Guidelines for conducting interactive rapid reviews in software engineering -- from a focus on technology transfer to knowledge exchange

Rico, Sergio; Ali, Nauman Bin; Engström, Emelie; Höst, Martin

DOI:
10.5281/zenodo.4327725

2020

Guidelines for conducting interactive rapid reviews in software engineering – from a focus on technology transfer to knowledge exchange

Version 1.0

Sergio Rico[1 [0000−0002−9348−2912]], Nauman Bin Ali[2 [0000−0001−7266−5632]], Emelie Engström[1 [0000−0001−6736−9425]], and Martin Höst[1 [0000−0002−9360−8693]]

[1] Department Computer Science, Lund University, Lund, Sweden
(sergio.rico, emelie.engstrom, martin.host)@cs.lth.se
http://serg.cs.lth.se/

[2] Department of Software Engineering, Blekinge Institute of Technology, Sweden
nauman.ali@bth.se

Abstract. Evidence-based software engineering (EBSE) aims to improve research utilization in practice. It relies on systematic methods (like systematic literature reviews, systematic mapping studies, and rapid reviews) to identify, appraise, and synthesize existing research findings to answer questions of interest. However, the lack of practitioners' involvement in the design, execution, and reporting of these methods indicates a lack of appreciation for knowledge exchange between researchers and practitioners. Within EBSE, the main reason for conducting these systematic studies is to answer the practitioner's questions and impact practice. However, in many cases, academics have undertaken these studies without any direct involvement of practitioners. This report focuses on the rapid review guidelines and presents practical advice on conducting these with practitioner involvement to facilitate knowledge co-creation. Based on a literature review of rapid reviews and stakeholders engagement in medicine and our experience of using secondary studies in software engineering, we propose extensions to an existing proposal for rapid reviews in software engineering to increase researchers-practitioners knowledge exchange. We refer to the extended method as an interactive rapid review. An interactive rapid review is a streamlined approach to conduct agile literature reviews in close collaboration between researchers and practitioners in software engineering. This report describes the process and discusses possible usage scenarios and some reflections from the proposal's ongoing evaluation. The proposed guidelines will potentially boost knowledge co-creation through active researcher-practitioner interaction by streamlining practitioners' involvement and recognizing the need for an agile process.

Keywords: industry-academia communication · rapid reviews · software engineering

1 Introduction

Software engineering research aims to establish software development practice on scientific foundations. This ambition requires that research is relevant and accessible for practice. Evidence-based software engineering (EBSE) is one such initiative to provide the best available evidence to support software development and maintenance. Often, a single empirical study provides insufficient confidence in the strength of evidence. There is a need to synthesize available research (where individual studies often have contradictory results) on a topic of interest. The EBSE) [31] approach has the following five steps: (1) convert a practical information need to an answerable question, (2) identify available evidence to help answer the question, (3) critically appraise the evidence, (4) make evidence-informed decisions, and (5) evaluate the effectiveness and efficiency of steps 1-4.

The EBSE community has developed several systematic secondary study methods for steps 2-3, including systematic literature reviews (SLRs) [32], systematic mapping studies (SMS) [43], and rapid reviews (RRs) [11]. Similarly, several authors have proposed solutions to facilitate step 4 in the EBSE process by introducing knowledge translation [7] or the technology transfer models [38].

Among the secondary study methods, mainly RR and SLRs are intended to support changes in practice. The SMSs only develop an overview of existing research on a topic. They are not intended to provide actionable insights for practice. SLRs risk being less attractive for practitioners because of the time frame needed to complete them. The time limitation of SLRs is overcome with the use of RRs. RRs are a variant of SLRs that simplify several steps of SLRs to provide information under time restrictions.

However, secondary studies are often conducted without any participation of practitioners. This lack of involvement can be partly explained by the implied objectivist view of knowledge [26] in the five-step EBSE process. In steps 2-3, knowledge is treated as objective, disembodied from the context, and codified, which in step 4 is transferred or communicated to practice. We overcome this limitation by extending the guidelines for RRs guided by the following principles: 1) *Prioritize exchange* between researchers and practitioners. 2) The review is conducted to be *relevant for practitioners* according to their context. 3) A *close collaboration* is expected while doing the review.

This report presents an extension to the existing guidelines for designing and conducting RRs in SE [11]. It includes an emphasis on iterative and flexible design and ways to increase practitioner involvement in RRs, we refer to this extended version as interactive rapid review (IRR).

Like agile software development, IRR aims to bring the stakeholders (practitioners and researchers) of the product (in this case, literature syntheses) closer together with shorter lead times, increased communication, and flexibility in the process. The iterative and flexible design recognizes that the information need will be refined and may change during an IRR. Similarly, the interaction is critical to developing a deeper understanding of the context where practical information need is situated and to improve the relevance of the results.

The extension is based on a review of the literature from evidence-based medicine (EBM) where rapid reviews are extensively used [28, 30, 39, 50]. We further supplement these with our own experience of having conducted several SLRs targeting industrial needs (e.g., [1,3,15,16]) and several industry-academia collaboration projects.

We envision that conducting an IRR based on the proposed guidelines may foster knowledge co-creation, bringing several benefits. The IRR results tailored for the practitioners' needs, improve research utilization in practice. Besides, conducting the IRR favors mutual understanding between practitioners and academics that paves the way for further collaboration.

The remainder of the report is structured as follows: we describe the related work and our approach for developing the IRR guidelines in Section 2. In Section 3, we describe the complete proposed guidelines for interactive rapid reviews. We further discuss the use and implication of IRRs in Section 4 and conclude the report in Section 5.

2 Background

2.1 Secondary studies in software engineering

Researchers in software engineering have widely adopted the use of secondary studies as a means to synthesize software engineering knowledge [5]. A large number of SLRs and SMS have been published in software engineering. Also the process itself, to conduct these secondary studies, has been a research topic, and some researchers have proposed improvements to the methods and new strategies. Some examples are snowballing as a search strategy [57], reporting guidelines for search [4], study selection procedures [2, 42], use of machine learning for automation of search and selection [46], and studies about when to update SLRs [37].

