COMPUTATION E INFORMATICA

Recibido 15 abr. 2022 *ReCIBE, Año 11 No. 1, may. 2022*

Aceptado 09 jun. 2022

User- Centered Methodologies for Requirements Engineering: A Comparative Analysis using ISO/IEC/IEEE 29148

Metodologías Centradas en el Usuario para la Ingeniería de Requisitos: un Análisis Comparativo utilizando ISO/IEC/IEEE 29148

Alex Llanque[1]
allanquef@unam.edu.pe

Vaneza Flores[1]
vfloresg@unam.edu.pe

Abraham Dávila[2]
abraham.davila@pucp.edu.pe

[1]Escuela de Ingeniería de Sistemas e Informática Universidad Nacional de Moquegua, UNAM, Moquegua, Perú
[2]Departamento de Ingeniería Pontificia Universidad Católica del Perú, PUCP Lima, Perú

ABSTRACT

Requirement engineering is a fundamental process of the software development life cycle processes that allows defining the functionalities, quality and scope of a software. Requirement engineering is a human-intensive participation process and that have in ISO/IEC/IEEE 29148 a standard that defines a set of process (activities, tasks, information items, etc.) that provide support for its formalization. However, in the software industry there are many types of deficiencies have been reported. In this context, proposals for User-Centered Methodologies for Requirement Engineering (UCMRE) have emerged and need to be studied to determine their alignment to the ISO 29148. The objective of this study is to compare User-Centered Methodologies for Requirement Engineering taking process elements from ISO 29148 as criteria. For this research, a systematic mapping study and a comparative analysis of the UCMREs obtained were performed. In the systematic mapping study, 4,463 studies were obtained from three relevant digital databases in the first stage, and after the selection process, five User-Centered Methodologies for Requirement Engineering were identified. These methodologies were characterized and analyzed from process elements perspective for comparison to ISO 29148. The DoRCU methodology is the closest to ISO 29148, followed by Ammeth and Borja methodologies; and the XRE methodology is the most away to ISO 29148.

Keywords: Software requirements, Requirement engineering, User centered, Software methodology, ISO/IEC 29148.

RESUMEN

La ingeniería de requisitos es un proceso fundamental de los procesos del ciclo de vida del desarrollo de software que permite definir las funcionalidades, la calidad y el alcance de un software. La ingeniería de requisitos es un proceso de participación humana intensiva y que tiene en ISO/IEC/IEEE 29148 un estándar que define un conjunto de procesos (actividades, tareas, elementos de información, etc.) que brindan soporte para su formalización. Sin embargo, en la industria del software se han reportado muchos tipos de deficiencias. En este contexto, han surgido propuestas de Metodologías Centradas en el Usuario para Ingeniería de Requisitos (MCUIR) y necesitan ser estudiadas para determinar su alineación con la ISO 29148. El objetivo de este estudio es comparar Metodologías Centradas en el Usuario para Ingeniería de Requisitos tomando elementos de proceso de ISO 29148 como criterio. Para esta investigación se realizó un mapeo sistemático de la literatura y un análisis comparativo de las UCMRE obtenidas. En el mapeo sistemático de la literatura, se obtuvieron 4.463 estudios de tres bases de datos digitales relevantes en la primera etapa, y luego del proceso de selección, se identificaron cinco Metodologías Centradas en el Usuario para Ingeniería de Requisitos. Estas metodologías se caracterizaron y analizaron desde la perspectiva de los elementos del proceso para compararlas con la norma ISO 29148. La metodología DoRCU es la más cercana a la norma ISO 29148, seguida de las metodologías Ammeth y Borja; y la metodología XRE es la que más se aleja de la ISO 29148.

Palabras clave: Requisitos de software, Ingeniería de requisitos, Centrado en el usuario, Metodología de Software, ISO/IEC 29148.

1 INTRODUCTION

Requirement engineering (RE) is the initial phase of the software life cycle (Nuseibeh & Easterbrook, 2000) and represents, in the software industry, a process where many defects are introduced (Kelly et al., 1992), (Westfall, 2011). These defects affect the scope and performance of the project, the work teams and the quality of the product (Hu et al., 2017). By the nature of software development, defects introduce in requirement phase are related to people (users, customers, developers and other stakeholders) (Westfall, 2011) and have been classified in different ways (Anu et al., 2018).

For several decades, some proposals and researches have been performed in the RE domain. In Nuseibeh & Easterbrook (2000) the roadmap of RE is identified, in Nazir et al. (2017) RE is presented related to natural language processing, and in Ambreen et al. (2018) with formal methods, among others related to techniques. There has also been research on RE practices, methodologies and process (Niazi, 2005), (Beecham et al., 2005), (Niazi & Shastry, 2003) and RE maturity model (Shafiq et al., 2019). Finally, the international standard ISO/IEC/IEEE 29148 (or simply ISO 29148) on RE in systems and software has become an important reference for the industry (ISO/IEC/IEEE, 2018).

In the current context, ISO standards represent an accepted reference in different industries and have allowed various companies to increase their competitiveness (Miotti, 2009). This benefit has also been noted in the software industry, where process models have contributed to increase the competitiveness of companies (UNCTAD, 2012), as the case with CMMI and ISO/IEC 12207. In this sense, the software industry, have in ISO 29148, a highly specialized reference for requirements engineering processes, so it is convenient to consider it as a reference to evaluate existing methodologies.

On the other hand, recognizing that RE is a human insensitive activity, and that the concept of "user-centered (UC) design" has already been consolidated as a new paradigm in software engineering (Chang, 2018), (De Bellis & Haapala, 1995), it is necessary to identify the proposals of UCMRE (User-Centered Methodology of Requirement Engineering) and contrast them with respect to the ISO/IEC/IEEE 29148 international standard (ISO/IEC/IEEE, 2018). In this contrast, it is possible to evaluate the alignment that a UCMRE has with respect to the standard; and that information is one more element of judgment to make future decisions about which UCMRE to adopt.

