

**Автономная некоммерческая организация высшего образования
«Университет Иннополис»**

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(БАКАЛАВРСКАЯ РАБОТА)
по направлению подготовки
09.03.01 - «Информатика и вычислительная техника»**

**GRADUATION THESIS
(BACHELOR'S GRADUATION THESIS)**

**Field of Study
09.03.01 – «Computer Science»**

**Направленность (профиль) образовательной программы
«Информатика и вычислительная техника»
Area of Specialization / Academic Program Title:
«Computer Science»**

**Тема /
Topic**

SciLRtool: Онлайн инструмент, поддерживающий все этапы систематических обзоров литературы в программной инженерии / SciLRtool: Online Tool Supporting All Stages of Systematic Reviews in Software Engineering

Работу выполнил /
Thesis is executed by

**Гинзбург Данил Маркович /
Ginzburg Danil**

подпись / signature

Руководитель
выпускной
квалификационной
работы /
Supervisor of
Graduation Thesis

**Конюхов Иван
Владимирович /
Konyukhov Ivan**

подпись / signature

**Автономная некоммерческая организация высшего образования
«Университет Иннополис»**

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(БАКАЛАВРСКАЯ РАБОТА)
по направлению подготовки
09.03.01 - «Информатика и вычислительная техника»**

**GRADUATION THESIS
(BACHELOR'S GRADUATION THESIS)**

**Field of Study
09.03.01 – «Computer Science»**

**Направленность (профиль) образовательной программы
«Информатика и вычислительная техника»
Area of Specialization / Academic Program Title:
«Computer Science»**

**Тема /
Topic**

SciLRtool: Онлайн инструмент, поддерживающий все этапы систематических обзоров литературы в программной инженерии / SciLRtool: Online Tool Supporting All Stages of Systematic Reviews in Software Engineering

Работу выполнил /
Thesis is executed by

**Гинзбург Данил Маркович /
Ginzburg Danil**

подпись / signature

Руководитель
выпускной
квалификационной
работы /
Supervisor of
Graduation Thesis

**Силлитти Альберто /
Sillitti Alberto**

подпись / signature

Contents

1	Introduction	9
1.1	Domain Area and Applicability	9
1.2	Problem Statement	10
1.3	Proposed Solution	11
2	Literature Review	12
2.1	Systematic Literature Reviews	12
2.1.1	The Importance of Systematic Literature Reviews	13
2.1.2	Why Do a Systematic Review?	13
2.1.3	Differences of SLR from Conventional Literature Review	14
2.2	Systematic Reviews in Software Engineering	14
2.3	Software Tool Supporting Systematic Reviews	15
2.3.1	What Can Be Automated?	16
2.4	Related works	17
3	Methodology	19
3.1	Setting Up the Review	21
3.1.1	Define the Question Type (PICO, PIT, PO)	21
3.1.2	Define if SR or SM Will Be Performed	22
3.2	Pi Scoping/Protocol Development	22

3.2.1	Formulate the Review Question	22
3.2.2	Write Protocol	23
3.2.3	Devise Search Strategy	24
3.3	Literature Searching	25
3.4	Duplicate Checking	26
3.5	Article Screening/Study Selection	28
3.6	Quality Assessment	30
3.6.1	Planning	31
3.6.2	Conducting	32
3.7	Data Extraction	33
3.7.1	planning	34
3.7.2	conducting	34
3.8	Quantitative and Qualitative syntheses of results	35
3.8.1	Qualitative synthesis	35
3.8.2	Quantitative synthesis	36
3.9	Generation of Documentation	38
3.9.1	Protocol Reporting	39
3.9.2	Final Review Reporting	40
4	Implementation	42
4.1	Technology Adaption	42
4.2	Setting Up the Review	43
4.2.1	Define the Question Type (PICO, PIT, PO)	43
4.2.2	Define if SR or SM Will Be Performed	43
4.3	Literature Searching	43
4.4	Duplicate Checking	44
4.5	Quality Assessment	44

4.5.1	Planning	44
4.5.2	Conducting	47
4.6	Quantitative and Qualitative Synthesis	50
4.7	Generation of Documentation	52
4.7.1	Documentation Interface	52
4.7.2	Publishing Evidence Synthesis	53
5	Evaluation and Discussion	56
5.1	Evaluation	56
5.1.1	Setting Up the Review and Protocol Definition	57
5.1.2	Literature Searching	58
5.1.3	Duplicate Checking	58
5.1.4	Study Selection	59
5.1.5	Quality Assessment	59
5.1.6	Data Extraction	60
5.1.7	Data Analysis	60
5.1.8	Generation of Documentation	60
5.1.9	Publishing Evidence Synthesis	61
5.2	Discussion	61
6	Conclusion	63
	Bibliography cited	65
A	Existing Steps For Systematic Reviews	70
B	PRISMA documents	72
C	SciLRtool improvements over Parsifal	74

List of Tables

I	Related tools	18
II	SciLRtool improvements over Parsifal shortly described for each stage with corresponding information formats.	75

List of Figures

- 3.1 Parsifal Tool: Review Details 20
- 3.2 Parsifal: Research Questions and PICOC 23
- 3.3 Parsifal literature searching, search results have no links to the articles 26
- 3.4 Parsifal duplicate checking. Two articles are identical, but the word Disease is capitalized in the second article; thus, the article will be marked as duplicated 28
- 3.5 Parsifal study selection tool. Red lines and red text are added to divide features logically. 30
- 3.6 Parsifal article details example 31
- 3.7 Quality Assessment Questions and Answers in Parsifal and Cadima 32
- 3.8 Conflict System Example: User 1 and User 2 answer the same question for the same included study differently 33
- 3.9 Parsifal data analysis; source-studies distribution example. 2 studies per source were chosen. 36
- 3.10 Parsifal data analysis; accepted vs rejected number of studies for every source example. 2 studies chosen and 1 accepted per source. 37

3.11 Parsifal data analysis; publication year example. Out of 3 accepted studies, 1 has 2009 pub.y., 1 has 2018 pub.y. and 1 has 2019 pub.y. 37

3.12 Parsifal reporting 38

4.1 Quality Assessment checklist new interface 45

4.2 Quality Assessment checklist example of a new modal window to add a new quality question with answers. 46

4.3 Quality Assessment checklist example of settings for main author. 47

4.4 Quality Assessment: conducting. 48

4.5 Quality Assessment: manual reassignment of article. 49

4.6 Quality Assessment: conflicts example. 50

4.7 Interface for quantitative and qualitative synthesis 51

4.8 New reporting stage, export tab 53

4.9 New Browse navbar-menu 54

5.1 The Likert scale applied in interviews 56

A.1 Existing steps for systematic reviews (possible to have some deviations) [9] 71

B.1 PRISMA flow diagram template 72

B.2 PRISMA report-assessment checklist 73

Abstract

Systematic Literature Review (SLR) is a comprehensive literature review summarising all available research relevant to a particular domain area; it is applied to understand a domain and establish a possible domain gap. Consequently, some tools exist to support the process of conducting SLR's. We investigated that no existing tool provides support for all stages of SLR in the Software Engineering area; thus, we decided to contribute to this field by creating a new tool called SciLRtool. Our tool combines best practices of such tools as Parsifal and CADIMA and proposes its unique features. We evaluated our system by interviewing 11 people experienced with SLR's. According to results, SciLRtool is estimated as "useful" in the practice of experts. However, the competitiveness of SciLRtool with regards to other tools is yet to be estimated.

Chapter 1

Introduction

This document examines the domain of Systematic Literature Reviews in Software Engineering and describes the implementation of a new tool supporting Systematic Literature Reviews - SciLRtool. Then it discusses the obtained results and proposes future work.

1.1 Domain Area and Applicability

Many research works are available nowadays, which differ in quality, contribution, and scientific value. It becomes crucial to identify the most relevant research works for a specific problem. Often new research starts with a literature review. Nevertheless, it has little scientific value unless a literature review is fair and thorough [1]. Recently, a new problem area raised that studies Systematic Literature Reviews (SLR). SLR is a secondary study that summarises all available studies in a particular research area in a fair manner. It is helpful to identify existing gaps in a research domain and examine a background to propose a new research activity [1]. We discuss SLR's in more details in the Literature Review chapter 2

Consider the following example. A team of post-graduate students study the agile development processes. They want to obtain a comprehensive understanding of this domain. They can do a thorough literature review by themselves - read a large number of papers and select the most relevant works from them. In contrast, they can utilize the existing Systematic Literature Review for Agile Development Processes, and User Centred Design Integration by Salah et al. [2] that summarises the most relevant works, including state-of-art solutions. This SLR gives a comprehensive overview of the domain area, groups studies by pre-defined classes, provides quality assessment for every work presented and determines gaps in the domain area. In our example, the team of post-graduate students can quickly recognize that Lack of Documentation is the primary gap found by Salah et al. that needs further improvements.

1.2 Problem Statement

Clearly, there are some tools available that support researchers with doing Systematic Literature Reviews. The most notable examples are: EPPI-Reviewer 4 [3] and CADIMA [4]. The tools provide automated solutions to different problem areas in SLR's: effective team collaboration, protocol and report generation, duplicate checking, quantitative data representation and others. However, some of them focus on concrete features and provide powerful functionality for them (e.g. EPPI-reviewer facilitates the quantitative and qualitative synthesis of data), while others are dedicated to specific domains (e.g. medicine).

From the thorough analysis of existing tools, we found that not a single tool applied in the Software Engineering domain provides a solution to every problem area existing. The main reason is that the Software Engineering

domain drastically differs from the medicine domain [1] for which SLR was initially developed. It is also vital that Software Engineering is a considerably new scientific domain.

1.3 Proposed Solution

In our project, we aim at developing a web tool supporting all stages of Systematic Literature Reviews in Software Engineering. For this purpose, we examined several existing tools to create a new one that combines the best qualities of other tools. Therefore, we introduce SciLRtool - a tool based on the Parsifal [5], which is solely made by Vitor Freitas that focuses on the Software Engineering domain and provides open-source code. Parsifal features literature searching and facilitates the quantitative synthesis of results.

We aim at creating our tool that features Parsifal and CADIMA best practices and also proposes its unique features. We utilize the best solutions from the CADIMA [4] - the giant in a world of SLR's, which provides the ability to develop SLR's in any domain. CADIMA features the quality assessment of research works and the generation of documentation with publishing documents to be publicly available. We consider all stages and approaches of the SLR in the Methodology chapter 3. In the Implementation chapter 4 we describe the development process of the SciLRtool in details.

Chapter 2

Literature Review

This chapter describes the systematic literature reviews field, its applicability in the software engineering discipline and software tools designed to support it. Section 2.1 and its subsections are dedicated to systematic literature review, its importance (2.1.1), difference from conventional literature review (2.1.3) and reasons to undertake it (2.1.2). The second section 2.2 mentions the applicability of systematic reviews in the software engineering field. The third section 2.3 explains the need for an automated process of systematic reviews, makes a brief overview of the review process and what parts of it can be automated (2.3.1). Finally, section 2.4 discusses the existing tools that support systematic reviews.

2.1 Systematic Literature Reviews

"A systematic review attempts to collate all the empirical evidence that fits pre-specified eligibility criteria to answer a specific research question" [6]. A systematic review is a secondary study that summarises all available studies in a particular research area.

2.1.1 The Importance of Systematic Literature Reviews

The start of every research is to examine some research area and write a literature review chapter. However, if the literature review is not fair and thorough, it is of little scientific value. This problem necessitates a systematic approach to literature reviews for it to be fair. Such an approach is straightforwardly called Systematic Literature Review. SLR is fair and seen to be fair because it requires researchers to follow a predefined protocol and search strategy. For example, the search strategy is formulated so that every reader of a systematic review related paper must be able to assess the completeness of the search. Most importantly, researchers that undertake a systematic review must report all research that does not support their chosen research hypothesis as well as reporting research that does. Otherwise, a systematic review is unfair and considered to be pseudoscience.

"True ignorance is not the absence of knowledge, but the refusal to acquire it."

Karl R. Popper, "In Our Time's Greatest Philosopher Notes"

2.1.2 Why Do a Systematic Review?

There are many particular reasons to perform a systematic review. First of all, to review and identify current and outgoing studies to indicate specific gaps in knowledge and research area or lack of evidence. Secondly, to summarise the up-to-date evidence about specific methodology or technology; this might be used, for example, to provide a background for those methodologies or technologies to position a new research activity. Although writing systematic literature reviews is a highly time-consuming process, it is often rewarding.

They allow researchers to identify priorities for further research.

