Journal of Software Engineering Research and Development, 2022, 10:7, doi: 10.5753/jserd.2021.1978

Using evidence from systematic studies to guide a PhD research in Requirements Engineering: an experience report

Taciana Novo Kudo [**Universidade Federal de Goiás** | ***taciana@ufg.br***]
Renato F. Bulcão-Neto [**Universidade Federal de Goiás** | ***rbulcao@ufg.br***]
Auri Marcelo Rizzo Vincenzi [**Universidade Federal de São Carlos** | ***auri@ufscar.br***]
Érica Ferreira de Souza [**Universidade Tecnológica Federal do Paraná** | ***ericasouza@utfpr.edu.br***]
Katia Romero Felizardo [**Universidade Tecnológica Federal do Paraná** | ***katiascannavino@utfpr.edu.br***]

Abstract

Conducting systematic studies during a postgraduate program, such as systematic review, systematic mapping, and tertiary review, can benefit the project's success. They provide an overview of the literature considering currently available research findings, establish baselines for other research activities, and support decisions made throughout the research project. However, there is a shortage of research that presents systematic studies experiences in supporting academic projects. This paper's main contribution is reporting our experience on how the evidence found in tertiary and secondary studies positively influenced a PhD project's decisions. Initially, a tertiary study was conducted, followed by a systematic mapping. The evidence returned by the tertiary study led to the definition of the PhD research proposal in the Requirement Engineering field. Moreover, a systematic mapping contributed to the definition of the PhD research problem. From this experience in undertaking systematic studies to support a PhD project, the paper also presents lessons learned and recommendations to guide PhD students' decisions.

Keywords: *Evidence-based software engineering, Graduate education, Tertiary study, Secondary study*

1 Introduction

A systematic study[1] aims to identify, select, evaluate, interpret, and summarize available studies considered relevant to a topic or phenomenon of interest. Individual studies that contribute to a systematic study (Systematic Literature Reviews – SLR or Systematic Mapping – SM) are primary, while the systematic study itself is considered secondary.

Historically, systematic studies, especially SLRs, have been employed in the medical area and are recognised as critical components to support evidence-based medicine (Clarke and Chalmers, 2018). Inspired by the success in the medical field, Evidence-Based Software Engineering (EBSE) was first proposed to advance and improve the discipline of Software Engineering (SE) (Kitchenham et al., 2015). Currently, a larger community is formed around EBSE and composed of researchers who have conducted systematic studies in SE.

Informal literature reviews are relevant for research initiatives, especially in cases that use good practices. However, they lack scientific rigour, such as investigation bias. Reviews based on a rigorous process ensure auditable, reproducible, and unbiased results for all stakeholders.

One of the reasons systematic studies have been conducted in SE compared to informal reviews is its advantages, including the reduction of biases in results and the possibility of identifying and combining the main differences between data from the various studies selected in the review (Egger et al., 1997). Another advantage is identifying gaps in current research, which may suggest new research themes and provide a suitable way to position these themes in the context of existing research. Other benefits include (Kitchenham and Brereton, 2013):

- A well-planned systematic study avoids bias in the analysis of primary studies;
- A systematic study allows researchers to answer research questions that can not be answered based on a single primary study;
- A systematic study can help researchers to test theoretical hypotheses that otherwise could not be tested based on primary studies; and
- Results of a systematic study can be used to understand the efficacy and the efficiency of a method or a technology; alternatively, they can point out the strengths and weaknesses of methods and technologies under certain circumstances.

In that context, Felizardo et al. (2020) affirm that systematic studies are valuable to graduate students. Regarding the main benefits of conducting systematic studies during a PhD research project, the most significant one is providing an overview of the literature, finding out research opportunities, learning from studies, and providing baselines to assist new research efforts.

In particular, SMs can significantly benefit researchers in establishing baselines for further research activities, such as choosing a dissertation topic for a PhD degree considering research trends that can not be tracked over time (research gaps) (Souza et al., 2015). Another advantage includes using the reviews' findings to support decisions made in the research project.

[1] Throughout this work, the term "systematic study" encompasses Systematic Literature Review (SLR), Systematic Mapping (SM), as a more open form of SLR, and Tertiary Studies, as SLR of SLRs. Details on functional similarities and differences between SLR and SM are found elsewhere (Napoleão et al., 2017).

One expects that PhD students produce a compelling literature review. This review is a critical doctoral component since it allows students to thoroughly understand the topic they will work on and be familiar with the results obtained by other researchers. Therefore, secondary studies are the proper methodology to write a compelling literature review. Moreover, during the review conduction, students are trained in searching and selecting relevant literature, assessing the quality of the selected literature, and summarising/presenting the achievements. These are skills that every PhD candidate must procure during his/her doctoral. There are numerous motivations for conducting a secondary study, such as those reported in Felizardo et al. (2020):

- Systematic studies' results may identify suitable areas for future research – i.e., the original topic of investigation and the research questions to be answered during a PhD project – aiming at the advance of state of the art in the research topic;
- Those studies can replace traditional narrative literature providing the currently available research findings;
- Results of primary studies selected in a systematic study can be used as a baseline for comparison with ongoing, recent research results;
- The findings of systematic studies guide PhD research efforts, e.g., researchers could consider the systematic studies' findings for choosing appropriate research methods; and
- The systematic study may be published, externalising the acquired knowledge, contributing to the EBSE field.

Because of these advantages, several SE researchers advocate for PhD students using systematic studies (Clear, 2015; Pejcinovic, 2015; Kuhrmann, 2017; Kaijanaho, 2017). For example, Souza et al. (2015) describe a case of such a successful application of secondary studies to guide the decisions of a doctorate.

