

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/260290630>

A Tertiary Study: Experiences of Conducting Systematic Literature Reviews in Software Engineering

Conference Paper · April 2013

DOI: 10.1145/2460999.2461025

CITATIONS

69

READS

2,150

4 authors:



Salma Imtiaz

International Islamic University, Islamabad

37 PUBLICATIONS 527 CITATIONS

SEE PROFILE



Naveed Ikram

Riphah International University

74 PUBLICATIONS 1,141 CITATIONS

SEE PROFILE



Muneera Bano

The Commonwealth Scientific and Industrial Research Organisation

104 PUBLICATIONS 2,378 CITATIONS

SEE PROFILE



Mahmood Niazi

King Fahd University of Petroleum and Minerals

155 PUBLICATIONS 8,455 CITATIONS

SEE PROFILE

A Tertiary Study: Experiences of Conducting Systematic Literature Reviews in Software Engineering

Salma Imtiaz¹, Muneera Bano², Naveed Ikram³, Mahmood Niazi⁴

Faculty of Computing, Riphah International University, Islamabad, Pakistan^{1,3}

Faculty of Engineering and Information Technology, University of Technology, Sydney, Australia²

Department of Information and Computer Science, King Fahd University of Petroleum and Minerals, KSA⁴

salma.imtiaz@iiu.edu.pk¹, Muneera.Bano@uts.edu.au², naveed.ikram@riu.edu.pk³,
mkniazi@kfupm.edu.sa⁴

ABSTRACT

Context: The use of Systematic Literature Review (SLR) requires expertise and poses many challenges for novice researchers. The experiences of those who have used this research methodology can benefit novice researchers in effectively dealing with these challenges. **Objective:** The aim of this study is to record the reported experiences of conducting Systematic Literature Reviews, for the benefit of new researchers. Such a review will greatly benefit the researchers wanting to conduct SLR for the very first time. **Method:** We conducted a tertiary study to gather the experiences published by researchers. Studies that have used the SLR research methodology in software engineering and have implicitly or explicitly reported their experiences are included in this review. **Results:** Our research has revealed 116 studies relevant to the theme. The data has been extracted by two researchers working independently and conflicts resolved after discussion with third researcher. Findings from these studies highlight Search Strategy, Online Databases, Planning and Data Extraction as the most challenging phases of SLR. Lack of standard terminology in software engineering papers, poor quality of abstracts and problems with search engines are some of the most cited challenges. **Conclusion:** Further research and guidelines is required to facilitate novice researchers in conducting these phases properly.

Categories and Subject Descriptors

D.2 [Software Engineering]: D.2.0 General Standards

General Terms

Performance, Human Factors.

Keywords

Experiences, Lessons Learnt, Systematic Literature Reviews, Tertiary Study, Empirical Software Engineering.

1. INTRODUCTION

Since its first introduction in 2004 [12], Systematic Literature Review has gained focus of software engineering research community. It is a rigorous way of finding empirical evidence from literature with help of proper planning. The idea of the SLR

in software engineering was adopted from Evidence Based Medicine and introduced in an Evidence Based Software Engineering (EBSE) domain [10] [12]. Systematic Reviews, Mapping Studies and Tertiary Studies are an important part of EBSE. Kitchenham provided guidelines for researchers to help them with this new methodology [11]. New and, at times, interesting facts can be discovered about a research area based on existing evidences using systematic review [15]. In software engineering, the researchers have covered numerous diverse areas via SLR [9] [13] [16]. The growing interest in SLR by software engineering researchers highlight the importance of evidence based research. Most of these SLRs are conducted by experienced researchers, who are even involved in the development of the SLR guidelines. A novice researcher conducting the Systematic Review often faces issues which may not be bothersome for an experienced researcher [1] [14]. Additional knowledge and guidelines from the experienced researchers can be of great value to novice researchers undertaking SLR. Many of these experienced researchers have already documented their observations about SLRs in the field of EBSE research. In this paper we are presenting initial results of a systematic review on published experiences of conducting SLR or mapping study in software engineering. The objective of the study is to systematically identify and compile experiences of conducting SLR and mapping study. The results of this paper can be used to further improve SLR guidelines for novice researchers. We have identified the SLR phases with most of the issues, most frequently cited problems in each phase and year wise frequency of each problem.

