AtoZ
novas práticas em informação e conhecimento

ISSN: 2237-826X
PAPER | ARTIGO | ARTÍCULO

Requirements elicitation techniques for software development: a systematic review of literature

Técnicas de elicitação de requisitos no desenvolvimento de software: uma revisão sistemática da literatura

Naiara Crislaine Alflen [1], Edmir Parada Vasques Prado[2]

[1] *Universidade de São Paulo (USP), São Paulo, SP, Brasil. ORCID: https://orcid.org/0000-0001-9666-1417*

[2] *Universidade de São Paulo (USP), São Paulo, SP, Brasil. ORCID: http://orcid.org/0000-0002-3505-6122*

Mail to/*Autor para correspondência/Correo a:* Naiara Crislaine Alflen, naiara.alflen@usp.br

Submitted/*Recibido*: 19 de outubro de 2020; **Approved/*Aceptado*:** 22 de novembro de 2020

Abstract

Introduction: Requirements elicitation (RE) is a difficult task in which there are issues related to information ambiguity, incomplete and inconsistent data. It seeks to discover and understand the users' problems and needs. The success of software development depends on the correct elicitation of requirements, and its quality is influenced by the techniques used. This research aims to identify the RE techniques most cited in the literature of software development projects within this context. **Method:** a systematic literature review was carried out, which analyzed 61 articles from the Scopus database. **Results:** We identified the 10 RE techniques most cited in the literature. The ranking of these techniques showed that those obtained by stakeholders' groups' involvement were the least mentioned. **Conclusions:** the research identified the opportunity and relevance for developing a descriptive or confirmatory analysis of RE techniques based on the interaction between groups of users and members of the software development team.

Keywords: Requirements elicitation; Information systems; Team participation.

Resumo

Introdução: *A elicitação de requisitos (ER) é uma tarefa difícil na qual é necessário lidar com ambiguidade de informações, dados incompletos e inconsistentes. Ela busca descobrir e entender o real problema e as necessidades dos usuários. O sucesso do desenvolvimento de software depende da elicitação correta dos requisitos, e a qualidade deles é influenciada pelas técnicas usadas. Dentro deste contexto, o objetivo desta pesquisa é identificar as técnicas de ER mais citadas na literatura em projetos de desenvolvimento de software.* ***Método:*** *Foi realizada uma revisão sistemática da literatura, que analisou 61 artigos da base de dados Scopus.* ***Resultados:*** *Foram identificadas as 10 técnicas de ER mais citadas na literatura. A classificação dessas técnicas mostrou que aquelas obtidas pelo envolvimento de grupos de stakeholders foram as menos citadas.* ***Conclusão:*** *A pesquisa identificou a oportunidade e relevância para o desenvolvimento de pesquisa acadêmica descritiva ou confirmatória sobre técnicas de ER baseadas na interação entre grupos de usuários e membros da equipe de desenvolvimento de software.*

Palavras-chave: *Elicitação de requisitos; Sistemas de informação; Participação da equipe.*

INTRODUCTION

Requirements engineering is the most complex phase in software development (Buitrón, Flores-Rios, & Pino, 2018; Fernandes et al., 2012; Kiran & Ali, 2018). According to Alexa and Avasilcai (2018), one of the stages of requirements engineering is requirements elicitation (RE), which is dedicated to the discovery, extraction, and revelation of user needs.

RE is the understanding of the user's real need (Hickey & Davis, 2003). It is a difficult task in which it is necessary to deal with ambiguity of information, incomplete and inconsistent data, where requirements are not clearly known (Vijayan, Raju, & Joseph, 2016). Moreover, RE is not only about writing requirements, but in discovering and understanding the real problem and the users' needs (Araujo, Anjos, & Silva, 2015). Misunderstanding the user's need is one of the main factors in the failure of a project (Gonzales & Leroy, 2011). Most of the time, users have difficulty in expressing their requirements (Nuseibeh & Easterbrook, 2000). According to Mishra, Mishra, and Yazici (2008), the biggest failure in software projects are incomplete and incorrect requirements.

Conventional methodologies for ER focus on gathering all software requirements to document them and subsequently move on to the development phase (Alexa & Avasilcai, 2018; Batool et al., 2013). However, in constantly evolving environments, these methodologies are not suitable approaches, as in today's scenario change is considered an essential feature of software development (Jayatilleke & Lai, 2018). In this context, Knauss, Yussuf, Blincoe, Damian, and Knauss (2018) consider traditional requirements engineering insufficient to achieve adequate ER. Similarly, Batool et al. (2013) and Sheffield and Lemétayer (2013) mention the need for a flexible

and fast software development process, where it is necessary to work with short delivery cycles to deal with requirements changes and uncertainties. As a consequence, agile methodologies with fast and incremental deliveries support requirement changes and the elicitation process occurs in the course of software development (Asghar, Tabassum, Bhatti, & Jadi, 2017).

The quality of requirements is influenced by the techniques used in RE. This is, according to Hickey and Davis (2003), because RE is a time of learning and communication with users. However, there are RE techniques in which user participation is not addressed, and there are other techniques in which user participation is fundamental. Regardless of the greater or lesser participation of the user, Babar, Bunker, and Gill (2018) argue that the success of software development depends on the correct elicitation of requirements. Within this context, the objective of this research is to identify the RE techniques most often cited in the literature in software development projects.