This article reflects upon our experience using systematic studies in developing a PhD project. Therefore, this study aims to present how systematic studies' findings impact an academic project. Specifically, the main goals of this research are to:

- present a successful case in which systematic studies had great importance in the conduction of a PhD project;
- exemplify how the best available evidence provided by systematic studies can base project's decisions;
- reinforce the importance of systematic studies in conducting a research project;
- report our experiences conducting secondary and tertiary studies as part of a PhD research project (Kudo, 2021); and
- inspire graduate students with our lessons learned and recommendations for undertaking systematic studies in their research projects.

In summary, one tertiary review and one secondary study were conducted to support a PhD project's decisions in the Requirements Engineering (RE) domain. Our main conclusion is that systematic studies have many advantages, and therefore, graduate students should consider doing at least one review during the doctorate.

The remainder of the paper is organized as follows. Section 2 introduces the software requirements patterns theme. Section 3 presents a PhD research project showing how systematic studies' results guided its conduction. Section 4 and 5 discuss the lessons learned and the threats to this work's validity, respectively. Section 6 addresses the related work, focusing on using systematic studies to guide a PhD research. Finally, Section 7 presents our concluding remarks.

2 Software Requirement Pattern

Incorrect, omitted, misinterpreted, or conflicting requirements usually result from poorly executed RE activities (Franch, 2015). As a result, software projects in such a scenario often struggle with software that does not meet quality requirements, cost and time overruns, and unsatisfied users.

Requirements reuse is a practical approach to mitigate those issues (Irshad et al., 2018): the core idea is reusing the knowledge acquired in previous projects to make RE activities more prescriptive and systematic.

A widely discussed reuse approach is the Software Requirement Pattern (SRP) abstraction, which aggregates behaviours and services observed in multiple similar applications (Withall, 2007). Usually, SRP guides requirements elicitation and specification through well-defined templates that can be reused in later specifications (Costal et al., 2019). For instance, one can create an SRP for representing a user authentication feature, commonly found in several applications, and make appropriate adaptations, if necessary.

An SRP's anatomy defines its structure and content, not the requirements that might result from it. However, to be helpful as a guide to writing software requirements, the SRP needs to consider situations likely to be encountered in the type of requirement built upon this SRP. Thus, SRP is more substantial than a requirement, and its specification is quite a demanding task (Withall, 2007).

There are SRP proposals for multiple sorts of systems such as embedded (Konrad and Cheng, 2002), cloud computing (Beckers et al., 2014), and call-for-tender (Costal et al., 2019). These studies demonstrate that SRP can promote greater efficiency in requirements elicitation, quality and consistency improvement in the requirements specification, gain in the development team's productivity, and better requirements management support.

3 From systematic studies to a PhD research project

This section's goal is three-fold: first, it introduces research method types that helped ground the doctoral project; second, it describes two systematic studies performed from planning to results analysis; and it demonstrates how these studies' results contributed to the definition of the PhD research proposal (Kudo, 2021).

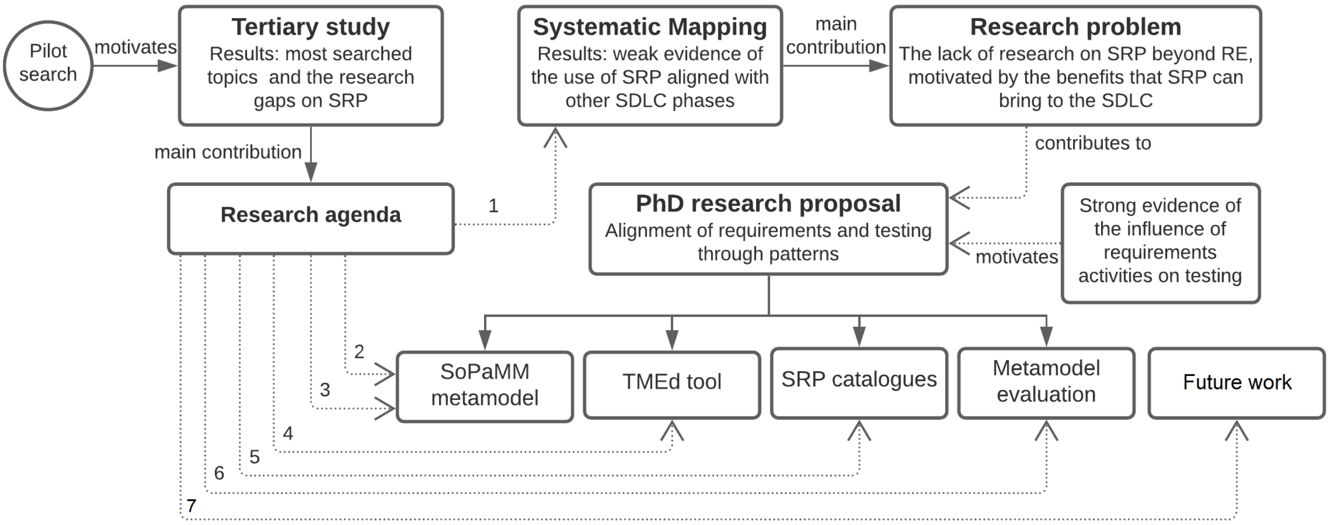
Pilot search
motivates
Tertiary study
Results: most searched topics and the research gaps on SRP
main contribution
Research agenda
1
Systematic Mapping
Results: weak evidence of the use of SRP aligned with other SDLC phases
main contribution
Research problem
The lack of research on SRP beyond RE, motivated by the benefits that SRP can bring to the SDLC
contributes to
PhD research proposal
Alignment of requirements and testing through patterns
motivates
Strong evidence of the influence of requirements activities on testing
2
3
4
5
6
7
SoPaMM metamodel
TMEd tool
SRP catalogues
Metamodel evaluation
Future work

Figure 1. An overview of our experience with PhD research decisions based on evidence provided by systematic studies.

