Evaluation of Software Requirement Patterns for Electronic Health Record Systems

Mariana Crisostomo Martins
Taciana Novo Kudo
Renato F. Bulcão-Neto
maricrisostomo.martins@gmail.com
Federal University of Goiás
Goiânia, GO, Brazil

Abstract

Context: An Electronic Health Record System (EHR-S) captures, stores, presents, transmits, or prints health information that allows the individualization of a patient. An EHR-S can be classified into one or more categories, depending on its scope (e.g., an electronic prescription system). **Problem**: Despite existing several software requirements in common between these categories, much effort and time may be spent in the development of such complex systems, specially in requirements engineering. **Solution:** We developed a Software Requirement Patterns Catalogue (SRPC), which groups analogous behaviors and services from EHR-S categories into well-defined templates that can be reused in future specifications. **IS Theory:** We used Representation Theory to elaborate on software patterns representing EHR-S requirements. **Method:** Fifty-five undergraduate students produced EHR-S requirements specifications with and without SRPC support. Quantitatively, the completeness, scope, and consistency of these documents were evaluated, as well as the SRPC's usefulness and ease of use. Positive and negative aspects of the SRPC were also qualitatively analyzed. **Results:** The SRPC helps build more complete and correct specifications and conveys a perception of shortened elicitation and specification times. **Contributions:** A quantity-quality evaluation of an SRPC for EHR-S indicates requirement patterns improve the quality of specifications and save effort and time.

CCS Concepts

• **Software and its engineering** → **Software organization and properties**; • **Applied computing** → *Life and medical sciences.*

Keywords

Requirement pattern, reuse, experimentation, EHR-S, health

1 Introduction

In a software requirements reuse approach, analysts revisit and refine requirement specifications each time they are reused in similar software projects [12]. Software Requirement Patterns (SRP) [22] have demonstrated significant potential as a requirements reuse methodology. An SRP serves as an abstraction that consolidates common behaviors and services across different applications [20]. The advantages of SRP usage include improved consistency and quality in specifications, reduced elicitation and specification time, and enhanced requirements management [22].

Despite these benefits, the literature highlights a lack of experimental studies focusing on SRP [19]. For example, Da Silva and Benitti [8] and Barcelos and Penteado [1] examined SRP catalogues (SRPC) – collections of SRPs classified into one or more categories – specifically within the Information Systems domain. The authors [15, 19] emphasized the need for experimenting with SRP in other domains involving a large number of common requirements.

Electronic Health Record Systems (EHR-S) capture, store, present, transmit, or print health information that allows the identification of a patient [13]. Requirements allow an EHR-S be classified into categories, depending on its scope, such as electronic prescription or a telemedicine system. Despite the great number of common requirements across these categories, significant effort and time are still required for EHR-S development, specially in Requirements Engineering (RE) activities. The redundancy of these requirements makes this scenario a proper place for experimentation.

This paper evaluates an SRPC designed for EHR-S. The research question for this study is: "*To what extent does SRPC support the elicitation and design of requirements?*" We focus on a complex domain to assess how the SRPC usage facilitates the construction of high-quality specifications based on ISO 29148 [11]. Additionally, we analyze the SRPC usability and usefulness through the Technology Acceptance Model (TAM) [9].

The experimentation involved a case study to investigate a real-world instance, engaging undergraduate students from a Brazilian public university working with the S-RES domain for the first time. The objective was to measure the extent to which the catalog assists non-expert users [23]. Quantitatively, we evaluated the SRPC's usefulness and ease of use and the completeness, scope, and consistency of requirements specifications built upon the SRPC. We also conducted a survey with these students to collect insights regarding the description, comparison of knowledge, attitudes, and behaviors [23]. Finally, qualitative analysis was performed to identify both the positive and negative aspects of our SRPC [5, 6].

Although the evaluation has been performed in an academic setting, the results show pieces of evidence that the SRPC for EHR-S is useful and easy to use, helps improve the quality of requirements, and saves time in requirements elicitation and specification.

This paper is organized as follows: Section 2 reviews related work; Section 3 provides background on SRPC; Section 4 details the SRPC evaluation methodology; Sections 5 and 6 present results and discussion, respectively; and Section 7 brings final remarks.

2 Related Work

This section describes and compares related work considering SRP evaluation. We conducted a systematic study and found some proposed requirements patterns for some systems.

Da Silva and Benitti [8] developed and evaluated SRP for Information Systems with the support of a software tool. They performed a quasi-experiment involving a case study in which students were divided into experimental and control groups after learning about RE, patterns, and the tool as part of the experiment's design process. The study assessed quantity, reuse, time efficiency, and effectiveness. Additionally, the authors conducted an expert review, gathering specialists' opinions on efficiency, completeness of specification documents, support for writing requirements, and the tool's assistance in elicitation and specification activities.

Barcelos and Penteado [1] evaluated SRP for Information Systems through four case studies involving undergraduate students from computer science, electrical engineering, and information systems courses. Their research focused on measuring the quantity of functional requirements covered, usability and comprehension of the tool, productivity, time commitment, reuse perception, ease of use, and overall productivity.

Camelo and Alves [3] developed a privacy patterns catalogue and a guide to assist in forecasting. They conducted interviews with five professionals to evaluate the guide using Grounded Theory for data analysis. Additionally, they surveyed 18 participants using a questionnaire based on the Technology Acceptance Model (TAM) to assess the accessibility of the proposal's features. The results indicated that the approach supports requirements analysts in a practical and agile manner.

Carneiro et al. [4] developed an SRPC based on the Brazilian General Personal Data Protection Law (LGPD) and evaluated it with two experts in legislation and Software Engineering. To construct the catalogue, they performed a semantic and syntactic analysis of LGPD textual transcripts. Experts assessed the correctness and completeness of the catalogue through a questionnaire, revealing that approximately 85% of the privacy patterns were highly accurate and complete, with potential for additional patterns.

None of these works mentioned sought to evaluate quantitative and qualitative aspects together in a scenario such as S-RES, which benefits from SRPC by having a large number of duplicated requirements. Therefore this work presents different aspects of the aforementioned works:

- Quantitatively, we evaluate the completeness, correctness, and consistency of the requirements specifications generated by undergraduates, their perceptions of time spent on elicitation and specification, and the usefulness and ease of use of the SRPs;
- Qualitatively, we consider aspects of students' perception to generate theories about positive and to be improved aspects based on Grounded Theory.

