

MODELO DE MÉTRICAS PARA APOYAR LA EVALUACIÓN DE LA DEUDA DE LA
DOCUMENTACIÓN EN PROYECTOS DE SOFTWARE ÁGIL



JUAN CARLOS NARVÁEZ NARVÁEZ

Tesis de Maestría en Computación

Director:

César Jesús Pardo Calvache, Ph.D. MSc.

Universidad del Cauca

Facultad de Ingeniería Electrónica y Telecomunicaciones

Departamento de Sistemas

Grupo de Investigación y Desarrollo en Tecnologías de la Información - GTI

Línea de investigación en Ingeniería de Software

Popayán, diciembre de 2023

JUAN CARLOS NARVÁEZ NARVÁEZ

MODELO DE MÉTRICAS PARA APOYAR LA
EVALUACIÓN DE LA DEUDA DE LA DOCUMENTACIÓN
EN PROYECTOS DE SOFTWARE ÁGIL

Tesis presentada a la Facultad de Ingeniería Electrónica y
Telecomunicaciones de la Universidad del Cauca para la
obtención del título de

Magister en:
Computación

Director:
César Jesús Pardo Calvache, Ph.D. MSc.

Popayán
2023

Dedicado a...

Mi amada esposa, Alma, por todo su amor, apoyo y comprensión incondicionales en cada etapa de esta aventura extrema.

A mi familia por ser uno de mis motivos más importantes para seguir adelante y no desfallecer.

A mi estimado colega y director César Pardo, por guiarme todo este tiempo para convertirme en investigador y docente.

Juan Carlos Narváez Narváez.

Nota de aceptación

Presidente del jurado

Jurado

Popayán, 2023

Agradecimientos

Culminar este proceso de formación, junto con esta investigación, han sido tareas titánicas, pero me ha permitido conocer personas increíbles que de una u otra manera han estado conmigo en las diferentes etapas de mi formación como magister en computación. De todos ellos he aprendido mucho y por eso quiero agradecerles a todos.

En primer lugar, a mi amada esposa, Alma Geny Muñoz, que con su amor, apoyo y comprensión me permitió salir adelante con valentía, motivándome a no rendirme jamás. A mis padres, Eucaris Narváez y Arcenio Narváez, que son mi modelo de vida y cuyas enseñanzas me han convertido en la persona soy, a mi hermana, Deicy Liliana Narváez, por su apoyo en todo este proceso, y a mi sobrino, Carlos Santiago Vargas Narváez, que muy pronto será parte de esta maravillosa familia y se ha convertido en una bendición para todos.

A mi estimado colega y director, el PhD. César Jesús Pardo Calvache, por guiarme en todo este proceso y quien le debo tantas reflexiones de vida que me han llevado a ver posibilidades más allá de mis capacidades. Al PhD. Raúl Iván Mazo Peña, de la universidad ENSTA Bretagne en Brest, Francia, por permitirme vivir una gran experiencia en este increíble país.

A todos los profesores que durante toda mi carrera me brindaron todos los conocimientos necesarios para culminar este proceso. También, quiero agradecer a todos y cada uno de mis compañeros con los que compartí grandiosos momentos a lo largo de toda mi formación; de cada uno me llevo lo mejor y sus enseñanzas, gracias por tantas alegrías, recochas y en algunos casos tristezas.

Finalmente quiero agradecer a la Universidad de Cauca y a la Maestría en Computación de la cual me siento orgulloso de pertenecer y en donde tuve la oportunidad de formarme como Magister en Computación, carrera a la que amo y con la que espero aportar cambios positivos a mi región.

Dedico este triunfo a todos ellos y en especial a mi familia.

Juan Carlos Narváez Narváez.
Diciembre de 2023.

Resumen estructurado

Antecedentes: La deuda de la documentación de software se refiere a la falta o insuficiencia de información detallada y actualizada acerca del código, diseño, y funcionamiento de un programa informático. Esta carencia puede acumularse debido a restricciones de tiempo, cambios rápidos en los requisitos o prioridades, y puede resultar en dificultades para comprender, mantener y mejorar el software a lo largo del tiempo. Abordar la deuda de documentación implica dedicar esfuerzos para crear y mantener una documentación clara y completa, lo que contribuye a la comprensión efectiva y al desarrollo sostenible del software.

Objetivos: Diseñar un modelo de métricas para medir la deuda de la documentación que se genera durante la ejecución de proyectos de desarrollo ágil de software mediante la definición de mecanismos de evaluación que integren características de transparencia sobre los atributos de la documentación en este tipo de proyectos.

Método(s): Se utilizaron los siguientes métodos de investigación: (i) el método de investigación-acción, (ii) enfoque GQM (Goal-Question-Metric), y (iii) el estudio de caso.

Resultados: (i) una revisión sistemática de la literatura para recolectar toda la información relevante sobre deuda de la documentación en el desarrollo ágil de software, (ii) una versión inicial de una ontología para organizar el conocimiento sobre la deuda de la documentación, (iii) una definición de las características y dimensiones de la documentación, así como un conjunto de 15 riesgos para documentación, (iv) un modelo de referencia compuesto por 39 métricas asociadas a las características de la documentación, las cuales fueron evaluadas mediante un estudio de caso, (v) una publicación en una revista indexada.

Conclusiones: La solución propuesta es una valiosa iniciativa para estructurar el conocimiento sobre deuda de la documentación en el desarrollo ágil de software a partir de una revisión sistemática y la caracterización de este tipo de deuda. Asimismo, el modelo de métricas diseñado permite realizar un diagnóstico sobre el nivel de deuda en la documentación de los proyectos software para diseñar estrategias de solución o mitigación.

Palabras clave: Desarrollo ágil de software, deuda de la documentación, deuda técnica, documentación de software, gestión de la deuda de la documentación.

Structured abstract

Background: Software documentation debt refers to the lack or insufficiency of detailed and updated information about the code, design, and operation of a computer program. This deficiency can accumulate due to time constraints, rapid changes in requirements or priorities, and can result in difficulties understanding, maintaining, and improving the software over time. Addressing documentation debt involves efforts to create and maintain clear and comprehensive documentation, contributing to effective understanding and sustainable software development.

Objectives: Design a metrics model to measure documentation debt generated during the execution of agile software development projects by defining evaluation mechanisms that integrate transparency features regarding the attributes of documentation in these types of projects.

Method(s): The following research methods were employed: (i) the action research method, (ii) the GQM (Goal-Question-Metric) approach, and (iii) the case study.

Results: (i) a systematic literature review to gather all relevant information about documentation debt in agile software development, (ii) an initial version of an ontology to organize knowledge about documentation debt, (iii) a definition of documentation characteristics and dimensions, along with a set of 15 documentation risks, (iv) a reference model consisting of 39 metrics associated with documentation characteristics, evaluated through a case study, and (v) a publication in a indexed journal.

Conclusions: The proposed solution is a valuable initiative to structure knowledge about debt documentation in agile software development through a systematic review and characterization of this type of debt. Likewise, the designed metrics model allows for a diagnosis of the level of debt in the documentation of software projects to design solution or mitigation strategies.

Keywords: Agile software development, documentation debt, technical debt, software documentation, documentation debt management.

Contenido

	Pág.
Lista de Figuras	X
Listado de Tablas	XI
Capítulo 1. Introducción	14
1.1. Problemática y justificación	14
1.1.1. Hipótesis	16
1.2. Objetivos	16
1.2.1. Objetivo general.....	16
1.2.2. Objetivos específicos (OE).....	16
1.3. Metodología	16
1.4. Resultados obtenidos	18
1.5. Organización del documento.....	18
Capítulo 2. Marco teórico y estado del arte	20
2.1. Marco teórico	20
2.1.1. Documentación de software.....	20
2.1.2. Definición de deuda y su presencia en los procesos.....	20
2.1.3. Deuda de la documentación.....	21
2.1.4. Modelo de referencia.....	21
2.1.5. Goal, Question, Metric (GQM).....	21
2.1.6. Métrica, medida, medición e indicador.....	22
2.1.7. Ontología	22
2.2. Trabajos relacionados.....	22
2.2.1. Protocolo de investigación	22
2.2.2. Ejecución de la búsqueda	26
2.2.3. Discusión	31
Capítulo 3. Caracterización de la documentación de software	33
3.1. Proceso de identificación de los elementos de la documentación	33
3.2. Elementos fundamentales en la documentación de software	34
3.2.1. Tipos de documentación	34
3.2.2. Roles que intervienen en la documentación	36
3.2.3. Herramientas para documentar.....	38
3.3. Dimensiones y características de la documentación de software	39
3.3.1. Descripción de las dimensiones y características	39
Capítulo 4. Hacia una ontología de la deuda de la documentación de software	46
4.1. Propósito	46

4.2.	Metodología de construcción	46
4.3.	Integración con otras ontologías.....	46
4.4.	Esquema gráfico	47
4.5.	Conceptos y relaciones de la ontología.....	48
4.6.	Discusión de algunos conceptos y sus relaciones	53
4.7.	Verificación de la ontología.....	54
4.8.	Conclusiones y trabajos futuros.....	55
Capítulo 5. Modelo de métricas para medir la deuda de la documentación		57
5.1.	Proceso de construcción.....	58
5.1.1.	Fase 1: Recolección de información	58
5.1.2.	Fase 2: Análisis de causas y efectos	58
5.1.3.	Fase 3: Definición de riesgos	59
5.1.4.	Fase 4: Construcción del modelo de métricas	61
Capítulo 6. Evaluación del modelo de métricas		88
6.1.	Fase 1: Diseño del estudio de caso	88
6.1.1.	Contexto y propósito	88
6.1.2.	Preguntas de investigación.....	89
6.2.	Fase 2: Preparación	89
6.2.1.	Diseñar los instrumentos de recopilación de datos	89
6.2.2.	Definir el protocolo	90
6.2.3.	Seleccionar el sujeto de estudio.....	90
6.2.4.	Establecer las condiciones	91
6.3.	Fase 3: Recolección de datos.....	91
6.3.1.	Ejecutar el protocolo	91
6.3.	Fase 4: Análisis de resultados	107
6.4.1.	Análisis cualitativo.....	107
6.4.2.	Análisis de la encuesta	110
6.4.3.	Análisis del esfuerzo y el costo	112
6.4.4.	Análisis de la validez	113
6.4.5.	Análisis de las limitaciones	114
Capítulo 7: Conclusiones y trabajos futuros		116
7.1.	Análisis de la pregunta, hipótesis y objetivos de investigación.....	116
7.1.1.	Respuesta a la pregunta e hipótesis de investigación.....	116
7.1.2.	Respuesta al Objetivo general	117
7.1.3.	Respuesta a los Objetivos específicos (OE).....	117
7.2.	Conclusiones.....	118
7.3.	Trabajos futuros	119
Bibliografía		121
Anexos		126
Anexo A: Informe técnico del modelo de métricas.....		126
Anexo B: Formulario de recolección de datos		134
Anexo C: Encuesta de evaluación		138

Lista de Figuras

	Pág.
Figura 2.1. Proceso para la revisión sistemática.....	23
Figura 2.2. Resumen del proceso de filtrado de los hallazgos.	26
Figura 2.3. Cantidad de artículos por año.	27
Figura 2.4. Clasificación de los artículos primarios.	29
Figura 2.5. Distribución de las propuestas evidenciadas.	30
Figura 3.1. Proceso de identificación de los elementos de la documentación.	33
Figura 3.2. Dimensiones y características de la documentación de software.	39
Figura 4.1. Ontología de la deuda de la documentación de software.....	47
Figura 4.2. Características de la documentación de software.	48
Figura 4.3. Extracto de la instanciación del proceso de documentación de requisitos.	54
Figura 4.4. Ejemplo de instanciación de la medición de una característica.	55
Figura 5.1. Resumen de los elementos de la propuesta.....	57
Figura 5.2. Proceso general para la construcción del modelo de métricas.	58
Figura 5.3. Estructura básica del modelo GQM. Tomado de [32].....	63
Figura 5.4. Ejemplo de grafo dirigido entre documentos.....	67
Figura 6.1. Proceso diseñado para realizar el estudio de caso.	88
Figura 6.2. Preferencias sobre el formato de la documentación.....	98
Figura 6.3. Valoración de cada dimensión de la documentación.....	107
Figura 6.4. Distribución porcentual de las respuestas de la encuesta.....	112
Figura 6.5. Distribución de las respuestas de la encuesta.....	112

Listado de Tablas

	Pág.
Tabla 2.1. Preguntas de investigación.	23
Tabla 2.2. Palabras y frases clave.....	24
Tabla 2.3. Cadena de búsqueda básica.....	24
Tabla 2.4. Criterios de evaluación de la calidad.....	25
Tabla 2.5. Categorías propuestas por Wieringa. Tomado de [40]......	27
Tabla 3.1. Artefactos de la documentación del usuario. Tomado de [45].	35
Tabla 3.2. Artefactos de la documentación del sistema. Tomado de [45]......	36
Tabla 4.1. Glosario de conceptos de la ontología.	48
Tabla 4.2. Relaciones en la ontología.	51
Tabla 5.1. Formato para la definición de riesgos.	59
Tabla 5.2. Listado de riesgos propuesto para el modelo de métricas.	59
Tabla 5.3. Glosario de conceptos importantes. Tomado de [75].	61
Tabla 5.4. Plantilla para la descripción de métricas.....	64
Tabla 5.5. Métrica de percepción de uniformidad.	64
Tabla 5.6. Métrica para medir el nivel de fidelidad con la plantilla.	65
Tabla 5.7. Métrica de percepción de adaptabilidad.	65
Tabla 5.8. Métrica de percepción de modularidad.	66
Tabla 5.9. Métrica para medir el índice de acoplamiento de la documentación.	67
Tabla 5.10. Métrica de percepción de legibilidad.....	67
Tabla 5.11. Índice Flesch-Kincaid para el idioma español. Tomado de [77].	68
Tabla 5.12. Métrica de percepción de concisión.....	69
Tabla 5.13. Métrica de percepción de claridad.	69
Tabla 5.14. Promedio de palabras por oración.	70
Tabla 5.15. Promedio de palabras por oración.	70
Tabla 5.16. Métrica de percepción de accesibilidad.	71
Tabla 5.17. Índice de solicitudes de acceso satisfactorias.	71
Tabla 5.18. Indicador de percepción de la confidencialidad de la documentación.	72
Tabla 5.19. Porcentaje de artefactos con marcas de confidencialidad.	72
Tabla 5.20. Definición de niveles de acceso a la documentación.	73
Tabla 5.21. Índice de documentación de libre acceso.	73
Tabla 5.22. Documentación extremadamente confidencial.....	74
Tabla 5.23. Conocimiento de implicaciones legales.	74
Tabla 5.24. Porcentaje de relaciones incorrectas.	75
Tabla 5.25. Densidad de relaciones entre la documentación.	75
Tabla 5.26. Indicador de percepción de precisión de la documentación.....	76

Tabla 5.27. Ejemplo del diseño de una matriz de trazabilidad.	77
Tabla 5.28. Cobertura de requisitos en la matriz de trazabilidad.	77
Tabla 5.29. Indicador de percepción de la portabilidad de la documentación.	78
Tabla 5.30. Documentación independiente de plataformas específicas.	79
Tabla 5.31. Documentación que requiere configuraciones especiales.	79
Tabla 5.32. Indicador de percepción de integridad de la documentación.	80
Tabla 5.33. Porcentaje de artefactos con un archivo de seguimiento.	80
Tabla 5.34. Percepción de actualidad de la documentación.	81
Tabla 5.35. Porcentaje de artefactos revisados.	81
Tabla 5.36. Porcentaje de problemas de inconsistencia.	82
Tabla 5.37. Cobertura de trazabilidad.	82
Tabla 5.38. Nivel de dependencia entre los documentos.	83
Tabla 5.39. Nivel de asociación entre los artefactos.	84
Tabla 5.40. Indicador de percepción de expansibilidad de la documentación.	84
Tabla 5.41. Porcentaje de uso de plantillas.	85
Tabla 5.42. Porcentaje de artefactos aprobados por la empresa.	85
Tabla 5.43. Artefactos revisados y aprobados.	86
Tabla 5.44. Documentos que tienen un historial de comentarios o revisiones.	87
Tabla 6.1. Pregunta de investigación principal del estudio de caso.	89
Tabla 6.2. Preguntas de investigación específicas del estudio de caso.	89
Tabla 6.3. Protocolo definido para el estudio de caso. Adaptado de [79].	90
Tabla 6.4. Actividades realizadas durante la planificación.	91
Tabla 6.5. Caracterización de los participantes del estudio de caso.	92
Tabla 6.6. Escala de implementación de las características de la documentación.	92
Tabla 6.7. Resumen de los resultados de medición obtenidos durante el estudio de caso. ...	93
Tabla 6.8. Resultados de aplicación de la métrica PDU.	93
Tabla 6.9. Resultados de aplicación de la métrica UPD.	94
Tabla 6.10. Resultados de aplicación de la métrica PAD.	94
Tabla 6.11. Resultados de aplicación de la métrica PMD.	95
Tabla 6.12. Resultados de aplicación de la métrica IAD.	95
Tabla 6.13. Resultados de aplicación de la métrica PLD.	96
Tabla 6.14. Resultados de aplicación de la métrica IFKE.	96
Tabla 6.15. Resultados de aplicación de la métrica PCD.	96
Tabla 6.16. Resultados de aplicación de la métrica PCLD.	97
Tabla 6.17. Resultados de aplicación de la métrica PPO.	97
Tabla 6.18. Resultados de aplicación de la métrica PFO.	97
Tabla 6.19. Resultados de aplicación de la métrica PACD.	98
Tabla 6.20. Resultados de aplicación de la métrica SAS.	98
Tabla 6.21. Resultados de aplicación de la métrica PConD.	99
Tabla 6.22. Resultados de aplicación de la métrica AMC.	99
Tabla 6.23. Resultados de aplicación de la métrica DNA.	99
Tabla 6.24. Resultados de aplicación de la métrica DLA.	100
Tabla 6.25. Resultados de aplicación de la métrica DEC.	100
Tabla 6.26. Resultados de aplicación de la métrica CIL.	100
Tabla 6.27. Resultados de aplicación de la métrica PRI.	100
Tabla 6.28. Resultados de aplicación de la métrica DRD.	101
Tabla 6.29. Resultados de aplicación de la métrica PPD.	101

Tabla 6.30. Resultados de aplicación de la métrica RMT.....	101
Tabla 6.31. Resultados de aplicación de la métrica PPRD.	102
Tabla 6.32. Resultados de aplicación de la métrica DIP.....	102
Tabla 6.33. Resultados de aplicación de la métrica DCE.....	102
Tabla 6.34. Resultados de aplicación de la métrica PID.....	103
Tabla 6.35. Resultados de aplicación de la métrica PAAS.....	103
Tabla 6.36. Resultados de aplicación de la métrica PAcD.	103
Tabla 6.37. Resultados de aplicación de la métrica PARev.	103
Tabla 6.38. Resultados de aplicación de la métrica PIRU.....	104
Tabla 6.39. Resultados de aplicación de la métrica CDT.	104
Tabla 6.40. Resultados de aplicación de la métrica NDD.....	104
Tabla 6.41. Resultados de aplicación de la métrica NAA.....	105
Tabla 6.42. Resultados de aplicación de la métrica PED.....	105
Tabla 6.43. Resultados de aplicación de la métrica PUP.....	105
Tabla 6.44. Resultados de aplicación de la métrica PAE.....	106
Tabla 6.45. Resultados de aplicación de la métrica ARA.....	106
Tabla 6.46. Resultados de aplicación de la métrica DHR.....	106
Tabla 6.47. Resumen gráfico de los resultados de la medición por dimensión.	108
Tabla 6.48. Clasificación de los resultados por característica.....	109
Tabla 6.49. Probabilidad de ocurrencia de los riesgos para la documentación de la empresa.	110
Tabla 6.50. Cuestionario de evaluación del estudio de caso con escala Likert.	111
Tabla 6.51. Preguntas abiertas de la encuesta.....	112
Tabla 6.52. Esfuerzo en horas aplicados.....	112

Capítulo 1. Introducción

En este capítulo se describe el problema de investigación abordado, los objetivos propuestos, la metodología utilizada para el desarrollo de la investigación y la estructura del resto del documento.

1.1. Problemática y justificación

Actualmente, la documentación de software es un activo de gran valor para las empresas que desarrollan software debido a su importancia estratégica en el éxito de los proyectos. Esto se refleja, por ejemplo, en permitir que los equipos tengan una comprensión holística del sistema a través de los artefactos que describen sus características en aspectos como: la especificación de sus requisitos, el diseño de sus interfaces, su arquitectura, entre otros [1], [2]. Además, permite que el software cumpla con las necesidades reales de los clientes y satisfaga sus expectativas de la mejor manera posible [3].

Sin embargo, la falta de documentación eficaz, confiable y pragmática que describa correctamente las características de un sistema software y, que además, permita su sana evolución, se ha convertido en un problema recurrente para las empresas [4]. Este problema se presenta en todo tipo de proyectos software, independientemente de su enfoque de desarrollo, y se origina, principalmente; por la ausencia, omisión o malas prácticas relacionadas con la creación, actualización y utilización de la documentación [5].

Los enfoques de desarrollo de software se componen de etapas, artefactos, roles y otros elementos de proceso que guían la construcción del software, ofreciendo ventajas y desventajas para su documentación [6]. Por un lado, los enfoques tradicionales son bastante rigurosos y formales en su documentación, donde la construcción del sistema solo iniciará cuando se tomen las decisiones finales sobre sus requisitos, arquitectura y diseño; pero carecen de flexibilidad ante los cambios que se puedan presentar en el futuro [7], [8]. Por otra parte, los enfoques ágiles buscan disminuir la rigurosidad en la documentación priorizando la entrega de valor continuo en poco tiempo [9]. Sin embargo, esto puede llevar a que ciertas actividades sean malinterpretadas y que no se realicen o se ejecuten mal deliberada o inadvertidamente como resultado de procesos subóptimos, poco claros, improvisados, mal diseñados, inmaduros y desgastantes que llevan a los equipos a no crear la documentación necesaria por la necesidad de cumplir con los plazos cortos de entrega [10], [11], [12].

Según [13], en general, a todas las actividades pendientes o mal ejecutadas durante el ciclo de vida del software se les denomina: *deuda técnica* y coincidentalmente una de las actividades que más evitan los equipos de desarrollo es documentar el software en un 60% aproximadamente [4]. Las razones son variadas, pero es posible destacar dos principalmente: la primera, es que se percibe como una actividad poco creativa que resta tiempo a actividades cuyo avance puede ser más visible, como el diseño, la arquitectura, la codificación, las pruebas, entre otros, y la segunda, que despierta poco interés por que puede tomar mucho tiempo [13]. En consecuencia, no documentar el software junto a la existencia de procesos

deficientes pueden desencadenar fenómenos bastante nocivos para el software como la *deuda de procesos* [14], la *deuda social* [15] y la *deuda de la documentación*. Específicamente, este último hace referencia a las deficiencias en la documentación de software, como escasez, falta de integridad, incoherencias, falta de accesibilidad, entre otras.

Sin embargo, a pesar que la deuda en cualquiera de sus variaciones puede traer beneficios a corto plazo, genera intereses si no se gestiona adecuadamente [9]. Estos intereses se reflejan en reprocesos, incumplimientos, sobrecostos, sobrecarga de trabajo, insatisfacción y baja productividad [13]. Un ejemplo de lo mencionado ocurre con un código mal diseñado cuya refactorización puede resultar costosa e incidir gravemente en otros bloques de código. Por el contrario, este escenario sería más conveniente si desde el principio del proyecto se documentan claramente buenas prácticas de diseño y codificación [16], [17].

Ahora bien, los enfoques ágiles enfrentan constantemente el desafío de lograr que la documentación sea precisa, comprensible y completa respecto a las funcionalidades del sistema que describe [13], lo que mejoraría la visibilidad y la transparencia en el acceso a la información [18], [19]. Por consiguiente, es necesario mantener un equilibrio entre flexibilizar los procesos y documentar el software, para evitar que la documentación sea escasa, incompleta, superficial o desactualizada, porque esto puede incrementar el costo de tareas futuras como el mantenimiento o la actualización [3]. Por otro lado, la documentación también describe los procesos de negocio que permiten la construcción del software; por lo tanto, una gestión adecuada de la documentación facilita que los procesos sean transparentes, es decir; accesibles, usables, informativos, entendibles y auditables [16, 20]. De esta forma, y según Martini et al. [12], los procesos también se deben documentar de forma clara y coherente para que tengan una sana evolución. En este sentido, es necesario que la documentación no sea pesada, estresante, rígida, voluminosa, compleja o difícil de actualizar [18].

A partir de los resultados obtenidos de una revisión sistemática de la literatura realizada en 2022 y posteriormente actualizada hasta el 30 de junio de 2023, enfocada en la deuda de la documentación en el desarrollo ágil de software, se evidenciaron definiciones conceptuales, causas y efectos [13], buenas prácticas para prevenir o pagar la deuda [21], y otros aportes importantes como un framework de gestión de la documentación, antipatrones de acumulación de la deuda, diferentes perspectivas de la documentación en enfoques ágiles [22], y olores o problemas en la documentación [23], lo que demuestra un interés reiterativo por parte de la comunidad científica; sin embargo, aún es necesaria una definición más precisa de este fenómeno, especialmente en el desarrollo ágil de software, y comprender cuales son las características más importantes que debe tener la documentación para que sea más transparente para todos los interesados.

Adicionalmente, autores como Rios et al. en [13] y Alves et al. en [17], destacan la importancia de construir herramientas que conduzcan hacia la automatización de la documentación de software, como por ejemplo, la herramienta denominada *Scrumconix* propuesta en [24]. Además, es importante determinar cómo la deuda de la documentación se relaciona con los demás tipos de deuda técnica y proponer mecanismos para su medición, ya que no fue posible evidenciar propuestas sólidas al respecto.

Por todo lo anterior y teniendo en cuenta que en la literatura actual no se evidencian suficientes iniciativas sobre deuda de la documentación en el desarrollo ágil de software, surge la siguiente pregunta de investigación: **¿Es posible reducir la deuda de la documentación generada durante el desarrollo ágil de software a través de la evaluación de la calidad de la documentación desde el punto de vista de la transparencia?**

1.1.1. Hipótesis

De acuerdo con la pregunta de investigación planteada se han establecido dos hipótesis de investigación. La primera, nombrada como H1, es de tipo causal, mientras que la segunda, nombrada como H0, será la hipótesis alternativa. A continuación, se presentan las hipótesis:

- **H1:** Mediante la evaluación de la calidad de la documentación desde el punto de vista de la transparencia se reduce la deuda de la documentación en los proyectos de desarrollo ágil de software.
- **H0:** Mediante la evaluación de la calidad de la documentación desde el punto de vista de la transparencia NO se reduce la deuda de la documentación en los proyectos de desarrollo ágil de software.

1.2. Objetivos¹

A continuación, se presentan los objetivos planteados de acuerdo con las palabras claves propuestas por la taxonomía de Bloom [25].

1.2.1. Objetivo general

Diseñar un modelo de métricas para medir la deuda de la documentación que se genera durante la ejecución de proyectos de desarrollo ágil de software mediante la definición de mecanismos de evaluación que integren características de transparencia sobre los atributos de la documentación en este tipo de proyectos.

1.2.2. Objetivos específicos (OE)

- OE 1.** Establecer los atributos fundamentales a tenerse en cuenta para caracterizar la deuda de la documentación en proyectos de desarrollo ágil de software a partir de un mapeo sistemático de la literatura.
- OE 2.** Construir un modelo de métricas asociadas a un subconjunto de las características identificadas en el OE1 a partir del enfoque Goal-Question-Metric (GQM) para apoyar la medición de la deuda en la documentación.
- OE 3.** Evaluar el modelo propuesto a través de su aplicación en un estudio de caso en un proyecto de desarrollo ágil de software para medir el nivel de deuda en su documentación.

1.3. Metodología

Este proyecto de investigación seguirá la metodología de investigación-acción propuesta por Checkland et al. [26] con cuatro ciclos lineales. Asimismo, la evaluación de la propuesta se llevará a cabo a través de un estudio de caso en una empresa de software, siguiendo el protocolo propuesto por Ridder [27].

Ciclo 1. Análisis conceptual: En este ciclo se realizó un mapeo sistemático de la literatura enfocado en la deuda de la documentación en el desarrollo ágil de software con el propósito de identificar las propuestas e iniciativas científicas existentes que representen un aporte

¹ Objetivos aprobados por el consejo de facultad de la FIET mediante la resolución 8.4.3-90.2/329 de 2021

importante para la definición del modelo. De esta forma, se ejecutaron las siguientes actividades:

- **Recolectar la información:** Se realizó un mapeo sistemático de la literatura en fuentes académicas confiables sobre estudios relacionados con la deuda de la documentación en el desarrollo ágil de software.
- **Sintetizar los estudios:** Se seleccionó la información más relevante aplicando criterios de clasificación y lectura crítica sobre los hallazgos para extraer las propuestas sobre deuda de la documentación en el desarrollo ágil de software.

Ciclo 2. Elaboración de la propuesta: En este ciclo se construirá el modelo de métricas basadas en transparencia asociadas a un subconjunto de características identificadas en la actividad 1.2 para cuantificar la deuda de la documentación en el desarrollo ágil de software. De esta forma, se ejecutaron las siguientes actividades:

- **Identificar los aspectos de la documentación a medir:** Se analizaron los conceptos, actividades, roles y prácticas propios de la documentación de software en proyectos ágiles que pueden ser medidos.
- **Categorizar los aspectos de la documentación a medir:** Se categorizaron los aspectos de la documentación y se relacionaron con las características de transparencia propuestas por [28].
- **Definir el conjunto de métricas:** Se definieron las métricas para cada aspecto identificado en la actividad 2.2 a través del enfoque GQM donde se integran las variables necesarias para realizar la medición de la deuda de la documentación.

Ciclo 3. Evaluación de la propuesta: La evaluación de la propuesta se realizó mediante un estudio de caso en un proyecto de desarrollo de software ágil en una empresa de la localidad. Para esto se realizaron las siguientes actividades:

- **Diseñar el estudio de caso:** Se definió el protocolo para el estudio de caso compuesto por: objetivos, insumos y forma de presentación a los involucrados.
- **Ejecutar el estudio de caso:** Se ejecutó el estudio de caso mediante un cuestionario, entrevistas y recopilación de datos siguiendo el protocolo definido en la actividad 3.1.
- **Analizar la evidencia:** Se desarrolló una estrategia de análisis estadístico de los resultados obtenidos en la ejecución del estudio de caso.
- **Desarrollar las conclusiones, recomendaciones e implicaciones:** Se documentaron de manera técnica todos los procedimientos ejecutados, los resultados obtenidos y los análisis realizados.

Ciclo 4. Documentar y socializar: Este ciclo se realizó de forma transversal durante todo el proyecto con el fin de organizar la documentación y socialización de los resultados obtenidos durante la ejecución de la investigación. Las actividades ejecutadas fueron las siguientes:

- **Elaborar la monografía:** Se elaboró la monografía teniendo en cuenta los anexos que resultaron durante el proceso de investigación.

- **Redacción de artículos científicos:** Se elaboraron artículos científicos para revistas reconocidas sobre los diferentes resultados obtenidos durante la investigación.
- **Divulgar los resultados:** Se presentaron y sustentaron los resultados obtenidos durante el desarrollo del proyecto de investigación.

1.4. Resultados obtenidos

Los resultados obtenidos con la realización de esta investigación son los siguientes:

- Monografía de trabajo de grado, el cual se estructura con los elementos definidos en la siguiente subsección.
- Modelo de métricas para soportar la medición del nivel de deuda en la documentación en el desarrollo de software ágil.
- Artículo científico a partir del mapeo sistemático de la literatura sobre deuda de la documentación en el desarrollo ágil de software publicado en Revista Científica de la Universidad Distrital Francisco José de Caldas.
- Estancia de investigación realizada en el Laboratoire des Sciences et Techniques de L'information, de la Communication et de la Connaissance (Lab-STICC) de la Universidad ENSTA Bretagne en Brest, Francia, bajo la tutoría del PhD. Raúl Mazo Peña donde se socializaron los resultados obtenidos hasta el momento y se ajustó el modelo.

1.5. Organización del documento

Los siguientes capítulos del presente documento se encuentran organizados así:

- **Capítulo 2.** Presenta el marco teórico donde se incluyen los conceptos teóricos básicos relacionados con la temática de esta investigación. De igual forma, contiene el estado del arte obtenido a partir de una revisión sistemática de la literatura donde se describen las propuestas existentes y las brechas en el área de la deuda de la documentación.
- **Capítulo 3.** Presenta la caracterización de los elementos más importantes de la documentación de software obtenidos a partir de la revisión sistemática de la literatura. Este capítulo contiene iniciativas sobre el proceso de documentación, los tipos de documentación, roles y herramientas para documentar. De igual forma, en este capítulo se describen las características de la documentación y su categorización en forma de dimensiones para estructurar su análisis.
- **Capítulo 4.** Presenta la definición de la ontología de la deuda de la documentación para organizar la información obtenida en el capítulo 3. Específicamente, se presenta la metodología de construcción, el esquema gráfico de la ontología, la descripción de los conceptos y relaciones que la componen, su validación mediante una instanciación y finalmente los trabajos futuros y conclusiones.
- **Capítulo 5.** Presenta la definición del modelo de métricas mediante el paradigma GQM asociadas a las características de la documentación descritas en el Capítulo 3.

- **Capítulo 6.** Presenta la evaluación de la propuesta a través de un estudio de caso para validar la capacidad del modelo para determinar el nivel de deuda de la documentación, indicando el procedimiento aplicado y el análisis estadístico correspondiente.
- **Capítulo 7.** Presenta las conclusiones obtenidas a partir del desarrollo de la investigación además de las lecciones aprendidas, los aportes de la investigación realizada y finalmente los posibles trabajos futuros a desarrollarse.
- **Capítulo 8.** Presenta las fuentes bibliográficas consultadas para llevar a cabo el trabajo de investigación.
- **Anexos.** Contiene toda la información adicional que apoya el documento final, los cuales se dividen en:
 - **Anexo A:** Informe técnico del modelo de métricas
 - **Anexo B:** Formulario de recolección de datos
 - **Anexo C:** Encuesta de evaluación

Capítulo 2. Marco teórico y estado del arte

Este capítulo presenta el marco teórico de la propuesta con los conceptos más relevantes de esta investigación, además, presenta los resultados de una revisión sistemática de la literatura realizada con el objetivo de proporcionar un estado del arte actualizado, donde se sintetizan los trabajos relacionados, así como propuestas e iniciativas en el área de la deuda de la documentación en el desarrollo ágil de software.

2.1. Marco teórico

A continuación, se presentan los conceptos más importantes que facilitarán la lectura del documento y la comprensión de la propuesta de investigación.

2.1.1. Documentación de software

La documentación de software se refiere al proceso de creación de los artefactos de documentación (documentos físicos, digitales, páginas web, aplicaciones móviles o wikis) en cada etapa del ciclo de vida del software (requisitos, diseño, codificación, pruebas, despliegue y mantenimiento), los cuales describen las funcionalidades del sistema y sirven de apoyo para la comunicación entre todos los participantes [29]. Estos artefactos se crean con base en las necesidades de los interesados [30] y se clasifican según su propósito en:

- **Documentación de procesos** [29]: donde se describen los procesos de desarrollo y mantenimiento que se encuentran alineados con los planes, cronogramas y estándares de calidad de la empresa. Este tipo de documentación está dirigida hacia los directivos y líderes de los equipos.
- **Documentación del producto** [30]: donde se describen las funcionalidades del software desde la perspectiva técnica, dirigida hacia el equipo de desarrollo, y la perspectiva de utilización del software dirigida hacia los usuarios finales.

Teniendo en cuenta lo anterior, la documentación debe ser capaz de proporcionar la información necesaria, útil y precisa para planificar, presupuestar y programar el proceso de desarrollo del software, así como la manera de administrarlo, mantenerlo y utilizarlo correctamente. Por el contrario, como lo afirman Chomal y Saini en [30], si la documentación es incorrecta, escasa o superficial puede llevar a una baja calidad del software y dificultar exponencialmente tareas futuras como el mantenimiento o la actualización de sus funcionalidades. Este hecho debería motivar a que las empresas presten el mismo nivel de atención a la documentación de sus proyectos como a la fabricación y comercialización.

2.1.2. Definición de deuda y su presencia en los procesos

Etimológicamente, la deuda se define como una obligación que se adquiere voluntariamente para obtener beneficios inmediatos, pero que genera intereses que deben ser retribuidos a través de compromisos de pago. Si esta definición se traslada al desarrollo de software, se le

denomina deuda técnica y se traduce en los beneficios obtenidos a corto plazo a partir de actividades subóptimas ejecutadas en cualquier etapa del ciclo de vida del software [15]. Sin embargo, admitir la deuda técnica puede traer diversos impactos negativos a mediano y largo plazo si no se gestionan adecuadamente los intereses que genera, los cuales se pueden manifestar como: trabajo duplicado, comunicación deficiente, disminución de la productividad, baja calidad del software, entre otros [14], [15].