Figure 1 illustrates how the best available evidence provided by the systematic studies — tertiary review (Kudo et al., 2020a) and systematic mappings (Kudo et al., 2019a,b) — guided decisions during the PhD project reported in this paper. Each step in Figure 1 is described next.

3.1 Research Method

Despite the differences between the methods, systematic studies (SLR and SM) are conducted using a process composed of three main phases (Kitchenham et al., 2015): planning, conduction, and reporting.

During the first phase, the review objectives and a protocol are defined. The protocol formalises the criteria and procedures for selecting, extracting, and summarising the data, including the research questions' definition through the search strategy to the final report. The protocol aims to reduce likely bias and ensure researchers can reproduce the review, adopting the same criteria and procedures.

According to the protocol, primary studies are retrieved, selected, and evaluated during the conduction phase. Then, in the reporting phase, studies that meet the review purpose are summarised, together with data extraction and synthesis that can be descriptive, complemented with a quantitative summary obtained through a statistical calculation.

SMs and tertiary studies are other types of reviews that complement SLRs. SM is a more open form of SLR, providing an overview of a research area to assess the quantity of evidence existing on a topic of interest (Petersen et al., 2015). A tertiary study is considered a review that focuses only on secondary studies (SLR/SM). The conduction of a tertiary study is proper in domains where some high-quality SLRs or SMs exist. The process used to conduct a tertiary study is the same as SLRs' (Kitchenham et al., 2015).

As depicted in Figure 1, we conducted a pilot search for SLRs/SMs on the SRP topic performed by third parties. We then conducted a tertiary review on the state of the art and practice in SRP (Kudo et al., 2020a) as we found some high-quality secondary studies on the same topic. In the tertiary review, we mapped the main topics covered and research gaps on SRP (the tertiary study's main contribution in Figure 1). We elaborated on a seven-item research agenda with lines of investigation (details in the next section) to approximate academics' and professionals' interests regarding improving requirements quality through SRP to lessen these gaps.

Remarkably, we noticed that secondary studies reported SRP only in the RE phase (*item 1* in Figure 1). As software requirements influence the remaining phases of the development process, we have identified a potential research gap about the benefits of using SRP in other development phases besides RE. This finding motivated us to conduct an SM to identify primary studies reporting the use of SRP in software design, construction, testing, and maintenance. The SM results pointed out eight primary studies in SRP applied to design, one to construction, one to testing, and none to maintenance. These results revealed a research problem to investigate: the lack of evidence on the SRP benefits for other development phases (the SM's main contribution in Figure 1).

As RE activities significantly impact other development phases, such as testing, we contributed to a novel approach to aligning RE and testing in which reuse through SRP and Software Test Patterns (STP) are core elements (PhD research proposal in Figure 1). An STP is an abstraction for generic testing solutions to recurrent behaviours from different scenarios. Unfortunately, recent literature reports that most companies still face adverse effects (cost, rework, and delay) from a weak alignment between requirements and testing (Bjarnason and Borg, 2017; Ebert and Ray, 2021).

Further details about how the findings of the tertiary review and the SM drove our efforts throughout the doctoral research are presented next.

3.2 Tertiary Study on Requirement Patterns

Recognised the importance of systematic studies for powerfully grounding a PhD research proposal, a question arose: *are there already systematic literature studies on SRP?* To respond to this question, a tertiary study was performed, as described follows.

The tertiary review employed the methodology used in classic tertiary studies in SE (Kitchenham et al., 2010). Besides, it took advantage of the StArt tool (Fabbri et al., 2016) support throughout the study protocol, from planning to reporting.

The tertiary review protocol included three general research questions (RQ) defined in the planning phase:

RQ1 – *What is the state of the art in requirement patterns?*
RQ2 – *What are the most searched topics on requirement patterns?*
RQ3 – *What are the current gaps in requirement patterns research?*

Activities performed in this tertiary review include automatic search, elimination of duplicate, selection of secondary studies on SRP, snowballing (Wohlin, 2014), quality assessment (Zhou et al., 2015), and data extraction and synthesis.

The following is the final search string used in the automatic search activity:

("requirement pattern" OR "requirement template") AND (survey OR "systematic review" OR "systematic literature review" OR "systematic mapping" OR "systematic literature mapping")

This process identified 40 secondary studies organised as follows: ACM DL (4), Engineering Village (13), IEEE Xplore (2), Science Direct (11), and Scopus (10). After excluding duplicate papers and applying selection criteria, four secondary studies remained.

Concerning the snowballing technique, we examined the bibliographic references of each of these four papers to identify further relevant studies. However, we found no relevant paper. Next, we assessed the quality of each secondary study using four criteria: description level of inclusion and exclusion criteria, search coverage, primary studies quality evaluation, and description level of primary studies.

As no paper was removed after data extraction, four secondary studies on SRP (Irshad et al., 2018; Palomares et al., 2017; Da Silva and Benitti, 2011; Justo et al., 2018) contributed to formulating answers to the review's research questions. The conclusions made are as follows:

- The number and publication dates of secondary studies on SRP (representing 44 non-duplicate primary studies) confirm that SRP is not a stagnant research topic, with contributions throughout the decade – (RQ1).
- The most searched topics regarding SRP are representation format, availability, scope, and purpose – (RQ2).
- Research gaps found include the professionals' unfamiliarity with SRP, few validations in the industry, the need for metrics and tools to enable the effective use of SRP in the industry, and the lack of secondary studies on how SRP benefits the software life cycle – (RQ3).