2.1.3 Differences of SLR from Conventional Literature Review

The main difference between a systematic review and a conventional literature review is a review protocol that specifies the research question and the methodology of performing a review. Furthermore, systematic reviews specify particular search strategies so that readers can assess the completeness of the search and replicate it if needed. Also, systematic reviews require inclusion and exclusion criteria because not all the studies found by the search are helpful for the research purpose. Besides inclusion and exclusion criteria, systematic reviews are more flexible in terms of information extracted from the studies; also, they specify quality criteria by which to evaluate the studies.

2.2 Systematic Reviews in Software Engineering

The systematic literature review is one of the primary methodologies of Evidence-Based Software Engineering [8]. A systematic review is an evidence-based approach that originates from the medical field. However, the medical systematic review approach is not appropriate for software engineering researchers. The protocol for the software engineering field is well defined by Kitchenham et al. (2007) [1] for systematic review process as we concentrate on the software engineering field. Budgen et al. (2006) [7] conducted several interviews with researchers to compare evidence-based approaches in different fields, and results showed that the agreement between clinical medicine methodology and software engineering methodology is 0.17 [1]. This exper-

iment demonstrates how software engineering is different nowadays from the medical area.

2.3 Software Tool Supporting Systematic Reviews

Software tools have been developed to support researchers during the systematic review process (they are also applicable for systematic maps, which are similar to systematic reviews in terms of rigour protocol and search strategy; however, they do not provide quantitative and qualitative analyses of the studies [8]). Software tools provide increased efficiency for the reviewing team throughout the conduct of their review. Nevertheless, there appear potential downsides: some tools are aimed at particular research disciplines (e.g. medicine) and are not applicable for others. It is also possible they are not open-access. It is worth mentioning that some software tools might be oriented solely on systematic maps and does not provide systematic reviews features.

Kitchenham et al. published an interesting document in 2008 that shows the systematic review activity in software engineering from 2004 till 2008. In this period, 20 systematic review related papers were published. However, only half of them positioned themselves as related to evidence-based software engineering [6]. Moreover, the number of studies done every year is steady, and the quality is consistently improving. Although many researchers prefer to undertake informal and manual literature reviews, the need for an online tool supporting SLR is growing.

2.3.1 What Can Be Automated?

The review protocol can be split up into three main stages: Planning the Review, Conducting the Review and Reporting the Review. The software tools should deal with Conducting the Review and Reporting the Review stages. One example of a systematic review conducted in the software engineering field that follows the protocol specified by Kitchenham et al. (2007) [1] is Dina Salah et al. (2014) [2]. However, the authors conducted their review manually, with no use of external tools. It is clear how much hard work was done during the review since the authors provide detailed manual explanations of the Data Extraction/Synthesis methods and search results from digital libraries, conference proceedings and Journals. All those methods and search results can be auto-generated by special tools such as the one explained in this document.

To understand which parts of Conducting the review and Reporting the review can be automated, it is necessary to dive into the systematic review process (see Figure A.1).

The process itself is partly technical and partly creative [9]. For example, the creation of the research question(s) and the review protocol is a creative task: that is the part of the review where a team of reviewers should utilize their experience and creativity. Usually, peer-review is used to develop the protocol to ensure objectivity and fulfilment of the review question(s) [6].

Once the protocol is defined, now it can be executed by a machine [10]. Tasks are ordered in such a way that manual tasks come first, and automated tasks come second. It is also beneficial for reviewers to monitor and assure the quality of the review during the execution of technical tasks. Some tasks are impossible or seem to be impossible to automate. However, the development of software tools is incremental [9], and what seems a fantasy now might be

implemented in a few decades.

2.4 Related works

The related work is based on related work published by Kohl et al. (2018) [4] since it gives a complete and comprehensive overview of online tools available. This publication describes the new tool supporting systematic reviews and systematic maps, which is called CADIMA. The authors did a great job searching for existing solutions. Their search strategy includes:

- searches via online databases;
- searches via links in relevant websites;
- relevant publications searches.

Excluding tools that are not free to use, currently in development or no longer available, 22 remaining software tools were identified. However, only 3 out of 22 tools are designed primarily for the Software Engineering field, and nine are suitable for any research field. The rest is designed for medical science and experimental animal studies and are not considered to be related tools for this document. Thus, 12 remaining tools suit the definition of related tools. Nevertheless, 5 of them are not available online, meaning they are downloadable applications. Finally, we are left with a total of 7 similar tools in terms of purposes and availability (see Table I).

Table I: Related tools.

name	field	stages	open source
CADIMA [4]	Any	Qu, Pi, Du, Sc, Co, Cr, Do	No
Colandr	Any	Pi, Se, Du, Sc, Co, Sy, Do	Yes
DistillerSR	Any	Se (PubMed), Du, Sc, Co, Sy, Do	No
EPPI-Reviewer 4 [3]	Any	Se, Du, Sc, Co, Cr, Sy, Do	No
PARSIFAL [5]	SE	Pi, Se, Du, Sc, Co, Sy	Yes
Rayyan [11]	Any	Pi, Se, Du, Sc	No
SESRA	Any	Qu, Pi, Sc, Co, Sy, Do	No

The tools differ in features they support, and, most importantly, in stages¹ of SR. It is crucial to emphasize that none of them supports all stages. Some of them concentrate on particular features such as machine learning during screening, data extraction or synthesis stages and bias assessments. For example, EPPI-reviewer 4 [4] provides the article screening feature, and Distiller SR [5] provides capabilities of different character sets management. Several existing solutions are not free to use (e.g. Distiller SR) and provide subscription plans. Furthermore, all of the solutions are designed in English and provide documentation (e.g. Rayyan [6] has only an online form as user support).

¹Stages of a systematic review: Qu setting up the review, with question formulation and/or stakeholder engagement, Pi scoping/pilot study, protocol development (e.g. PICO elements specified), Se literature searching (e.g. via integration with publication databases), Du duplicate checking (e.g. automated marking of duplicates, or identification of potential duplicates for manual checking), Sc article screening/study selection, Co facilitates data coding/tagging and extraction to support meta-analyses, Cr critical appraisal/risk of bias assessments, Sy facilitates quantitative/ qualitative syntheses of results, Do generation of documentation/output of text, figures or tables to assist with report writing" from Kohl et al. (2018) Table 1 [4].

Chapter 3

Methodology

Among several publicly available open-source SLR tools (namely, Colandr and Parsifal), we chose Parsifal [5] to be my starting point of SciLRtool. It is a system supporting Systematic Literature Reviews, which is dedicated to the Software Engineering field that follows Kitchenham et al. [1] protocol (see Figure 3.1). It implements many features and supports many SR stages: Pi, Se, Du, Sc, Co, Sy (see Table I).

Along with Parsifal, Cadima [4] supports the following SR stages: Qu, Pi, Du, Sc, Co, Cr, Do (see Table I). The synthesis of those tools will result in all existing stages of SR: Qu, Pi, Se, Du, Sc, Co, Co, Cr, Do. Furthermore, in contrast with Parsifal, Cadima provides better usability: help text for features, more flexible protocol setup and others. Therefore, SciLRtool is a synthesis of these two tools and implements the best parts of each.

The list of all SLR stages (see Table I) (the stages marked as "done" are already present in Parsifal) is as follows:

- Setting up the review
- Pi scoping/protocol development

The screenshot shows the Parsifal Tool interface for a review. At the top, there is a navigation bar with the Parsifal logo, links for Blog, About, and Help, and the user's name Daniel Ginzburg with settings and share icons. Below this, the user's name is repeated, and a 'Review settings' button is visible. A horizontal menu contains four tabs: Review (selected), Planning, Conducting, and Reporting. The main content area is divided into two sections. The 'Review details' section on the left has a 'Title' input field and a larger 'Description' input field, with a green 'Save' button at the bottom. The 'Authors' section on the right lists 'Daniel Ginzburg' as the 'main author' and includes a '+ Add author' button. The footer contains the copyright notice '© 2014-2018 Parsifal Ltd.' and social media icons for GitHub, Twitter, YouTube, and Google+.

Figure 3.1: Parsifal Tool: Review Details

- Literature searching
- Duplicate checking
- Article screening/study selection
- Quality Assessment
- Data extraction
- Quantitative and qualitative syntheses of results stages
- Generation of documentation

In the following sections, we will discuss every stage in details.

3.1 Setting Up the Review

The planning part of any Evidence Synthesis consists of Setting up the Review and Protocol definition.

According to Cadima analysis, Parsifal lacks setting up the Review stage (Qu stage). However, it is only partially accurate since Parsifal implements a small number of features regarding this stage. Cadima defines the following features of the Qu stage (marked features are present in Parsifal):

- Invite registered users to become part of the review team
- Define the title of the Review
- Define the question type (PICO, PIT, PO)
- Define if an SR or an SM will be performed

3.1.1 Define the Question Type (PICO, PIT, PO)

Parsifal allows users to define PICOC - population, intervention, comparison, outcomes and context. Nonetheless, other types of research questions exist and require different approaches, such as PIT or PO. They could be applied in different circumstances. For example, if a research question is related to the accuracy of a test method, PIT (population, index test and target condition) should be utilized. PO (population and outcome) the critical elements in case questions are related to outcomes for a population [12].

3.1.2 Define if SR or SM Will Be Performed

In contrast with Parsifal, Cadima provides tools for Systematic Mapping Approach [8], which is a simplified way of doing a Systematic Review. The systematic Mapping option allows users to skip some fields (e.g. Quality Assessment). This option is implemented in SciLRtool - it adds more functionality to the system and expands the target user base.

3.2 Pi Scoping/Protocol Development

This stage is basically the preparation part of the review (see Figure A.1). It consists of **formulate review question, find previous SR, write the protocol and devise search strategy** tasks. Although those tasks are required to undertake, only a few of them can be automated. Parsifal allows users to record the review question, write the protocol and devise the search strategy.

3.2.1 Formulate the Review Question

There are many possible ways to formulate the review questions [13]. Parsifal is a Software Engineering dedicated tool; therefore, the review topic is constrained to the SE area. However, such factors as proficiency in the area and personal interest are common [13]. Research questions must be explained in detail to avoid ambiguity and help with the quality assessment stage. PICO (population, intervention, comparison, outcomes) elements were recommended as default specification of any research question [14].

Parsifal has an elegant way of writing research question and PICOC key-

words(PICOC is an extended version of PICO with Context) (see Figure 3.2)

The image shows two panels from the Parsifal interface. The top panel is titled 'PICOC' and contains instructions: 'Separate the terms used in the PICOC using commas. This will make possible to save them separately as keywords so we can help you design your search string.' and 'If any of the sections of PICOC doesn't apply to your research, please leave it blank.' Below the instructions are five input fields: 'Population' with 'P', 'Intervention' with 'I', 'Comparison' with 'C', 'Outcome' with 'O', and 'Context' with 'C'. A green 'Save' button is at the bottom left of this panel. The bottom panel is titled 'Research Questions' and features a large empty text input field. To the right of the input field are two buttons: a green 'save' button and a white 'cancel' button. Below the input field is a blue '+ Add Question' button.

Figure 3.2: Parsifal: Research Questions and PICOC

3.2.2 Write Protocol

After formulating the review question and finding previous SR to establish that the current SR is needed, the next step in planning the review is protocol writing. This task requires expertise in the research area and creativity because researchers need to have a general idea about the research outcomes. To ensure unbiasedness and consistency of the review, peer review is used.

Parsifal implements the following features of writing the protocol:

- objectives
- selection criteria

3.2.3 Devise Search Strategy

A good search strategy is not limited to only easily accessible studies. It describes what keywords will be used in the searches, which databases will be searched, how non-database sources will be tracked and checked for trustworthiness [6]. It is also a good practice to have a peer-review of the search strategy before searching.

Parsifal provides the ability to write the following parts of the search strategy:

- keywords and synonyms - users can specify keywords, synonyms, and how they are related to PICOC.
- search string - user can define a search string using words, boolean operators AND and OR, parentheses to logically separate the keywords and synonyms and double quotes for composite words.
- sources - user can specify databases (integrated databases: El Compendex, IEEE Digital Library, ISI Web of Science, Science@Direct, Scopus, Springer Link) and other sources.

Overall, Parsifal has an excellent implementation of the protocol definition stage; thus, SciLRtool has not changed it.