Recently, Felizardo et al. [19] published a systematic mapping study and a survey on the value of using secondary studies in software engineering. They observed that secondary studies mainly have been used in academic environments, for teaching purposes and to identify gaps in research. The value of conducting the studies is described in terms of ability to develop research skills in students and junior researchers and to provide insights to plan future research. Little is mentioned about the interaction with practitioners while conducting the studies or about the impact of secondary studies in industry.

Some voices in the software engineering research community have claimed that secondary studies need to connect more with practice. Budgen et al. [6] suggested aspects to improve when reporting systematic reviews to make the results more meaningful for teachers and practitioners. Le Goues et al. [34] reflected on the advantage to connect research evidence with recommendations for practitioners.

2.2 Rapid reviews in software engineering

Rapid reviews were introduced in software engineering by Cartaxo et al. with the primary goal to transfer knowledge from academia to industry [8–10]. Like previously introduced EBSE methods the rapid review term originates from evidence-based medicine. Cartaxo et al. [11] describe rapid reviews as secondary studies that aim to "provide evidence to support decision-making towards the solution, or at least attenuation, of issues practitioners face in practice". The reviews may be seen as a variation of systematic literature reviews where some steps are omitted or simplified to reduce completion time. In medicine, there are variations of the method to conduct a rapid review, however, the approaches share the following common aspects:

- The review is conducted in collaboration with practitioners and refers to practical problems in their context.
- The review is conducted in a short time and at a low cost.
- The review's results are "reported through mediums appealing to practitioners."

Rapid reviews should not be misunderstood as ad-hoc literature reviews or lax reviews. Instead, rapid reviews are a systematic approach with a transparently documented process. Cartaxo et al. propose rapid reviews in software engineering to be lightweight secondary studies to deliver evidence to practitioners in a short time to support decision making [11].

Rapid Reviews have two characteristics that make them a good candidate for connecting research and practice. First, they are conducted in a short period of time, which is probably appreciated by practitioners. Second, the studies are framed in the context of practitioners making the results relevant for them. This report elaborates on the researcher-practitioner interaction in such studies and describes the procedure for conducting interactive rapid reviews (IRRs).

2.3 Stakeholder engagement in secondary studies

In EBM, rapid reviews are used to support policy decision [30, 40, 41, 53], support decision-making under tight schedule restrictions [25, 44, 49, 52, 53] and to a lesser extent to identify areas for further research [39]. Deverka et al. [14] investigated the engagement of stakeholders in secondary studies, and concluded that stakeholder engagement contributes to developing a shared understanding of the knowledge and increasing the outcomes' relevance. In their study stakeholder refers to any person or organization with a direct interest in the secondary studies' process or outcomes and stakeholder engagement as "an iterative process of actively soliciting the knowledge, experience, judgment, and values of individuals selected to represent a broad range of direct interests in a particular issue". In 2017, the world health organization (WHO) published a guide about rapid reviews to strengthen health policy [51]. The guide was compiled by researchers and provide practical advice regarding various aspects of rapid reviews. Among other things, the guide addresses how to engage policymakers and health system managers in conducting rapid reviews.

3 Interactive Rapid Reviews

In this section we describe the preliminary steps for conducting an IRR and propose ways for researchers and practitioners to interact throughout the process. We base the proposal on a literature review of the use of rapid reviews in EBM, including 48 meta-studies and reflections on the method. The presented procedure is aligned with the one proposed by Cartaxo et al. [11] and reflects our own experiences of conducting interactive literature reviews [1]. Fig. 1 shows the activity flow to conduct the review.

Our proposal for IRR consists of five steps that are described in more detail later in this Section. The first step is to prepare the IRR and identify information needs based on a practical problem. In the second step, the research questions are identified, and an initial version of the IRR protocol is developed. The protocol keeps track of decisions and activities throughout the IRR. The third step consists of searching and selecting papers to find a limited set of papers to answer the research questions. Decisions about terminology and relevance are validated with practitioners. Based on the selected set of papers, the IRR report and dissemination documents are co-designed and developed during the fourth step. Finally, in the fifth step, the results are disseminated among the practitioners. Notice in Fig. 1. that the steps are conducted interactively with practitioners and that the general flow is iterative, where according to the feedback, the step outcomes are refined.

Table. 2. shows the central steps of an IRR in the first column (these have been adapted from Cartaxo et al. [11]). The second column highlights the contribution of our proposal with activities suggested to promote interaction with practitioners, and the third column lists the outcomes for each step. In the rest of this section we discuss each of these steps and possible interaction in more detail. Note that, when conducting an IRR the following general aspects should be considered:

- An IRR can be conducted in many scenarios throughout the researcher-practitioner relationship. The main goal of this type of review is not to publish a research paper, but to align communication between stakeholders and gain relevant knowledge to solve a practical problem.
- An IRR is preferably lead by researchers as they have more experience dealing with the scientific literature. Practitioners provide insights to keep the IRR relevant for practice with a consideration of their context.
- Conducting an IRR is an agile process. Similar to agile software development, our proposal for IRR embraces the following principles: smooth communication between researchers and practitioners; meaningful results in context; joint work with practitioners; and response to change and flexibility.

3.1 Prepare the review

Fig. 2. shows the activities to prepare the review. In this step, the review team is formed, and information need is identified and described in context. The in-