The analysis of those four secondary studies resulted in a research agenda to cover the gaps found between the states of art and practice in SRP. A research agenda is a formal plan of actions that summarises specific activities to guide the PhD conduction and the time to execute them. As depicted in Figure 1, the tertiary review's main contribution is a research agenda composed of the following items:

1. the demonstration of the benefits of SRP in other phases of the software development process in industry software projects – none of the secondary studies analysed explicitly identified this gap;
2. traceability mechanisms between requirements represented as patterns and artefacts produced in other development phases — this is another research topic not reported in any of the secondary studies analysed, and it is complementary to the *item 1*;
3. the joint use of SRP and existing and well-established methodologies in the software industry, such as agile approaches;
4. the development of tools that effectively support professionals' practices in the use of SRP;
5. the dissemination of current and future catalogues of SRP in a systematised manner;
6. the definition of objective metrics to help professionals measure the impact of the use of SRP as described in *items 1* to *3*;
7. collecting evidence of the effective use of SRP, particularly in the RE process of industry software projects.

3.3 SM on requirement patterns and software life cycle

According to Brereton et al. (2009), summarising the results of primary studies through secondary studies is a valuable research mechanism for providing knowledge of a given topic and supporting the identification of topics for future research. Therefore, influenced by *items 1* and *2* of the tertiary review's research agenda (see Figure 1), an SM was planned and conducted (Kudo et al., 2019a,b) to investigate the SRP usage in other phases of the software development life cycle (SDLC) and the traceability between SRP and specifications produced in these phases. Based on this goal, the SM included three research questions:

RQ1 – *At what SDLC phases are SRP used: design, construction, testing, and/or maintenance?*
RQ2 – *Is there evidence of SRP usage in practice at those SDLC phases?*
RQ3 – *Are there reported benefits of using requirement patterns at those phases? If so, what metrics are used to measure these benefits?*

A trade-off analysis between coverage and relevance of the results of a pilot search preceded the definition of the final search string presented next:

("*requirement pattern*" OR "*requirement patterns*" OR "*requirements pattern*" OR "*requirements patterns*") AND ("*software development*" OR "*development process*" OR "*life cycle*" OR *design* OR *construction* OR *coding* OR *implementation* OR *test* OR *integration* OR *maintenance*)

Activities performed in this SM study include automatic search, elimination of duplicates, the application of selection criteria, snowballing, quality assessment, and data extraction and synthesis. Target studies in this SLM are, thus, primary studies on SRP not employed in RE.

The automatic search identified 303 primary studies organised as follows: ACM DL (26), Engineering Village (106), IEEE Xplore (25), Science Direct (9), Scopus (76) and Web of Science (61). Ten primary studies remained after excluding duplicate papers, applying selection criteria, and full-text reading (155, 107 and 31 papers excluded, respectively).

After data extraction, we examined the bibliographic references and citations of these ten relevant papers found (i.e., backward and forward snowballing). Alike, we found no relevant additional paper.

We also assessed the quality of each primary study using nine criteria, including description level of the research problem and design, contributions, insights, limitations, and SRP-oriented criteria.

As a result, SM included a ten-primary-study group whose individual contributions regarding the use of SRP were analysed and synthesised in the form of a bubble chart, as depicted in Figure 2. Such contributions allowed us the formulation of the following answers to the research questions:

- eight primary studies used SRP in software design, one study in software construction and testing, and none in software maintenance – (RQ1).
- from these ten primary studies, eight are still at the proof of concept level, and none reports evidence of SRP usage in the software industry – (RQ2).
- there is only one primary study that demonstrates, through metrics and experimentation, that SRP integrated with software design artefacts implies significant development time savings; the corresponding metrics are DRR (degree of requirement realisation) and DPR (degree of pattern realisation) – (RQ3).

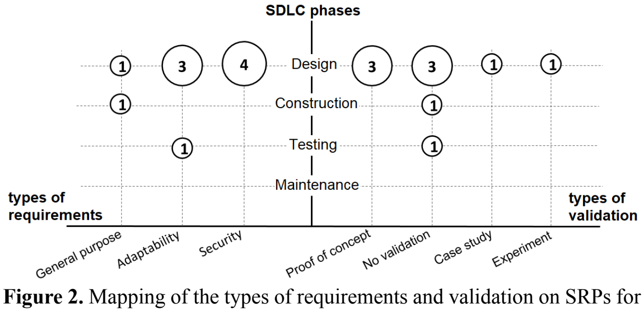

Figure 2. Mapping of the types of requirements and validation on SRPs for software design, construction, testing, and maintenance (Kudo et al., 2019b).

Then, we drew two conclusions from the analysis of these data:

1. there is an open field for research on SRP adoption in other SDLC stages (only 10); in contrast, most research efforts still focused on RE (76 primary studies).
2. there was little empirical evidence of the benefits of SRP beyond RE as we found only one case study and one experiment report.

Those SM results contributed to the PhD research problem definition, as shown in Figure 1: *the lack of research on SRP in other SDLC stages*.

3.4 Project's decisions based on the best available evidence

This section recaps the evidence found in our secondary studies that guided a PhD research in RE. Moreover, it associates each primary study developed by that graduate student with the pieces of evidence resulting from the tertiary review and the systematic mapping. Additional information on each thesis product is available in Kudo (2021) and Kudo et al. (2019c, 2020b,c, 2022).

Items 1 and *2* of the research agenda (Kudo et al., 2020a) inspired the conduction of the SM as none of the target studies described the benefits of SRPs in other SDLC phases, nor how to trace such support upon the development process.

The evidence found in the SM (Kudo et al., 2019a,b), in turn, led to the definition of the PhD research problem: *the lack of research on SRP beyond RE, motivated by the potential benefits that SRP can bring to the SDLC* (e.g., better quality specifications, reduced development time, and improved team productivity).

Moreover, the tertiary review's research agenda items combined with the strong influence of RE activities on software testing contributed to the definition of the PhD research proposal (see Figure 1): *the alignment of RE and testing phases through patterns*, i.e., SRP and STP.

