

Hacia la construcción de un proceso de trabajo colaborativo para la elicitación de requisitos basado en entendimiento compartido

Valentina Vergara Realpe¹[0000-0002-1302-7398], Jesús-David Perea¹[0000-0002-4140-7582],
Vanessa Agredo Delgado^{1,2}[0000-0003-0870-6895], Pablo H. Ruiz^{1,2}[0000-0003-2098-2614] and
César A. Collazos²[0000-0002-7099-8131]

¹Corporación Universitaria Comfacauca - Unicomfacauca, Popayán, Colombia
{valentinavergara, jesusperera, vagredo,
pruiz}@unicomfacauca.edu.co

²Universidad del Cauca, Popayán, Colombia
ccollazo@unicauca.edu.co

Resumen. La elicitación de requisitos se considera como una actividad compleja y vital del proceso de desarrollo de software, en la cual se busca abstraer y comprender problemas y necesidades que se pueden resolver con la construcción de un producto software. Así como es conocida su importancia, sigue siendo una de las actividades más débiles del proceso, siendo esencialmente, una actividad humana en la cual se intercambian diferentes perspectivas, experiencias y conocimientos para recopilar y modelar lo que el software debe hacer. Además, en esta se conjugan una mezcla de estrategias, herramientas y colaboraciones entre personas, lo que hace que el entendimiento común y homogéneo de estos problemas y necesidades no se logre fácilmente. En consecuencia, este artículo presenta el avance inicial de la investigación que ha permitido articular y fundamentar una propuesta para la definición de un proceso de trabajo colaborativo, en la elicitación de requisitos basado en entendimiento compartido. Este proceso busca guiar y apoyar la construcción del entendimiento compartido para que la elicitación de los requisitos sea más confiable y por ende la construcción de producto software también.

Palabras clave. Entendimiento compartido, Elicitación de requisitos, Trabajo colaborativo, Proceso.

1 Planteamiento del Problema

La Ingeniería de Software es una disciplina que abarca diversas etapas del ciclo de vida del software como son: la elicitación de requisitos, análisis, diseño, implementación, pruebas y mantenimiento [1]. Los modelos del ciclo de vida del software pretenden abarcar todo el proceso del desarrollo del software, creando para cada etapa elementos de proceso que apoyan y describen la ejecución del proyecto software [2]. A través de

los años se ha podido evidenciar que la elicitación de los requisitos es la etapa más importante en un proyecto de desarrollo de software [3]. Esta etapa se utiliza para traducir las necesidades y deseos imprecisos e incompletos de los usuarios potenciales del software en especificaciones completas, precisas y formales. En esta etapa se busca abstraer y comprender problemas y necesidades que se pueden resolver con la construcción de un producto software. Lograr una correcta elicitación de requisitos es una actividad compleja y vital, para desarrollar software eficaz y reducir los errores en su construcción [3], puesto que establece el punto de inicio para la planeación, estimación de recursos y la elaboración de cronogramas que soportaran toda la gestión del proyecto software. Es conocido que los fracasos de la gran mayoría de los productos de software se basan en una mala elicitación, gestión, análisis y especificación de los requisitos, incrementando los costos, generando re-trabajo y así, excediendo los tiempos de desarrollo [4]. Pero, así como es conocida su importancia, sigue siendo una de las etapas más débiles del proceso, se trata, esencialmente, de una actividad humana en la cual se intercambian diferentes perspectivas, experiencias y conocimientos para recopilar, especificar y modelar lo que el software debe hacer [5]. Además, en esta etapa se conjugan una mezcla de estrategias, herramientas y colaboraciones entre personas, lo que hace que el entendimiento común y homogéneo de estos problemas y necesidades no se logre fácilmente, es decir el entendimiento compartido entre los participantes de esta etapa es difícil de construir [6]. En este sentido, el entendimiento compartido se refiere al grado en que los miembros de un equipo coinciden en los pasos de un proceso de trabajo, el significado de esos pasos, el orden, la relación de las actividades y su comunicación [7]. Realizar una correcta gestión de la comunicación en el proceso de elicitación, es uno de los principales elementos para su éxito, debido a que pueden existir diversos intereses con diferentes expectativas de comprensión e interpretación de los problemas y necesidades del contexto del producto software, que pueden producir desinformación que al no ser tratadas a tiempo, puede generar lagunas evidentes en la comunicación y entendimiento entre los usuarios y el equipo de desarrollo [8]. Además, en la elicitación se debe generar un entorno abierto y de colaboración, donde todo aquel que tenga algo que decir o aportar, pueda hacerlo libremente y sin sentirse coaccionado y sobre todo que todos entiendan lo mismo [9]. La elicitación de requisitos se basa en la interacción y la colaboración entre personas donde su base es el entendimiento mutuo, por lo tanto, al no existir, es otra de las razones de su fracaso [6].