The contributions of this work for IS is framework evaluated for a reuse approach of recurring requirements described as patterns, in a complex and open domain such as EHR-S ecosystem with systems encompassing people, machines, software, and processes to collect, transmit, process, and disseminate patient information.

3 Background

This section outlines two concepts as theoretical background: a software metamodel and the certification manual that both underlie our SRP catalogue.

3.1 SoPaMM

The Software Pattern Metamodel (SoPaMM) defines how software patterns can be written, organized, stored, related and classified. It addresses the concepts of Behavior Driven Development (BDD) describing functional requirement patterns as user stories using the notation *As, I can, So that.* And it is also possible to describe success and failure scenarios with the *Given-When-Then* syntax.

Our SRPC is then structured according to the SoPaMM grammar, which facilitates the organization of various patterns based on similar problems. The SoPaMM also supports patterns grouping into "*Bags*", which allows for the categorization of different types of patterns (such as functional and testing patterns) by subject. The description of non-functional requirements patterns (NFRP) consists of a composition of software system properties, including behavioral constraints and quality attributes.

For the development of our SRPC for EHR-system, the SoPaMM aligns requirement patterns and acceptance test patterns for validation purposes. The latest version of the metamodel enhances flexibility and is designed to be technology-agnostic. More details about the metamodel are found in [15, 17].

3.2 EHR-S Certification Manual

The chosen domain of EHR-S can take advantage of the SRPC approach by presenting several requirements common to different systems modalities. This overlap of requirements is observed in the EHR-S Certification Manual [13], which is an endeavor of the Brazilian Society of Health Informatics and the Federal Council of Medicine (BHIS/FCM). The specification of EHR-S functionalities in the manual is not standardized and easy to understand for software professionals with no (or little) knowledge in Health.

The manual consists of a list of requirements grouped into three sets: structure, content, and functionality (SCF) and security requirements level 1 and 2 (SRL1 and SRL2, respectively). These requirements apply to three categories of EHR-S: Electronic Patient Record (EPR), Telehealth, and Electronic Prescription, that are in turn organized into three maturity levels, being the Level 1 those requirements that every EHR-S should present.

4 Requirements Pattern Catalogue for EHR-S

The SRPC for EHR-S was built according to the structure and semantics of the elements of the Metamodel. To aid in the construction of the SRPC, the TMed tool [16] was used. The tool implements the SRPC instantiation based on the Metamodel. With *Eclipse* environment interface, the TMed tool helps domain and requirements patterns experts to build SRPCs.

With TMed tool, the authors created collections of patterns by subject *Bag* described in the certification manual (ex. *userAuthentication*). Following the Metamodel scheme, *pattern groupings* were created to represent the requirements.

Functional requirements were mapped to *functional requirements patterns* (FRP) as describing the problem to be solved, the solution, and the context of the use of the pattern, among other metadata. Each FRP is represented by one or more *features* in the form of user stories – *As*, *I can*, *So that*. For each *feature*, one or more behaviors are defined in the form of scenarios (success or failure) described in the form *Given*, *When*, *Then*. An example of FRP can be seen

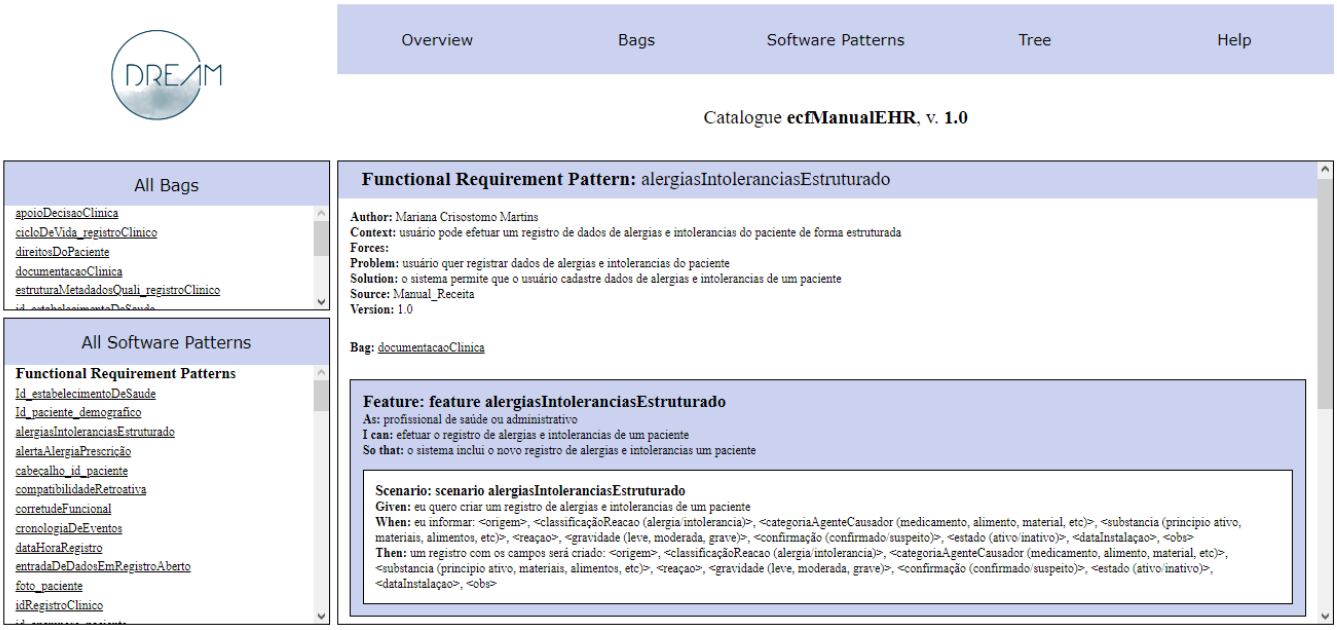

Figure 1: Snip from SRPC for EHR-S, which describes a FRP for recording a patient's allergies and intolerances.

in Figure 1 which refers to the registration of intolerances and allergies, where the structure of the pattern can be observed with its metadata and attributes.