The paper has been organized as follows: Section 2 describes the Background and Motivation. Section 3 describes the Protocol and gives detail of the planning phase. Section 4 highlights the Execution Phase. Section 5 outlines Results. Section 6 is the Discussion. Section 7 contains Threats to Validity. Section 8 elaborates lessons learnt. Section 9 gives the Conclusion and Future Work.

2. BACKGROUND AND MOTIVATION

Systematic Literature Reviews differ from traditional literature reviews due to meticulous planning and scheduling of activities, the rigor of the process, and exhaustive search strategy to find all relevant evidence. We need to properly define the protocol before execution and keep ourselves focused on it throughout. For new researchers the initial phase could be difficult and time consuming. There are various factors that can increase difficulty of the review process. Developing a search strategy can be challenging as the software engineering research papers do not use common and standard terms and formats. Similarly searching

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

EASE '13, April 14 - 16 2013, Porto de Galinhas, Brazil

Copyright 2013 ACM 978-1-4503-1848-8/13/04...\$15.00.

different online databases may yield inconsistent results, as they all support variant semantic format and the abstracts of the published papers do not follow any set standard. The limitation of the online databases itself makes searching a challenging task. Many researchers have shared their experiences of conducting SLR [1] [4] [8] [15] and some have explored different phases of SLR to provide in-depth knowledge on how to perform them [1] [2] [6] [7] [11]. A research work [17] also compares results of two independently conducted SLR addressing the same research question, to determine the reliability of the methodology. It also identifies some issues and best practices while comparing the SLRs. Almost all of the reviewed researchers are experienced and some of them are also part of the team that developed SLR guidelines. These guidelines provide valuable knowledge and important lessons for practitioners and researchers about the different phases of systematic review process. We aim to compile and analyze the experiences of conducting SLR and mapping study, and develop additional guidelines which will be helpful for researchers especially novice researchers.

3. The SLR Protocol

The protocol was developed by one member and executed by two team members. The details of the planning phase are given below.

3.1. The Research Question

The research question for our study is as follows;

RQ: What are the reported experiences of conducting Systematic Literature Reviews in Software Engineering?

3.2. Search Strategy

We conducted search in two phases; primary search and secondary search. Based on our research question, the following research strategy was used for primary search:

1. *Derive major search terms from our research question*
2. *Find alternate terms and synonyms of our major search terms*
3. *Join the search terms using Boolean operators to create a search string*
4. *Customize the search string for various online databases for execution*

The following databases were selected for searching the primary search;

1. IEEE Xplore
2. ACM DL
3. Science Direct
4. CiteSeer
5. Evidence Based Software Engineering Bibliography
<http://www.dur.ac.uk/ebse/biblio.php>
6. Google Scholar

For secondary search we plan to scan references of included studies to identify relevant evidence. This step is not complete yet.

3.3. Search String

We have two major search terms: "Systematic Literature Reviews" and "Software Engineering". We did not use the term "experiences" due to the fact that some papers that report SLR, also report experiences. Explicit experience reports would also be captured by the term systematic literature review. To identify the alternatives, we consulted and used existing established terms used to conduct a tertiary study [13]. The search string is formed by combining search terms using Boolean operators as was done in the tertiary study [16];

("Software engineering") AND ("review of studies" OR "structured review" OR "systematic review" OR "literature review" OR "literature analysis" OR "in-depth survey" OR

"literature survey" OR "meta analysis" OR "past studies" OR "subject matter expert" OR "analysis of research" OR "empirical body of knowledge" OR "overview of existing research" OR "body of published research" OR "Evidence based" OR "evidence based" OR "study synthesis" OR "study aggregation" OR "systematic literature review" OR SLR)

The string was customized for selected databases and applied on Abstracts of the papers.