El trabajo colaborativo requiere que los participantes que ejecutan una actividad compartan el proceso de creación de conocimiento [10], [11] lo que implica brindar espacios de interacción entre las personas para cumplir un objetivo basado en la utilización de diferentes elementos de colaboración (roles, tareas, estrategias, herramientas, métricas, etc.) que permitan incorporar las perspectivas de los participantes y de su trabajo en equipo [12], [13]. La elicitación de requisitos demanda trabajo colaborativo entre todos los participantes del proyecto, para conocer el dominio del problema, descubrir las necesidades reales y llegar a un acuerdo en cuanto a la visión y objetivos del producto a desarrollar, además de hablar el mismo lenguaje de los usuarios finales [14]. Por todo lo anterior la pregunta de investigación para esta propuesta es: *¿Cómo apoyar la construcción del entendimiento compartido*

en la elicitación de requisitos para el desarrollo de productos software? En consecuencia, este artículo presenta el avance inicial de la investigación que ha permitido articular y fundamentar una propuesta para la definición de una estrategia de trabajo colaborativo, en la elicitación de requisitos basada en la construcción de entendimiento compartido.

El resto de este documento se estructura de la siguiente manera: la sección 2 describe la justificación. La sección 3 presenta trabajos relacionados. La sección 4 define los objetivos de la propuesta. La sección 5 describe la metodología, y la sección 6 se describen las conclusiones preliminares del trabajo.

2 Justificación

El entendimiento compartido se refiere al grado en que los miembros de un equipo coinciden en los pasos de un proceso de trabajo, el significado de esos pasos, el orden, la relación de las actividades y su comunicación [7]. Por otro lado, la elicitación de requisitos demanda trabajo colaborativo [14], debido que es una actividad humana en la cual pueden participar diversas personas con pluralidad cultural, diferentes niveles de experiencia, diversidad de conocimientos, lo cual puede plantear desafíos en la comunicación y en la resolución de problemas conjunta, para lo que es necesario construir un entendimiento compartido para lograr los objetivos de la actividad [15]. Por lo tanto, en este trabajo de investigación se pretende definir una estrategia de elicitación de requisitos basada en el trabajo colaborativo y en la construcción de entendimiento compartido, que pretende brindar a la comunidad empresarial un enfoque de apoyo a la elicitación de requisitos la cual es fundamental para el éxito de los proyectos de desarrollo de software, y de esta forma brindarles elementos que puedan beneficiar la calidad de los productos software. Diferentes autores han propuesto técnicas y modelos para obtener requisitos, donde la mayoría describen estrategias en las cuales no se trabaja la construcción del entendimiento compartido [2], por lo tanto, se requiere ofrecer a la comunidad de investigación una estrategia novedosa para la elicitación de requisitos que incorpore elementos de trabajo colaborativo en combinación de estrategias de construcción de entendimiento compartido que beneficien el desarrollo de esta ardua actividad. Finalmente, a la comunidad académica, se pretende brindar un enfoque útil y adecuado al contexto de los laboratorios de ingeniería de software para que pueda ser considerado como parte de las metodologías de desarrollo de software utilizadas en este campo.