Except for *item 7*, every research agenda item guided this PhD work experience, as illustrated in Figure 1. Following *item 2*, the PhD proposal endeavoured a novel SRP approach, called Software Pattern MetaModel (SoPaMM) (Kudo et al., 2019c, 2020b). SoPaMM is a metamodel that represents, relates, and classifies software patterns in general and SRP and STP in particular.

Influenced by *item 3*, SoPaMM borrows concepts and practices from the Behaviour-Driven Development (BDD) agile methodology (Chelimsky et al., 2010). In SoPaMM, Functional Requirement Patterns (FRP) are described as user stories associated with behaviours and test data using the Gerkhin language. FRP's behaviours, in turn, are linked to Acceptance Test Patterns (ATP) through test cases.

Inspired by *item 4*, the Terminal Model Editor (TMEd) tool was developed to help with the elaboration of SoPaMM-based pattern catalogues. A catalogue is a means of systematically gathering patterns, usually addressing the most common problems for a particular application domain. What differentiates TMEd from related tools (Palomares et al., 2011; Barcelos and Penteado, 2017) is that it handles other software patterns instead of SRP only.

With the support of the TMEd tool, four pattern catalogues with SRP and STP aligned (the research agenda's *item 5*) were developed. One supports the certification of electronic health record systems (Kudo et al., 2019c, 2020b; Martins et al., 2021), another represents behaviour-driven requirements of Internet of Things (IoT) systems, and two catalogues describe common functionalities and behaviours for user authentication and registration.

Finally, as the quality of the SoPaMM metamodel may impact the quality of pattern catalogues, which may influence software specifications quality, the Metamodel Quality Requirements and Evaluation (MQuaRE) framework was devised (Kudo et al., 2020c). MQuaRE is a metamodel quality requirements and metrics, a metamodel quality model, and an integrated evaluation process. Using MQuaRE, the SoPaMM's levels of compliance, conceptual suitability, usability, maintainability, and portability were recently evaluated in a controlled experiment (Kudo et al., 2022). Thus, MQuaRE is the first effort toward addressing the research agenda's *item 6*, providing objective metrics to evaluate metamodel's characteristics that may affect the quality of the software artefacts relied upon it.

Finally, the research agenda's *item 7* is a future work of the PhD thesis reported. It demands empirical work in collaboration with the software industry, a later effort of our research group.

4 Discussion

This section presents our lessons learned from undertaking systematic studies in an academic context. We believe these lessons can help PhD candidates perform systematic studies in their research.

1. *Choose the correct systematic study type* – PhD students can conduct three types of reviews: SLR, SM, or tertiary review. In particular, in the example of this paper, the PhD student conducted two different systematic literature studies, one tertiary and one secondary (an SM).
   The choice for conducting a systematic study must consider, for example, the amount of evidence available. An SM may be more appropriate than an SLR in domains with very little evidence related to a research topic, or the topic is vast. On the other hand, in domains where several SLRs exist already, it may be possible to conduct a tertiary review (an SLR of SLRs) to answer broader research questions.
   SMs may also be helpful to PhD students who are required to draw an overview of the existing evidence concerning a research topic. Despite that, it is essential to consider that the mapping study results may be more limited than the SLR. An SLR would be inappropriate if the research question is too vague or broad but also if the question is too narrow. The first case would yield hundreds of studies, and the second one would yield too few studies to be helpful.
   The conduction of a tertiary review is potentially less resource-intensive than conducting a new SLR. However, its conduction is dependent on sufficient quality SLRs being available. In our experience, the quality aspect of existing systematic studies on SRP geared us towards a tertiary review on that topic. Moreover, the tertiary review's results were determinant for the conduction of an SM.
2. *Use systematic studies conducted by third parties, when appropriate* – PhD students should consider three critical points to using systematic studies conducted by third parties:
   - It is indicated to use the results from already published SLR in SE since it meets the PhD's goal, i.e., the SLR research questions are related to the subject the student wants to investigate;
   - If the published SLR uses valid methods and was well-conducted to ensure its credibility; and
   - If the SLR is updated, avoid the understanding of the outdated state of the art. In this context, Mendes et al. (2020) recommend updating SLRs in SE using a third-party decision framework to decide whether they need updates.

   We have recently noticed an increasing number of SLR published in the SE area. However, occasionally we see that independent research teams have been conducting SLRs on the same topic, leading to duplication of the reviews and potentially wasted efforts.
   Therefore, before undertaking a systematic study, PhD students should ensure that a review is necessary, i.e., they should identify and review any existing study related to their research focus. In addition, when PhD students decide to conduct a systematic investigation, they must be aware that findings may be helpful for future students.
   Moreover, conducting an SLR that does not benefit only specific research can yield benefits: avoid duplicate work from other students, increase confidence in findings, and catalyse new collaborations among students and other researchers. In our experience, we conducted a novel SM on the SRP topic as the existing secondary studies focused on SRP solely applied to the RE phase. Research collaborations have arisen from the findings reported in this PhD experience (Martins et al., 2021; Kudo et al., 2022).
3. *The need for a previous pilot search step* – a pilot search is a reasonable first step before conducting systematic studies on the same or closely related target topic. A pilot search may reveal high-quality systematic studies on the topic of interest, motivating a tertiary review's conduction (as we did) rather than a new secondary study.
4. *Experience reduces effort* – establishing the first review protocol was a complex task and consumed considerable time and effort. However, it was essential for the assurance of the tertiary review quality. The knowledge acquired from the first review facilitated elaborating the SM protocol since procedures and forms were reused and adapted. Moreover, we could find the quality level of candidate studies more quickly, comparing them to studies previously read. The access to information was also faster as we already knew its organisation (i.e., the paper structure in the SRP context).
   Finally, an experienced researcher familiar with the review subject must compose the review team. In our experience, she supported in defining keywords and synonyms for the search string's main terms, synthesis of results, among other important decisions.
5. *Attention to open research issues in secondary studies* – when conducting a tertiary review, identify the open research issues described in each secondary study. Under the assessment for an experienced researcher, e.g., the PhD advisor, these open issues may result in candidate research gaps.
   In this PhD experience, we found three open issues found from the secondary studies analysis: the lack of

professionals' knowledge about SRP, the low number of evaluation research on SRP, and the need for tools for the effective use of SRP in the industry. These were essential to derive the seven lines of action of the tertiary review's research agenda.