RESEARCH METHOD

The research proposed in this work is characterized as an exploratory study. According to Cervo and Bervian (2007), exploratory research aims to provide information about the object of study. Thus, this research explores scientific databases to identify RE techniques in software development projects. This study is a qualitative and cross-sectional research of the single type, because data collection was done only once. The stages of the research and the methodological procedures for data collection and analysis are presented below.

Research Steps

The research steps are schematized in Figure 1 and served as the basis for conducting this research. Initially, from a systematic literature review (SLR), the techniques used for ER in software development projects were identified and listed. Then, based on the selected articles, the techniques were analyzed, identifying their characteristics and frequency of use in the organizational environmen

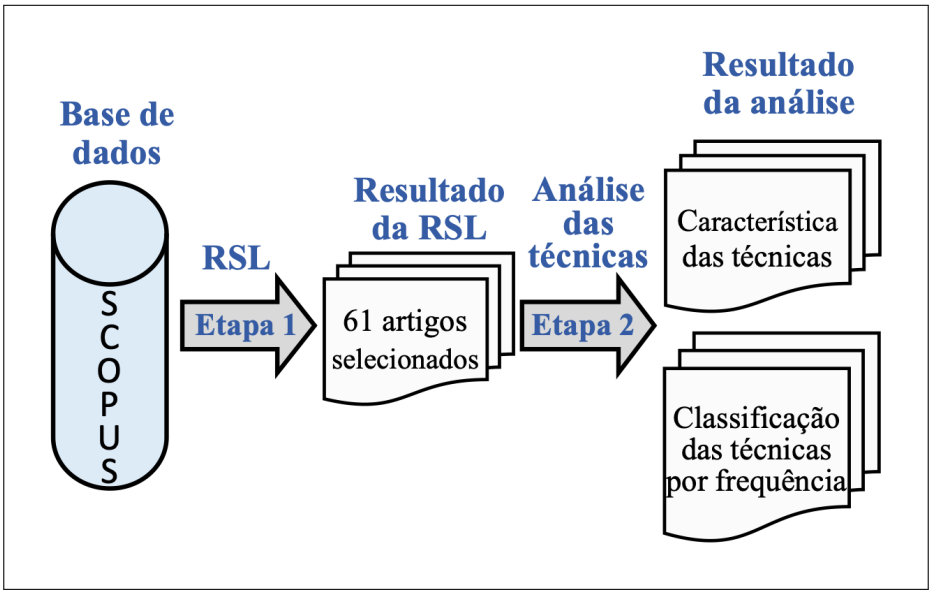

Figure 1. Research steps.

Data Collection and Analysis

The identification of ER techniques was performed by means of an RSL. RSL is a method to identify and analyze papers available in scientific databases and answer research questions (Baptista & de Campos, 2007). Kitchenham (2004) mentions that RSL uses a rigorous, reliable, and auditable method. To perform the RSL, the protocol proposed by Kitchenham and Charters (2007) was used. This protocol establishes research strategies for structuring the work and for identifying and evaluating the materials found. The protocol was carried out in three phases: planning, selection and results.

Planning

The planning of the RSL consists of two items: defining the objective and defining the research protocol. The objective of this RSL is to identify RE techniques, and the protocol definition was carried out in three steps presented below:

a) **Research question.** RE is a topic addressed in different areas of knowledge and in this research the area of software development was specified. Thus, the question defined for this RSL was: What are the RE techniques used in software development projects?

b) **Identification of the studies.** A broad database search was conducted for studies answering the proposed research question. This RSL used the Scopus database (www.scopus.com), because this database indexes the works in the ACM Digital Library and IEEEXplore libraries. The definition of key words considered the terms "requirements elicitation," "techniques," and "software," in the English language and with their respective synonyms, as presented in Table **??**.

c) **Study selection criteria.** Inclusion, exclusion, and quality criteria were used to select the articles. The articles were included when they met the three inclusion criteria, but eliminated when they met one of the exclusion criteria. The criteria defined were:

- Inclusion criteria: (I1) articles that present techniques for RE; (I2) articles that present techniques for RE in software development; and (I3) articles published from 2009 to 2020.
- Exclusion criteria: (E1) article is not presented entirely in English; (E2) article is not related to the Computer Science, Information Systems or Engineering areas; (E3) article is not from a peer-reviewed journal or conference; (E4) article is workshop, lecture notes, work in progress or short article; (E5) article is not available electronically or has access restrictions; and (E6) article is not a primary study.
- Quality criterion: (C1) based on the average citation score of journals in the Scopus database.

Keyword	**Synonyms**
Requirements elicitation	*requirements gathering*
Techniques	*methods* **OR** *procedures* **OR** *tools* **OR** *artifacts* **OR** *specification*
Software	*system* **OR** *systems*
Search term	
TITLE-ABS-KEY ((“Requirements elicitation” OR “requirements gathering”) AND (Techniques OR methods OR procedures OR tools OR artifacts or specification) AND (software OR system OR systems))	

Quadro 1. Search strategy applied to the database

The articles belonging to the quartile with the highest score were selected.

