Requirements Communication in Safety-Critical Systems

Jéssyka Vilela[1], Jaelson Castro[1], Luiz Eduardo G. Martins[2], Tony Gorschek[3], and Camilo Almendra[1,4]

[1] Universidade Federal de Pernambuco, Brazil {jffv,jbc}@cin.ufpe.br
[2] Universidade Federal de São Paulo legmartins@unifesp.br
[3] Blekinge Institute of Technology, Sweden tony.gorschek@bth.se
[4] Universidade Federal do Ceará, Brazil camilo.almendra@ufc.br

Abstract. Context: Safety-critical systems (SCS) are mainly controlled by software. Accordingly, the development of these systems must be carefully planned since inadequate or misunderstood requirements have been recognized as the major cause of a significant proportion of accidents and safety-related catastrophes. **Objective**: We investigate the integration and requirements communication in the requirements engineering (RE) process among different parties when developing SCS. **Method**: We used a Systematic Mapping Study as the basis for our work. **Results**: We analyze the challenges and needs involved, application context, research type, evaluation methods, type of contribution, domain, requirements activity as well as languages and tools used to specify safety requirements. Furthermore, we also analyze stakeholders involved, communication format, and for what safety standards have the approaches been proposed. **Conclusions**: We believe the results of such a study will benefit both researchers and practitioners. This information contributes to setting up possible collaborative networks and as a reference when developing new research projects.

Keywords: Safety-Critical Systems · Requirements Communication · Requirements Engineering · Safety Engineering · Systematic Literature Review

1 Introduction

Safety-critical systems are mainly controlled by software nowadays [10][4]. New generations of medical devices, means of transportation (aircraft, automated trains and cars), nuclear power generating stations, banking and investment systems, as well as a growing number of automated systems rely on software to enable new functions, provide pre-existing functions more efficiently, and reduce time to service a user need as well as the effort and competence required by people providing services.

There are many cases in the literature [6] where inadequate or misunderstood requirements were the major cause (not coding or implementation) of a significant proportion of accidents and safety-related catastrophes. Therefore, these systems must be carefully specified, demanding more rigorous RE approaches [6].

RE focuses on good specification practices but has yet to find working solutions for effective requirements communication. Furthermore, the competences of requirements engineers and safety engineers normally work independently of each other and have inherently different tools and engineering practices - resulting in a lack of coordination that can compromise the quality of safety analysis and safety specifications [11].

In this work, we investigate the approaches proposed to improve the integration of requirements communication in the RE process among different parties when developing SCS. We adopted the systematic mapping study as a research method. We believe the results of such novel study will benefit both researchers and practitioners. The review will provide researchers with important research gaps regarding the requirements communication between safety and RE. For the industrial readership, the review will provide practitioners with useful information about the state-of-the-art and advances so far. This information contributes to setting up possible collaborative networks and as a reference when developing new research projects.

This paper is organized as follows. Section 2 presents background and related work. The research methodology adopted to conduct the mapping study is presented in Section 3. The results and the analysis related to our research questions are presented in Section 4. Our conclusions are presented in Section 5.

2 Background and Related Work

SCS are those software or system operations that, if not performed, performed out of sequence, or performed incorrectly could result in improper control functions, or lack of control functions required for proper system operation. Such problems can directly or indirectly cause or allow a hazardous condition to exist [6].

In order to set the scope and make clear the adopted definition of requirements communication used in this mapping study, and to ensure consistency throughout this paper, we discuss this concept in the next section.

2.1 Requirements Communication

Requirements communication is a traversal process of exchanging information [3] about the requirements among all stakeholders [1] involved in the system lifecycle. This concept does not only comprise the communication itself but the specification and analysis of all artifacts involved in the RE process. Since changes occur throughout the project, requirements communication must also continue during the entire life cycle [1].