3.3 Literature Searching

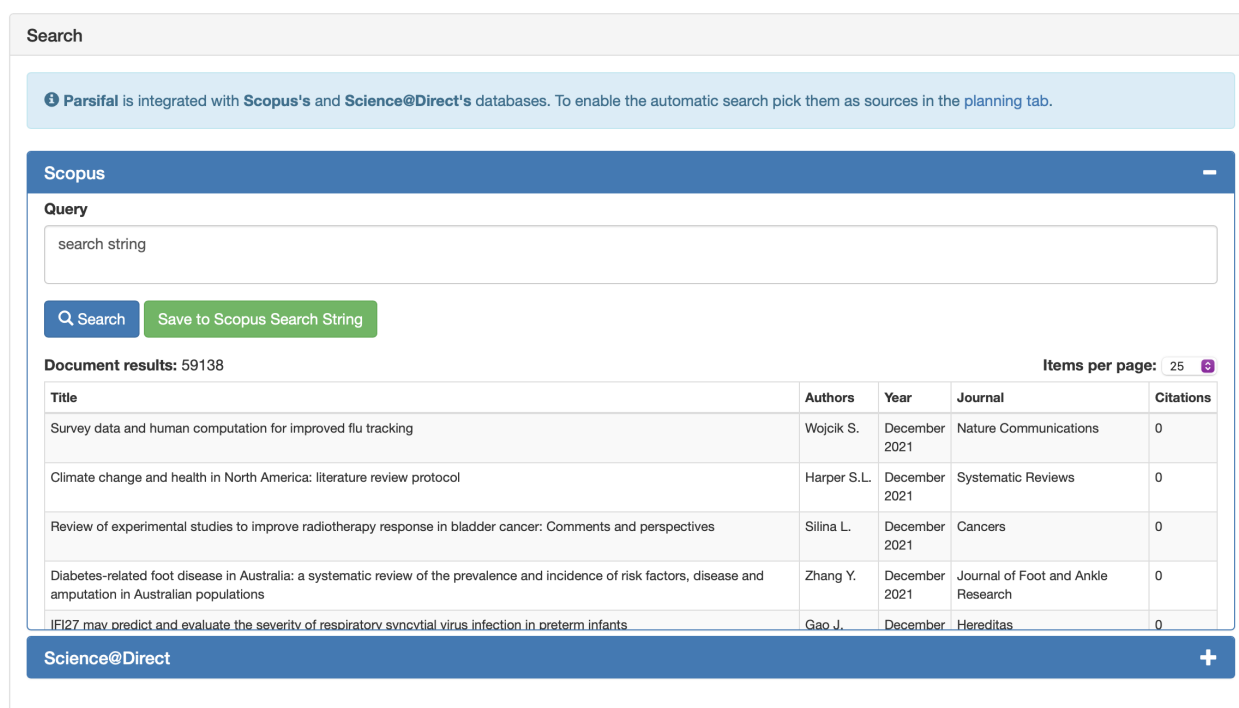
Mainly, online databases, such as Science@Direct or Digital Library, are used nowadays. However, grey literature and other sources might be utilized as well [15]. For the literature review to be systematic, all the relevant studies must be invoked; thus, multiple databases have to be searched. However, interoperability among databases is relatively rare [9]. For example, different databases may support different query languages (e.g. AND, OR and NOT), the syntax for referencing specific fields and operations (e.g. ADJ or NEAR). Bearing in mind the enumerated factors, researchers may struggle with Literature Searching.

Parsifal allows users to define a unique search string per source. In addition, Parsifal helps users search for scientific studies. It is integrated with Elsevier: "Elsevier is a leader in information and analytics for customers across the global research and health ecosystems" [16]. It provides an API for searching over 500000 articles annually in 2500 journals. Although it is not perfect and it has good alternatives, such as AIP, IOP, or Springer [17], it still covers a tremendous number of articles and provides fast and simple API endpoints. Parsifal utilizes two APIs from Elsevier: Scopus API and Science@Direct API.

- Scopus is an abstract and citation database that includes trade publications, conference proceedings, patent records, peer-reviewed literature and Web sites. It has the cited references of studies from 1996 forward and provides author and article citation data [18].
- Science@Direct is a large bibliographic database that provides over 18 million pieces of content from more than 4000 journals, and 30000 e-books from Elsevier [19]. Access to the full text requires a subscription;

however, Science@Direct provides open access to some studies.

Since Parsifal is integrated with Scopus and Science@Direct, users can find the most relevant studies to their research in one place, without using other systems except Parsifal. However, a noteworthy drawback we discovered is that the result returned by the search is plain text and is not clickable (see Figure 3.3). Users need to find already found articles once more on the Internet. Although Elsevier APIs provide links to the articles, Parsifal does not include them in the search result.



The screenshot shows the Parsifal search interface. At the top, there is a 'Search' header. Below it, a blue banner states: 'Parsifal is integrated with Scopus's and Science@Direct's databases. To enable the automatic search pick them as sources in the planning tab.' The main section is titled 'Scopus' and contains a 'Query' input field with the placeholder text 'search string'. Below the input field are two buttons: 'Search' and 'Save to Scopus Search String'. The search results are displayed as a table with the following data:

Title	Authors	Year	Journal	Citations
Survey data and human computation for improved flu tracking	Wojcik S.	December 2021	Nature Communications	0
Climate change and health in North America: literature review protocol	Harper S.L.	December 2021	Systematic Reviews	0
Review of experimental studies to improve radiotherapy response in bladder cancer: Comments and perspectives	Silina L.	December 2021	Cancers	0
Diabetes-related foot disease in Australia: a systematic review of the prevalence and incidence of risk factors, disease and amputation in Australian populations	Zhang Y.	December 2021	Journal of Foot and Ankle Research	0
IFI27 may predict and evaluate the severity of respiratory syncytial virus infection in preterm infants	Gao J.	December	Hereditas	0

At the bottom of the interface, there is a 'Science@Direct' section with a plus sign icon.

Figure 3.3: Parsifal literature searching, search results have no links to the articles

3.4 Duplicate Checking

The purpose of duplicate checking is to detect two separate reports of the same study. This step is required to undertake whenever combining the

obtained citations [20]. Duplicates appear due to variations of indexed metadata (e.g. DOI, ISBN and page numbers might not be included) or typos (in the article title or journal name).

In case the same study is reported more than once - due to variation in author lists, titles or different journals - all those studies should be cited but marked as one trial in meta-analysis [21]. Citation data are not enough to detect such duplicates, so the article's text is required.

To deal with duplicates, Parsifal has a mere duplicate checking engine. It checks if some of the included articles have the same title based slug, which is a detection mechanism for wrong cases (lowercase/uppercase), unnecessary or wrong punctuation marks or extra white spaces (see Figure 3.4).

While detection of wrong cases is a required step to undertake to detect duplicates, there should be some matching technique to detect typos and misspellings in article titles [20]. According to Elmagarmid et al. [20], there is a vast number of such techniques, and they are split into the following groups: Character-Based Similarity Metrics, Token-Based Similarity Metrics, Phonetic Similarity Metrics and Numeric Similarity Metrics. However, Elmagarmid et al. reference the work of Bilenko et al. [22], who compare the effectiveness of different metrics and come to the conclusion that SoftTF.IDF metric, which is a token-based similarity metric, works better than any other metric overall. Although Bilenko et al. emphasize that no single metric is appropriate for all data sets SoftTF.IDF has shown itself to be the best. Thus, SciLRtool achieved it.

Duplicates ×

Resolve Duplicates

Click on **Resolve Button** to change the article status to Duplicated

Title	Authors	Year	Source	Status	Action
Coronavirus Disease (COVID-19) prevention and treatment methods and effective parameters: A systematic literature review	Rahmani, Amir Masoud and Mirmahaleh, Seyedeh Yasaman Hosseini	2021	Scopus	Unclassified	Resolve
Coronavirus disease (COVID-19) prevention and treatment methods and effective parameters: A systematic literature review	Rahmani, Amir Masoud and Mirmahaleh, Seyedeh Yasaman Hosseini	2021	ACM Digital Library	Accepted	Resolve

Cancel
Resolve All

Figure 3.4: Parsifal duplicate checking. Two articles are identical, but the word Disease is capitalized in the second article; thus, the article will be marked as duplicated

3.5 Article Screening/Study Selection

This is the so-called appraisal stage that corresponds to tasks "screen abstracts" and "screen full text" according to Tsafnat et. al. [9] (see Figure A.1). These tasks aim to exclude all irrelevant studies. When the literature searching part is done right, most commonly, the vast majority of articles are removed [6].

In the first part of study selection (screen abstracts task), only titles and abstracts exclude irrelevant studies. Usually, this part of the excluded articles is the biggest. In the second part (screen full-text task), the entire text of the articles (not excluded by the first task) is used to select studies.

Parsifal provides a uniform tool supporting these two tasks simultaneously. This tool has a great feature set, as shown in Figure 3.5. The figure is divided into logical blocks from 1 to 6:

1. This block allows sorting by chosen sources. Default is the All Sources that includes aggregated articles from all sources.
2. The second block allocates the button which opens a modal dialogue to find and resolve duplicates (see subsection 3.4), and a button which exports articles in .slx format in the form of a table with all relevant fields (see item 6 for more details).
3. The third block allows choosing the following actions to perform on articles: mark as accepted, mark as rejected, mark as duplicated and remove selected. Then the chosen action will be performed after clicking on the "Go" button.
4. The fourth block allows sorting of articles by their status: accepted, rejected, unclassified and duplicated.
5. The fifth block allocates the table of articles only with the most important subset of fields. Users can select/deselect all articles and select/deselect a particular article to perform some action. It is also possible to sort the articles by any field.
6. The final block appears with a more detailed configuration whenever an article is clicked (see example in Figure 3.6). It allows editing all the meta fields of the article: status (e.g. accepted, rejected), selection criteria (either inclusion or exclusion criteria predefined in the protocol, applied to the article), title, abstract, year (publication year), author, keywords, author keywords, BibTex key, Journal, Document Type, pages, volume, DOI, URL, Affiliation, Publisher, ISSN, language and note. It is also possible to leave comments for the article, open the article's URL by clicking

on the upper-right corner button "External link", go to a previous/next article according to the table ordering and save the edited article.

Study Selection *

All Sources [ACM Digital Library](#) [Science@Direct](#) [Scopus](#) **1.**

[Find Duplicates](#) [Export Articles](#) **2.**

Action: 1 of 5 selected **3.**

Show: All Accepted Rejected Unclassified Duplicated **4.**

<input type="checkbox"/>	Bibtex Key 5.	Title	Author	Journal	Year	Added by	Added at	Status
<input checked="" type="checkbox"/>		Coronavirus disease (COVID-19) prevention and treatment methods and effective parameters: A systematic literature review	Rahmani, Amir Masoud and Mirmahaleh, Seyedeh Yasaman Hosseini	Sustainable cities and society	2021	thedanzjl	18 Nov 2020 09:50:35	Accepted
<input type="checkbox"/>		Guidance on Conducting a Systematic Literature Review	Yu Xiao and Maria Watson	Journal of Planning Education and Research	2019	Daniel Gin	02 Jan 2021 14:07:06	Accepted

6 (click on a study).

Figure 3.5: Parsifal study selection tool. Red lines and red text are added to divide features logically.

Parsifal's implementation of the study selection stage is satisfactory and does not require further improvements in SciLRtool.

3.6 Quality Assessment

From the Cadima analysis, Parsifal does not have a Critical appraisal stage (Cr stage). However, it is not thoroughly true because Parsifal has a practical implementation of this stage, but it is rather weak (Parsifal uses the term Quality Assessment instead of Critical Appraisal). In contrast, Cadima has more flexible and advanced settings, but it does not implement the Quality Assessment itself. In other words, Cadima has the best Quality Assessment in terms of Planning and Parsifal has the best Quality Assessment in terms of Conducting. As already mentioned, SciLRtool aims to implement the best parts of each.

Article Details (2/5) ✕

DetailsComments[External Link](#)

Status

Selection Criteria

Title

Abstract

Literature reviews establish the foundation of academic inquires. However, in the planning field, we lack rigorous systematic reviews. In this article, through a systematic search on the methodology of literature review, we categorize a typology of literature reviews, discuss steps in conducting a systematic literature review, and provide suggestions on how to enhance rigor in literature reviews in planning education and research.

Year

Author

Automatically save the status on change and move next

PreviousNextCloseSave

Figure 3.6: Parsifal article details example

3.6.1 Planning

To define Quality Assessment Checklist, users need to create Quality Assessment Questions and corresponding Quality Assessment Answers; together, they are crucial to a Systematic Literature Review (Systematic Mapping does not require this stage) [23]. Parsifal has a simplified way of defining QA Questions and QA Answers: users can define Questions and Answers (a set of Answers is applied to every Question). However, Cadima defines a separate Answer set for every Question, which is a significant structural difference. Moreover, it greatly expands the system's flexibility (see Figure 3.7).