Selection of Articles

The base search was conducted in the month of May 2020. The application of the search term in the Scopus database resulted in 1,972 articles. From these articles, inclusion, exclusion, and quality criteria were applied: inclusion criteria I1, I2, and I3 resulted in 972 articles; exclusion criteria E1, E2, E3, E4, E5, and E6 resulted in 197 articles; and quality criterion C1 resulted in 61 articles.

PRESENTATION AND ANALYSIS OF RESULTS

The RE techniques found in the literature, through RSL, are from works related to the software development process. However, these same techniques are also found in works from other fields of knowledge. In most of the papers, more than one technique was found in the RE process. However, several nomenclatures were found for similar techniques, such as the terms “questionary” and “survey”.

The similar techniques were grouped and 40 techniques for RE were identified. In some papers, the techniques are described and in others just cited. Table 1 shows the techniques in descending order of citation frequency and the articles in which they were found. Seven techniques had only one citation and are not included in the table.

Technique	Related articles in Appendix A
Interviews	[02] [04] [05] [06] [07] [09] [10] [11] [12] [14] [15] [17] [18] [20] [21] [22] [24] [25] [28] [29] [30] [31] [32] [33] [34] [36] [38] [40] [41] [43] [44] [45] [46] [47] [48] [52] [53] [54] [55] [56] [57] [58] [59] [60] [61]
Questionnaire	[03] [04] [05] [07] [08] [09] [10] [13] [14] [15] [18] [20] [22] [24][26] [28] [29] [30] [34] [35] [36] [37] [41] [43] [44] [45] [46] [47] [48] [50] [55] [56] [58] [59]
Prototype	[02] [03] [05] [07][08] [09] [10] [12] [13] [15] [17] [21] [22] [27] [31] [33] [34] [39] [41] [43] [47] [48] [53] [54] [59]
Use cases	[01] [05] [07] [09] [10] [14] [15] [16] [19] [22] [26] [32] [35] [36] [38] [42] [43] [45] [47] [49] [50] [56] [57] [60] [61]
Brainstorming	[05] [07] [14] [15] [16] [17] [27] [28] [31] [33] [34] [36] [37] [39] [40] [41] [44] [45] [46] [47] [54] [55] [58] [59]
Scenario	[01] [02] [07] [08] [11] [14] [16] [18] [21] [22] [27] [28] [33] [34] [35] [38] [41] [47] [49] [50] [51] [54] [61]
Feedback	[01] [03] [04] [07] [08] [11] [12] [21] [24] [25] [27] [33] [39] [40] [44] [47] [50] [52] [55]
Workshop	[04] [05] [11] [12] [14] [15] [16] [28] [32] [34] [38] [41] [44] [46] [48] [50] [54]
Focus Group	[03] [07] [14] [15] [17] [21] [22] [24] [31] [38] [39] [41] [46] [48]
User Story	[07] [11] [14] [17] [19] [22] [23] [30] [34] [48] [49] [54]
Observation	[02] [05] [07] [11] [17] [21] [22] [27] [30] [31] [38]
Ethnography	[04] [22] [23] [24] [38] [40] [42] [43] [45] [61]
JAD / RAD	[04] [07] [28] [38] [41] [42] [44] [50] [52] [55]
Storyboards	[15] [21] [23] [36] [38] [39] [45]
Document Analysis	[14] [15] [25] [38] [41] [52] [60]
Natural language	[11] [12] [15] [17] [20] [31]
Storytelling	[08] [22] [23] [39] [42] [45]
Goal Based	[01] [29] [34] [38] [51]
Personas	[08] [24] [41] [53] [60]
Crowdsourcing	[12] [16] [19] [20] [60]
Quality Function Deployment	[29] [34] [57] [60]
Domain Analysis	[34] [49] [57] [60]
Ontologia	[18] [38] [57] [60]
Issue-based information systems	[45] [55] [57] [60]
Win-Win	[44] [54] [55]
Repertory grids	[18] [25] [41]
Controlled requirements expression	[57] [60]
Critical discourse analysis	[57] [60]
Collaborative Tools	[49] [59]
Introspection	[15] [34]
Appreciative Inquiry	[31] [47]
User Experience	[08] [22]
Gamification	[39] [42]

Table 1. RSL selected articles

The analysis of the RE techniques is presented in two topics. In the first, the characteristics and frequency of use of the techniques are presented. In the second, the techniques are described and analyzed, that is, the techniques are classified in relation to their sources of obtaining the requirements and the benefits and limitations of the most cited techniques in the literature are pointed out.

Characteristics and Frequency of Technique Use

The quartile statistic was applied to the frequency of the RE techniques found in the literature. The first quartile was considered, for the purpose of analysis in this research, as containing the most cited RE techniques in the literature, i.e., the set containing 25% of the most cited techniques. This set is composed of the 10 techniques, which are presented below:

a) **Interview**. It was the most mentioned technique in the articles. It was present in 45 of the 61 selected articles. The interview usually involves a representative of the project team with a stakeholder and is known to be the most common of the ER techniques.

b) **Questionnaire**. It was the second most cited technique, found in 34 out of 61 articles. From Abd-Elmonem, Nasr, and Gheith (2017) point of view, the questionnaire is a traditional and commonly used ER technique. It is a simple tool, usually applied in the early phase of ER to collect as many requirements as possible from different stakeholders who may be in different places.