6. *SM results may identify suitable areas for future research* – SM results are usually synthesised in a bubble chart, as depicted in Figure 2. When synthesising the findings of an SM, a PhD student should choose and group three relevant pieces of information, assign the most important one according to the study's objective to the ordinate axis, and distribute the remainder to the positive and negative abscissa axes. In our SM, SDLC phases, software requirements types, and research validation types are the ordinate axis and the negative and positive abscissa axes, respectively.
   Then, the PhD student should count the number of primary studies addressing two information axes simultaneously — for instance, crossing information from the ordinate and negative abscissa axes. The smaller the number of primary studies crossing two axes, the smaller the bubble size. In our experience, we identified suitable areas for future research in Figure 2: a few research on SRP in construction (one), testing (one), and maintenance (none), the predominance of studies in non-functional requirement patterns (8 of 10), and the need for more mature research on SRP in the SDLC (1 of 10).

5 Threats to validity

Finding all relevant papers on a particular theme is challenging. For this reason, both systematic studies included a previous pilot search under the supervision of an SRP expert, and a standard vocabulary for SE helped the search strings definition process. Furthermore, both search strategies comprise automatic search – in at least five relevant sources for SE – and the snowballing technique.

We also assess the quality of target papers to reduce a likely bias in the analysis and synthesis steps. The tertiary review protocol includes quality criteria widely accepted (Centre for Reviews and Dissemination, 2002; Cruzes and Dybå, 2011), and the SM protocol describes nine criteria regarding general (Jamshidi et al., 2013) and specific aspects of primary studies on SRP. Thus, both the quality criteria and the scores for each study analysed better weighed the value of individual studies after synthesising results, guaranteeing the evidence's reliability.

Besides, three researchers conceived the protocols of both systematic studies:

- Researcher A has expertise in RE and conducted the identification, selection, quality assessment, extraction, and synthesis of relevant secondary and primary studies;
- Researcher B is an expert in SE, and to mitigate the possibility of biases throughout the process, he verified all results phases; and
- Researcher C is the team leader with vast experience in SE, being consulted in the case of divergences not solved between researchers A and B.

Finally, we summarise our recommendations to common threats that PhD students can face during the planning and conduction of a systematic study. These general recommendations include:

- a previous pilot search for systematic studies on the graduate's topic of interest;
- the aid of both an expert and a standard glossary in the search string definition process;
- a hybrid search strategy to expand the search coverage;
- and the quality assessment of target studies and a well-coordinated team both to mitigate research bias.

6 Related Work

Felizardo et al. (2020) highlight the relevance of using secondary studies as a research methodology for conducting SE research projects. This study aimed to explore SE researchers' perceptions, mainly MSc and PhD students and their supervisors, about the value of secondary studies and how these perceptions impact decisions on conducting their research. The authors conducted two empirical research methods. First, they performed an SM to identify primary studies that used secondary ones as a research methodology for conducting SE research projects. Second, the authors surveyed SE researchers to determine their perception of the value of performing secondary studies to support their research projects. In summary, Felizardo et al. (2020) showed the main benefits of using secondary studies as a research methodology, identifying relevant research, finding reasons to explain why a research project should be approved, and supporting decisions made. The study reflected upon the value of secondary studies in developing academic projects.

In agreement with other authors (Dybå et al., 2005; Kitchenham et al., 2011; Zhang and Babar, 2011), Felizardo et al. (2020) highlight that a systematic secondary study is a valuable research mechanism for providing knowledge of a given topic and identifying gaps for future research. However, what is not clear yet is how this knowledge helps to conduct MSc/PhD research projects. One of the categories investigated in the SM shown by Felizardo et al. (2020) was the application of secondary studies. This classification summarises how the findings of such analyses can guide efforts in research projects. To the best of the authors' knowledge, only Souza et al. (2015)'s work fits this category.

Souza et al. (2015) show how the findings of the SM drove their research efforts in conducting a project on Knowledge Management (KM) in Software Testing. Among the SM results, the following stand out: (i) the central problem in software organisations related to software testing is low knowledge reuse rate and barriers in knowledge transfer; and (ii) reuse of test cases is the perspective that has received more attention. From SM results, the authors decided to conduct two SLRs, developed an ontology testing, and performed a survey to define a scenario to apply KM in software testing. The survey aimed to identify the testing activities in which

KM is more valuable or appropriate for reuse. From the survey results, the most suitable scenario in the software testing domain was established for applying KM. Finally, considering the survey results and ontology, a KM system was developed to manage testing knowledge repositories, such as test case reuse.

Comparatively, our work followed a similar approach. The results of the secondary studies served as a basis for follow-on research activities. Before accomplishing a secondary study, we and Souza et al. (2015) performed a tertiary review looking for secondary studies investigating the same topic. Likewise, based on the results of the tertiary review, an SM was planned and conducted in both studies, directing the project's decisions or defining other empirical approaches later used.

7 Conclusion

Especially for PhD research projects, originality is mandatory. Moreover, once students research the advanced state of the art, it is essential to do it correctly. This work reports our experience conducting a PhD research guided by systematic studies. We also highlight our lessons learned and recommendations that other researchers can use to guide their doctoral process.