This article presents a comparison of a set of UCMRE, taking the RE standard processes and activities as the criteria for comparison. The article is organized as follows: in Section 2, fundamental concepts and related works are reviewed; in Section 3, the research model followed is presented; in Section 4, the selected methodologies are described and their comparison is presented; in Section 5, conclusions are presented.

2 BACKGROUND

This section briefly describes ISO 29148 emphasizing the suggested process items with which the axes of analysis are established and presents the relevant aspects of RE and the UC paradigm.

2.1 ISO/IEC/IEEE 29148

ISO 29148 is a systems and software engineering standard that specifies the required processes to determine the product requirements (including services) of systems and software throughout the life cycle (ISO/IEC/IEEE, 2018), (IEEE-CS, 2014). ISO 29148 is aligned with ISO/IEC/IEEE 12207 standard for software life cycle processes and ISO/IEC/IEEE 15288 standard for system life cycle processes (IEEE-CS, 2014). ISO 29148, in its first edition (2011), replaced the IEEE 830-1998, IEEE 1233-1998 and IEEE 1362-1998 standards, also related to requirements engineering (IEEE SA, 2011), becoming the main standard in RE by providing a model adaptable to different organizations (Selvyanti & Bandung, 2017). In 2018, the second edition was published, due to the need to adopt a framework oriented towards a more detailed

technical approach (Ward et al., 2018), and in Boyarchuk et al. (2020) the importance of having a structured process in RE to build a good software product is highlighted.

ISO 29148 establishes (ISO/IEC/IEEE, 2018): (i) a set of 3 main processes that are iteratively and recursively applied in the system and software life cycle (see Fig. 1); (ii) a set of 2 related technical processes; (iii) each main process is composed of activities with established objectives to achieve a correct identification and good description of requirements; (iv) a set of 4 types of documents for requirements specification at business, stakeholder, system and

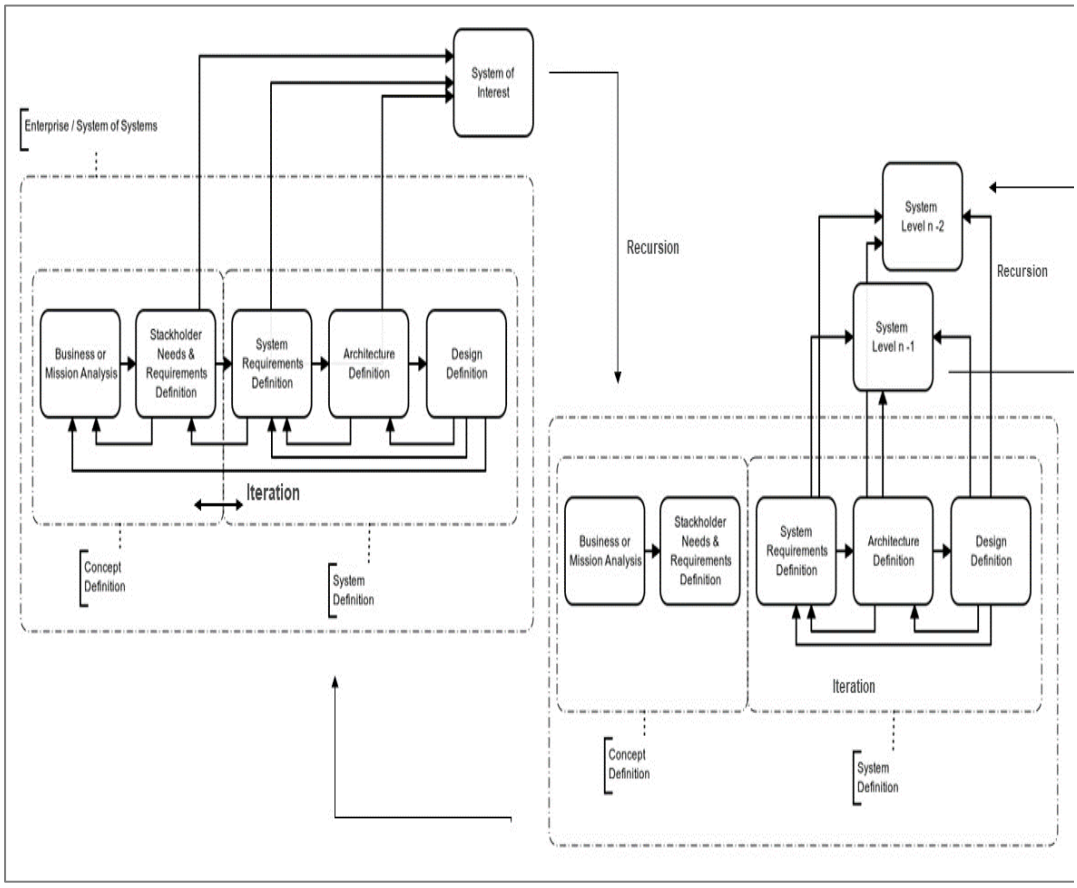

Figure 1. Requirements Engineering Process set out in 29148:2018 adapted from (ISO/IEC/IEEE, 2018)

software levels; (v) the concept of operations and System operational concept; and, (vi) guidelines to manage the evolution of requirements.