c) **Prototyping**. It is an incomplete or early version of the software. That is, it can be disposable or evolutionary (Younas, Jawawi, Ghani, & Kazmi, 2017). It collects feedback from stakeholders and identifies changes that should be incorporated in the next version.

d) **Use Case**. It represents a possible use of the system by an actor using some service. The use case narrates the interaction between the system and the actors involved. For Hajri, Goknil, Briand, and Stephany (2018) it is the main technique employed to extract requirements and communicate with customers.

e) **Brainstorming**. It is a meeting in which each participant can freely express the system requirements. For Younas et al. (2017), it is a way to attune the user's mind to the requirements.

f) **Scenarios**. They represent the users' interactions with the system. They allow descriptions of the current and future process, necessary for the development of the software project. According to Adem and Kasirun (2010) they help to discover the goals of the software and contemplate the interaction with the user.

g) **Feedback**. According to Hosseini et al. (2015), this technique provides participants with the feeling that their ideas matter and can lead to convergence of opinions.

h) **Workshop**. It is a collaborative technique to define the requirements of a software and can be used to clarify ambiguities (Angelis, Ferrari, Gnesi, & Polini, 2018; Mishra, Aydin, Mishra, & Ostrovska, 2018). It is a kind of seminar or discussion group in which a speaker presents a specific content and encourages reflection in the group.

i) **Focus Group**. It is an objective discussion that introduces a topic to a group of participants and directs their discussion about the topic, in an unstructured way. For Younas et al. (2017) it is a way to learn about the user's wants and perceptions of the software, as well as the definition of requirements.

j) **User Stories**. These are brief descriptions of software functionality (Mobasher & Cleland-Huang, 2011; Younas et al., 2017) and are discussed during all phases of the project to clarify requirements (Knauss et al., 2018; Younas et al., 2017).

Technique Description and Analysis

Classification Regarding the Source from which the Requirements are Obtained Batista (2003) classified the RE techniques into five categories according to the sources of obtaining the requirements. The RE techniques identified in this research were classified based on these categories and are presented in Table 2.

Source of requirements	Techniques	RSL citation order	Average Rating
Individual	Questionnaire	2	4,5
	Feedback	7	
Group	Brainstorming	5	7,3
	Workshop	8	
	Focus Group	9	
Mixed	Interview	1	4,8
	Prototyping	3	
	Use Cases	4	
	Scenarios	6	
	User stories	10	

Table 2. Most cited RE techniques and ranked by source of requirements

None of the ten most cited techniques in the RSL has documentation or observation as a source of requirements. They were classified into only three sources: individual, group, and mixed. This indicates that the most cited techniques are based on interaction between user and developer. On the other hand, the techniques based on individual or mixed interactions are more cited than those based exclusively on groups, i.e., Brainstorming, Workshop and Focus Group had an average rating of 7.3, lower than the other types of requirements sources. On the other hand, there is an increase in the use of RE techniques that involve more the user and seek to provide creativity and innovation to the process, such as Design Thinking (Araujo et al., 2015).

Benefits

The benefits obtained by applying these ten techniques address the main issues surrounding RE. The questionnaire technique meets the low cost characteristic. Similarly, according to Ramakrishnan et al. (2014) the scenario technique reduces the time spent on development and therefore the cost. The questionnaire, besides having low cost, allows for massive data collection. Al-Qudah, Cristea, and Lei (2013) also highlight this characteristic for the Brainstorming technique, in which the collected data is later refined.

Another characteristic observed is creativity. Caleb-Solly, Dogramadzi, Ellender, Fear, and Heuvel (2014) point out that Brainstorming stimulates creativity. Similarly, Smith, Strauss, and Maher (2010) claim that the interview gathers information for new projects and allows to seek new ideas, contributing to creativity. This can also be achieved by using the focus group technique. According to Alvertis, Papaspyros, Koussouris, Mouzakitis, and Askounis (2016) with the application of this technique it is possible to collect diverse opinions.

Security is also an issue addressed by these techniques. The Use Case technique can predict malicious user behavior and software misuse (Odusote, Daramola, & Adigun, 2018). Furthermore, Odusote et al. (2018) applied the Misuse Case and noted that it provides an intuitive approach to eliciting requirements that meet the reliability criterion. The user story technique, according to Ramesh and Reddy (2016), because a specific case, which are the abuse stories, and which describe stories in the context of the software intruder, contributing to addressing security issues.

It is also worth noting that these techniques facilitate user engagement, clarify problems, and improve requirements definition. For Ramakrishnan et al. (2014) and Hidalga, Hardisty, and Jones (2016), prototyping facilitates user engagement and early problem identification. Workshop and Feedback also help in problem identification, as the former clarifies ambiguities (Ramakrishnan et al., 2014) and the latter allows the end user to communicate problems, needs, and options when using a software product (Oriol et al., 2018). Finally, the Scenarios technique helps to reduce misunderstood requirements.