3 Trabajos Relacionados

Entendimiento compartido en ingeniería del software

En [16] se propone una teoría en las organizaciones de software basada en entendimiento compartido, donde se resalta que la coordinación y la comunicación son elementos esenciales y problemáticos. El papel, el valor y el uso del entendimiento compartido en la ingeniería de software se investiga en [17], mostrando una recopilación de prácticas, así como una hoja de ruta para mejorar el conocimiento y la

práctica en esta área. Por su parte en [18] se aborda la influencia de la distribución del equipo en el éxito del proyecto, utilizando la teoría de modelos mentales compartidos para la construcción y el mantenimiento del entendimiento compartido. En [19] se analiza la comunicación, mediante el entendimiento compartido, de los conceptos o relaciones subyacentes de un equipo multidisciplinar en el desarrollo una aplicación móvil. Específicamente, en la elicitación de requisitos en [20] se investiga el papel de los elementos cognitivos para mejorar la claridad de las historias de usuario, proponiendo un conjunto de elementos de escritura desde diferentes dominios con el fin de mitigar la ambigüedad y mejorar el entendimiento compartido. En [21] se hace un análisis sobre cómo la cultura afecta el entendimiento compartido en los ingenieros de requisitos y el equipo de desarrollo así como en la organización y el progreso del proyecto.

Trabajo colaborativo en ingeniería del software

En [22] se aplica la ingeniería de colaboración para derivar un módulo de proceso (thinkLet “MindMerger”) para apoyar sistemáticamente a grupos de trabajo heterogéneos en la construcción entendimiento compartido en procesos industriales. En [23] se describe una infraestructura de soporte a la ingeniería de software colaborativa, donde se consideran roles de la evaluación heurística, análisis de registros y visualización para cuantificar los beneficios de esta infraestructura. Por su parte en [24] se propone un modelo para la enseñanza de la ingeniería de software que busca experimentar los principales problemas del trabajo colaborativo en el desarrollo de productos software. Similarmente en [25] se muestran objetivos, métodos y herramientas que se utilizan en una clase de ingeniería de software global enfatizando en el trabajo colaborativo, además, esfuerzos para automatizar datos de medición participativa y prácticas utilizadas en equipos globales de desarrollo de software. En [10] se muestra SmartWiki, una herramienta colaborativa para ayudar enfrentar los desafíos de la ingeniería de requisitos distribuida, ayudando a mejorar la calidad de los requisitos y a perfeccionar las anotaciones para que sus especificaciones sean coherentes. Finalmente, en [26] se muestra una revisión sistemática de la literatura que hace un análisis de la ingeniería de software transcultural, identificando impactos potenciales de los factores culturales sobre los enfoques colaborativos y de comportamiento, además, identifica cómo las dimensiones culturales pueden afectar el trabajo colaborativo en la ingeniería de software.

De acuerdo con los trabajos relacionados, existen diferentes acercamientos en la literatura respecto al trabajo colaborativo e independientemente del entendimiento compartido, pero ninguno, define un proceso completo que involucre el trabajo colaborativo y la construcción del entendimiento compartido en la elicitación de requisitos desde que se hace su planeación, posteriormente su ejecución y validación, brindando el qué y cómo hacerlo, por lo tanto, esta pretende ser la mayor contribución de este trabajo de investigación.

4 Objetivos

Objetivo General

Definir un proceso¹ basado en trabajo colaborativo para la elicitación de requisitos en el desarrollo de productos software mediante la construcción de entendimiento compartido.

Objetivos Específicos

- Caracterizar e identificar de la literatura los diferentes elementos disponibles para el trabajo colaborativo en la elicitación de requisitos
- Identificar los elementos de entendimiento compartido que sean necesarios para elicitar requisitos en el desarrollo de productos software
- Construir un proceso de trabajo colaborativo para la elicitación de requisitos utilizando elementos de entendimiento compartido
- Validar la viabilidad del proceso propuesto mediante su aplicación en un contexto académico o industrial

5 Metodología

Este trabajo se llevará a cabo siguiendo la metodología de investigación-acción multi-ciclo con bifurcación que define tres ciclos: conceptual, metodológico y de evaluación [27]. Los ciclos y sus respectivas actividades se describen brevemente a continuación: *Ciclo Conceptual*: Planificación y ejecución de la identificación de características y elementos de los diferentes enfoques que servirán para la creación del proceso. Las actividades que están relacionadas con el primer y segundo objetivo son: a) Planeación del mapeo sistemático, se definen los objetivos, preguntas, criterios de inclusión y exclusión y las posibles bases de datos fuentes de información, b) Las cadenas de búsqueda son definidas y se ejecutan en las bases de datos de donde se extraerán los contenidos y datos relevantes, c) La información recolectada se analiza respecto a los diferentes elementos del trabajo colaborativo que apoyen la elicitación de requisitos, d) Identificar de acuerdo a la literatura los elementos del entendimiento compartido que puedan apoyar la elicitación de requisitos, e) Clasificar y/o definir los elementos del entendimiento compartido para ser integrados en la elicitación de requisitos.