3.4. Publication Inclusion/Exclusion Criteria

We included the studies that have reported experiences of conducting SLR in Software Engineering. The below given criteria should be valid for final inclusion;

1. *If it is related to the Systematic Literature Reviews or Mapping Studies.*
2. *If it is from Software Engineering Domain.*
3. *If it is describing any Experience or Lessons or Findings or Problems or Issues of planning and conducting SLR or mapping study.*

3.5. Quality Assessment

The included studies were assessed for their reporting quality. The following checklist was applied on the reports to assess them with either YES=1 or NO=0;

1. *Are the experiences or findings drawn after actually conducting an SLR?*
2. *Is the SLR from which the findings are drawn, sufficiently described to create the context and understanding for experience or findings?*
3. *Are the experiences of other researchers consulted while describing their own?*
4. *Are the experiences or findings linked clearly to one of the phases of SLR?*

3.6. Data Extraction

The following data was extracted namely; Publication, Context and Findings [8]. The publication and context information includes data variables like; *Paper ID, Title, Author(s), Conference/Journal, Date of Publication, Type of Study (Experience Report/SLR reporting Experiences)* whereas the below given data was extracted for findings category.

1. *Experiences*
2. *Phase of SLR specific to Experiences*

3.7. Data Synthesis

Having extracted the data in the forms, the next step was to synthesize it to find the following information;

1. *Frequency of Conference and Journal Evidence*
2. *Frequency of published evidence in individual conference/Journal*
3. *Frequency of the Experiences*
4. *Percentage of reported experiences in each phase of SLR*
5. *Year wise Frequency of Experience*
6. *Frequency of type of Research Methodology*
7. *Frequency of Reported Experiences in the most Problematic Phase.*

Data extracted was performed according to thematic synthesis [8]. The steps of the synthesis are given below. The initial categorization of the experiences (step 3) has been performed according to the phases of SLR.

1. Extract data
2. Code data
3. Translate codes into themes
4. Create a model of higher order themes
5. Assess the trustworthiness of the synthesis

3.8. Pilot Testing

We performed pilot testing of the search strategy, inclusion/exclusion criteria and data extraction forms. The search string was applied on IEEE xplore, ACM DL, Science Direct and Google Scholar. Citation and abstracts were downloaded in Zotero. After applying inclusion/exclusion check list we found 25 relevant studies. The quality assessment and data extraction forms were also pilot tested and the protocol was finalized.

4. EXECUTION

The search string was customized and executed on selected databases by one member. IEEE Xplore, ACM DL, Science Direct, CiteSeer, EBSE Bibliography and Google scholar. The results of the query were saved in Zotero. Trace numbers were given to each study for future reference. Inclusion/Exclusion was performed by the same person responsible for searching the databases. Table 1 shows the results of query execution and inclusion / exclusion. The final inclusion is based on the condition that all three checks of inclusion criteria are true.

Table 1: Result of Primary Search and Inclusion

DATABASE	FOUND	FINALLY INCLUDED
IEEE Xplore	69	23
Science Direct	35	22
ACM DL	126	48
CiteSeer x	39	6
EBSE Bibliography	146	45
Google Scholar	500	130
TOTAL	915	273

After discarding duplicates, we have a final list of 174 studies as a result of primary search. These were published during the period from 2005 to 2011. Some of these studies were related to same SLR, thus the number 174 was reduced to 116 studies after careful selection.

Data extraction was performed on the selected studies independently by two researchers to remove bias. The results were then compared and conflicts were resolved by discussion in a meeting with the third researcher. Data was extracted from a total of 116 studies, as the remaining studies were either not relevant or did not report experience. We were unable to find and download 7 of searched studies. These studies were apart from the finally included 116 studies. The data was recorded in pre designed forms. Integrated approach was use for coding data [5].

Quality assessment was also performed individually by two researchers on the criteria mentioned in section 3.6 and conflicts were resolved by the third researcher. We did not exclude any study on the basis of its QA score, as the minimum requirement of SLR execution by the authors of the study was present in all studies.

The initial data synthesis was performed with the help of excel sheet where all the data of experiences was input in tabular format. The frequency of experiences for the most problematic phase was counted manually. Data synthesis was performed according to thematic synthesis [8]. The steps of the synthesis are given below. The initial categorization of the experiences (step 3) has been performed according to the phases of SLR.