2.2 User-Centered Software Requirements

The user-centered software development approach (UCD), was stablished as a response to existing software development schemes, which in practical terms can be noted, were focused on the technical aspect of the software and associated hardware (De Bellis & Haapala, 1995), (Norman & Draper, 1987). The UCD gained some relevance with the publication of ISO 13407:1999 Human-centred Design Processes for Interactive Systems (Juárez-Ramírez, 2017) and after that, when the stakeholders are included into ISO 9241-210:2010 Ergonomics of human-system interaction - Part 210: Human-centred Design for Interactive Systems (Vidal et al., 2012). Finally, the latest edition of the ISO 9241-210 standard is, recently, from 2019 (IEEE SA, 2011). In short, as pointed out by several authors (Juárez-Ramírez, 2017), (Forbrig, 2016), (Nieva et al., 2016), (Wanderley et al., 2014), and the ISO 13407 standard itself (ISO, 2019), RE should allow to reach an understanding of the needs, attitudes, motivations and behaviors of users that are the basis for achieving successful products and services, with a wide margin of end-user satisfaction.

3 RESEARCH METHOD

In this research a hybrid model was used. In the first part, a Systematic Mapping Study (SMS) was performed to identify user-centric methodologies for requirement engineering (UCMRE); and in the second part, a comparative analysis was performed using process elements from ISO 29148 as criteria.

3.1 Systematic Mapping Study

In order to select the methodologies, an SMS was carried out, taking as a reference what was established by Petersen (2015). The stages carried out were:

3.1.1 SMS Planification

The main research question established was “what user-centered software requirements engineering methodologies (UCMREs) have been proposed?” To answer this question, the search string was elaborated using the concepts of Population and Intervention, according to Petersen (2015). For the population the terms "requirements engineering" and for the intervention "methodology" were considered. From these main terms and the alternate terms, the final search string was set as "("Software Requirement" OR "Requirements Engineering") AND (methodology OR process) AND (elicitation OR analysis OR specification OR validation)". In addition, we searched the bibliographic references of some secondary studies related to RE,

Id	Study	Author(s)
S01	DoRCU Methodology for Requirement Engineering.	G. Báez & B. Brunner
S02	Extreme Requirement Engineering (XRE)	N. Ikram & S. Naz
S03	AMMETH: A Methodology for Requirements Analysis of Advanced Human-System Interfaces	G. Guida & G. Lamperti
S04	Methodology for the Specification of Software Requirements Based on the IEEE 830-1998 Standard	C. Borja & V. Cuji
S05	Methodology for developing requirements in applications with web services	P. Páez, C. Arias & L. Wanumen

Table **1**. Comparison of processes / activities between ISO 29148 and UCMRE

with the purpose of identifying other possible candidates. We established 1999 as the cut-off date for the search, which was when the ISO 13407 Human-Centred design processes for interactive systems standard was published for the first time, which is a reference point for this study. Although English keywords were used, articles whose contents were also in Portuguese or Spanish were accepted. The inclusion criteria applied were: (i) refer to RE methodologies; (ii) are in the range 2000 to date; and, (iii) refer to user-centered. The exclusion criteria applied were: (i) refer to RE techniques (ii) refer to secondary studies; (ii) are not accessible in the databases consulted. In addition, the contents were quickly reviewed and compliance with the criteria was verified.

3.1.2 SMS Performed

The selected databases were IEEE Xplore, Scopus and Web of Science. The query was carried out in April 2021, obtaining 4,463 studies. The selection process used inclusion and exclusion criteria by stages, applied to titles, abstracts and content. In the title reading stage, 231 articles remained, in the abstract reading stage, 39 articles were selected, and finally, in the rapid review of abstracts, 5 articles remained (see Table 1). In Section 4, every methodology identified is described.

3.2 Comparative Analysis

The comparison of methodologies is an activity that is carried out constantly as can be seen in the studies of (Saleh et al., 2017), (Chandra, 2015), (Subbarayudu et al., 2017), (Cano et al., 2015). In these studies, they stablishing, in all cases, the comparison criteria according to the purpose sought. In addition, literature research has been carried out on the characteristics of the methodologies (Toro & Gálvez, 2016).

Id*	Process element	Description(s)
6.2	Business or mission analysis process	Identify problems and opportunities in the organization. Define the analysis strategy to propose solutions aligned with the organization's objectives.
6.3	Stakeholder needs and requirement definition processes	Define and perform activities related to defining the needs and requirements of stakeholders in the established environment and its context.
6.3.3.2	Prepare for Stakeholder Needs and Requirement Definitions	Define and perform activities related to preparing for Stakeholder Needs and Requirements Definition.
	Identify the stakeholders	Define and perform activities related to identify stakeholders in the system and organize them in levels according to their influence.
	Requirements definition strategy	Define and perform activities related to planning the necessary tasks and resources to obtain and manage the needs of stakeholders
6.3.3.3	Define stakeholder needs	Define and perform activities related to defining the objectives and problems of stakeholders.
	Identify stakeholder needs.	Define and perform activities related to describing and documenting needs of stakeholder.
	Prioritize and down-select needs.	Define and perform activities related to reviewing, classifying and selecting needs of the stakeholder.
6.3.3.4	Develop the operational concept and other life cycle concepts	Define and perform activities related to define usage sceneries aimed at developing the operational concept and other life cycle concepts.
	Identify the interaction between users and the system.	Define and perform activities related to understanding and documenting the relationship between the user and the system.
	Identify the factors affecting interactions between users and the system.	Define and perform activities related to understand and document the factors affecting interactions between users and systems.
6.3.3.5	Transform stakeholder needs into stakeholder requirements	Define and perform activities related to define stakeholder requirements based on their need statements and the constraints of the environment.
6.3.3.6	Analyze stakeholder requirements.	Define and perform activities related to classifying, reviewing and prioritizing the requirements, to validate them against the stakeholders.
6.3.3.7	Manage the stakeholder needs and requirements definition	Define and perform activities related to manage the stakeholder needs and requirements definition.
6.4	System [System/Software] Requirements definition process	Define and perform activities related to System/Software requirement definition.
6.4.3.2	Prepare for [System/Software] Requirement Definitions	Define and perform activities related to prepare for System/Software requirements definition.
6.4.3.3	Define system[/software] requirements	Define and perform activities related to define System/Software requirements.
6.4.3.4	Analyze system[/software] requirements	Define and perform activities related to analyze System/Software requirements.
6.4.3.5	Manage system[/software] requirements	Define and perform activities related to manage System/Software requirements.
	Maintain traceability of the system[/software] requirements.	Define and perform activities related to maintain traceability of the System/Software requirements.
6.5	Requirements engineering activities in other technical processes	Define and perform activities related to requirements in other technical processes.
6.5.1	Requirements activities in architecture definition	Define and perform activities related to requirements in the architecture definition phase.
6.5.2	Requirements activities in verification	Define and perform activities related to requirements in the verification phase.
6.5.3	Requirements activities in validation	Define and perform activities related to requirements in the validation phase.
6.6	Requirements management	Manage requirements (needs and change) and keep them consistent.
9.3	Business requirement specification (BRS)	Describe elements of Business requirements specification (BRS).
9.4	Stakeholder requirements specif. (StRS)	Describe elements of Stakeholder requirement specification (StRS).
9.5	System requirements specification (SyRS)	Describe elements of System requirement specification (SyRS).
9.6	Software requirements specification (SRS)	Describe elements of Software requirement specification (SRS).