In addition, Cadima provides an excellent possibility for the main author (review coordinator) to nominate other team members to be involved during

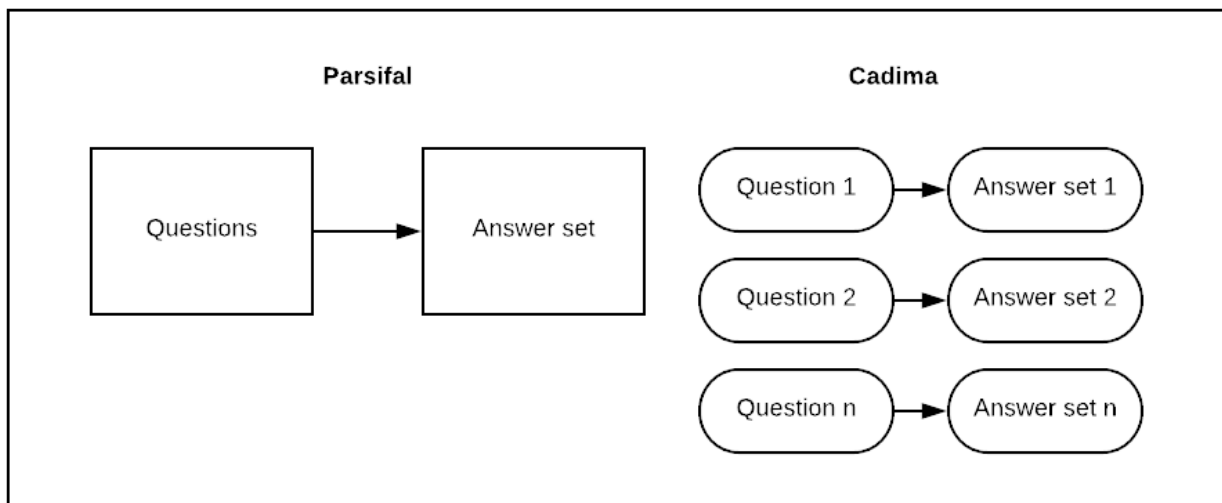


Figure 3.7: Quality Assessment Questions and Answers in Parsifal and Cadima

Quality Assessment, which allows splitting the work between team members.

The key features implemented are outlined below:

- Users can set each Quality Question to have its own set of Answers.
 - Users can copy the existing set of Quality Answers to a new Quality Question.
- The main author can nominate (either manually or automatically) team members to be involved during Quality Assessment.
- Users nominated by the main author for assessment can assess the corresponding included studies.

3.6.2 Conducting

Parsifal allows a group of researchers to assess the included studies concurrently. Nonetheless, different team members may assess the same study, and they have some disagreements about a particular QA Question and its Answer.

Then the conflict system is required. The conflict system provides a great option to deal with conflicts (whenever a QA Question has been answered differently by multiple persons). Besides creating conflicts, the system should have the ability to resolve them (see Figure 3.8). Thus, the new features implemented in SciLRtool are:

- Create conflicts
- Resolve conflicts

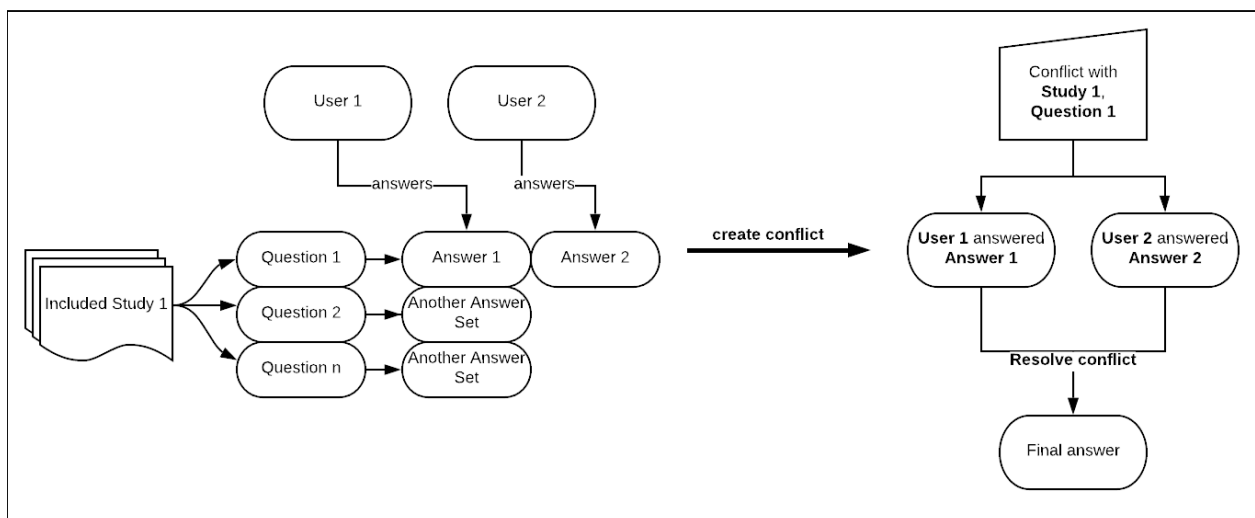


Figure 3.8: Conflict System Example: User 1 and User 2 answer the same question for the same included study differently

3.7 Data Extraction

Data extraction is the determination of primary information in the text of articles. It is one of the most time-consuming steps of systematic literature reviews. Often, the relevant information is placed in graphs, tables or images, and the information should be extracted as accurately as possible. Usually,

two researchers perform the extraction and then resolve conflicts [9]. The automation potential of this task is low. However, it is still possible to partially automate data extraction [9]; for example, ExaCT is the algorithm that highlights the most relevant information automatically, which helps in reducing the text size and thus saving time for performing the extraction task [24].

Nevertheless, to automate data extraction, the text of articles is required, but Elsevier provides full text of articles only by subscription. Parsifal is a non-sponsored project; thus, it uses free APIs and does not access the text. SciLRtool is also a non-sponsored and research-oriented project, and it inherits this problem from Parsifal.

Although Parsifal can not automate data extraction, it does help to extract information by providing a user-friendly interface. Parsifal logically divides data extraction into two parts: planning and conducting.

3.7.1 planning

A reviewer aims to define which fields of studies will be extracted in conducting part and what are their types (integer, float, string, boolean, date, select one field and select many fields).

3.7.2 conducting

Respectively, in conducting part, a user is to extract data by hands and write it into the respective fields of each article. Additionally, Parsifal can mark articles as done or undone; it can also sort articles by done/undone markings. Finally, when the data extraction task is finished, users can download an XLS file with a table of the extracted data.

3.8 Quantitative and Qualitative syntheses of results

The synthesis of results is one of the essential stages of SLR. The synthesis will lead to the SLR objective - analyze the current state of the research area and identify gaps.

3.8.1 Qualitative synthesis

Qualitative or narrative results, such as population, intervention, comparison, outcomes, context (PICOC), sample sizes, and study quality, should be presented in a manner consistent with the review question. Tables should be organized to show the differences and similarities between study outcomes. It is crucial to determine whether outcomes from studies are consistent with one another (i.e. homogeneous) or inconsistent (e.g. heterogeneous) [1].

Parsifal neither automates qualitative synthesis nor provides any interface to support it. CADIMA supports neither quantitative nor qualitative synthesis, but CADIMA provides an interface to upload reviewer's files corresponding to synthesis. This approach is expected since automating Data synthesis is somewhat hard and currently beyond the capabilities of any available ML and NLP tools [25]. Moreover, according to Shelby and Vaske [26], analysis depends on the personal opinions of the reviewer, reviewers proficiency in the research area and study purpose. It becomes transparent that qualitative synthesis is highly dependent on the reviewer's team. Thus SciLRtool, following CADIMA's recipe, intends to allow reviewers to upload their synthesized data in the form of a DOCX file.

3.8.2 Quantitative synthesis

Additionally, according to Kitcheman et al. [1], quantitative information should be presented in the form of tables as well; this includes:

- Intervention sample size.
- Intervention effect size with errors.
- Intervention mean values and confidence interval for the difference between mean values.
- Effect units used for measuring.

Parsifal implements quantitative synthesis of data, such as publication year, source-studies distribution and accepted vs rejected number of studies for every source in a form of interactive figures (see Figures 3.9, 3.10 and 3.11 as examples).

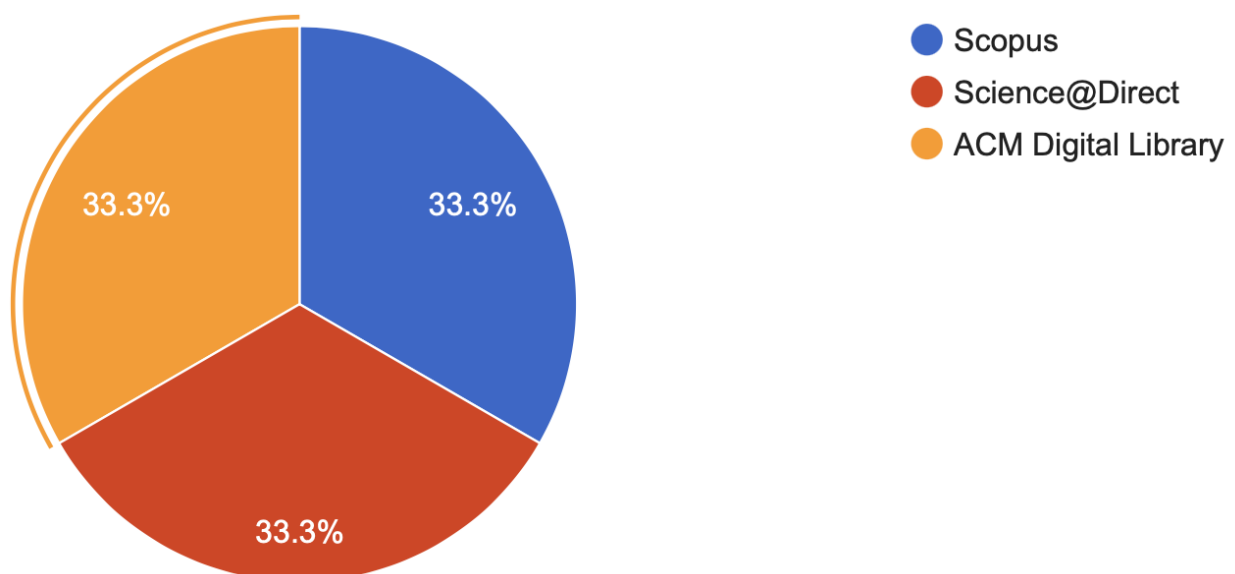


Figure 3.9: Parsifal data analysis; source-studies distribution example. 2 studies per source were chosen.

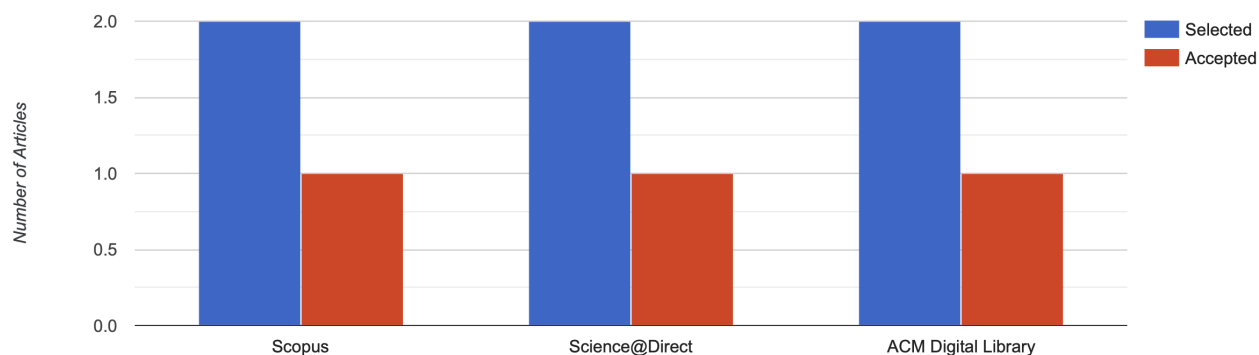


Figure 3.10: Parsifal data analysis; accepted vs rejected number of studies for every source example. 2 studies chosen and 1 accepted per source.

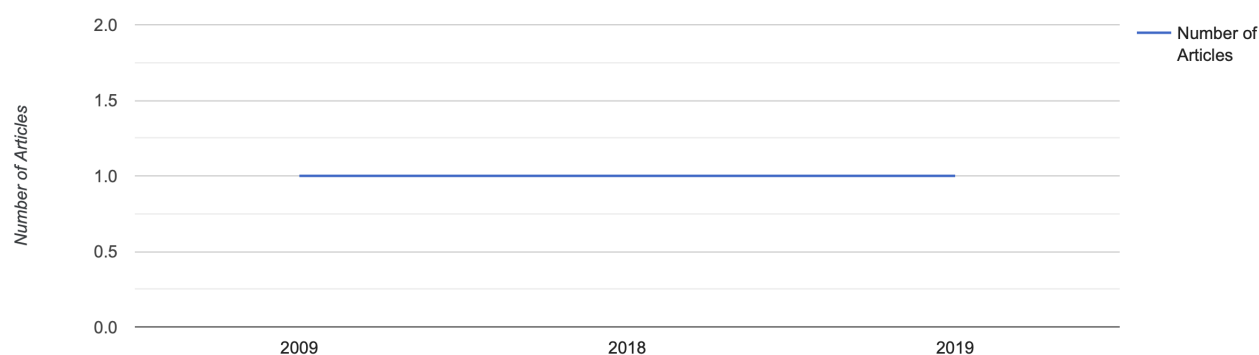


Figure 3.11: Parsifal data analysis; publication year example. Out of 3 accepted studies, 1 has 2009 pub.y., 1 has 2018 pub.y. and 1 has 2019 pub.y.