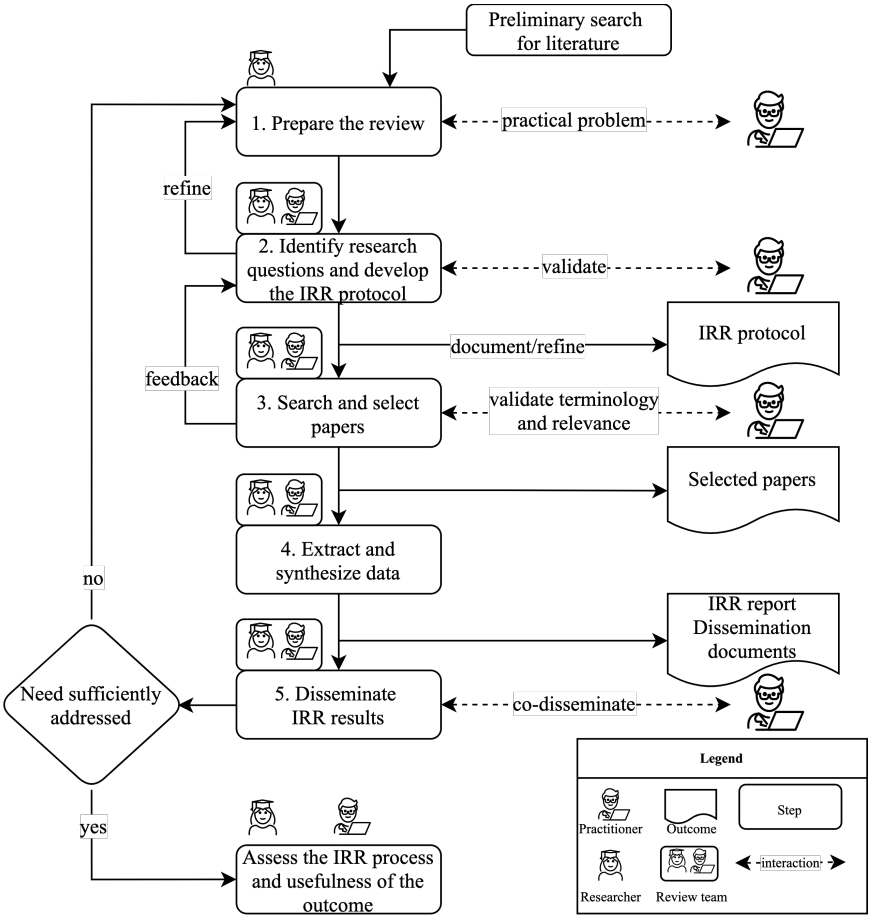

Fig. 1. Workflow for performing an IRR

teraction between researchers and practitioners aims to get a commitment to performing the IRR and identifying a context-relevant problem for the IRR.

Researchers lead the process to conduct the IRR. First, they form an initial review team based on the broad SE knowledge area (like software testing or requirements engineering) and the practitioners' interests. Ideally, the review team should comprise at least two researchers, but it may be formed only by one researcher. Having at least two researches enriches the discussion and helps to improve the reliability of the study. It is even better if one of the researchers has experience conducting a systematic secondary study like SLR, SMS, or RR. During the review, the review team performs the search, selects papers, extracts, and synthesizes knowledge. Practitioners may or may not directly participate in these tasks depending on their degree of involvement. However, throughout the IRR, they are expected to, at the very least, have communication channels open

Step	Activity to promote interaction	Outcomes
1. Prepare the review	1.1. Form an initial review team of researchers and practitioners. 1.2. Present IRRs (typical aims, process, expected outcomes, timeline and commitment). 1.3. Collaboratively, identify and rank a list of information needs appropriate for an IRR. 1.4. Choose a prioritized information need of mutual interest. 1.5. Agree on practitioners' involvement and update the review team. 1.6. Input meeting: to present current practices and context.	Review team Description of information need Review topic
2. Identify research questions and develop the IRR protocol	2.1. Jointly, define the research questions. 2.2. Prepare and validate with practitioners the search strategy and inclusion/exclusion criteria.	IRR Protocol
3. Search and select papers	3.1. Perform the search. Present and validate the search results. 3.2. Apply inclusion/exclusion criteria 3.3. Update / extend the search	Papers to analyze
4. Extract and synthesize data	4.1. Co-design IRR reports and dissemination documents 4.2. Extract information and elaborate reports 4.3 Reaction meeting: present the initial results to the practitioners involved	Reports and dissemination documents
5. Disseminate IRR results	5.1. Identify the audience and medium of communication 5.2. Disseminate results to practitioners 5.3. Practitioners disseminate to other practitioners 5.4. Disseminate results to academic audiences	Reports and dissemination documents

Table 2. A list of activities proposed to increase the involvement of practitioners in rapid reviews (the steps in the first column are adapted from Cartaxo et al. [11]

with the review team to answer questions and provide feedback related to the relevance and context. Before starting with the review, researchers and practitioners need to clarify mutual expectations, agree non-disclosure agreements if applicable, and define roles and responsibilities [29].

In an initial presentation meeting, researchers introduce an overview of the IRR method, outcomes, roles, and responsibilities. This presentation helps to develop a shared understanding of expected outcomes and commitment. Before, the meeting, researchers do a preliminary search to get a sense of the literature in the field and support the dialogue with practitioners. Secondary studies are especially useful for this purpose [30,36].

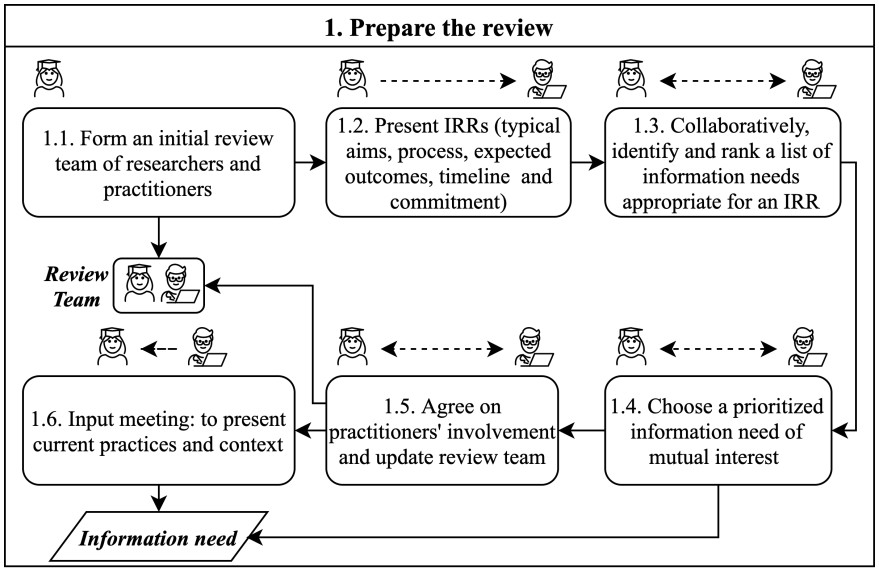

Fig. 2. Prepare the review aims at get a shared understanding of what is an IRR, the expected outcomes, and to plan the work ahead

When practitioners have proposed the IRR topic concerning a practical problem, researchers and practitioners continue to identify context elements and research questions. Although, they have identified a practical problem they may need to specify the IRR scope further. To narrow the review topic, researchers may propose a shortlist of topics to the practitioners based on the results of the preliminary search and the practical problem [14]. With the list of topics, the practitioners rank the suggested topics according to their problem in context or suggest other directions. This exchange helps to agree on the IRR topic and contributes to making it interesting for both researchers and practitioners.