Por lo tanto, como se describe en [15], la deuda es una metáfora utilizada para establecer una analogía con la deuda técnica para explicar sus características, sin embargo, también se utiliza para describir otros tipos de deuda en el desarrollo de software. Una de ellas es la deuda social relacionada con las actividades subóptimas enfocadas en las relaciones interpersonales entre los miembros de un equipo de trabajo, por ejemplo: al momento de comunicarse, relacionarse entre sí o con otros, trabajar en equipo, discutir propuestas, entre otros. Por otro lado, existe la deuda de procesos [12], la cual está relacionada con aquellos procesos mal diseñados, pesados, confusos y poco claros que, por consiguiente; son ignorados y mal ejecutados por los miembros de un equipo.

2.1.3. Deuda de la documentación

Como se comenta en [29], documentar es un proceso crítico en el ciclo de vida del software compuesto de actividades de documentación, roles que intervienen en la creación de un documento y artefactos de documentación específicos generados en cada etapa. Es así como el objetivo primordial de la documentación es apoyar la comunicación eficaz entre todos los interesados en el sistema. Para lograrlo, la documentación debe ser capaz de describir todas las características, funcionalidades y restricciones del sistema que sean necesarias para comprender completamente su funcionamiento.

Ahora bien, teniendo en cuenta lo mencionado y también la definición de deuda, es posible afirmar que la deuda de la documentación de software es un tipo de deuda técnica y de procesos relacionada con la mala o nula ejecución de las actividades de documentación del sistema [12]. Este tipo de deuda puede traer consecuencias negativas tales como: falta de claridad o confusión por información escasa o poco detallada o, por el contrario, excesivamente detallada; información ignorada por los equipos debido a que es demasiado sustancial y puede considerarse irrelevante; afectar la comunicación por no utilizar el lenguaje y elementos audio visuales adecuados para el consumidor de la información, entre otros.

2.1.4. Modelo de referencia

Un modelo de referencia es un modelo conceptual genérico que formaliza las actividades recomendadas para un dominio y con las cuales se puede abordar un problema específico, ofreciendo además la posibilidad de ser replicadas y adaptadas a situaciones similares [31].

2.1.5. Goal, Question, Metric (GQM)

Goal-Question-Metric (en español Objetivo-Pregunta-Métrica) es un enfoque que permite la creación de modelos de medición mediante el análisis de tres niveles de abstracción [32]. En el primer nivel, denominado conceptual, se establecen los objetivos o metas de medición que se desean alcanzar para todos los productos que se van a analizar (objetos de medición), teniendo en cuenta los procesos o actividades que permiten su creación y los recursos o elementos que requieren. En el segundo nivel, denominado operacional, se definen una serie de preguntas asociadas a cada objetivo para caracterizar la evaluación y el logro de los objetivos del nivel anterior. Finalmente, en el tercer nivel, denominado cuantitativo, se definen

y relacionan un conjunto de variables asociadas a las preguntas del nivel anterior para responderlas de manera cuantitativa y, por ende; evaluar el cumplimiento de los objetivos del primer nivel. Estas variables pueden ser objetivas, si dependen exclusivamente del objeto de medición y no del punto de vista de quien realiza la medición; o subjetivas, si dependen tanto del objeto de medición como del punto de vista de quien realiza la medición.

2.1.6. Métrica, medida, medición e indicador

Según la Asociación de Auditoría y Control de Sistemas de información ISACA, por sus siglas en inglés [33], una *métrica* es una entidad cuantificable que permite medir el logro de un objetivo. Se caracteriza por ser específica, medible, procesable, relevante y oportuna, y se considera completa si se define claramente la unidad de medida, la frecuencia de medición, el valor objetivo ideal (si se requiere) y la descripción de los procedimientos para medir y analizar los resultados. Por otro lado, una *medida* es una magnitud cualitativa o cuantitativa que se obtiene a partir de la aplicación de una o varias métricas en un proceso de medición. Estos dos conceptos convergen en lo que se conoce como *indicador* [34], el cual, a pesar de no tener una definición generalizada, la ISO/IEC/IEEE 15939:2017 lo define como una medida que permite cuantificar atributos específicos de un contexto o fenómeno de estudio. Un *indicador* se caracteriza por ser específico, explícito, disponible, claro, sólido y sensible al cambio, y son especialmente útiles para evaluar, hacer seguimiento y predecir tendencias hacia el cumplimiento de metas y objetivos puntuales [34].

2.1.7. Ontología

Como se explica en [35] una ontología es una representación abstracta y simplificada de un dominio específico a partir de una base de conocimiento denominada universo del discurso. El propósito de una ontología es estructurar el conocimiento acerca de un dominio, reducir la ambigüedad de conceptos y términos, y permitir que distintas personas o sistemas puedan intercambiar conocimiento entre sí [36]. Existen diferentes formas para representar una ontología, pero siempre involucran el uso de entidades llamadas objetos y como se relacionan entre sí para lograr una comprensión global del dominio.

2.2. Trabajos relacionados

Con el propósito de brindar al lector una visión más clara del estado del arte relacionado con la investigación descrita en este documento, en esta sección se presenta un resumen de la revisión sistemática de la literatura realizada en el año 2021 la cual fue publicada en [37] y posteriormente actualizada hasta el 30 de junio del año 2023. Es importante aclarar que la información presentada en esta sección corresponde a la actualización de la revisión sistemática y puede variar con respecto al artículo publicado.

2.2.1. Protocolo de investigación

A continuación, se presenta el protocolo seguido para la revisión sistemática de la literatura sobre deuda de la documentación en el desarrollo ágil de software a partir de los lineamientos de Petersen et al. [38] y Kitchenham et al. [39]. De esta forma, se ha diseñado el proceso de búsqueda presentado en la Figura 2.1 para recopilar, evaluar e interpretar toda la información disponible ya que hasta el momento no se evidencian otras revisiones similares o equivalentes sobre el tema. Por lo tanto, esta revisión es relevante debido a la importancia de la deuda de documentación en el desarrollo ágil de software y sería la primera en motivar a la comunidad científica a investigar en profundidad esta temática.

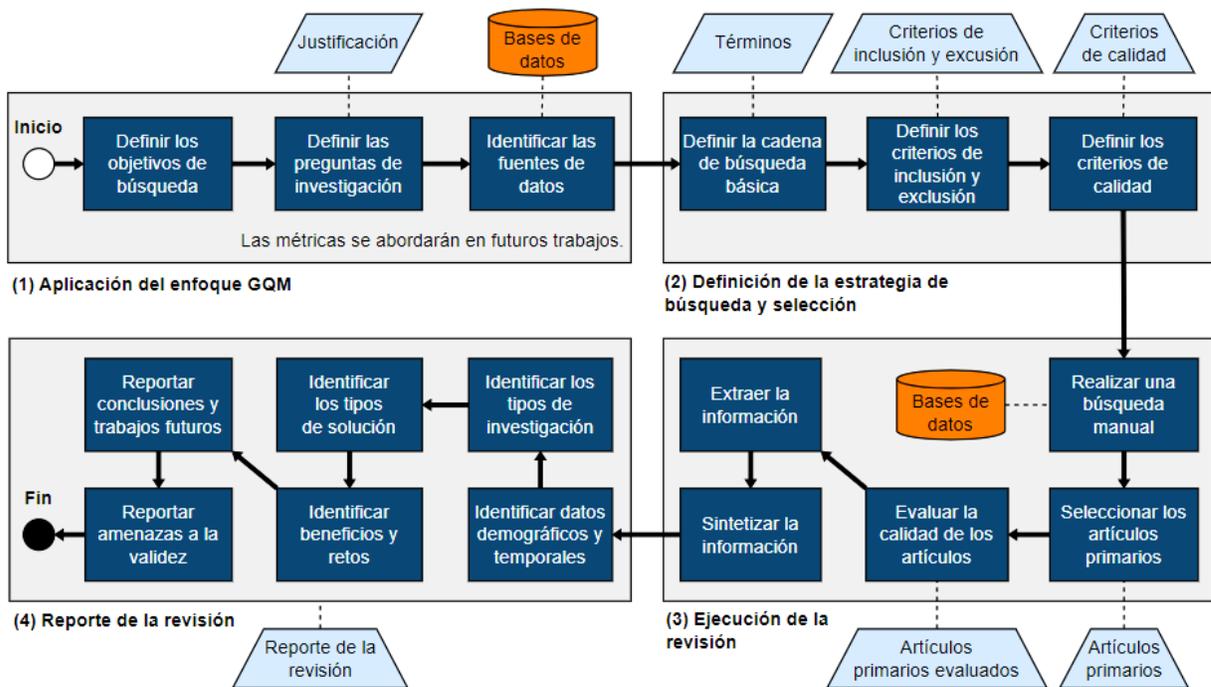


Figura 2.1. Proceso para la revisión sistemática.

2.2.1.1. Definición de las preguntas de investigación

Se definieron seis preguntas de investigación siguiendo el enfoque GQM relacionadas con los siguientes objetivos de búsqueda (ver Tabla 2.1):

- **OB1.** Categorizar las evidencias para dimensionar el interés científico e investigativo de la deuda de la documentación en el desarrollo ágil de software.
- **OB2.** Conocer las principales iniciativas científicas sobre deuda de la documentación en el desarrollo ágil de software para clasificarlas según el esquema propuesto en [40] y según su nivel de relevancia.
- **OB3.** Recopilar información sobre definiciones conceptuales, propuestas, validaciones, procesos y métodos de investigación para identificar el grado de desarrollo de las iniciativas.

Tabla 2.1. Preguntas de investigación.

No	Pregunta	Motivación	OB
P1	¿Cuál es la distribución temporal de los artículos primarios?	Presentar una tendencia macro de la literatura entre 2015 y junio de 2023	OB1
P2	¿Cuáles son las revistas y conferencias más importantes del área?	Descubrir los congresos y revistas científicas donde se publican la mayoría de los temas de nuestro interés.	OB1
P3	¿Qué regiones han mostrado mayor interés en el tema?	Conocer qué regiones han realizado mayor número de aportes científicos sobre el tema.	OB1
P4	¿Cuáles son los tipos de investigación desarrollada?	Clasificar los tipos de investigación evidenciados según el esquema de clasificación propuesto por Wieringa et al. [40].	OB2

P5	¿Cuál es la calidad de los artículos primarios?	Determinar la calidad de los artículos seleccionados de acuerdo con los criterios de calidad descritos en la Tabla 2.4.	OB2
P6	¿Qué aportes se han realizado hasta el momento?	Identificar el tipo de contribución en una o varias de las siguientes categorías: definición conceptual; causas, efectos, impactos y limitaciones; métodos o técnicas de evaluación; herramientas tecnológicas; validación en la industria; metodologías de documentación; otros	OB3
P7	¿Qué resultados se han obtenido con las propuestas presentadas?	Identificar el impacto que han tenido las evidencias con base en los resultados obtenidos de su validación en la industria del software.	OB3
P8	¿Qué beneficios y desafíos conlleva la investigación en el tema?	Identificar la incidencia de la deuda de la documentación en el desarrollo ágil de software.	OB3

2.2.1.2. Búsqueda de artículos relevantes

Inicialmente se hizo una revisión de algunos artículos relacionados con metodologías ágiles y documentación de software proporcionados por expertos para tener un primer acercamiento a los conceptos y términos utilizados en el contexto de la documentación de software y los problemas o desafíos que enfrenta. Posteriormente se identificaron un conjunto de las palabras y frases clave más frecuentes, eliminando duplicados y términos similares y unificando los términos relacionados. Como resultado de este proceso, en la Tabla 2.2 se presentan las palabras clave y frases encontradas.

Tabla 2.2. Palabras y frases clave.

Palabras y frases clave encontradas	Palabras y frases clave seleccionadas
Documentation, artifacts, problems, problem, issues, software, software development, software documentation, debt, technical, technical debt, documentation debt, deficit, agile software, agile development, agile approach, agile software development, agile, projects, agile software projects, causes, effects, process, process debt, documentation smells.	Documentation smells, documentation debt, process debt, agile software projects, agile software development, agile development, agile approach, documentation smells.

Con base en las palabras y frases clave presentadas en la Tabla 2.2 se definió la cadena de búsqueda básica presentada en la Tabla 2.3.

Tabla 2.3. Cadena de búsqueda básica.

Cadena de búsqueda básica
"documentation debt" OR "documentation smells" OR "process debt" AND ("agile software projects" OR "agile software development" OR "agile development" OR "agile approach")

A partir de la cadena de búsqueda básica presentada en la Tabla 2.3 se ejecutó la búsqueda de artículos en las siguientes bases de datos: IEEE Xplore, Science Direct, Scopus, Google Scholar, Springer Link, ACM y Web Of Science. Inicialmente, se consideró una ventana de tiempo de 2015 a 2021, ya que en el año 2015 es donde se evidencian las primeras investigaciones enfocadas en deuda de la documentación. Posteriormente la ventana se amplió hasta el 30 de junio de 2023 durante la actualización de la búsqueda.

2.2.1.3. Criterios de inclusión y exclusión

Los artículos relevantes fueron seleccionados mediante una revisión en tres pasos. Primero se revisó el título, segundo se revisó el resumen o abstract, la introducción y las conclusiones, y tercero se revisó el contenido. El propósito fue determinar su pertinencia de acuerdo a los siguientes criterios de inclusión (CI):

- **CI1.** Artículos cuyo tema principal sea la deuda de la documentación en el desarrollo ágil de software.
- **CI2.** Artículos cuyo tema se relacione con la documentación de software.
- **CI3.** Artículos que traten temas relacionados con la deuda técnica.
- **CI4.** Artículos que hayan sido publicados en revistas, congresos o conferencias de prestigio con revisión por pares.

Posteriormente, se seleccionaron los artículos primarios, descartando aquellos que cumplieran con al menos uno de los siguientes criterios de exclusión (CE):

- **CE1.** Artículos duplicados (considerando solo el más completo y reciente).
- **CE2.** Artículos donde se aborde superficialmente el tema de investigación.
- **CE3.** Artículos donde no se aborden problemas relacionados con la deuda de la documentación en el desarrollo ágil de software.
- **CE4.** Artículos de debate o disponibles en forma de presentaciones o resumen.

2.2.1.4. Evaluación de la calidad de los estudios primarios

Si bien *calidad* es un concepto abstracto carente de una definición estándar, es necesaria su medición, ya que así se minimizan los sesgos y se maximiza la validez interna y externa de una investigación [39]. Siguiendo esta premisa y con el propósito de mantener una rigurosidad científica en la investigación, se evaluó la calidad de los artículos primarios a partir de una adaptación del instrumento propuesto por Kitchenham y Charters [39] y de las categorías propuestas por Dybå y Dingsøyr [41]. De esta forma, en la Tabla 2.4 se presenta los doce criterios de calidad en forma de preguntas agrupados en las siguientes cuatro categorías.

- **Claridad:** Esta categoría permite conocer el nivel de relación de cada artículo primario con la deuda de la documentación en el desarrollo ágil de software, identificando claramente el contexto y el propósito de la investigación.
- **Credibilidad:** Esta categoría permite conocer el nivel en el que los resultados de la investigación son significativos, válidos y efectivos a través de la evaluación de los métodos científicos utilizados, la discusión de las limitaciones del proceso de investigación y el análisis de los resultados obtenidos [42].
- **Rigurosidad:** Esta categoría permite conocer el nivel de confiabilidad de los hallazgos a partir de la evaluación de los métodos de investigación, las herramientas de recolección de datos y los métodos de análisis utilizados.
- **Relevancia:** Esta categoría permite establecer el nivel de importancia y el impacto de los hallazgos en los ámbitos: (i) de la industria del software; mediante la aplicación de las propuestas y (ii) en el ámbito académico; brindando la posibilidad de continuar a futuro con investigaciones similares o afines [43].

Tabla 2.4. Criterios de evaluación de la calidad.

No.	Criterio de calidad	Categoría
1	¿El artículo se enfoca en investigar la deuda de la documentación en el desarrollo ágil de software?	Claridad

2	¿En el artículo describen claramente el problema de investigación abordado y los objetivos de investigación desarrollados?	
3	¿El artículo sigue un proceso de investigación estructurado y fundamentado?	Rigurosidad
4	¿El artículo proporciona una definición clara sobre deuda de la documentación?	
5	¿El artículo propone un conjunto de elementos a considerar para evaluar la deuda de la documentación en el desarrollo ágil de software?	
6	¿El artículo propone una forma de evaluar la deuda de la documentación en el desarrollo ágil de software?	
7	¿El artículo presenta una validación de su propuesta, explicando de forma clara y detallada los resultados alcanzados?	Relevancia
8	¿El artículo presenta claramente las contribuciones de la investigación hacia la industria o la academia?	
9	¿El artículo describe claramente los trabajos futuros o alternativas de investigación sobre deuda la documentación en el desarrollo ágil de software?	
10	¿El artículo describe claramente los métodos científicos utilizados?	Credibilidad
11	¿El artículo presenta un conjunto de conclusiones coherentes mediante el análisis crítico de los resultados obtenidos?	
12	¿El artículo describe claramente las limitaciones y los sesgos de la propuesta?	

Por lo tanto, se evaluaron los artículos primarios con respecto a cada criterio de calidad presentados en la Tabla 2.4 utilizando una puntuación entre -12 y +12 obtenida a partir de la suma de tres valores posibles: -1 (si no cumple el criterio), 0 (si lo cumple parcialmente) y 1 (si cumple el criterio). Esto permite determinar el nivel de calidad de cada artículo de acuerdo con las categorías y los criterios definidos, así como su pertinencia para la investigación sobre deuda de la documentación en el desarrollo ágil de software. Sin embargo, una baja calificación no fue motivo de exclusión ya que solo se busca cuantificar esos dos aspectos.

2.2.2. Ejecución de la búsqueda

En la Figura 2.2. se presenta el resumen de los artículos encontrados luego de ejecutar la cadena de búsqueda en las bases de datos seleccionadas. Como puede apreciarse, se encontraron 482 artículos, de los cuales se seleccionaron 176 como relevantes luego de aplicar sobre ellos los criterios de inclusión. Posteriormente, se descartaron 115 artículos debido a que aparecieron en más de una base de datos. Finalmente, a los artículos restantes se les aplicó los criterios de exclusión para seleccionar 27 artículos como primarios.

Además, se incluyeron otros 11 artículos en el grupo de artículos primarios como resultado de dos etapas de búsqueda adicionales; la primera, una revisión hacia atrás (backward snowballing) [44] sobre los artículos primarios, obteniendo 4 artículos y; la segunda, una actualización de la búsqueda ampliando la venta de tiempo hasta el 30 de junio, obteniendo 7 artículos. Como resultado, se dispone de un grupo de 38 artículos primarios que pueden ser consultados en el siguiente link: <https://bit.ly/3WAM4NT>.

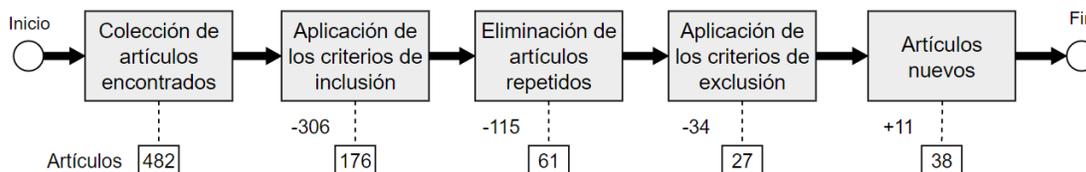


Figura 2.2. Resumen del proceso de filtrado de los hallazgos.

2.2.2.1. P1: ¿Cuál es la distribución temporal de los artículos primarios?

De acuerdo con los resultados, se aprecia un interés variable por parte de la comunidad científica en la deuda de la documentación. Como se muestra en la Figura 2.3, los años de

mayor publicación fueron 2016, 2018 y 2020 con 8, 7 y 8 publicaciones respectivamente, mientras que, en los años 2015, 2017, 2019, 2021, 2022 y 2023 las publicaciones se reducen en un 50%. Particularmente, el 51,61% de las publicaciones estudian la deuda de la documentación y su relación con la deuda técnica, el 22,58% menciona aspectos de la deuda de la documentación, pero en enfoques tradicionales y el 12,9% proponen formas de gestionar este tipo de deuda en enfoques ágiles.

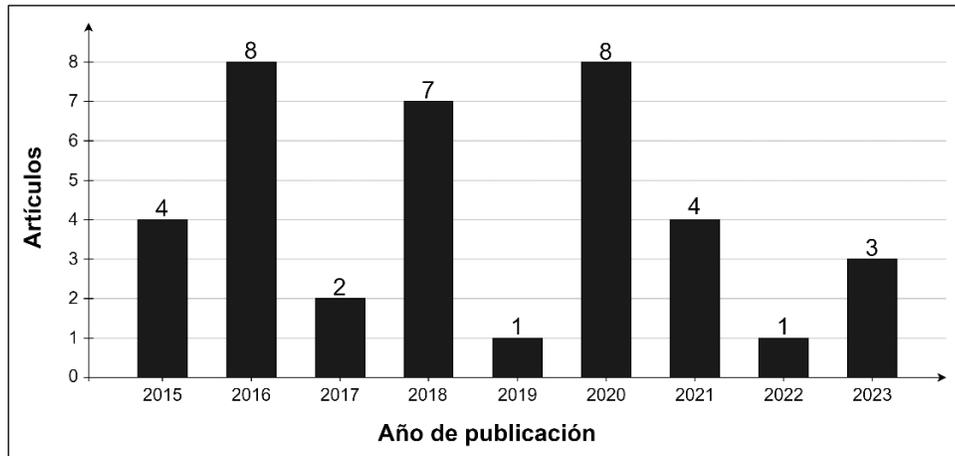


Figura 2.3. Cantidad de artículos por año.

2.2.2.2. P2: ¿Cuáles son las revistas y conferencias más importante en el área?

Los resultados de la búsqueda muestran que 17 artículos fueron publicados en revistas mientras que 21 fueron presentados en conferencias. Específicamente, el 9.3% de los estudios, a pesar de que no se enfocan directamente con deuda de la documentación, se han presentado en ventos exclusivos sobre Deuda Técnica. Por otro lado, el 18.75% de los estudios han sido presentados en eventos más amplios donde se aborda la deuda de la documentación en el desarrollo ágil de software, junto a la deuda de procesos y deuda social. El resto de los estudios han sido presentados en revistas y eventos de genéricos donde se abordan diversas áreas de conocimiento en el campo de la ingeniería de software, entre ellos: el desarrollo ágil de software y la documentación de software.

2.2.2.3. P3: ¿Qué regiones han mostrado mayor interés en el tema?

Los resultados de la búsqueda muestran que la distribución de los artículos primarios por la región donde fueron publicados tiene la siguiente distribución: América del Norte es la región donde se realiza la mayor contribución científica con 22 de las publicaciones relacionadas con deuda de la documentación, seguido de Europa con 15 y Asia con 1.

2.2.2.4. P4: ¿Cuáles son los tipos de investigación desarrollada?

El esquema propuesto por Wieringa et al. [40] permitió categorizar los artículos primarios de acuerdo con la información presentada en cada uno. De esta forma, en la Tabla 2.5 se presenta las categorías que se utilizaron en la clasificación junto a su definición.

Tabla 2.5. Categorías propuestas por Wieringa. Tomado de [40].

Categoría	Definición
Artículos de validación	Son artículos donde se investigan las propiedades de una propuesta rigurosa y metodológicamente sólida pero que aún no se implementa. Hacen parte de esta clasificación:

	experimentos, simulaciones, creación de prototipos, análisis matemáticos, prueba matemática de propiedades, revisiones de la literatura y estudios exploratorios.
Artículos de evaluación	Son artículos donde el método de investigación es novedoso, sólido y capaz de generar nuevo conocimiento a partir de su aplicación en la práctica. Hacen parte de esta clasificación: estudios de caso, estudios de campo, encuestas, entrevistas, pruebas conceptuales aplicando matemáticas y la lógica.
Propuestas de solución	Son artículos donde se propone una solución, pero con una validación parcial, caracterizada por ser novedosa o una mejora significativa a una solución existente. Hacen parte de esta clasificación: las pruebas de concepto.
Artículos filosóficos	Son artículos con juicios de valor por parte de los autores con base en revisiones previas de la temática. Hacen parte de esta clasificación: las taxonomías y los artículos de reflexión.
Artículos de opinión	Son artículos donde se describen opiniones a partir de juicios de valor. Hacen parte de esta clasificación los documentos personales sin investigación estructurada.
Artículos de experiencia personal	Son artículos anecdóticos centrados en la experiencia personal del autor donde se consignan lecciones aprendidas y que proviene, por lo general, de profesionales de la industria o de investigadores que llevan algo a la práctica sin discutir los métodos de investigación utilizados. Hace parte de esta clasificación todos los documentos personales sin investigación estructurada.

Como se muestra en la Figura 2.4, 18 artículos se clasifican en la categoría de *validación* debido a que se evidencian: (i) seis revisiones de la literatura, de las cuales cuatro recopilan información sobre deuda técnica y deuda de la documentación en proyectos de software ágil (A2, A6, A12, A32) y dos recopilan información de la deuda técnica en general (A19, A31), y (ii) doce estudios exploratorios, de los cuales; uno explora el impacto de los requerimientos ágiles en la deuda de la documentación (A4) y once exploran los beneficios, desventajas y desafíos de documentar el software (A10, A13, A17, A20, A21, A22, A23, A25, A27, A35, A36).

Por otro lado, 15 artículos se clasifican en la categoría de *evaluación* donde se evidencian seis estudios de caso, de los cuales tres se enfocan en estudiar cual es la documentación realmente importante en proyectos de software ágil (A8, A28, A33) y tres se enfocan en los factores que llevan a la deuda técnica y a la priorización de lo que se debe documentar (A11, A18, A26). Asimismo, se encontraron ocho artículos que describen encuestas, entrevistas y observaciones en la industria sobre las causas y efectos de la deuda de la documentación en proyectos de software ágil (A1, A5, A15, A16, A29, A34, A37, A38). También, fue posible encontrar una propuesta de solución enfocada en la automatización de la documentación de software en el enfoque ágil Scrum (A9). Por último, para las demás categorías no fue posible evidenciar propuestas, estas son: Artículos filosóficos y Artículos de opinión.

Adicionalmente, en A29 se presenta el concepto de *olores en la documentación* para describir malos estilos de documentación que no necesariamente producen documentación incorrecta, pero si dificultan su comprensión y uso adecuados, en especial para reutilizar funcionalidades del sistema en el futuro. Específicamente se presentan cinco olores: (i) documentación inflada, (ii) exceso de información estructural, (iii) documentación errada, (iv) documentación fragmentada, y (v) documentación con muy pocos detalles o Lazy. Además, con el artículo A30 donde se presentan técnicas para detectar olores en la documentación del código fuente.

Asimismo, fue posible observar una combinación de categorías en el resto de los artículos. La primera combinación encontrada fue: *evaluación y validación* a través de dos estudios basados en entrevistas, observaciones y estudios exploratorios donde se relaciona a la deuda de la documentación con la deuda de procesos, la deuda técnica y la deuda social (A14, A37), y el diseño de una solución a partir de estudios exploratorios para automatizar la creación de la documentación (A24). La segunda combinación encontrada fue: *evaluación y solución* mediante el hallazgo de un enfoque basado en herramientas para prevenir la deuda de la documentación, el cual fue evaluado en un estudio de caso (A3). Finalmente, la tercera

combinación fue: *solución y experiencia personal* mediante una propuesta de solución para prevenir la deuda de la documentación que permitió recoger una serie de experiencias y lecciones aprendidas con su uso en la industria del software (A7). Finalmente, no se evidenciaron artículos filosóficos ni de opinión.

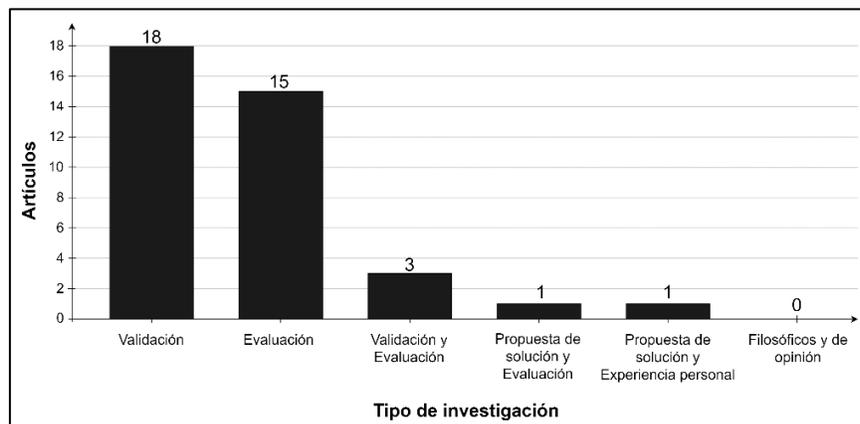


Figura 2.4. Clasificación de los artículos primarios.

2.2.2.5. P5: ¿Cuál es la calidad de los artículos primarios?

De acuerdo a los resultados el 15,79% de los artículos lograron una valoración de 3 puntos, el 13,16% lograron 6 puntos, el 10,53% lograron 2 puntos, el 5,26% lograron 7 puntos. Los detalles de evaluación se pueden consultar en: <https://bit.ly/3lfTjge>.

Con base en los criterios de calidad definidos en la Tabla 2.4, fue posible calificar el nivel de calidad de los artículos primarios evaluando minuciosamente aspectos como: el foco de la investigación, la claridad del problema, claridad de los objetivos, la metodología utilizada y la claridad de los resultados que se presentaban. Toda la información descrita en esta sección puede ser consultada en: <https://bit.ly/3lfTjge>.

Como resultado, la categoría de *claridad* puede ser calificada entre -2 y 2 debido a que se evalúan dos criterios de calidad para este ítem. De esta forma, el 5.3% (2 artículos) lograron una calificación de 2, el 63.2% (24 artículos) lograron un puntaje de 1, el 26.3% (10 artículos) lograron un puntaje de 0 y el 5.3% (2 artículos) lograron un puntaje de -1. Esto se debe a que en la mayoría de los artículos se describe claramente el problema y los objetivos de investigación, pero se enfocan en estudiar la deuda técnica de forma general, incluyendo a la deuda de la documentación como una temática adicional. Por lo tanto, el promedio de calificación de *claridad* de los artículos primarios es de 0,5.

Por otro lado, la categoría de *rigurosidad* puede ser calificada entre -4 y 4 debido a que se evalúan cuatro criterios de calidad para este ítem. De esta forma, el 2,6% (1 artículo) logro un puntaje de 3, el 21,1% (8 artículos) lograron un puntaje de 2, el 13,2% (5 artículos) lograron un puntaje de 1, el 21,1% (8 artículos) lograron un puntaje de 0, el 18,4% (7 artículos) lograron un puntaje de -2 y el 5,3% (2 artículos) lograron un puntaje de -3. Esta diferencia se debe a que en 30 artículos no fue posible evidenciar una forma de evaluar la deuda de la documentación y en 18 artículos no se evidenciaron propuestas claras de características en la documentación que pudieran ser evaluadas. Sin embargo, fue posible evidenciar una definición clara sobre deuda de la documentación en más del 50% de los artículos y el 90% de los artículos utilizaron una metodología de investigación sólida. Por lo tanto, el promedio de calificación de *rigurosidad* de los artículos primarios es 0.

Sobre la categoría de *relevancia* puede ser calificada entre -3 y 3 debido a que se evalúan tres criterios de calidad. De esta forma, el 18,4% (7 artículos) lograron una calificación de 3, el 21,1% (8 artículos) obtuvieron un puntaje de 2, el 31,6% (12 artículos) se calificaron con 1, el 18,4% (7 artículos) se calificaron con 0, el 2,6% (1 artículo) se calificó con -2 y 5,3% (2 artículos) se calificaron con -3. Esta diferencia se debe a que en 23 artículos se presentó una validación incompleta de su propuesta o no la realizaron, manifestando que será un trabajo a futuro. Además, en 29 artículos se expone de manera completa o parcial la contribución académica o industrial de la propuesta. Por lo tanto, el promedio de *relevancia* es 0.

Por último, la categoría de *credibilidad*, puede ser calificada entre -3 y 3 debido a que se evalúan tres criterios de calidad. De esta forma, el 15,8% (6 artículos) obtuvieron una calificación de 3, al igual que otros 6 artículos que obtuvieron el mismo puntaje. Algo similar ocurre con el 26,3% (10 artículos) que obtuvieron un puntaje de 0, el 10,5% (4 artículos) se calificaron con -1 y el 5,3% (2 artículos) se calificaron con -2. Esto se debe a que la mayoría de los artículos primarios describen claramente los métodos científicos que utilizan, la coherencia crítica de las conclusiones y presentan claramente las limitaciones de sus propuestas. Por lo tanto, el promedio de *credibilidad* es de 0,5.

2.2.2.6. P6: ¿Qué tipos de soluciones se han propuesto?

De acuerdo con los resultados presentados en la Figura 2.4, se observa que el 38,71% de los artículos proponen definiciones conceptuales que ayudan a comprender, estructurar y estandarizar el conocimiento sobre la deuda de la documentación. Por otro lado, en el 45,16% de los artículos se identifican causas, efectos, limitaciones e impactos que de este tipo de deuda. Específicamente, se identificaron 45 causas y 42 efectos como, código incorrecto, baja calidad del software, retrasos, entre otros. Los detalles de lo mencionado se pueden consultar en los siguientes links: <https://bit.ly/3WBY5mz>, <https://bit.ly/3WZGYdU>.

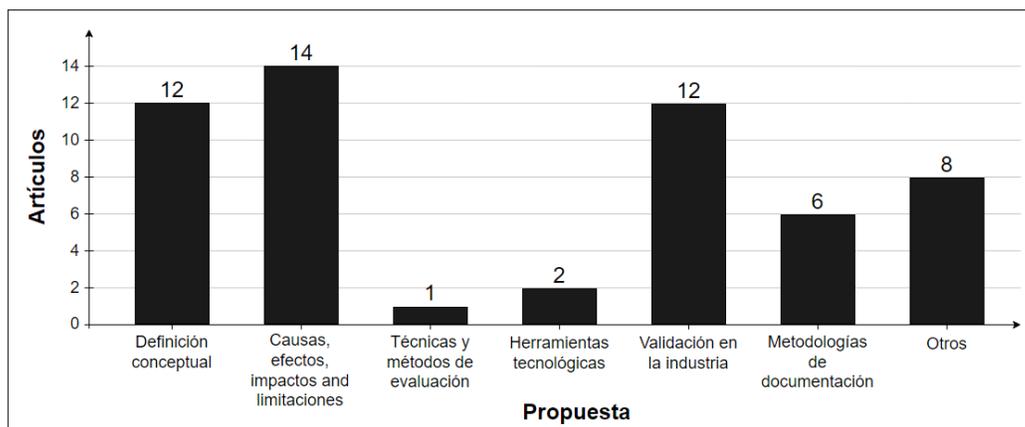


Figura 2.5. Distribución de las propuestas evidenciadas.

Por otro lado, se identificaron 54 buenas prácticas, 47 para prevenir la deuda y 7 para el pago de la deuda, a través de estrategias como: comentar el código, definir procesos claros de documentación, documentar requisitos funcionales y no funcionales, entre otros. El detalle se puede consultar en: <https://bit.ly/3Z575Ck>.

2.2.2.7. P7: ¿Qué resultados se han obtenido con las propuestas presentadas?

Además de lo mencionado anteriormente, la principal evidencia se relaciona con la definición de deuda de la documentación, asociándola con documentación faltante, inconsistente,

desactualizada o incompleta. Adicionalmente, se afirma que la deuda de la documentación se produce por actividades de documentación pendientes o mal ejecutadas en cualquier etapa del ciclo de vida del software, lo cual conduce a un sobre esfuerzo para mantener o actualizar el sistema. En los enfoques ágiles particularmente, este tipo de deuda puede originarse por una mala interpretación o implementación de los principios ágiles, llevando a que la documentación de baja calidad o no exista. Sin embargo, los estudios concluyen que una interpretación correcta de los principios ágiles permitiría crear documentación de alto valor, precisa, comprensible y eficaz para describir adecuadamente un sistema software.

2.2.2.8. P8: ¿Qué beneficios y desafíos conlleva la investigación en el tema?

Inicialmente, se observó que la deuda de la documentación puede beneficiar a las empresas de software en el corto plazo por que podrían reducir los tiempos de entrega de sus productos, sin embargo, este beneficio inmediato puede afectar a la empresa en el futuro debido a los sobre esfuerzos que podrían darse en la actualización del sistema. Por lo tanto, la existencia progresiva de malas prácticas de documentación puede llevar a niveles de acumulación de deuda que pueden ser muy difíciles de solucionar. En ese sentido, el concepto de deuda de la documentación es relativamente reciente y hasta el momento no ha sido completamente entendido ni definido. Por esta razón es necesario profundizar en su definición e investigar nuevas formas o prácticas de documentación para reducir la presencia de la deuda en la documentación, sobre todo en escenarios tan complejos como los enfoques ágiles o el desarrollo global de software.

Por otro lado, si la documentación es escasa o inadecuada puede afectar el proceso de gestión del cambio y la capacidad del sistema para evolucionar o escalar en el futuro. Esto es especialmente crítico en el desarrollo ágil de software ya que una documentación correcta es un factor de éxito muy importante en los proyectos software, pero al mismo tiempo es un desafío muy grande lograr un equilibrio entre documentar solo aquello que es realmente necesario, útil para describir correctamente un sistema y no perder de vista los principios ágiles que priorizan la entrega de valor por encima de la documentación excesiva.