While these figures are helpful, Parsifal lacks the feature of exporting them in a file format in the case reviewers wish to put figures in their report. Moreover, SciLRtool is indeed keen to fix this, allowing users to export figures in PNG format.

Furthermore, PRISMA (Preferred Reporting Items for Systematic reviews and Meta-Analysis) Statement [27] provides a four-phase flow diagram. It aims at improving the quality of systematic reviews and quantitative synthesis of results (see Figure B.1).

To generate such a diagram, SciLRtool needs to know how many studies are included in quantitative synthesis and qualitative synthesis (the remaining numbers can be obtained from the system database). As automating this task

is currently impossible, reviewers should enter those numbers by themselves and provide data synthesis files. When the system has all the numbers, SciLRtool users can download the flow diagram with all the numbers arranged accordingly.

3.9 Generation of Documentation

Parsifal supports reporting stage - users can download a report that includes selected stages or steps chosen by a reviewer. Instead of dividing report types, such as protocol, reference list, selection criteria, final review and others, Parsifal allows its users to toggle the stages and steps that would be exported in the DOCX file (see Figure 3.12). This approach is a simple and elegant way of doing a report that is transposed to SciLRtool.

The screenshot shows a web interface titled "Download a Report". It is organized into three main sections, each with a "Toggle all" button:

- Review:** Includes checkboxes for "Name", "Authors", and "Description".
- Planning:** Includes checkboxes for "Objectives", "PICOC", "Research Questions", "Keywords and Synonyms", "Search String", "Sources", "Selection Criteria", "Quality Assessment Checklist", and "Data Extraction Form".
- Conducting:** Includes checkboxes for "Source Search Strings" and "Number of Imported Studies".

At the bottom left, there is a blue "Download" button with a downward arrow.

Figure 3.12: Parsifal reporting

However, Moher et al. [28] propose a guideline for protocol and review reporting. The guide enumerates essential aspects and steps the researcher should complete for a report to be comprehensive. After comparing the proposed steps

and the Parsifal features, it was found that some steps were missing. The missing steps are grouped into protocol reporting and final review reporting groups.

3.9.1 Protocol Reporting

The enumerated below steps belong to protocol reporting and are identified as missing in Parsifal and integrated into SciLRtool:

- **Background.** The background explains why the user's study is important and how it can contribute to the field. It describes the role of commissioners and other stakeholders; then, it logically leads to the study's primary question. The background is different from the Project Description stage (which is already present in Parsifal), where a formal declaration of the project is written.
- **Search Strategy.** Draft of search strategy that will be used for at least one electronic database, including planned limits. Search strategy should be transparent, such that it could be repeated.
- **Scoping Exercise.** Estimation of the comprehensiveness of the search.
- **Study Inclusion Criteria** Rationale about the study inclusion/exclusion criteria.
- **Study Screening Mode.** How inclusion/exclusion criteria will be applied.
- **Quality Assessment.** How studies will be assessed.
- **Quality Assessment Mode.** How quality assessment question will be

applied, and how many team members will be involved during the appraisal.

- **Data Extraction Strategy.** How the data from included studies will be collected and recorded.
- **Data Analysis.** How the collected data will be analyzed and synthesized.

3.9.2 Final Review Reporting

The following steps that belong to final review reporting also were implemented in the reporting stage in SciLRtool:

- **Quantitative synthesis result document.** See subsection 3.8.2.
- **Qualitative synthesis result document.** See subsection 3.8.1.
- **Competing interests and sources of support.** Financial and non-financial competing interests.
- **Timeline.**
- **Author's contribution.**
- **Acknowledgements.**
- **Appendices.**

Once all the stages and documentation is complete, reviewers can download the final report and check its completeness. To support assessment of the final review, PRISMA Statement [27] provides with document that consist of 27

item-checklist (see Figure B.2). This report assessment document is available to every SciLRtool user to ensure comprehensive reports. Whenever researchers finish their work and proceed with report writing, SciLRtool advises them to look through this document.

The other important part of the reporting stage is to make synthesis results publicly available (i.e. displaying the evidence synthesis on the website). In such a case, every user can see its title, authors, approach (i.e. SLR or SM) and links to download every part of the report. Public availability intends to increase the transparency of evidence synthesis. This approach is utilized in CADIMA but is absent in Parsifal. Therefore, such a system has been developed in SciLRtool.

Chapter 4

Implementation

This chapter explains in detail the developed system SciLRtool, which is an extension of Parsifal. For convenience, the sections are aligned with the methodology chapter, except the first one, which is an introductory section, and sections that do not require improvements. A table II provides an overview of SciLRtool's improvements over Parsifal.

4.1 Technology Adaption

Initially, the Parsifal original code was rewritten from outdated and unsupported versions - Python v2.7 and Django v1.8.3 to the newer ones - Python v3.8.5 and Django v3.1.3 to meet modern standards. The adaption process required the following steps:

- Correcting syntax
- Finding alternatives for outdated and unsupported dependencies
- Finding alternatives for outdated features

- Adapting new Elsevier API changes

4.2 Setting Up the Review

4.2.1 Define the Question Type (PICO, PIT, PO)

An HTML select tag with a submit-type button is added to support a user's choice of a question type. Whenever a question type is changed, users will see the appropriate input fields, e.g. when a user changes PICOC to PIT, the one will only see P, I and T input fields.

4.2.2 Define if SR or SM Will Be Performed

Another HTML select field is added in the evidence synthesis definition stage. Users can choose from "Systematic Literature Review" and "Systematic Mappings". Since some stages can be skipped when a user decides to perform SM, red asterisks * were added beside some stage titles to indicate that the user can not skip them.

4.3 Literature Searching

Elsevier API endpoint was adapted to the new API specification. Moreover, after retrieving search results from Elsevier API, links to documents and their citations were added.

4.4 Duplicate Checking

SoftTF.IDF - a token-based similarity metric was utilized from the `py_stringmatching` library and integrated into a duplicate detection algorithm. This metric requires a threshold parameter t . An experiment was conducted to identify the optimal threshold parameter - ten sentences with the number of words from 10 to 15 were taken, then one synthetic misspelling was added for each sentence. It was discovered that to detect a typo or misspellings in a lengthy article title (from 10 to 15 words) and mark it as a duplicate, t should be equal to 0.85.

4.5 Quality Assessment

A new approach to quality assessment is described in the methodology section 3.6, and a new interface is designed to support new features of this approach.

4.5.1 Planning

First of all, database tables `QualityQuestion` and `QualityAnswer` were changed. If previously they had no relation entirely, now `QualityAnswer` has a "question" field, which is a `ForeignKey` relation to `QualityQuestion` table so that every question has its own set of answers (see Figure 3.7). Figure 4.1 illustrates a new interface of the quality assessment checklist.








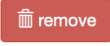













Quality Questions *			
Question	Answers	Weights	
  Are the research aims clearly specified?	yes partially no	1.0 0.5 0.0	 
  Was the study designed to achieve these aims?	yes no	1.0 0.0	 
  Are the prediction techniques used clearly described and their selection justified?	excellently described & justified described & justified described justified none	2.0 1.0 0.5 0.0	 
  Are the variables considered by the study suitably measured?	yes partially no	1.0 0.5 0.0	 
  Are the data collection methods adequately detailed?	yes no	2.0 0.0	 
			

Figure 4.1: Quality Assessment checklist new interface

For the researcher’s convenience, now questions can alter their relative position in a questions list.

Furthermore, when a user clicks the "edit" or "Add Question" buttons are clicked, a new modal window appears (see Figure 4.2).

New Quality Question ✕

Description

Are the research aims clearly specified?

Answers	Weight	Remove
yes	1	✖
partially	0.5	✖
no	0	✖

[+ Add answer](#)

You can append existing answer set from

[Cancel](#) [✓ Save](#)

Figure 4.2: Quality Assessment checklist example of a new modal window to add a new quality question with answers.

In this window, users can add a name, unlimited number of answers and corresponding weights to a new question. It is also possible to add an existing answer set to a new question, which is appropriate when a new question has a similar or the same answer set as a previously defined question.

Secondly, a new settings bar was added for main authors (i.e. creators of evidence synthesis) to support team members nomination for the article’s assessment. The main author can allocate included studies to different team members, including themselves (see Figure 4.3).

Settings

i As a main author of the review, you are able to allocate included articles to the different review team members. Please, choose the percentage of articles to assess for each team member.

Team member	Percentage
danilginzburg	30
thedanzjl	70

✓ Save

Figure 4.3: Quality Assessment checklist example of settings for main author.

In the planning stage, the number of included articles is still unknown. Thus, it was decided to use a percentage of the total number of articles as the nomination amount. SciLRtool warns a user in case a sum of percentages is less than 100%. However, the sum might be greater than 100%, as the system assumes that a peer-review will be conducted.

4.5.2 Conducting

To meet the new quality assessment checklist features, we redesigned the conducting part of the quality assessment (see Figure 4.4).

Quality Assessment *

i You have allocated 30% of the articles for assessment for yourself (about 1.2 articles).
Click on the button below to perform automated articles assignment to your team members according to quality assessment settings.

 Articles assignment

Detailed Summary Conflicts

Show: All Done Pending Score higher than 3.0 Score lower or equal to 3.0

Assigned to: Order by:

To answer the form you may click on the desired answer on the following tables.

Guidance on Conducting a Systematic Literature Review (2019) 3.5				
Assigned to Daniel G, thedanzli; Reassign				
Are the research aims clearly specified?	yes	partially	no	
Was the study designed to achieve these aims?	yes	no		
Are the prediction techniques used clearly described and their selection justified?	excellently described & justified	described & justified	described justified	none
Are the variables considered by the study suitably measured?	yes	partially	no	
Are the data collection methods adequately detailed?	yes	no		

Figure 4.4: Quality Assessment: conducting.

Beneath the quality assessment title, information text shows how many articles the main author has allocated for a particular team member. Exclusively for the main author, automated and manual assignment of articles are available.

- **Automated assignment.** Only the main author can see an "Articles assignment" button under the information text. It allows team members to perform cycled assignment of articles. For example, if 70% of articles allocated to user1 and 30% to user2, then the first 70% of randomly sorted articles are assigned to user1, and the remaining 30% + first 30% to user2, i.e. 60% of articles will be peer-reviewed.
- **Manual assignment.** Every individual article can be reassigned to another user or even be left without assignment. Beneath every article, there is a list of assigned users for this article and a link for manual re-assignment. The list will be coloured in green if the currently logged-in

user appears in this list; otherwise, it will be red. When a user clicks the "reassignment" link is clicked, a modal window appears where main authors can conduct reassignment (see Figure 4.5).

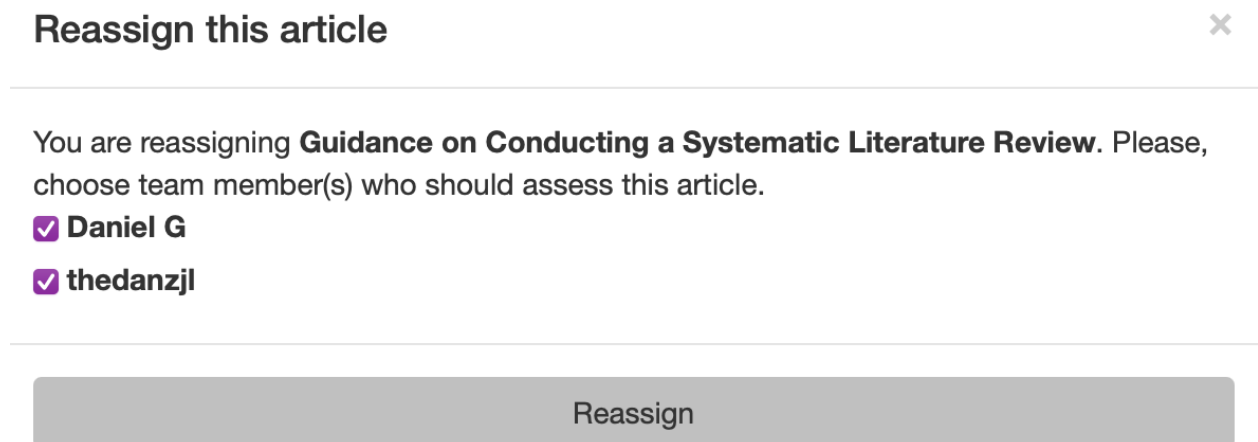


Figure 4.5: Quality Assessment: manual reassignment of article.