After the meeting, the review team may be updated with practitioners or new researchers. According to the practitioners' interest and familiarity with scientific literature, their participation may vary from being part of the review team to only provide feedback at specific points, e.g., clarifying terminology or the relevance of specific studies. The review team defines practical aspects like communication channels, file sharing, meetings calendar, and estimate the practitioners' time required to conduct the review, including both meetings and time required to answer questions.

Researchers need to get a good understanding of the practical problem and context variables. Researchers and practitioners may have an input meeting. During the input meeting, practitioners present the current practices in their context [14]. This meeting allows the review team to get a first approach to the research questions and keywords when preparing search queries.

At the end of this step, a team for the review has been formed. The team has an initial view of the problem in the context of practitioners. The review team has a preliminary sense of research in the field and defined some practicalities like communication channels, meetings calendar, and follow-up meetings.

3.2 Develop the IRR protocol

For this step, we suggest two activities (see Fig. 2) related to define research questions with practitioners and prepare and validate the search strategy and inclusion/exclusion criteria.

The IRR protocol keeps track of the decisions and steps to conduct the review [21,23]. During this step, the review team develops the protocol. However, this step may be revisited and the protocol updated in several iterations as new insights about both the context and the literature are gained [24,35]. This favors the rigor of the study and the trust in the results. The protocol should contain at least [24]:

- Problem definition
- Research questions
- Search strategy
- Exclusion criteria
- Synthesis methods
- Initial proposals on how to disseminate the results

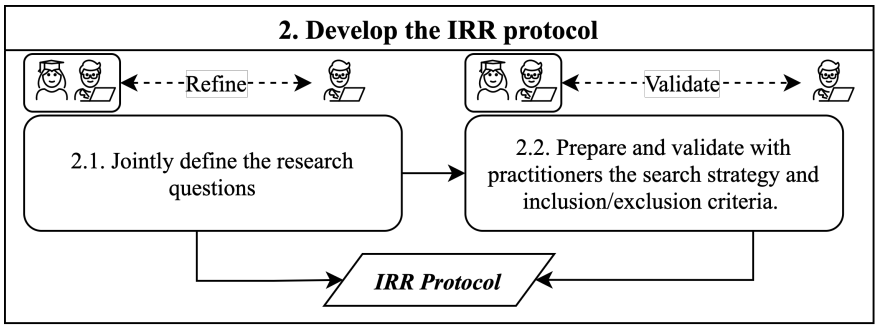

Fig. 3. IRR follows a protocol that keeps track of decisions during the method

Research questions are crucial in the review because the search and knowledge synthesis is based on them. Practical questions are more suitable for this type of review, instead of general and broad questions [20]. Compared with SLRs, the research question's scope is narrower as the questions in IRR address practical questions in a specific industry context [17,53].

Researchers are used to working with research questions; thus, they may guide the formulation. They frame preliminary questions based on available literature and the practical problem. When defining research questions for IRR, it

is essential to ensure alignment with practitioners' terminology. Questions are refined based on the exchange between the review team and practitioners to ensure that the final questions are relevant and include the particular practitioners' context [24,36]. After a preliminary search, the review teams should evaluate if the research questions are suitable for an IRR according to the existent primary studies. If a preliminary search does not find related studies, it is probably unsuitable to continue with this approach.

The IRR protocol includes the search strategy and the inclusion/exclusion criteria. To define the search strategy, the review teams may consider insights from the preliminary search, the terminology extracted from the interaction with practitioners, and the identified context elements.

In an IRR, the review team uses shortcuts to reduce the number of sources to analyze and find more specific papers. Some of the shortcuts include [17,20, 27,33,36,50]:

- Base the review only in secondary studies
- Use only one search engine e.g., Scopus, Google Scholar
- Limit to only studies published in English
- Limit to specific journals and conferences
- Limit from some specific date range
- Limit according to the methodology of the study e.g., case studies.

If the review team may consult researchers with experience in the IRR topic, they can conduct peer review on the search queries to verify that all related terms are included [36,47]. Some other search strategies like snowballing [20] or including grey literature may be considered if the review team has experience with these techniques. Regarding the inclusion/exclusion criteria, fixing strict exclusion criteria reduces the number of papers and thus favors rapidness [50].

This step should result in a preliminary version of the IRR protocol containing research questions, and a preliminary version of the search strategy, and inclusion/exclusion criteria. In addition, the review team may have initial ideas about how results will be communicated and the type of reports and documents to develop.

3.3 Search and select papers

Through the activities in this step (see Fig. 4), the review team performs the search and selection of papers. These activities require high interaction with practitioners to validate specific aspects such as terminology, the relevance of specific studies, and context elements. The review team may decide to update or extend the search of sources by conducting snowballing or manual search [20]. These decisions need to be updated in the IRR protocol.

With the search results, the review team applies the exclusion criteria to select the set of papers included in the review. As in SLRs, the papers' selection may be divided into the following activities: Review the titles, read the title and full abstract, and read the full paper. A common practice in medicine is