2.2.3. Discusión

En esta sección se presenta un análisis a los resultados de la revisión sistemática con el propósito de identificar oportunidades de mejora sobre las propuestas.

2.2.3.1. Consideraciones principales

Fue posible evidenciar un interés en la comunidad científica en los diferentes tipos de deuda técnica, pero la deuda de la documentación con menor frecuencia. Es posible que esto se deba a que no ha recibido la atención suficiente por los investigadores o la industria del software, ya documentar no es una actividad que se considere como algo crítico. Sin embargo, es un área nueva que poco a poco empieza a generar interés por sus implicaciones económicas y de gestión en las empresas, además de su relación directa con otros tipos de deuda como la social o de procesos.

La deuda es una metáfora que consta de: *deuda* o actividades incompletas o mal ejecutadas, *intereses* o impactos negativos sobre el software, y el *costo* de solucionar la deuda. Estos tres aspectos pueden presentarse por una mala interpretación de los principios ágiles que llevan a procesos deficientes en el desarrollo de software, generando diferentes tipos de deuda como la deuda de la documentación. Asimismo, la documentación permite una relación recíproca, ya que si los procesos no se documentan correctamente se contribuye a su deficiencia.

Los enfoques ágiles son una alternativa bastante llamativa en el desarrollo de software debido a que son menos rigurosos en la documentación, pero exigen una profunda evolución técnica y humana en los equipos de trabajo. Por eso, es necesario desarrollar y promover habilidades blandas relacionadas con la documentación asertiva para tomar decisiones, trabajo colaborativo, análisis crítico, auto organización y capacidad de negociación. Por lo general los profesionales con perfiles técnicos muy elevados no suelen prestar la suficiente atención a estas habilidades, aun cuando contribuyan a un crecimiento profesional.

Finalmente, es importante entender que la deuda, a pesar de traer beneficios inmediatos, se debe planificar la solución de los problemas que pueda generar en el futuro, de lo contrario, las empresas estarían expuestas a sobre costos. Sin embargo, esto dependerá del nivel de deuda que esté dispuesta a tolerar, por lo tanto, las empresas deben tener muy claro el umbral de deuda en su documentación mediante mecanismos capaces de medirla y monitorearla.

2.2.3.2. Limitaciones de la revisión sistemática

Es posible que las preguntas de investigación no cubrieran todos los aspectos del tema de investigación. Para mitigar esta limitación, se realizaron tres revisiones para ajustar el propósito de cada una. Esto puede haber llevado a que no se hayan recuperado algunos artículos, para esto se definió el protocolo de búsqueda y se utilizó la técnica de snowballing para nutrir más la búsqueda.

Es posible que no haber incluido el término *technical debt* haya aumentado el riesgo de excluir algunos artículos valiosos enfocados en la deuda técnica. Para esto, se realizaron búsquedas manuales solo con este término antes de la ejecución del protocolo, obteniendo resultados similares con el término *documentation debt*. A pesar de que no es posible garantizar que las búsquedas manuales hayan sido totalmente precisas si permitió identificar que los artículos enfocados en deuda técnica abordan muchos tipos de deuda, pero no son específicos del tema de investigación.

Es difícil garantizar que los artículos primarios fueron seleccionados y analizados adecuadamente ya que la inclusión exclusiva de artículos en inglés puede haber descartado artículos importantes escritos en otros idiomas. De igual forma, la técnica de extracción de información puede haber estado sesgada debido a que la clasificación se realizó bajo la opinión y conocimientos de los involucrados.

2.2.3.3. Importancia para la investigación y la práctica

Los resultados de esta búsqueda son valiosos para futuras investigaciones sobre deuda de la documentación en el desarrollo ágil de software porque evidencia la importancia de la documentación para mejorar la calidad del software. Esto presenta a la deuda de la documentación en un área de investigación necesaria y sobre la que existe poca evidencia, principalmente en enfoques ágiles. Por otro lado, la documentación es muy importante para una sana evolución y mantenimiento de los sistemas software y se podrían proponer nuevas formas de documentar el software enfocadas en la automatización de este proceso para reducir la subjetividad, informalidad y heterogeneidad en la documentación. Esto sería muy beneficioso para la industria mejoraría las prácticas de documentación en las empresas y ser más competitivos en el mercado.

Capítulo 3. Caracterización de la documentación de software

En este capítulo se presenta la caracterización de los elementos más representativos a tener en cuenta en la documentación generada en el ciclo de vida del desarrollo de software, estos han sido identificados a partir de los hallazgos de la revisión del estado del arte presentado en la sección anterior (ver sección 2.2) e información adicional obtenida a partir de otras fuentes consultadas. Esta caracterización es la base para la definición de la ontología presentada en el Capítulo 4 y el modelo de métricas presentado en el Capítulo 5.

3.1. Proceso de identificación de los elementos de la documentación

La identificación de los elementos más representativos de la documentación de software se realizó mediante el proceso presentado en la Figura 3.1.

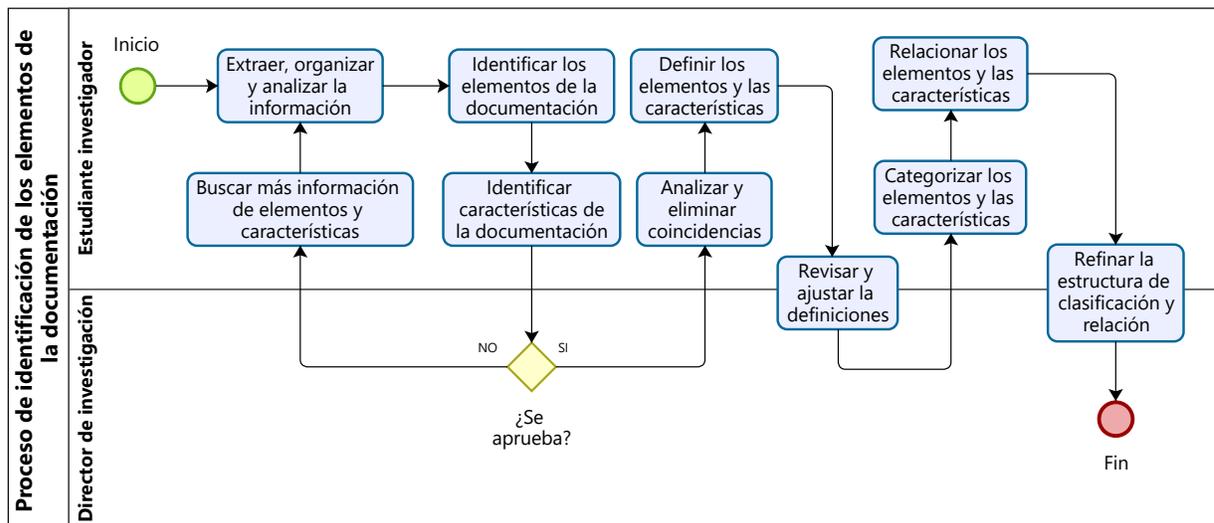


Figura 3.1. Proceso de identificación de los elementos de la documentación.

A continuación, se describen las actividades que conforman el proceso:

- 1. Extraer, organizar y analizar la información:** Actividad donde se recolectó la información necesaria de los artículos obtenidos en la revisión sistemática de la literatura y de búsquedas manuales por separado.
- 2. Identificar elementos y características de la documentación:** A partir de la información recolectada se identificaron los elementos que pudieran pertenecer al contexto de la documentación y las posibles características de la documentación.

3. **Buscar más información para los elementos y las características:** Actividad donde se consultaron otras fuentes de información para encontrar detalles adicionales de los elementos y las características y resolver dudas o ambigüedades.
4. **Analizar y eliminar las coincidencias:** Actividad donde se revisó la similitud o relación en la información de los elementos y características identificados para unificarlos.
5. **Definir los elementos y las características:** Actividad donde se construyó o adaptó una definición para cada elemento y característica final a partir de la información recolectada. Esta actividad se realizó principalmente para aquellas características sobre las cuales no fue posible encontrar una definición suficientemente completa para ser utilizada.
6. **Revisar y ajustar las definiciones:** Actividad donde se revisaron y ajustaron las definiciones de cada elemento y característica para corregir posibles errores ortográficos o gramaticales, eliminar ambigüedades y mejorar su coherencia.
7. **Categorizar los elementos y las características:** Actividad donde se clasificaron los elementos y las características en categorías acorde a su significado y propósito para facilitar su análisis y comprensión.
8. **Relacionar los elementos y las características:** Actividad donde se analizó si podría existir una relación entre los elementos y características identificadas teniendo en cuenta su definición y los detalles de cada uno.
9. **Refinar la estructura:** Actividad donde se revisaron las relaciones entre los elementos y las características para ajustarlas y establecer una versión preliminar de la ontología.

3.2. Elementos fundamentales en la documentación de software

3.2.1. Tipos de documentación

Se identificaron diferentes tipos de documentación que podrían tenerse en cuenta en el proceso de documentación de software; aunque, los autores coinciden en que no es posible establecer un tipo específico ya que esto depende de aspectos como: los compromisos contractuales, el tipo de software, su vida útil, la cultura y el tamaño de la empresa [45]. Sin embargo, la documentación de software se puede clasificar en dos grandes grupos:

3.2.1.1. Documentación del proceso

Esta documentación formaliza el proceso de desarrollo, permite que sea visible por todos los interesados y se gestione adecuadamente y puede contemplar los siguientes artefactos:

- **Planes, estimaciones y cronogramas:** producidos por la alta gerencia para predecir y controlar el proceso de desarrollo.
- **Informes:** producidos por los equipos asociados al proyecto donde se explica cómo se utilizaron los recursos durante el proceso de desarrollo.
- **Estándares:** producidos por los líderes del proyecto donde se establecen los estándares que se van a seguir durante el proceso.

- **Documentos de trabajo:** producidos por el equipo de desarrollo para la comunicación técnica donde se registran las estrategias de implementación, la solución de problemas y las justificaciones de diseño. En particular, estos artefactos son versiones provisionales de la documentación del producto ya que pueden ser actualizados.
- **Memorandos y mensajes de correo electrónico:** documentos donde se registran los detalles de las comunicaciones diarias entre los gerentes y los ingenieros.

Sin embargo, a pesar de que la mayor parte de esta documentación puede quedar obsoleta ante cambios en el sistema, hay documentos que pueden ser útiles a medida que el software evoluciona, como, por ejemplo, los planes de prueba, que son la base para planificar la validación del sistema y los documentos de trabajo que justifican las decisiones de diseño.

3.2.1.2. Documentación del producto

Esta documentación describe las características del software desde el punto de vista técnico para los ingenieros de desarrollo y desde el punto de vista de uso para los usuarios finales. Suele tener un tiempo de vida más largo que la documentación de procesos y debe construirse a la par del sistema. La documentación del producto puede contemplar:

- **Documentación de usuario:** Es una documentación creada para ayudar a dos tipos de usuarios; primero, a los usuarios finales para utilizar correctamente el software sin preocuparse por los detalles técnicos; y segundo, a los administradores del software para realizar tareas de gestión de la información, monitoreo del funcionamiento o solución de problemas a los usuarios finales. A continuación, en la Tabla 3.1 se describen los artefactos de la documentación de usuario propuestos por [45]:

Tabla 3.1. Artefactos de la documentación del usuario. Tomado de [45].

Artefacto	Propósito
Descripción funcional	Describir de manera general de las características del sistema y los servicios que presta. Ofrece una introducción a su funcionamiento para ayudar a los usuarios a decidir si es lo que ellos necesitan.
Documento de instalación del sistema	Describir los detalles de cómo instalar el sistema en un entorno en particular y los requisitos para ello. También describe los archivos que componen el sistema, el hardware mínimo requerido, los archivos permanentes, como iniciar el sistema y los archivos de configuración necesarios para su adaptación. Asimismo, ayuda a los administradores a descubrir y solucionar problemas relacionados con la instalación.
Manual introductorio	Presentar una introducción informal al sistema describiendo su uso correcto mediante numerosos ejemplos sencillos y prácticos que ilustren, principalmente, como recuperarse de errores y reiniciar el trabajo útil.
Manual de referencia del sistema	Ofrecer una descripción completa de los mensajes de error y como solucionarlos. Es un artefacto donde la información debe ser integra con el sistema y debe expresarse mediante técnicas descriptivas formales.
Guía del administrador del sistema	Describir de manera general como interactúa el sistema construido con otros sistemas software o hardware ya existentes para comprender lo que se debe hacer en determinadas circunstancias.

- **Documentación del sistema:** Es la documentación que describe el sistema desde el punto de vista técnico, es decir, la especificación de requisitos, el diseño, la implementación, el plan de pruebas, y cualquier otro documento que permita comprender los detalles del desarrollo del sistema y como mantenerlo [46]. La documentación del sistema puede incluir los artefactos presentados en la Tabla 3.2:

Tabla 3.2. Artefactos de la documentación del sistema. Tomado de [45].

Artefacto	Propósito
Especificación de requisitos	Describir de manera completa del comportamiento que debe tener el sistema.
Arquitectura del sistema	Describir los detalles de la estructura arquitectónica del sistema desde diferentes perspectivas.
Arquitectura de cada componente	Describir los detalles de la estructura arquitectónica de cada componente del sistema desde diferentes perspectivas.
Especificación de cada funcionalidad y sus interfaces	Describe los servicios que ofrece el sistema para satisfacer cada requerimiento. En su descripción, estos servicios pueden tener asociadas una o más interfaces para la comunicación entre el usuario y el sistema.
Documentos de código fuente	Explicación de secciones complejas del código y la justificación de las técnicas de codificación utilizadas. Es recomendable que estos artefactos sean digitales y estén enfocados solo en las secciones más críticas.
Documentos de validación	Describen como se valida cada componente del sistema y su relación con los requisitos especificados.
Guía de mantenimiento del sistema	Describe los problemas conocidos del sistema, la interdependencia de los componentes software y hardware, y cómo ha evolucionado el diseño del sistema.

Por otra parte, en la sección 3.2.1 del presente documento, donde se describieron los flujos del proceso de documentación, se mencionaron los artefactos que pueden tenerse en cuenta para documentar el software y que se pueden clasificar como documentación del sistema o de usuario. El más importante de ellos es *el plan de documentación* donde se consigna toda la estrategia de documentación, las herramientas, los actores, las funciones de cada actor y los artefactos que se van a crear.

Otro artefacto muy importante es *la especificación de la arquitectura de la Información* la cual organiza y estructura la información para satisfacer las necesidades de los usuarios, teniendo en cuenta los aspectos estéticos, funcionales, de navegación y de recuperación de los artefactos [47]. De esta forma, definir una arquitectura para la información permite organizarla de manera coherente, presentarla de una forma estéticamente agradable, comprender e interiorizar rápidamente las funcionalidades del sistema, encontrar fácilmente la información que requiere cada usuario, y ampliar su contenido o descomponerla para ser reutilizada en diferentes contextos o para diferentes estrategias de implementación [47].

3.2.2. Roles que intervienen en la documentación

En el proceso de documentación de software pueden existir diferentes roles forman parte de un equipo de documentación e intervienen en las etapas del proceso. A continuación se describen las responsabilidades de los roles más importantes según [47]:

- **Arquitecto de la información:** Especifica como se debe organizar, estructurar y navegar la información. Adicionalmente, supervisa que la información en línea o impresa mantenga su integridad con el sistema durante su construcción, desarrolla y mantiene el plan de documentación para el proyecto e indica al equipo de documentación cuáles serán los artefactos que se producirán en cada etapa [47].
- **Escritor técnico:** Implementa el plan de documentación para cada componente del sistema redactando los artefactos necesario y garantiza que sean revisados y aprobados por el arquitecto de la información de acuerdo al plan [47].

- **Editor técnico:** Junto con el escritor técnico, es responsable de mantener el estilo y la consistencia de todos los artefactos. De igual forma, define las reglas y pautas de estilo editorial para todos los artefactos de documentación [47].
- **Tester o revisor de documentos:** Valida que la documentación cumpla con el propósito para el que fue diseñada. De forma similar al software, este rol puede realizar pruebas de usabilidad de la documentación, incluir comentarios de revisión, diseñar planes e informes de control de calidad y presentar solicitudes de cambio para la documentación [47].
- **Director de documentación:** Trabaja en conjunto con el director del proyecto en gestionar los recursos necesarios para la documentación, definir las prioridades de documentación, y coordinar los objetivos, las actividades y el cronograma de los entregables de documentación. Asimismo, establece las prácticas y herramientas que se deben utilizar para asegurar la integridad y la calidad de todos los artefactos [47].
- **Ingeniero de soporte:** Realiza el soporte técnico de las herramientas para documentar definidas por el director de documentación para desplegarlas adecuadamente [47].
- **Especialista en producción:** Construye los archivos fuente de todos los artefactos de documentación, realizan una verificación final de integridad y calidad, y gestiona la producción de la documentación física y en línea con los proveedores [47].
- **Diseñador de página:** Estandariza la apariencia final de los artefactos impresos o en línea. Para esto, construye todas las plantillas necesarias de los artefactos y define las pautas para que la información se produzca de acuerdo con el diseño [47].
- **Ilustrador técnico:** Produce los gráficos de todos los artefactos impresos y en línea. Para esto, define las pautas y reglas de diseño para garantizar que todas las ilustraciones sean coherentes con el propósito de cada artefacto [47].

Adicionalmente, Haneef en [48] propone la creación de un grupo denominado **grupo de documentación y legibilidad** encargado exclusivamente de:

- Construir guías que expliquen los lineamientos que debe seguir la documentación y como se debe utilizar para comprender correctamente su contenido.
- Revisar periódicamente el diseño del sistema para mejorar su legibilidad.
- Revisar la legibilidad de la documentación necesaria luego de un cambio en el sistema.
- Apoyar las revisiones de código para sugerir mejoras en su legibilidad.
- Verificar la legibilidad del código en su versión final para sugerir y negociar cambios.
- Crear la documentación que sea necesaria a medida que se desarrolla el sistema. Esto incluye desde la actualización de artefactos hasta la producción de nuevos artefactos.
- Garantizar la legibilidad de la documentación durante el control de calidad del software mediante la revisión y corrección de errores de legibilidad y actualización de artefactos.

Haneef [48] también propone que el grupo de legibilidad y documentación sea pequeño en las primeras etapas de desarrollo del sistema, pero con la capacidad de crecer a medida que avanza el proyecto integrando programadores novatos y experimentados para que el grupo tenga diferentes puntos de vista y perspectivas sobre cada situación. Finalmente, propone que uno de cada diez desarrolladores se encargue de la documentación del sistema y se rote esta responsabilidad para potenciar la habilidad de documentar el software.

3.2.3. Herramientas para documentar

Sommerville en [45] menciona diferentes herramientas que apoyan el proceso de documentación en todas sus etapas; sin embargo, explica que el conocimiento y la habilidad en el uso de una herramienta influirá enormemente en el proceso de documentación. A continuación, se describen las herramientas identificadas las cuales han sido clasificadas en tres categorías para facilitar su comprensión:

3.2.3.1. Procesadores de texto

Los *procesadores de texto* como Word son las herramientas más comunes debido a su versatilidad, facilidad de uso, popularidad y porque ofrecen una variedad de opciones de formato muy intuitivas. Por otro lado, existen los *sistemas de formateo de texto* como Latex donde se pueden construir documentos de alta calidad tipográfica por medio de instrucciones similares a las de un lenguaje de programación, con lo cual es necesario este tipo de conocimiento especializado y de un intérprete para presentar el resultado final. Sin embargo, la selección de uno u otro al momento de documentar dependerá de la rapidez con la que se pueda utilizar. De esta forma, podría ser más probable seleccionar los procesadores de texto debido a su facilidad de uso, ya que los formateadores de texto exigen dedicar tiempo para entender el lenguaje de programación que manejan, aunque permitirían construir documentos mucho más profesionales [45].

3.2.3.2. Herramientas de producción visual

En este grupo se destacan los *paquetes de autoedición* los cuales, a diferencia de los procesadores y formateadores de texto, ofrecen un control total sobre el maquetado, el espaciado, las dimensiones, y la disposición de imágenes y texto en el artefacto. De esta forma, los paquetes de autoedición garantizan un resultado profesional y de excelente calidad, tanto para documentos digitales como impresos. Actualmente, existen en el mercado herramientas muy populares como Adobe InDesign o Corel Draw, sin embargo, la selección de un paquete de autoedición dependerá de su costo, su rendimiento, la facilidad de uso, sus funcionalidades y sus características técnicas. Además, estas herramientas se pueden complementar con *paquetes de material gráfico* y *programas de estilo tipográfico* [45]. Ahora bien, es importante tener en cuenta que los paquetes de autoedición no automatizan las habilidades de un diseñador gráfico, a pesar de que actualmente ofrezcan una seductora facilidad de uso y permitan que usuarios inexpertos produzcan documentos poco atractivos y mal diseñados. Por lo tanto, se requiere de los conocimientos de un diseñador gráfico, y sistemas de escaneo y procesamiento de fotografías para que los ingenieros puedan producir una documentación de alta calidad visual [45].

3.2.3.3. Herramientas de administración

Por lo general, se produce una gran cantidad de artefactos en el desarrollo de un proyecto software, los cuales, para administrarse correctamente, requieren de *sistemas de gestión documental*, necesarios para controlar de manera eficiente y sistemática la creación, recepción, mantenimiento, utilización y disposición de la documentación. Adicionalmente, se requieren *sistemas de gestión de la configuración*, que permiten supervisar y controlar los cambios en los artefactos de documentación. Estos sistemas se apoyan en el *control de versiones* donde se justifica el propósito del cambio junto a los detalles que serán actualizados y disminuye la probabilidad de que los usuarios de la documentación cometan errores por acceder a versiones desactualizadas de un documento. Otras herramientas para administrar

la documentación son los *sistemas de recuperación* que respaldan el mantenimiento de la documentación y permiten su reutilización en otros proyectos [45].

3.3. Dimensiones y características de la documentación de software

A partir de la información recolectada en la revisión sistemática presentada en la sección 2.2, Capítulo 2 de este documento, se identificaron diferentes características de la documentación; sin embargo, no se evidenciaron categorías en la literatura analizada, asimismo, se pudo observar que su definición carece de profundidad, estructura y detalle [19], [29], [49]. Por lo tanto, en esta sección se presentan definiciones detalladas de cada característica y se clasifican en cinco dimensiones de análisis para mejorar su comprensión.

3.3.1. Descripción de las dimensiones y características

Tomando como base la definición descrita en [49], las dimensiones de la documentación hacen referencia a los aspectos más representativos de un artefacto de documentación que permiten estudiarlo y extraer la mayor cantidad posible de información para analizar y comprender correctamente su contenido, facilitar su uso adecuado y clasificar sus características más importantes. En ese sentido, como se muestra en la Figura 3.2, se han definido 5 dimensiones: (i) Estructura, (ii) Formato, (iii) Usabilidad, (iv) Correlación y (v) Auditabilidad; las cuales agrupan un conjunto de 17 características o atributos de la documentación de software. Adicionalmente, se presenta una definición tanto para las dimensiones como para las características con el propósito de facilitar su comprensión.

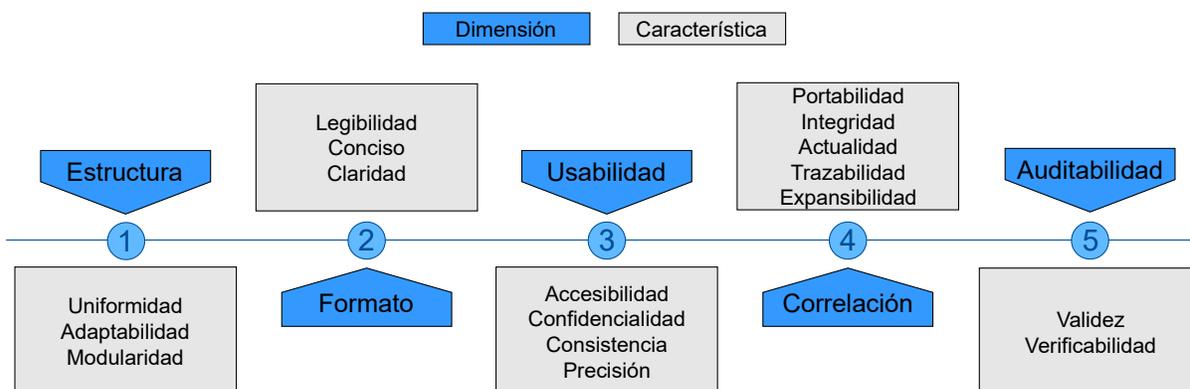


Figura 3.2. Dimensiones y características de la documentación de software.

3.3.1.1. Primera dimensión: Estructura

Según Koznov [50], esta dimensión se refiere a la distribución del contenido en un artefacto de documentación cuyo orden e interrelación permiten su comprensión y utilización de manera eficaz. Cada parte del artefacto debe cumplir con una función específica y correlacionarse con las demás partes para mantener el sentido y el propósito del artefacto. Dicha correlación puede ser interna —con partes del mismo artefacto— o externa —con partes de otros artefactos—. Los elementos que hacen parte de la estructura son: capítulos, subcapítulos, índice, listas de ilustraciones, listas de tablas, introducción, instrucciones de uso, posibles errores y su solución, glosario, artefactos relacionados, navegación y conceptos importantes. A continuación, se describen las características que hacen parte de esta dimensión.

3.3.1.1.1. Uniformidad (Uniformity)

Esta característica hace referencia a que la información del artefacto sea coherente y no existan conflictos [29]. Para esto, se debe detectar, analizar y corregir las contradicciones y la duplicidad de información en la documentación [1], [29], [50]. En consecuencia, se encuentra fuertemente relacionada con la característica de *consistencia*, ya que su ausencia puede confundir a los usuarios del artefacto y obligarlos a sobre esforzarse para comprender su contenido. También se relaciona con la *integridad* ya que si el artefacto no contiene la información necesaria sobre el sistema los profesionales y usuarios finales no podrían realizar sus tareas adecuadamente. Asimismo, se relaciona con la característica de *actualidad* por que facilita que la información permanezca correcta y completa después de actualizar el sistema y permite la reutilización de la documentación en futuros desarrollos. La uniformidad también facilita que la documentación pueda revisarse adecuadamente durante el control de calidad, corregir rápidamente los errores y reestructurar los artefactos. Para esto, se pueden considerar cinco tipos de uniformidad [1], [51]:

- **Uniformidad en el formato:** Se refiere a adecuar el estilo de redacción y utilizar correctamente las reglas gramaticales para mantener la unanimidad de la información de cada artefacto de acuerdo a su propósito.
- **Uniformidad en la estructura:** Se refiere a estandarizar la presentación de la información en el artefacto mediante la numeración de páginas, encabezados, notas al pie, figuras, tablas, esquema de navegación, entre otros, teniendo en cuenta el propósito del artefacto y sus posibles usuarios.
- **Uniformidad en el almacenamiento de la información:** Se refiere a estandarizar los mecanismos de almacenamiento y preservación de todos los artefactos teniendo en cuenta las características particulares de cada uno.
- **Uniformidad en el acceso a la información:** Se refiere a estandarizar los mecanismos de acceso a la información de acuerdo al nivel de acceso otorgado a cada interesado en la documentación.

Uniformidad en la actualización de la documentación: Se refiere a estandarizar los mecanismos que permiten actualizar la documentación, los cuales deben mantener una concordancia con las políticas de actualización del sistema que describe.

3.3.1.1.2. Adaptabilidad (Customizable)

Esta característica ha sido poco abordada en la literatura y corresponde a la capacidad que tiene el documento para evolucionar y mantenerse relevante a medida que cambian las necesidades y los requisitos del proyecto [52]. Beneficia en gran medida a la portabilidad de la documentación, sin embargo, se debe gestionar con precaución ya que en la práctica no es totalmente viable personalizar los artefactos ante cada nueva necesidad debido al elevado costo de planificar, crear y mantener esta nueva documentación. En su lugar, se puede crear modelos comunes a partir de los diferentes puntos de vista de todos los interesados que posteriormente sirvan para crear diferentes tipos de artefactos. Estos modelos también se utilizarían como una plantilla para añadir los aspectos específicos de cada interesado y así adaptar mejor la documentación [52].

3.3.1.1.3. Modularidad (Modularity)

Esta característica permite fragmentar un artefacto en partes o secciones cohesivas capaces de cumplir con un objetivo específico, interactuar entre sí y complementarse para presentar,

estructurar y explicar el contenido del artefacto. La modularidad le permite a los usuarios de la documentación conocer ciertos detalles del sistema sin necesidad de estudiar toda la documentación de principio a fin [1]. Para lograrlo, es necesario que cada sección sea lo más independiente posible, ya sea dentro de un artefacto o entre artefactos, y mantener solamente las relaciones necesarias. De esta forma se podrían modificar integrar o eliminar secciones con el menor impacto posible sobre toda la documentación.

3.3.1.2. Segunda dimensión: Formato

Tomando como referencia lo mencionado en [1], [29], [50], esta dimensión se refiere a la uniformidad en la presentación visual del contenido, así como al lenguaje y la gramática utilizados en la documentación. Ahora bien, aunque esta dimensión se preocupa por el aspecto y la forma de presentar la información no representa una barrera muy grave para el uso de la documentación a corto plazo debido a que mucho del conocimiento adquirido por los equipos de desarrollo, así como los modelos mentales de sus integrantes, son recientes y pueden utilizarse de manera casi inmediata para comprender la información que parezca confusa. Sin embargo, a largo plazo el conocimiento tácito puede degradarse y el formato que se utilice en la documentación de software influirá notoriamente en la comprensión de la información en el futuro. En ese sentido, los elementos que hacen parte de esta dimensión son: estilo de redacción, cantidad de páginas o vistas, uso de diagramas o ejemplos, disposición espacial, complejidad técnica, ortografía y gramática. A continuación, se describen las características que hacen parte de esta dimensión.

3.3.1.2.1. Legibilidad (Readability)

Esta característica se relaciona con la facilidad de lectura de un artefacto, lo cual convierte a la legibilidad en un atributo subjetivo que depende de la percepción de los usuarios [1], [53]. En ese sentido, la información abstracta, demasiado técnica o sobre cargada, y errores tipográficos son aspectos en la documentación que puede afectar su legibilidad [1]. Por otro lado, es importante aclarar la diferencia entre legibilidad y comprensibilidad; puntualmente, la legibilidad mide el esfuerzo de los usuarios para acceder a la información contenida en el artefacto, mientras que la comprensibilidad mide la complejidad de dicha información. La legibilidad puede estudiarse desde cuatro perspectivas: visual, espacial, de alineación y lingüística, así como la coherencia textual y la cantidad de conceptos utilizados [53].

3.3.1.2.2. Conciso (Conciseness)

Esta característica hace referencia a la capacidad de la documentación para utilizar la información estrictamente necesaria para explicar los detalles del sistema sin perder la exactitud en su contenido. Está fuertemente relacionada con la legibilidad debido a que el contenido del artefacto debe ser lo suficientemente detallado para describir correctamente las funcionalidades del sistema sin incluir secciones inútiles que ocasionen discrepancias o lo hagan pesado, incoherente e incomprensible [1].

3.3.1.2.3. Claridad (Clarity)

Esta característica se relaciona con la capacidad del contenido para ser comprensible e inteligible sin que genere dudas o incertidumbres. La claridad depende en gran medida de utilizar un lenguaje apropiado y un estilo de redacción simple y eficiente, que permita a los usuarios encontrar rápidamente la información que requieren sin necesidad de releer el contenido. Asimismo, la claridad debe tener en cuenta la estructura, el lenguaje visual y el diseño del contenido para que el mensaje sea transmitido de manera efectiva a los usuarios, de acuerdo a conocimientos técnicos y capacidades cognitivas. Por lo tanto, la claridad es un

atributo subjetivo que depende también de la percepción de quien utiliza el artefacto [29], con lo cual, también se relaciona con la legibilidad de la información. Sin embargo, es posible identificar la falta de claridad en aspectos tan específicos como ejemplos de código poco claros, nombres de métodos confusos, títulos confusos, suposición de conocimientos previos, uso excesivo de siglas, entre otros [1].

3.3.1.3. Tercera dimensión: Usabilidad

En general, la usabilidad se define como la capacidad, conveniencia o idoneidad de algo para ser utilizado. En relación a la documentación de software, según lo mencionado por [1], [54], la usabilidad indica la facilidad con la que un lector puede extraer toda la información necesaria para cumplir con sus objetivos inmediatos. En ese sentido, esta dimensión permite estudiar el uso efectivo de la información cumpliendo con requisitos que influyen en su accesibilidad a través del esquema de navegación en los artefactos. Por lo tanto, esta dimensión tiene en cuenta la estructura del artefacto, su formato, la facilidad para acceder y utilizar su información, la efectividad de su navegación y los mecanismos para ser preservado. A continuación, se describen las características que hacen parte de esta dimensión.

3.3.1.3.1. Accesibilidad (Accessibility)

Esta característica se refiere a la capacidad de la documentación para ser localizada y operativa cuando sea necesario para recuperar, presentar, leer e interpretar su contenido [29]. La accesibilidad es una de las características sobre la cual se ha discutido bastante junto con la integridad y la consistencia [1] debido a que considera la disponibilidad de la información, el nivel de ocultamiento que se maneja para cada usuario y la facilidad de localización [29]. Sin embargo, esta característica va más allá de solo permitir el acceso a la información, ya que involucra entender el contexto organizacional, empresarial y técnico sobre el cual se va a utilizar la documentación para comprender claramente en qué momento se debe crear, consultar, modificar o eliminar un artefacto.

3.3.1.3.2. Confidencialidad (Privacy)

Tomando como base la norma ISO 27001 sobre seguridad de la información, la confidencialidad es una característica de la documentación para estar disponible solo para aquellas personas, entidades o procesos autorizados. Es una característica clave, junto con la disponibilidad e integridad, para mantener segura la información. Para que la documentación cuente con esta característica, es necesario clasificar los artefactos de acuerdo a su nivel de accesibilidad: público (abierto a todo el público), de uso interno (exclusivo para el equipo de trabajo), restringido (accesible de acuerdo a las políticas de la empresa) y confidencial (accesible solo con autorización). Debe ser una característica claramente visible en los artefactos, por ejemplo, utilizando etiquetas de color que indiquen su clasificación.

3.3.1.3.3. Consistencia (Consistency)

Esta característica hace referencia a la capacidad de la documentación para evitar discrepancias potenciales o reales entre el contenido y el funcionamiento real del sistema [55]. La consistencia puede estudiarse desde diferentes perspectivas, por ejemplo: en la formalidad del artefacto, la semántica, contenido visual, organización del contenido y en el formato [29], [55]. La consistencia se apoya en la uniformidad e integridad del artefacto, ya que si la información es inconsistente y contradictoria causará confusión y sobreesfuerzos para comprenderla. La consistencia también involucra la gestión adecuada de las relaciones entre artefactos debido a que la información podría estar duplicada o dispersa en varios artefactos

[56]. Sin embargo, no se suele mantener correctamente la consistencia y trazabilidad de la documentación debido, generalmente, a los tiempos de entrega del software [57]. El resultado de esto es una documentación envejecida y que fracasa en su función fundamental de comunicar [57]. De esta forma, la trazabilidad se convierte en un indicio del grado de consistencia en la descripción de los distintos aspectos de un sistema [57].

3.3.1.3.4. Precisión (Accuracy)

Esta característica se relaciona con la capacidad del artefacto para transmitir la información de manera clara, exacta y sin ambigüedades [29], reflejando el estado real del sistema que describe [54]. Por el contrario, cuando la documentación es imprecisa o demasiado abstracta puede ser mucho peor que si no existiera y esto se debe a que puede impedir o confundir a los usuarios cuando realizan sus actividades [29]. Por lo tanto, la documentación debe ser lo más precisa posible al describir las características y funcionalidades del software para que sea altamente efectiva durante la comunicación entre todas las partes interesadas [58].

Ahora bien, según Arthur y Stevens en [54], la falta de precisión en la documentación se pueda dar por diferentes razones, pero se destacan dos; la primera, por cometer errores en la traducción de los requisitos del sistema a instrucciones de diseño; y la segunda, por una mala interpretación de las especificaciones de diseño o la actualización del sistema sin actualizar la documentación respectiva. En cualquiera de los dos casos, el resultado es que toda la documentación, o al menos una parte, será incorrecta e imprecisa con respecto al funcionamiento real del sistema [54].

3.3.1.4. Cuarta dimensión: Correlación

Esta dimensión se refiere a la correspondencia que debe existir entre el sistema y su documentación. Dado que la documentación ayuda a comprender el sistema mediante la descripción formal de sus funcionalidades, sus características técnicas, los procesos que se siguen, y en general todo aquello que permite desarrollar, implementar, utilizar y soportar el software; debe asegurarse la coherencia entre la información de los artefactos de documentación y el comportamiento del sistema. Por lo tanto, es una dimensión que abarca los atributos de la documentación enfocados en mantener actualizada la información cada vez que ocurre un cambio en el sistema; manteniendo su correspondencia con la última versión para evitar confusiones, inconvenientes o sobre esfuerzos en el futuro. A continuación, se describen las características que hacen parte de esta dimensión.