This process aims to achieve a shared understanding [3] of the system's requirements to increase completeness and correctness of the requirements specifications. It encompasses all the activities needed to inform the stakeholders of the content, meaning and status of requirements. The elicited requirements need to be communicated, and changes to those requirements negotiated and communicated among all affected roles, e.g. requirements engineers, developers, and testers [1].

2.2 Related Work

The communication of requirements among different parties in the development of SCS is critical for the quality of the system. This occurs since requirements should be understood in the same way by different roles in the development. We argue that the requirements engineers and safety engineers should collaborate, exchange information and work jointly and in an iterative way. However, they usually work independently of each other and have inherently different tools and engineering practices - resulting in a lack of coordination that can compromise the quality of safety analysis [12], and therefore, the quality of safety specifications.

Communication problems in software development were investigated by some authors such as Brady et al. [2], Pernstal [8], Rasmussen and Lundell [9], Wang et al. [12] as well as Nakamura et al. [7]. Although these works explore several challenges related to the integration of RE and safety, little has been done to date to perform an extensive identification and mapping, in a comprehensive manner, the state-of-the-art on the communication of requirements among different parties in the development activities/process when developing SCS. Hence, to the best of our knowledge, this is the first mapping study with such specific focus. In the next section, we detail our research protocol.

3 Research Methodology

In this section, we present the design and the execution of the mapping study. The research methodology used was based on the guidelines and template proposed by Kitchenham and Charters [5].

The focus of this review is the integration between RE and safety engineering and the requirements communication among different parties during the RE process. We included only English primary studies, published in any year until February 2018, that address in their objectives the communication in the RE process among different parties when developing SCS, related Requirements and Safety in the context of RE process, or covered Design in the relationship with Requirements and Safety.

We excluded Secondary studies, Short-papers ($\leq$ 3 pages), Duplicated studies, Studies clearly irrelevant to the research, taking into account the research questions, Gray literature, Redundant paper of same authorship, Publications whose text was not available (through search engines or by contacting the authors), and Studies whose focus was not the communication in the RE among different parties when developing SCS or safety requirements specification.

Our study was guided by the research questions presented in Table 1. This table also presents the descriptions and motivations of each question. The search strategy included two types of search to find studies relevant to the scope of the review. The first type was an automatic search, using a string validated by experts on RE and SCS. The second strategy was the manual inclusion of papers well-known about requirements communication.

Table 1. Research questions and motivations.

Research Question	Description and Motivation
RQ1. What challenges have been identified pertaining to the communication among engineers during the RE process when developing SCS?	The goal is to identify the challenges addressed in the literature regarding the communication among engineers during the RE process for SCS. The results obtained will be useful to identify emerging trends and provide an overall view of the problems tackled in the literature.
RQ2. Which approaches have been proposed to improve the communication in the RE process among engineers when developing SCS?	The aim is to identify and analyze the approaches proposed to improve the communication in the RE process among engineers when developing SCS.
RQ2.1. What are the types of these approaches?	We want to analyze the types of the approaches to understand how the community is evolving regarding communication in SCS. Therefore, this question intends to classify the approaches by its type, for instance, Approach, Framework, Method, Tool, Process, Model, Methodology, Template, Comparison, Metrics, Protocol, Checklist and Language, proposed in the approach.
RQ2.2. For which domains were these approaches proposed?	Domain understanding narrows the amount of domain knowledge to be shared and lays the foundation for successfully communicating domain concepts. Accordingly, this question provides an overview of the domains (Generic, Automotive, Avionics, Medical, Railway and so on) for which the approaches were proposed.
RQ2.3. What RE activities were supported by these approaches?	We are concerned with investigating the proposals in relation to requirements communication, hence this question provides a starting point to understand what are the main activities (elicitation, analysis, specification, validation and management) of the RE process supported by the approaches.
RQ2.4. Which requirements specification languages are used by these approaches?	The languages play an important role in the requirements communication process. Therefore, the intent of this question is to identify the requirements specification languages adopted in the development of SCS.
RQ2.5. Which tools are used for the requirements specification?	The process of development of SCS encompasses the elaboration of many documents and models. Hence, this question maps the tools used to develop the requirements specification of SCS.
RQ2.6. For which stakeholder were they proposed?	The requirements should be understood by all stakeholders involved in the development process. Accordingly, this question aims to analyze the stakeholders involved in the proposed approaches.
RQ2.7. What are the communication formats used?	The intent is to analyze the formats used for the requirements communication (Model-based collaboration, Process support, Awareness, Collaboration infrastructure, Artifacts-based, Analysis tools, and Face-to-face verbal communication).
RQ2.8. For what safety standards have the approaches been proposed?	Considering that SCS should be certified by regulatory bodies, this question intends to analyze for what safety standards have the approaches proposed to improve the communication in the RE process among engineers when developing SCS been proposed.