Ciclo Metodológico: Se define la estructura base de los elementos que formarán parte del proceso propuesto. Las actividades relacionadas con el tercer objetivo son: a) Definir la estructura base de elementos que harán parte del proceso propuesto, b) Incorporar incrementalmente cada uno de los componentes que serán parte del proceso.

Ciclo de Evaluación: Se elegirá un mecanismo de evaluación mediante el cual se

¹ Proceso para este proyecto se considera como una secuencia de pasos organizados de forma lógica que se enfocan en lograr algún resultado específico.

validará el proceso propuesto. Las actividades relacionadas con el cuarto objetivo son:
a) Escoger un mecanismo de evaluación con el cual se validará el proceso propuesto,
b) Planear, diseñar y ejecutar el mecanismo de validación, c) Refinar el proceso teniendo en cuenta el análisis de los resultados obtenidos en su validación.

6 Conclusiones

Este artículo presenta el avance inicial de la investigación que ha permitido articular y fundamentar una propuesta para definir un proceso basado en trabajo colaborativo para la elicitación de requisitos mediante la construcción de entendimiento compartido, desde que se planea hasta que se valida, definiendo el qué y cómo hacerlo, considerando que según los trabajos relacionados, existen diferentes acercamientos en la literatura respecto al trabajo colaborativo y entendimiento compartido, de forma aislada, pero ninguno, define una estrategia que involucre ambos elementos y que brinde un paso a paso, por lo tanto, esta pretende ser la mayor contribución de este trabajo de investigación, buscando mediante la coordinación y colaboración entre personas que comparten conocimientos, técnicas y herramientas se pueda lograr entendimiento compartido en la elicitación de requisitos. Además, se identifica que la elicitación de requisitos es una actividad compleja que requiere la participación de diversas personas con pluralidad cultural, con ideas y conocimientos diferentes, donde es necesario comprender e interpretar adecuadamente las opiniones heterogéneas que surgen en la elicitación, estableciendo un entendimiento homogéneo.

Referencias

1. L. Garcés y L. M. Egas, «Evolución de las Metodologías de desarrollo de la Ingeniería de software en el proceso la Ingeniería de Sistemas Software,» *Revista Científica y Tecnológica UPSE*, vol. 1, n° 3, 2013.
2. I. Sommerville, *Ingeniería del software*, Madrid: Pearson educación, 2006.
3. A. Chakraborty, M. K. Baowaly, A. Arefin y A. N. Bahar, «The Role of Requirement Engineering in Software Development Life Cycle,» *Journal of Emerging Trends in Computing and Information Sciences*, vol. 3, n° 5, pp. 723-729, 2012.
4. L. R. Wong Portillo, «Contribuciones en el proceso de elicitación de requisitos: factores, actividades y cualidades,» Universidad Nacional Mayor de San Marcos, Lima, 2019.
5. M. Arias Chaves, «La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software,» *InterSedes: Revista de las Sedes Regionales*, vol. 6, n° 10, pp. 1-13, 2005.
6. M. Corvera Charaf, C. Rosenkranz y R. Holten, «The emergence of shared understanding: applying functional pragmatics to study the requirements development process,» *Information Systems Journal*, vol. 23, n° 2, pp. 115-135, 2013.
7. E. A. Christiane Bittner y J. M. Leimeister, «Creating Shared Understanding in Heterogeneous Work Groups: Why It Matters and How to Achieve It,» *Journal of*