3.3.1.4.1. Portabilidad (Portability)

Según el diccionario de la Real Academia Española, la portabilidad es la cualidad de un objeto para que sea movable y fácil de transportar, mientras que la ISO 25010 la define como la capacidad de un producto para ser transferido de un entorno a otro de manera efectiva y eficiente. Teniendo en cuenta estas definiciones en el ámbito de la documentación, la portabilidad es una característica de la documentación que permite transferirla de un entorno tecnológico, operacional o de uso a otro de forma eficiente y efectiva. Para lograrlo, la documentación debe ser: (i) adaptable a diferentes entornos hardware, software u operacionales, (ii) fácil de disponer en un determinado entorno, y (iii) fácil de reemplazar ante un fallo o inconveniente. Actualmente, existen diferentes tipos de artefactos (física, digital, páginas web, wikis y aplicaciones móviles), con lo cual se pueden usar estándares como PDF (Portable Document Format), el cual es reconocido por la Organización Internacional para la Estandarización (ISO) como un formato que permite presentar e intercambiar documentos de

manera fiable sin depender del software, hardware o sistema operativo, con lo cual se logra que la documentación sea totalmente independiente e interoperable.

3.3.1.4.2. Integridad (Completeness)

Esta característica describe hasta qué punto la información de la documentación permanece correcta y completa después de un cambio en el sistema y mantiene su capacidad de apoyar las tareas de desarrollo y mantenimiento. La integridad, es una característica que asegura que la documentación contenga la información suficiente para describir el sistema, de tal manera que los usuarios puedan recuperar la información necesaria para realizar sus tareas. Por el contrario, la falta de integridad ocasiona que la documentación no cumpla con su propósito y sea inútil para los usuarios [1], [29].

3.3.1.4.3. Actualidad (Up-to-dateness)

Es una característica que describe hasta qué punto la documentación se mantiene actualizada a lo largo del ciclo de vida del software. Es una característica que depende de una buena trazabilidad de los artefactos para que el sistema y su documentación evolucionen al mismo tiempo manteniendo una correspondencia [1]. En ese sentido, la desactualización de la documentación ocurre cuando su elaboración no está sincronizada con el desarrollo del sistema. Por lo tanto, es muy importante mantener una documentación actualizada, de lo contrario, se describirían versiones anteriores del sistema, la información sería incorrecta y puede inducir al error a los usuarios [29].

3.3.1.4.4. Trazabilidad (Traceability)

Esta característica describe la evolución del artefacto con el propósito de identificar donde se hizo el cambio, en qué momento, quién lo realizó y la razón del cambio. De esta forma, la trazabilidad ayuda a rastrear la modificación de un artefacto y mantener su consistencia con el sistema [1]. Sin embargo, a pesar de su importancia, es un aspecto muy descuidado en la documentación de software que requiere de mucha atención [55], en especial, en la documentación técnica debido a la correspondencia que debe mantener con el software. Por otro lado, la trazabilidad permite entender la relación entre dos o más artefactos y el nivel de dependencia entre ambos [57], [59]. Por lo tanto, es una característica que permite justificar y verificar todos los cambios que ocurren en el software [29], además de ser muy útil en procesos técnicos de ingeniería inversa, mantenimiento y pruebas de software [60].

3.3.1.4.5. Expansibilidad (Expandability)

Esta característica hace referencia a la capacidad agregar nueva información en un artefacto de documentación o modificar la que ya existe, por ejemplo, aumentando la extensión de su contenido, integrando otros tipos de contenido, cambiando el volumen de los artefactos o el alcance de los mismos, entre otros. Es una característica que se relaciona con la evolución y el mantenimiento del sistema ya que son etapas del ciclo de vida del software donde pueden surgir nuevas funcionalidades o actualizar las que ya existen. En ambos casos, se debe documentar cualquier tipo de expansión y verificar la integridad y la consistencia entre la documentación y el software. Por lo tanto, cada artefacto debe permitir la integración de nueva información sin afectar a las demás características [54].

3.3.1.5. Quinta dimensión: Auditabilidad

Etimológicamente auditar significa revisar de manera sistemática una actividad o una situación para evaluar el cumplimiento de las reglas o criterios objetivos a los que debe someterse. Si

esta definición se traslada a la documentación, hace referencia a la inspección y verificación sistemáticas de cada artefacto con respecto al comportamiento del software, al tipo de información que se registra y a los criterios establecidos para la revisión. Como bien se sabe, una auditoría es un proceso que puede ser externo o interno donde la documentación debe servir como evidencia confiable de los procedimientos ejecutados y su alcance, así como las acciones y decisiones tomadas para la construcción del software. Es así como esta dimensión involucra los mecanismos utilizados para preservar adecuadamente la información para ser accedida durante los procesos de auditoría. A continuación, se describen las características que hacen parte de esta dimensión.

3.3.1.5.1. Validez (Validity)

Esta característica acredita la autenticidad, veracidad, legalidad y fiabilidad de la información contenida en un artefacto. La validez se logra cuando un artefacto cuenta con dos elementos: (1) la aceptación y firma de las partes involucradas en su creación y (2) un sello de autenticidad por parte de la organización. De esta forma es posible formalizar los acuerdos entre las partes interesadas, verificar el cumplimiento de los compromisos adquiridos y certificar la pertinencia y propiedad de la información. Es una característica poco estudiada, pero es muy importante porque permite evaluar si el software y su documentación cumplen con las expectativas de los usuarios.

3.3.1.5.2. Verificabilidad (Correctness)

Esta característica permite corroborar si la documentación proporciona información exacta conforme a hechos reales sin entrar en conflictos con otros artefactos [1]. De esta forma, si un artefacto presenta información incorrecta o que no haya sido verificada puede traer graves consecuencias no previstas, como, por ejemplo, inducir al error luego de utilizar la documentación, reproducir código fuente erróneamente o seguir instrucciones equivocadas de instalación y despliegue [1], [29]. Por lo tanto, es una característica que se relaciona con la exactitud del contenido y ayuda a mejorar su consistencia. Asimismo, se apoya en la legibilidad y estructuración del contenido ya que una buena gestión de cada uno aumentará la probabilidad de realizar una verificación efectiva [55].

Capítulo 4. Hacia una ontología de la deuda de la documentación de software

En este capítulo se presenta una versión inicial la ontología diseñada como una propuesta conceptual preliminar para organizar formalmente el conocimiento relacionado con la deuda de la documentación en el desarrollo ágil de software. Esta propuesta permite que dicho conocimiento pueda ser reutilizado y compartido por los interesados en esta temática, además de ayudar a comprender el fenómeno de la deuda de la documentación mediante la identificación e interacción de sus elementos fundamentales.

4.1. Propósito

La ontología propuesta surge como una iniciativa de organización para reducir la elevada heterogeneidad que se evidencia en la literatura actual sobre deuda de la documentación en el desarrollo ágil de software. Por consiguiente, la ontología presentada en este documento busca ser un punto de partida para estructurar el conocimiento relacionado con este tipo de deuda a partir de la homogenización de conceptos en este dominio, la forma como están relacionados entre sí y una propuesta de definición para cada uno. Finalmente, la ontología busca servir como una base para la construcción de un instrumento de valoración y evaluación de la deuda de la documentación en el desarrollo ágil de software.

4.2. Metodología de construcción

Luego de analizar diferentes alternativas para definir ontologías, tales como: la gestión del conocimiento basado en ontologías [61], Methontology [62], el formalismo de representación para ontologías en ingeniería de software (REFSENO) [63] y otros trabajos como los de Dhanushka [64], Alves et al. [65] y Garcia et al. [66], se decidió utilizar la metodología REFSENO para el desarrollo de la propuesta. La selección se basó en los siguientes criterios; primero, REFSENO fue diseñada especialmente para la definición de ontologías en el ámbito del desarrollo web utilizando constructores que permiten la definición de conceptos, atributos y relaciones; segundo, es una adaptación de Methontology, la cual es una metodología ampliamente utilizada para definir ontologías en diferentes contextos; y tercero, REFSENO utiliza un esquema de representación visual más simple e intuitivo a comparación de otras metodologías, con lo cual, es más adecuado para personas que no están familiarizadas con los conceptos de la lógica de predicados de primer orden o similares.

4.3. Integración con otras ontologías

Teniendo en cuenta que la ontología descrita en este capítulo soporta la evaluación de la deuda de la documentación en el desarrollo ágil de software, fue necesario incluir conceptos relacionados con los procesos de software y la medición de software. De esta forma se realizó una integración con algunos conceptos de la ontología denominada *Ontology of Process-reference Models (PrMO)* [67], que establece los conceptos principales para expresar

enfoques basados en procesos, y la ontología Software Measurement Ontology (SMO) [68], que establece los conceptos principales en el contexto de la medición de software.

4.4. Esquema gráfico

En la Figura 4.1 se presenta la representación gráfica de la ontología propuesta, los conceptos y las diferentes relaciones mediante el Lenguaje de Modelado Unificado (UML por sus siglas en inglés).

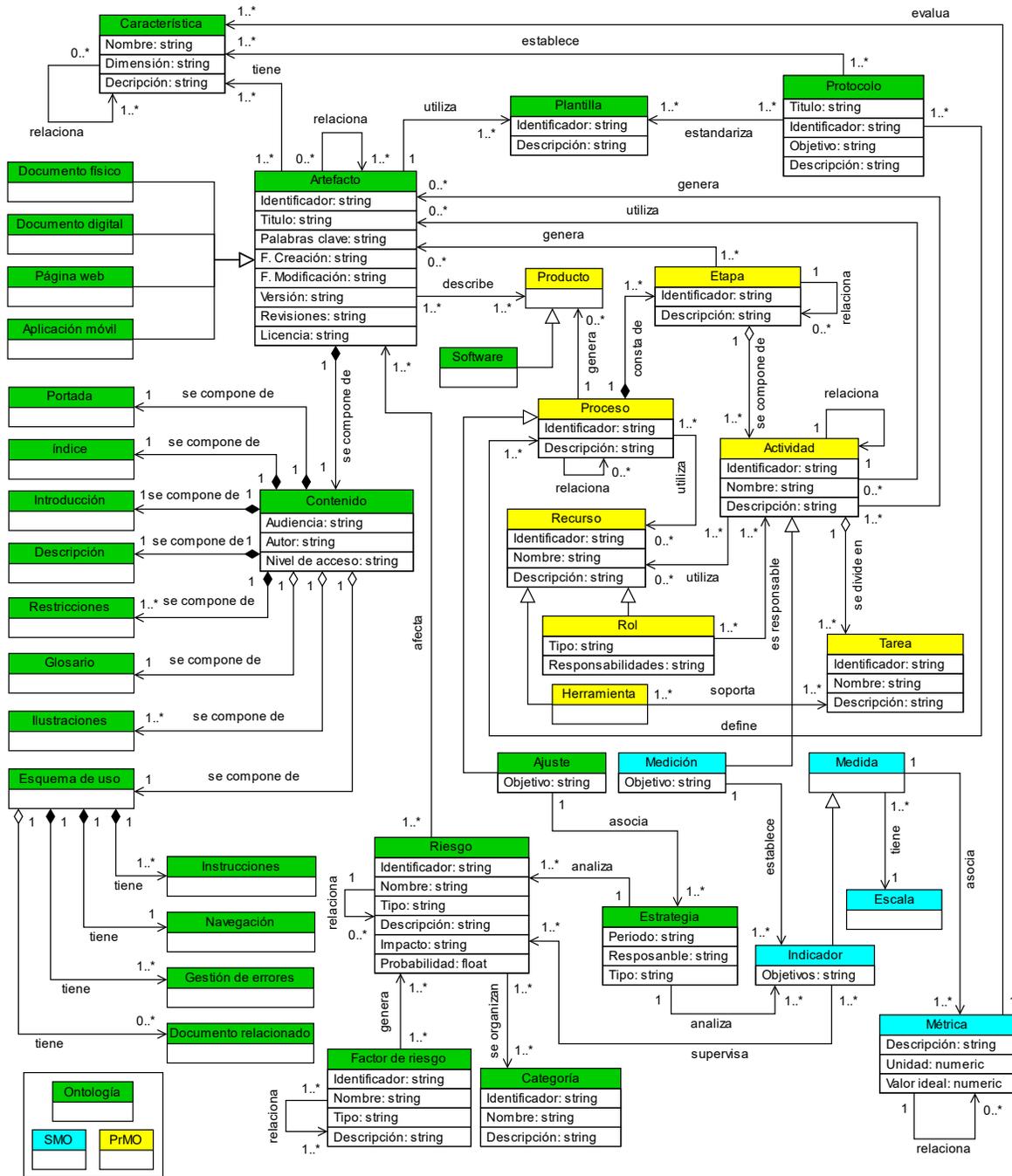


Figura 4.1. Ontología de la deuda de la documentación de software.

Adicionalmente, en la Figura 4.2 se presentan las diferentes relaciones entre las características de la documentación de software identificadas en la literatura actual y que fueron definidas en la sección 3.3.1 de este documento.

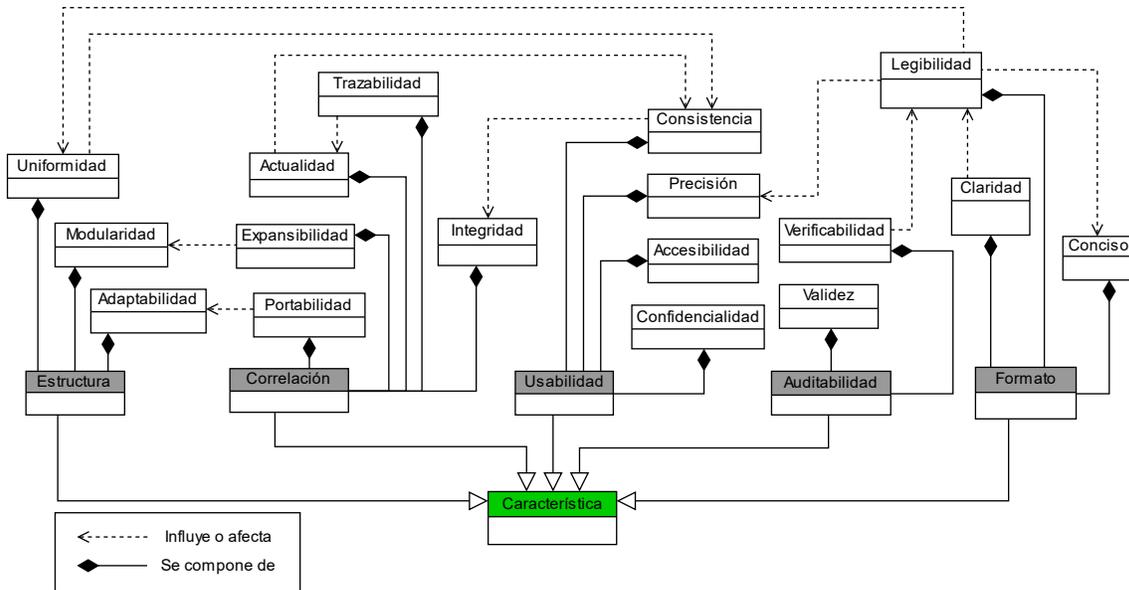


Figura 4.2. Características de la documentación de software.

4.5. Conceptos y relaciones de la ontología

En la Tabla 4.1 se presentan las definiciones de los conceptos que componen la ontología propuesta, los cuales están ordenados alfabéticamente, asimismo, la tabla se organiza de la siguiente forma; la primera columna muestra el nombre del concepto; la segunda columna muestra el super-concepto al que puede estar relacionado; la tercera columna muestra la definición del concepto; y la cuarta columna muestra la fuente usada para adaptar o adoptar el concepto. Por otro lado, en la Tabla 4.2 se presentan las diferentes relaciones entre los conceptos presentados en la Tabla 4.1. Por último, las fuentes presentadas en la cuarta columna corresponden a uno de los siguientes tipos:

- **Fuente propia:** Si el concepto ha sido definido de forma particular para esta ontología sin ninguna fuente adicional.
- **Adaptado de [fuente]:** Si la definición del concepto ha sido adaptada a partir de una fuente que proporciona información relacionada con el propósito del concepto.
- **Tomado de [fuente]:** Si la fuente ofrece una definición completa del concepto que se puede utilizar sin necesidad de cambiarla o adaptarla.

Tabla 4.1. Glosario de conceptos de la ontología.

Concepto	Super-concepto	Definición	Fuente
Actividad	Concepto	Las actividades corresponden a la descomposición de las etapas del proceso de documentación que guían la producción y mantenimiento de la documentación del sistema. Cada actividad siempre asocia un conjunto de recursos humanos y tecnológicos para realizarla.	Adaptado de [69].

Ajuste	Proceso	Corresponde al conjunto de actividades diseñadas y ejecutadas de manera coherente a partir de las políticas y procedimientos de la empresa para documentar el desarrollo de un sistema software. Un proceso involucra un grupo de roles y herramientas para diseñar, desarrollar, implementar y mantener los artefactos de documentación necesarios para describir el sistema.	Adaptado de [33], [67].
Aplicación móvil	Artefacto	Corresponde a un artefacto de documentación basado en el desarrollo de una aplicación móvil capaz de presentar la documentación de un sistema software de manera amigable y fácilmente navegable. Sus principales ventajas son la accesibilidad permanente que ofrece a la documentación, notificaciones sobre actualizaciones y su fácil integración con recursos adicionales como videos o foros de soporte.	Fuente propia.
Artefacto	Concepto	Corresponde al artefacto de documentación que describe uno o varios aspectos del sistema. Dentro del proceso, pueden representar la salida o la entrada de las actividades y brindan una retroalimentación efectiva sobre lo que se está desarrollando.	Adaptado de [69].
Característica	Concepto	Corresponden a las propiedades particulares de un artefacto de documentación relacionadas con su contenido, su estructura y su accesibilidad que contribuyen a la utilidad y eficacia del documento para comunicar la información correspondiente.	Definición propia.
Categoría	Concepto	Corresponde a la agrupación de riesgos para la documentación que presenten características comunes con el fin de facilitar la comprensión, el análisis, la evaluación y el tratamiento de los riesgos a los que pueden estar expuestos la documentación.	Definición propia.
Contenido	Concepto	Corresponde a toda la información que se encuentra dentro de un artefacto de documentación compuesto por texto, gráficos u otros elementos que transmiten y comunican la información deseada a un público objetivo.	Definición propia.
Descripción	Concepto	Corresponde a la sección del contenido donde se presenta la información más relevante del artefacto la cual permite comunicar el mensaje deseado.	Definición propia.
Documento digital	Artefacto	Corresponde a una representación electrónica del contenido de un documento que existe en formato digital que puede ser almacenado, transmitido y visualizado en dispositivos electrónicos como computadores, teléfonos, tabletas, entre otros. Este tipo de documentación puede ser archivos de texto, videos, imágenes, presentaciones, archivos PDF o correos electrónicos, los cuales se crean, modifican y almacenan utilizando software y herramientas digitales.	Definición propia.
Documento físico	Artefacto	Corresponde a un documento impreso en un soporte físico como papel, película fotográfica o discos compactos. Este tipo de documentación se puede manipular y requiere de un almacenamiento en un espacio físico y tienen una capacidad limitada para ser modificados o actualizados.	Definición propia.
Documento relacionado	Esquema de uso	Corresponde a un artefacto de documentación que guarda una estrecha relación con otro artefacto específico mediante una referencia, una dependencia, una vinculación temática o una asociación. Esta relación puede ser necesaria para entender completamente el contenido o contexto del documento principal.	Definición propia.
Escala	Concepto	Corresponde a un conjunto de valores con propiedades definidas.	Tomado de [68].
Esquema de uso	Concepto	Corresponde a la sección del contenido donde se explica cómo utilizar correctamente el artefacto y como acceder rápidamente a la información que se contiene.	Definición propia.
Estrategia	Concepto	Corresponde al conjunto de acciones diseñadas de forma sistemática para reducir la probabilidad de ocurrencia de los riesgos para la documentación. La estrategia permite tomar decisiones basadas en la evaluación de la situación actual, la definición de objetivos, la identificación de recursos y la planificación de acciones.	Definición propia.

Etapa	Concepto	Corresponde a un periodo de tiempo dentro del proceso de documentación del sistema donde se realizan un conjunto de actividades particulares encaminadas a cumplir con un propósito específico en la creación, actualización o mantenimiento de los artefactos de documentación. De esta forma, las etapas se relacionan con los principales hitos de progreso de la documentación a lo largo de su ciclo de vida.	Adaptado de [70].
Factor de riesgo	Concepto	Corresponde a un elemento —tangible o intangible— que solo o en combinación con otros tiene el potencial intrínseco de originar un riesgo.	Tomado de [71].
Gestión de errores	Esquema de uso	Corresponde a la sección del esquema de uso donde se explican los problemas más comunes que puede enfrentar el usuario del artefacto y como solucionarlos. Esta sección puede orientar al usuario e identificar las áreas donde la documentación puede ser confusa o insuficiente.	Definición propia.
Glosario	Concepto	Corresponde a la sección del contenido donde se proporciona definiciones claras y concisas de los términos técnicos o especializados utilizados en el artefacto. Esta sección permite reducir la ambigüedad, mantener la coherencia y reducir el tiempo para comprender adecuadamente la información del artefacto.	Definición propia.
Herramienta	Recurso	Corresponde a una herramienta tecnológica —software o hardware— diseñada específicamente para facilitar y agilizar el proceso de creación, gestión o colaboración de la documentación; brindando funcionalidades para que los equipos de documentación puedan crear, organizar, editar, revisar y distribuir la documentación de manera más eficiente.	Definición propia.
Ilustraciones	Concepto	Corresponde a las secciones del artefacto que contienen representaciones visuales, gráficas o diagramas relevantes para complementar o explicar conceptos, procesos o información de manera más clara y comprensible para los usuarios del artefacto.	Definición propia.
Indicador	Medida	Corresponde a una medida que se deriva de otras medidas a través de un modelo de análisis como enfoque de medición.	Tomado de [68].
Índice	Concepto	Corresponde a la sección del contenido donde se enumera de manera ordenada los temas, secciones o elementos importantes de un artefacto con el propósito de facilitar la navegación, ofrecer una visión general del contenido, organizar la información de forma lógica y mejorar la usabilidad del artefacto.	Definición propia.
Instrucciones	Esquema de uso	Corresponde a la sección del esquema de uso del artefacto donde se describe un conjunto de indicaciones para orientar a los usuarios como utilizar el artefacto correctamente y aprovechar al máximo su contenido.	Definición propia.
Introducción	Concepto	Corresponde a la sección del artefacto donde se proporciona una visión general de su contenido, el propósito y el alcance del artefacto. De igual forma puede especificar cual es la audiencia objetivo y la estructura del contenido.	Definición propia.
Medición	Actividad	Corresponde al proceso de asignar valores numéricos a las características de la documentación para determinar su nivel de deuda con respecto a diferentes umbrales.	Adaptado de [68].
Medida	Concepto	Corresponde a una cuantificación del nivel de deuda en una o varias características de la documentación. Normalmente son de naturaleza cuantitativa pero también pueden ser cualitativa y ayudan a medir el progreso de la estrategia de documentación.	Adaptado de [33], [68].
Métrica	Concepto	Corresponde a una entidad cuantificable que permite la medición del logro de un objetivo. Una métrica se caracteriza por ser medible, procesable, relevante y oportuna; y se describe en términos de la unidad utilizada, la frecuencia de medición, el valor ideal, y los procedimientos para utilizarla e interpretarla.	Tomado de [33].
Navegación	Esquema de uso	Corresponde a una sección del esquema de uso donde se especifica como se relaciona el artefacto con otros artefactos para conectar diferentes partes del contenido, lo cual permite que los	Definición propia.

		usuarios puedan consultar la información que les interesa y moverse de manera fluida por el contenido.	
Página web	Artefacto	Corresponde a un artefacto de documentación basado en el desarrollo de una página web interna o externa para presentar la documentación de un sistema software a los miembros del equipo y los usuarios finales. De esta forma es posible mejorar la accesibilidad a la documentación, se facilitan los procesos de seguimiento y actualización y se mejora la comprensión gracias al aprovechamiento de las herramientas multimedia.	Definición propia.
Plantilla	Concepto	Corresponde a un artefacto de documentación predefinido que contiene un diseño y un formato preestablecidos de acuerdo a los estándares de documentación de la empresa. Su propósito es servir como punto de partida para crear nuevos documentos y mantener una consistencia visual y estructural en toda la documentación para cumplir con dichos estándares.	Definición propia.
Portada	Concepto	Corresponde a la primera página del contenido del artefacto donde se refleja la identidad de la empresa y del sistema. Una portada ayuda a los usuarios a comprender rápidamente de qué se trata el documento y cuál es su propósito.	Definición propia.
Proceso	Producto	Corresponde a un conjunto de actividades influenciadas por las políticas y procedimientos de la empresa que se ejecutan de manera sistemática para cumplir con uno o varios objetivos estratégicos. Para esto, se definen roles con funciones y responsabilidades claras que manipulan diferentes entradas para producir uno o varios productos de trabajo requeridos para crear, organizar, editar, revisar y distribuir la documentación.	Adaptado de [33], [67].
Producto	Concepto	Corresponde a un entregable de tipo software o proceso que se desarrolla, se despliega y se mantiene. Pueden ser de entrada o de salida; obligatorio u opcional. Por lo general, los productos son tangibles y producidos, consumidos o modificados por tareas.	Tomado de [67].
Protocolo	Concepto	Corresponde a un conjunto de pautas, normas o procedimientos que especifican como crear, organizar, almacenar, mantener y distribuir los artefactos de documentación. Su propósito es garantizar que la documentación sea clara, precisa, coherente y accesible para los usuarios.	Definición propia.
Recurso	Concepto	Corresponde a cualquier activo empresarial que pueda ayudar a la empresa a lograr los objetivos de documentación. De esta forma, los dos recursos más importantes en el proceso de documentación son los roles y las herramientas.	Tomado de [33], [67].
Restricciones	Concepto	Corresponde a la sección del contenido donde se describen las limitaciones o condiciones especiales que se deben tener en cuenta sobre la información del artefacto.	Definición propia.
Riesgo	Concepto	Corresponde a la posibilidad de que suceda un evento o situación que ocasione un impacto positivo o negativo en el logro de los objetivos del proceso de documentación.	Adaptado de [71].
Rol	Recurso	Corresponde a un conjunto de responsabilidades, deberes y habilidades requeridas para realizar una actividad específica.	Tomado de [71].
Software	Producto	Corresponde a una colección de programas e instrucciones utilizados para operar y controlar el funcionamiento de dispositivos electrónicos y otros sistemas informáticos.	Adaptado de [33].
Tarea	Concepto	Corresponde a un elemento del proceso que define el trabajo realizado por un rol. Se relaciona con las actividades que se desarrollan en cada etapa del proceso de documentación.	Adaptado de [67].

Tabla 4.2. Relaciones en la ontología.

Relación	Conceptos	Descripción
afecta	Riesgo → Artefacto	Cuando se materializan uno o muchos riesgos de la documentación pueden afectar a uno o varios artefactos de documentación.

analiza	Estrategia → Riesgo	La estrategia diseñada para solucionar los problemas en los artefactos de documentación analiza los riesgos que los ocasionaron.
analiza	Estrategia → Indicador	La estrategia diseñada para solucionar los problemas en los artefactos de documentación analiza los indicadores de la medición de la deuda.
asocia	Indicador → Medida	Uno o varios indicadores utilizados en el proceso de medición asocian una o varias medidas del nivel de deuda.
asocia	Medida → Métrica	Cada medida que se utiliza para los indicadores asocia una o varias métricas para cuantificar el nivel de deuda.
asocia	Ajuste → Estrategia	Cada proceso de ajuste de la documentación asocia una o varias estrategias para gestionar la deuda.
consta de	Proceso → Etapa	Un proceso está conformado o compuesto por una o varias etapas.
define	Protocolo → Proceso	Uno o varios protocolos definen el diseño y ejecución de uno o varios procesos de la documentación.
describe	Artefacto → Producto	Uno o varios artefactos de documentación describen pueden describir uno o varios productos construidos.
es responsable	Rol → Actividad	Uno o varios roles pueden ser responsable de la ejecución de una o varias actividades del proceso de documentación.
establece	Medición → Indicador	Cada proceso de medición establece uno o varios indicadores que se van a evaluar.
establece	Protocolo → Característica	Uno o varios protocolos establecen una o varias características que se manejarán en la documentación.
estandariza	Protocolo → Plantilla	Uno o varios protocolos permiten estandarizar una o varias plantillas para crear los artefactos de documentación.
evalúa	Métrica → Característica	Una métrica evalúa una o muchas características de los artefactos de documentación.
genera	Actividad → Artefacto	Es posible que una o varias actividades del proceso de documentación no generen ningún artefacto o generen varios.
genera	Factor de riesgo → Riesgo	Uno o varios factores de riesgo pueden contribuir a la generación de uno o varios riesgos para la documentación.
genera	Proceso → Producto	Un proceso puede generar cero o muchos productos.
relaciona	Característica → Característica	Es posible que cero o varias características de la documentación se relacionen con otras características.
relaciona	Artefacto → Artefacto	Es posible que cero o varios artefactos de documentación se relacionen con otros artefactos.
relaciona	Actividad → Actividad	Una actividad del proceso de documentación puede estar relacionada con otras actividades.
relaciona	Etapa → Etapa	Una etapa del proceso de documentación puede estar relacionada con otras actividades.
relaciona	Proceso → Proceso	Un proceso puede estar relacionado con otro proceso.
relaciona	Riesgo → Riesgo	Un riesgo para la documentación de software puede estar relacionado con otros riesgos.
relaciona	Factor de riesgo → Factor de riesgo	Un factor de riesgo para la documentación de software puede estar relacionado con otros factores de riesgo.
relaciona	Métrica → Métrica	Una métrica puede estar relacionada con muchas métricas o con ninguna.
se compone de	Etapa → Actividad	Una etapa del proceso de documentación se compone de una o varias actividades de documentación.
se compone de	Artefacto → Contenido	Un artefacto de documentación está compuesto por un contenido específico para ese artefacto.
se compone de	Contenido → Portada	El contenido de un artefacto de documentación se compone de una portada.
se compone de	Contenido → índice	El contenido de un artefacto de documentación se compone de un índice.
se compone de	Contenido → Introducción	El contenido de un artefacto de documentación se compone de una introducción.

se compone de	Contenido → Descripción	El contenido de un artefacto de documentación se compone de una descripción.
se compone de	Contenido → Restricciones	El contenido de un artefacto de documentación se compone de una o varias restricciones.
se compone de	Contenido → Glosario	El contenido de un artefacto de documentación puede tener un glosario.
se compone de	Contenido → Ilustraciones	El contenido de un artefacto de documentación puede tener una o varias ilustraciones.
se compone de	Contenido → Esquema de uso	El contenido de un artefacto de documentación puede tener un esquema de uso.
se divide en	Actividad → Tarea	Una actividad del proceso de documentación se divide en una o varias tareas.
se organizan	Riesgo → Categoría	Uno o varios riesgos para la documentación se organizan en una o varias categorías
soporta	Herramienta → Tarea	Una o varias herramientas tecnológicas soportan la realización de una o varias tareas.
supervisa	Indicador → Riesgo	Uno o varios indicadores supervisan la probabilidad de ocurrencia de uno o varios riesgos.
tiene	Artefacto → Característica	Uno o varios artefactos de documentación tienen una o varias características.
tiene	Esquema de uso → Instrucciones	El esquema de uso de cada artefacto de documentación tiene una o varias instrucciones de uso.
tiene	Esquema de uso → Navegación	El esquema de uso de cada artefacto de documentación tiene un esquema de navegación.
tiene	Esquema de uso → Gestión de errores	El esquema de uso de cada artefacto de documentación tiene una o varias indicaciones para gestionar los posibles errores de uso.
tiene	Esquema de uso → Documento relacionado	El esquema de uso de cada artefacto de documentación podría tener otros artefactos de documentación relacionados.
utiliza	Artefacto → Plantilla	Un artefacto de documentación utiliza una o varias plantillas de documentación estandarizadas.
utiliza	Actividad → Artefacto	Una actividad del proceso de documentación podría utilizar los artefactos que se han generado en otras actividades.
utiliza	Proceso → Recurso	Los procesos diseñados para documentar el software utilizan uno o varios recursos humanos y tecnológicos.
utilizan	Medida → Escala	Una o varias medidas obtenidas durante el proceso de medición utilizan una escala para clasificarlas.
utiliza	Actividad → Recurso	Una o varias actividades pueden utilizar cero o muchos recursos.

4.6. Discusión de algunos conceptos y sus relaciones

Es pertinente que algunos de los conceptos utilizados en la ontología propuesta sean discutidos de manera más profunda mediante el análisis y la ampliación de su definición. En ese sentido, en esta sección se profundizará en los siguientes términos:

Riesgo: Corresponde a cualquier evento que puede afectar la calidad de la documentación de un sistema software que en el futuro puede comprometer la calidad del software, la productividad del equipo de desarrollo, y la capacidad del software para escalar o ser mantenible [72]. Uno de los riesgos más significativos y comunes en la documentación es que se vuelva obsoleta a medida que el software evoluciona. Esto puede llevar a que la documentación tenga altos niveles de incoherencia con respecto al funcionamiento real del sistema, ser insuficiente o, por el contrario, ser sobreabundante.

Protocolo: Corresponde a un conjunto de reglas, pautas y estándares establecidos en una empresa desarrolladora de software para la creación, organización y mantenimiento de la documentación relacionada con un proyecto software. Su propósito es garantizar la

consistencia, claridad y eficiencia de la documentación para facilitar su comprensión, uso y mantenimiento. Por otro lado, un protocolo ayuda a mantener la coherencia en la documentación a lo largo del tiempo, permitiendo la colaboración entre desarrolladores, usuarios y otros interesados [73].

4.7. Verificación de la ontología

De acuerdo con lo propuesto en [74], una ontología debe verificarse con base en los siguientes aspectos: *consistencia*, si es posible inferir conocimiento desde la ontología; *completitud*, si es posible probar que la ontología incluye todas las clases, propiedades y relaciones necesarias para describir de manera exhaustiva y precisa un dominio o área de conocimiento particular; y *concisión*, si es posible demostrar que la ontología no es redundante y no tiene definiciones innecesarias.

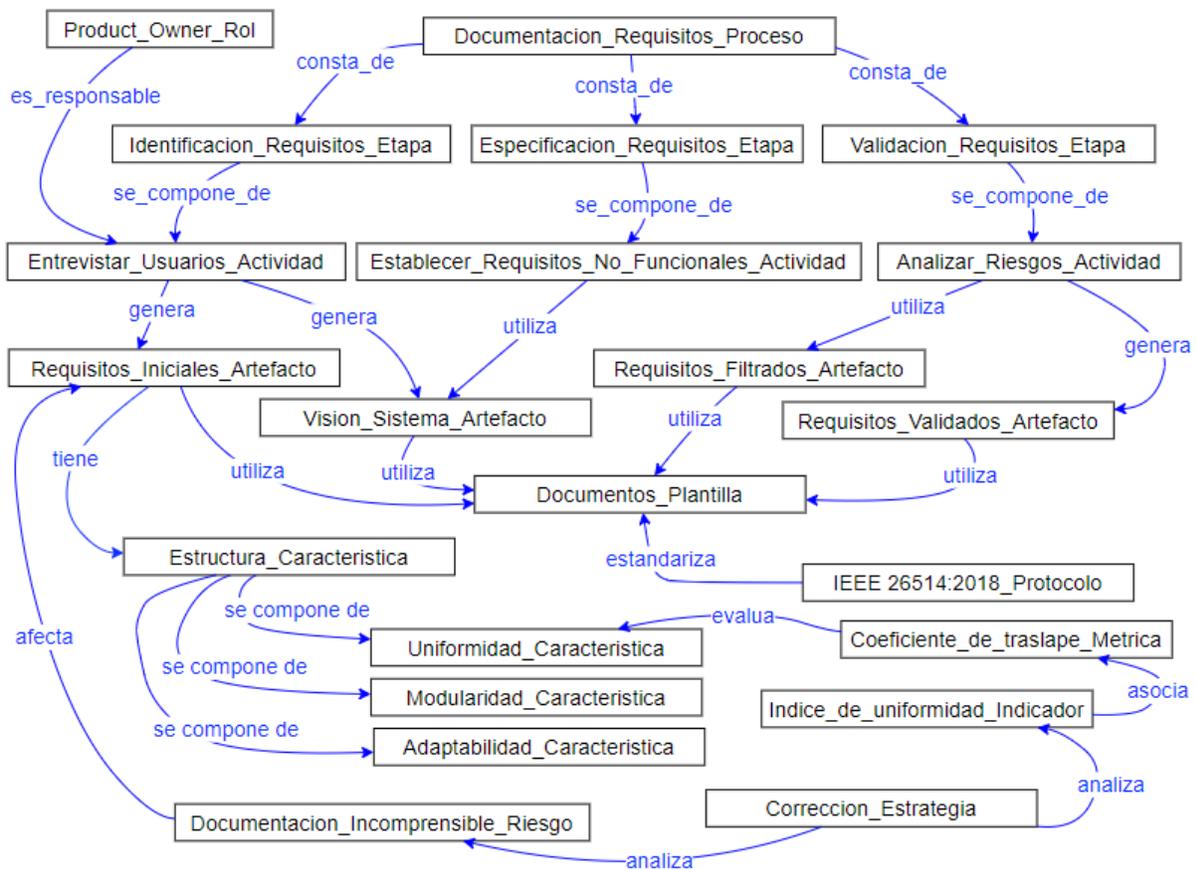


Figura 4.3. Extracto de la instancia del proceso de documentación de requisitos.