We developed a review protocol in which the main elements are as follows: the ***selected resources*** chosen were Science Direct, SpringerLink, ACM Digital Library, IEEE Xplore, Scopus, and Compendex; the ***search method*** used web search engines; the ***population*** was composed of peer-reviewed publications reporting approaches to improve the communication in the RE process among parties when developing SCS; the aim of the ***intervention*** was to collect empirical evidence in relation to the research questions.

We developed the search string by specifying the main terms of the phenomena under investigation (SCS and requirements communication). After several iterations, we defined the search string below to search within keywords, title, abstract and full text of the publications:

(1) *("safety critical system" OR "safety critical systems" OR "safety-critical system" OR "safety-critical systems") AND*
(2) *("requirements engineering" OR "requirements engineer" OR "requirements team" OR "requirements specification") AND*
(3) *("safety requirements" OR "safety engineering" OR "safety engineer" OR "safety team" OR "safety analysis" OR "safety specification") AND*
(4) *("communication" OR "integration" OR "interaction" OR "collaboration" OR "alignment" OR "understanding" OR "relationship" OR "share" OR "sharing" OR "combination" OR "interrelation" OR "interplay" OR "interdependency")*

Figure 1 depicts the steps of the selection process showing the number of studies in each step. The data were extracted from the 60 primary studies using an extraction form fully aided by the StArt tool.

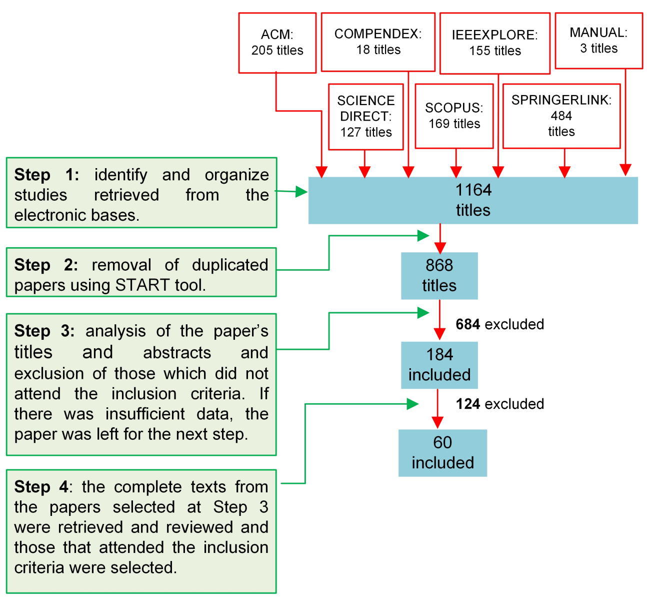

Fig. 1. Paper selection flowchart.

The quality assessment (QA) of selected studies was achieved by a scoring technique to evaluate selected studies in terms of credibility, completeness and relevance. All papers were evaluated against a set of 20 quality criteria whose assessment instrument we developed and used in a previous work [11] and described in the supplementary material.