We explained the criteria PhD candidates should choose to undertake the correct systematic study type and use a systematic study conducted by a third party. We also showed that a previous pilot search is desirable before conducting a secondary study on any topic. In addition, the experience acquired performing systematic studies reduces effort in similar works. Moreover, a deep analysis of the open research issues found in secondary studies may be valuable to delimit gaps that can gear other investigations on the same topic, e.g., including a new secondary study with a more profound view of that theme. We also explained how the results of a systematic mapping help identify future research. Finally, we also helped PhD students with recommendations to mitigate common threats they can face during a systematic study.

We believe any PhD candidate can adapt or reuse the lessons and recommendations outlined in our experience in research to any area of study.

Acknowledgements

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001.

References

Barcelos, L. and Penteado, R. (2017). Elaboration of software requirements documents by means of patterns instantiation. *J Softw Eng Res Dev*, 5(3):1–23.

Beckers, K., Côté, I., and Goeke, L. (2014). A catalog of security requirements patterns for the domain of cloud computing systems. In *Proceedings of the ACM Symposium on Applied Computing*, pages 337–342.

Bjarnason, E. and Borg, M. (2017). Aligning requirements and testing: Working together toward the same goal. *IEEE Software*, 34(1):20–23.

Brereton, P. O., Turner, M., and Kaur, R. (2009). Pair programming as a teaching tool: a student review of empirical studies. In *22nd Conference on Software Engineering Education and Training (CSEE&T' 09)*.

Centre for Reviews and Dissemination (2002). title=The Database of Abstracts of Reviews of Effects (DARE). *Effectiveness Matters*, 6(2):1–4.

Chelimsky, D., Astels, D., Helmkamp, B., North, D., Dennis, Z., and Hellesoy, A. (2010). *The RSpec Book: Behaviour Driven Development with Rspec, Cucumber, and Friends*. Pragmatic Bookshelf, Raleigh, NC, 1st edition.

Clarke, M. and Chalmers, I. (2018). Reflections on the history of systematic reviews. *BMJ Evidence-Based Medicine*, 23:121–122.

Clear, T. (2015). Follow the moon' development: Writing a systematic literature review on global software engineering education. In *15th Koli Calling Conference on Computing Education Research*, Koli Calling '15, pages 1–4. ACM.

Costal, D., Franch, X., López, L., Palomares, C., and Quer, C. (2019). On the use of requirement patterns to analyse request for proposal documents. In Laender, A. H. F., Pernici, B., Lim, E., and de Oliveira, J. P. M., editors, *Conceptual Modeling - 38th International Conference, ER 2019, Salvador, Brazil, November 4-7, 2019, Proceedings*, volume 11788 of *Lecture Notes in Computer Science*, pages 549–557. Springer.

Cruzes, D. and Dybå, T. (2011). Research synthesis in software engineering: A tertiary study. *Information & Software Technology*, 53(5):440–455.

Da Silva, R. C. and Benitti, F. B. V. (2011). Writing standards requirements: A systematic literature mapping. In *Proceedings of the 14th Workshop on Requirements Engineering*, pages 259–272, Rio de Janeiro, RJ, Brazil.

Dybå, T., Kitchenham, B. A., and Jørgensen, M. (2005). Evidence-based software engineering for practitioners. *IEEE Software*, 22(1):58–65.

Ebert, C. and Ray, R. (2021). Test-driven requirements engineering. *IEEE Software*, 38(01):16–24.

Egger, M., Smith, G., and Philips, A. (1997). Meta-analysis: Principles and procedures. *BMJ*, 315(1533–1537).

Fabbri, S. C. P. F., Silva, C., Hernandes, E. M., Octaviano, F., Di Thommazo, A., and Belgamo, A. (2016). Improvements in the start tool to better support the systematic review process. In *20th International Conference on Evaluation and Assessment in Software Engineering (EASE' 16)*, pages 21:1–21:5.

Felizardo, K. R., de Souza, E. F., Napoleão, B. M., Vijaykumar, N. L., and Baldassarre, M. T. (2020). Secondary studies in the academic context: A systematic mapping and survey. *Journal of Systems and Software*, 170:110734.

Franch, X. (2015). Software requirements patterns: A state of the art and the practice. In *Proceedings of the 37th International Conference on Software Engineering - Volume 2*, ICSE '15, pages 943–944, Piscataway, NJ, USA. IEEE Press.

Irshad, M., Petersen, K., and Poulding, S. (2018). A systematic literature review of software requirements reuse approaches. *Inf. Softw. Technol.*, 93(C):223–245.

Jamshidi, P., Ghafari, M., Ahmad, A., and Pahl, C. (2013). A framework for classifying and comparing architecture-centric software evolution research. In *2013 17th European Conference on Software Maintenance and Reengineering*, pages 305–314.

Justo, J. L. B., Benitti, F. B. V., and Leal, A. C. (2018). Software patterns and requirements engineering activities in real-world settings: A systematic mapping study. *Computer Standards & Interfaces*, 58:23–42.

Kaijanaho, A.-J. (2017). Teaching master's degree students to read research literature: Experience in a programming languages course 2002–2017. In *17^{th} Koli Calling Int. Conference on Computing Education Research (Koli Calling '17)*, pages 143–147, New York, NY, USA. ACM.

Kitchenham, B. and Brereton, O. (2013). A systematic review of systematic review process research in software engineering. *Information and software technology*, 55(12):2049–2075.

Kitchenham, B., Budgen, D., and Brereton, O. (2011). Using mapping studies as the basis for further research - a participant-observer case study. *Information and Software Technology*, 53(6):638–651.