When the main author finishes their assignment, team members can filter articles by their assignment; the available filters of the HTML select tag "assigned to" are: me, all, unassigned and the list of all other users except the current one.

Another noteworthy feature developed is the Conflict System described in 3.8. When two or more users create conflicts, they will be displayed in the "Conflicts" tab (see Figure 4.6) will display them.

Quality Assessment *

📌 Main author has allocated 70% of the articles for assessment for you (about 2.8 articles).

Detailed

Summary

Conflicts

Show conflicts: All Mine

Here you can resolve conflicts between team members appeared during quality assessment process.

Guidance on Conducting a Systematic Literature Review <small>(2019)</small>			2.0
Are the research aims clearly specified?	Daniel G: yes	thedanzjl: partially	no Resolve
Was the study designed to achieve these aims?	Daniel G: yes	thedanzjl: no	Resolve

Figure 4.6: Quality Assessment: conflicts example.

Only articles with questions that produced conflicts are displayed. Besides every answer to a conflicted question, the list of authors who chose it is displayed. Every team member of a team can resolve conflicts by selecting appropriate answers and clicking a corresponding "Resolve" button. Another way to resolve a conflict is if some team members change the conflicting answer so that all the answers to one particular question agree.

In case of many conflicts, filtering by "all" and "mine" conflicts were implemented.

4.6 Quantitative and Qualitative Synthesis

In the reporting stage, users can upload their quantitative and qualitative synthesis files in DOCX, TXT, JPG, JPEG, PNG, BMP and GIF formats up to 1Gb size. The system will include those files in a final report. Along with files, users can specify the number of articles used in quantitative and qualitative synthesis so that an integrated flow diagram by PRISMA will be generated (see Figure 4.7). The generation of flow diagram consists of several parts:

- Extract all relevant numbers needed for the diagram from the database.

- Upon existing flow diagram template in a format of a PNG image, put numbers beside corresponding arrows of the diagram using Pillow library.

In the data analysis tab in the conducting stage, beside every generated figure (see Figures 3.9, 3.10 and 3.11) new button "Export as PNG" added, so that users can download those figures for later usage.

Quantitative Synthesis

Number of studies synthesised: 1

Supported file types: .jpg .jpeg .png .gif .bmp .docx .txt (max file size: 1GB)

quantitative_synthesis_final.docx (click to change the file)

Save

Qualitative Synthesis

Number of studies synthesised: 1

Supported file types: .jpg .jpeg .png .gif .bmp .docx .txt (max file size: 1GB)

qualitative_synthesis_final.docx (click to change the file)

Save

Figure 4.7: Interface for quantitative and qualitative synthesis

4.7 Generation of Documentation

4.7.1 Documentation Interface

Many textual fields were added in the planning stage - protocol reporting and reporting stage - final review reporting. In the planning stage, a new tab called "Documentation" was created with the following text fields: Search Strategy, Scoping Exercise, Study Inclusion Criteria, Study Screening Mode, Quality Assessment, Quality Assessment Mode, Data Extraction Strategy and Data Analysis. Every field has a round button with a question inside it; whenever a user clicks it, supporting information is provided to users according to one particular field.

In the reporting stage, a new tab called "Final Review Documentation" was produced. Along with quantitative and qualitative synthesis fields (see section 4.6), we added Competing Interests and Sources of Support, Timeline, Author's Contribution, Acknowledgements and Appendices fields were added. Appendices is a file field that accepts DOCX, TXT, JPG, JPEG, PNG, BMP, and GIF files up to 1Gb size. Only the Competing Interests and Sources of Support field has the help button with supporting information since other fields are self-explanatory.

In the reporting stage in the "Export" tab, new toggle tags were added following the new features (see Figure 4.8).

Download a Report and Publish

Review Toggle all

Name Authors Description

Planning Toggle all

Approach Background Objectives PICOC

Research Questions Keywords and Synonyms Search String Sources

Selection Criteria Quality Assessment Checklist Data Extraction Form

Protocol Documentation Toggle all

Search Strategy Scoping Exercise Study Inclusion Criteria Study Screening Mode

Quality Assessment Quality Assessment Mode Data Extraction Strategy Data Analysis

Conducting Toggle all

Source Search Strings Number of Imported Studies Flow diagram

Final Review Documentation Toggle all

Quantitative Synthesis Qualitative Synthesis Competing Interests Timeline

Author's Contribution Acknowledgments Appendices

Download ▾
Publish ▾

Figure 4.8: New reporting stage, export tab

Compared with Parsifal's reporting (see Figure 3.12), it now has much more toggle fields. A bug when some stage has all fields toggled off, but the title of the stage appears in the report anyway, has been fixed.

4.7.2 Publishing Evidence Synthesis

The dropdown "Download" button from the bottom of the reporting stage previously had the only option ".docx". The option's name was changed to "report," and the new option "Self-assessment checklist" was appended. When

a user clicks the "Self-assessment checklist" option, the system will download PRISMA's reporting assessment checklist in DOC format.

The "Publish" button now allows users to make their evidence synthesis publicly available. It appears in the new "Browse" navbar menu of the SciLRtool web site whenever it is published. This menu is dedicated to published systematic reviews and systematic mappings (see Figure 4.9).

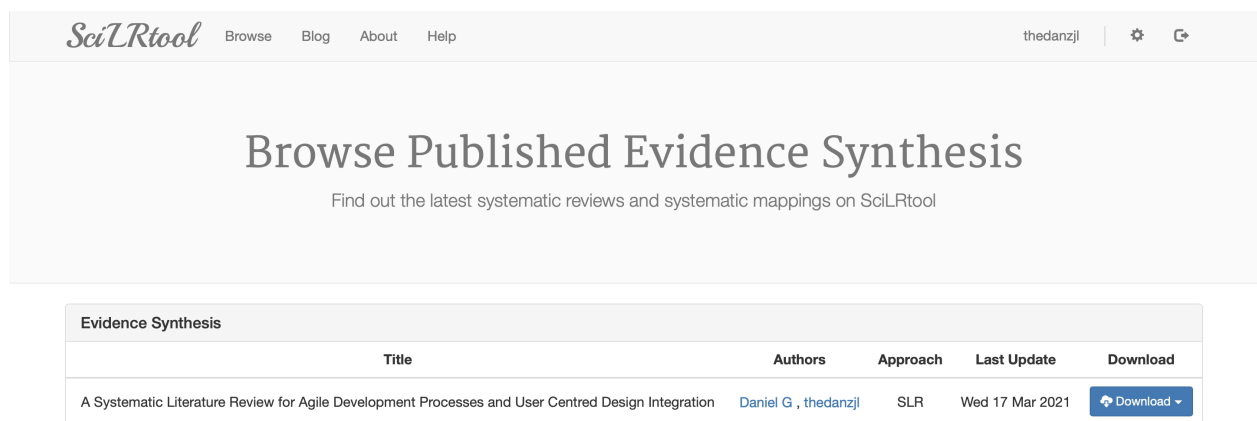


Figure 4.9: New Browse navbar-menu

The table of published evidence synthesis has the title, authors, approach, last update and download columns. The system displays every author. Every author is displayed along with a link to the author's profile page. Clicking the "Download" button will display a list of options; every option specifies what part of the evidence synthesis will be downloaded. The options are the following:

- **Review Description.** This option includes the title, authors and description.

- **Planning.** The planning option includes all the fields specified in the planning stage, excluding protocol documentation.
- **Protocol.** The protocol option includes review description, planning and protocol documentation.
- **Conducting.** The conducting option includes Source Search Strings, Number of Imported Studies and Flow Diagram.
- **Data Analysis.** The data analysis option includes only Quantitative and Qualitative synthesis files.
- **Data Extraction Sheet.** The data extraction sheet option includes the outcome of the data extraction step.
- **Study Selection Outcome.** The study selection outcome option includes the outcome of the study selection step with inclusion/exclusion criteria, source, publisher and others.
- **Selection Criteria.** This option introduces only inclusion/exclusion criteria.
- **Full Report.** The full report option includes everything except the data extraction sheet and study selection outcome.

The table of published evidence synthesis is also visible in a profile page of authors but is limited to only those evidence synthesis where the author has contributed. The table appears beneath the table "Work in progress", which is not visible to external users. If some author opens their page, (s)he will see both tables and the links to evidence synthesis in both tables.

Chapter 5

Evaluation and Discussion

5.1 Evaluation

To evaluate the developed system, we conducted several individual interviews with people who conducted Systematic Literature Reviews. Beforehand, we created and specified a list of questions and validation criteria. We applied the Likert scale [29] as our validation criteria. Figure 5.1 shows the questions and their weights of the Likert scale.

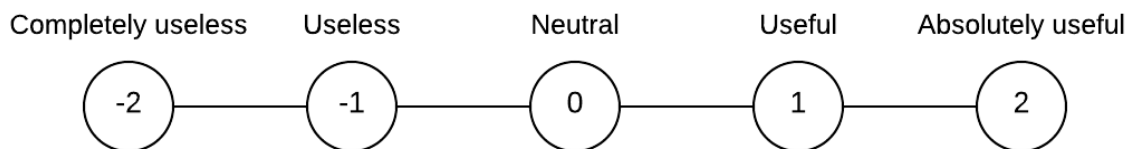


Figure 5.1: The Likert scale applied in interviews

The list of questions is the following:

1. What experience is SLR you have?
2. In what domains have you conducted SLR's?

3. Have you ever used any tools supporting SLR's?
4. In your opinion, what features such tools should possess?

The purpose of those questions is to understand an interviewee's experience in SLR's. The last question serves the purpose of gaining new ideas and inspirations for future work. If we notice that the interviewee does not possess enough knowledge or experience in the SLR domain (e.g. interviewee have only read a couple of SLR's but did not conduct it), we terminate our interview. After asking questions, we demonstrate SciLRtool to an interviewee. We explain every stage in details and ask them to evaluate it on the Likert scale.

The interviews took place in Russia, Innopolis city, Innopolis University. We interviewed eleven people, where 3 are professors, 6 are master students, and 3 are bachelor students. The message inviting people to conduct interviews was sent via an email to our University colleagues. It explicitly describes the interview procedure. It also asks only people with SLR experience to respond to this email.

To obtain the general feedback of a concrete stage of SciLRtool, we take the average of interviewees' results. The final feedback of a stage then appears on the range from -2 to +2. The following section discusses the results of every stage and new ideas we received from interviewees.

5.1.1 Setting Up the Review and Protocol Definition

- **Result AVG 0; Neutral**

Setting up the Review and Protocol Definition are the first stages of any SLR, and its implementation in SciLRtool gives the users the very first impression of

our tool. As it appeared, many people find the interface of our tool non-friendly - "this design looks outdated" - said one of our interviewees. Furthermore, some people claimed they still prefer to undertake the Protocol Definition step via google sheets or overleaf because they are accustomed to it.

5.1.2 Literature Searching

- **Result AVG 0; Neutral**

Most of our interviewees expect more functionality of our built-in Literature Searching and mostly do not see a reason to use it. We found that some people do not prefer Elsevier, our search engine, and instead use Google Scholar. A professor complained about the small number of literature sources (only Science@Direct and Scopus). Moreover, one master student said she wants to see a journal rating in the search results table during interviews.

The feedback we received is fairly reasonable - the researchers who conduct SLR's require multiple search engines and search sources in one place with comprehensive metadata of every research work. We consider those features our primary course of future work as the Literature Searching stage is the defining reason few researchers might choose SciLRtool over other tools.

5.1.3 Duplicate Checking

- **Result AVG +1; Useful**

Duplicate Checking is a minor yet reasonably helpful feature. People agree on its usefulness and consider it a required feature of any tool supporting SLR's; however, they are not impressed.

5.1.4 Study Selection

- **Result AVG +2; Absolutely useful**

Although SciLRtool did not contribute to Parsifal's implementation of the Study Selection stage, all interviewees agree on its usefulness and are satisfied with the result the stage produces - an XLS table with all studies, its inclusion/exclusion criteria, author's comments and other metadata. Nevertheless, some experts still made a couple of remarks and suggestions. The first is to add filtering of articles by year, journal and journal rating since SciLRtool possesses only sorting feature. Furthermore, the other suggestion is to display the author's comments on a particular article near its status (Accepted, Rejected or Duplicated) so that users can explicitly see an article's status and the reasoning behind it.

We strongly agree with the first suggestion and include it in our plan for future work since some users might have a dozen hundreds of different articles and might want to search or filter them using various methods. However, we consider the second suggestion (adding comments near the status of an article) to be a personal preference instead of a needed functionality since only one person suggested it, and we think this will overcomplicate the interface of SciLRtool.