- Management Information Systems*, vol. 31, nº 1, pp. 111-144, 2014.
8. L. Westfall, «Las fallas en la ingeniería de requisitos,» *Ingenierías USBMed*, vol. 2, nº 2, pp. 40-47, 2011.
 9. J. C. Ramírez-Leal, W. J. Giraldo-Orozco y R. Anaya-Hernández, «Una propuesta metodológica para mejorar la comunicación en ingeniería de requisitos,» *Revista EIA*, vol. 13, nº 26, pp. 121-139, 02 052016.
 10. P. Dillenbourg, M. J. Baker, A. Blaye y C. O'Malley, «The evolution of research on collaborative learning,» de *Learning in humans and machine: Towards an interdisciplinary learning science*, Oxford, Elsevier, 1996, pp. 189-211.
 11. J. Roschelle y S. D. Teasley, «The Construction of Shared Knowledge in Collaborative Problem Solving,» de *Computer supported collaborative learning*, Berlin, Springer, 1995, pp. 69-97.
 12. O. Kozar, «Towards Better Group Work: Seeing the Difference between Cooperation and Collaboration,» de *English Teaching Forum*, Washington, ERIC, 2010, pp. 16-23.
 13. J. Gómez, V. Agredo-Delgado y P. H. Ruiz, «Proposal for construction of an approach for defining feature models in Software Product Lines based on Collaborative Work,» *CEUR Workshop Proceedings*, vol. 747, 2020.
 14. A. Azadegan, K. N. Papamichail y P. Sampaio, «Applying collaborative process design to user requirements elicitation: A case study,» *Computers in Industry*, vol. 64, nº 7, pp. 798-812, 06 2013.
 15. J. Langan-Fox, J. Anglim y J. R. Wilson, «Mental models, team mental models, and performance: Process, development, and future directions,» *Human Factors and Ergonomics in Manufacturing & Service Industries*, vol. 14, nº 4, pp. 331-352, 2004.
 16. J. Aranda, *A Theory of Shared Understanding*, Toronto: University of Toronto, 2010.
 17. M. Glinz y S. A. Fricker, «On shared understanding in software engineering: an essay,» *Computer Science-Research and Development*, vol. 30, nº 3, pp. 363-376, 2014.
 18. H. Markus, C. Rosenkranz y R. Holten, «The Role of Shared Understanding in Distributed Scrum Development: An Empirical Analysis,» *Engineering, Computer Science*, 2016.
 19. A. Hoffmann, E. A. Christiane Bittner y J. M. Leimeister, «The emergence of mutual and shared understanding in the system development process,» de *International Working Conference on Requirements Engineering: Foundation for Software Quality*, Berlin, 2013.
 20. A. R. Amna, «The Use of Context-Free Grammar and Gamification to Reduce User Story Ambiguity and Raise Ambiguity Awareness: A Proposal,» de *CEUR Workshop Proceedings*, Belgium, 2020.
 21. Y. Hsieh, «Culture and Shared Understanding in Distributed Requirements Engineering,» de *IEEE International Conference on Global Software Engineering (ICGSE'06)*, 2006.
 22. E. A. Christiane Bittner y J. M. Leimeister, «Creating Shared Understanding in Heterogeneous Work Groups: Why It Matters and How to Achieve It,» *Journal of Management Information Systems*, vol. 31, nº 1, pp. 111-144, 2014.
 23. C. Cook y N. Churcher, «Modelling and Measuring Collaborative Software,»

- University of Canterbury, New Zealand, 2004.
24. M. I. Alfonso y F. Mora, «Learning software engineering with group work,» de *Proceedings 16th Conference on Software Engineering Education and Training*, Alicante, 2003.
 25. G. D. Thompson, D. Petkovic y S. Huang, «Teaching distributed collaborative development techniques in a software engineering class setting,» *Integrating FOSS into the Undergraduate Computing Curriculum*, San Francisco, 2009.
 26. F. Fazli y E. A. Christiane Bittner, «Cultural Influences on Collaborative Work in Software Engineering Teams,» de *Proceedings of the 50th Hawaii international conference on system sciences*, 2017.
 27. V. Lencinas, F. Nataloni, S. Hannover Saavedra, T. G. Villanueva, S. Lacolla, M. López y I. G. Zabczuk, «Una oportunidad para generar conocimiento desde la practica profesional de bibliotecarios y archiveros,» *Revista Prefacio* , vol. 1, nº 1, pp. 10-23, 2017.