018b) The Contribution that Empirical Studies Performed in Industry make to the Findings of Systematic Reviews: A Tertiary Study. Inf Softw Technol 94:234–244. https://doi.org/10.1016/j.infsof.2017.10.012

Carver JC (2010) Towards reporting guidelines for experimental replications: a proposal. In: International Workshop on Replication in Empirical Software Engineering

Collins JA, Fauser BC (2005) Balancing the strengths of systematic and narrative reviews. Hum Reprod Update 11(2):103–104. https://doi.org/10.1093/humupd/dmh058

da Silva FQ, Santos AL, Soares S, França ACC, Monteiro CV, Maciel FF (2011) Six years of systematic literature reviews in software engineering: an updated tertiary study. Inf Softw Technol 53(9):899–913. https://doi.org/10.1016/j.infsof.2011.04.004

Dresch A, Lacerda DP, Antunes JAV Jr (2014) Design Science Research: A Method for Science and Technology Advancement. Springer, Berlin. https://doi.org/10.1007/978-3-319-07374-3

Dybå T, Dingsøyr T (2008) Strength of Evidence in Systematic Reviews in Software Engineering. In: International Symposium on Empirical Software Engineering and Measurement. ACM, pp 178–187, https://doi.org/10.1145/1414004.1414034

Dybå T, Kitchenham BA, Jørgensen M (2005) Evidence-Based Software engineering for practitioners. IEEE Softw 22(1):58–65. https://doi.org/10.1109/MS.2005.6

Dybå T, Dingsøyr T, Hanssen GK (2007) Applying Systematic Reviews to Diverse Study Types: An Experience Report. In: International Symposium on Empirical Software Engineering and Measurement. IEEE, pp 225–234, https://doi.org/10.1109/ESEM.2007.59

Falagas ME, Pitsouni EI, Malietzis GA, Pappas G (2008) Comparison of PubMed, Scopus, Web of Science, and Google Scholar: Strengths and Weaknesses. FASEB J 22(2):338–342. https://doi.org/10.1096/fj.07-9492LSF

Felizardo KR, Mendes E, Kalinowski M, Souza EF, Vijaykumar NL (2016) Using Forward Snowballing to Update Systematic Reviews in Software Engineering. In: International Symposium on Empirical Software Engineering and Measurement. ACM, pp 53:1–53:6. https://doi.org/10.1145/2961111.2962630

Fuhr N, Tsakonas G, Aalberg T, Agosti M, Hansen P, Kapidakis S, Klas CP, Kovács L, Landoni M, Micsik A, Papatheodorou C, Peters C, Sølvberg I (2007) Evaluation of Digital Libraries. Int J Digit Libr 8(1):21–38. https://doi.org/10.1007/s00799-007-0011-z

Galvan JL, Galvan MC (2016) Writing Literature Reviews: A Guide for Students of the Social and Behavioral Sciences. Routledge, Evanston

Giles CL (2006) The Future of Citeseer: Citeseer^{X}. In: European Conference on Principle and Practice of Knowledge Discovery in Databases. Springer, pp 2–2, https://doi.org/10.1007/11871637_2

Gómez OS, Juristo N, Vegas S (2014) Understanding replication of experiments in software engineering: a classification. Inf Softw Technol 56(8):1033–1048

Harzing AW, Alakangas S (2017) Microsoft Academic: Is the Phoenix Getting Wings? Scientometrics 110(1):371–383. https://doi.org/10.1007/s11192-016-2185-x

Hassler E, Carver JC, Hale D, Al-Zubidy A (2016) Identification of SLR tool needs – results of a community workshop. Inf Softw Technol 70:122–129. https://doi.org/10.1016/j.infsof.2015.10.011

Ilyas M, Khan SU (2015) Software integration in global software development: Success factors for GSD vendors. In: International conference on software engineering, artificial intelligence, Networking and Parallel/Distributed Computing. IEEE, pp 1–6. https://doi.org/10.1109/SNPD.2015.7176187

Imtiaz S, Bano M, Ikram N, Niazi M (2013) A Tertiary Study: Experiences of Conducting Systematic Literature Reviews in Software Engineering. In: International Conference on Evaluation and Assessment in Software Engineering. ACM, pp 177–182. https://doi.org/10.1145/2460999.2461025

Jalali S, Wohlin C (2012) Systematic Literature Studies: Database Searches vs. Backward Snowballing. In: International Symposium on Empirical Software Engineering and Measurement. ACM, pp 29–38. https://doi.org/10.1145/2372251.2372257

Khan KS, Kunz R, Kleijnen J, Antes G (2003) Five steps to conducting a systematic review. J R Soc Med 96(3):118–121. https://doi.org/10.1177/014107680309600304

Kitchenham BA, Brereton OP (2013) A systematic review of systematic review process research in software engineering. Inf Softw Technol 55(12):2049–2075. https://doi.org/10.1016/j.infsof.2013.07.010