Kitchenham, B., Budgen, D., and Brereton, P. (2015). *Evidence-Based Software Engineering and Systematic Reviews*. Chapman & Hall/CRC Innovations in Software Engineering and Software Development Series. Chapman & Hall/CRC.

Kitchenham, B. A., Pretorius, R., Budgen, D. an Brereton, P. O., Turner, M., Niazi, M., and Linkman, S. G. (2010). Systematic literature reviews in software engineering - A tertiary study. *Information & Software Technology*, 52(8):792–805.

Konrad, S. and Cheng, B. H. (2002). Requirements patterns for embedded systems. In *Proceedings IEEE Joint International Conference on Requirements Engineering*, pages 127–136, Essen, Germany. IEEE.

Kudo, T. N. (2021). *A metamodel for the alignment of requirement patterns and test patterns and a metamodel evaluation framework*. Phd thesis, Federal University of São Carlos, São Carlos-SP, Brazil. (In Portuguese).

Kudo, T. N., Bulcão-Neto, R. F., Macedo, A. A., and Vincenzi, A. M. R. (2019a). Padrão de requisitos no ciclo de vida de software: Um mapeamento sistemático. In *Proceedings of the XXII Iberoamerican Conference on Software Engineering (CIbSE' 19)*, pages 420–433.

Kudo, T. N., Bulcão-Neto, R. F., Macedo, A. A., and Vincenzi, A. M. R. (2019b). A revisited systematic literature mapping on the support of requirement patterns for the software development life cycle. *Journal of Software Engineering Research and Development*, 7:9:1–9:11.

Kudo, T. N., Bulcão Neto, R. F., and Vincenzi, A. M. R. (2019c). A conceptual metamodel to bridging requirement patterns to test patterns. In *Proceedings of the XXXIII Brazilian Symposium on Software Engineering*, pages 155–160, New York, NY, USA. ACM.

Kudo, T. N., Bulcão-Neto, R. F., and Vincenzi, A. M. R. (2020a). Requirement patterns: a tertiary study and a research agenda. *IET Software*, 14(1):18–26.

Kudo, T. N., Bulcão-Neto, R. F., and Vincenzi, A. M. R. (2020b). Uma ferramenta para construção de catálogos de padrões de requisitos com comportamento. In *Anais do WER20 - Workshop em Engenharia de Requisitos, São José dos Campos, SP, Brasil, August 24-28, 2020*. Editora PUC-Rio.

Kudo, T. N., Bulcão-Neto, R. F., Graciano Neto, V. V., and Vincenzi, A. M. R. (2022). Aligning requirements and testing through metamodeling and patterns: Design and evaluation. *Requirements Engineering Journal*, pages 1–25. (to be published).

Kudo, T. N., Bulcão-Neto, R. F., and Vincenzi, A. M. R. (2020c). Toward a metamodel quality evaluation framework: Requirements, model, measures, and process. In *Proceedings of the XXXIV Brazilian Symposium on Software Engineering, SBES 2020*, pages 102–107.

Kuhrmann, M. (2017). Teaching empirical software engineering using expert teams. In *SEUH*, pages 20–31.

Martins, M. C., Kudo, T. N., and Bulcão-Neto, R. F. (2021). Padrões de requisitos para sistemas de registro eletrônico de saúde. In *Anais do WER21 - Workshop em Engenharia de Requisitos, Brasília, DF, Brasil, August 23-27, 2021*. Editora PUC-Rio.

Mendes, E., Wohlin, C., Felizardo, K. R., and Kalinowski, M. (2020). When to update systematic literature reviews in software engineering. *Journal of Systems and Software*, 167:110607.

Napoleão, B., Felizardo, K. R., Souza, E. F., and Vijaykumar, N. L. (2017). Practical similarities and differences between systematic literature reviews and systematic mappings: a tertiary study. In *29^{th} International Conference on Software Engineering and Knowledge Engineering (SEKE' 17)*, pages 1–10.

Palomares, C., Quer, C., and Franch, X. (2011). Pabre-Man: Management of a requirement patterns catalogue. In *2011 IEEE 19th International Requirements Engineering Conference*, pages 341–342.

Palomares, C., Quer, C., and Franch, X. (2017). Requirements reuse and requirement patterns: a state of the practice survey. *Empirical Software Engineering*, 22(6):2719–2762.

Pejcinovic, B. (2015). Development and uses of iterative systematic literature reviews in electrical engineering education. *Electrical and Computer Engineering Faculty Publications and Presentations*, 327(1):1–10.

Petersen, K., Vakkalanka, S., and Kuzniarz, L. (2015). Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and Software Technology*, 64:1–18.

Souza, E. F., Falbo, R. A., and Vijaykumar, N. L. (2015). Using the findings of a mapping study to conduct a research project: A case in knowledge management in software testing. In *41^{st} Euromicro Conference on Software Engineering and Advanced Applications (SEAA'15)*, pages 208–215.

Withall, S. (2007). *Software Requirement Patterns*. Best practices. Microsoft Press, Redmond, Washington.

Wohlin, C. (2014). Guidelines for snowballing in systematic literature studies and a replication in software engineering. In *18th International Conference on Evaluation and Assessment in Software Engineering, EASE '14, London, England, United Kingdom, May 13-14, 2014*, pages 38:1–38:10.

Zhang, H. and Babar, M. (2011). An empirical investigation of systematic reviews in software engineering. In *5^{th} International Symposium on Empirical Software Engineering and Measurement (ESEM' 11)*, pages 1–10.

Zhou, Y., Zhang, H., Huang, X., Yang, S., Babar, M. A., and Tang, H. (2015). Quality assessment of systematic reviews in software engineering: a tertiary study. In *Proceedings of the 19th International Conference on Evaluation and Assessment in Software Engineering, EASE 2015, Nanjing, China, April 27-29, 2015*, pages 14:1–14:14.