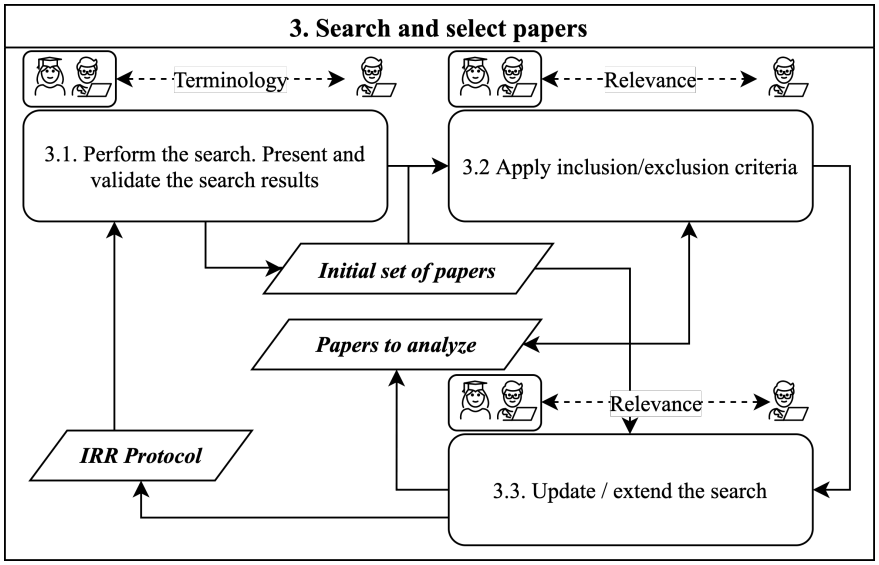

Fig. 4. The search and selection of papers is a critical step to ensure the rapidness and relevance of the IRR

that only one team member make decisions about inclusion/exclusion of studies. Leaving the responsibility to only one reviewer reduces the time and avoids solving discrepancies about including/excluding specific studies [17, 23, 30, 50].

During this step, the review team may use tools to support the selection of papers. Felizardo and Carver [18] conducted a systematic search for approaches and tools to automate the SLR process. They found that selection of studies is the activity with most tool support. In their study, the authors analyze the different approaches and provide references to tools. At this point, the review team has a set of papers to analyze to answer the research questions.

3.4 Extract and synthesize data

The activities in this step, see Fig.5, aim to prepare and develop the material that will be used to disseminate the IRR results.

Before extracting information from research papers, the review team designs initial reports that will be shared with practitioners. This allows the reviewers to focus on what to search for in the papers. We suggest presenting the result as narrative summaries. A narrative summary is a text that summarizes the findings of the synthesis. More advanced methods like thematic analysis [13] may be used only when having a large number of primary studies, and the process will not impact the time to completion. The synthesis is mainly oriented to describe research results through a narrative summary [22, 45].

In a reaction meeting [14], the review team presents the IRR results to the initial group of practitioners. The practitioners provide feedback and suggestions

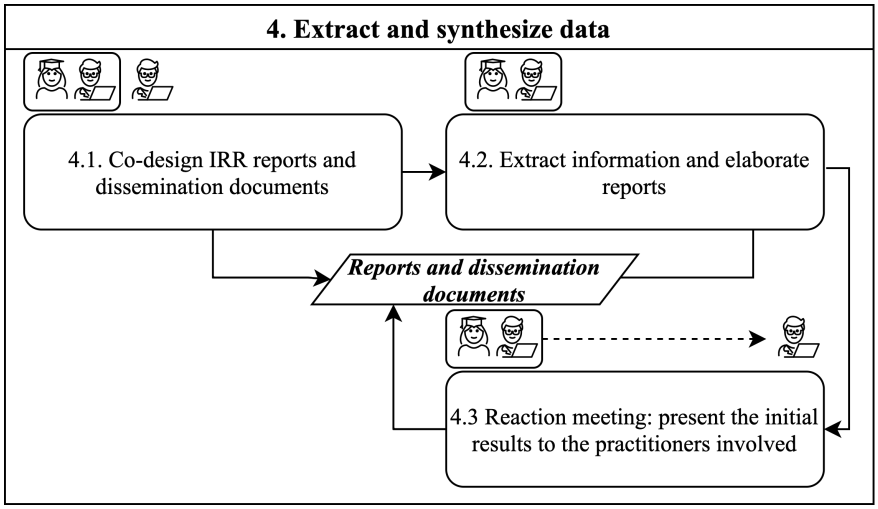

Fig. 5. During this task, the review team extracts and synthesizes information from the selected paper to answer the research questions.

on how to communicate them to a larger audience. Keep in mind that software engineers, with few exceptions, do not read scientific papers. Thus, the reports need to be designed in a practitioner friendly manner [30]. Some alternatives are visual abstracts [48], evidence briefings [12], presentations, seminars, and posters.

3.5 Disseminate IRR results

Fig. 6. shows the suggested activities in this step to disseminate the IRR results.

Initially, the results are communicated to the practitioners involved in the review. Later, the results may be shared with other practitioners in the same organization. For some groups, the diffusion may require to adapt or create new ways to share the results. For example, one group may need less scientific details, while others may require only to present tools or source code. These strategies and diffusion actions need to be coordinated with practitioners who know their context and colleagues better.

Although an IRR's main goal is not to produce a scientific publication, some results may be relevant for academic audiences [33, 39]. If it is the case, the researcher may find the appropriate medium and publish the results. Otherwise, and following non-disclosure agreements, the results may be shared via social networks or in other academic spaces such as workshops, university courses, and online discussion.

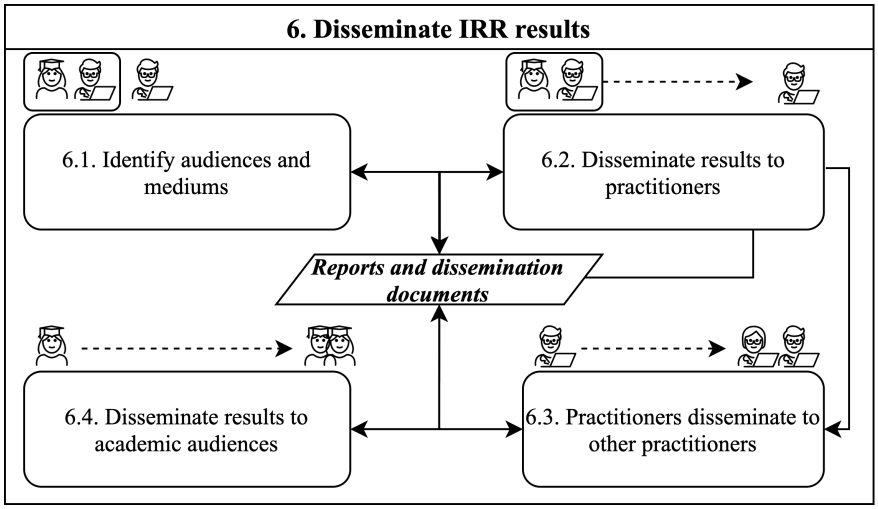

Fig. 6. The last step in conducting the review is to disseminate the results.