1. Extract data
2. Code data
3. Translate codes into themes
4. Create a model of higher order themes
5. Access the trustworthiness of the synthesis

The Creation of the model for higher order themes and judgment of trustworthiness of the synthesis is still to be done.

5. Results

A total of 116 SLR/ Experience reports were retrieved as part of searching phase. The categorization of these reports into journal or conference publications was done, where 68 studies were published in conferences and 48 were published in journals. The names of the specific conferences and journals, which have contributed most of the evidence, are given below. We were unable to acquire publication information of 2 papers.

1. 25 from Journal of Information and Software Technology(IST)
2. 17 from Empirical Software Engineering and Measurement(ESEM)
3. 12 from Evaluation and Assessment in Software Engineering(EASE)

The information highlights the most probable sources of empirical research specifically with respect to SLR.

The year wise distribution of published material is presented in Figure 1 the progressive increase over the period indicates popularity of SLR as a research methodology.

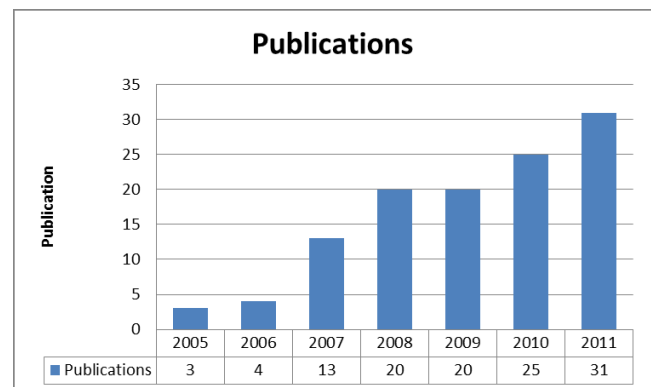


Figure 1: Year wise distribution of included publications

The Experience of conducting SLR has been reported explicitly in published Experience Reports and implicitly while explaining SLR steps. The number of SLR performed while executing case studies is 4, whereas SLRs performed as a result of experimentation are 10. Three papers report experience of conducting SLR with the help of interviews whereas 10 experience reports explicitly highlighted experiences of planning and executing SLR.

The Number of experiences reported in each phase of SLR is shown in figure 2. It shows that Planning the Search Strategy is the most difficult phase of SLR. Inclusion/ Exclusion and Data extraction also demand careful planning. The least problematic phases as stated are Pilot Testing and Reporting. We are using the terms experiences and problems alternatively because the data extraction shows that most of the experiences reported are issues or problems faced during planning and execution of SLR.

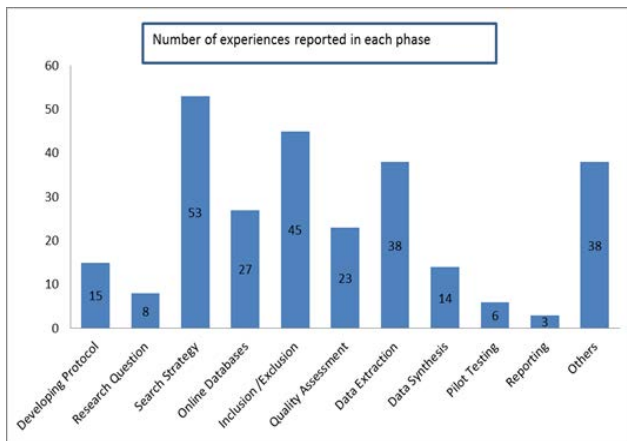


Figure 2: Frequency of Experiences reported in each Phase.

The percentage of these experiences is highlighted in figure 3 given below.