Limitations

Much of the limitations of these techniques are associated with the person applying it and the user. According to Gill, Zaidi, and Kiani (2014), the successful outcome of an interview depends on the interviewer's ability to conduct requirements gathering. In addition, techniques whose requirements source is obtained through group of people also have these limitations: (1) for Mishra et al. (2018), the user experience should be considered for the correct use of the brainstorming technique, because users used to the software development context can better express their needs; (2) Fernandes et al. (2012) warn that, because it is a collaborative technique, in the workshop dominant and biased participants may inhibit the opinions of other participants; (3) for Pitula and Radhakrishnan (2011), in the application of the Focus Group there is the disadvantage of participants feeling uncomfortable when stating opinions different from those raised by the group, in addition to dominant and biased participants, who cause valid ideas from other participants to go unexplored (Fernandes et al., 2012).

Another common limitation of these techniques is associated with the issue of time and resources. The first technique that presents this disadvantage is prototyping. Gill et al. (2014) point out that despite the advantages of prototyping, it has the disadvantage of being time and resource consuming. The scenario technique presents a similar problem, as it can require several meetings and doing demonstrations of concepts with scenarios can take considerable time.

Finally, it should be noted that some techniques have limitations when used for certain purposes. Thomas, Bandara, Price, and Nuseibeh (2014) argue that questionnaires do not provide sufficiently rich information about user decisions and how they are influenced by the emerging context in a particular situation. Moreover, their application has presented difficulties in the RE process. Other techniques are not suitable for certain types of requirements: Sadiq, Ghafir, and Shahid (2009) state that Brainstorming is not suitable for safety RE, i.e., it does not result in a consistent set of safety requirements; and the scenario technique is not suitable for gathering the non-functional requirements of a system.

FINAL CONSIDERATIONS

The objective of this research was to identify the ER techniques most often cited in the literature in software development projects. To achieve this goal, a qualitative and exploratory research was carried out based on an RSL. The conclusions of this work are presented below.

Techniques achieved through group interaction were cited less than the others, despite the increasing emphasis on user involvement. This shows that there is opportunity and relevance for academic research that seeks to describe and analyze RE techniques based on interaction between groups of users and software development team members. Activity Theory (Wertsch, 1981) can assist in understanding how this social relationship influences RE activity.

This research is part of a project to study RE techniques. From the results of this research, it is intended to elaborate a comparison with the most commonly used RE techniques by public and private companies in software development. And then elaborate propositions for confirmatory research about the most appropriate use of RE techniques in specific contexts.

APPENDIX A

Table 3 presents the RSL articles in descending order of year of publication.

(continua)

#	Authors	Titles	Year
01	Ferraris, D., Fernandez GC.	TrUStAPIS: a trust requirements elicitation method for IoT	2020
02	Martinez A.	An Experience Report on the Use of Experience Maps and Sketches in a Database Course Project	2019
03	Rizk N.M. et al.	Crowdsourcing based requirements elicitation for eLearning Systems	2019
04	Alwadain A., Alshargi M.	Crowd-generated data mining for continuous requirements elicitation	2019
05	Martins H.F., et al.	Design thinking: Challenges for software requirements elicitation	2019
06	Garcia I. et al.	Experiences of using a game for improving learning in software requirements elicitation	2019
07	Fatima R. et al.	Improving software requirements reasoning by novices: A story-based approach	2019
08	Li C. et al.	Automatically classifying user requests in crowdsourcing requirements engineering	2018
09	Elmonem M.A. et al.	Automating requirements elicitation of cloud based ERPs	2018
10	Hajri I. et al.	Configuring use case models in product families	2018
11	Knauss E. et al.	Continuous clarification and emergent requirements flow in open-commercial software ecosystems	2018
12	Oriol M. et al.	FAME: Supporting continuous requirements elicitation by combining user feedback and monitoring	2018
13	Ohashi K. et al.	Focusing requirements elicitation by using a UX measurement method	2018
14	Babar A. et al.	Investigating the relationship between business analysts' competency and IS requirements elicitation: A thematic-analysis approach	2018
15	Mishra D. et al.	Knowledge management in requirement elicitation	2018
16	Odusote B. et al.	Towards an extended misuse case framework for elicitation of cloud dependability requirements	2018
17	Younas, M. et al.	Non-Functional Requirements Elicitation Guideline for Agile Methods	2017
18	Dey S., Lee S.-W.	REASSURE: Requirements elicitation for adaptive socio-technical systems using repertory grid	2017
19	Raja R.M.R., Satyananda R.C.	A survey on security requirement elicitation methods: Classification, merits, and demerits	2016
20	Elsaid A.H. et al.	Automatic framework for requirement analysis phase	2016
21	de la Hidalga A.N. et al.	SCRAM–CK: applying a collaborative requirement engineering process for designing a web-based e-science toolkit	2016
22	Alvertis I. et al.	Using crowdsourced and anonymized Personas in the requirements elicitation and software development phases of software engineering	2016
23	Maiti R.R., Mitropoulos F.J.	Capturing, eliciting, predicting, and prioritizing (CEPP) non-functional requirements metadata during the early stages of agile software development	2015
24	Hosseini M. et al.	Configuring crowdsourcing for requirements elicitation	2015
25	Dey S., Lee S.-W.	From requirements elicitation to variability analysis using repertory grid: A cognitive approach	2015
26	Faßbender S. et al.	Problem-based security requirements elicitation and refinement with pressure	2015
27	Ellis L. et al.	Thinking out loud and e-health for coordinated care – Lessons from user requirements gathering in the 4C project	2015

(continua)