3.6 IRR evaluation

Once the IRR results have been disseminated, the review team and the practitioners evaluate if the IRR results support the initial information needs. A possible result is that researchers and practitioners want to explore further a specific topic or take another perspective. Thus, they identify new research questions and apply the steps again. Another possible result may be the identification of a gap in research. If it is the case, the results are a starting point to design and support new research.

In our view, conducting an IRR is an opportunity for mutual understanding between research and practice. When evaluating the IRR, consider besides the outcomes the learnings by participating in the review. By getting involved in the IRR, practitioners develop an awareness of research results and their application in practice while researchers better understand industry challenges and their context.

4 Discussion

RR emerged in medicine as a faster approach than systematic reviews to synthesize knowledge from primary studies. While systematic reviews are well-defined, rapid reviews is an umbrella term that includes a spectrum of related methods. An important aspect of the approach presented in this work is the knowledge exchange between researchers and practitioners. In medicine, there are review groups that work on synthesizing knowledge for decision-making by following standardized protocols accepted by the community. In software engineering, knowledge synthesis is done by the knowledge-users themselves, either researchers or engineers, with different approaches and varying degrees of rigor.

In medicine, practitioners rely on and expect input from academia, while in software engineering, new ideas may be more important than evidence for practitioners approaching academia [55]. Proposed interventions need to be adapted to and re-evaluated in the new context [54]. This can be seen as an argument for allowing synthesizing knowledge in an earlier stage. However, to enable the validity assessment of the conclusions drawn, transparency and context-dependency is key.

RR lack a unique method, but there are some similarities to traditional systematic reviews. Even if the RR approaches are expeditious, they follow a structural set of steps where the research questions are defined at the beginning of the review, making it possible to track the review process and, if necessary, repeat it. Transparency is important since the processes and decision making are faster than in systematic reviews. For these reasons, all the decisions are documented and reported.

Interactive Rapid reviews are conducted in less time than systematic reviews since there is a requirement to have shorter feedback cycles when working with practitioners who want to receive knowledge to affect their products, processes, etc. One way of shortening the time in an IRR is to keep a narrow scope. Here a balance must be decided between answering all relevant questions for a subject and answering only the questions of interest in the collaboration between the practitioners and the researcher. Compared to a traditional review, the selection of subject scope is probably more dependent on practitioners' interests. To what extent this means that relevant and important areas in the literature is not prioritized can be a question for further research.

Another way to decrease the time of IRR is to use shortcuts to expedite the process. To satisfy the time restrictions, rapid reviews skip steps carried out in traditional systematic reviews or limit some steps. Some examples are: avoiding analysis of inter reviewer agreement, not conducting quantitative analysis, and limiting the search, e.g. by language, time, or the number of databases. Here, a balance must be decided between traditional rigor and obtaining information in a timely way.

Rapid reviews have the potential to bring researchers closer to practitioners and improve communication between them. IRRs aim to maintain professionals' interest and commitment during the review and provide them with useful results. For researchers, we see in IRR an opportunity to get closer to the industry, gather data and information, which we believe is essential in software engineering research.

We consider, like Wohlin [56], that working with industry is more about knowledge exchange than about knowledge transfer. Consequently, our proposal for IRRs is base on the idea that conducting a rapid review with practitioners is an opportunity to establish a bidirectional dialogue where researchers and practitioners get the chance to learn from each other. This interaction facilitates mutual understanding, favors research relevance, and paves the way for future collaborations.

5 Conclusion and Future Work

Our proposal for IRR reinforces the interaction between researchers and practitioners while performing the review. We believe such researcher-led, interactive reviews may improve the knowledge exchange between researchers and software engineering professionals. An IRR starts from a specific knowledge need from practitioners, which implies that the topic is relevant for practitioners from the beginning. During the review, practitioners are highly involved in refining the research questions and defining the protocol, which increases the researchers' understanding of the specific context. Throughout the selection of studies and information extraction, researchers and practitioners keep communicating, contributing to learning from each other. IIR results are disseminated in a practitioner friendly way, making them easier to use.

According to the points mentioned above, we included in our proposal opportunities to focus on the researcher-practitioner exchange during the review. Overall, we recognize in conducting rapid reviews an opportunity to establish a bi-directional exchange between researchers and practitioners that enables future joint work.

Finally, we identified the some potential benefits and challenges of conducting rapid reviews in software engineering. We envision that conducting rapid reviews in collaboration with practitioners may: 1) incentivize a dialogue between researchers and practitioners, 2) provide research results to the industry that are relevant for their context, 3) provide researchers opportunities to learn about the practitioner's problems and their context, and 4) develop networks that could be the base for new collaborative projects.

Similarly, we find the following points as challenging while conducting a rapid review. 1) Time constraints can influence the quality of the review. 2) There is a lack of clear guidance on how to perform rapid reviews and tools to verify the review's quality. 3) There could be misunderstandings about the depth and breadth of a rapid review. 4) There may be a lack of research results on the topic selected. 5) Practitioners' involvement may lead to bias due to practitioners' oriented results.

To address these challenges, we suggest to: 1) keep a protocol that contain all the decisions made in the review to evaluate the strength of conclusions, 2) follow the guidelines proposed in this paper, 3) reinforce transparency as an essential practice when working with industry, and 4) conduct a preliminary search and refine the research questions to identify when there is no available literature in the area, and 5) declare expectations from the beginning about the goals and role of researchers.

As future work, we plan to validate this proposal empirically by studying actual cases of rapid reviews with the industry and evaluate how rapid reviews impact researcher-practitioner communication within and beyond a research collaboration.