Table **2**. Criteria description based on ISO29148 to compare UCMRE's model

For the comparative analysis, in this study, some process elements (activities and information items) proposed in ISO 29148 were used as comparison criteria. In Table 2, every criterion is shortly described.

In order to compare the methodologies obtained in the SMS, the following activities were carried out: (i) a detailed review of the existing documentation of each methodology was performed; (ii) a process diagram of each methodology including activities and deliverables was elaborated; (iii) criteria and a base structure for comparison of the methodologies was established from ISO 29148; (iv) the ISO 29148 standard was carefully reviewed understanding what applies to the software engineering context; (v) the similarity of the methodology with respect to the ISO 29148 standard was analyzed and discussed in a qualitatively way.

4 RESULTS

This section presents, in summary, the user-centric methodologies for requirement engineering (UCMRE) identified in the SMS and their comparison.

4.1 DoRCU Methodology

DoRCU, according to Griselda Báez & Barba Brunner (2001), has been developed in an academic environment and validated in a project during a public health context in Cuba. It is called DoRCU for Documentation of User-Centered Requirements. The article presents the following characteristics (Griselda Báez & Barba Brunner, 2001): (i) it consists of 4 stages: elicitation, analysis, specification and validation and certification of requirements; (ii) it describes an iterative process; (iii) it involves high user participation in all its stages; (iv) it prioritizes the satisfaction of user needs; and (v) it is adaptable to other models (flexibility). Fig. 2 shows a representation of DoRCU based on its proposal.

4.2 XRE Methodology

XRE, according to Ikram & Naz (2015), is based on an agile approach, from a survey of RE experts around the world, on the valuable practices of Extreme Programming (XP) and Scrum. The features noted are (Ikram & Naz, 2015): (i) it comprises the following activities Daily Standup Meeting, Sprint Planning, Initial/Product Planning Meeting, Sprint Review, Sprint Retrospective Meeting and Sprint Planning; (ii) it is described as an iterative and evolutionary process with ongoing communication; (iii) the role nature of an agile team is maintained, integrating the user as part of the team; (iv) it employs graphical requirements management in order to maintain traceability and communication with users; and (v) the requirements document is adaptable to the needs. In Fig. 3, XRE has been represented based on what is established in its proposal.

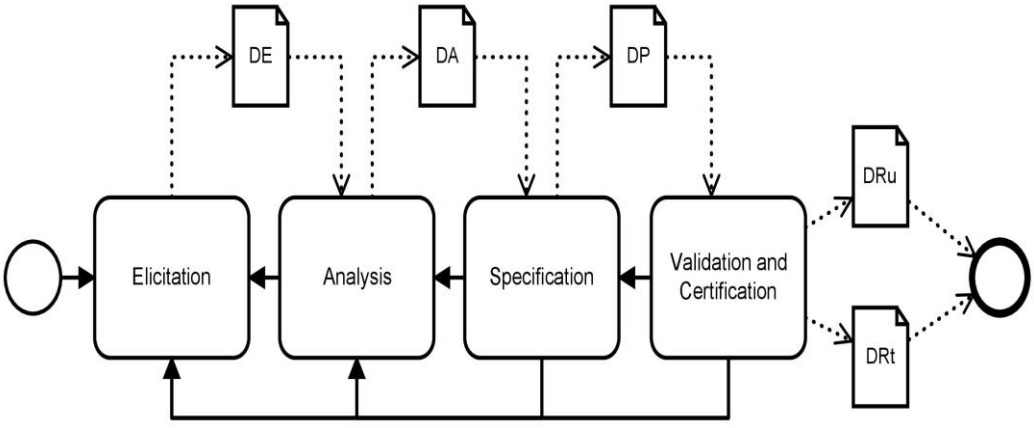

Figure 2. DoRCU Methodology adapted from (Griselda Báez & Barba Brunner, 2001)

4.3 AMMETH Methodology

AMMETH is a methodology that can help in performing Human-System interface (HSI) requirements analysis in a disciplined and effective way (Guida & Lamperti, 2000). AMMETH was applied in a complex diagnostic and process monitoring system of a power plant (Guida & Lamperti, 2000). The methodology involves seven steps based on some proven analysis techniques presented below (Guida & Lamperti, 2000): (i) analyzing the context, (ii) stating interaction goals, (iii) eliciting user needs and expectations, (iv) identifying and rating interaction features, (v) defining interaction patterns, (vi) collecting usability feedback, and (vii) defining requirements. Finally, AMMETH (Guida & Lamperti, 2000), is declared iterative because it allows refinement by stages, involves user participation from early stages to the evaluation of the scenarios, being the heart of the methodology. In Fig. 4, AMMETH has been represented based on what is established in its proposal.