4 Results and Analysis

A total of 60 studies satisfied the inclusion criteria and their data were extracted. The quality scores of the selected studies are presented in Table S1 on supplementary material [5]. The mean of quality was 83.12%, hence, the overall quality of the selected studies is acceptable.

4.1 Overview of the Studies

The reviewed papers were published between 1994 and February 2018. From a temporal point of view (Figure 2), we can notice that the number of studies about requirements communication in SCS is low over the years. Despite the apparently increasing number of studies on this topic (peak in 2009-2011), this result corroborates the statement that the collaboration of safety analysis and RE has been somewhat neglected [6]. It is also worth noting that, as the search process of this review was performed in February 2018, a slight decrease in the number of publications would be expected in 2018 because some papers might have been in press.

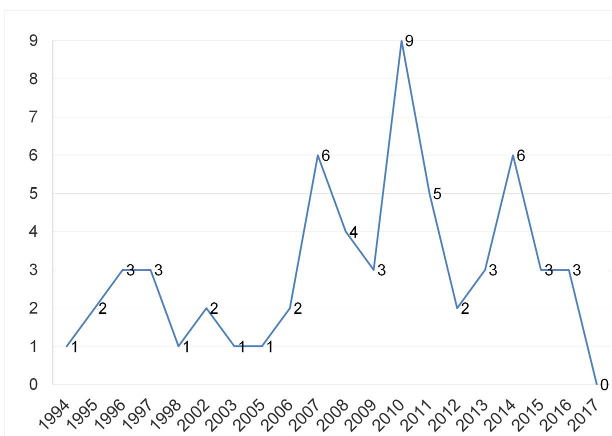

Fig. 2. Temporal view of the studies.

[5] Available at: www.cin.ufpe.br/~jffv/papers/wer2019

Figure 3 presents a bubble plot distributed over three dimensions regarding three characteristics of the studies: evaluation method, research type and application context (academic, industrial or both). The left part in this figure denotes the relationship between the research type of the studies and their evaluation method. The number in a bubble represents the number of studies that present both characteristics. On the other hand, in the right part of this figure, the number in a bubble represents the number on a specific research type in a certain application context.

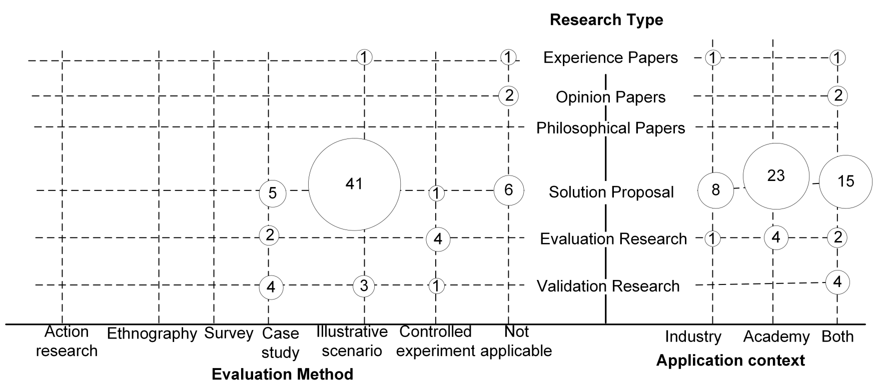

Fig. 3. Bubble plot with application context, research type and evaluation method dimensions.

The results of each research question are presented and discussed in the next sections.

4.2 RQ1: What challenges have been identified pertaining to the communication among engineers during the RE process when developing SCS?

The selected studies point out many challenges as listed in Section 2 supplement material. In this section, we also discuss the details the elicited challenges.

Many challenges of requirements communication are related to the concept of shared understanding [3]. Shared understanding among a group of people has two facets: explicit shared understanding is about interpreting explicit specifications, such as requirements, design documents, and manuals, in the same way by all group members. Implicit shared understanding denotes the common understanding of non-specified knowledge, assumptions, opinions, and values. The shared context provided by implicit shared understanding reduces the need for explicit communication and, at the same time, lowers the risk of misunderstandings.