References

1. Ali, N.B., Engström, E., Taromirad, M., Mousavi, M.R., Mehmood Minhas, N., Helgesson, D., Kunze, S.: On the search for industry-relevant regression testing research. Empirical Software Engineering **24**(4), 2020–2055 (2019)
2. Ali, N.B., Petersen, K.: Evaluating strategies for study selection in systematic literature studies. In: Morisio, M., Dybå, T., Torchiano, M. (eds.) Proceedings of the International Symposium on Empirical Software Engineering and Measurement, ESEM '14. pp. 45:1–45:4. ACM (2014)
3. Ali, N.B., Petersen, K., Wohlin, C.: A systematic literature review on the industrial use of software process simulation. Journal of Systems and Software **97**, 65–85 (2014)
4. Ali, N.B., Usman, M.: Reliability of search in systematic reviews: Towards a quality assessment framework for the automated-search strategy. Information & Software Technology **99**, 133–147 (2018)
5. Ali, N.B., Usman, M.: A critical appraisal tool for systematic literature reviews in software engineering. Information & Software Technology **112**, 48–50 (2019)
6. Budgen, D., Brereton, P., Drummond, S., Williams, N.: Reporting systematic reviews: Some lessons from a tertiary study. Information and Software Technology **95**, 62–74 (2018)
7. Budgen, D., Kitchenham, B.A., Brereton, P.: The case for knowledge translation. In: Proceedings of the International Symposium on Empirical Software Engineering and Measurement, ESEM. pp. 263–266. IEEE Computer Society, Baltimore, Maryland, USA (2013)
8. Cartaxo, B., Pinto, G., Fonseca, B., Ribeiro, M., Pinheiro, P., Baldassarre, M.T., Soares, S.: Software engineering research community viewpoints on rapid reviews. In: Proceedings of the International Symposium on Empirical Software Engineering and Measurement, ESEM. pp. 1–12. IEEE, Porto de Galinhas, Recife, Brazil (2019)
9. Cartaxo, B., Pinto, G., Soares, S.: The role of rapid reviews in supporting decision-making in software engineering practice. In: Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering EASE. pp. 24–34. ACM, Christchurch, New Zealand (2018)
10. Cartaxo, B., Pinto, G., Soares, S.: Towards a model to transfer knowledge from software engineering research to practice. Information and Software Technology **97**, 80–82 (2018)
11. Cartaxo, B., Pinto, G., Soares, S.: Rapid Reviews in Software Engineering. In: Felderer, M., Travassos, G.H. (eds.) Contemporary Empirical Methods in Software Engineering, pp. 357–384. Springer International Publishing (2020)
12. Cartaxo, B., Pinto, G., Vieira, E., Soares, S.: Evidence briefings: Towards a medium to transfer knowledge from systematic reviews to practitioners. In: Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement. p. 57. ACM (2016)
13. Cruzes, D.S., Dyba, T.: Recommended steps for thematic synthesis in software engineering. In: Proceedings of the 2011 International symposium on empirical software engineering and measurement. pp. 275–284. IEEE (2011)
14. Deverka, P.A., Lavallee, D.C., Desai, P.J., Esmail, L.C., Ramsey, S.D., Veenstra, D.L., Tunis, S.R.: Stakeholder participation in comparative effectiveness research: defining a framework for effective engagement. Journal of comparative effectiveness research **1**(2), 181–194 (2012)

15. Edison, H., Ali, N.B., Torkar, R.: Towards innovation measurement in the software industry. Journal of Systems and Software **86**(5), 1390–1407 (2013)
16. Engström, E., Runeson, P., Skoglund, M.: A systematic review on regression test selection techniques. Information & Software Technology **52**(1), 14–30 (2010)
17. Featherstone, R.M., Dryden, D.M., Foisy, M., Guise, J.M., Mitchell, M.D., Paynter, R.A., Robinson, K.A., Umscheid, C.A., Hartling, L.: Advancing knowledge of rapid reviews: An analysis of results, conclusions and recommendations from published review articles examining rapid reviews. Systematic Reviews **4**(1) (2015)
18. Felizardo, K.R., Carver, J.C.: Automating Systematic Literature Review, pp. 327–355. Springer International Publishing, Cham (2020)
19. Felizardo, K.R., de Souza, É.F., Napoleão, B.M., Vijaykumar, N.L., Baldassarre, M.T.: Secondary studies in the academic context: A systematic mapping and survey. Journal of Systems and Software **170**, 110734 (2020)
20. Ganann, R., Ciliska, D., Thomas, H.: Expediting systematic reviews: Methods and implications of rapid reviews. Implementation Science **5**(1) (2010)
21. Garritty, C., Stevens, A., Gartlehner, G., King, V., Kamel, C.: Cochrane rapid reviews methods group to play a leading role in guiding the production of informed high-quality, timely research evidence syntheses. Systematic Reviews **5**(1) (2016)
22. Guzmán, L., Lampasona, C., Seaman, C., Rombach, D.: Survey on research synthesis in software engineering. In: Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering. pp. 1–10 (2014)
23. Harker, J., Kleijnen, J.: What is a rapid review? a methodological exploration of rapid reviews in health technology assessments. International Journal of Evidence-Based Healthcare **10**(4), 397–410 (2012)
24. Hartling, L., Guise, J.M., Hempel, S., Featherstone, R., Mitchell, M.D., Motu'apuaka, M.L., Robinson, K.A., Schoelles, K., Totten, A., Whitlock, E., et al.: Fit for purpose: Perspectives on rapid reviews from end-user interviews. Systematic Reviews **6**(1) (2017)
25. Hartling, L., Guise, J.M., Kato, E., Anderson, J., Belinson, S., Berliner, E., Dryden, D.M., Featherstone, R., Mitchell, M.D., Motu'Apuaka, M., et al.: A taxonomy of rapid reviews links report types and methods to specific decision-making contexts. Journal of Clinical Epidemiology **68**(12), 1451–1462.e3 (2015)
26. Hislop, D., Bosua, R., Helms, R.: Knowledge management in organizations: A critical introduction. Oxford University Press (2018)
27. Kaltenthaler, E., Cooper, K., Pandor, A., James, M.M.S., Chatters, R., Wong, R.: The use of rapid review methods in health technology assessments: 3 case studies. BMC Medical Research Methodology **16**(1) (2016)
28. Kelly, S.E., Moher, D., Clifford, T.J.: Defining rapid reviews: a modified delphi consensus approach. International Journal of Technology Assessment in Health Care **32**(4), 265–275 (2016)
29. Khangura, S., Polisena, J., Clifford, T., Farrah, K., Kamel, C.: Rapid review: An emerging approach to evidence synthesis in health technology assessment. International Journal of Technology Assessment in Health Care **30**(1), 20–27 (2014)
30. Khangura, S., Konnyu, K., Cushman, R., Grimshaw, J., Moher, D.: Evidence summaries: The evolution of a rapid review approach. Systematic Reviews **1**(1) (2012)
31. Kitchenham, B.A., Dybå, T., Jørgensen, M.: Evidence-based software engineering. In: Proceedings of the 26th International Conference on Software Engineering (ICSE). pp. 273–281 (2004)
32. Kitchenham, B.A., Budgen, D., Brereton, P.: Evidence-based software engineering and systematic reviews, vol. 4. CRC press (2015)