For non-functional requirements, a *non-functional requirement pattern* (NFRP) describes system properties (e.g., security) described textually, without the existing structuring of a FRP. The SRPC for EHR-S used in this research includes 669 unique requirements of the ECF, NGS1, and NGS2 level 1 categoriy of version 5.1 of the BHIS/FCM certification manual, organized into 23 subject groupings, containing 170 FRP and 19 NFRP. The general method for creating SRPCs based on the SoPaMM grammar was followed [14]:

(1) All documentation regarding the requirements for the certification of electronic health record systems was gathered;
(2) All requirements were studied and classified as functional or non-functional requirement;
(3) If a pattern can be applied, it was used;
(4) Otherwise, a new pattern was developed, a name suggested and added to the list of candidates;
(5) After analyzing all the requirements, we reviewed the candidate patterns, duplicates and inconsistencies were resolved;
(6) Patterns were grouped in pockets defined according to the theme of the patterns;
(7) Write the patterns;
   - Non-functional requirement pattern (NFRP) is a composition of software system properties (behavioral constraints or quality attributes) described by textual attributes with name and description;
   - Functional requirement pattern (FRP) create Feature then it was used the user story syntax;

To build the catalogue, we consulted the manual and organized requirements by subject to facilitate the relationships between them, following the manual's structure. We carefully read and interpreted the requirements, defining the functional requirements through metadata that includes the *author*, *context* of use, *name* of the requirement pattern, *solution*, *source*, and *version*, as illustrated in Figure 1. Next, we identified one or more features associated with each requirement using the user's perspective, articulated through the structure: *As*, *I can*, and *so that*. Finally, we defined success and failure scenarios based on preconditions, actions, and expected results, using the framework of *given*, *when*, and *then*. The non-functional requirements identified in the manual were described as text due to the grammar of the metamodel.

By repeating the manual's requirements and the structure of their representation, we note that the SBIS requirements need an organizational structure that facilitates not only their understanding, but also the reuse of these requirements in the development and certification of S-RES.

Given that the TMed tool is designed to facilitate the development of SRPC—an uncommon practice in software industry projects—it is essential to instantiate SRPC to effectively support requirements engineering (RE) activities. To achieve this, the Dream tool [21] was developed. This tool features a user interface that allows requirements analysts to create requirements specifications by leveraging existing patterns within the SRPC. In this study, the Dream tool generates a Software Requirements Specification (SRS) that includes instances of all functional and non-functional requirements defined in the SRPC for Electronic Health Records Systems (EHR-S). This SRS serves as a benchmark for comparison with the SRS produced by participants in the case study. Figure 1 displays the main screen of the Dream tool's catalogue.

5 Evaluation Method

This section describes materials and methods used in evaluating the SRPC for EHR-S. The planning, characterization, metrics and dynamics of the application of the case study and the survey with undergraduate students are reported.

An evaluation protocol was developed, with the full text available Link of the evaluation protocol: https://zenodo.org/records/13996012.. The evaluation plan was submitted to and approved by the Ethics Committee, registered under CAAE number 51957221.5.0000.508

The evaluation followed the Goal-Question-Metric (GQM) method [2], which applies a top-down measurement approach: defining objectives, formulating research questions, and establishing evaluation metrics. The objective was to evaluate SRPC in the EHR-S scenario. The research questions (RQ) included: "[RQ1.] Does SRPC reduce requirements elicitation time? [RQ2.] Does SRPC reduce requirements specification time? Does SRPC improve completeness, correctness, organization, and writing of elicited requirements? Is SRPC usable and useful? Does SRPC aid in domain understanding? Would practitioners adopt SRPC in professional settings? What are the users' perceptions of SRPC?". Metrics included perceived elicitation and specification time, completeness, organization, correctness, ease of use, usefulness, and potential future adoption.

The evaluation assessed SRPC's impact on requirements elicitation and specification, considering SRS quality based on ISO/IEC standards [11] and perceived ease of use and usefulness according to the Technology Acceptance Model (TAM) [9]. Two prior SRPC evaluation studies [1, 8], identified through systematic mapping [19], were also referenced.

The case study was conducted during the 2021/1 academic semester with students from the Requirements Engineering and Software Requirements courses in the Computer Science (BCC) and Software Engineering (BES) programs. Due to the COVID-19 pandemic, classes were held remotely.

After learning software requirements fundamentals and RE process activities, students formed groups to develop an SRS, specifying and validating requirements using user stories and success/failure scenarios. The scope included an EHR-S system integrating individual clinics, teleconsultation, and electronic prescriptions. Participation required signing an Informed Consent Form (ICF).

Initially, students developed an SRS without SRPC under instructor supervision. Subsequently, they were introduced to reuse concepts and software requirement patterns before redoing the same task using SRPC, without prior detailed explanation of the tool. A summary of the case study roadmap is shown in Figure 2.

Participants were selected non-randomly, forming groups voluntarily. The study involved 39 BCC students and 30 BES students, grouped into teams of 4-5 members. Due to withdrawals, 34 BCC students and 21 BES students completed the perception survey. Both courses are mandatory and offered in the 7th semester, as per the curriculum.

For quantitative analysis, the study compared requirements specified with and without the SRPC catalogue, evaluating completeness, correctness, and scope of the SRS. Additional quantitative data was gathered from surveys on elicitation/specification time, SRS quality with the catalogue, and SRPC's usability and usefulness.

For qualitative analysis, open-ended responses were examined to identify positive aspects, areas for improvement, and criticisms or suggestions. The open-source tool Taguette [22] was used to tag and categorize student responses, facilitating semantic analysis and visualization.