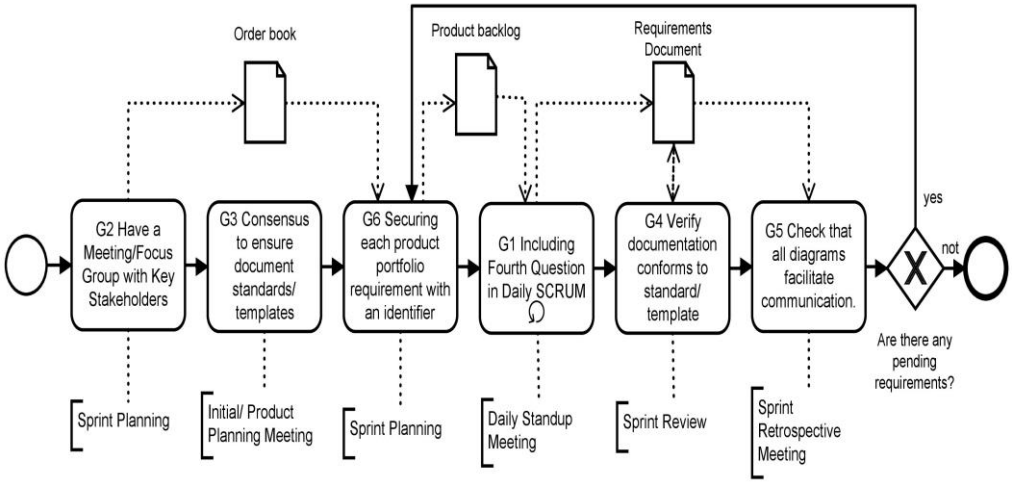

Figure 3. XRE Methodology adapted from (Ikram & Naz, 2015)

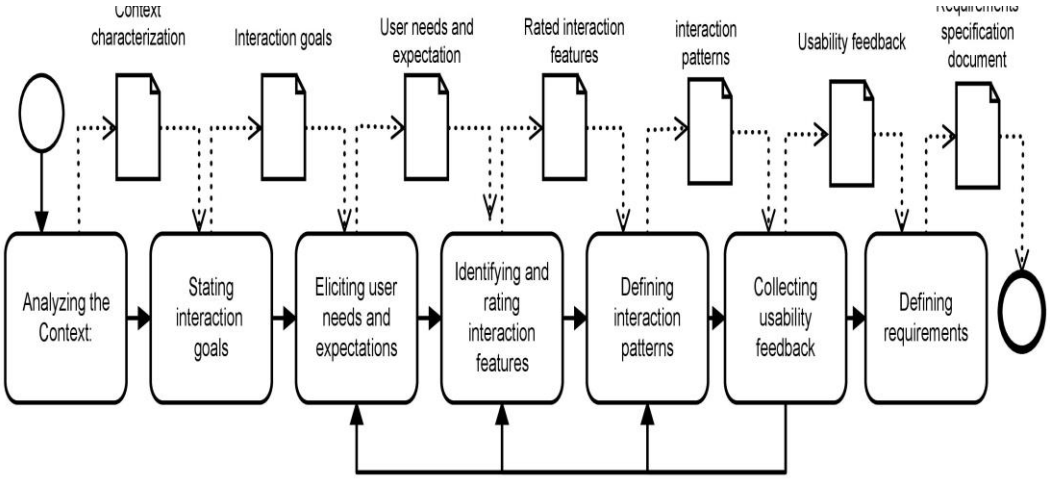

Figure 4. AMMETH Methodology adapted from (Guida & Lamperti, 2000)

4.4 Borja Methodology

The Borja's methodology is based on the IEEE 830-1998 standard and was applied for its validation in a management system of an industrial technical college (Borja Buestán & Cuji Torres, 2013). The methodology has the following characteristics (Borja Buestán & Cuji Torres, 2013): (i) it has four phases such as elicitation, analysis, specification and validation of requirements; (ii) it is defined as an iterative, incremental and cooperative process; (iii) it incorporates the participation of the user as the main actor in all stages; (iv) it recommends to

apply the use case diagrams; and, (v) the elaboration of a software requirements specification document under the structure of the standard. Fig. 5 shows Borja's proposal based on what is established in his proposal.

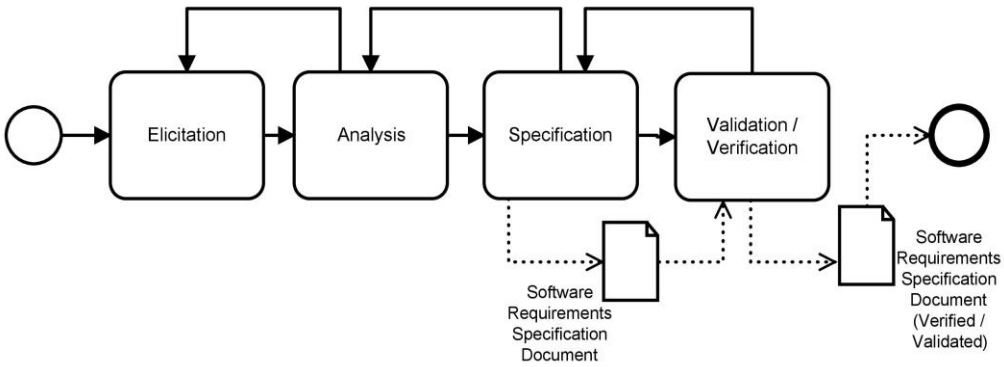

Figure 5. Borja Methodology adapted from (Borja Buestán & Cuji Torres, 2013)

4.5 Paez Methodology

The Paez´s methodology for requirements elaboration in web services applications (Paez Cardenas et al., 2018) was validated by means of a survey applied to 2 software development teams using the Technology Acceptance Model (TAM). The Paez´s methodology (Paez Cardenas et al., 2018) comprises 7 phases: soft requirements capture, elicitation, analysis, semantic validation, specification, validation and verification. The methodology has the following characteristics (Paez Cardenas et al., 2018): (i) it is presented as an iterative process; (ii) it incorporates aspects related to System Soft Methodology; (iii) it is oriented to address the goals of the end users as well as that of the development team; and (iv) it employs a set of adaptable artifacts in the different stages. Fig. 6 shows Paez's proposal based on what is established in his model.