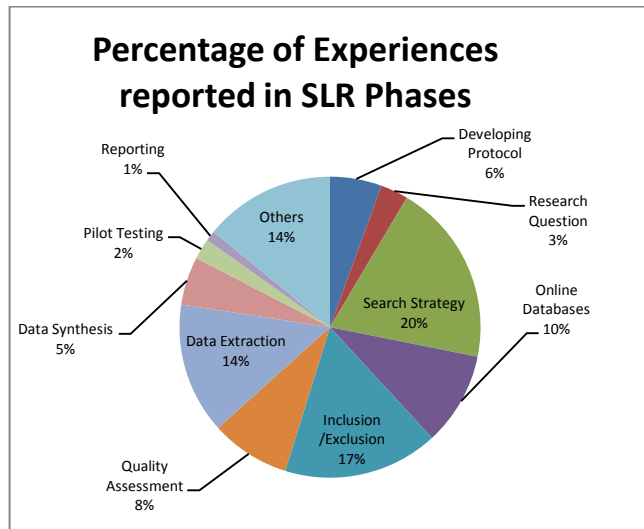


Figure 3: Percentage of Experiences reported in each Phase.

The frequency of different reported problems of search strategy phase are given in Table 2. Due to page limitation we are only presenting the frequency of experiences of the most problematic phase of SLR i.e. Search Strategy.

Table 2: Frequency of reported problems in Search Strategy Phase.

EXPERIENCES	FREQ.	REFERENCES
Careful selection of keywords, synonyms, alternative terms and avoiding homonyms: 1. Searched material depends on terms used to build the query; the coverage of papers can suffer if keywords have many synonyms and unknown alternatives which are not used. 2. Identify keywords and synonyms by reading	15	[1][16][25][26][33][38][46][58][64][69][74][76][77][80][86][87]

relevant papers 3. Exclude unrelated synonyms		
Pilot testing of search strings is beneficial to ensure representative evidence 1. Tradeoff required between generic and specific search string 2. Search strings should be tested if they retrieve known results 3. New strings made by reading reference of included study or headings of tutorials and databases	18	[16][19][54][58][68][74][75][78][79][80][81][84][85][87][88][89][90]
Combination of automatic and manual search can ensure coverage 1. Broad automatic search require more effort as it retrieves large number of papers which need to be filtered to find relevant material. 2. Manual restricted search requires less effort as it identifies relevant papers by omitting low quality paper, but manual search required too much effort and have probability of missing relevant evidence 3. Generic searches produce noise in results but good to ensure quality of SLR	13	[1][25][45][47][51][68][69][71][73][76][78][80]
Relevant papers retrieved if searched in title, abstract and keywords	1	[72]
Two phases strategy ensures representative set of empirical evidence 1. References of included studies should be scanned to identify secondary studies. 2. Contact authors of included studies to identify unpublished new relevant material	7	[13][16][38][60][62][82][87]
Limitations of online databases create problems during SLR	6	[1][12][20][60][73]

1. Some databases are not subscribed by organizations and institutes 2. Strings need to be customized for each database 3. Constraints and limitation of databases makes searching difficult		[79]
Retrieved results depend on searched Conferences/Journals, databases and indexing systems. 1. Use of SE indexing system e.g. Scopus and EI Compendex etc ensures coverage of evidence from less known journals and conferences. 2. Known related conferences and Journals should be separately searched	7	[61][74][76] [78][79][86][87]
Quality of abstract matters 1. No, bad quality or small abstract can be problematic for execution of SLR 2. Manual Search done in such cases	4	[22][82][70][91]
Search Strategy depends on Research Question	1	[14]
Tools that help locate, organize and summarize SLR information required	1	[63]
Including grey literature broadens research 1. Snow balling can be done to identify grey literature, white papers etc	3	[1][19][88]

The citations of the 116 included references are uploaded on <http://www.scribd.com/doc/124354851/A-Tertiary-Study-Experiences-of-Conducting-Systematic-Literature-Reviews-in-Software-Engineering>.