4.3 RQ2: Which approaches have been proposed to improve the communication in the RE process among engineers when developing safety-critical systems?

This research question was divided into eight sub research questions (RQ2.1 to RQ2.8) aiming to analyze many aspects of requirements communication of SCS.

4.4 RQ2.1: What are the types of these approaches?

The contribution types are reported considering the classification presented in the selected studies. The final list of contribution types are presented in Figure 4.

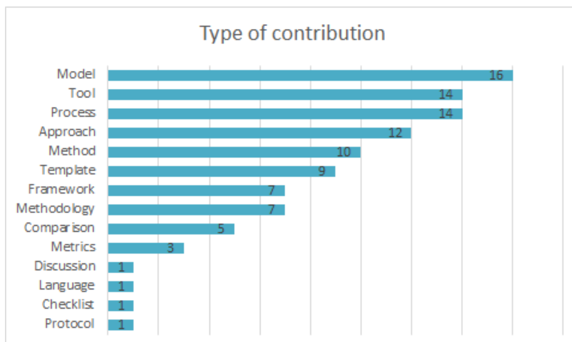

Fig. 4. Type of contributions on requirements communication of SCS.

Note that, similarly to other research questions, this question also allows a study to be included in more than one category. The different types of contributions may be an indication that not all artifacts types are equally suited for all activities in software and RE. Moreover, several persons with various roles and different requests use artifacts based on their individual work throughout the project.

4.5 RQ2.2: For which domains were these approaches proposed?

Figure 5 shows the distribution of the studies by application domain. 78.3% of the studies were classified as domain-independent, the remainder of the studies were developed in the following application domains: robotics, automotive, avionics, medical, railway, and mechatronics.

Analyzing the publication year of the 13 domain-specific approaches, we noticed that 69.23% are recent contributions (published in 2010 or after). This may suggest that the model-driven architecture and the domain specific languages as well as the frameworks for model-driven development might be influencing the approaches for considering domain-specific concerns.

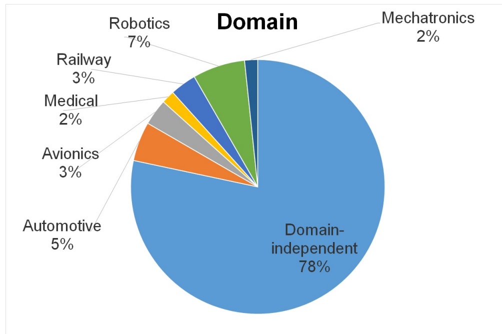

Fig. 5. Application Domain.

4.6 RQ2.3: What RE activities were supported by these approaches?

We categorized these activities according to the main steps of a RE process: elicitation, analysis and negotiation, specification, validation and management[6] (see Table 2). In summary, the results suggest that all RE activities are covered by the studies.

Table 2. RE activities supported by the approaches in requirements communication.

Activity	Count	%
Elicitation	13	21.67%
Analysis and Negotiation	31	51.67%
Specification	38	63.33%
Validation	28	46.67%
Management	12	20%

The *Specification* activity is addressed by more than 60% of the studies. In fact, to some extent, this result was expected, since SCS are submitted to certification processes and many of them must have to be compliant with some safety standard.

Among all 60 studies, only five papers address all activities of RE process and only one study proposed an entire RE process. This may be an indication that a holistic approach to improving the requirements communication that supports all activities of the RE process is needed. This may be one of the reasons for so many problems and challenges faced (see RQ1) in the development of SCS.

4.7 RQ2.4: Which requirements specification languages are used by these approaches?

The languages used by the studies to specify the requirements are listed in Table 3. Furthermore, five papers did not cite any language. We identified a

[6] In this paper, we considered requirements management as a requirements activity.

great variety of requirements specification languages adopted by the approaches. In the supplement material (Section 3.2), we discuss the pros and cons of the identified languages.