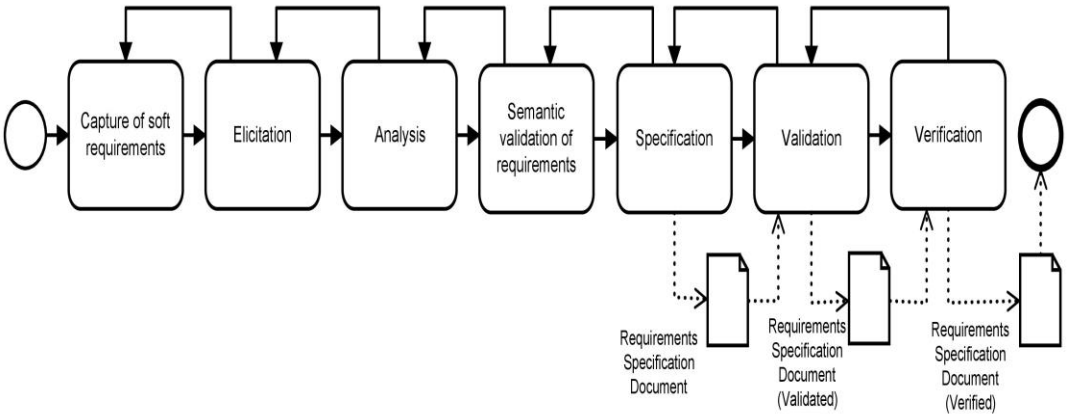

Figure 6. Paez Methodology adapted from (Paez Cardenas et al., 2018)

4.6 Comparison of Methodologies

In Table 3 shows the ISO 29148 processes and activities. The numbers in the item column correspond to the section of the process/activity within the Standard, followed by the description column. The following columns correspond to the methodologies considered.

5 DISCUSSION AND CONCLUSIONS

In this study, an SMS has been carried out where five user-centric methodologies for requirement engineering (UCMRE) were obtained: DoRCU, XRE, Ammeth, Borja and Paéz. The SMS has followed a process based on the methodology based on Petersen (2015) in three digital databases of interest for software engineering.

Subsequently, considering the relevance of ISO 29148, a standard updated in 2018, process elements from this standard were used as criteria to compare the five UCMRE identified. This comparison has made it possible to identify that DoRCU is the most aligned with ISO 29148, followed by, without an important distinction between them, Ammeth, Borja and Paéz; finally, XRE is the least aligned of the group of UCMRE evaluated. Likewise, it is observed that all

Id*	Process element	DoRCU	XRE	AMMETH	Borja	Páez
6.2	Business or mission analysis process	Y	Y	Y	Y	Y
6.3	Stakeholder needs and requirements definition process	---	---	---	---	---
6.3.3.2	Prepare for Stakeholder Needs and Requirements Definition	Y	-	Y	Y	Y
	Identify the stakeholders	Y	Y	Y	Y	Y
	Requirements definition strategy	Y	-	Y	Y	Y
6.3.3.3	Define stakeholder needs	Y	-	Y	Y	Y
	Identify stakeholder needs.	Y	Y	Y	Y	Y
	Prioritize and down-select needs.	Y	Y	Y	Y	Y
6.3.3.4	Develop the operational concept and other life cycle concepts	Y	-	Y	Y	Y
	Identify the interaction between users and the system.	Y	-	Y	Y	Y
	Identify the factors affecting interactions between users and the system.	Y	-	Y	Y	-
6.3.3.5	Transform stakeholder needs into stakeholder requirements	Y	Y	Y	Y	Y
6.3.3.6	Analyze stakeholder requirements.	Y	Y	Y	Y	Y
6.3.3.7	Manage the stakeholder needs and requirements definition	Y	Y	-	Y	Y
6.4	System [System/Software] Requirements definition process	---	---	---	---	---
6.4.3.2	Prepare for System [System/Software] Requirements Definition	Y	Y	Y	-	Y
6.4.3.3	Define system[/software] requirements	Y	Y	Y	Y	Y
6.4.3.4	Analyze system[/software] requirements	Y	Y	Y	Y	Y
6.4.3.5	Manage system[/software] requirements	Y	-	Y	Y	-
	Maintain traceability of the system[/software] requirements.	Y	Y	Y	-	-
6.5	Requirements engineering activities in other technical processes	---	---	---	---	---
6.5.1	Requirements activities in architecture definition	-	-	-	Y	-
6.5.2	Requirements activities in verification	Y	Y	Y	Y	Y
6.5.3	Requirements activities in validation	Y	Y	Y	Y	Y
6.6	Requirements management	Y	Y	Y	Y	Y
9.3	Business requirement specification (BRS)	Y	-	-	-	-
9.4	Stakeholder requirements specification (StRS)	Y	-	Y	Y	Y
9.5	System requirements specification (SyRS)	Y	-	-	-	-
9.6	Software requirements specification (SRS)	Y	Y	Y	Y	Y
	Total	**25**	**15**	**22**	**22**	**20**

Table 3. Comparison of processes / activities between ISO 29148 and UCMRE identified

complete the SRS document (software requirements specifications); all but XRE also complete the StRS (stakeholder requirements specifications) document; and neither covers the Concept of Operations item, nor the System Operational Systems; that are more related to the system level and less to the software level.