En ese sentido, para verificar que la ontología propuesta organiza de manera formal y estructurada el conocimiento relacionado con la deuda de la documentación en el desarrollo ágil de software, ha sido instanciada con los elementos de proceso más importantes durante la documentación de los requisitos de un sistema software, los cuales han sido mapeados en los conceptos: Etapa, Actividad, Rol, Artefacto, Plantilla, Característica y Protocolo, Métrica, Indicador, Estrategia y Riesgo. La Figura 4.3 presenta un extracto de la instancia donde se puede observar que el proceso de documentación de requisitos está compuesto por un conjunto de etapas como la identificación, especificación, validación, entre otros. Asimismo,

cada etapa implica la ejecución de una serie de actividades por parte de uno o varios roles en las cuales se generan diferentes artefactos de documentación. Para construir dichos artefactos se pueden utilizar plantillas que han sido estandarizadas previamente por la empresa a partir de uno o varios protocolos de documentación con el propósito de homogenizar el proceso de documentación. Finalmente, cada artefacto que se genera puede verse afectado por uno o varios riesgos los cuales se encuentran asociados a las características del artefacto y que son evaluados a partir de una o varias métricas que establecen una serie de indicadores capaces de apoyar la definición de estrategias que permitan disminuir su probabilidad de ocurrencia.

Por otro lado, la Figura 4.4 muestra un ejemplo de los aspectos relacionados en la medición de una característica en un artefacto de documentación utilizando una métrica asociada a dicha característica. En este caso, una historia de usuario debe describir una funcionalidad específica del sistema software, la cual debe estar construida según la plantilla que haya definido la organización para la construcción de historias de usuario con base en los protocolos de documentación. Por lo tanto, estos protocolos establecen las características de la documentación y que posteriormente podrán ser medidas por una o varias métricas.

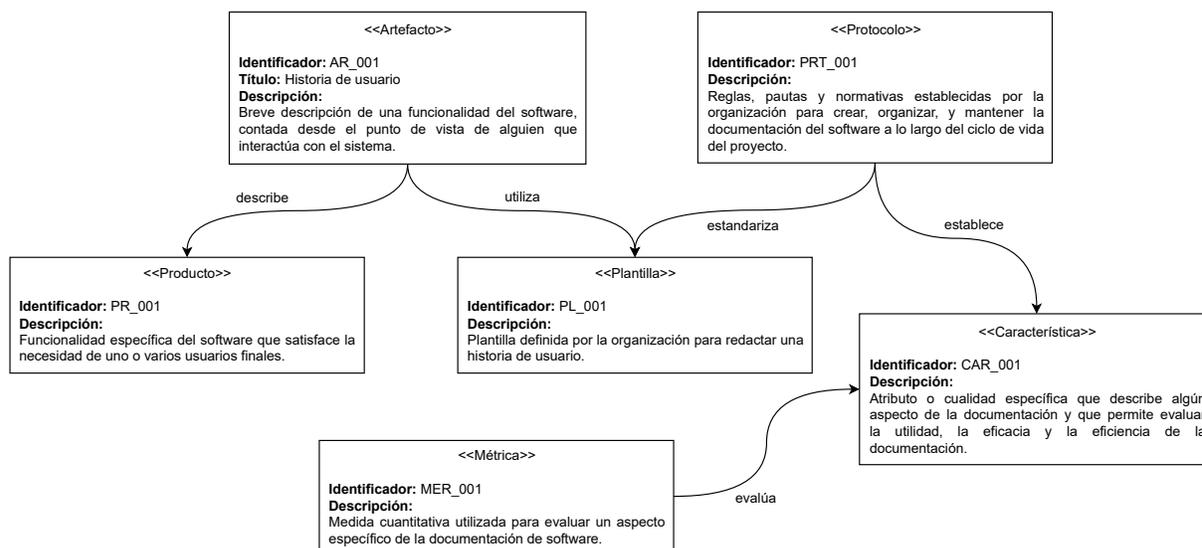


Figura 4.4. Ejemplo de instanciación de la medición de una característica.

Es importante aclarar que, si bien la ontología propuesta no está finalizada y validada por expertos en documentación de software o similares, los ejemplos presentados en las Figuras 4.3 y 4.4 permiten identificar su potencial para estructurar y comprender correctamente el contexto de este tipo de deuda y mejorar su gestión. Por otro lado, a medida que la ontología evolucione podría integrarse con los procesos de desarrollo para fortalecer su utilidad práctica ya que su carácter incompleto destaca la necesidad de un esfuerzo continuo por mejorar la documentación en el ámbito del software.

4.8. Conclusiones y trabajos futuros

En este capítulo se presentó una versión inicial de una ontología diseñada para organizar el conocimiento en torno a la deuda de la documentación en proyectos de software ágil. Esta propuesta representa un punto de partida esencial para abordar esta importante problemática en el desarrollo ágil de software estableciendo las categorías principales de este tipo de deuda, las relaciones entre ellas y la estructura semántica para comprender y gestionar de manera

efectiva este tipo de deuda en un contexto ágil. Como trabajo futuro, se espera que la ontología evolucione enriqueciéndola con más conceptos, detalles, ejemplos específicos y estrategias para mejorar su precisión. Adicionalmente, se espera que dicha evolución pueda recibir retroalimentación por parte de la comunidad ágil y profesionales de desarrollo para que con cada iteración se mejore la comprensión y gestión de este tipo de deuda en los proyectos ágiles. Otro aspecto fundamental que se espera desarrollar en el futuro es la validación teórica de la ontología a través de métodos formales como preguntas de competencia, consultas con SPARQL y pruebas de calidad o utilidad.

Capítulo 5. Modelo de métricas para medir la deuda de la documentación

En este capítulo se presenta el modelo de métricas para evaluar la deuda de la documentación en el desarrollo ágil de software, el cual está asociado a las características de la documentación que fueron descritas en el Capítulo 3 de este documento. En la Figura 5.1 se presenta el modelo de métricas, el cual está compuesto de tres elementos importantes, ellos son: (i) dimensiones de la documentación, las cuales permiten analizar la deuda de la documentación desde cinco diferentes perspectivas; (ii) características de la documentación, que corresponden a los diecisiete atributos de calidad de la documentación propuestos; y (iii) métricas, que corresponden a treinta y nueve métricas asociadas a las características de la documentación sugeridas. Adicionalmente, el modelo propone la definición de quince riesgos que pueden ocasionar deuda en la documentación, los cuales fueron obtenidos a partir de las causas y efectos descritos en el Capítulo 2, sección 2.2.2.4. Sin embargo, es importante aclarar que las métricas genéricas y pueden aplicarse por igual a la documentación generada en enfoques tradicionales y ágiles.

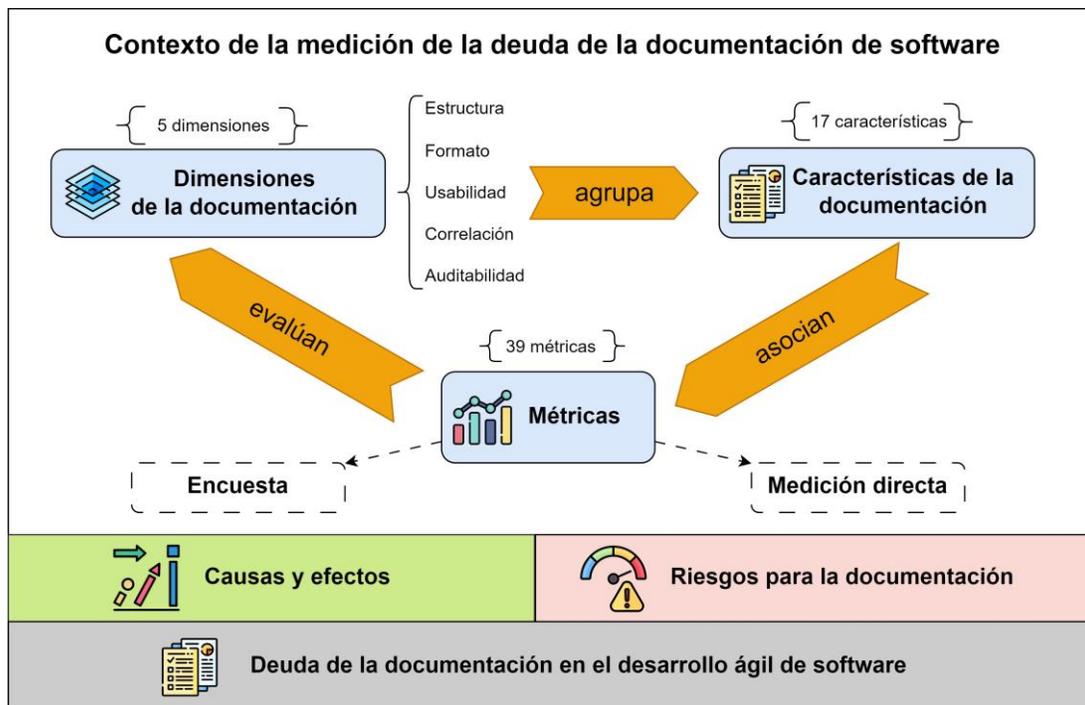


Figura 5.1. Resumen de los elementos de la propuesta.

La construcción del modelo de métricas siguió el proceso presentado en la Figura 5.2, el cual fue diseñado para organizar los pasos que permitieron la definición del modelo a partir de la caracterización de la deuda de la documentación en el desarrollo ágil de software. A continuación, en la siguiente sección se explican los detalles de cada paso del proceso.

5.1. Proceso de construcción

El proceso presentado en la Figura 5.2 consta de cuatro fases en las cuales se ejecutaron una serie de actividades enfocadas en recolectar, analizar y estructurar toda la información necesaria para construir el modelo de métricas. A continuación, se explican las fases del proceso, las actividades que componen cada fase y su importancia para la definición del modelo de métricas.

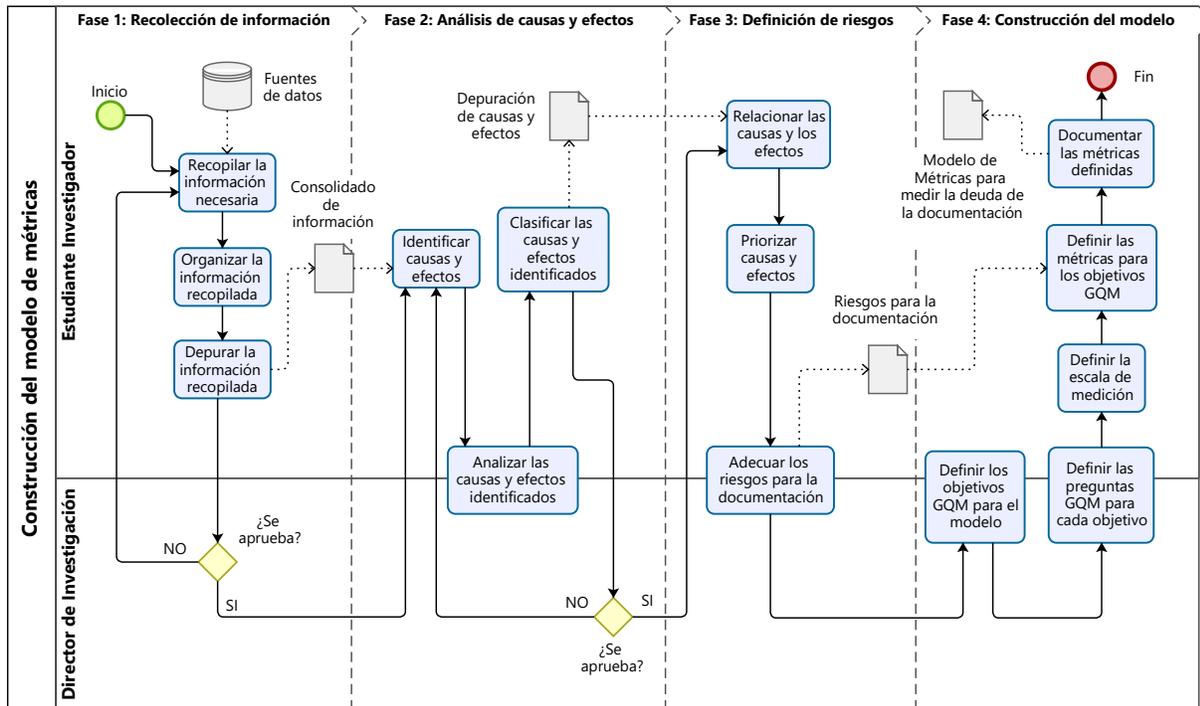


Figura 5.2. Proceso general para la construcción del modelo de métricas.

5.1.1. Fase 1: Recolección de información

En esta fase se utilizó la información recolectada durante el análisis del estado del arte descrito en el Capítulo 2, sección 2.2 y la caracterización de la documentación descrita en el Capítulo 3 de este documento. Dichos resultados permitieron entender el contexto de la documentación de software, y en especial; las causas que la originan y los efectos que se producen cuando se materializa la deuda (ver sección 2.2.2.4). Teniendo en cuenta lo anterior, se diseñó un conjunto de métricas relevantes, útiles y confiables para medir el nivel de deuda en la documentación de software de una manera objetiva. Como se muestra en la Figura 5.2, se estableció un ciclo de refinamiento de la información recolectada por parte del rol *Estudiante Investigador* con el apoyo del rol *Director de Investigación* para generar un *consolidado de la información* que posteriormente se utilizó como insumo para la Fase 2.

5.1.2. Fase 2: Análisis de causas y efectos

A partir de la información consolidada de la Fase 1, se identificó un total de 45 causas que contribuyen a la ocurrencia del fenómeno de la deuda de documentación, y 42 efectos o consecuencias que suceden como respuesta a dichas causas como se describe en el Capítulo 2, sección 2.2.2.4. Posteriormente, esta información fue analizada y depurada con el apoyo

del *Director de Investigación*, esto con el objetivo de clasificar las causas y los efectos identificados mediante un ciclo de refinamiento con posteriores aprobaciones por parte del *Director de Investigación*. Como resultado de esta etapa, se obtuvo un listado depurado de causas y efectos que posteriormente fue utilizado como insumo para la Fase 3.

5.1.3. Fase 3: Definición de riesgos

En esta fase del proceso, se utilizó el listado depurado de causas y efectos producto de la Fase 2, esto con el objetivo de construir un conjunto de riesgos que pueden generar deuda en la documentación de software, sin embargo, muchas de las causas y efectos identificados presentaban problemas de ambigüedad, para solucionar estos inconvenientes, fue necesario analizar y relacionar las causas y los efectos y de esta manera eliminar coincidencias o similitudes, priorizar los resultados para establecer su importancia, y finalmente; construir una definición homogénea de los riesgos que podrían afectar a la documentación de software siguiendo el formato presentado en la Tabla 5.1. De esta forma, cada riesgo fue estandarizado para facilitar su análisis y comprensión, permitiendo además que tengan un contexto más preciso compuesto por aquello que origina el riesgo, quien o quienes pueden verse afectados directamente cuando el riesgo se manifiesta y cuál es su efecto sobre el proyecto cuando se materializa el riesgo. Esta adecuación se presenta en la Tabla 5.2, donde se describe cada riesgo utilizando el formato de la Tabla 5.1 y las características de la documentación con las que más podría estar relacionado cada riesgo; las cuales permiten comprender los diferentes aspectos de la documentación que podrían materializar el riesgo.

Tabla 5.1. Formato para la definición de riesgos.

Para (quien o quienes se ven afectados por la ocurrencia del riesgo) + **existe el riesgo** (especificación del riesgo) + **debido a** (una o varias causas del riesgo) + **esto puede** (las consecuencias de la materialización del riesgo)

Tabla 5.2. Listado de riesgos propuesto para el modelo de métricas.

Id	Descripción	Características asociadas
R1	Para los equipos de desarrollo y de mantenimiento, existe el riesgo de no comprender correctamente el código fuente del sistema en el futuro, esto, debido a que no tiene comentarios que respalden la explicación de su intención y la documentación técnica es escasa, incomprensible, desactualizada o nula. Esto puede afectar la mantenibilidad y modificabilidad del sistema, el trabajo en equipo, la transferencia de conocimientos a nuevo personal, y la resolución de problemas.	<ul style="list-style-type: none"> • Uniformidad • Claridad • Legibilidad • Integridad • Precisión • Modularidad
R2	Para los equipos de desarrollo, los administradores y usuarios finales, existe el riesgo que el sistema sea altamente vulnerable, esto, debido a que su documentación no describe correctamente su arquitectura, diseño, funcionalidades, interfaces y restricciones. Esto puede afectar la comprensión de riesgos y amenazas, dificultar la implementación o mejora de prácticas de seguridad, y aumentar la complejidad en la gestión y atención de incidentes.	<ul style="list-style-type: none"> • Legibilidad • Conciso • Claridad • Consistencia • Verificabilidad
R3	Para la empresa existe el riesgo que el sistema sea muy difícil de mantener en el futuro debido a que su documentación no permite comprender adecuadamente su funcionamiento, su arquitectura, su diseño y las decisiones que se tomaron durante su construcción. Esto puede afectar negativamente la refactorización del código fuente, disminuye la coherencia en el desarrollo del sistema, dificulta la trazabilidad de los cambios e incrementa el esfuerzo necesario para identificar, comprender y solucionar los incidentes.	<ul style="list-style-type: none"> • Uniformidad • Claridad • Consistencia • Actualidad • Trazabilidad • Validez • Verificabilidad
R4	Para la empresa existe el riesgo que el sistema sea de baja calidad, esto, debido a que su documentación no ofrece información clara y precisa que garantice la comprensión de su diseño, su arquitectura y su funcionamiento. Esto puede dificultar el control de calidad del sistema, su escalabilidad y mantenibilidad, la	<ul style="list-style-type: none"> • Legibilidad • Conciso • Claridad • Accesibilidad

	transferencia de conocimiento a nuevo personal, y aumentar la dependencia del personal que participó en su desarrollo inicialmente. Por otro lado, el sistema puede causar insatisfacción en los interesados y hacer que la empresa deteriore sus relaciones comerciales perdiendo competitividad en el mercado.	<ul style="list-style-type: none"> • Confidencialidad • Verificabilidad
R5	Para la empresa existe el riesgo que se generen reprocesos durante el desarrollo del sistema, esto, debido a que su documentación no es clara, está incompleta, es inconsistente o está desactualizada. Esto puede generar malos entendidos en la interpretación de los requisitos, se incrementa la probabilidad de cometer errores, incluso omitir información importante durante el diseño o la implementación. Esto obliga a realizar revisiones y reprocesos para corregir errores o incorporar información faltante, lo cual puede retrasar las entregas.	<ul style="list-style-type: none"> • Legibilidad • Conciso • Claridad • Accesibilidad • Confidencialidad • Integridad • Validez • Verificabilidad
R6	Para la empresa existe el riesgo de acumular progresivamente niveles inmanejables de deuda técnica, esto, debido a la poca claridad en la documentación, la ausencia de buenas prácticas y estándares, falta de detalles sobre las dependencias y riesgos, y documentación inexistente o desactualizada del diseño y la arquitectura. Esto puede llevar a decisiones apresuradas o incorrectas durante el desarrollo, lo que resulta en soluciones subóptimas y de baja calidad.	<ul style="list-style-type: none"> • Uniformidad • Claridad • Accesibilidad • Confidencialidad • Consistencia • Portabilidad
R7	Para la empresa existe el riesgo que los procesos de desarrollo y de calidad del sistema tengan un alto grado de informalidad, esto, debido a que la documentación no registra claramente los lineamientos, las pautas y los estándares que se deben seguir en cada etapa. Esto puede llevar a una falta de uniformidad y coherencia en la ejecución, seguimiento y control de los procesos, debido a la ambigüedad en la definición de los roles y responsabilidades. Por otro lado, se incrementa la probabilidad de perder conocimiento valioso debido a que los procesos se ejecutan con base en la intuición en lugar de seguir una guía documentada y registrar las lecciones aprendidas en cada paso.	<ul style="list-style-type: none"> • Uniformidad • Claridad • Accesibilidad • Confidencialidad • Precisión • Portabilidad • Verificabilidad
R8	Para la empresa existe el riesgo de no mejorar sus procesos debido a que la documentación no registra adecuadamente como se deben ejecutar para que sean eficientes, consistentes y mejoren la calidad de las operaciones. Esto puede afectar la visibilidad de los procesos, el análisis y la medición de su desempeño, incrementa la incertidumbre, y limita la capacidad de mejora continua al no existir una base sólida de buenas prácticas y lecciones aprendidas.	<ul style="list-style-type: none"> • Claridad • Accesibilidad • Confidencialidad • Consistencia
R9	Para los equipos encargados de implementar el sistema, existe el riesgo que el sistema no se pueda implementar correctamente en su ambiente, esto, debido a que la documentación no cuenta con la información precisa y completa para comprender su funcionamiento y cómo se debe instalar, configurar y utilizar correctamente. Esto puede afectar la planificación de la implementación, cometer errores u omisiones de etapas o actividades durante el proceso y dificulta la transferencia de conocimiento o el funcionamiento del sistema luego de su implementación.	<ul style="list-style-type: none"> • Accesibilidad • Confidencialidad • Consistencia • Integridad • Verificabilidad
R10	Para la empresa existe el riesgo de perder el conocimiento aprendido por los miembros del equipo de desarrollo, esto, debido a documentación deficiente, incompleta o inexistente donde se registre información de aspectos relevantes como detalles y decisiones de diseño, funcionalidades, configuración, integraciones, gestión de requisitos, soluciones a problemas y lecciones aprendidas. Esto puede incrementar la dependencia del conocimiento tácito de ciertos miembros del equipo, el esfuerzo para adquirir nuevo conocimiento y se disminuye la productividad del equipo.	<ul style="list-style-type: none"> • Modularidad • Accesibilidad • Confidencialidad • Integridad • Verificabilidad
R11	Para el equipo de desarrollo existe el riesgo de una mala implementación de los requisitos del sistema, esto, debido a que la documentación los describe con ambigüedades. Esto puede ocasionar que los requisitos sean mal interpretados, inconsistentes o contradictorios entre si causando confusión durante el diseño y el desarrollo del sistema. Por otro lado, aumenta la probabilidad de omitir detalles importantes en las funcionalidades clave, restricciones, reglas de negocio o requisitos no funcionales, sobrecargar al equipo y reducir su productividad.	<ul style="list-style-type: none"> • Uniformidad • Claridad • Accesibilidad • Confidencialidad • Consistencia • Precisión • Verificabilidad
R12	Para la empresa existe el riesgo de no integrar eficazmente a nuevo personal al equipo de desarrollo, esto, debido a que la documentación del sistema no permite comprender adecuadamente el sistema, su estado actual, su funcionamiento, los requisitos que debe satisfacer, su diseño y su arquitectura. Esto puede prolongar	<ul style="list-style-type: none"> • Legibilidad • Conciso • Accesibilidad

	la curva de aprendizaje para comprender los procesos, las políticas y los procedimientos, así como retrasar la adaptación del nuevo personal al contexto del sistema para contribuir plenamente al cumplimiento de los objetivos del equipo.	<ul style="list-style-type: none"> • Confidencialidad • Consistencia
R13	Para el equipo de control de calidad existe el riesgo que sea muy difícil determinar qué aspectos del sistema deben ser probados, esto, debido a que la documentación presenta información ambigua o incompleta de los requisitos y no explica claramente cuáles son los criterios de éxito bajo los cuales se consideran cumplidos o hechos. Esto puede llevar al diseño de casos de prueba poco efectivos que no cubran todos los escenarios y datos de prueba relevantes, gestionar de forma inadecuada los defectos del sistema, perder la trazabilidad para determinar que partes del sistema se han probado y que casos de prueba certifican el cumplimiento o no de los requisitos.	<ul style="list-style-type: none"> • Uniformidad • Modularidad • Legibilidad • Conciso • Accesibilidad • Confidencialidad • Precisión • Integridad • Actualidad • Trazabilidad • Validez • Verificabilidad
R14	Para la empresa existe el riesgo de perder la capacidad de realizar un seguimiento adecuado de la evolución y los cambios realizados a lo largo del tiempo en uno o varios artefactos de documentación debido a la deficiencia o ausencia de un control de versiones en la información de cada artefacto. Esto puede afectar la gestión del historial de versiones, dificultar la comprensión del contexto o la necesidad de las modificaciones, generar confusión y pérdida de información.	<ul style="list-style-type: none"> • Modularidad • Accesibilidad • Confidencialidad • Consistencia • Precisión • Actualidad • Trazabilidad • Validez • Verificabilidad
R15	Para el equipo de desarrollo existe el riesgo que se generen problemas de comunicación debido a que la documentación es ambigua, poco clara o desactualizada en la descripción del sistema, de sus funcionalidades y sus condiciones de uso. Esto puede llevar a interpretaciones erradas, malentendidos, y confusiones en las comunicaciones internas y externas, afectando de igual forma la colaboración y la resolución de problemas con base en información obsoleta.	<ul style="list-style-type: none"> • Uniformidad • Modularidad • Legibilidad • Conciso • Claridad • Accesibilidad • Consistencia • Precisión

5.1.4. Fase 4: Construcción del modelo de métricas

En esta fase se construyen las métricas para las características de la documentación relacionadas con cada uno de los riesgos descritos en la Tabla 5.2. Inicialmente, se muestra en la Tabla 5.3 el glosario de los elementos que componen el modelo de métricas, posteriormente se presenta una breve descripción del paradigma Goal-Question-Metric (GQM), luego se explica el propósito, el alcance y las limitaciones del modelo, y, por último; se presentan las métricas construidas a partir del paradigma GQM.

Tabla 5.3. Glosario de conceptos importantes. Tomado de [75].

Concepto	Descripción	Ejemplos
Entidad	Es un objeto, tangible o intangible, que va a ser caracterizado mediante la medición de sus atributos.	Documento físico o digital
Atributo	Es una característica de una entidad que puede distinguirse cuantitativa o cualitativamente por medios humanos o automatizados.	Uniformidad, Legibilidad, Integridad
Medición	Corresponde a la ejecución de una serie de operaciones para determinar el valor de una medida.	Aplicación de una métrica para determinar la legibilidad de un documento físico o digital
Medida	Es una variable a la que se le asigna el resultado cuantitativo de una medición.	El porcentaje de documentos que presentan baja legibilidad
Medida base	Medida de un atributo que no depende de otras medidas.	CP (Cantidad de palabras)

Medida derivada	Medida que se obtiene a partir de la relación entre dos o más medidas base, medidas provenientes de otras métricas o transformando una medida base usando una función matemática.	CPF (Cantidad de Palabras por Frase)
Métrica	Indica la manera como se va a obtener el valor de una medida. Se compone de la forma de medir (función matemática, método de medición, etc.) y de la escala de medición para interpretar los resultados.	Corresponde a la función matemática para calcular la medida
Métrica directa	Es una métrica que no depende de ninguna otra métrica.	CP = 6
Métrica indirecta	Es una métrica que depende de las medidas que se obtienen a partir de otras métricas.	CP = CPF * 5
Indicador	Es una medida derivada que provee una estimación del atributo que se está midiendo y que al compararse con algún nivel de referencia señalaría una desviación sobre la cual se pueden tomar acciones correctivas o preventivas según sea el caso.	NLEG (Nivel de Legibilidad)
Escala	Es un conjunto ordenado de valores, continuo o discreto, o un conjunto de categorías a las que se asigna el atributo.	Escala del nivel de transparencia (bajo, medio, alto, muy alto)
Unidad de medición	Es una cantidad particular definida y adoptada por convención, con la que se comparan otras entidades del mismo tipo para expresar su magnitud en relación con esa cantidad.	Cantidad de palabras, Número de enlaces entre documentos

5.1.4.1. Generalidades sobre el paradigma GQM

El modelo fue construido siguiendo los lineamientos del paradigma GQM propuesto por Basili et al. [32], el cual permite definir e interpretar modelos de medición basados en objetivos. En este sentido, GQM propone establecer objetivos de medición claros los cuales serán analizados a través de preguntas potencialmente medibles, y construir métricas que intentarán dar respuesta a dichos objetivos.

A continuación, una breve descripción de los niveles que considera GQM para establecer un modelo de medición basado en objetivos:

1. **Nivel conceptual (Objetivos):** Corresponde a la definición de metas claras y específicas que se desean alcanzar con el modelo en términos de calidad, rendimiento o cualquier otro aspecto relevante con respecto a:
 - a. **Productos:** Hace referencia a los artefactos, entregables y documentos que se producen durante el ciclo de vida de un sistema como: especificaciones, diseños, programas, unidades de prueba, entre otros.
 - b. **Procesos:** Hace referencia a las actividades que permiten la creación de los productos como: especificar, diseñar, implementar, entre otros.
 - c. **Recursos:** Hace referencia a los elementos utilizados por los procesos para producir sus salidas, como el personal, el hardware, el software, los espacios de oficina, etc.
2. **Nivel operacional (Preguntas):** Corresponde al desarrollo de preguntas específicas que ayuden a comprender y evaluar si se están logrando los objetivos establecidos. Estas preguntas deben ser concretas y medibles.
3. **Nivel cuantitativo (Métricas):** Corresponde a la definición de métricas e indicadores para medir el progreso hacia el logro de los objetivos y responder a las preguntas planteadas. Las métricas pueden ser de carácter:

- a. **Objetivo:** si dependen exclusivamente del objeto de medición y no del punto de vista con el que se toman. Por ejemplo: número de versiones de un documento, horas trabajadas, tamaño del programa, etc.
- b. **Subjetivo:** si dependen del punto de vista con el que se tomaron y del objeto de medición. Por ejemplo: nivel de satisfacción del usuario o legibilidad de un texto.

La Figura 5.2 presenta la estructura general del paradigma GQM partiendo desde los objetivos de medición, los cuales se descomponen en preguntas para reducir su ambigüedad y facilitar la definición de métricas capaces de verificar su cumplimiento.

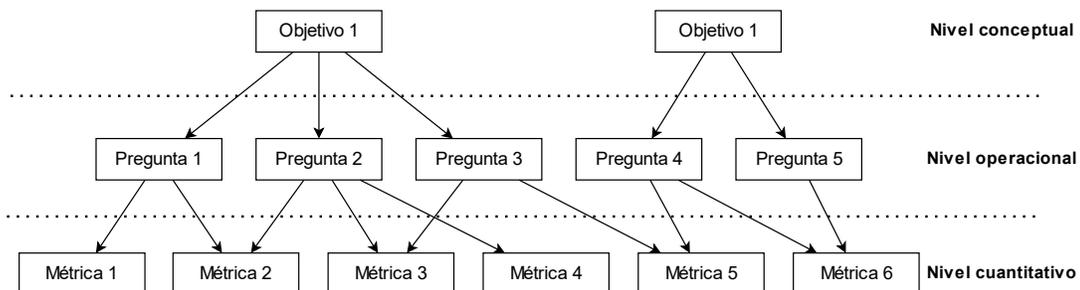


Figura 5.3. Estructura básica del modelo GQM. Tomado de [32].

Es importante mencionar que una organización puede definir un sistema de medición compuesto por varios modelos GQM que midan diferentes aspectos de la problemática que se desee abordar con preguntas y métricas en común que permitan emitir conclusiones más amplias y posiblemente más detalladas después de su ejecución.

5.1.4.2. Propósito y alcance del modelo de métricas

El modelo de métricas permite medir la deuda de la documentación en proyectos de software ágil, además, es una herramienta que establece un conjunto de indicadores capaces de valorizar los riesgos relacionados con los atributos más críticos e importantes de la documentación de software. Según algunos autores como [29], [55] y [76], los problemas más típicos de la documentación son: la incompletitud, ambigüedad, ejemplos inexplicados, obsolescencia, incoherencia, incorrección, hinchazón, fragmentación, exceso de información estructural e información enmarañada; estos problemas pueden llevar a que la documentación no sea confiable, sea inadecuada, incompleta, inexistente, carecer de trazabilidad, desactualizada y carente de integridad. En consecuencia, pueden verse comprometidos aspectos como: la eficiencia, la calidad y la sostenibilidad del software.

El modelo propuesto permitiría evaluar algunas características de la documentación de software para identificar sus falencias y diseñar estrategias para construir una documentación sólida que beneficie a todos los aspectos del ciclo de vida del software, desde el diseño y la implementación hasta el mantenimiento y la adopción por parte de los usuarios.

5.1.4.3. Limitaciones del modelo

El modelo de métricas propuesto no se encuentra exento de ciertas limitaciones que deben tenerse en cuenta en el momento de aplicarlo e interpretar los resultados que se obtengan a partir de él. Específicamente, el modelo propuesto podría no ser capaz de capturar o representar adecuadamente la subjetividad inherente en la evaluación de la deuda en la documentación, ya que diferentes individuos que lo utilicen pueden poseer percepciones distintas sobre lo que constituye una documentación de calidad o sobre el nivel de deuda que

podría ser tolerable. En ese sentido, otra limitación del modelo se relaciona con la dificultad de medir con precisión la capacidad de la documentación para ser comprendida por los usuarios, ya que es un proceso extremadamente subjetivo que depende en gran medida de la experiencia y habilidades del lector. Asimismo, es posible que para el modelo resulte difícil medir la precisión técnica de la documentación debido a que puede contener errores técnicos o inexactitudes. Finalmente, a pesar de que el modelo intenta trabajar con valores cuantitativos en cantidades exactas como, por ejemplo, la cantidad de documentos con marcas de confidencialidad, la cantidad de secciones de un documento que coincide con una plantilla, entre otros; no tiene en cuenta la medición específica de aspectos con una alta carga de subjetividad como: la calidad de la documentación o su utilidad real para la organización.

5.1.4.4. Descripción del modelo de métricas

El modelo propuesto consta de 39 métricas relacionadas con 17 características de la documentación, los cuales se clasifican en 5 dimensiones que permiten analizar el fenómeno de la deuda de la documentación desde diferentes perspectivas. A continuación, en las siguientes secciones, se presentan los detalles de las 39 métricas construidas para cada característica, cuya descripción utiliza la plantilla presentada en la Tabla 5.4.

Tabla 5.4. Plantilla para la descripción de métricas.

Nombre de la métrica	
Identificador	En este campo describe el identificador único de la métrica.
Propósito	En este campo describe el propósito de la métrica.
Unidad	En este campo expone la unidad que soporta la métrica.
Escala	En este campo expone la escala de valores que utiliza la métrica.
Ecuación	En este campo se expresa la expresión matemática de la métrica.
Variables	En este campo se listan las variables utilizadas en la ecuación.
Observaciones	En este campo se describen los aspectos que se deben tener en cuenta para la métrica.
Ejemplo de utilización	En este campo se describe un ejemplo de utilización de métrica.

5.1.4.4.1. Métricas para la característica de uniformidad

Para medir la uniformidad de la documentación de software se han definido dos métricas. Inicialmente, la métrica PUD (Percepción de Uniformidad de la Documentación) es descrita en la Tabla 5.5, y permite establecer el promedio de una encuesta compuesta por diez preguntas y cinco opciones de respuestas ponderadas de 1 a 5 con las cuales se evalúa la percepción que tienen los usuarios sobre la uniformidad de la documentación en relación con su formato, estructura, almacenamiento, accesibilidad y actualización.

Tabla 5.5. Métrica de percepción de uniformidad.

Indicador de percepción de uniformidad de la documentación				
Identificador	PUD			
Propósito	Determinar la percepción de uniformidad en la documentación por parte de los usuarios de dicha documentación con base en la siguiente encuesta: https://bit.ly/46nwOci .			
Unidad	Cantidad entera positiva			
Escala	[20, 100]			
Ecuación	$PUD = 2 * (P1 + P2 + P3 + P4 + P5 + P6 + P7 + P8 + P9 + P10)$			
Variables	<ul style="list-style-type: none"> • P1, P2, ..., P10: Hace referencia al valor entre 1 y 5 que se le puede asignar a las preguntas 1 a 10 según las posibles respuestas en la encuesta. • PUD: Indicador de la percepción general de uniformidad en la documentación. 			
Observaciones	Las opciones de respuesta de la encuesta tienen la siguiente ponderación:			
	Totalmente en desacuerdo	Nunca	Muy insatisfecho	1
	En desacuerdo	Casi nunca	Insatisfecho	2
	Neutral	Algunas veces	Neutral	3
	De acuerdo	Frecuentemente	Satisfecho	4
Totalmente de acuerdo	Siempre	Muy satisfecho	5	

Ejemplo de utilización	Para aplicar la métrica PUD sobre un artefacto de documentación, se debe responder la encuesta que tiene asociada seleccionando una de las opciones de respuesta disponibles en cada pregunta las cuales tienen un puntaje entre 1 y 5 como se describe en las <i>observaciones</i> . Suponiendo que las respuestas a las preguntas P1 a P10 de la encuesta hayan sido las siguientes:												
	<table style="width: 100%; border: none;"> <tr> <td style="width: 33%;">P1: Neutral</td> <td style="width: 33%;">P4: Nunca</td> <td style="width: 33%;">P7: Algunas veces</td> </tr> <tr> <td>P2: En desacuerdo</td> <td>P5: Frecuentemente</td> <td>P8: Muy satisfecho</td> </tr> <tr> <td>P3: Totalmente de acuerdo</td> <td>P6: Siempre</td> <td>P9: Satisfecho</td> </tr> <tr> <td></td> <td></td> <td>P10: Muy satisfecho</td> </tr> </table>	P1: Neutral	P4: Nunca	P7: Algunas veces	P2: En desacuerdo	P5: Frecuentemente	P8: Muy satisfecho	P3: Totalmente de acuerdo	P6: Siempre	P9: Satisfecho			P10: Muy satisfecho
	P1: Neutral	P4: Nunca	P7: Algunas veces										
P2: En desacuerdo	P5: Frecuentemente	P8: Muy satisfecho											
P3: Totalmente de acuerdo	P6: Siempre	P9: Satisfecho											
		P10: Muy satisfecho											
Entonces se reemplazarán los siguientes valores en la ecuación: $PUD = 2 * (3 + 1 + 5 + 1 + 4 + 5 + 3 + 5 + 4 + 5) = 2 * (36) = 72$ Por lo tanto, se tiene una percepción del 72% de uniformidad en el artefacto.													