Table 3. Requirements specification languages per domain.

RE language	Count
Natural Language	27
Use Cases Description	20
UML	16
Block diagram design language, and State machine design language, SysML	8 (each)
Context Diagram	6
Logics, and Formal methods	5 (each)
Mathematical notations	3
Problem Frames, and Event Time Diagram (ETD)	2(each)
KAOS, RSML language, SpecTRM- RL modeling language, System Diagrams, HIVE requirements language, Goal Model of ATRIUM, Alloy, VDM++, Structured Analysis and Design Technique (SADT), Event-B, EAST-ADL, Bayesian Belief Networks (BBNs), ALTARICA, AUTOSAR, User Requirements Notation (URN), Requirements Definition and Analysis Language (RDAL), and Architecture Analysis and Design Language (AADL)	1 (each)

4.8 RQ2.5: Which tools are used for the requirements specification?

We believe that requirements communication is also improved by the use of shared tools, hence this question maps the ones used to develop the requirements specification of SCS. Table 4 lists the tools mentioned more than once in the selected studies.

Table 4. Tools used in the requirements specification.

Tool	Count	%
It does not cite	36	60%
A proposed one	10	16.67%
Sparx Systems Enterprise Architect	6	10%
ARTi-SAN Studio, DOORS, SystemWeaver, mCRL2, Rodin platform	2 (each)	3.33%
IBM Rational Software Architect, IBM Rational Rhapsody, IBM Rational Harmony for Embedded Real-Time Development tool, HIVE (Hierarchical Verification Environment) tool, Siemens Teamcenter Systems Engineering and Requirements Management, Elektra, Spreadsheet tool, Visio, SafeSlice, EATOP, Artop, Supremica, TCT, NBC, UPPAAL, UML4PF, Papyrus UML, ERRSYS, SRSV, OSATE, and jUCMNav	1 (each)	1.67%

The majority of the studies (36 studies - 60%) did not mention any tool for requirements specification. This lack of tools is a substantial issue since they can contribute to the requirements communication and should consider safety concerns to improve shared understanding.

Other works (ten studies - 16.67%) report that they developed a tool to support their proposals but they did not present their names. The results might indicate that the tools are not adapted for SCS or to enable communication in large teams. Most tools are expensive per license and this forces companies to buy few licenses, limiting access to the central repository and thus hindering communication. Perhaps, the use of no tool (or using internal ones like excel) is a reaction towards the expensive licenses.

4.9 RQ2.6: For which stakeholder were they proposed?

The stakeholders mentioned in the selected studies are listed in Table 5. The majority of the approaches were designed to be used by safety engineers and developers.

Table 5. Stakeholders involved in the approaches.

Stakeholder	Count	%
Safety Engineer	29	48.33%
Developer	23	38.33%
Software Engineer	19	31.67%
Requirements Engineer	18	30%
Design Engineer	10	16.67%
Architect	9	15%
Customer	7	11.67%
System Engineer	6	10%
Certification authorities and Project manager	3 (each)	5%
Human factors expert, Manufacturing (MAN), and Product development (PD)	2 (each)	3.33%
Supplier, Test engineer, Quality Manager, Cognitive engineer, Operator, Constraints Engineer, Domain Engineer, and Reliability Engineer	1 (each)	1.67%

The results presented in Table 5 suggest that, as expected, *safety engineers* are the stakeholders for which most studies have been proposed. The next most cited stakeholders in the selected studies were *Developer*, *Software Engineer*, and *Requirements Engineer*. This outcome might indicate that there is some confusion in the selected studies, perhaps not in the industry, of their roles and the division of attributions is not clearly defined. Moreover, there is a tendency of sharing the responsibility of safety analysis conduction by all these stakeholders mentioned above.