33. Lambert, R., Vreugdenburg, T.D., Marlow, N., Scott, N.A., McGahan, L., Tivey, D.: Practical applications of rapid review methods in the development of australian health policy. Australian Health Review **41**(4), 463–468 (2017)
34. Le Goues, C., Jaspan, C., Ozkaya, I., Shaw, M., Stolee, K.T.: Bridging the gap: From research to practical advice. IEEE Software **35**(5), 50–57 (2018)
35. Mattivi, J.T., Buchberger, B.: Using the amstar checklist for rapid reviews: Is it feasible? International Journal of Technology Assessment in Health Care **32**(4), 276–283 (2016)
36. McIntosh, H.M., Calvert, J., Macpherson, K.J., Thompson, L.: The healthcare improvement scotland evidence note rapid review process: Providing timely, reliable evidence to inform imperative decisions on healthcare. International Journal of Evidence-Based Healthcare **14**(2), 95–101 (2016)
37. Mendes, E., Wohlin, C., Felizardo, K., Kalinowski, M.: When to update systematic literature reviews in software engineering. Journal of Systems and Software p. 110607 (2020)
38. Mikkonen, T., Lassenius, C., Männistö, T., Oivo, M., Järvinen, J.: Continuous and collaborative technology transfer: Software engineering research with real-time industry impact. Information and Software Technology **95**, 34–45 (2018)
39. Moore, G., Redman, S., Rudge, S., Haynes, A.: Do policy-makers find commissioned rapid reviews useful? Health Research Policy and Systems **16**(1) (2018)
40. O'Leary, D.F., Casey, M., O'Connor, L., Stokes, D., Fealy, G.M., O'Brien, D., Smith, R., McNamara, M.S., Egan, C.: Using rapid reviews: an example from a study conducted to inform policy-making. Journal of Advanced Nursing **73**(3), 742–752 (2017)
41. Patnode, C.D., Eder, M.L., Walsh, E.S., Viswanathan, M., Lin, J.S.: The use of rapid review methods for the u.s. preventive services task force. American Journal of Preventive Medicine **54**(1), S19–S25 (2018)
42. Petersen, K., Ali, N.B.: Identifying strategies for study selection in systematic reviews and maps. In: Proceedings of the 5th International Symposium on Empirical Software Engineering and Measurement. pp. 351–354. IEEE Computer Society (2011)
43. Petersen, K., Vakkalanka, S., Kuzniarz, L.: Guidelines for conducting systematic mapping studies in software engineering: An update. Information and Software Technology **64**, 1–18 (2015)
44. Polisena, J., Garritty, C., Kamel, C., Stevens, A., Abou-Setta, A.M.: Rapid review programs to support health care and policy decision making: A descriptive analysis of processes and methods. Systematic Reviews **4**(1) (2015)
45. Pope, C., Mays, N., Popay, J.: Synthesising qualitative and quantitative health evidence: A guide to methods: A guide to methods. McGraw-Hill Education (UK) (2007)
46. Ros, R., Bjarnason, E., Runeson, P.: A machine learning approach for semi-automated search and selection in literature studies. In: Mendes, E., Counsell, S., Petersen, K. (eds.) Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering. pp. 118–127. ACM (2017)
47. Spry, C., Mierzwinski-Urban, M.: The impact of the peer review of literature search strategies in support of rapid review reports. Research Synthesis Methods **9**(4), 521–526 (2018)
48. Storey, M.A., Engström, E., Höst, M., Runeson, P., Bjarnason, E.: Using a visual abstract as a lens for communicating and promoting design science research in

software engineering. In: Proceedings of the 11th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement. p. 181–186. ESEM '17, IEEE Press (2017)

49. Taylor-Phillips, S., Geppert, J., Stinton, C., Freeman, K., Johnson, S., Fraser, H., Sutcliffe, P., Clarke, A.: Comparison of a full systematic review versus rapid review approaches to assess a newborn screening test for tyrosinemia type 1. Research Synthesis Methods **8**(4), 475–484 (2017)
50. Tricco, A.C., Antony, J., Zarin, W., Strifler, L., Ghassemi, M., Ivory, J., Perrier, L., Hutton, B., Moher, D., Straus, S.E.: A scoping review of rapid review methods. BMC Medicine **13**(1) (2015)
51. Tricco, A.C., Langlois, E., Straus, S.E., Organization, W.H., et al.: Rapid reviews to strengthen health policy and systems: a practical guide. World Health Organization (2017)
52. Tricco, A.C., Zarin, W., Antony, J., Hutton, B., Moher, D., Sherifali, D., Straus, S.E.: An international survey and modified delphi approach revealed numerous rapid review methods. Journal of Clinical Epidemiology **70**, 61–67 (2016)
53. Watt, A., Cameron, A., Sturm, L., Lathlean, T., Babidge, W., Blamey, S., Facey, K., Hailey, D., Norderhaug, I., Maddern, G.: Rapid reviews versus full systematic reviews: An inventory of current methods and practice in health technology assessment. International Journal of Technology Assessment in Health Care **24**(2), 133–139 (2008)
54. Wieringa, R.J.: Design science methodology for information systems and software engineering. Springer (2014)
55. Williams, A.: Do software engineering practitioners cite research on software testing in their online articles? a preliminary survey. In: Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018. pp. 151–156 (2018)
56. Wohlin, C.: Empirical software engineering research with industry: Top 10 challenges. In: 2013 1st International Workshop on Conducting Empirical Studies in Industry (CESI). pp. 43–46. IEEE (2013)
57. Wohlin, C.: Guidelines for snowballing in systematic literature studies and a replication in software engineering. In: Proceedings of the 18th international conference on evaluation and assessment in software engineering. pp. 1–10 (2014)