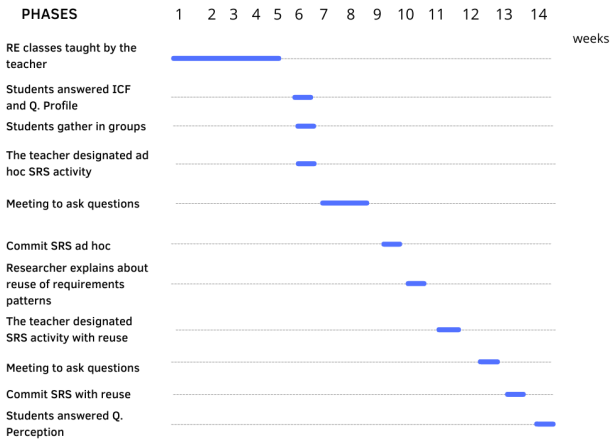

Figure 2: Steps of applying the case study.

6 Results

The results[1] are presented according to the type of analysis: (i) Student profile analysis, examining participants' backgrounds and experience levels; (ii) Quantitative analysis of the SRS, comparing the SRSs developed with and without SRPC support; (iii) Quantitative perception analysis, based on multiple-choice questions assessing students' perceptions of elicitation and specification time, SRS quality, and SRPC's ease of use and utility; (iv) Qualitative analysis of the SRS, gathering positive aspects, areas for improvement, and suggestions or criticisms.

6.1 Profile Student Analysis

Approximately 75% of students reported having some professional experience, though fewer than 20% had more than two years of experience. Regarding their professional roles, 60% worked as programmers, while 4% were engineers or requirements analysts. In terms of academic progress, 85% of students were between the 5th and 8th semesters, aligning with the 7th semester, when the two required courses are offered in the BCC and BES programs.

Regarding experience in requirements elicitation and specification, around 35% of students stated they had never engaged in these activities, either in academic or industry projects. Among those with industry experience, only 6% (2 students per course) had elicitation experience. For requirements specification in industry projects, only 2 BES students had experience, while none from BCC did.

In contrast, for academic projects, 65% of BES students reported having experience with elicitation and specification, compared to

[1]The graphs illustrating the results can be found on Zenodo.

36% of BCC students. This difference aligns with the practical nature of the BES program and its pedagogical approach. Despite BES classes being offered at night, no significant differences were observed between the profiles of students in both programs. Therefore, the student population was considered homogeneous, and the courses were analyzed together.

6.2 Quantitative Analysis

As described in Section 3.3, the software requirements specifications (SRS) developed by the student groups were evaluated based on completeness, consistency, and scope, and compared to the SRS generated automatically using SRPC for EHR-S. Each group produced an SRS without SRPC, which was then analyzed for complete, inconsistent, and in-scope requirements.

On average, the BCC groups specified 30 requirements without SRPC, while the BES groups specified 40 requirements. With the SRPC catalogue, these numbers increased to 171 requirements (BCC) and 100 requirements (BES). However, this growth was not evenly distributed between functional requirements (FR) and non-functional requirements (NFR). In particular, the completeness of NFRs was approximately 2

A similar trend was observed in terms of scope. The use of SRPC led to a 50% increase in in-scope requirements. However, the scope expansion for NFRs was 20% lower compared to FRs.

Regarding consistency, all inconsistencies were eliminated when using SRPC. Additionally, few groups modified the reused SRPC requirements, showing minimal deviation from the original source requirements in the ad hoc approach.

The following results reflect students' average perceptions, based on total and partial agreement ratings.

Elicitation and Specification Time Overall, 78% of students totally agreed and 20% partially agreed that SRPC improved elicitation and specification time. The contributions of each class can be observed in Figure 3.

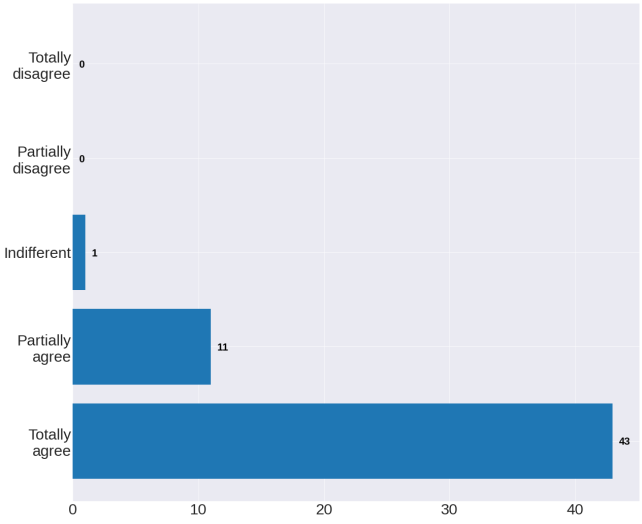

Figure 3: Perception of elicitation and specification time.

Regarding SRS quality, students reported that SRPC for EHR-S made the SRS: more complete (89%), better written (71%), better organized (83%), more correct (83%).

However, BES students were more critical of writing and correctness, as they showed a higher rate of partial agreement on these aspects. A detailed evaluation per class is shown in Figure 4.

Approximately 76% of students found SRPC for EHR-S easy to learn, while 83% reported gaining proficiency quickly. Additionally, 93% found it easy to remember how to use, and 80% considered it intuitive to use. However, BES students showed higher partial agreement on ease of learning and usage. The contributions of each class are displayed in Figure 5.

Regarding SRPC's usefulness, students reported that it: facilitated requirements elicitation (91%), was useful for elicitation (98%), facilitated specification (96%), helped them understand the domain (89%), could be valuable in professional practice (94%). The contributions of each class regarding SRPC's usefulness are illustrated in Figure 6.

6.3 Qualitative Analysis

The analysis data exclusively correspond to the information provided in the open questions, having been answered on the positive aspects (31 comments) and an improvement in the SRPC (34 comments) and about the criticisms and suggestions (24 comments).

Grounded Theory (GT), introduced by Glaser and Strauss [7], provides a systematic qualitative research approach to studying phenomena and generating new theories from data. It follows an iterative process involving research question formulation, participant recruitment, data collection, transcription, coding (open, axial, and selective), and theory writing. The process continues until theoretical saturation is reached.

From the three open-ended questions in the questionnaire, we identified 21 distinct codes in the responses. These codes were categorized into three main themes: catalogue benefits, usability, and criticisms or suggestions, as illustrated in Figure 7. Notably, many aspects related to usability overlapped with criticisms and suggestions.