4.10 RQ2.7: What are the communication formats used?

We based our analysis on the work of Jim Whitehead [13] that classifies the collaboration tools as *Model-based, artifacts-based, Process support, Awareness*, and *Collaboration infrastructure* in a roadmap about collaboration in software engineering. We complemented such classification with *Analysis tools*, and *Face-to-face verbal communication* categories according to the formats presented in the selected studies. Table 6 lists the communication formats used in the approaches.

The *model-based collaboration* was used by 70% of the selected studies (42 studies). Hence, there is a tendency, in the selected studies, of using models

Table 6. Communication format used in the approaches.

Communication Format	Count	%
Model-based collaboration	42	70%
Process support	26	43.33%
Artifacts-based	21	35%
Analysis tools	19	31.67%
Face-to-face verbal communication	4	6.67%
Collaboration infrastructure	3	5%
Awareness	2	3.33%

to improve the requirements communication in SCS. Model-based specifications are consistent and less ambiguous than informal specification documents, forcing the stakeholders to make clear all aspects of the system early in the design process. Therefore, models provide a shared meaning that engineers use when coordinating their work, as when stakeholders consult a requirements specification to determine how to design a portion of the system or to perform the safety analysis.

4.11 RQ2.8: For what safety standards have the approaches been proposed?

The safety standards presented in the approaches are exhibited in Table 7. Table 7 shows that the great majority of the approaches of requirements communication (46 studies - 76.67%) *does do not follow* any safety standard.

Table 7. Safety standards adopted.

Safety Standard	Year	Domain	Count	%
No			46	76.67%
IEC 61508	2010	Generic	4	6.67%
ISO 26262	2011	Automotive	3	5%
DO-178B	1992	Avionics	1	1.67%
ISO/IEC 15504	2003	Generic	1	1.67%
ISO 12207	1995	Generic	1	1.67%
ISO 12100	2010	Machinery	1	1.67%
IEC/SC65A	1992	Generic	1	1.67%
Australian Defence Standard Def (Aust) 5679	1998	Generic	1	1.67%
ANSI/RIA R15.06-1999	1999	Robotics	1	1.67%
ISO/IEC 9126	2001	Generic	1	1.67%
IEC 1508	1995	Generic	1	1.67%
IEC 61499	2011	Generic	1	1.67%
IEC 61131	1993	Generic	1	1.67%
EIA-632	1994	Generic	1	1.67%

From the approaches which based their concepts in part on the definitions given by the international standards for safety, we identified fourteen standards (see Table 7). The date of the safety standard release varies between 1992 and 2011. 64.29% of the followed safety standards are developed for general purposes such as defining the safety life cycle, the requirements for evaluation of the software development process, the terminology and guidelines.

5 Conclusions

The RE activities are critical to avoid the introduction of defects and misunderstandings among engineers and developers when developing SCS. Communication among workgroups that develop interdependent pieces of a system is crucial for a successful outcome of software development projects [8]. This is an important question in the development of SCS considering that many safety problems occur due to errors and misunderstandings in safety requirements specifications.

Our mapping study draws on 60 studies, selected out of 1164, through a multi-stage process. An important feature of the review is that it does not restrict itself to a particular domain or safety standard. This broad scope in the search gives us deeper insights into the state-of-the-art about how the requirements communication is conducted in the RE process. Currently, we are working on the analysis of safety standards and the comparison of the results of the state-of-art with the state-of-practice is in progress.

5.1 Threats to validity

We adopted the classification of threats to validity well adopted in the literature which corresponds to Internal, External, Construct and Conclusion categories. *Construct validity*: For all concepts, we used many synonyms to ensure high coverage of potentially-relevant studies from a database search. *Internal validity*: In order to minimize selection and extraction mistakes, the selection process was performed in an iterative way. It is also worth noting that the all authors are lecturers and experienced researchers with expertise in RE, Software Engineering or SCS. *External validity*: In order to mitigate external threats, the search process was defined after several trial searches and validated with the consensus of the authors. *Conclusion validity*: The research protocol was carefully designed and discussed by the authors to minimize the risk of exclusion of relevant studies. It is worth highlighting that we did not restrict the time period of published studies to obtain the maximum coverage possible.