Kitchenham BA, Charters S (2007) Guidelines for performing systematic literature reviews in software engineering. Tech. Rep EBSE-2007-01. Keele University, University of Durham

Kitchenham BA, Brereton OP, Budgen D, Turner M, Bailey J, Linkman S (2009) Systematic literature reviews in software engineering - a systematic literature review. Inf Softw Technol 51(1):7–15. https://doi.org/10.1016/j.infsof.2008.09.009

Kitchenham BA, Pretorius R, Budgen D, Brereton OP, Turner M, Niazi M, Linkman S (2010) Systematic literature reviews in software engineering – a tertiary study. Inf Softw Technol 52(8):792–805. https://doi.org/10.1016/j.infsof.2010.03.006

Kitchenham BA, Brereton OP, Li Z, Budgen D, Burn A (2011) Repeatability of Systematic Literature Reviews. In: International Conference on Evaluation and Assessment in Software Engineering. IET, pp 46–55

Kitchenham BA, Budgen D, Brereton OP (2016) Evidence-Based Software Engineering and Systematic Reviews. CRC Press, Boca Raton

Levy Y, Ellis TJ (2006) A systems approach to conduct an effective literature review in support of information systems research. Inform Sci Int J Emerging Transdiscip 9:181–212. https://doi.org/10.28945/479

Lex A, Gehlenborg N, Strobelt H, Vuillemot R, Pfister H (2014) Upset: Visualization of Intersecting Sets. IEEE Trans Vis Comput Graph 20(12):1983–1992. https://doi.org/10.1109/TVCG.2014.2346248

Lindsay RM, Ehrenberg ASC (1993) The design of replicated studies. Amer Stat 47(3):217–228. https://doi.org/10.1080/00031305.1993.10475983

MacDonell S, Shepperd M, Kitchenham BA, Mendes E (2010) How Reliable Are Systematic Reviews in Empirical Software Engineering? IEEE Trans Softw Eng 36(5):676–687. https://doi.org/10.1109/TSE.2010.28

Mbiydzenyuy G (2015) Arrival Times with Hours of Service Regulations for Truck Drivers-Tracks and Gaps from Current Research. In: International Conference on Intelligent Transportation Systems. IEEE, pp 2631–2636. https://doi.org/10.1109/ITSC.2015.423

Moher D, Liberati A, Tetzlaff J, Altman DG, Group TP (2009) Preferred reporting items for systematic reviews and Meta-Analyses: The PRISMA statement. PLOS Med 6(7):1–6. https://doi.org/10.1371/journal.pmed.1000097

Okoli C, Schabram K (2010) A Guide to Conducting a Systematic Literature Review of Information Systems Research. Sprouts Working Papers on Information Systems. https://doi.org/10.2139/ssrn.1954824

Orduña-Malea E, Martín-Martín A, Ayllon JM, Delgado Lopez-Cozar E (2014) The Silent Fading of an Academic Search Engine: The Case of Microsoft Academic Search. Online Inf Rev 38(7):936–953. https://doi.org/10.1108/OIR-07-2014-0169

Petersen K, Feldt R, Mujtaba S, Mattsson M (2008) Systematic Mapping Studies in Software Engineering. In: International Conference on Evaluation and Assessment in Software Engineering. BCS, pp 68–77

Riaz M, Sulayman M, Salleh N, Mendes E (2010) Experiences Conducting Systematic Reviews from Novices' Perspective. In: International Conference on Evaluation and Assessment in Software Engineering. BCS, pp 44–53

Riedl R (2007) On the Replication of Positivist Case Study Research. In: European Conference on Information Systems, pp 1515–1526

Sackett DL, Straus SE, Richardson WS, Rosenberg W, Haynes RB (1997) Evidence-Based Medicine: How to Practice and Teach EBM. WB Saunders Company, Philadelphia

Shakeel Y, Krüger J, von Nostitz-Wallwitz I, Lausberger C, Campero Durand G, Saake G, Leich T (2018) (Automated) Literature Analysis - Threats and Experiences. In: International Workshop on Software Engineering for Science. ACM, pp 20–27. https://doi.org/10.1145/3194747.3194748

Shaw M (2003) Writing Good Software Engineering Research Papers. In: International Conference on Software Engineering. IEEE, pp 726–736. https://doi.org/10.1109/ICSE.2003.1201262

Soni G, Kodali R (2011) A critical analysis of supply chain management content in empirical research. Bus Process Manag J 17(2):238–266. https://doi.org/10.1108/14637151111122338

Tranfield D, Denyer D, Smart P (2003) Towards a methodology for developing Evidence-Informed management knowledge by means of systematic review. Br J Manag 14(3):207–222. https://doi.org/10.1111/1467-8551.00375 vom Brocke J, Simons A, Riemer K, Niehaves B, Plattfaut R, Cleven A (2015) Standing on the shoulders of giants: Challenges and recommendations of literature search in information systems research. Commun Assoc Inf Syst 37(9):205–224

von Nostitz-Wallwitz I, Krüger J, Siegmund J, Leich T (2018) Knowledge Transfer from Research to Industry: A Survey on Program Comprehension. In: International Conference on Software Engineering. ACM, pp 300–301. https://doi.org/10.1145/3183440.3194980