5.1.5 Quality Assessment

- **Result AVG +2; Absolutely useful**

The new interface and features we developed for Quality Assessment satisfy the needs of all our interviewees. Especially they find helpful the new Conflicts

system. We are satisfied with the obtained results and do not include the Quality Assessment stage in the list for future work.

5.1.6 Data Extraction

- **Result AVG -1; Useless**

Since the process of data extraction is done by hands and is not automated. However, extracting obtained articles in XLS format is unsuitable since most users do their SLR's in L^AT_EX.

5.1.7 Data Analysis

- **Result AVG +1; Useful**

Most people spend much time creating qualitative analysis diagrams by hands in third-party programs. SciLRtool auto-generates such diagrams and allows users to download them. Most people consider the diagrams useful; however, some interviewees suggested generating diagrams in TEX format according to the L^AT_EX TikZ package.

5.1.8 Generation of Documentation

- **Result AVG +2; Absolutely useful**

People are primarily positive about the auto-generation of a final report. Many people especially noted the new PRISMA diagram flow. However, the DOCX format of the final report is unsuitable for most users. Around half of the interviewees suggest generating the final report in L^AT_EX, concretely, Overleaf

[30]. Overleaf is a popular Tex editor, and most of our colleagues at Innopolis University utilise it.

We set our primary goal with the highest priority for future work to integrate our system with Overleaf. The open problem remains to define an Overleaf template to suit most users.

5.1.9 Publishing Evidence Synthesis

- **Result AVG +2; Absolutely useful**

Publishing the work done to be publically available is beneficial both for authors and users. Users can find relevant SLR's, while authors can receive feedback from other users about their evidence synthesis. We obtained the only suggestion to add searching and filtering by evidence synthesis title in the "Browse" section of SciLRtool. We think this suggestion is correct since, ordinarily, people want to get SLR's in the concrete domain area and not the entire list of existing evidence synthesis.

5.2 Discussion

We attempted to create a new product that includes best practices of other tools and supports all stages of the Systematic Literature Review. We definitely succeeded as the average result of evaluation for our system is +1 (useful); however, SciLRtool supports some stages but does not automate them. We can observe it by the example of the Data Extraction stage - people consider it useless because they need to conduct data extraction by hands. The same applies to Setting up the review and Protocol Definition. Those stages can

primarily help people who are new to SLR's since they force researchers to accompany every stage so that literature review becomes systematic.

Additionally, all stages in SciLRtool provide helpful information so that users understand the purpose. However, we investigated that many proficient researchers prefer writing, extracting, and analysing data directly in TEX editors, such as Overleaf. We consider that integrating SciLRtool with Overleaf will engage more potential users and make our tool more competitive.

Besides, none of our interviewees has ever applied any tools supporting SLR's so we could not estimate the competitiveness of SciLRtool and what elements of it would make people favour it over other tools. Therefore, we aim at conducting such interviews after the proposed future work will be accomplished.

Chapter 6

Conclusion

Initially, we researched the domain of Systematic Literature Reviews in Software Engineering and the tools supporting it as described in the Literature Review chapter 2. Then we discovered a niche in this domain - there are no existing tools dedicated to Software Engineering that support all stages of Systematic Literature Reviews; thus, we decided to contribute to it by creating SciLRtool. Accordingly, we designed SciLRtool in a way that combines best practices of Parsifal and CADIMA tools so that it supports every stage described in the Methodology chapter 3. Henceforth, we implemented SciLRtool and explained our design decisions in the Implementation chapter 4.

Finally, we evaluated the developed system by interviewing our colleagues at Innopolis University, demonstrating SciLRtool to them and asking questions as described in the Evaluation and Discussion chapter 5. By the end of the interviews, we learned that most people experienced in SLR's acknowledge our tool helpful in their practices, especially Study Selection, Quality Assessment and Publishing Evidence Synthesis stages. Moreover, we collected feedback from our interviewees and defined our future work requirements, essentially

integration with Overleaf. Furthermore, we will continue to work on other stages that were evaluated less than "Absolutely useful".

Bibliography cited

- [1] K. BA and S. Charters, “Guidelines for performing systematic literature reviews in software engineering,” vol. 2, Jan. 2007.
- [2] D. Salah, R. Paige, and P. Cairns, “A systematic literature review for agile development processes and user centred design integration,” *ACM International Conference Proceeding Series*, May 2014. DOI: 10.1145/2601248.2601276.
- [3] J. Thomas and J. Brunton, “Eppi-reviewer 4: Software for research synthesis,” Jan. 2010.
- [4] C. Kohl, E. McIntosh, S. Unger, N. Haddaway, S. Kecke, J. Schiemann, and R. Wilhelm, “Online tools supporting the conduct and reporting of systematic reviews and systematic maps: A case study on cadima and review of existing tools,” *Environmental Evidence*, vol. 7, Feb. 2018. DOI: 10.1186/s13750-018-0115-5.
- [5] V. Freitas, *Parsifal*, <https://parsif.al>, [Online; accessed 28-January-2021], 2018.
- [6] E. Akl, D. Altman, P. Aluko, L. Askie, D. Beaton, J. Berlin, B. Bhau-mik, C. Bingham, M. Boers, A. Booth, I. Boutron, S. Brennan, M. Briel, S. Briscoe, J. Busse, D. Caldwell, M. Cargo, A. Carrasco-Labra,

- A. Chaimani, and C. Young, *Cochrane Handbook for Systematic Reviews of Interventions*. Oct. 2019, ISBN: 9781119536604.
- [7] D. Budgen, S. Charters, M. Turner, P. Brereton, B. Kitchenham, and S. Linkman, “Investigating the applicability of the evidence-based paradigm to software engineering,” *Proceedings - International Conference on Software Engineering*, Apr. 2006. DOI: 10.1145/1137661.1137665.
- [8] D. Budgen, M. Turner, P. Brereton, and B. Kitchenham, “Using mapping studies in software engineering,” *Proceedings of PPIG 2008*, vol. 2, Jan. 2008.
- [9] G. Tsafnat, P. Glasziou, M. K. Choong, A. Dunn, F. Galgani, and E. Coiera, “Systematic review automation technologies,” *Systematic reviews*, vol. 3, p. 74, Jul. 2014. DOI: 10.1186/2046-4053-3-74.
- [10] G. Tsafnat, A. Dunn, P. Glasziou, and E. Coiera, “The automation of systematic reviews,” *BMJ (Clinical research ed.)*, vol. 346, f139, Jan. 2013. DOI: 10.1136/bmj.f139.
- [11] M. Ouzzani, H. Hammady, Z. Fedorowicz, and A. Elmagarmid, “Rayyan—a web and mobile app for systematic reviews,” *Systematic Reviews*, vol. 5, Dec. 2016. DOI: 10.1186/s13643-016-0384-4.
- [12] K. James, N. Randall, and N. Haddaway, “A methodology for systematic mapping in environmental sciences,” *Environmental Evidence*, vol. 5, p. 7, Apr. 2016. DOI: 10.1186/s13750-016-0059-6.
- [13] C. Counsell, “Formulating questions and locating primary studies for inclusion in systematic reviews,” *Annals of Internal Medicine*, vol. 127, pp. 380–387, 1997.

- [14] D. Sackett, W. Richardson, W. Rosenberg, and b. Haynes, “Evidence-based medicine. how to practice and teach ebm. evidence-based medicine,” *Churchill Livingstone*, vol. 2, Jan. 2005.
- [15] P. Doshi, M. Jones, and T. Jefferson, “Rethinking credible evidence synthesis,” *BMJ*, vol. 344, 2012, ISSN: 0959-8138. DOI: 10.1136/bmj.d7898. eprint: <https://www.bmj.com/content/344/bmj.d7898.full.pdf>. [Online]. Available: <https://www.bmj.com/content/344/bmj.d7898>.
- [16] *Elsevier*, <https://www.elsevier.com>, [Accessed: 2021-01-08].
- [17] A. Lunev, *Alternatives to elsevier?* May 2020.
- [18] B. Ballew, “Elsevier’s scopus® database,” *Journal of Electronic Resources in Medical Libraries*, vol. 6, pp. 245–252, Jul. 2009. DOI: 10.1080/15424060903167252.
- [19] *Science@direct*, <http://www.sciencedirect.com/>, [Accessed: 2021-01-09].
- [20] A. Elmagarmid, P. Ipeirotis, and V. Verykios, “Duplicate record detection: A survey,” *Knowledge and Data Engineering, IEEE Transactions on*, vol. 19, pp. 1–16, Feb. 2007. DOI: 10.1109/TKDE.2007.250581.
- [21] R. Aabenhus, J. U. Jensen, and J. Cals, “Incorrect inclusion of individual studies and methodological flaws in systematic review and meta-analysis,” *The British journal of general practice : the journal of the Royal College of General Practitioners*, vol. 64, pp. 221–2, May 2014. DOI: 10.3399/bjgp14X679615.
- [22] M. Bilenko, R. Mooney, W. Cohen, P. Ravikumar, and S. Fienberg, “Adaptive name matching in information integration.,” *Intelligent Systems, IEEE*, vol. 18, pp. 16–23, Oct. 2003. DOI: 10.1109/MIS.2003.1234765.

- [23] Y. Zhou, H. Zhang, X. Huang, S. Yang, M. Ali Babar, and H. Tang, "Quality assessment of systematic reviews in software engineering," Apr. 2015, pp. 1–14. DOI: 10.1145/2745802.2745815.
- [24] S. Kiritchenko, B. de Bruijn, S. Carini, J. Martin, and I. Sim, "Exact: Automatic extraction of clinical trial characteristics from journal publications," *BMC medical informatics and decision making*, vol. 10, p. 56, Sep. 2010. DOI: 10.1186/1472-6947-10-56.
- [25] I. J. Marshall and B. C. Wallace, "Toward systematic review automation: A practical guide to using machine learning tools in research synthesis," *Systematic Reviews*, vol. 8, no. 1, p. 163, Jul. 2019, ISSN: 2046-4053. DOI: 10.1186/s13643-019-1074-9. [Online]. Available: <https://doi.org/10.1186/s13643-019-1074-9>.
- [26] L. B. Shelby and J. J. Vaske, "Understanding meta-analysis: A review of the methodological literature," *Leisure Sciences*, vol. 30, no. 2, pp. 96–110, 2008. DOI: 10.1080/01490400701881366. eprint: <https://doi.org/10.1080/01490400701881366>. [Online]. Available: <https://doi.org/10.1080/01490400701881366>.
- [27] D. Moher, A. Liberati, J. Tetzlaff, D. G. Altman, and T. P. Group, "Preferred reporting items for systematic reviews and meta-analyses: The prisma statement," *PLOS Medicine*, vol. 6, no. 7, pp. 1–6, Jul. 2009. DOI: 10.1371/journal.pmed.1000097. [Online]. Available: <https://doi.org/10.1371/journal.pmed.1000097>.
- [28] D. Moher, A. Liberati, J. Tetzlaff, and D. G. A. and, "Preferred reporting items for systematic reviews and meta-analyses: The PRISMA statement," *PLoS Medicine*, vol. 6, no. 7, e1000097, Jul. 2009. DOI: 10.1371/

- journal.pmed.1000097. [Online]. Available: <https://doi.org/10.1371/journal.pmed.1000097>.
- [29] R. Likert, "A technique for the measurement of attitudes," English, OCLC: 812060, Ph.D. dissertation, The Science Press, New York, 1932.
- [30] J. Hammersley and J. Lees-Miller, 2012. [Online]. Available: <https://www.overleaf.com/>.