#	Authors	Titles	Year
28	Caleb-Solly P. et al.	A mixed-method approach to evoke creative and holistic thinking about robots in a home environment	2014
29	Sadiq M., Jain S.K.	Applying fuzzy preference relation for requirements prioritization in goal-oriented requirements elicitation process	2014
30	Thomas K. et al.	Distilling privacy requirements for mobile applications	2014
31	Gill K.D. et al.	Eliciting futuristic end-user requirements through contributory appreciative inquiry	2014
32	Wohlrab R. et al.	Experience of pragmatically combining RE methods for performance requirements in industry	2014
33	Ramakrishnan L. et al.	Experiences with user-centered design for the tiger's workflow API	2014
34	Kassab M. et al.	State of practice in requirements engineering: Contemporary data	2014
35	Shirogane J.	Support Method to Elicit Accessibility Requirements	2014
36	Katina P.F. et al.	System requirements engineering in complex situations	2014
37	Al-Qudah D.A. et al.	An exploratory study to design an adaptive hypermedia system for online advertisement	2013
38	Al Balushi T.H. et al.	Eliciting and prioritizing quality requirements supported by ontologies: A case study using the ElicitO framework and tool	2013
39	Rice M., Carmichael A.	Factors facilitating or impeding older adults' creative contributions in the collaborative design of a novel DTV-based application	2013
40	Zhang W. et al.	Feature-oriented Stigmergy-based collaborative requirements modeling: An exploratory approach for requirements elicitation and evolution based on web-enabled collective intelligence	2013
41	Todoran I. et al.	How cloud providers elicit consumer requirements: An exploratory study of nineteen companies	2013
42	Daramola O. et al.	A comparative review of i*-based and use case-based security modelling initiatives	2012
43	Dias, A.L. et al.	Increasing the quality of web systems: By inserting requirements of accessibility and usability	2012
44	Fernandes J. et al.	I Think: A game-based approach towards improving collaboration and participation in requirement elicitation	2012
45	Salini P., Kanmani S.	Security requirements engineering process for web applications	2012
46	Lim S.L., Finkelstein A.	Stakerare: Using social networks and collaborative filtering for large-scale requirements elicitation	2012
47	Gonzales C.K., Leroy G.	Eliciting user requirements using Appreciative inquiry	2011
48	Pitula K.,Radhakrishnan T.	On eliciting requirements from end-users in the ICT4D domain	2011
49	Mobasher, B. et al.	Recommender systems in requirements engineering	2011
50	Fabian B. et al.	A comparison of security requirements engineering methods	2010
51	Adem, N.A.Z., Kasirun Z.M.	Automating function points analysis based on functional and non-functional requirements text	2010
52	Smith, C. et al.	Data structure visualization: The design and implementation of an animation tool	2010
53	Winschiers-Theophilus H. et al.	Determining requirements within an indigenous knowledge system of African rural communities	2010
54	Seyff N. et al.	Using mobile RE tools to give end-users their own voice	2010
55	Castro-Herrera C. et al.	A recommender system for requirements elicitation in large-scale software projects	2009
56	Pu Y., Liu Q.	A viewpoint-oriented requirements elicitation integrated with aspects	2009
57	Sadiq Mohd. et al.	An approach for eliciting software requirements and its prioritization using analytic hierarchy process	2009

(conclusão)

#	Authors	Titles	Year
58	Guerra-García C. et al.	DQ-VORD: A methodology for managing and integrating data quality requirements into software requirement specification	2009
59	Castro-Herrera, C. et al.	Enhancing stakeholder profiles to improve recommendations in online requirements elicitation	2009
60	Romero M. et al.	Teaching requirements elicitation within the context of global software development	2009
61	Motta G. et al.	User goal-oriented requirements elicitation to improve acceptance and use: A case study on document management	2009

REFERENCES

Abd-Elmonem, M. A., Nasr, E. S., & Gheith, M. H. (2017). Automating Requirements Elicitation of Cloud-Based ERPs. In *International Conference on Advanced Intelligent Systems and Informatics* (pp. 171–180). doi: 10.1007/978-3-319-64861-3_16

Adem, N. A. Z., & Kasirun, Z. M. (2010, Feb.). Automating function points analysis based on functional and non functional requirements text. In *The 2nd International Conference on Computer and Automation Engineering (ICCAE)*. IEEE. Retrieved from https://doi.org/10.1109/iccae.2010.5451322 doi: 10.1109/iccae.2010.5451322

Alexa, L., & Avasilcai, S. (2018). The requirement elicitation process of designing a collaborative environment – the CRE@TIVE.BIZ case. *MATEC Web of Conferences*, *184*, 1-4. Retrieved from https://doi.org/10.1051/matecconf/201818404010 doi: 10.1051/matecconf/201818404010

Al-Qudah, D. A., Cristea, A. I., & Lei, S. (2013). An Exploratory Study to Design an Adaptive Hypermedia system for online-advertisement. In *The 9th International Conference on Web Information Systems and Technologies* (p. 368–374). Retrieved from http://rgdoi.net/10.13140/2.1.3107.9368 doi: 10.13140/2.1.3107.9368

Alvertis, I., Papaspyros, D., Koussouris, S., Mouzakitis, S., & Askounis, D. (2016). Using Crowdsourced and Anonymized Personas in the Requirements Elicitation and Software Development Phases of Software Engineering. In *11th international conference on availability, reliability and security (ARES)*. IEEE. Retrieved from https://doi.org/10.1109/ares.2016.71 doi: 10.1109/ares.2016.71