ACKNOWLEDGMENTS

This work is partially supported by Department of Engineering and the Software Engineering Research and Development Group (GIDIS) from the Pontifical Catholic University of Peru (PUCP). The work is also funded by the Data Science and Technology Research Group (CITEC) of the National University of Moquegua (UNAM).

REFERENCES

Ambreen, T., Ikram, N., Usman, M., & Niazi, M. (2018). Empirical Research in Requirements Engineering: Trends and Opportunities. *Requirements Engineering*, *23*(1), 63–95. https://doi.org/10.1007/s00766-016-0258-2

Anu, V., Hu, W., Carver, J. C., Walia, G. S., & Bradshaw, G. (2018). Development of a Human Error Taxonomy for Software Requirements: A Systematic Literature Review. *Information and Software Technology*, *103*, 112–124. https://doi.org/10.1016/j.infsof.2018.06.011

Beecham, S., Hall, T., & Rainer, A. (2005). Defining a Requirements Process Improvement Model. *Software Quality Journal*, *13*, 247–279. https://doi.org/10.1007/s11219-005-1752-9

Borja Buestán, C. D., & Cuji Torres, V. A. (2013). Metodología para la Especificación de Requerimientos de Software Basado en el Estándar IEEE 830-1998. In *Tesis - Universidad Politécnica Salesiana - Sede Cuenca*.

Boyarchuk, A., Pavlova, O., Bodnar, M., & Lopatto, I. (2020). Approach to the Analysis of Software Requirements Specification on its Structure Correctness. *CEUR Workshop*, 85–95. http://ceur-ws.org/Vol-2623/paper9.pdf

Cano, C., Melgar, A., Dávila, A., & Pessoa, M. (2015). Comparison of Software Process Models. A Systematic Literature Review. *41st Latin American Computing Conference - CLEI 2015*, 1–6. https://doi.org/10.1109/CLEI.2015.7360025

Chandra, V. (2015). Comparison between Various Software Development Methodologies. *International Journal of Computer Applications*, *131*(9), 7–10. https://doi.org/10.5120/ijca2015907294

Chang, C. K. (2018). Situation Analytics — at the Dawn of a New Software Engineering Paradigm. *Science China - Information Sciences*, *61*(5), 1–14. https://doi.org/10.1007/s11432-017-9372-7

De Bellis, M., & Haapala, C. (1995). User-Centric Software Engineering. *IEEE Expert*, *10*(1), 34–41.

Forbrig, P. (2016). Continuous Requirements Engineering and Human-Centered Agile Software Development. *International Working Conference on Requirement Engineering Foundation for Software Quality - REFSQ 2016*, 1–11.

Griselda Báez, M., & Barba Brunner, S. I. (2001). Metodología DoRCU para la Ingeniería de Requerimientos. *WorkShop on Requirement Engineering*, 1–13.

Guida, G., & Lamperti, G. (2000). AMMETH: a Methodology for Requirements Analysis of Advanced Human-System Interfaces. *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans.*, *30*(3), 298–321. https://doi.org/10.1109/3468.844355

Hu, W., Carver, J. C., Anu, V., Walia, G., & Bradshaw, G. (2017). Defect Prevention in Requirements Using Human Error Information: An Empirical Study. *International Working Conference on Requirements Engineering: Foundation for Software Quality - REFSQ 2017. In Lecture Notes in Computer Science*, *10153 LNCS*, 61–76. https://doi.org/10.1007/978-3-319-54045-0_5

IEEE-CS. (2014). SWEBOK Guide V3.0 - Guide to the Software Engineering Body of Knowledge. In P. Bourque & R. Fairley (Eds.), *IEEE*.

IEEE SA. (2011). *IEEE SA Standards*. Web EEE/ISO/IEC 29148-2011. https://standards.ieee.org/standard/29148-2011.html

Ikram, N., & Naz, S. (2015). Extreme Requirements Engineering (XRE). *Requirements Engineering in the Big Data Era. In Communications in Computer and Information Science*, *558*, 95–108. https://doi.org/10.1007/978-3-662-48634-4_7

ISO/IEC/IEEE. (2018). ISO/IEC/IEEE 29148:2018 Systems and Software Engineering — Life Cycle Processes — Requirements Engineering. In *ISO* (2nd ed.). https://www.iso.org/standard/72089.html

ISO. (2019). ISO 9241-210: 2019 Ergonomics of Human-System Interaction — Part 210: Human-Centred Design for Interactive Systems Ergonomie. 2do Edition. In *ISO*.

Juárez-Ramírez, R. (2017). User-Centered Design and Adaptive Systems: Toward Improving Usability and Accessibility. *Universal Access in the Information Society*, *16*, 361–363. https://doi.org/10.1007/s10209-016-0480-1

Kelly, J. C., Sherif, J. S., & Hops, J. (1992). Analysis of Defect Densities Found During Software Inspections. *NASA -15th Annual Software Engineering Workshop*, *17*(2), 111–117.

Miotti, H. (2009). The Economic Impact of Standardization. Technological Change, Standards Growth in France. In *AFNOR*.

Nazir, F., Butt, W. H., Anwar, M. W., & Khan Khattak, M. A. (2017). The Applications of Natural Language Processing (NLP) for Software Requirement Engineering - A Systematic Literature Review. *ICISA 2017 Information Science and Applications. In Lecture Notes in Electrical Engineering*, *424*, 485–493. https://doi.org/10.1007/978-981-10-4154-9_56

Niazi, M. (2005). An Instrument for Measuring the Maturity of Requirements Engineering Process. *Product Focused Software Process Improvement - PROFES 2005 . In Lecture Notes in Computer Science*, *3547*, 574–585. https://doi.org/10.1007/11497455_44

Niazi, M., & Shastry, S. (2003). Critical Success Factors for the Improvement of Requirements Engineering Process. *International Conference on Software Engineering Research and Practise*, 433–439.