Webster J, Watson RT (2002) Analyzing the Past to Prepare for the Future: Writing a Literature Review. Manag Inf Syst Q 26(2):xiii–xxiii

Wohlin C (2014) Guidelines for Snowballing in Systematic Literature Studies and a Replication in Software Engineering. In: International Conference on Evaluation and Assessment in Software Engineering. ACM, pp 1–10. https://doi.org/10.1145/2601248.2601268

Wolfswinkel JF, Furtmueller E, Wilderom CP (2013) Using grounded theory as a method for rigorously reviewing literature. Eur J Inf Syst 22(1):45–55. https://doi.org/10.1057/ejis.2011.51

Yin RK (2018) Case Study Research and Applications: Design and Methods. Sage, Newbury Park

Zhang H, Babar MA (2010) On Searching Relevant Studies in Software Engineering. In: International Conference on Evaluation and Assessment in Software Engineering. BCS Learning & Development Ltd., pp 111–120

Zhang H, Babar MA, Tell P (2011) Identifying relevant studies in software engineering. Inf Softw Technol 53(6):625–637. https://doi.org/10.1016/j.infsof.2010.12.010

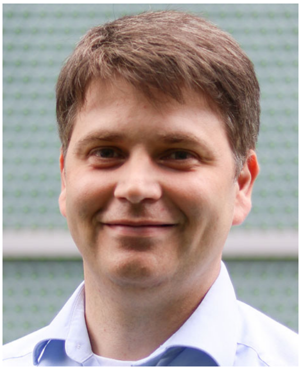
Zhang H, Babar MA (2013) Systematic reviews in software engineering: an empirical investigation. Inf Softw Technol 55(7):1341–1354. https://doi.org/10.1016/j.infsof.2012.09.008

Jacob Krüger is a PhD student and associated researcher at the Databases and Software Engineering group of the Otto-von-Guericke University of Magdeburg. He received his M.Sc. degree in Business Informatics at the University of Magdeburg in 2016, has been working as research associate at the Harz University of Applied Sciences Wernigerode, and visited Chalmers University of Technology | University of Gothenburg in Sweden. His research focuses on feature-oriented software development, with particular interests on software evolution, program comprehension, and human factors.

Christian Lausberger is a software developer at METOP GmbH in Magdeburg, an affiliated institute to the Otto-von-Guericke University of Magdeburg. He received his B.Sc. and M.Sc. degrees in Computer Science at the Otto-von-Guericke University of Magdeburg.

Ivonne von Nostitz-Wallwitz is a software developer at METOP GmbH. She received her Diploma in Computer Science at the Otto-von-Guericke-University of Magdeburg in September 2012. Since 2016, she is an external PhD student at the Otto-von-Guericke-University of Magdeburg. Her research focuses on program comprehension, empirical studies, and knowledge transfer.

Gunter Saake received his diploma and PhD in Computer Science from the Technical University of Braunschweig, F.R.G., in 1985 and 1988, respectively. From 1988 to 1989, he was a visiting scientist at the IBM Heidelberg Scientific Center, where he joined the Advanced Information Management project and worked on language features and algorithms for sorting and duplicate elimination in nested relational database structures. In January 1993, he received the Habilitation degree (venia legendi) for Computer Science from the Technical University of Braunschweig. Since May 1994, Gunter Saake is a fulltime professor for Databases and Information Systems at the Otto-von-Guericke University, Magdeburg. His research interests include database integration, tailor-made data management, object-oriented information systems, and information fusion.



Thomas Leich is Professor for Requirements Engineering at Harz University of Applied Sciences in Wernigerode, Germany. He is also executive director of METOP GmbH, an affiliate institute to the University of Magdeburg. Since 2001, he worked for several DAX 30 companies as consultant and software architect. In 2004, he initiated FeatureIDE as a part of the FeatureC++ project at the University of Magdeburg. Until today, he is responsible for industrial extensions and consulting of FeatureIDE.

Affiliations

Jacob Krüger[1,2] · **Christian Lausberger**[1,3] · **Ivonne von Nostitz-Wallwitz**[1,3] · **Gunter Saake**[1] · **Thomas Leich**[2,3]

Christian Lausberger
Christian.Lausberger@metop.de

Ivonne von Nostitz-Wallwitz
Ivonne.Nostitz@metop.de

Gunter Saake
saake@ovgu.de

Thomas Leich
tleich@hs-harz.de

1 Otto-von-Guericke University, Magdeburg, Germany

2 Harz University of Applied Sciences, Wernigerode, Germany

3 METOP GmbH, Magdeburg, Germany