The most frequently mentioned benefit of using the catalogue was the completeness of the generated specification document. Many responses also highlighted improved domain understanding and ease of elicitation. Other commonly cited advantages included time savings and knowledge reuse, while fewer participants mentioned catalogue usefulness and ease of specification as key benefits. Figure 7 presents the identified codes along with their frequency.

The G and D values in the figure represent magnitude and density, respectively. Magnitude indicates the number of times a code was applied to a response, while density reflects the relationships between different codes.

Examples of the positive responses obtained in the questionnaire and the association to the code can be seen below:

- "Many requirements would go unnoticed if not for the catalogue" - **Completeness of the specification document**
- "It was very positive in relation to the agility when carrying out the elicitation of requirements" - **Time savings**
- "It makes it much easier to delve into a specific part of the domain" - **Understanding the application domain**

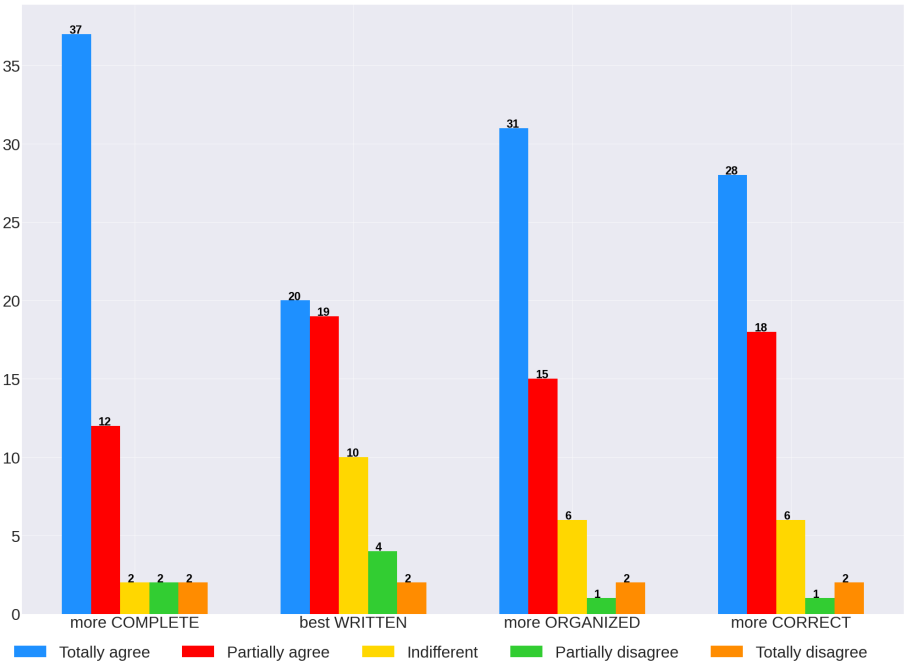

Figure 4: Perception of quality of the requirements specification.

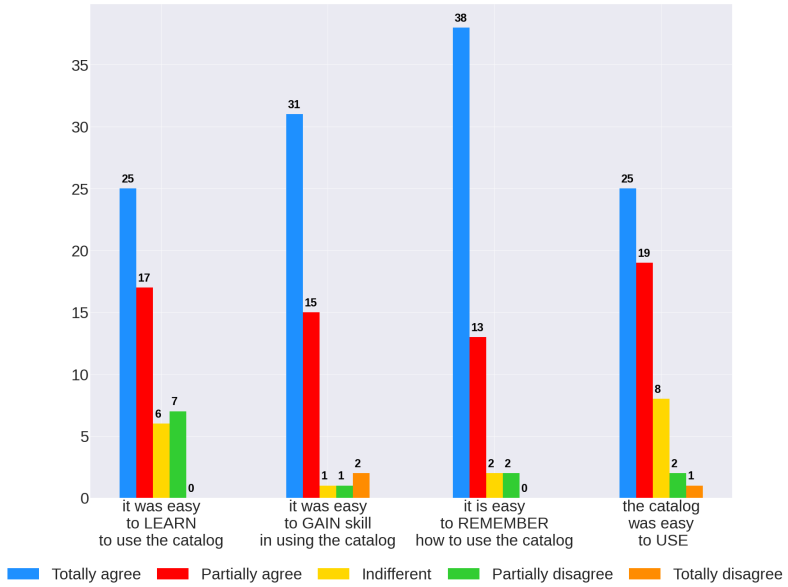

Figure 5: Perception of ease of use.

Regarding criticisms and suggested improvements, participants provided several insights: interface improvements, better organization, the need for a supporting tool. Despite these suggestions, some participants also praised the catalogue's organization and user-friendly interface.

Another common criticism was difficulty in visualizing patterns, which participants identified as an area needing improvement in future iterations. Based on these insights, usability aspects were categorized into both positive and negative aspects. While some participants suggested improvements to organization, interface design, and clarity, others appreciated the catalogue's simplicity and usability enhancements.

As shown in Figure 7, participants also pointed out: spelling and semantic issues, the need to enhance completeness, redundant

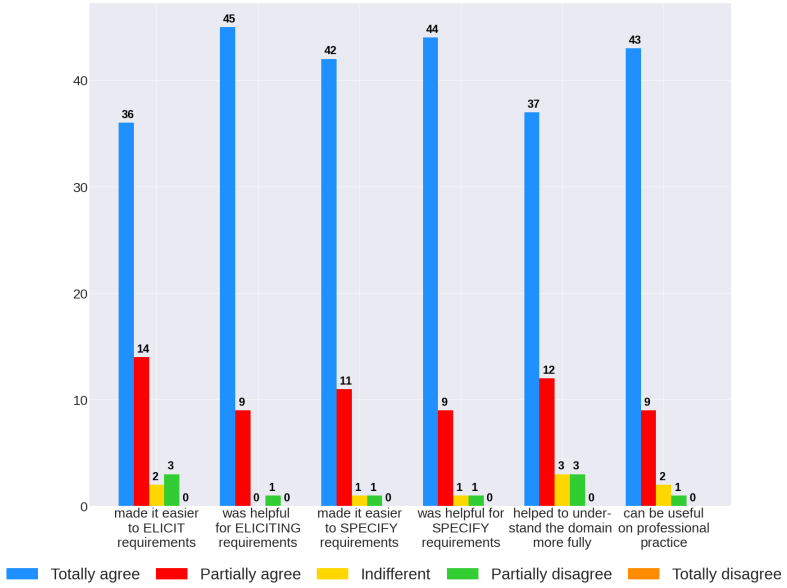

Figure 6: Perception of usefulness.

requirements, validation of the catalogue, integration with traditional elicitation and specification methods and the need to refine standardization practices.