Por otro lado, la métrica UPD (Utilización de las Plantillas de Documentación) es descrita en la Tabla 5.6, y permite determinar el nivel de correspondencia entre el artefacto de documentación que se está analizando y la plantilla que haya definido la organización para dicho artefacto. Esta correspondencia se determina a partir del número de secciones y los títulos de cada sección que coinciden entre el artefacto y su plantilla.

Tabla 5.6. Métrica para medir el nivel de fidelidad con la plantilla.

Indicador de utilización de las plantillas de documentación	
Identificador	UPD
Propósito	Determinar el nivel de fidelidad del artefacto con la plantilla que haya diseñado la organización comparando la cantidad de secciones que coinciden entre el documento y la plantilla definida para ese documento.
Unidad	Porcentaje (%)
Escala	[0, 100]
Ecuación	$UPD = \left(\frac{SC}{SP}\right) * 100$
Variables	<ul style="list-style-type: none"> • <i>SC</i>: Cantidad de secciones del documento que coinciden con la plantilla. • <i>SP</i>: Total de secciones de la plantilla.
Observaciones	Para esta métrica, si la organización no ha definido una plantilla para documentar el artefacto que se está analizando, entonces el valor de UPD será igual a cero.
Ejemplo de utilización	Para aplicar la métrica UPD se debe contar la cantidad de secciones que coinciden entre el artefacto de documentación y la plantilla que haya definido la organización para construirlo. Sin embargo, si la organización no ha definido una plantilla, entonces UPD = 0. De esta forma, si la plantilla consta de 6 secciones (SP) y artefacto coincide con 4 secciones (SC), entonces: $UPD = \left(\frac{4}{6}\right) * 100 = 0,667 * 100 = 66.7\%$ Por lo tanto, se tiene que el artefacto coincide en un 66.7% con respecto a su plantilla.

5.1.4.4.2. Métricas para la característica de adaptabilidad

Para medir la adaptabilidad de la documentación de software se ha definido la métrica PAD (Percepción de Adaptabilidad de la Documentación) descrita en la Tabla 5.7, la cual utiliza una encuesta de cinco preguntas para determinar cómo se percibe la capacidad que ofrece la documentación para modificar su contenido, estructura o formato de presentación sin muchas complicaciones, luego de ocurrir cambios en el sistema, preferencias de los usuarios, condiciones del entorno o disponibilidad de recursos.

Tabla 5.7. Métrica de percepción de adaptabilidad.

Indicador de percepción de adaptabilidad de la documentación	
Identificador	PAD
Propósito	Determinar la percepción de adaptabilidad en la documentación por parte de los usuarios de dicha documentación con base en la siguiente encuesta: https://bit.ly/45nUtYF
Unidad	Cantidad entera positiva
Escala	[20, 100]
Ecuación	$PAD = 4 * (P11 + P12 + P13 + P14 + P15)$
Variables	<ul style="list-style-type: none"> • <i>P11, P12, P13, P14, P15</i>: Hace referencia al valor entre 1 y 5 que se le puede asignar a las preguntas 11 a 15 según las posibles respuestas en la encuesta. • <i>PAD</i>: Indicador de la percepción general de adaptabilidad de la documentación.

Observaciones	Las opciones de respuesta de la encuesta tienen la siguiente ponderación:		
	Totalmente en desacuerdo	Nunca	1
	En desacuerdo	Casi nunca	2
	Neutral	Algunas veces	3
	De acuerdo	Frecuentemente	4
	Totalmente de acuerdo	Siempre	5
Ejemplo de utilización	<p>Para aplicar la métrica PAD sobre un artefacto de documentación, se debe responder la encuesta que tiene asociada seleccionando una de las opciones de respuesta disponibles en cada pregunta las cuales tienen un puntaje entre 1 y 5 como se describe en las <i>observaciones</i>. Suponiendo que las respuestas a las preguntas P11 a P15 de la encuesta hayan sido las siguientes:</p> <p>P11: De acuerdo P13: De acuerdo P15: Siempre P12: En desacuerdo P14: Frecuentemente</p> <p>Entonces se reemplazarán los siguientes valores en la ecuación:</p> $PAD = 4 * (4 + 2 + 4 + 4 + 5) = 4 * (19) = 76$ <p>Por lo tanto, se tiene una percepción del 76% de adaptabilidad en el artefacto.</p>		

5.1.4.4.3. Métricas para la característica de modularidad

Para medir la modularidad de la documentación de software, se han definido dos métricas. La primera, denominada PMD (Percepción de Modularidad de la Documentación), es descrita en la Tabla 5.8, y permite determinar la percepción de modularidad de la documentación por parte de los usuarios de los artefactos. Para hacerlo, la métrica promedia los resultados de una encuesta de cinco preguntas con cinco opciones de respuesta ponderadas de uno a cinco.

Tabla 5.8. Métrica de percepción de modularidad.

Indicador de percepción de modularidad de la documentación		
Identificador	PMD	
Propósito	Determinar la percepción de modularidad en la documentación por parte de los usuarios de dicha documentación con base en la siguiente encuesta: https://bit.ly/3RTS2dq .	
Unidad	Cantidad entera positiva	
Escala	[20, 100]	
Ecuación	$PMD = 4 * (P16 + P17 + P18 + P19 + P20)$	
Variables	<ul style="list-style-type: none"> • P16, P17, ..., P20: Hace referencia al valor entre 1 y 5 que se le puede asignar a las preguntas 16 a 20 según las posibles respuestas en la encuesta. • PMD: Indicador de la percepción general de modularidad en la documentación. 	
Observaciones	Las opciones de respuesta de la encuesta tienen la siguiente ponderación:	
	Totalmente en desacuerdo	Muy difícil 1
	En desacuerdo	Difícil 2
	Neutral	Algo difícil 3
	De acuerdo	Fácil 4
	Totalmente de acuerdo	Muy fácil 5
Ejemplo de utilización	<p>Para aplicar la métrica PMD sobre un artefacto de documentación, se debe responder la encuesta que tiene asociada seleccionando una de las opciones de respuesta disponibles en cada pregunta las cuales tienen un puntaje entre 1 y 5 como se describe en las <i>observaciones</i>. Suponiendo que las respuestas a las preguntas P16 a P20 de la encuesta hayan sido las siguientes:</p> <p>P16: Neutral P18: Muy difícil P20: Totalmente de acuerdo P17: En desacuerdo P19: Totalmente de acuerdo</p> <p>Entonces se reemplazarán los siguientes valores en la ecuación:</p> $PMD = 4 * (3 + 2 + 1 + 5 + 5) = 4 * (16) = 64$ <p>Por lo tanto, se tiene una percepción del 64% de modularidad en el artefacto.</p>	

Por otro lado, se definió la métrica IAD (índice de Acoplamiento de la Documentación), descrita en la Tabla 5.9, mide el nivel de acoplamiento o dependencia entre los artefactos de documentación, es decir; la cantidad de referencias hacia un documento. Para esto, la métrica se apoya en la construcción de un grafo dirigido utilizando las referencias que haya entre los documentos donde, gráficamente, cada nodo del grafo representa un artefacto y el peso de la arista entre dos nodos representa la cantidad de veces que un documento referencia al otro. Un ejemplo de cómo se vería el grafo es el presentado en la Figura 5.4:

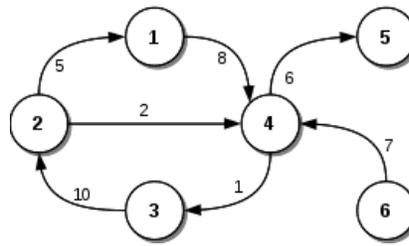


Figura 5.4. Ejemplo de grafo dirigido entre documentos.

Este grafo indica que hay cinco documentos que se relacionan entre sí, por ejemplo: el documento 2 referencia cinco veces al documento 1, mientras que documento 1 referencia ocho veces al documento 4, por lo tanto, el peso de cada arista representará la cantidad de veces que un documento referencia a otro, por eso la importancia de que el grafo sea dirigido ya que indicará claramente cuál es el documento referenciado y cuantas veces.

Tabla 5.9. Métrica para medir el índice de acoplamiento de la documentación.

Índice de acoplamiento de la documentación	
Identificador	IAD
Propósito	Determinar el nivel de acoplamiento o dependencia entre todos los artefactos que componen la documentación del sistema con respecto a la cantidad de enlaces o aristas en un grafo dirigido y ponderado, donde cada nodo del grafo representa un artefacto y el peso de la arista representa la cantidad de veces que se referencia un documento.
Unidad	Porcentaje (%)
Escala	[0, 100]
Ecuación	$IAD = \left(\frac{\sum_{i=1}^n PA_i}{TNG} \right) * 100$
Variables	<ul style="list-style-type: none"> • PA: Peso de la arista <i>i</i> entre dos nodos. • TNG: Cantidad de nodos del grafo. • n: Cantidad de aristas del grafo sin su peso.
Observaciones	Ninguna
Ejemplo de utilización	<p>Para aplicar la métrica IAD sobre un artefacto de documentación, se debe construir el grafo dirigido y ponderado a partir de las relaciones entre documentos donde la ponderación es la cantidad de veces que se relacionan dos documentos. Suponiendo que el grafo construido sea como el presentado en la Figura 5.4, entonces los valores a reemplazar en la ecuación serían:</p> <p>TNG=6 N=7 PA= depende del valor que tenga la arista <i>i</i></p> $IAD = \left(\frac{5 + 8 + 2 + 6 + 10 + 1 + 7}{6} \right) * 100 = \frac{39}{6} * 100 = 650\%$ <p>Por lo tanto, la documentación tiene un nivel de acoplamiento del 650%, extremadamente alto.</p>

5.1.4.4.4. Métricas para la característica de legibilidad

Para medir la legibilidad de la documentación se ha definido la métrica PLD (Percepción de Legibilidad de la Documentación) descrita en la Tabla 5.10, que utiliza una encuesta de cinco preguntas para determinar cómo se percibe la facilidad de lectura de la documentación, teniendo en cuenta que la información abstracta, demasiado técnica o sobre cargada, y errores tipográficos son aspectos en la documentación que puede afectar su legibilidad.

Tabla 5.10. Métrica de percepción de legibilidad.

Indicador de percepción de legibilidad de la documentación	
Identificador	PLD
Propósito	Determinar la percepción de legibilidad en la documentación por parte de los usuarios de dicha documentación con base en la siguiente encuesta: https://bit.ly/3ZZwvSz .
Unidad	Cantidad entera positiva
Escala	[20, 100]

	Es totalmente claro	Muy fácil	Totalmente de acuerdo	5
Ejemplo de utilización	Para aplicar la métrica PCLD sobre un artefacto de documentación, se debe responder la encuesta que tiene asociada seleccionando una de las opciones de respuesta disponibles en cada pregunta las cuales tienen un puntaje entre 1 y 5 como se describe en las <i>observaciones</i> . Suponiendo que las respuestas a las preguntas P31 a P34 de la encuesta hayan sido las siguientes:			
	P31: Es totalmente claro P32: De acuerdo		P33: Difícil P34: De acuerdo	
	Entonces se reemplazarán los siguientes valores en la ecuación:			
	$PCLD = 5 * (5 + 4 + 2 + 4) = 75$			
Por lo tanto, se tiene una percepción del 75% de claridad en el artefacto.				

Por otro lado, se definieron las métricas PPO (Promedio de Palabras por Oración) y PFO (Promedio de Frases por Oración), las cuales se describen en las tablas 5.14 y 5.15, teniendo en cuenta que se pueden dar de dos aspectos que afectan la legibilidad de un texto; estos son: entre más largas sean las oraciones, es decir, entre más palabras contengan, se reducirá su legibilidad; de manera similar, si las oraciones de un texto contienen muchas frases puede afectar la legibilidad.

Tabla 5.14. Promedio de palabras por oración.

Promedio de palabras por oración	
Identificador	PPO
Propósito	Medir la claridad de la documentación desde el punto de vista de la cantidad promedio de palabras que se utilizan en las oraciones en el contenido, ya que entre más palabras conformen una oración menor será su claridad.
Unidad	Valor entero positivo
Escala	[1, TPC]
Ecuación	$PPO = \left(\frac{TPC}{TOC} \right)$
Variables	<ul style="list-style-type: none"> • TPC: Hace referencia a la cantidad de palabras de un texto. • TOC: Hace referencia a la cantidad de oraciones de un texto.
Observaciones	<ul style="list-style-type: none"> • Esta métrica solo se podría aplicar a la documentación digital evaluando un texto muy corto. • Para el conteo de palabras y oraciones: https://www.separarensilabas.com/index.php.
Ejemplo de utilización	<p>Para aplicar la métrica PPO se usará la siguiente historia de usuario como ejemplo:</p> <p>Como cliente interesado en la lectura digital, María, una entusiasta lectora, desea realizar una compra en línea de un libro electrónico de su autor favorito para poder acceder a su última obra de manera rápida y conveniente. María es una profesional con un horario ocupado, y la compra en línea le brinda la flexibilidad de obtener el libro en cualquier momento que le sea conveniente, sin tener que visitar una tienda física.</p> <p>TPC = 72, TOC = 2</p> <p>Entonces se reemplazarán los siguientes valores en la ecuación:</p> $PPO = \left(\frac{72}{2} \right) = 36$ <p>Por lo tanto, en el texto de ejemplo se utiliza un promedio 36 palabras por oración.</p>

Tabla 5.15. Promedio de palabras por oración.

Promedio de frases por oración	
Identificador	PFO
Propósito	Medir la claridad de la documentación desde el punto de vista de la cantidad promedio de frases que se utilizan en las oraciones en el contenido, ya que entre más frases conformen una oración menor será su claridad.
Unidad	Porcentaje
Escala	[0, 100]
Ecuación	$PFO = \left(\frac{TFC}{TOC} \right) * 100$
Variables	<ul style="list-style-type: none"> • TFC: Hace referencia a la cantidad de frases de un texto. • TOC: Hace referencia a la cantidad de oraciones de un texto.
Observaciones	<ul style="list-style-type: none"> • Esta métrica solo se podría aplicar a la documentación digital evaluando un texto muy corto. • Para el conteo de frases y oraciones: https://www.separarensilabas.com/index.php.
	Para aplicar la métrica PFO se usará la siguiente historia de usuario como ejemplo:

Unidad	Porcentaje
Escala	[0, 100]
Ecuación	$AMC = \left(\frac{TAM}{TAE}\right) * 100$
Variables	<ul style="list-style-type: none"> • TAM: Hace referencia al total de artefactos que han sido marcados. • TAE: Hace referencia al total de documentos del sistema.
Observaciones	Ninguna
Ejemplo de utilización	<p>Para aplicar la métrica AMC se debe contar la cantidad de documentos del sistema que tienen indicaciones que expliquen claramente su nivel de confidencialidad. De esta forma, si la documentación del sistema se compone de 50 documentos, de los cuales 13 tienen marcas de confidencialidad entonces los valores de las variables serían los siguientes: TAM = 13, TAE = 50</p> <p>Entonces se reemplazarán los siguientes valores en la ecuación:</p> $AMC = \left(\frac{13}{50}\right) * 100 = 26$ <p>Por lo tanto, el nivel de confidencialidad se especifica en el 26% de la documentación.</p>

La tercera métrica, denominada DNA (Definición de Niveles de Acceso), es descrita en la Tabla 5.20, y permite conocer la proporción de miembros del equipo a los cuales se les han definido exactamente las restricciones que pueden tener para acceder a la documentación con respecto a las políticas de la empresa.

Tabla 5.20. Definición de niveles de acceso a la documentación.

Índice de definición de niveles de acceso a la documentación	
Identificador	DNA
Propósito	Determinar el porcentaje de miembros del equipo sobre los cuales se han definido responsabilidades claras de acceso a la documentación.
Unidad	Porcentaje
Escala	[0, 100]
Ecuación	$DNA = \left(\frac{MCNA}{ME}\right) * 100$
Variables	<ul style="list-style-type: none"> • MCNA: Hace referencia a la cantidad de miembros del equipo que tienen claro su nivel de acceso. • ME: Hace referencia al total de miembros del equipo de desarrollo.
Observaciones	Ninguna
Ejemplo de utilización	<p>Para aplicar la métrica DNA se debe contar la cantidad de miembros del equipo que conocen claramente su nivel acceso a la documentación y la cantidad de miembros del equipo que participa en el proyecto. De esta forma, si el equipo está conformado por 20 personas y a 7 de ellos se ha definido claramente su nivel de acceso a la documentación entonces los valores que se utilizarán para la ecuación serían. ME = 20, MCNA = 7</p> <p>Entonces se reemplazarán los siguientes valores en la ecuación:</p> $DNA = \left(\frac{7}{20}\right) * 100 = 35$ <p>Por lo tanto, solo el 35% del equipo tiene claramente definido su nivel de acceso a la documentación.</p>

La cuarta métrica, denominada DLA (Documentación de Libre Acceso), es descrita en la Tabla 5.21, y permite conocer la proporción de documentación que no tiene restricciones de confidencialidad ni requieren ningún tipo de autorización. Este resultado puede visibilizar un problema de vulnerabilidad en la confidencialidad de la información del sistema.

Tabla 5.21. Índice de documentación de libre acceso.

Índice de documentación de libre acceso	
Identificador	DLA
Propósito	Determinar el porcentaje de documentos que son de libre acceso.
Unidad	Porcentaje
Escala	[0, 100]
Ecuación	$DLA = \left(\frac{TDP + TDSN}{TDS}\right) * 100$
Variables	<ul style="list-style-type: none"> • TDP: Hace referencia a la cantidad de documentos que tienen un nivel de confidencialidad público. • TDSN: Hace referencia a la cantidad de documentos que NO tienen un nivel de confidencialidad definido.

	<ul style="list-style-type: none"> • TDS: Hace referencia a la cantidad de documentos que componen la documentación del sistema.
Observaciones	<ul style="list-style-type: none"> • Para esta métrica, cualquier documento que no tenga una indicación clara de su nivel de confidencialidad se considera público, es decir, que cualquier usuario puede acceder a su contenido. • $(TDP + TDSN) \leq TDS$
Ejemplo de utilización	<p>Para aplicar la métrica DLA se debe contar la cantidad de documentos que tienen un nivel de confidencialidad público y la cantidad de documentos que no tienen un nivel de confidencialidad definido. De esta forma, si la documentación del sistema está compuesta por 25 documentos, de los cuales 5 son acceso público y 7 no tienen un nivel de confidencialidad definido, entonces los valores que se utilizarán para la ecuación serían.</p> <p>TDP = 5, TDSN = 7, TDS = 25</p> <p>Entonces se reemplazarán los siguientes valores en la ecuación:</p> $DLA = \left(\frac{5 + 7}{25}\right) * 100 = 48$ <p>Por lo tanto, el 48% de la documentación es de carácter público.</p>

La quinta métrica, denominada DEC (Documentación Extremadamente Confidencial), es descrita en la Tabla 5.22, y permite determinar la proporción de la documentación del sistema que tiene el máximo nivel de confidencialidad y, por ende, exige mayores controles de acceso.

Tabla 5.22. Documentación extremadamente confidencial.

Índice de documentación extremadamente confidencial	
Identificador	DEC
Propósito	Determinar el porcentaje de documentos que cuentan con el máximo nivel de confidencialidad.
Unidad	Porcentaje
Escala	[0, 100]
Ecuación	$DEC = \left(\frac{TDC}{TDS}\right) * 100$
Variables	<ul style="list-style-type: none"> • TDC: Hace referencia a la cantidad de documentos que tienen el máximo nivel de confidencialidad. • TDS: Hace referencia a la cantidad de documentos que componen la documentación del sistema.
Observaciones	Ninguna
Ejemplo de utilización	<p>Para aplicar la métrica DEC se debe contar la cantidad de documentos que tienen el nivel máximo de confidencialidad. De esta forma, si la documentación del sistema está compuesta por 25 documentos, de los cuales 5 son extremadamente confidenciales, entonces los valores que se utilizarán para la ecuación serían.</p> <p>TDC = 5, TDS = 25</p> <p>Entonces se reemplazarán los siguientes valores en la ecuación:</p> $DLA = \left(\frac{5}{25}\right) * 100 = 20$ <p>Por lo tanto, el 20% de la documentación es extremadamente confidencial.</p>

La sexta y última métrica se denomina CIL (Conocimiento de Implicaciones Legales) y es descrita en la Tabla 5.23, esta métrica permite conocer la proporción de miembros del equipo que han sido informados sobre las implicaciones legales de divulgar información confidencial, ya sea de la empresa, del sistema o de los procesos internos.

Tabla 5.23. Conocimiento de implicaciones legales.

Índice de conocimiento de implicaciones legales	
Identificador	CIL
Propósito	Determinar el porcentaje de miembros del equipo que han sido informados o tienen conocimiento de las implicaciones legales de divulgar información confidencial del sistema consignada en la documentación.
Unidad	Porcentaje
Escala	[0, 100]
Ecuación	$CIL = \left(\frac{MCI}{ME}\right) * 100$
Variables	<ul style="list-style-type: none"> • MCI: Hace referencia a la cantidad de miembros que conocen las implicaciones legales. • ME: Hace referencia al total de miembros del equipo de desarrollo.
Observaciones	Ninguna
Ejemplo de utilización	Para aplicar la métrica CIL se debe contar la cantidad de miembros del equipo que conocen las implicaciones legales de divulgar información confidencial. De esta forma, si el equipo se compone de 15

<p>personas y 6 han sido informadas de las implicaciones legales, entonces los valores que se utilizarán para la ecuación serían.</p> <p>MCI = 6, ME = 15</p> <p>Entonces se reemplazarán los siguientes valores en la ecuación:</p> $CIL = \left(\frac{6}{15}\right) * 100 = 40$ <p>Por lo tanto, el 40% de los miembros del equipo han sido informados de las implicaciones legales de divulgar información confidencial sin autorización.</p>
--

5.1.4.4.9. Métricas para la característica de consistencia

Para medir la consistencia de la documentación es necesario conocer cómo se relacionan los diferentes artefactos de documentación del sistema. Para esto, de forma similar a la métrica IAD descrita en la Tabla 5.9, se ha tomado el concepto de grafo dirigido y ponderado para identificar dichas relaciones y verificar la duplicidad de información o su posible excesiva fragmentación. Esto permite determinar qué tan complejo sería modificar un artefacto debido a la cantidad de relaciones que tiene con otros artefactos e indicaría que una funcionalidad puede estar demasiado fragmentada y que, por ende, se deban actualizar todos los documentos relacionados. Con base en esta información, se proponen dos métricas.

La primera de ellas, denominada PRI (Porcentaje de Relaciones Incorrectas), es descrita en la Tabla 5.24, y permite conocer la proporción de relaciones incorrectas entre documentos con respecto a todas las relaciones que tiene la documentación del sistema.

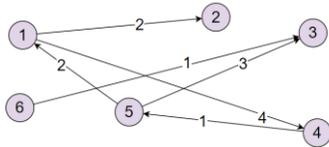
Tabla 5.24. Porcentaje de relaciones incorrectas.

Porcentaje de relaciones incorrectas	
Identificador	PRI
Propósito	Determinar la proporción de relaciones entre documentos que no debería relacionarse entre sí o que son incorrectas con respecto a todos los documentos relacionados.
Unidad	Porcentaje (%)
Escala	[0, 100]
Ecuación	$PRI = \left(\frac{TRI}{TRG}\right) * 100$
Variables	<ul style="list-style-type: none"> • TRI: Hace referencia a la cantidad de relaciones en el grafo que son incorrectas. • TRG: Hace referencia a la cantidad de relaciones del grafo.
Observaciones	Si el grafo no tiene ninguna relación entre sus nodos, entonces la métrica PRI será igual a cero.
Ejemplo de utilización	<p>Para aplicar la métrica PRI, se debe construir el grafo dirigido y verificar si todas las relaciones entre los nodos son correctas o no. De esta forma, si el grafo tiene 20 relaciones, pero 4 son incorrectas entonces los valores que se utilizarán para la ecuación serían.</p> <p>TRI = 4 y TRG = 20</p> <p>Entonces se reemplazarán los siguientes valores en la ecuación:</p> $PRI = \left(\frac{4}{20}\right) * 100 = 20$ <p>Por lo tanto, el 20% de las relaciones entre los artefactos son incorrectas.</p>

La segunda métrica, denominada DRD (Densidad de Relaciones de la Documentación), es descrita en la Tabla 5.25, y permite determinar la proporción de relaciones entre los documentos del sistema, lo cual ayuda a conocer el nivel de fragmentación de la información y como esto puede afectar la consistencia de la documentación debido a la cantidad de partes que se deben interconectar para explicar una característica del sistema.

Tabla 5.25. Densidad de relaciones entre la documentación.

Densidad de relaciones de la documentación	
Identificador	DRD
Propósito	Determinar el nivel de densidad de las relaciones entre los artefactos de documentación con base en la cantidad de aristas del grafo.
Unidad	Porcentaje (%)

Escala	[0, 100]
Ecuación	$DRD = \left(\frac{2 * TAG}{TNG * (TNG - 1)} \right) * 100$
Variables	<ul style="list-style-type: none"> • TAG: Hace referencia a la cantidad de aristas que tiene el grafo sin su ponderación. • TNG: Hace referencia a la cantidad de nodos del grafo.
Observaciones	<ul style="list-style-type: none"> • La documentación del sistema debe tener al menos dos documentos. • Entre mayor sea el resultado significa que la información estará más fragmentada y por ende, tendrá menos consistencia.
Ejemplo de utilización	<p>Para aplicar la métrica DRD, se debe construir el grafo dirigido y contar la cantidad de aristas del grafo, así como los nodos que lo componen. De esta forma, si la documentación del sistema se compone de 6 documentos con 6 relaciones entre ellos como se muestra en la imagen entonces los valores que se utilizarán para la ecuación serían.</p>  <p>TAG = 6 y TNG = 6</p> <p>Entonces se reemplazarán los siguientes valores en la ecuación:</p> $DRD = \left(\frac{2 * 6}{6 * (6 - 1)} \right) * 100 = \left(\frac{12}{6 * 5} \right) * 100 = 40$ <p>Por lo tanto, el grafo tiene una densidad del 40%.</p>

5.1.4.4.10. Métricas para la característica de precisión

Para medir la precisión en la documentación de software, se han definido dos métricas. La primera de ellas, denominada PPD (Percepción de la Precisión de la Documentación), es descrita en la Tabla 5.26, y permite medir la percepción que tienen los usuarios sobre la precisión de la documentación en relación con la descripción real del sistema.

Tabla 5.26. Indicador de percepción de precisión de la documentación.

Indicador de percepción de precisión de la documentación				
Identificador	PPD			
Propósito	Determinar la percepción de los usuarios de la documentación con respecto a la precisión de la documentación con base en la siguiente encuesta: https://bit.ly/46vQHOe .			
Unidad	Cantidad entera positiva			
Escala	[20, 100]			
Ecuación	$PPD = 4 * (P45 + P46 + P47 + P48 + P49)$			
Variables	<ul style="list-style-type: none"> • P45, P46, P47, P48, P49: Hace referencia al valor entre 1 y 5 que se le puede asignar a las preguntas 45 a 49 según las posibles respuestas en la encuesta. • PPD: Indicador de la percepción general de la precisión de la documentación. 			
Observaciones	Las opciones de respuesta de la encuesta tienen la siguiente ponderación:			
	Totalmente en desacuerdo	He encontrado muchas discrepancias extremadamente críticas	Totalmente insatisfecho	1
	En desacuerdo	He encontrado varias discrepancias importantes	Insatisfecho	2
	Neutral	He encontrado algunas discrepancias que se deben tener en cuenta	Algo satisfecho	3
	De acuerdo	He encontrado discrepancias menores que podrían solucionarse	Satisfecho	4
	Totalmente de acuerdo	No he encontrado ninguna discrepancia	Totalmente satisfecho	5
Ejemplo de utilización	<p>Para aplicar la métrica PPD sobre un artefacto de documentación, se debe responder la encuesta que tiene asociada seleccionando una de las opciones de respuesta disponibles en cada pregunta las cuales tienen un puntaje entre 1 y 5 como se describe en las <i>observaciones</i>. Suponiendo que las respuestas a las preguntas P45 a P49 de la encuesta hayan sido las siguientes:</p> <p>P45: En desacuerdo P47: No he encontrado ninguna discrepancia P49: Satisfecho P46: En desacuerdo P48: Totalmente de acuerdo</p> <p>Entonces se reemplazarán los siguientes valores en la ecuación:</p> $PPD = 4 * (2 + 2 + 5 + 5 + 4) = 5 * (18) = 90$ <p>Por lo tanto, se tiene una percepción de precisión del 90% en la documentación.</p>			

La segunda métrica, denominada RMT (Requisitos en la Matriz de Trazabilidad), es descrita en la Tabla 5.28, y permite medir la cantidad de requisitos que han sido cubiertos en una estructura igual o similar a una matriz de trazabilidad de requisitos, y que permita el rastreo de cada funcionalidad y como se relaciona con las demás funcionalidades del sistema. Específicamente, una matriz de trazabilidad es una herramienta que se utiliza para registrar, relacionar y rastrear requisitos de un proyecto software, la cual puede ser diseñada de diferentes maneras [78]. En Tabla 5.27, se presenta un ejemplo de cómo podría ser diseñada una matriz de trazabilidad donde se relacionan los requisitos del sistema con otros requisitos, casos de prueba y componentes.

Tabla 5.27. Ejemplo del diseño de una matriz de trazabilidad.

Id	Requisitos / Casos de Uso / Casos de Prueba / Componentes	Descripción	CU 1 / HU 1	CU 2 / HU 2	CU 3 / HU 3	Caso de prueba 1	Caso de prueba 2	Caso de prueba 3	Componente 1	Componente 2
REQ-FUNC-001	Registro de usuarios		X		X	X			X	
REQ-FUNC-002	Inicio de sesión		X					X	X	
REQ-FUNC-003	Mostrar catálogo de productos			X	X		X			X
REQ-SEG-001	Datos encriptados		X			X			X	
CP1	Probar el registro de usuarios		X					X	X	
CP2	Probar el inicio de sesión			X		X	X		X	
CP3	Probar el catálogo de productos			X			X			X
Componente 1	Módulo de usuarios		X		X	X			X	
Componente 2	Módulo de catálogo			X	X		X			X

Tabla 5.28. Cobertura de requisitos en la matriz de trazabilidad.

Cobertura de requisitos en la matriz de trazabilidad	
Identificador	RMT
Propósito	Determinar el porcentaje de requisitos que han sido cubiertos en la matriz de trazabilidad y que pueden ser rastreados.
Unidad	Porcentaje
Escala	[0, 100]
Ecuación	$RMT = \left(\frac{NRC}{NTR} \right) * 100$
Variables	<ul style="list-style-type: none"> • NRC: Hace referencia al total de requisitos que se han cubierto en la matriz de trazabilidad. • NTR: Hace referencia al total de requisitos del sistema.
Observaciones	Ninguna
Ejemplo de utilización	<p>Para aplicar la métrica RMT se cuenta la cantidad de requisitos que han sido registrados en la matriz de trazabilidad y la cantidad de requisitos del sistema. De esta forma, si se han definido 50 requisitos para el sistema y solo se han registrado 27 requisitos entonces los valores que se utilizarán para la ecuación serían.</p> <p>NTR = 50, NRC = 27</p> <p>Entonces se reemplazarán los siguientes valores en la ecuación:</p> $RMT = \left(\frac{27}{50} \right) * 100 = 54$ <p>Por lo tanto, en la matriz de trazabilidad solo se han cubierto el 54% de los requisitos.</p>

5.1.4.4.11. Métricas para la característica de portabilidad

Para medir la portabilidad de la documentación de software, se han definido tres métricas. La primera de ellas, denominada PPRD (Percepción de la Portabilidad de la Documentación), es

descrita en la Tabla 5.29, y permite medir la percepción que tienen los usuarios de la documentación con respecto al formato en que se presenta, la organización del contenido, su capacidad de despliegue en diferentes plataformas y la satisfacción de las herramientas para su creación, almacenamiento y accesibilidad.

Tabla 5.29. Indicador de percepción de la portabilidad de la documentación.

Indicador de percepción de la portabilidad de la documentación			
Identificador	PPRD		
Propósito	Determinar la percepción de los usuarios de la documentación con respecto a su portabilidad con base en la siguiente encuesta: https://bit.ly/3Q0ySA9 .		
Unidad	Cantidad entera positiva		
Escala	[20, 100]		
Ecuación	$PPRD = 5 * (P50 + P51 + P52 + P53)$		
Variables	<ul style="list-style-type: none"> • P50, P51, P52, P53: Hace referencia al valor entre 1 y 5 que se le puede asignar a las preguntas 50 a 53 según las posibles respuestas en la encuesta. • PPRD: Indicador de la percepción general de la portabilidad de la documentación. 		
Observaciones	Las opciones de respuesta de la encuesta tienen la siguiente ponderación:		
	Totalmente en desacuerdo	Totalmente insatisfecho	1
	En desacuerdo	Insatisfecho	2
	Neutral	Algo satisfecho	3
	De acuerdo	Satisfecho	4
	Totalmente de acuerdo	Totalmente satisfecho	5
Ejemplo de utilización	<p>Para aplicar la métrica PPRD sobre un artefacto de documentación, se debe responder la encuesta que tiene asociada seleccionando una de las opciones de respuesta disponibles en cada pregunta las cuales tienen un puntaje entre 1 y 5 como se describe en las <i>observaciones</i>. Suponiendo que las respuestas a las preguntas P50 a P53 de la encuesta hayan sido las siguientes:</p> <p style="text-align: center;">P50: En desacuerdo P52: Neutral P51: En desacuerdo P53: Algo satisfecho</p> <p>Entonces se reemplazarán los siguientes valores en la ecuación:</p> $PPD = 5 * (2 + 2 + 3 + 3) = 5 * (10) = 50$ <p>Por lo tanto, se tiene una percepción de portabilidad del 50% en la documentación.</p>		

La segunda métrica, denominada DIP (Documentación Independiente de las Plataformas), es descrita en la Tabla 5.30, y permite conocer la proporción de la documentación que no depende de un Sistema Operativo o plataforma específica para ser visualizada o editada. En particular, para esta métrica se han considerado los siguientes tipos de documentos como independientes:

- **Formato de documento portátil PDF:** Archivos que pueden ser visualizados y editados en diferentes plataformas y Sistemas Operativos.
- **Formato de texto plano TXT:** Documentos sin formato que pueden ser visualizados y editados en cualquier plataforma y Sistema Operativo con un editor de texto básico.
- **Formato de documento enriquecido RTF:** Documentos con formato básico que pueden ser abiertos y editados en múltiples programas de procesamiento de texto.
- **Formato de hipertexto HTML:** Utilizados para crear páginas web y pueden ser visualizados a través de un navegador web.
- **Formato de hoja de cálculo CSV:** Son datos tabulares separados por comas que pueden ser abiertos y editados aplicaciones de hojas de cálculo.
- **Formato de imagen JPEG, PNG, GIF:** Archivos de imagen compatibles con múltiples Sistemas Operativos.

- **Formato de audio MP3, WAV:** Archivos en formato de audio con capacidad de ser reproducidos en múltiples plataformas y Sistemas Operativos.
- **Formato de video MP4, AVI, MKV:** Archivos de video con capacidad de ser reproducidos en múltiples plataformas y Sistemas Operativos.

Tabla 5.30. Documentación independiente de plataformas específicas.

Documentación independiente de plataformas específicas	
Identificador	DIP
Propósito	Determinar la proporción de documentos que están hechos en un formato independiente del Sistema Operativo o de las plataformas tecnológicas que maneja la empresa.
Unidad	Porcentaje (%)
Escala	[0, 100]
Ecuación	$DIP = \left(\frac{TAFI}{TAS}\right) * 100$
Variables	<ul style="list-style-type: none"> • TAFI: Hace referencia a la cantidad de artefactos en un formato independientes. • TAS: Hace referencia a la cantidad artefactos de documentación del sistema.
Observaciones	Ninguna
Ejemplo de utilización	<p>Para aplicar la métrica DIP, se cuenta los artefactos que son de alguno de los tipos de documentación independiente listados anteriormente y la cantidad total de artefactos de documentación del sistema. De esta forma, si el sistema tiene 20 artefactos y 9 son independientes, entonces la ecuación sería. TAFI = 9 y TAS = 20</p> <p>Entonces se reemplazarán los siguientes valores en la ecuación:</p> $DIP = \left(\frac{9}{20}\right) * 100 = 45$ <p>Por lo tanto, el 45% de la documentación es independiente de un Sistema Operativo o de plataformas específicas para su visualización y edición.</p>

La tercera métrica, denominada DCE (Documentación que requiere configuraciones especiales), es descrita en la Tabla 5.31, y complementa la métrica anterior debido a que mide la proporción de documentación que depende de un Sistema Operativo o software específicos para ser visualizada, editada o gestionada.