Angelis, G. D., Ferrari, A., Gnesi, S., & Polini, A. (2018). Requirements elicitation and refinement in collaborative research projects. *Journal of Software: Evolution and Process*, *30*(12), 1–18. Retrieved from https://doi.org/10.1002/smr.1990 doi: 10.1002/smr.1990

Araujo, R., Anjos, E., & Silva, D. R. (2015). Trends in the use of design thinking for embedded systems. In *15th International Conference on Computational Science and Its Applications* (p. 82–86). IEEE. Retrieved from https://doi.org/10.1109/iccsa.2015.25 doi: 10.1109/iccsa.2015.25

Asghar, A. R., Tabassum, A., Bhatti, S. N., & Jadi, A. M. (2017). Impact and challenges of requirements elicitation & prioritization in quality to agile process: Scrum as a case scenario. In *International conference on communication technologies (ComTech)*. IEEE. Retrieved from https://doi.org/10.1109/comtech.2017.8065749 doi: 10.1109/comtech.2017.8065749

Babar, A., Bunker, D., & Gill, A. Q. (2018). Investigating the Relationship between Business Analysts' Competency and IS Requirements Elicitation: A Thematic-analysis Approach. *Communications of the Association for Information Systems*, *42*(1), 334–362. Retrieved from https://doi.org/10.17705/1cais.04212 doi: 10.17705/1cais.04212

Baptista, M. N., & de Campos, D. C. (2007). *Metodologias de pesquisa em ciências: análises quantitativa e qualitativa* (2nd ed.). Rio de Janeiro: Livros Técnicos e Científicos.

Batista, E. A. (2003). *Uma taxonomia facetada para técnicas de elicitação de requisitos* (Master's thesis, Instituto de Computação, UNICAMP, Campinas, SP, Brasil). Retrieved from http://repositorio.unicamp.br/bitstream/REPOSIP/276373/1/Batista_EdinelsonAparecido_M.pdf (Dissertação de Mestrado)

Batool, A., Motla, Y. H., Hamid, B., Asghar, S., Riaz, M., Mukhtar, M., & Ahmed, M. (2013). Comparative study of traditional requirement engineering and agile requirement engineering. In *15th International Conference on Advanced Communications Technology (ICACT)* (pp. 1006–1014).

Buitrón, S. L., Flores-Rios, B. L., & Pino, F. J. (2018). Elicitación de requisitos no funcionales basada en la gestión de conocimiento de los stakeholders. *Ingeniare. Revista chilena de ingeniería*, *26*(1), 142–156. doi: 10.4067/S0718-33052018000100142

Caleb-Solly, P., Dogramadzi, S., Ellender, D., Fear, T., & Heuvel, H. v. d. (2014). A mixed-method approach to evoke creative and holistic thinking about robots in a home environment. In *ACM/IEEE International Conference on Human-Robot Interaction* (pp. 374–381). ACM Press. doi: 10.1145/2559636.2559681

Cervo, A. L., & Bervian, P. A. (2007). Metodologia científica: para uso dos estudantes universitários. In *Metodologia científica: para uso dos estudantes universitários* (6th ed.). S ao Paulo: Pearson Prentice Hall.

Fernandes, J., Duarte, D., Ribeiro, C., Farinha, C., Pereira, J. M., & da Silva, M. M. (2012). iThink: A game-based approach towards improving collaboration and participation in requirement elicitation. *Procedia Computer Science*, *15*(1), 66–77. doi: 10.1016/j.procs.2012.10.059

Gill, K. D., Zaidi, A. M., & Kiani, M. M. (2014). Eliciting futuristic end-user requirements through contributory appreciative inquiry. In *National software engineering conference* (pp. 49–54). IEEE. doi: 10.1109/NSEC.2014.6998240

Gonzales, C. K., & Leroy, G. (2011). Eliciting user requirements using appreciative inquiry. *Empirical Software Engineering*, *16*(6), 733–772. doi: 10.1007/s10664-011-9156-x

Hajri, I., Goknil, A., Briand, L. C., & Stephany, T. (2018). Configuring use case models in product families. *Software & Systems Modeling*, *17*(3), 939–971. doi: 10.1007/s10270-016-0539-8

Hickey, A. M., & Davis, A. M. (2003). Elicitation technique selection: how do experts do it? In *11th ieee international requirements engineering conference* (pp. 169–178). doi: 10.1109/ICRE.2003.1232748

Hidalga, A. N. d. l., Hardisty, A., & Jones, A. (2016). Scram–ck: applying a collaborative requirements engineering process for designing a web based e-science toolkit. *Requirements Engineering*, *21*(1), 107–129. doi: 10.1007/s00766-014-0212-0

Hosseini, M., Shahri, A., Phalp, K., Taylor, J., Ali, R., & Dalpiaz, F. (2015). Configuring crowdsourcing for requirements elicitation. In *9th International Conference on Research Challenges in Information Science (RCIS)* (pp. 133–138). doi: 10.1109/RCIS.2015.7128873