Examples of criticisms and suggestions obtained in the questionnaire and the association with the code can be seen below:

- ”The catalogue interface could be better organized. To be more organized and intuitive” - **Improved interface aspects**
- ”As it has many repeated requirements, it might be interesting to better organize their division” - **Improved catalogue organization** and **Repeated requirements**
- ”The catalogue can be enhanced to include search functionality by Source and name. In the current state, it is difficult to find the patterns without having to go through each one individually” - **Difficulty in visualizing the patterns** and **Improvement of interface aspects** and **Need for a tool**

The benefits identified in this study—time savings, knowledge reuse, and improved completeness of the specification document—align with findings in previous research [20]. Additional advantages reported by participants include usefulness, ease of elicitation and specification, and enhanced domain understanding.

However, several critical observations were also raised: the need to improve completeness, redundant requirements Further evolution of the catalogue, semantic and spelling refinements, challenges in visualizing patterns, the necessity of validation and a supporting tool, which is consistent with previous research findings [14].

Interestingly, while some participants criticized the organization and interface, others praised them, indicating diverging perspectives on usability aspects.

7 Discussion

The results of the **quantitative analysis** of the SRS made by students with the aid of the SRPC for EHR-S show the set of indication:

- improved completeness and scope of the requirements specification.;
- reduced inconsistencies between the specified requirements;
- and enhanced quality of the requirements specification..

With the results obtained in the **qualitative analysis**, the catalogue contributions are:

Completeness of the SRS: Students indicated that the SRS developed with SRPC was more complete, which was confirmed by both quantitative and qualitative data, with 17 students citing completeness as a positive aspect.

Time saving: Students perceived that SRPC made elicitation and specification faster, with seven participants mentioning this benefit.

Organization of the SRS: While most students agreed that the SRS became more organized, ten participants praised the catalogue’s structure. However, 15 students suggested improvements in this area, and nine noted navigability issues. This feedback suggests the SRPC requires better organization, particularly in how FRP and NFRP are structured.

Quality of writing of the SRS: 71% of students expressed partial or total agreement regarding the quality of the SRS’s writing. Some students highlighted the standardization of requirements, such as user stories with validation scenarios, as a positive feature. However, six students pointed out syntax issues, and eight felt that the syntax of features and scenarios could be more intuitive.

Utility of SRPC: The utility of the catalogue was universally recognized as beneficial by the students.

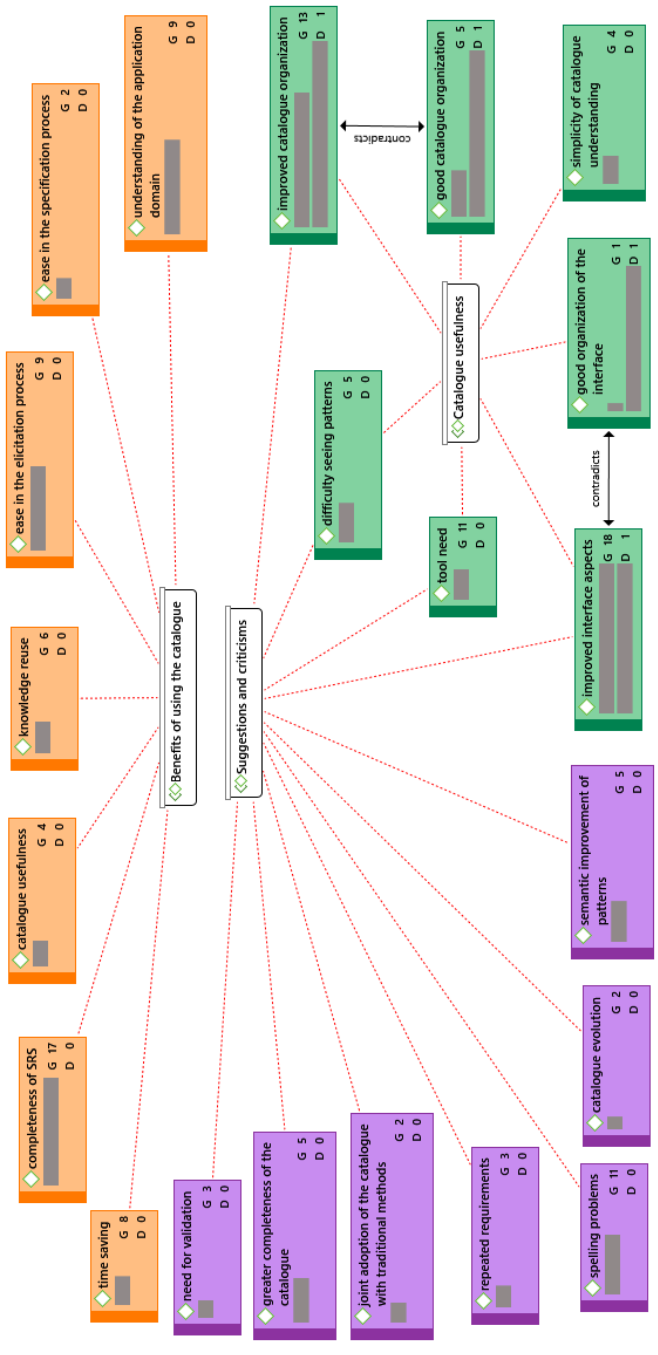

Figure 7: Codes obtained from qualitative analysis with Grounded Theory.

Understanding the domain: Nine students noted that SRPC helped them better understand the EHR-S domain, corroborating findings from the quantitative data.

As described by [10],triangulation is a research validation strategy that can be done in various ways, including methodological triangulation (within or between methods, e.g., quantitative and qualitative). The Figure 8 summarizes the contributions calculated from the catalog evaluation. We list the points that converge, complement and diverge when compared with the data from the quantitative and qualitative analysis. Finally, we illustrate the conclusions obtained.