Tabla 5.31. Documentación que requiere configuraciones especiales.

Documentación que requiere configuraciones especiales	
Identificador	DCE
Propósito	Determinar la proporción de documentos que requieren de alguna configuración especial o dependen de un Sistema Operativo o de una plataforma específicos para ser visualizados, editados o gestionados.
Unidad	Porcentaje (%)
Escala	[0, 100]
Ecuación	$DCE = \left(\frac{TAD}{TAS}\right) * 100$
Variables	<ul style="list-style-type: none"> • TAD: Hace referencia a la cantidad de artefactos dependientes de una configuración especial o de un Sistema Operativo o plataforma específica. • TAS: Hace referencia a la cantidad artefactos de documentación del sistema.
Observaciones	Ninguna
Ejemplo de utilización	<p>Para aplicar la métrica DCE, se cuenta los artefactos que requieren una configuración especial y la cantidad total de artefactos de documentación del sistema. Así, si el sistema tiene un total de 20 artefactos, de los cuales 11 son dependientes, entonces los valores que se utilizarán para la ecuación serían: TAD = 11 y TAS = 20. Entonces se reemplazarán los siguientes valores en la ecuación:</p> $DIP = \left(\frac{11}{20}\right) * 100 = 55$ <p>Por lo tanto, el 55% de la documentación es dependiente de un Sistema Operativo o de plataformas específicas para su visualización y edición.</p>

5.1.4.4.12. Métricas para la característica de integridad

Para medir la integridad de la documentación de software se han definido tres métricas. La primera de ellas, denominada PID (Percepción de Integridad de la Documentación), es descrita

	$PAR_{ev} = 100 - \left(\left(\frac{11}{20} \right) * 100 \right) = 45$
<p>Por lo tanto, el 45% de la documentación del sistema se encuentra desfasada con respecto a su última versión.</p>	

La tercera métrica, denominada PIRU (Problemas de Inconsistencia Reportados por los Usuarios) es descrita en la Tabla 5.36, y permite determinar cuál es la proporción de los problemas de inconsistencia entre el contenido de la documentación y las funcionalidades del sistema reportados que hayan sido reportados por los usuarios.

Tabla 5.36. Porcentaje de problemas de inconsistencia.

Problemas de inconsistencia reportados por los usuarios	
Identificador	PIRU
Propósito	Determinar la proporción de problemas de inconsistencia reportados por los usuarios.
Unidad	Porcentaje (%)
Escala	[0, 100]
Ecuación	$PIRU = \left(\frac{PIR}{TPR} \right) * 100$
Variables	<ul style="list-style-type: none"> • PIR: Hace referencia a la cantidad de problemas que han reportado los usuarios de la documentación. • TPR: Hace referencia a la cantidad de problemas
Observaciones	Si no es posible determinar la cantidad de reportes por inconsistencia el valor de esta métrica será cero.
Ejemplo de utilización	<p>Para esta métrica, es necesario conocer la cantidad de problemas que han reportado sobre una inconsistencia en la documentación y la cantidad total de problemas reportados. De esta forma, si el sistema tiene un total de 50 problemas reportados y 17 están relacionados con inconsistencia en la documentación, entonces los valores que se utilizarán para la ecuación serían.</p> <p>PIR = 17 y TPR = 50</p> <p>Entonces se reemplazarán los siguientes valores en la ecuación:</p> $PIRU = \left(\frac{17}{50} \right) * 100 = 34$ <p>Por lo tanto, el 34% de los problemas reportados corresponden a problemas de inconsistencia entre el contenido de la documentación y las funcionalidades del sistema.</p>

5.1.4.4.14. Métricas para la característica de trazabilidad

Para medir la característica de trazabilidad de la documentación de software se han definido tres métricas a partir de una herramienta denominada Matriz de Trazabilidad que vincula los requisitos y otros elementos de un sistema software para rastrear su origen y seguimiento a lo largo del tiempo. Para entender mejor esta herramienta, en la Tabla 5.27 se presenta un ejemplo de cómo se puede diseñar y los diferentes elementos que podría relacionar.

Con base en la Matriz de Trazabilidad, la primera métrica que se ha diseñado ha sido denominada CDT (Cobertura de Trazabilidad) la cual se describe en la Tabla 5.37, y mide la capacidad de la documentación para permitir la trazabilidad de los requisitos del sistema.

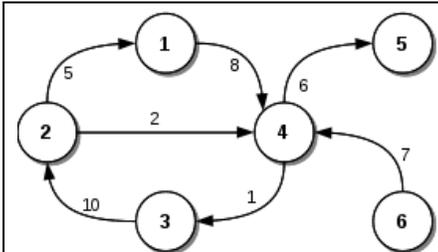
Tabla 5.37. Cobertura de trazabilidad.

Cobertura de trazabilidad	
Identificador	CDT
Propósito	Determinar la proporción de requisitos han sido listados en la matriz de trazabilidad pero que no tienen relación con otros requisitos en la matriz.
Unidad	Porcentaje (%)
Escala	[0, 100]
Ecuación	$CDT = \left(\frac{RSR}{NTR} \right) * 100$
Variables	<ul style="list-style-type: none"> • RSR: Hace referencia a la cantidad de requisitos que no se relacionan con ningún elemento registrado en la matriz. • NTR: Hace referencia al total de requisitos del sistema.
Observaciones	Ninguna

Ejemplo de utilización	<p>Para esta métrica, se deben contar los requisitos que han sido registrados en la matriz de trazabilidad pero que no se han relacionado con ningún otro requisito o componente del sistema. De esta forma, si la matriz tiene un total de 110 registros entre requisitos y componentes de los cuales 11 no se relacionan con otro elemento de la matriz, entonces los valores que se utilizarán para la ecuación serían.</p> <p>RSR = 11 y NTR = 110</p> <p>Entonces se reemplazarán los siguientes valores en la ecuación:</p> $CDT = \left(\frac{11}{110}\right) * 100 = 10$ <p>Por lo tanto, el 10% de los registros de la matriz no pueden ser rastreados.</p>
-------------------------------	--

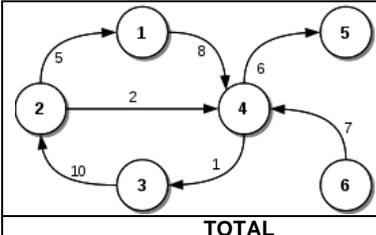
La segunda métrica, denominada NDD (Nivel de Dependencia entre los Documentos) es descrita en la Tabla 5.38, y utiliza un grafo dirigido y ponderado como el presentado en la Figura 5.4 para establecer el nivel de dependencia entre los diferentes artefactos de documentación representados en el grafo con base en el peso de la arista, la cual representa la cantidad de veces que un artefacto es citado o referenciado en otro. De esta manera, esta métrica permite determinar la dificultad de realizar un ajuste en la documentación debido al nivel de dependencia entre los artefactos.

Tabla 5.38. Nivel de dependencia entre los documentos.

Nivel de dependencia entre los documentos																																									
Identificador	NDD																																								
Propósito	Determinar el nivel de dependencia entre los artefactos de documentación.																																								
Unidad	Cantidad																																								
Escala	Números enteros positivos, negativos y cero																																								
Ecuación	$NDD = \left(\sum_1^n GS_i * \left(\sum_1^n PAS_i\right)\right) - \left(\sum_1^n GE_i * \left(\sum_1^n PAE_i\right)\right)$																																								
Variables	<ul style="list-style-type: none"> • GS: Grado de salida de un nodo. • PAS: Peso de las aristas que salen del nodo. • GE: Grado de entrada de un nodo. • PAE: Peso de las aristas que entran al nodo. • n: Hace referencia al total de nodos del grafo. 																																								
Observaciones	<ul style="list-style-type: none"> • Si el resultado es negativo, indica que los artefactos tienen una baja dependencia entre sí. • Si el resultado es positivo, indica que los artefactos tienen una alta dependencia entre sí. • Si el resultado es cero, todos los artefactos manejan el mismo nivel de dependencia. 																																								
Ejemplo de utilización	<p>Para aplicar esta métrica, se debe conocer el grado de salida y de entrada de cada nodo, así como el peso de las aristas en cada caso. De esta forma, si al construir el grafo resulta en una estructura como la presentada en la imagen, el resultado de aplicar la métrica sería el siguiente:</p> <div style="display: flex; align-items: center;">  <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Nodo</th> <th>GS</th> <th>PAS</th> <th>GE</th> <th>PAE</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1</td> <td>8</td> <td>1</td> <td>5</td> </tr> <tr> <td>2</td> <td>2</td> <td>7</td> <td>1</td> <td>10</td> </tr> <tr> <td>3</td> <td>1</td> <td>10</td> <td>1</td> <td>1</td> </tr> <tr> <td>4</td> <td>2</td> <td>7</td> <td>3</td> <td>17</td> </tr> <tr> <td>5</td> <td>0</td> <td>0</td> <td>1</td> <td>6</td> </tr> <tr> <td>6</td> <td>1</td> <td>7</td> <td>0</td> <td>0</td> </tr> <tr> <td>TOTAL</td> <td>7</td> <td>39</td> <td>7</td> <td>39</td> </tr> </tbody> </table> </div> <p>Entonces se reemplazarán los valores de la tabla en la ecuación:</p> $NDD = (7 * 39) - (7 * 39) = 0$ <p>Por lo tanto, si NDD = 0, indica que todos los documentos manejan el mismo nivel de dependencia.</p>	Nodo	GS	PAS	GE	PAE	1	1	8	1	5	2	2	7	1	10	3	1	10	1	1	4	2	7	3	17	5	0	0	1	6	6	1	7	0	0	TOTAL	7	39	7	39
Nodo	GS	PAS	GE	PAE																																					
1	1	8	1	5																																					
2	2	7	1	10																																					
3	1	10	1	1																																					
4	2	7	3	17																																					
5	0	0	1	6																																					
6	1	7	0	0																																					
TOTAL	7	39	7	39																																					

La tercera métrica, denominada NAA (Nivel de Asociación entre artefactos) es descrita en la Tabla 5.39, y permite determinar el nivel de asociación entre los artefactos que componen la documentación del sistema con respecto a la cantidad de relaciones que se pueden establecer entre todos los artefactos representados en el grafo. De esta forma, entre más cercano se encuentre el resultado a 100, mayor será la dificultad para actualizar la documentación debido a la cantidad de relaciones que existen.

Tabla 5.39. Nivel de asociación entre los artefactos.

Nivel de asociación entre los artefactos																									
Identificador	NAA																								
Propósito	Determinar el nivel de asociación entre los artefactos de documentación.																								
Unidad	Porcentaje																								
Escala	[0, 100]																								
Ecuación	$NAA = \left(\frac{\sum_1^n GS_i + \sum_1^n GE_i}{2(n)(n-1)} \right) * 100$																								
Variables	<ul style="list-style-type: none"> • GS: Grado de salida de un nodo. • GE: Grado de entrada de un nodo. • n: Hace referencia al total de nodos del grafo. 																								
Observaciones	Ninguna																								
Ejemplo de utilización	<p>Para aplicar esta métrica, se debe conocer el grado de salida y de entrada de cada nodo. Así, si al construir el grafo resulta en una estructura como la presentada en la imagen, la aplicación de la métrica:</p>  <table border="1" data-bbox="963 642 1198 884"> <thead> <tr> <th>Nodo</th> <th>GS</th> <th>GE</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>2</td> <td>2</td> <td>1</td> </tr> <tr> <td>3</td> <td>1</td> <td>1</td> </tr> <tr> <td>4</td> <td>2</td> <td>3</td> </tr> <tr> <td>5</td> <td>0</td> <td>1</td> </tr> <tr> <td>6</td> <td>1</td> <td>0</td> </tr> <tr> <td>TOTAL</td> <td>7</td> <td>7</td> </tr> </tbody> </table> <p>Entonces se reemplazarán los valores de la tabla en la ecuación:</p> $NAA = \left(\frac{7 + 7}{2(6)(5)} \right) * 100 = 81,66$ <p>Por lo tanto, la documentación tiene un nivel de dependencia del 81.66%, lo que indica que un cambio en un artefacto de documentación tiene un impacto muy alto en el resto de la documentación.</p>	Nodo	GS	GE	1	1	1	2	2	1	3	1	1	4	2	3	5	0	1	6	1	0	TOTAL	7	7
Nodo	GS	GE																							
1	1	1																							
2	2	1																							
3	1	1																							
4	2	3																							
5	0	1																							
6	1	0																							
TOTAL	7	7																							

5.1.4.4.15. Métricas para la característica de expansibilidad

Para medir que tan expansible es la documentación de software se han definido dos métricas. La primera, denominada PED (Percepción de Expansibilidad de la Documentación) es descrita en la Tabla 5.40, y permite medir la percepción de los usuarios acerca de la capacidad que ofrece la documentación para agregar nuevo contenido o modificar el que ya existe sin afectar a otras características de la documentación como la validez, trazabilidad, integridad, etc.

Tabla 5.40. Indicador de percepción de expansibilidad de la documentación.

Indicador de percepción de expansibilidad de la documentación				
Identificador	PED			
Propósito	Determinar la percepción de los usuarios de la documentación con respecto a la confiabilidad que se maneja en dicha documentación con base en la siguiente encuesta: https://bit.ly/46QSIF5 .			
Unidad	Cantidad entera positiva			
Escala	[20, 100]			
Ecuación	$PED = 10 * \frac{(P16 + P17 + P18 + P19 + P20 + P62)}{3}$			
Variables	<ul style="list-style-type: none"> • P16, P17, P18, P19, P20, P62: Hace referencia al valor entre 1 y 5 que se le puede asignar a las preguntas 16 a 20 y la pregunta 62 según las posibles respuestas en la encuesta. • PED: Indicador de la percepción general de expansibilidad de la documentación. 			
Observaciones	Las opciones de respuesta de la encuesta tienen la siguiente ponderación:			
	Totalmente en desacuerdo	Muy difícil	Totalmente insatisfecho	1
	En desacuerdo	Difícil	Insatisfecho	2
	Neutral	Algo difícil	Algo satisfecho	3
	De acuerdo	Fácil	Satisfecho	4
Totalmente de acuerdo	Muy fácil	Totalmente satisfecho	5	
Ejemplo de utilización	Para aplicar la métrica PED sobre un artefacto de documentación, se debe responder la encuesta que tiene asociada seleccionando una de las opciones de respuesta disponibles en cada pregunta las cuales tienen un puntaje entre 1 y 5 como se describe en las <i>observaciones</i> . Suponiendo que las respuestas a las preguntas P16 a P20 y P62 de la encuesta hayan sido las siguientes:			

	P16: En desacuerdo P17: Algo satisfecho	P18: Muy difícil P19: De acuerdo	P20: Totalmente de acuerdo P62: Fácil
	Entonces se reemplazarán los siguientes valores en la ecuación:		
	$PED = 10 * \frac{(2 + 3 + 1 + 4 + 5 + 4)}{3} = 10 * \frac{(19)}{3} = 63.63\%$		
	Por lo tanto, se tiene una percepción de expansibilidad del 60% en la documentación.		

La segunda métrica, denominada PUP (Porcentaje de Uso de Plantillas) es descrita en la Tabla 5.41, y permite determinar el nivel de uso de las plantillas que ha definido la organización para documentar sus proyectos. Para lograrlo, esta métrica se apoya en la métrica UDP (Utilización de las Plantillas de Documentación) descrita en la Tabla 5.5, la cual permite obtener la proporción de cumplimiento de dichas plantillas. De esta forma, entre mayor sea el porcentaje de fidelidad de los artefactos con su plantilla correspondiente, mayor será la posibilidad de agregar nueva información de manera consistente y confiable; lo que indica una mayor capacidad de expansión de la documentación.

Tabla 5.41. Porcentaje de uso de plantillas.

Porcentaje de uso de plantillas	
Identificador	PUP
Propósito	Determinar la proporción de artefactos que se han construido con base en las plantillas definidas por la organización.
Unidad	Porcentaje
Escala	[0, 100]
Ecuación	$PUP = \left(\frac{\sum_{i=1}^n UPD_i}{n} \right)$
Variables	<ul style="list-style-type: none"> • UPD: Hace referencia a la fidelidad de cada artefacto hacia su plantilla utilizando la métrica UDP para cada artefacto como se describe en la Tabla 5.6. • n: Hace referencia al total de artefactos que componen la documentación del sistema.
Observaciones	Ninguna
Ejemplo de utilización	Para aplicar esta métrica, es necesario calcular el valor de UDP para cada artefacto. Por ejemplo, si se analizan cuatro artefactos cuyos valores de UDP son los siguientes: Doc1 = 54%, Doc2 = 25%, Doc3 = 85%, Doc4 = 48% Entonces se reemplazarán los valores de la tabla en la ecuación: $PUP = \left(\frac{54 + 25 + 85 + 48}{4} \right) = \frac{212}{4} = 54$ Por lo tanto, la documentación ha utilizado en un 54% las plantillas que ha definido la organización para documentar el sistema.

5.1.4.4.16. Métricas para la característica de validez

Para medir la validez de la documentación de software, se han definido dos métricas. La primera, denominada PAE (Porcentaje de artefactos aprobados por la empresa) es descrita en la Tabla 5.42, y permite medir la proporción de artefactos que cuentan con aprobación o visto bueno de parte de la empresa. Dicha aprobación representa la validación del artefacto y puede darse de diferentes formas, por ejemplo, mediante un sello de autenticidad, firmas de altos directivos o líderes, certificados de autenticidad, entre otros.

Tabla 5.42. Porcentaje de artefactos aprobados por la empresa.

Porcentaje de artefactos aprobados por la empresa	
Identificador	PAE
Propósito	Determinar la proporción de artefactos que han sido aprobados por la empresa.
Unidad	Porcentaje
Escala	[0, 100]
Ecuación	$PAE = \left(\frac{TDAP}{TAS} \right) * 100$
Variables	<ul style="list-style-type: none"> • TDAP: Hace referencia a la cantidad de artefactos que han sido aprobados por la empresa. • TAS: Hace referencia a la cantidad de artefactos que componen la documentación del sistema.
Observaciones	Ninguna
Ejemplo de utilización	Para aplicar esta métrica, se debe contar todos los artefactos que cuenten con una aprobación por parte de la empresa y la cantidad de artefactos que componen la documentación del sistema. De esta forma, si

	<p>la documentación del sistema está compuesta por 50 documentos, de los cuales 32 cuentan con una aprobación por parte de la empresa, los valores que se utilizarían en la ecuación serían: $TAS = 50$, $TDAp = 32$</p> <p>Entonces se reemplazarán los valores de la tabla en la ecuación:</p> $PAE = \left(\frac{32}{50}\right) * 100 = 64$ <p>Por lo tanto, el 64% de la documentación del sistema ha sido aprobada por la empresa.</p>
--	--

La segunda métrica, denominada ARA (Artefactos Revisados y Aprobados) es descrita en la Tabla 5.43, y permite determinar la proporción de artefactos que describen claramente un historial de revisiones a probaciones por parte de la empresa. El historial de revisiones puede ser descrito de diversas maneras, pero debería contener, al menos, la siguiente información: Número de revisión, número de versión del documento, versión del sistema a la que está asociado el documento, fecha de revisión, descripción de la revisión o modificación, nombre del revisor, y firma o sello de aprobación.

Tabla 5.43. Artefactos revisados y aprobados.

Porcentaje de artefactos revisados y aprobados	
Identificador	ARA
Propósito	Determinar la proporción de artefactos que cuentan con un historial de revisiones y aprobaciones.
Unidad	Porcentaje
Escala	[0, 100]
Ecuación	$ARA = \left(\frac{\sum_{i=1}^n \frac{SCN_i}{7}}{n}\right) * 100$
Variables	<ul style="list-style-type: none"> • SCN_i: Hace referencia a la cantidad de secciones del historial de revisiones que coinciden con las secciones propuestas en la descripción de esta métrica. • n: Hace referencia a la cantidad de artefactos que componen la documentación del sistema.
Observaciones	<p>Esta métrica considera las siguientes secciones para que el historial de revisiones sea completo:</p> <ul style="list-style-type: none"> • Número de revisión • número de versión del documento • versión del sistema a la que está asociado el documento • fecha de revisión • descripción de la revisión o modificación • nombre del revisor • firma o sello de aprobación
Ejemplo de utilización	<p>Para aplicar esta métrica, se debe contar la cantidad de secciones del historial de revisiones que coinciden en cada documento con las secciones propuestas en esta métrica. De esta forma, si la documentación del sistema está compuesta por 3 documentos, en los cuales coinciden las secciones de la siguiente forma: el documento 1 coincide con dos secciones, el documento 2 con cuatro secciones y el documento 3 con 7 secciones. De esta forma, los valores que se utilizarían en la ecuación serían:</p> <p>Entonces se reemplazarán los valores de la tabla en la ecuación:</p> $ARA = \left(\frac{\frac{2 + 4 + 7}{7}}{3}\right) * 100 = \left(\frac{13}{3}\right) * 100 = 61.9$ <p>Por lo tanto, en toda la documentación del sistema, la información del historial de revisiones solo está completa en un 61.9% con base en las secciones propuestas en esta métrica.</p>

5.1.4.4.17. Métricas para la característica de verificabilidad

Para medir la verificabilidad de la documentación se ha definido la métrica DHR (Documentos con un Historial de Revisiones) es descrita en la Tabla 5.44, la cual permite medir la proporción de artefactos que tienen asociado un archivo sobre el cual se han registrado las revisiones del artefacto para comprender quien fue el responsable de hacer los cambios, en qué momento se hicieron, si fueron aprobados, entre otras cosas. De esta forma, posible determinar el volumen de documentación sobre la cual es posible hacer un seguimiento de su evolución.

Tabla 5.44. Documentos que tienen un historial de comentarios o revisiones.

Documentos que tienen un historial de comentarios o revisiones	
Identificador	DHR
Propósito	Determinar la proporción de artefactos que cuentan con un archivo con los detalles de los diferentes ajustes que se han realizado sobre la documentación del sistema.
Unidad	Porcentaje
Escala	[0, 100]
Ecuación	$DHR = \left(\frac{DAH}{n}\right) * 100$
VARIABLES	<ul style="list-style-type: none"> • DAH: Hace referencia a la cantidad de documentos que tienen asociado un documento de historial de revisiones. • n: Hace referencia a la cantidad de artefactos que componen la documentación del sistema.
Observaciones	En esta métrica se considera que cada artefacto cuenta con un archivo adicional que registra la descripción de todos los cambios que se han realizado en ese artefacto.
Ejemplo de utilización	<p>Para aplicar esta métrica, se debe contar la cantidad de artefactos que tienen asociado un documento donde se registren todos los cambios que se han realizado y también se debe contar la cantidad de documentos que componen el sistema. De esta forma, si la documentación del sistema está compuesta por 15 documentos, en los cuales 6 tienen asociado un archivo de historial, entonces los valores que se utilizarían en la ecuación serían:</p> <p>DAH = 7, n = 15</p> <p>Entonces se reemplazarán los valores de la tabla en la ecuación:</p> $DHR = \left(\frac{7}{15}\right) * 100 = 46.66$ <p>Por lo tanto, solo el 46.66% de la documentación cuenta con un archivo donde se hayan registrado los diferentes cambios realizados, por lo tanto, solo esa proporción de la documentación es verificable.</p>

Capítulo 6. Evaluación del modelo de métricas

En este capítulo se presenta la evaluación del modelo de métricas para medir la deuda de la documentación en el desarrollo ágil de software mediante su aplicación en una organización desarrolladora de software como estudio de caso siguiendo los lineamientos establecidos por Runeson y Höst [79]. A continuación, en la Figura 6.1 se presenta el proceso compuesto por las siguientes fases: (i) diseño del estudio de caso, (ii) preparación para el estudio de caso, (iii) recolección de datos, y (iv) el análisis de los resultados. A continuación, en las siguientes secciones se explican en detalle cada una de las fases del proceso.

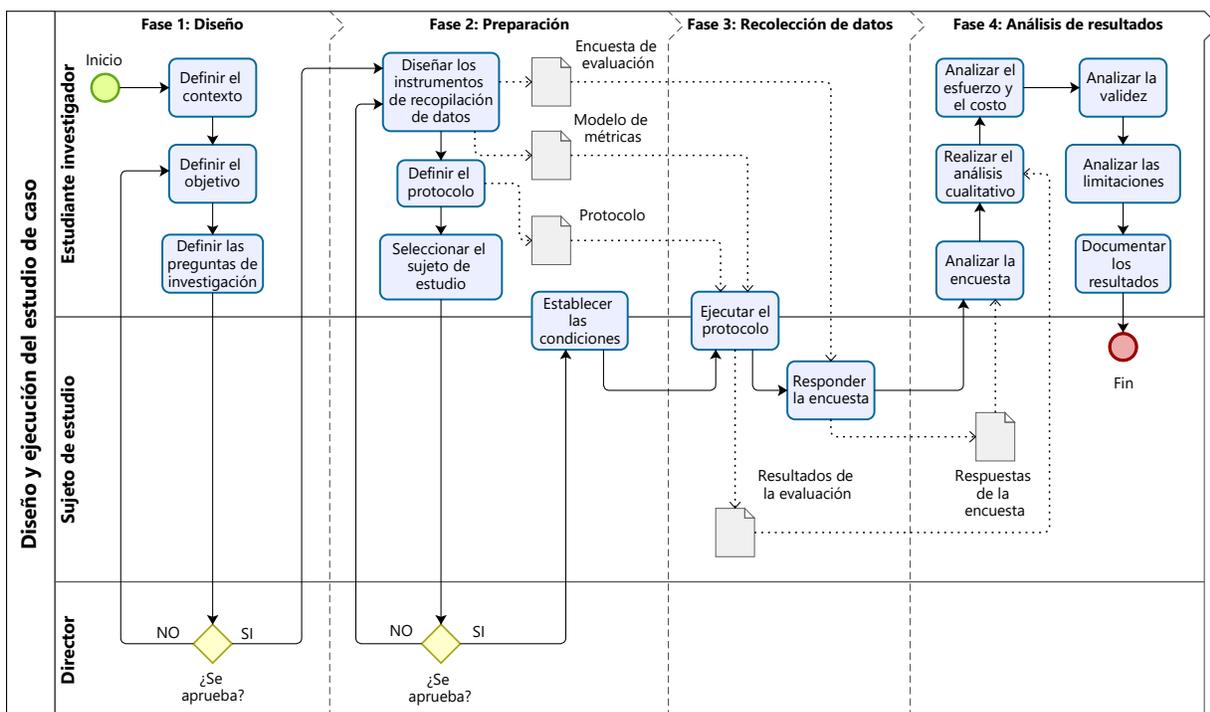


Figura 6.1. Proceso diseñado para realizar el estudio de caso.

6.1. Fase 1: Diseño del estudio de caso

En esta fase se establece el contexto y el propósito del estudio de caso, así como las preguntas de investigación que permitieron orientar la ejecución del estudio y limitar su alcance. A continuación, se describen las actividades de esta fase:

6.1.1. Contexto y propósito

La ejecución del estudio de caso buscó dos objetivos; el primero, establecer una medición del nivel de deuda de la documentación en un proyecto de software ágil; y el segundo, evaluar la capacidad del modelo propuesto para identificar y cuantificar la deuda de la documentación.

Para esto, se cuenta con un instrumento de medición compuesto por 39 métricas relacionadas con las 17 características de la documentación de software descritas en la Sección 3.3 del Capítulo 3 de este documento. Por otro lado, se esperaba que a través del estudio de caso se pudieran identificar factores de riesgo para la documentación y comprender como este tipo de deuda influye en el desarrollo de software. Finalmente, el estudio de caso buscó identificar posibles aspectos de mejora en las prácticas de documentación en proyectos de software ágil para reducir la influencia de este tipo de deuda.

6.1.2. Preguntas de investigación

Con base en el propósito descrito en el apartado anterior, se definió la pregunta de investigación presentada en la Tabla 6.1, de la cual se derivan las preguntas de investigación específicas presentadas en la Tabla 6.2.

Tabla 6.1. Pregunta de investigación principal del estudio de caso.

Pregunta de investigación principal	
	¿El modelo de métricas propuesto permite evaluar la deuda de la documentación en el desarrollo ágil de software de una manera comprensible, útil y práctica en una empresa desarrolladora de software?

Tabla 6.2. Preguntas de investigación específicas del estudio de caso.

Id	Preguntas de investigación específicas
P1	¿El modelo de métricas propuesto les permite a las empresas de software identificar el nivel de deuda en la documentación de sus proyectos de software ágiles?
P2	¿El modelo de métricas propuesto permite identificar oportunidades de mejora en la documentación de software de los proyectos ágiles?
P3	¿El esfuerzo de aplicar el modelo de métricas propuesto es adecuado para medir la deuda de la documentación en proyectos de software ágiles?

Asimismo, teniendo en cuenta la propuesta de Runeson y Höst [79] para clasificar los estudios de caso según su propósito, el estudio de caso presentado en este documento es de carácter exploratorio y holístico con una unidad de análisis, la cual correspondió a la aplicación del modelo de métricas para medir la deuda de la documentación en una organización dedicada al desarrollo de software. Las medidas utilizadas en el estudio de caso fueron: (i) el esfuerzo requerido por la organización para generar un diagnóstico acerca del nivel de deuda en su documentación aplicando el modelo propuesto, (ii) las oportunidades de mejora identificadas luego de aplicar el modelo de métricas en la organización, y (iii) los beneficios percibidos por la organización tras aplicar el modelo propuesto.

6.2. Fase 2: Preparación

En esta fase se realizó una preparación de los instrumentos para la recolección de los datos, se diseñó el protocolo para el estudio de caso y se establecieron las condiciones con la empresa anfitriona para la ejecución del estudio. A continuación, se explican los detalles:

6.2.1. Diseñar los instrumentos de recopilación de datos

Se desarrollaron tres instrumentos para la recopilación de datos con el objetivo de obtener una comprensión exhaustiva y detallada de la deuda de la documentación en el desarrollo ágil de software. En primer lugar, se creó un informe técnico presentado en el Anexo A, el cual presenta el propósito, el alcance y las limitaciones de la propuesta, así como las dimensiones, características de la documentación y métricas definidas para medir el nivel de deuda en la

documentación de software. Esto permitió que los participantes del estudio de caso tuvieran una versión resumida con los detalles de la propuesta que posteriormente se iban a evaluar. En segundo lugar, se diseñó el formulario presentado en el Anexo B el cual permitió capturar todos los datos necesarios que alimentarían el modelo, y de esta manera, realizar la medición de la deuda. Asimismo, se diseñó la encuesta presentada en el Anexo C, el cual permitió evaluar la claridad, aplicabilidad, idoneidad y completitud de la propuesta aplicada en el estudio de caso a partir de las percepciones de los participantes. Finalmente, todos los instrumentos fueron diseñados y revisados para que se complementaran entre sí, y ofrecieran una visión integral que combinara percepciones cualitativas con datos cuantitativos, enriqueciendo así los resultados obtenidos.

6.2.2. Definir el protocolo

Según Runeson Höst [79], la definición de un protocolo en un estudio de caso es fundamental para guiar adecuadamente su ejecución y asegurar su calidad metodológica. En ese sentido, fue diseñado el protocolo presentado en la Tabla 6.3 para formalizar cada etapa del estudio, —las cuales serán explicadas en la sección 6.3.1 del presente documento—.

Tabla 6.3. Protocolo definido para el estudio de caso. Adaptado de [79].

No.	Sección	Descripción
1	Preámbulo	Se realizó una descripción del modelo de métricas a los participantes del estudio de caso para comprender los detalles del instrumento. Asimismo, se brindó toda la información necesaria para entender el contexto, el propósito, el alcance y las limitaciones de la propuesta.
2	Descripción general	Se presentó a los participantes una descripción general del proyecto de investigación destacando el propósito del estudio de caso, el cual está relacionado con entender y medir la deuda de la documentación en el desarrollo ágil de software.
3	Procedimiento	Se explicó de forma detallada como se llevaría a cabo el estudio de caso teniendo en cuenta el tiempo disponible de los participantes y los recursos con los que contarían para la sesión. Asimismo, se resolvieron todas las dudas que tuvieran los participantes acerca de la propuesta o del instrumento.
4	Instrumentos	Se describieron los instrumentos que serían utilizados para la recopilación de datos durante el estudio de caso, la justificación de su elección y su relevancia para cumplir con el propósito del estudio. Puntualmente se presentaron los instrumentos que se definieron en la sección 6.2.1, los cuales son explicados en los anexos B y C de este documento.
5	Resultados	Se describieron las técnicas estadísticas para realizar el análisis y la interpretación de los datos obtenidos en el estudio de caso.

6.2.3. Seleccionar el sujeto de estudio

La selección de la empresa para ejecutar el estudio de caso se realizó con base en los siguientes criterios: (i) que la empresa ofreciera servicios relacionados con el desarrollo de software y se presentaran desafíos asociados con la deuda de la documentación, (ii) que la empresa utilice enfoques ágiles, principalmente Scrum en el desarrollo de sus productos software, y (iii) que la empresa estuviera interesada en llevar a cabo una evaluación del nivel de deuda que existe en la documentación de sus proyectos software con el objetivo de identificar oportunidades de mejora en la gestión de este tipo de deuda.

Como resultado, se seleccionó una empresa desarrolladora de software de carácter regional cuyo nombre no es posible revelarlo por razones de confidencialidad, pero se puede indicar que cuenta con 30 empleados y 15 de años de experiencia en el desarrollo y adecuación de sistemas de información en el campo educativo, redes de datos y de administración de cursos financieros y comerciales. Asimismo, la empresa se destaca por utilizar la metodología Scrum

en la totalidad de sus proyectos, así como la implementación de grupos de desarrollo con un enfoque híbrido de trabajo remoto y presencial.

Desde el punto de vista de la documentación de software, la empresa se caracterizó por haber realizado grandes esfuerzos en los últimos dos años para mejorar sus prácticas de documentación debido a la necesidad de preservar el conocimiento generado por su equipo de desarrollo en cada proyecto, y de esta manera; facilitar la integración de nuevo personal. Sin embargo, sigue presentando grandes dificultades para lograrlo, debido a esto, manifestó un gran interés en la propuesta y permitió un acceso significativo a los datos y recursos necesarios para llevar a cabo un diagnóstico del nivel de deuda en su documentación e identificar opciones de mejora.

6.2.4. Establecer las condiciones

Durante el primer encuentro entre el equipo de investigadores y la organización seleccionada para el estudio de caso, se establecieron acuerdos para la colaboración mutua enfocados en la comprensión de los objetivos del estudio y las expectativas de ambas partes. De esta forma, se establecieron los compromisos para salvaguardar la integridad y confidencialidad de la información a la que se tendría acceso reconociendo su carácter sensible y privado. Estos compromisos incluyeron: (i) limitar el acceso a la información solo a los miembros directamente involucrados en la investigación, (ii) solicitar al jefe del área de desarrollo de manera presencial el acceso a cualquier documento que se requiera, y (iii) garantizar que los resultados finales no revelarán información que pudiese comprometer la competitividad o seguridad de la empresa. Por lo tanto, con este enfoque proactivo hacia la confidencialidad de los resultados del estudio se estableció una base sólida para la colaboración ética y efectiva entre los investigadores y la empresa anfitriona del estudio.

6.3. Fase 3: Recolección de datos

En esta fase se planifica la ejecución del estudio de caso utilizando el protocolo y los instrumentos de recolección de datos definidos en la Fase 2. A continuación se presentan los detalles de esta fase:

6.3.1. Ejecutar el protocolo

6.3.1.1. Planificación de la evaluación

Con el objetivo de planificar la realización del estudio de caso, se estableció un primer contacto presencial con el *director* de la empresa y el *jefe del área de desarrollo* para presentarles el modelo de métricas y realizarles la propuesta de llevar a cabo el estudio de caso. Posteriormente, se realizó una presentación ejecutiva al jefe del área junto a su equipo de desarrollo, la cual tuvo una duración de 45 minutos donde se trataron los siguientes aspectos:

Tabla 6.4. Actividades realizadas durante la planificación.

No	Actividad	Descripción
1	Presentación del proyecto y del equipo investigador	Se presentó el contexto, el objetivo del trabajo de investigación, el alcance y las limitaciones de la propuesta a todos los participantes.
2	Presentación del modelo de métricas	Se presentó el modelo de métricas para medir la deuda de la documentación en proyectos de software ágil a todos los participantes.
3	Asignación de roles	Para esta actividad, el <i>jefe del área de desarrollo</i> asumió la responsabilidad de llevar a cabo el proceso de evaluación debido a su amplio conocimiento en los procesos de desarrollo, a liderar las iniciativas de documentación de

		los proyectos de software de la empresa en los últimos 4 años y a su interés en encontrar oportunidades de mejora que se pudieran aplicar.
4	Cronograma de evaluación	Se acordó un tercer encuentro para ejecutar el estudio de caso de manera presencial con la participación de cinco miembros del equipo de desarrollo.