The results of this mapping study showed that although there are some approaches to improve the requirements communication of SCS, several problems still remain since many studies do not support the real needs of the industry. Therefore, this mapping study has generated several promising research directions:

(1) How safety analysis techniques can be improved to evaluate shared understanding (**RQ2.1**)?
(2) To what extent do the domain-independent approaches cover the needs of domain-specific critical systems (**RQ2.2**)?
(3) Why the approaches do not cover the entire RE process? (**RQ2.3**)
(4) To what extent do the tools used in the requirements specification are capable of improving requirements communication (**RQ2.5**)?
(5) Which is the most effective communication format in requirements communication of safety-critical systems (**RQ2.7**)?
(6) Why do the approaches not follow the guidelines of safety standards (**RQ2.8**)?

6 ACKNOWLEDGMENTS

We would like to acknowledge that this work was partially supported by Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) and Fundação de Amparo à Ciência e Tecnologia do Estado de Pernambuco (FACEPE) from Brazil, and by KKS foundation through the S.E.R.T. Research Profile project at the Software Engineering Research Lab, Blekinge Institute of Technology, Sweden.

References

1. Bjarnason, E., Wnuk, K., Regnell, B.: Requirements are slipping through the gaps? a case study on causes & effects of communication gaps in large-scale software development. In: International Requirements Engineering Conference (RE). pp. 37–46. IEEE (2011)
2. Brady, A., Seigel, M., Vosecky, T., Wallace, C.: Addressing communication issues in software development: A case study approach. In: Software Engineering Education & Training, 2007. CSEET'07. 20th Conference on. pp. 301–308. IEEE (2007)
3. Glinz, M., Fricker, S.A.: On shared understanding in software engineering: an essay. Computer Science-Research and Development **30**(3-4), 363–376 (2015)
4. Hatcliff, J., Wassyng, A., Kelly, T., Comar, C., Jones, P.: Certifiably safe software-dependent systems: challenges and directions. In: Proceedings of the on Future of Software Engineering. pp. 182–200. ACM (2014)
5. Kitchenham, B., Charters, S.: Guidelines for performing systematic literature reviews in software engineering. Tech. Rep. EBSE 2007-001, Keele University and Durham University Joint Report (2007)
6. Leveson, N.: Engineering a safer world: Systems thinking applied to safety. Mit Press (2011)
7. Nakamura, H., Umeki, H., Kato, T.: Importance of communication and knowledge of disasters in community-based disaster-prevention meetings. Safety Science (2016)
8. Pernstal, J.: Towards Managing the Interaction between Manufacturing and Development Organizations in Automotive Software Development. Ph.D. thesis, Department of Computer Science and Engineering, CHALMERS UNIVERSITY OF TECHNOLOGY (2015)
9. Rasmussen, J., Lundell, Å.K.: Understanding "communication gaps" among personnel in high-risk workplaces from a dialogical perspective. Safety science **50**(1), 39–47 (2012)
10. Sikora, E., Tenbergen, B., Pohl, K.: Industry needs and research directions in requirements engineering for embedded systems. Requirements Engineering **17**(1), 57–78 (2012)
11. Vilela, J., Castro, J., Martins, L.E.G., Gorschek, T.: Integration between requirements engineering and safety analysis: A systematic literature review. Journal of Systems and Software **125**, 68–92 (2017)
12. Wang, Y., Graziotin, D., Kriso, S., Wagner, S.: Communication channels in safety analysis: An industrial exploratory case study. arXiv preprint arXiv:1804.08909 (2018)
13. Whitehead, J.: Collaboration in software engineering: A roadmap. FOSE **7**(2007), 214–225 (2007)