The cross-analysis of quantitative and qualitative data revealed that using the catalog provided benefits such as time savings and greater completeness of the specification document.Usability, usefulness, and domain understanding were also positively perceived. However, the qualitative evaluation pointed out areas for improvement, such as the need for validation, enhancements in writing (semantics and spelling), and a better interface for accessing the catalog, reinforcing the need for an appropriate tool.

Although students were asked to evaluate only the specification document, their feedback also evaluated the catalogue itself. The results indicated that SRPC facilitates elicitation and specification, promotes knowledge reuse, saves time, and contributes to a more complete and understandable SRS. Additionally, qualitative analysis identified the need for validation, a tool to support the catalogue, and improvements in organization, interface, semantics, and spelling. While some students criticized the interface and organization, others considered them adequate.

7.1 Threats to Validity

The case study was conducted following a well-defined and objective protocol. Student participation was voluntary and did not entail any direct benefits; however, engagement was encouraged through the evaluation of their activity completion. It is acknowledged that prior evaluation history and ad hoc knowledge in some cases may introduce biases in the research. The planning and execution of the evaluation followed the Wohlin framework [23].

The study design assumed a realistic scenario, where requirements professionals typically possess some prior domain knowledge before utilizing requirements pattern catalogues. One potential critique of the evaluation design is the lack of a control group; ideally, students could have been divided into two groups—one following the proposed approach, while the other first used the SRPC and then attempted to elicit and specify requirements without it. However, the chosen design aimed to reflect the learning effect that naturally occurs in professional practice.

Care was taken to limit external influences on students' perceptions. To avoid bias, the SRPC and Certification Manual were not discussed during the Software Requirements Specification (SRS) production phase in an ad hoc format. Additionally, prior literature on the benefits of SRPC was not explicitly presented to participants, ensuring that their perceptions remained uninfluenced.

Assessment tools such as TAM (Technology Acceptance Model) and ISO/IEC 29148 were employed to define key evaluation aspects and determine the appropriate methodology for assessing the SRPC.

To refine the evaluation instruments, pilot testing of the questionnaires was conducted with three graduate students. While no changes were made to the content of the questions, their structure was revised for improved clarity, and the items were subsequently enumerated.

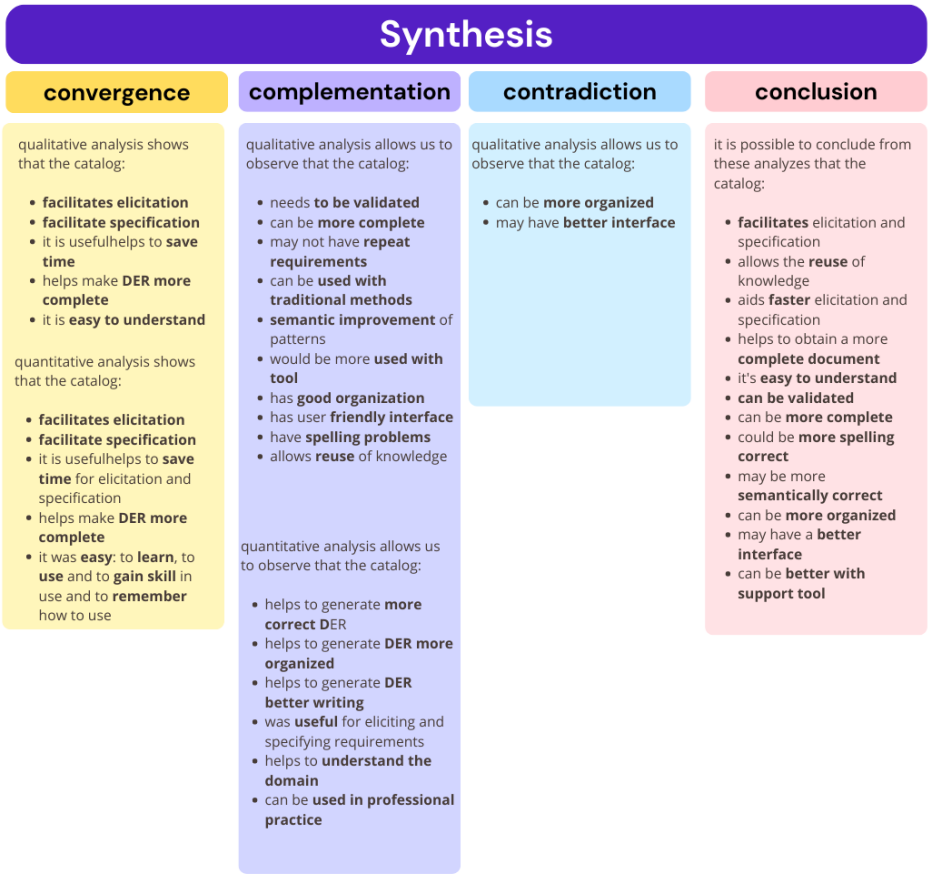

Figure 8: Synthesis of the qualitative analysis.

8 Final Remarks

This research contributes to the state of the art of requirements engineering with an experimental evaluation of SRPC, answering the gap pointed out in literature [15, 18, 19]. A case study and a *survey* were applied to undergraduates, who used a SRPC developed by the authors in an application domain not yet explored in the literature, i.e., EHR-S. The research also contributes with evaluation metrics and quantitative and qualitative vestige, referring to the use of SRPC in the elicitation and specification of requirements.

The indication pointed out by the research participants, is highlighted:

- enable the construction of a more complete, consistent SRS that better meets the scope of the domain, from a quantitative point of view;
- is perceived as a factor that enables building a SRS with higher quality, more complete, correct, and organized and better written;
- is perceived as useful and easy to use;
- is perceived as a factor that makes requirements elicitation and specification activities faster.

The contribution of this work is to provide a catalogue of evaluated requirements patterns that enables the agile, complete and correct development of requirements for EHR information systems. The catalogue also allows the use of the benefits of a use-based approach for the development of information systems in a complex domain such as healthcare.