Appendix A

Existing Steps For
Systematic Reviews

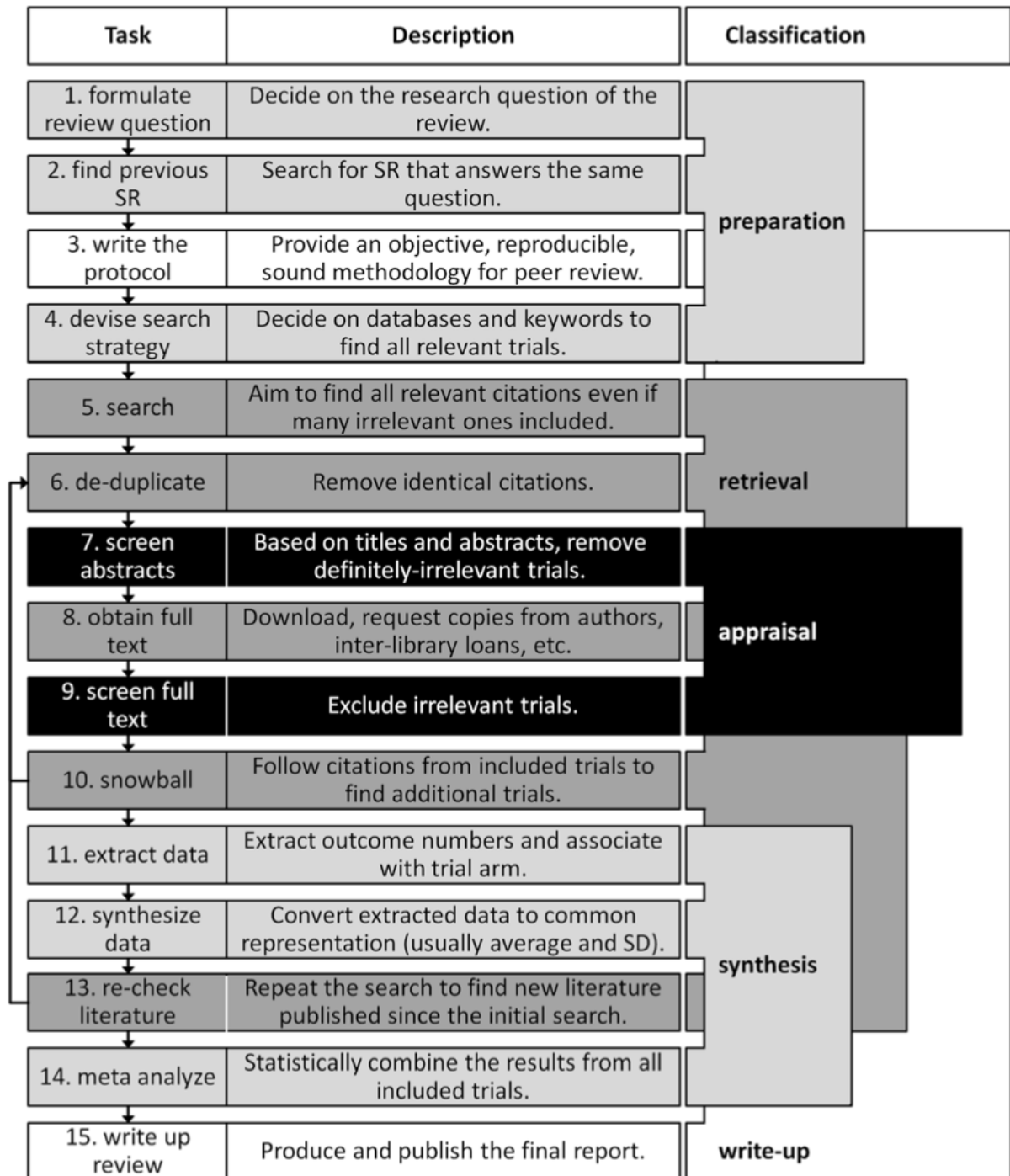


Figure A.1: Existing steps for systematic reviews (possible to have some deviations) [9]

Appendix B

PRISMA documents

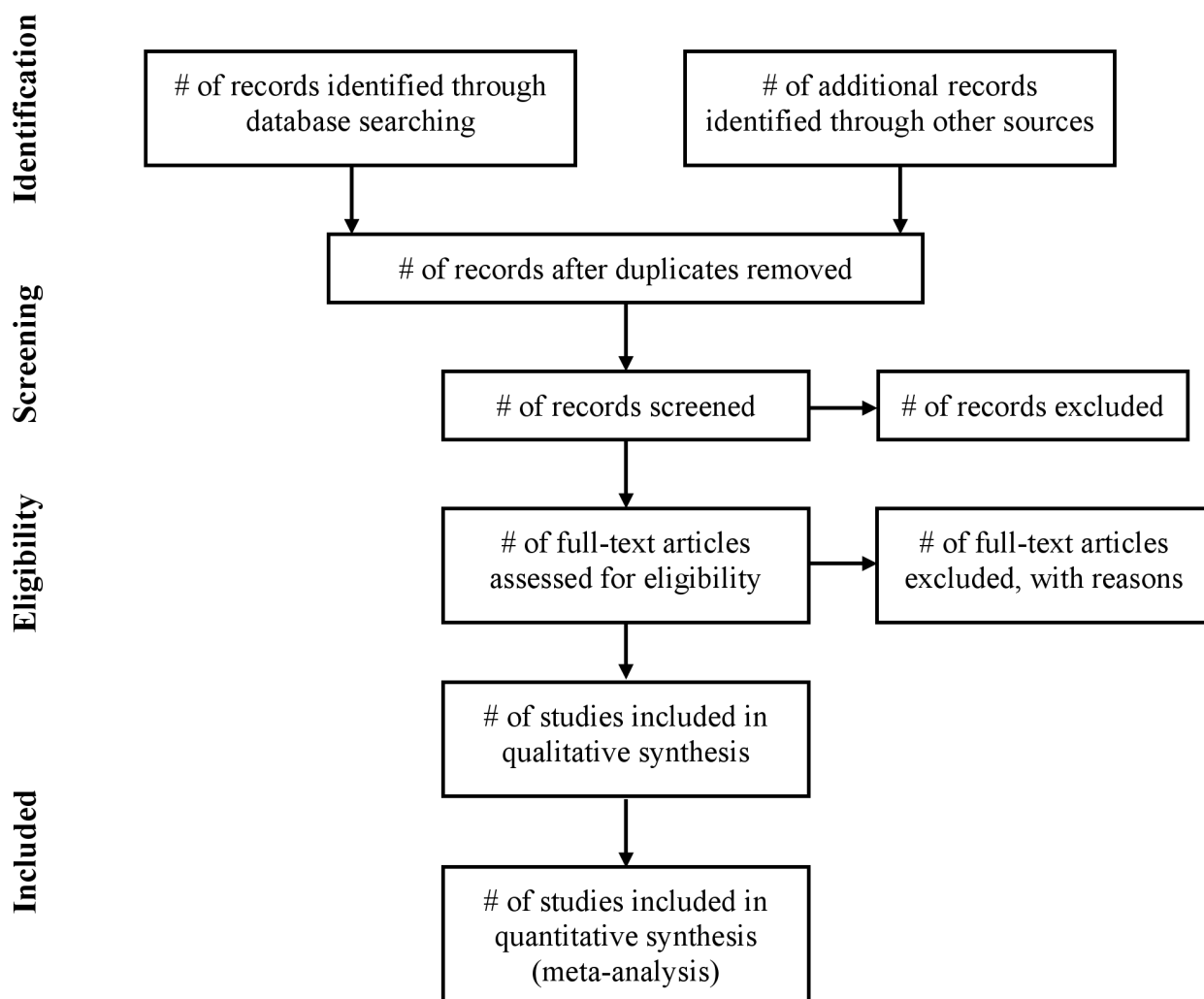


Figure B.1: PRISMA flow diagram template

Section/Topic	#	Checklist Item	Reported on Page #
TITLE			
Title	1	Identify the report as a systematic review, meta-analysis, or both.	
ABSTRACT			
Structured summary	2	Provide a structured summary including, as applicable: background; objectives; data sources; study eligibility criteria, participants, and interventions; study appraisal and synthesis methods; results; limitations; conclusions and implications of key findings; systematic review registration number.	
INTRODUCTION			
Rationale	3	Describe the rationale for the review in the context of what is already known.	
Objectives	4	Provide an explicit statement of questions being addressed with reference to participants, interventions, comparisons, outcomes, and study design (PICOS).	
METHODS			
Protocol and registration	5	Indicate if a review protocol exists, if and where it can be accessed (e.g., Web address), and, if available, provide registration information including registration number.	
Eligibility criteria	6	Specify study characteristics (e.g., PICOS, length of follow-up) and report characteristics (e.g., years considered, language, publication status) used as criteria for eligibility, giving rationale.	
Information sources	7	Describe all information sources (e.g., databases with dates of coverage, contact with study authors to identify additional studies) in the search and date last searched.	
Search	8	Present full electronic search strategy for at least one database, including any limits used, such that it could be repeated.	
Study selection	9	State the process for selecting studies (i.e., screening, eligibility, included in systematic review, and, if applicable, included in the meta-analysis).	
Data collection process	10	Describe method of data extraction from reports (e.g., piloted forms, independently, in duplicate) and any processes for obtaining and confirming data from investigators.	
Data items	11	List and define all variables for which data were sought (e.g., PICOS, funding sources) and any assumptions and simplifications made.	
Risk of bias in individual studies	12	Describe methods used for assessing risk of bias of individual studies (including specification of whether this was done at the study or outcome level), and how this information is to be used in any data synthesis.	
Summary measures	13	State the principal summary measures (e.g., risk ratio, difference in means).	
Synthesis of results	14	Describe the methods of handling data and combining results of studies, if done, including measures of consistency (e.g., I^2) for each meta-analysis.	
Risk of bias across studies	15	Specify any assessment of risk of bias that may affect the cumulative evidence (e.g., publication bias, selective reporting within studies).	
Additional analyses	16	Describe methods of additional analyses (e.g., sensitivity or subgroup analyses, meta-regression), if done, indicating which were pre-specified.	
RESULTS			
Study selection	17	Give numbers of studies screened, assessed for eligibility, and included in the review, with reasons for exclusions at each stage, ideally with a flow diagram.	
Study characteristics	18	For each study, present characteristics for which data were extracted (e.g., study size, PICOS, follow-up period) and provide the citations.	
Risk of bias within studies	19	Present data on risk of bias of each study and, if available, any outcome-level assessment (see Item 12).	
Results of individual studies	20	For all outcomes considered (benefits or harms), present, for each study: (a) simple summary data for each intervention group and (b) effect estimates and confidence intervals, ideally with a forest plot.	
Synthesis of results	21	Present results of each meta-analysis done, including confidence intervals and measures of consistency.	
Risk of bias across studies	22	Present results of any assessment of risk of bias across studies (see Item 15).	
Additional analysis	23	Give results of additional analyses, if done (e.g., sensitivity or subgroup analyses, meta-regression [see Item 16]).	
DISCUSSION			
Summary of evidence	24	Summarize the main findings including the strength of evidence for each main outcome; consider their relevance to key groups (e.g., health care providers, users, and policy makers).	
Limitations	25	Discuss limitations at study and outcome level (e.g., risk of bias), and at review level (e.g., incomplete retrieval of identified research, reporting bias).	
Conclusions	26	Provide a general interpretation of the results in the context of other evidence, and implications for future research.	
FUNDING			
Funding	27	Describe sources of funding for the systematic review and other support (e.g., supply of data); role of funders for the systematic review.	

doi:10.1371/journal.pmed.1000097.t001

Figure B.2: PRISMA report-assessment checklist

Appendix C

SciLRtool improvements
over Parsifal

Table II: SciLRtool improvements over Parsifal shortly described for each stage with corresponding information formats.

stage	new features	information format
Setting Up the Review	<ol style="list-style-type: none"> 1. Define a question type (PICOC, PIT, PO) 2. Define if a SLR or a SM will be performed 	Manual entry
Protocol Development	-	-
Literature Searching	Links to searched articles	-
Duplicate Checking	New SoffTF.IDF metric for strings comparison	-
Study Selection	-	-

Table II continued from previous page

<p>Quality Assessment</p>	<ol style="list-style-type: none"> 1. Nominating team members to be involved during Quality Assessment (main author) 2. Changing member's assigned articles manually or automatically (main author) 3. New "Each Quality Question has its set of Quality Answers" approach 4. Filtering articles by members assigned to them 5. New Conflict System, which resolves conflicts between team members 6. Filtering articles in conflict by members assigned to them 	<p>Manual entry</p>
<p>Data Extraction</p>	<p>-</p>	<p>-</p>

Table II continued from previous page

<p>Quantitative and Qualitative Synthesis</p>	<ol style="list-style-type: none"> 1. Downloading Data Analysis Files 2. Uploading Quantitative and Qualitative synthesis files 3. Entering numbers of studies involved in Quantitative and Qualitative synthesis 	<ol style="list-style-type: none"> 1. Downloading in PNG 2. Uploading in PNG, JPG, JPEG, BMP, GIF, DOCX, TXT 3. Manual entry
<p>Generation of Documentation</p>	<ol style="list-style-type: none"> 1. Filling final review documentation (timeline, contribution, etc.) 2. Filling protocol documentation (search strategy, data extraction strategy, etc.) 3. Downloading PRISMA's self-assessment checklist file for final review report assessment 4. Generating PRISMA's flow-diagram automatically 5. Publishing Evidence Synthesis for it to be publicaly available 	<ol style="list-style-type: none"> 1. Manual entry except acknowledgements, it is DOCX entry 2. Manual entry 3. Downloading in DOC 4. Included in conducting report DOCX file as PNG file 5. Downloading other Evidence Synthesis in DOCX (reporting parts) or XLS (data extraction sheet and study selection outcome)