Jayatilleke, S., & Lai, R. (2018). A systematic review of requirements change management. *Information and Software Technology*, *93*, 163–185. doi: 10.1016/j.infsof.2017.09.004

Kiran, H. M., & Ali, Z. (2018). Requirement elicitation techniques for open source systems: a review. *International Journal of Advanced Computer Science and Applications (IJACSA)*, *9*(1), 330–334. doi: 10.14569/IJACSA.2018.090145

Kitchenham, B. (2004). *Procedures for performing systematic reviews*. Department of Computer Science, Keele University, UK. (Technical Report)

Kitchenham, B., & Charters, S. (2007). *Guidelines for*

performing systematic literature reviews in software engineering. Keele University and Durham University Joint Report. (Technical Report)

Knauss, E., Yussuf, A., Blincoe, K., Damian, D., & Knauss, A. (2018). Continuous clarification and emergent requirements flows in open-commercial software ecosystems. *Requirements Engineering*, *23*(1), 97–117. doi: 10.1007/s00766-016-0259-1

Mishra, D., Aydin, S., Mishra, A., & Ostrovska, S. (2018). Knowledge management in requirement elicitation: Situational methods view. *Computer Standards & Interfaces*, *56*(1), 49–61. doi: 10.1016/j.csi.2017.09.004

Mishra, D., Mishra, A., & Yazici, A. (2008). Successful requirement elicitation by combining requirement engineering techniques. In *First international conference on the applications of digital information and web technologies (icadiwt)* (pp. 258–263). doi: 10.1109/ICADIWT.2008.4664355

Mobasher, B., & Cleland-Huang, J. (2011). Recommender systems in requirements engineering. *Association for the Advancement of Artificial Intelligence (AAAI)*, *32*(3), 81–89. doi: 10.1609/aimag.v32i3.2366

Nuseibeh, B., & Easterbrook, S. (2000). Requirements engineering: a roadmap. In *Proceedings of the conference on the future of software engineering* (pp. 35–46). doi: 10.1145/336512.336523

Odusote, B., Daramola, O., & Adigun, M. (2018). Towards an extended misuse case framework for elicitation of cloud dependability requirements. In *Saicsit '18 proceedings of the annual conference of the south african institute of computer scientists and information technologists* (pp. 135–144). doi: 10.1145/3278681.3278698

Oriol, M., Stade, M., Fotrousi, F., Nadal, S., Varga, J., Seyff, N., ... Schmidt, O. (2018). Fame: supporting continuous requirements elicitation by combining user feedback and monitoring. In *2018 ieee 26th international requirements engineering conference (re)* (pp. 217–227). doi: 10.1109/RE.2018.00030

Pitula, K., & Radhakrishnan, T. (2011). On eliciting requirements from end-users in the ict4d domain. *Requirements Engineering*, *16*(1), 323. doi: 10.1007/s00766-011-0127-y

Ramakrishnan, L., Poon, S., Hendrix, V., Gunter, D., Pastorello, G. Z., & Agarwal, D. (2014). Experiences with user-centered design for the tigres workflow api. In *2014 IEEE 10th International Conference on e-Science* (pp. 290–297). doi: 10.1109/eScience.2014.56

Ramesh, M. R., & Reddy, C. S. (2016). A survey on security requirement elicitation methods: classification, merits and demerits. *International Journal of Applied Engineering Research*, *11*(1), 64–70.

Sadiq, M., Ghafir, S., & Shahid, M. (2009). An approach for eliciting software requirements and its prioritization using analytic hierarchy process. In *2009 international conference on advances in recent technologies in communication and computing* (pp. 790–795). doi: 10.1109/ARTCom.2009.58

Sheffield, J., & Lemétayer, J. (2013). Factors associated with the software development agility of successful projects. *International Journal of Project Management*, *31*(1), 459–472. doi: 10.1016/j.ijproman.2012.09.011

Smith, C., Strauss, J., & Maher, P. (2010). Data structure visualization: the design and implementation of an animation tool. In *Proceedings of the 48th annual southeast regional conference* (pp. 1–6). doi: 10.1145/1900008.1900105

Thomas, K., Bandara, A. K., Price, B. A., & Nuseibeh, B. (2014). Distilling privacy requirements for mobile applications. In *Proceedings of the 36th international conference on software engineering* (pp. 871–882). doi: 10.1145/2568225.2568240

Vijayan, J., Raju, G., & Joseph, M. (2016). Collaborative requirements elicitation using elicitation tool for small projects. In *2016 international conference on signal processing, communication, power and embedded system (scopes)* (pp. 340–344). doi: 10.1145/10.1109/SCOPES.2016.7955848

Wertsch, J. V. (1981). The concept of activity in soviet psychology: an introduction. In J. V. Wertsch (Ed.), *The concept of activity in soviet psychology* (p. 86-99). Armonk: M. E. Sharpe.

Younas, M., Jawawi, D., Ghani, I., & Kazmi, R. (2017). Non-functional requirements elicitation guideline for agile methods. *Journal of Telecommunication, Electronic and Computer Engineering*, *9*(1), 137–142.

How to cite this article (APA):
Alflen, N. C. & Prado, E. P. V. (2021). . *AtoZ: novas práticas em informação e conhecimento, 10*(1), 118 – 128. Retrieved from: http://dx.doi.org/10.5380/atoz.v10i1.77393