6. DISCUSSION

The results of the SLR show that both conferences as well as journals have useful evidence. Journal of Information and Technology; Evaluation and Assessment in Software Engineering and Empirical Software Engineering conferences have contributed most of the evidence. This information is useful for researchers working in relevant area. The year wise distribution of evidence shows the growing interest in the field. The experiences gathered from the included evidence highlight Search Strategy as the most problematic phase of SLR. Data Extraction and Inclusion/Exclusion phases also have many problems and require experience to affectively handle them. Some of the reported problems in this phase are difficulty to select keywords, identifying representative synonyms and alternative terms, limitations of online databases, ensuring coverage of relevant

material etc. the detail of which is given in Table 2. The results show that most of the evidence is implicitly reported while detailing SLR rather than specific experience reports. There is a need to explicitly report experiences and best practices of conducting SLR for future refinement of guidelines.

7. THREATS TO VALIDITY

Due to limited resources we are unable to claim that we have used all the available digital libraries such as Scopus etc. However, the digital libraries used are enough to generalize the findings in our study. It is probable that we may have missed some evidence due to limited resources so secondary search will be performed as part of future work to ensure coverage of the relevant material. To minimize publication bias we have manually searched the evidence based software engineering bibliography. During data extraction it is always hard to control personal bias. Data extraction and quality assessment have been performed independently by two researchers to remove personal bias. Conflicts were resolved via discussion with third researcher.

8. LESSONS LEARNED

The reported experiences highlight the following problems and best practices of conducting SLR:

Careful selection of keywords, synonyms and alternative terms is required to identify relevant results and to make good quality search strings. The following best practices can help solve this problem:

1. Pilot testing of search strings.
2. Studying of known articles and scanning the references to identify alternative terms and useful synonyms etc.
3. Tradeoff between generic and specific search strings (search strings that are too general broadens the search but it is very difficult to miss important evidence as well as they require more effort to filter out noise. Specific search strings may omit relevant evidence but requires less effort and have high precision. Pilot testing helps in making this tradeoff. Grey literature also broadens search and can be identified with the help of snowballing.
4. A two phased search strategy has positive influence on coverage of the evidence. Online databases, indexing systems and specific conferences and journals serve as reference points to start the search. Scanning references of included studies will increase coverage.
5. Limitations and constraints of online databases also hinder the search process of SLRs. The constraints such as number of words allowed, number of operators allowed and ability to limit search to a specific criteria etc. determines the effort and time spent on this phase.
6. Search results also depend on quality of abstracts:- Poor quality of abstracts and absence of abstracts create problems in the search.
7. There is a need to encourage the software engineering community on writing of structured abstracts, in order to retrieve relevant results.

9. CONCLUSION AND FUTURE WORK

Search strategy was identified as the most problematic and time consuming step of SLR. The limitations of online databases, lack of quality and structured abstracts along with difficulty of creating search strings are some of the most cited problems during search

strategy step. The guideline by Kitchenham [15] and definition of search strategy by Beecham et al in their protocol [3] details how to effectively make search strings from research question. Nonetheless the ability to make search strings which are neither generic nor too specific, having the right mix of major terms and synonyms is not a trivial task. The search string need to be split into multiple smaller search strings due to constraint of size and Boolean operators imposed by online databases. This further complicates the problem. The format or syntax of the search string is also varies for online databases, thereby requiring customization of search strings for different databases. The web as well as help provided on the homepage of these online databases gives instructions for correct syntax of search strings. Experience shows that a lot of hit and trial testing is needed before the strings are properly customized and ready to run. Automated tool support for customization of the search strings for different databases can help in saving time and effort.

The difficulties to save and manage retrieved results also contribute towards complexity of searching phase. Online databases support different formats of exporting citation, where some do not support export at all. The manual saving of retrieved evidence one by one is a cumbersome and laborious work especially when the number is huge. A generic portal which searches and manages references from multiple databases can be of immense value to researchers. Tool support that facilitates in organizing and managing SLR information is also required.

The work will be extended by performing secondary search (by reviewing references of included evidence), to increase the coverage of the area. More detailed analysis will be performed once data extraction from secondary search is also completed. The remaining steps of thematic synthesis will be applied to come up with a model for the themes.

The study only reports on the most problematic phase of SLR, similar analysis and synthesis will be performed for remaining phases.