Nieva, A., Hardisty, A., & Jones, A. (2016). SCRAM–CK: Applying a Collaborative Requirements Engineering Process for Designing a Web Based e-Science Toolkit. *Requirements Engineering*, *21*, 107–129. https://doi.org/10.1007/s00766-014-0212-0

Norman, D. A., & Draper, S. W. (1987). User Centred System Design-New Perspectives on Human/Computer Interaction. *Journal of Educational Computing Research*, *3*(1), 129–134.

Nuseibeh, B., & Easterbrook, S. (2000). Requirements Engineering: A Roadmap. *International Conference on The Future of Software Engineering - ICSE 2000.*, 35–46. http://mcs.open.ac.uk/ban25/papers/sotar.re.pdf%5Cn./CitaviFiles/Nuseibeh2000.pdf

Paez Cardenas, P., Arias Peralta, C. D., & Wanumen Silva, L. F. (2018). Metodología para Elaboración de Requerimientos en Aplicaciones con Servicios Web. *Revista Vínculos*, *15*(2), 160–174. https://doi.org/10.14483/2322939x.13505

Petersen, K., Vakkalanka, S., & Kuzniarz, L. (2015). Guidelines for Conducting Systematic Mapping Studies in Software Engineering: An Update. *Information and Software Technology*, *64*, 1–18. https://doi.org/10.1016/j.infsof.2015.03.007

Saleh, S. M., Rahman, M. A., & Asgor, K. A. (2017). Comparative Study on the Software Methodologies for Effective Software Development. *International Journal of Scientific & Engineering Research*, *8*(4), 1018–1025.

Selvyanti, D., & Bandung, Y. (2017). The Requirements Engineering Framework Based on ISO 29148:2011 and Multi-View Modeling Framework. *International Conference on Information Technology Systems and Innovation - ICITSI 2017*, 128–133. https://doi.org/10.1109/ICITSI.2017.8267930

Shafiq, M., Zhang, Q., & Akbar, M. A. (2019). Software Requirements Engineering Maturity

Model (SREMM) in Offshore Software Development Outsourcing. *International Conference on Frontiers of Information Technology - FIT 2019*, 101–104. https://doi.org/10.1109/FIT47737.2019.00028

Subbarayudu, B., Srija Harshika, D., Amareswar, E., Reddy, R. G., & Reddy, K. K. (2017). Review and Comparison on Software Process Models. *International Journal of Mechanical Engineering and Technology*, *8*(8), 967–980.

Toro, A., & Gálvez, J. G. (2016). Especificación de Requisitos de Software: Una Mirada desde la Revisión Teórica de Antecedentes. *Entre Ciencia e Ingeniería*, *10*(19), 108–113. http://biblioteca.ucp.edu.co/OJS/index.php/entrecei/article/view/2803/2785

UNCTAD. (2012). Information Economy Report 2012. The Software Industry and Developing Countries. In *United Nations Publication*.

Vidal, D., Ibarra, J., Flores, B., & Lopez, G. (2012). Adopción del Estándar ISO 9241-210 : 2010 en la Construcción de Sistemas Interactivos Basados en Computadora. *Congreso Internacional de Investigación e Innovación En Ingeniería de Software - CONISOFT 2012*, 1–7.

Wanderley, F., Silva, A., Araujo, J., & Silveira, D. S. (2014). SnapMind: A Framework to Support Consistency and Validation of Model-Based Requirements in Agile Development. *IEEE 4th International Model-Driven Requirements Engineering Workshop - MoDRE 2014*, 47–56. https://doi.org/10.1109/MoDRE.2014.6890825

Ward, D., Rossi, M., Sullivan, B. P., & Pichika, H. V. (2018). The Metamorphosis of Systems Engineering Through the Evolution of Today's Standards. *4th IEEE International Symposium on Systems Engineering - ISSE 2018*, 1–8. https://doi.org/10.1109/SysEng.2018.8544426

Westfall, L. (2011). Las Fallas en la Ingeniería de Requisitos. *Ingenierías USBMed*, *2*(2), 40–47. https://doi.org/10.21500/20275846.257

Biographical Notes

Alex LLanque Flores. Systems Engineer from the National University of Moquegua. His research area is Requirements Engineering and process management. I work as a systems analyst in private entities. Specialist in Educational Informatics in mobile banking for financial institutions. Participate as an integrant in the software development team of the Provincial Municipality of Ilo (2013-2014).

Vaneza Flores Gutiérrez is a professor at the Professional School of Systems Engineering and Informatics at the National University of Moquegua (UNAM) in Peru, since 2008. Master in Business Management (2019), with concluded studies of Master in Systems Engineering (2008) and a diploma in University Teaching (2021). Initiating research activities in the line of artificial intelligence.

PhD student in Engineering at the Pontifical Catholic University of Peru (PUCP). He is a researcher and principal professor at the PUCP since 2000. He directed and was principal investigator of the ProCal-ProSer project phase II (2017-2019) and phase I (2013-2016) and a founding member of Software Engineering Research and Development Group at PUCP. He has published several scientific articles in journals and international congresses. Process evaluator using ISO/IEC 15504. Evaluator, auditor, consultant and trainer in models of Information Technology (MoProSoft, ISO/IEC 29110 VSE, CMMI, ISO/IEC 12207, ISO 9001, ISO/IEC 20000, CMMI, ISO/IEC 25000 series), formulator projects for competitive funds, IT solutions consultant.