With the evidence obtained and the considerations of the research participants with the use of SRPC for EHR-S, it can be pointed out as future works: the validation of the SRPC with specialists in the EHR-S domain (ex., representatives of companies with EHR-S certified by BHIS/FCM), realize experiments with SRPC for EHR-S, however, with the use of the tool in the automation of requirements elicitation and specification activities, and evolve SRPC in the way in which NFR in specified, possibly with previous modifications to the grammar of the metamodel. The same evaluation path can be done for other information systems with common requirements.

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

References

[1] L. V. Barcelos and R. Penteado. 2016. Especificação de Requisitos no Domínio de Sistemas de Informação com o Uso de Padrões. In *Proceedings of XIX Ibero-American Conference on Software Engineering, CIbSE 2016*. 338–351.

[2] Victor R. Basili. 1992. *Software modeling and measurement: the Goal/Question/Metric paradigm*. Technical Report. USA.

[3] Moisés Neves Camêlo and Carina Cristina Montalvany Alves. 2023. G-Priv: Um Guia para Apoiar a Especificação de Requisitos de Privacidade em Conformidade com a LGPD. *Braz. J. Inf. Syst.* 16 (2023), 1–32.

[4] Cinara Carneiro, Taciana Kudo, and Renato Bulcão Neto. 2024. Um método para transformação de requisitos legais em padrões de requisitos de software: Um estudo com a LGPD. In *Anais do XXVII Congresso Ibero-Americano em Engenharia de Software, CIbSE 2024*. 348–355. https://doi.org/10.5753/cibse.2024.28460

[5] Kathleen C Charmaz. 2006. *Constructing grounded theory*. SAGE Publications, Thousand Oaks, CA.

[6] Juliet M. Corbin and Anselm Strauss. 1990. Grounded theory research: Procedures, canons, and evaluative criteria. *Qualitative Sociology* 13, 1 (1990), 3–21. https://doi.org/10.1007/bf00988593

[7] J. M. Corbin and A. Strauss. 1994. Grounded theory methodology: An overview. In *Handbook of qualitative research. Sage Publications, Inc.* 271–285.

[8] Rodrigo Da Silva and Fabiane Benitti. 2013. Evaluation of a systematic approach to requirements reuse. *Journal of Universal Computer Science* 19, 2 (2013), 254–280.

[9] Fred D. Davis. 1989. Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Quarterly* 13, 3 (1989), 319–340. http://www.jstor.org/stable/249008

[10] Norman K. Denzin. 2017. *The Research Act: A Theoretical Introduction to Sociological Methods*. Routledge. https://doi.org/10.4324/9781315134543

[11] International Organization for Standardization. 2018. *Systems and software engineering — Life cycle processes — Requirements engineering*. Standard. International Organization for Standardization, Geneva, CH. 92 pages.

[12] Mohsin Irshad, Kai Petersen, and Simon Poulding. 2018. A systematic literature review of software requirements reuse approaches. *Information and Software Technology* 93 (2018), 223 – 245. https://doi.org/10.1016/j.infsof.2017.09.009

[13] L. G. Gasparini Kiatake, L. A. Virginio Junior, M L. da Silva, and O. A. Chamelette Sanzovo. 2020. Manual de Certificação de Sistemas de Registro Eletrônico em Saúde. http://sbis.org.br/documentos-e-manuais/ Sociedade Brasileira de Informática em Saúde. Versão 5.0. Novembro de 2020..

[14] Taciana Novo Kudo, Renato de Freitas Bulcão-Neto, Valdemar Vicente Graciano Neto, and Auri Marcelo Rizzo Vincenzi. 2022. Aligning requirements and testing through metamodeling and patterns: Design and evaluation. *Requirements Engineering* (2022), 1–19.

[15] Taciana N. Kudo, Renato F. Bulcão-Neto, and Auri M.R. Vincenzi. 2020. Requirement patterns: a tertiary study and a research agenda. *IET Software* 14, 1 (Feb. 2020), 18–26. https://doi.org/10.1049/iet-sen.2019.0016

[16] Taciana N. Kudo, Renato F Bulcao-Neto, and Auri MR Vincenzi. 2020. Uma Ferramenta para Construçao de Catálogos de Padroes de Requisitos com Comportamento. In *Anais do WER20 - Workshop em Engenharia de Requisitos, São José dos Campos, SP, Brasil, Agosto 24-28, 2020*. Editora PUC-Rio, 1–14.

[17] Taciana N. Kudo, Renato F. Bulcão-Neto, and Auri M. R. Vincenzi. 2019. A Conceptual Metamodel to Bridging Requirement Patterns to Test Patterns. In *Proceedings of the XXXIII Brazilian Symposium on Software Engineering*. ACM, 155–160. https://doi.org/10.1145/3350768.3351300

[18] Taciana Novo Kudo, Renato de Freitas Bulcão Neto, Auri Marcelo Rizzo Vincenzi, and Alessandra Alaniz Macedo. 2019. 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 (Dec. 2019), 9:1 – 9:11. https://doi.org/10.5753/jserd.2019.458

[19] Mariana C. Martins, Taciana N. Kudo, and Renato F. Bulcão-Neto. 2021. Estado da Arte e da Prática sobre Catálogos de Padrões de Requisitos. In *Proceedings of the Ibero-American Conference on Software Engineering, CIbSE 2021*. 1–14.

[20] C. Palomares, C. Quer, and X. Franch. 2014. Requirements reuse with the PABRE framework. *Requirements engineering magazine* 2014 (Jan 2014), 1–10. http://re-magazine.ireb.org/issues/2014-1-learning-to-fly/requirements-reuse/

[21] Pollyana Ribeiro, Ernesto Veiga, Mariana Martins, Auri Vincenzi, Taciana Kudo, and Renato Bulcão-Neto. 2022. Tool support to aligning requirements and testing through behaviour-driven requirements patterns. In *Anais do XXV Congresso Ibero-Americano em Engenharia de Software, CIbSE 2022*. 292–299. https://doi.org/10.5753/cibse.2022.20979

[22] S. Withall. 2007. *Software Requirement Patterns* (1 ed. ed.). Microsoft Press.

[23] Claes Wohlin. 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*. 38:1–38:10.