References

- [1] Babar, M. A., and Zhang, H., 2009, Systematic Literature Reviews in Software Engineering: Preliminary Results from Interviews with Researchers, *International Conference on Empirical Software Engineering and Measurement*, pp. 346-355.
- [2] Bailey, J., Zhang, C., Budgen, D., Charters, S., and Turner, M., 2007, Search Engine Overlaps: Do they Agree or Disagree?, *Second International Workshop on Realizing Evidence-Based Software Engineering*, p. 2-2.
- [3] Beecham, S., Baddoo, N., Hall, T., Robinson, H., and Sharp, H., 2006, Protocol for a Systematic Literature Review of Motivation in Software Engineering.
- [4] Brereton, O. P., Kitchenham, B. A., Budgen, D., Turner, M., and Khalil, M., 2007, Lessons from Applying the Systematic Literature Review Process within the Software Engineering Domain, *Journal of Systems and Software*, vol. 80, no. 4, pp. 571-583.
- [5] Cruzes, D.S and Dyba, T., 2011, Recommended Steps for Thematic Synthesis in Software Engineering, *International Symposium on Empirical Software Engineering and Measurement*, pp. 275-284.
- [6] Dieste, O., and Padua, O. A. G., 2007, Developing Search Strategies for Detecting Relevant Experiments for Systematic Reviews, *Proceedings of First International Symposium on Empirical Software Engineering and Measurement*, pp. 215-224.
- [7] Dybå, T., and Dingsøyr, T., 2008, Strength of Evidence in Systematic Reviews in Software Engineering, *Proceedings of the Second ACM-IEEE international symposium on Empirical software engineering and measurement*, pp. 178-187.
- [8] Dyba, T., Dingsøyr, T., Hanssen, G. K., and SINTEF, T., 2007, Applying Systematic Reviews to Diverse Study Types: An Experience Report, *First International Symposium on Empirical Software Engineering and Measurement, 2007*. pp. 225-234.
- [9] Kitchenham, B., Brereton, O. P., Budgen, D., Turner, M., Bailey, J., and Linkman, S., 2009, Systematic Literature Reviews in Software Engineering-A Systematic Literature Review, *Information and Software Technology*, vol. 51, no. 1, pp. 7-15.
- [10] Kitchenham, B. A., Budgen, D. and Brereton, O. P., 2006, Evidence-based Software Engineering and Systematic Literature Reviews, *Product-Focused Software Process Improvement*, p. 3-3, DOI=10.1007/11767718_3.
- [11] Kitchenham, B., and Charters, S., 2007, Guidelines for performing Systematic Literature Reviews in Software Engineering', Technical Report EBSE 2007-001, *Keele University and Durham University Joint Report*.
- [12] Kitchenham, B. A., Dyba, T., and Jorgensen, M., 2004, Evidence-Based Software Engineering, *Proceedings of 26th International Conference on Software Engineering*, pp. 273-281.
- [13] Kitchenham, B., Pretorius, R., Budgen, D., Brereton, O. P., Turner, M., Niazi, M., and Linkman, S., 2010, Systematic Literature Reviews in Software Engineering-A Tertiary Study, *Information and Software Technology*, vol. 52, no. 8, pp. 792-805.
- [14] Riaz, M., Sulayman, M., Salleh, N., and Mendes, E., 2010, Experiences Conducting Systematic Reviews from Novices' Perspective, *Proceedings of 14th International Conference on Evaluation and Assessment in Software Engineering (EASE)*.
- [15] Staples, M. and Niazi, M., 2007, Experiences using systematic review guidelines, *Journal of Systems and Software*, vol. 80, no. 9, pp. 1425-1437.
- [16] Silva, F. Q. B., Santos, A. L. M., Soares, S., França, A. C.C., Monteiro, C. V. F., and Maciel, F. F., 2011, Six years of Systematic Literature Reviews in Software Engineering: An Updated Tertiary Study, *Information and Software Technology*, vol. 53, no. 9, pp. 899-913.
- [17] MacDonell, S., Shepperd, M., Kitchenham B. A., and Mendes, E., How reliable are systematic reviews in empirical software engineering?, *IEEE Transactions on Software Engineering*, 2010, vol. 30, no. 5, pp. 676-687.