Una vez finalizado este segundo encuentro, se les entregó a todos los participantes el Informe Técnico del modelo de métricas presentado en el Anexo A de este documento, esto con el fin de realizar una lectura previa al inicio del estudio de caso.

6.3.1.2. Ejecución de la evaluación

La ejecución del estudio de caso contó con la participación de cinco integrantes del equipo de desarrollo de la empresa sobre los cuales se realizó la caracterización de sus perfiles (ver Tabla 6.5) con el propósito de profundizar un poco en sus características individuales.

Tabla 6.5. Caracterización de los participantes del estudio de caso.

Id. Participante	Edad	Profesión	Cargo	Experiencia (meses)
PEC1	25	Ingeniero de sistemas	Desarrollador	6
PEC2	22	Ingeniero de sistemas	Desarrollador	12
PEC3	22	Ingeniero de sistemas	Desarrollador	18
PEC4	26	Ingeniero de sistemas	Desarrollador	12
PEC5	23	Ingeniero de sistemas	Desarrollador	18

Luego de la caracterización de los participantes, se procedió con la aplicación del modelo de métricas sobre cinco historias de usuario, nombradas como HU-009, HU-004, HU-006, HU-005 y HU-008, de uno de los proyectos realizados por la empresa las cuales fueron provistas por el jefe del área de desarrollo para medir el nivel de deuda en cada una de ellas; sin embargo, por razones de confidencialidad no es posible presentar su contenido en este documento. Los participantes utilizaron el formato del Anexo B de este documento, lo que permitió capturar todos los datos que alimentarían el modelo y generar los resultados de la medición con respecto a las 17 características de la documentación definidas en el modelo (ver Sección 3.3, Capítulo 3). Adicionalmente, se definió la escala presentada en la Tabla 6.6 para clasificar el nivel de implementación de cada característica en las historias de usuario analizadas con base en los resultados de medición obtenidos con cada métrica.

Tabla 6.6. Escala de implementación de las características de la documentación.

No.	Clasificación	Acrónimo	Rango	Descripción
1	Totalmente implementada	TI	$\geq 80\%$	Indica que la característica ha sido integrada completamente en la documentación sin dificultades o con errores insignificantes.
2	Altamente implementada	AI	$\geq a 60\% \text{ y } < a 80\%$	Indica que la característica ha sido integrada en la documentación con falencias mínimas que pueden ser corregidas con poco esfuerzo y en poco tiempo.
3	Medianamente implementada	MI	$\geq a 40\% \text{ y } < a 60\%$	Indica que la característica ha sido integrada en documentación con falencias importantes que requieren mucho esfuerzo para ser corregidas.
4	Con implementación básica	IB	$\geq a 10\% \text{ y } < a 40\%$	Indica que la característica ha sido integrada en la documentación de una forma muy básica con falencias críticas que requieren de un esfuerzo extremadamente grande para ser solucionadas.
5	No implementada	NI	$< 10\%$	Indica que la característica aún no ha sido integrada en la documentación.

A continuación, la Tabla 6.7 presenta un consolidado con los resultados de medición de todas las métricas del modelo, así como los resultados por característica y por dimensión, los cuales serán explicados en las secciones posteriores.

Tabla 6.7. Resumen de los resultados de medición obtenidos durante el estudio de caso.

No.	Dimensión	Característica	Métrica	Resultado por métrica	Promedio por característica	Promedio por dimensión	
1	Estructura	Uniformidad	PUD	59.4%	67.9%	51.16%	
			UPD	76.4%			
		Adaptabilidad	PAD	52.8%	52.8%		
		Modularidad	PMD	65.6%	32.8%		
IAD	0%						
2	Formato	Legibilidad	PLD	61%	52.7%	65.11%	
			IFKE	44.4%			
		Concisión	PCD	62.65%	62.65%		
			Claridad	PCLD			80%
				PPO			23
PFO	1						
3	Usabilidad	Accesibilidad	PACD	60.8%	79.1%	38.25%	
			SAS	97.4%			
		Confidencialidad	PConD	55%	45.8%		
			AMC	0%			
			DNA	100%			
			DLA	100%			
			DEC	0%			
			CIL	20%			
		Consistencia	PRI	0%	0%		
			DRD	0%			
		Precisión	PPD	54.4%	27.2%		
			RMT	0%			
		4	Correlación	Portabilidad	PPRD		73%
DIP	0%						
DCE	100%						
Integridad	PID			32%	16%		
	PAAS			0%			
Actualidad	PAcD			52%	26.44%		
	PARev			0%			
	PIRU			27.34%			
Trazabilidad	CDT			100%	50%		
	NDD			0			
	NAA			0%			
Expansibilidad	PED	72.5%	74.45%				
	PUP	76.4%					
5	Auditabilidad	Validez	PAE	100%	64.3%	32.15%	
			ARA	28.57%			
		Verificabilidad	DHR	0%	0%		

6.3.1.2.1. Medición de la característica de uniformidad

Como se describió en la Sección 5.1.4.4.1 en la página 64 del presente documento, la característica de uniformidad tiene asociadas dos métricas: PUD (Percepción de Uniformidad de la Documentación) y UPD (Utilización de las Plantillas de Documentación). Para la métrica PUD los participantes respondieron la encuesta asociada donde las respuestas de cada pregunta tienen una ponderación de 1 a 5 como se explica en la Tabla 5.5. De esta forma, cada participante aplicó la encuesta a las historias de usuario registrando las ponderaciones en las columnas *P1* a *P10* en la Tabla 6.8, así como el resultado de la medición de uniformidad por cada historia de usuario registrado en la columna *medición* utilizando la ecuación de la métrica. De esta forma, los participantes perciben que las historias de usuario analizadas tienen un nivel de uniformidad del **59.4%**, es decir; Medianamente Implementado (MI)

Tabla 6.8. Resultados de aplicación de la métrica PDU.

Resultados de la métrica PDU: Percepción de uniformidad de la documentación													
Participante	H.U	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	Medición	Escala
PEC1	HU-009	4	5	3	2	4	3	2	3	2	2	60%	AI
PEC2	HU-004	3	4	2	3	5	2	3	3	4	1	60%	AI
PEC3	HU-006	3	4	3	1	4	4	2	3	4	3	61%	AI

PEC4	HU-005	3	4	4	3	3	1	1	3	1	2	50%	MI
PEC5	HU-008	3	4	3	2	5	3	3	3	4	3	66%	AI
Nivel de uniformidad percibido en las historias de usuario												59.4%	MI

Acrónimos: H.U: Historia de Usuario evaluada, P: Pregunta de la encuesta, **Medición**: Aplicación de la métrica.

Para la métrica UPD, los participantes analizaron la correspondencia entre las secciones de la plantilla definida por la empresa para la creación de historias de usuario y la estructura de las historias de usuario analizadas. Para cada una, se aplica la métrica descrita en la Tabla 5.6, con la cual se obtienen los resultados presentados en la Tabla 6.9. Como resultado, las historias de usuario analizadas tienen un nivel de fidelidad del **76.4%** con la plantilla definida por la empresa para documentar las historias de usuario de sus proyectos, es decir; Altamente Implementada (AI).

Tabla 6.9. Resultados de aplicación de la métrica UPD.

Resultados de la métrica UPD: Utilización de las Plantillas de Documentación					
Participante	H.U	SP: ¿Cuántas secciones tiene la plantilla definida por la empresa?	SC: ¿Cuántas secciones del artefacto coinciden con la plantilla?	Medición	Escala
PEC1	HU-009	12	6	50%	MI
PEC2	HU-004	12	10	83%	TI
PEC3	HU-006	12	10	83%	TI
PEC4	HU-005	12	10	83%	TI
PEC5	HU-008	12	10	83%	TI
Nivel de fidelidad de la documentación con su plantilla				76.4%	AI

Acrónimos: H.U: Historia de Usuario evaluada, SP: Número de secciones de la plantilla, SC: Número de secciones que coinciden con la plantilla, **Medición**: Aplicación de la métrica.

6.3.1.2.2. Medición de la característica de adaptabilidad

Como se describió en la Sección 5.1.4.4.2, página 65 del presente documento, la característica de adaptabilidad tiene asociada la métrica PAD (Percepción de Adaptabilidad de la Documentación), en la cual los participantes respondieron la encuesta asociada donde las respuestas de cada pregunta tienen una ponderación de 1 a 5 como se explica en la Tabla 5.7. De esta forma, cada participante aplicó la encuesta a las historias de usuario seleccionadas registrando las ponderaciones en las columnas P11 a P15 en la Tabla 6.10, así como el resultado de la medición de adaptabilidad por cada historia de usuario registrado en la columna *medición* utilizando la ecuación de la métrica. De esta forma, los participantes perciben que las historias de usuario analizadas tienen un nivel de adaptabilidad del **52.8%**, es decir; Medianamente Implementada (MI).

Tabla 6.10. Resultados de aplicación de la métrica PAD.

Resultados de la métrica PAD: Percepción de adaptabilidad de la documentación								
Participante	H.U	P11	P12	P13	P14	P15	Medición	Escala
PEC1	HU-009	3	1	4	3	1	48%	MI
PEC2	HU-004	4	2	1	3	1	44%	MI
PEC3	HU-006	5	1	3	4	1	56%	MI
PEC4	HU-005	4	1	4	4	2	60%	AI
PEC5	HU-008	4	3	2	3	2	56%	MI
Nivel de adaptabilidad percibido							52.8%	MI

Acrónimos: H.U: Historia de Usuario evaluada, P: Pregunta de la encuesta, **Medición**: Aplicación de la métrica.

6.3.1.2.3. Medición de la característica de modularidad

Como se describió en la sección 5.1.4.4.3, página 66 del presente documento, la característica de modularidad tiene asociadas las métricas: PMD (Percepción de Modularidad de la Documentación) e IAD (índice de Acoplamiento de la Documentación). Para la métrica PMD, los participantes evaluaron las historias de usuario respondiendo la encuesta asociada donde las respuestas de cada pregunta tienen una ponderación de 1 a 5 como se explica en la Tabla 5.8. De esta forma, cada participante aplicó la encuesta a las historias de usuario

seleccionadas registrando las ponderaciones en las columnas *P16* a *P20* en la Tabla 6.11, y el resultado de la medición en la columna *medición*. De esta forma, los participantes perciben que las historias de usuario analizadas tienen un nivel de modularidad del **65.6%**, es decir; Altamente Implementada (AI).

Por otro lado, en la Tabla 6.12 se presenta de las respuestas a tres preguntas realizadas de manera conjunta a todos los participantes para construir el grafo dirigido que representa la relación entre las historias de usuario analizadas. Como se puede apreciar en la imagen del grafo resultante presentada en la Tabla 6.12 las historias de usuario analizadas no presentaban ningún tipo de relación o dependencia evidente entre ellas, por lo tanto, la cantidad de aristas y su peso se toman como 0 para el cálculo de la métrica. De esta forma, el nivel de acoplamiento de la documentación es de 0%, es decir; No Implementado (NI).

Tabla 6.11. Resultados de aplicación de la métrica PMD.

Resultados de la métrica PMD: Percepción de modularidad de la documentación								
Participante	H.U	P16	P17	P18	P19	P20	Medición	Escala
PEC1	HU-009	4	4	3	3	4	72%	AI
PEC2	HU-004	4	2	3	3	2	56%	MI
PEC3	HU-006	3	4	3	4	4	72%	AI
PEC4	HU-005	4	3	4	3	3	68%	AI
PEC5	HU-008	3	2	4	3	3	60%	AI
Nivel de modularidad percibido							65.6%	AI

Acrónimos: **H.U:** Historia de Usuario evaluada, **P:** Pregunta de la encuesta, **Medición:** Aplicación de la métrica.

Tabla 6.12. Resultados de aplicación de la métrica IAD.

Resultados de la métrica IAD: índice de acoplamiento de la documentación					
Pregunta	Valor	Representación del grafo		Medición	Escala
TNG: ¿Cuántos documentos se utilizaron en el grafo?	5			0%	NI
PA: ¿Cuál es la suma de los pesos de todas las aristas del grafo?	0				
N: ¿Cuántas aristas tiene el grafo?	0				
Nivel de acoplamiento de la documentación				0%	NI

Acrónimos: **H.U:** Historia de Usuario evaluada, **Medición:** Aplicación de la métrica.

6.3.1.2.4. Medición de la característica de legibilidad

Como se describió en la sección 5.1.4.4.4, página 67 del presente documento, la característica de legibilidad tiene asociadas las métricas: PLD (Percepción de Legibilidad de la Documentación) e IFKE (Índice Flesch-Kincaid para el idioma español). Para la métrica PLD, los participantes evaluaron las historias de usuario respondiendo la encuesta asociada donde las respuestas de cada pregunta tienen una ponderación de 1 a 5 como se explica en la Tabla 5.10. De esta forma, cada participante aplicó la encuesta a las historias de usuario seleccionadas registrando las ponderaciones en las columnas *P21* a *P224* en la Tabla 6.13, y el resultado de la medición en la columna *medición*. De esta forma, los participantes perciben que las historias de usuario analizadas tienen un nivel de modularidad del **61%**, es decir; Altamente Implementada (AI).

Para la métrica IFKE, los participantes respondieron tres preguntas con las cuales se recogieron la cantidad de palabras, oraciones y sílabas en la descripción de cada historia de usuario para calcular su nivel de legibilidad mediante la ecuación asociada a la métrica. De esta forma se registró en la columna *medición* el índice de legibilidad de cada historia de

usuario, logrando un promedio de legibilidad del **44.4%**, es decir; Medianamente Implementada (MI), lo que hace que sea un texto difícil de leer según la escala de la métrica.

Tabla 6.13. Resultados de aplicación de la métrica PLD.

Resultados de la métrica PLD: Percepción de legibilidad de la documentación							
Participante	H.U	P21	P22	P23	P24	Medición	Escala
PEC1	HU-009	3	3	5	3	70%	AI
PEC2	HU-004	2	1	4	3	50%	MI
PEC3	HU-006	3	1	4	3	55%	MI
PEC4	HU-005	3	2	5	1	55%	MI
PEC5	HU-008	4	3	4	4	75%	AI
Nivel de legibilidad percibido						61%	AI

Acrónimos: **H.U:** Historia de Usuario evaluada, **P:** Pregunta de la encuesta, **Medición:** Aplicación de la métrica.

Tabla 6.14. Resultados de aplicación de la métrica IFKE.

Resultados de la métrica IFKE: Índice Flesch-Kincaid para el idioma español						
Participante	H.U	NTP: ¿Cuántas palabras tiene el artefacto?	NTO: ¿Cuántas oraciones tiene el artefacto?	NTS: ¿Cuántas sílabas tiene el artefacto?	Medición	Escala
PEC1	HU-009	29	1	67	38.64%	IB
PEC2	HU-004	18	1	41	51.81%	MI
PEC3	HU-006	19	1	45	45.35%	MI
PEC4	HU-005	30	1	74	28.24%	IB
PEC5	HU-008	19	1	41	57.98%	MI
Nivel de legibilidad de la documentación según el Índice Flesch-Kincaid					44.4%	MI

Acrónimos: **H.U:** Historia de Usuario evaluada, **NTP:** Número de palabras, **NTO:** Número de oraciones, **NTS:** Número de sílabas, **Medición:** Aplicación de la métrica.

6.3.1.2.5. Medición de la característica de concisión

Como se describió en la sección 5.1.4.4.5, página 68 del presente documento, la característica de legibilidad tiene asociadas las métricas: PLD (Percepción de Legibilidad de la Documentación) e IFKE (Índice Flesch-Kincaid para el idioma español). Para la métrica PLD, los participantes evaluaron las historias de usuario respondiendo la encuesta asociada donde las respuestas de cada pregunta tienen una ponderación de 1 a 5 como se explica en la Tabla 5.12. De esta forma, cada participante aplicó la encuesta a las historias de usuario seleccionadas registrando las ponderaciones en las columnas *P25* a *P30* en la Tabla 6.15, y el resultado de la medición en la columna *medición*. De esta forma, los participantes perciben que las historias de usuario analizadas tienen un nivel de modularidad del **62.65%**, es decir; Altamente Implementada (AI).

Tabla 6.15. Resultados de aplicación de la métrica PCD.

Resultados de la métrica PCD: Percepción de concisión de la documentación									
Participante	H.U	P25	P26	P27	P28	P29	P30	Medición	Escala
PEC1	HU-009	4	3	4	4	3	4	73.3%	AI
PEC2	HU-004	3	4	3	3	4	3	66.6%	AI
PEC3	HU-006	4	4	5	3	4	4	80%	TI
PEC4	HU-005	4	4	4	4	4	5	83.3%	TI
PEC5	HU-008	5	4	5	3	4	4	70%	AI
Nivel de concisión percibido								62.65%	AI

Acrónimos: **H.U:** Historia de Usuario evaluada, **P:** Pregunta de la encuesta, **Medición:** Aplicación de la métrica.

6.3.1.2.6. Medición de la característica de claridad

Como se describió en la sección 5.1.4.4.6, página 69 del presente documento, esta característica tiene asociadas las métricas: PCLD (Percepción de la claridad de la documentación), PPO (Promedio de Palabras por Oración) y PFO (Promedio de Frases por Oración). Para la métrica PCD, los participantes evaluaron las historias de usuario respondiendo la encuesta asociada donde las respuestas de cada pregunta tienen una

ponderación de 1 a 5 como se explica en la Tabla 5.13. De esta forma, cada participante aplicó la encuesta a las historias de usuario seleccionadas registrando las ponderaciones en las columnas *P31* a *P34* en la Tabla 6.16, y el resultado de la medición en la columna *medición*. De esta forma, los participantes perciben que las historias de usuario analizadas tienen un nivel de legibilidad del **80%**, es decir; Totalmente Implementada (TI).

Tabla 6.16. Resultados de aplicación de la métrica PCLD.

Resultados de la métrica PCLD: Percepción de claridad de la documentación						
Participante	H.U	P31	P32	P33	P34	Medición
PEC1	HU-009	4	3	4	5	80%
PEC2	HU-004	3	4	4	4	75%
PEC3	HU-006	4	4	5	4	85%
PEC4	HU-005	4	3	5	4	80%
PEC5	HU-008	4	4	4	4	80%
Nivel de claridad percibido						80%

Acrónimos: **H.U:** Historia de Usuario evaluada, **P:** Pregunta de la encuesta, **Medición:** Aplicación de la métrica.

Por otro lado, para las métricas PPO y PFO, se registraron los valores presentados en las Tablas 6.17 y 6.18, relacionados con la cantidad de palabras, oraciones y frases utilizadas en la descripción de cada historia de usuario. A partir de estos valores, se determinó que cada historia de usuario analizada maneja en promedio 23 palabras y 1 frase por oración.

Tabla 6.17. Resultados de aplicación de la métrica PPO.

Resultados de la métrica PPO: Promedio de palabras por oración				
Participante	H.U	TPC: ¿Cuántas palabras tiene el texto?	TOC: ¿Cuántas oraciones tiene el texto?	Medición
PEC1	HU-009	29	1	29
PEC2	HU-004	18	1	18
PEC3	HU-006	19	1	19
PEC4	HU-005	30	1	30
PEC5	HU-008	19	1	19
Promedio de palabras utilizadas por oración				23

Acrónimos: **H.U:** Historia de Usuario evaluada, **TPC:** Número de palabras, **TOC:** Número de oraciones, **Medición:** Aplicación de la métrica.

Tabla 6.18. Resultados de aplicación de la métrica PFO.

Resultados de la métrica PFO: Promedio de frases por oración				
Participante	H.U	TFC: ¿Cuántas frases tiene el texto?	TOC: ¿Cuántas oraciones tiene el texto?	Medición
PEC1	HU-009	1	1	1
PEC2	HU-004	1	1	1
PEC3	HU-006	1	1	1
PEC4	HU-005	1	1	1
PEC5	HU-008	1	1	1
Promedio de frases utilizadas por oración				1

Acrónimos: **H.U:** Historia de Usuario evaluada, **TFC:** Número de frases, **TOC:** Número de oraciones, **Medición:** Aplicación de la métrica.

6.3.1.2.7. Medición de la característica de accesibilidad

Como se describió en la sección 5.1.4.4.7, página 71 del presente documento, esta característica tiene asociadas las métricas: PACD (Percepción de Accesibilidad de la Documentación) y SAS (Índice de solicitudes de acceso satisfactorias). Para la métrica PACD, los participantes evaluaron las historias de usuario respondiendo la encuesta asociada donde las respuestas de cada pregunta tienen una ponderación de 1 a 5 como se explica en la Tabla 5.16. De esta forma, cada participante aplicó la encuesta a las historias de usuario seleccionadas registrando las ponderaciones en las columnas *P35* a *P39* en la Tabla 6.19, y el resultado de la medición en la columna *medición*. De esta forma, los participantes perciben que las historias de usuario analizadas tienen un nivel de legibilidad del **60.8%**, es decir; Altamente Implementada (AI).

Por otro lado, para la métrica SAS se preguntó a los participantes por la cantidad de solicitudes de acceso han realizado sobre cada historia de usuario y cuantas han sido respondidas satisfactoriamente. Con esto se determinó que las solicitudes de acceso a las historias de usuario son del **97.4%** satisfactorias, es decir; Totalmente Implementado (TI).

Tabla 6.19. Resultados de aplicación de la métrica PACD.

Resultados de la métrica PACD: Percepción de accesibilidad de la documentación								
Participante	H.U	P35	P36	P37	P38	P39	Medición	Escala
PEC1	HU-009	3	2	3	2	1	44%	MI
PEC2	HU-004	5	5	4	5	5	96%	TI
PEC3	HU-006	3	4	2	3	1	52%	MI
PEC4	HU-005	2	1	1	3	2	36%	IB
PEC5	HU-008	3	4	4	4	4	76%	AI
Nivel de accesibilidad percibido							60.8%	AI

Acónimos: **H.U:** Historia de Usuario evaluada, **P:** Pregunta de la encuesta, **Medición:** Aplicación de la métrica.

Tabla 6.20. Resultados de aplicación de la métrica SAS.

Resultados de la métrica SAS: Solicitudes de acceso satisfactorias					
Participante	H.U	SAR: ¿Cuántas solicitudes de acceso se han realizado?	SAA: ¿Cuántas solicitudes de acceso han sido aprobadas?	Medición	Escala
PEC1	HU-009	2	2	100%	TI
PEC2	HU-004	10	10	100%	TI
PEC3	HU-006	10	10	100%	TI
PEC4	HU-005	10	10	100%	TI
PEC5	HU-008	7	8	87%	TI
Promedio de solicitudes de acceso aprobadas				97.4%	TI

Acónimos: **H.U:** Historia de Usuario evaluada, **TFC:** Número de frases, **TOC:** Número de oraciones, **Medición:** Aplicación de la métrica.

Dentro de la evaluación de la característica de accesibilidad, se realizó la pregunta *P40* a los participantes, la cual no se aplica para ninguna métrica, pero si busca conocer las preferencias de los participantes para acceder a la documentación de software. La pregunta en específico fue: *P40: ¿Cuál de los siguientes formatos prefiere al momento de consultar información?* Con las siguientes opciones de respuesta: Documentos físicos, Documentos digitales (PDF o Word), Documentos en línea (sitio web o wiki) y Aplicación móvil. De esta forma, en la Figura 6.2 se aprecia como los participantes no desean utilizar documentos físicos y prefieren los documentos digitales en formato PDF o Word y la documentación en línea a través de un sitio web o una wiki y en menor medida la documentación a través de una aplicación móvil.

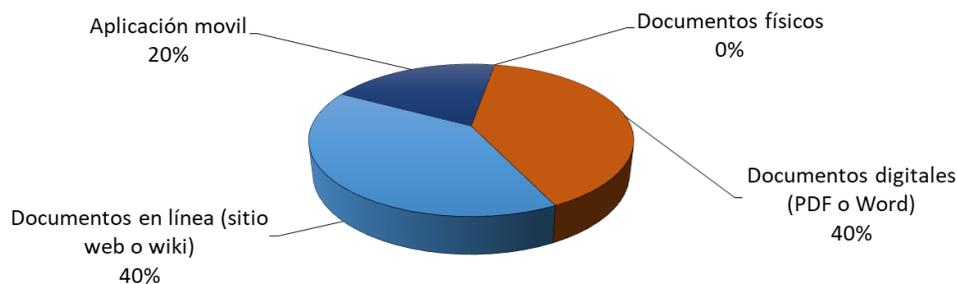


Figura 6.2. Preferencias sobre el formato de la documentación.

6.3.1.2.8. Medición de la característica de confidencialidad

Como se describió en la sección 5.1.4.4.8, página 72 del presente documento, esta característica tiene asociadas las métricas: PConD (Percepción de Confidencialidad de la Documentación), AMC (Artefactos con Marcas de Confidencialidad), DNA (Definición de Niveles de Acceso), DLA (Documentación de Libre Acceso), DEC (Documentación Extremadamente Confidencial) y CIL (Conocimiento de Implicaciones Legales). Para la

métrica PConD, los participantes evaluaron las historias de usuario respondiendo la encuesta asociada donde las respuestas de cada pregunta tienen una ponderación de 1 a 5 como se explica en la Tabla 5.18. De esta forma, cada participante aplicó la encuesta a las historias de usuario seleccionadas registrando las ponderaciones en las columnas P41 a P44 en la Tabla 6.21, y el resultado de la medición en la columna *medición*. De esta forma, los participantes perciben que las historias de usuario tienen un nivel de confidencialidad del **55%**, es decir; Medianamente Implementada (MI).

Tabla 6.21. Resultados de aplicación de la métrica PConD.

Resultados de la métrica PConD: Percepción de confidencialidad de la documentación							
Participante	H.U	P41	P42	P43	P44	Medición	Escala
PEC1	HU-009	1	3	3	4	55%	MI
PEC2	HU-004	1	3	4	5	65%	AI
PEC3	HU-006	2	4	3	3	60%	AI
PEC4	HU-005	1	2	3	2	40%	MI
PEC5	HU-008	1	3	3	4	55%	MI
Nivel de confidencialidad percibido						55%	MI

Acronimos: **H.U:** Historia de Usuario evaluada, **P:** Pregunta de la encuesta, **Medición:** Aplicación de la métrica.

Para la métrica AMC, se registraron los valores presentados en la Tabla 6.22, relacionados con la cantidad de historias de usuario que tienen etiquetas, marcas o algún tipo de indicación de su nivel de confidencialidad. A partir de estos valores pudo determinarse que ninguna de las historias de usuario analizadas tiene definido un nivel de confidencialidad, es decir que No está Implementada (NI).

Tabla 6.22. Resultados de aplicación de la métrica AMC.

Resultados de la métrica AMC: Artefactos con marcas de confidencialidad					
Participante	H.U	TAE: ¿Cuántos artefactos fueron analizados?	TAM: ¿Cuántos artefactos tienen marcas o etiquetas de confidencialidad?	Medición	Escala
PEC1	HU-009	1	0	0%	NI
PEC2	HU-004	1	0	0%	NI
PEC3	HU-006	1	0	0%	NI
PEC4	HU-005	1	0	0%	NI
PEC5	HU-008	1	0	0%	NI
Promedio de artefactos con marcas de confidencialidad				0%	NI

Acronimos: **H.U:** Historia de Usuario evaluada, **TAE:** Artefactos analizados, **TAM:** Artefactos con marcas de confidencialidad, **Medición:** Aplicación de la métrica.

Para la métrica DNA, se preguntó a los participantes cuantos de ellos tienen claro su nivel de acceso a las historias de usuario analizadas, registrando este valor en la Tabla 6.23. De esta forma se pudo determinar que todos los miembros del equipo de desarrollo que participaron del estudio de caso tienen claro el nivel de acceso sobre las historias de usuario analizadas, es decir que está Totalmente Implementada (TI).

Tabla 6.23. Resultados de aplicación de la métrica DNA.

Resultados de la métrica DNA: Índice de definición de niveles de acceso a la documentación			
Pregunta	Valor	Medición	Escala
MCNA: ¿Cuántos miembros del equipo tienen claro su nivel de acceso a la documentación?	5	100%	TI
ME: ¿Cuántos miembros tiene el equipo?	5		
Porcentaje del equipo que tiene claro su nivel de acceso a la documentación		100%	TI

Acronimos: **MCNA:** Miembros que conocen su nivel de acceso, **ME:** Miembros del equipo, **Medición:** Aplicación de la métrica.

Para la métrica DLA, se les preguntó a los participantes cuantas de las historias de usuario analizadas en el estudio de caso son de acceso público o no tienen un nivel de acceso definido, registrando los valores en la Tabla 6.24, donde se puede apreciar como la totalidad de las historias de usuario analizadas no tienen un nivel de acceso definido, y por consiguiente; cualquier persona puede tener acceso a su información. Para la métrica DEC, se les preguntó

a los participantes cuantas de las historias analizadas cuentan con el máximo nivel de confidencialidad, registraron los valores en la Tabla 6.25, donde se evidencia que ninguna de las historias de usuario maneja el máximo nivel de confidencialidad. Esto es consecuente con la métrica DNA, ya que ninguna historia de usuario tiene un nivel de confidencialidad definido.

Tabla 6.24. Resultados de aplicación de la métrica DLA.

Resultados de la métrica DLA: Índice de documentación de libre acceso			
Pregunta	Valor	Medición	Escala
TDP: ¿Cuántos documentos son de carácter público?	0	100%	TI
TDSN: ¿Cuántos documentos no tienen definido un nivel de acceso?	5		
TDS: ¿Cuántos documentos componen el sistema o fueron analizados?	5		
Porcentaje de documentación de acceso libre		100%	TI

Acrónimos: **TDP**: Documentos públicos, **TDSN**: Documentos sin acceso definido, **TDS**: Documentos analizados, **Medición**: Aplicación de la métrica.

Tabla 6.25. Resultados de aplicación de la métrica DEC.

Resultados de la métrica DEC: Índice de documentación extremadamente confidencial			
Pregunta	Valor	Medición	Escala
TDC: ¿Cuántos documentos cuentan con el máximo nivel de confidencialidad?	0	0%	NI
TDS: ¿Cuántos documentos componen el sistema o fueron analizados?	5		
Porcentaje de documentación extremadamente confidencial		0%	NI

Acrónimos: **TDC**: Documentos extremadamente confidenciales, **TDS**: Documentos analizados, **Medición**: Aplicación de la métrica.

Para la métrica CIL, se les preguntó a los participantes cuantos de ellos tienen claras las implicaciones legales de difundir información confidencial sobre la empresa y sus proyectos. De esta forma, se registraron los valores presentados en la Tabla 6.26, donde se puede apreciar que solamente el 20% de los participantes conocen dichas implicaciones legales, es decir; que cuenta con una implementación básica (IB).

Tabla 6.26. Resultados de aplicación de la métrica CIL.

Resultados de la métrica CIL: Índice de conocimiento de implicaciones legales			
Pregunta	Valor	Medición	Escala
MCI: ¿Cuántos miembros del equipo conocen las implicaciones legales	1	20%	IB
ME: ¿Cuántos miembros tiene el equipo?	5		
Porcentaje miembros del equipo que conocen las implicaciones legales de divulgar información		20%	IB

Acrónimos: **MCI**: Miembros del equipo que conocen las implicaciones legales, **ME**: Miembros del equipo, **Medición**: Aplicación de la métrica.

6.3.1.2.9. Medición de la característica de consistencia

Como se describió en la sección 5.1.4.4.9, página 75 del presente documento, esta característica asocia las métricas: PRI (Porcentaje de Relaciones Incorrectas) y DRD (Densidad de Relaciones de la Documentación) que dependen de la métrica IAD presentado en la Tabla 6.12. De esta forma, se registraron los valores en la Tabla 6.27 para la métrica PRI relacionados con la cantidad de aristas incorrectas del grafo y en la Tabla 6.28 para la métrica DRD relacionados con la cantidad de aristas sin ponderación. En ambos casos, se obtuvo un valor de **0%** debido a que las historias de usuario analizadas no se relacionaban entre sí, es decir; No Implementada (NI).

Tabla 6.27. Resultados de aplicación de la métrica PRI.

Resultados de la métrica PRI: Porcentaje de relaciones incorrectas			
Pregunta	Valor	Medición	Escala
TRI: ¿Cuántas relaciones del grafo son incorrectas?	0	0%	NI
TRG: ¿Cuántas relaciones o aristas tiene el grafo?	0		
Porcentaje de relaciones incorrectas o innecesarias entre documentos		0%	NI

Acrónimos: **TRI**: Relaciones incorrectas del grafo, **TRG**: Relaciones en el grafo, **Medición**: Aplicación de la métrica.

Tabla 6.28. Resultados de aplicación de la métrica DRD.

Resultados de la métrica DRD: Densidad de relaciones de la documentación			
Pregunta	Valor	Medición	Escala
TAG: ¿Cuántas aristas tiene el grafo sin tener en cuenta su ponderación?	0	0%	NI
TNG: ¿Cuántas nodos tiene el grafo?	0		
Porcentaje de densidad de las relaciones entre documentos		0%	NI

Acronimos: **TRI:** Relaciones incorrectas del grafo, **TRG:** Relaciones en el grafo, **Medición:** Aplicación de la métrica.

6.3.1.2.10. Medición de la característica de precisión

Como se describió en la sección 5.1.4.4.10, página 76 del presente documento, esta característica asocia las métricas: PPD (Percepción de la Precisión de la Documentación) y RMT (Requisitos en la Matriz de Trazabilidad). Para la métrica PPD, los participantes evaluaron las historias de usuario respondiendo la encuesta asociada donde las respuestas de cada pregunta tienen una ponderación de 1 a 5 como se explica en la Tabla 5.26. De esta forma, cada participante aplicó la encuesta a las historias de usuario seleccionadas registrando las ponderaciones en las columnas *P45* a *P49* en la Tabla 6.29, y el resultado de la medición en la columna *medición*. De esta forma, los participantes perciben que las historias de usuario analizadas tienen un nivel de precisión del **55.4%**, es decir; Medianamente Implementada (MI).

Tabla 6.29. Resultados de aplicación de la métrica PPD.

Resultados de la métrica PPD: Indicador de percepción de precisión de la documentación								
Participante	H.U	P45	P46	P47	P48	P49	Medición	Escala
PEC1	HU-009	3	4	2	1	3	52%	MI
PEC2	HU-004	2	3	2	2	3	48%	MI
PEC3	HU-006	3	3	1	2	3	48%	MI
PEC4	HU-005	3	4	2	2	2	52%	MI
PEC5	HU-008	4	4	2	4	4	72%	AI
Nivel de precisión percibida							54.4%	MI

Acronimos: **H.U:** Historia de Usuario evaluada, **P:** Pregunta de la encuesta, **Medición:** Aplicación de la métrica.

Para la métrica RMT, se preguntó a los participantes si se utilizaba una matriz de trazabilidad o una estructura similar para mantener una trazabilidad de las historias de usuario analizadas, a lo cual respondieron que no se utilizaba. Debido a esto, el porcentaje de requisitos que pueden rastrearse es de **0%**, es decir; No Implementada (NI).

Tabla 6.30. Resultados de aplicación de la métrica RMT.

Resultados de la métrica RMT: Cobertura de requisitos en la matriz de trazabilidad			
Pregunta	Valor	Medición	Escala
NRC: ¿Cuántos requisitos se han cubierto en una matriz de trazabilidad o similares?	0	0%	NI
NTR: ¿Cuántos requisitos componen el sistema o han sido analizados?	5		
Porcentaje de requisitos en la matriz de trazabilidad		0%	NI

Acronimos: **NRC:** Requisitos en la matriz de trazabilidad, **NTR:** Requisitos del sistema, **Medición:** Aplicación de la métrica.

6.3.1.2.11. Medición de la característica de portabilidad

Como se describió en la sección 5.1.4.4.11, página 77 del presente documento, esta característica asocia las métricas: PPRD (Percepción de la Portabilidad de la Documentación), DIP (Documentación Independiente de las Plataformas) y DCE (Documentación que requiere configuraciones especiales). Para la métrica PPRD, los participantes evaluaron las historias de usuario respondiendo la encuesta asociada donde las respuestas de cada pregunta tienen una ponderación de 1 a 5 como se explica en la Tabla 5.29. De esta forma, cada participante aplicó la encuesta a las historias de usuario seleccionadas registrando las ponderaciones en las columnas *P50* a *P53* en la Tabla 6.31, y el resultado de la medición en la columna *medición*.

De esta forma, los participantes perciben que las historias de usuario analizadas tienen un nivel de portabilidad del **73%**, es decir; Altamente Implementada (AI).

Tabla 6.31. Resultados de aplicación de la métrica PPRD.

Resultados de la métrica PPRD: Percepción de portabilidad de la documentación							
Participante	H.U	P50	P51	P52	P53	Medición	Escala
PEC1	HU-009	4	4	4	4	80%	TI
PEC2	HU-004	5	4	5	5	95%	TI
PEC3	HU-006	4	3	3	3	65%	AI
PEC4	HU-005	3	4	2	1	50%	MI
PEC5	HU-008	4	4	4	3	75%	AI
Nivel de portabilidad percibido						73%	AI

Acrónimos: H.U: Historia de Usuario evaluada, P: Pregunta de la encuesta, Medición: Aplicación de la métrica.

Para la métrica DIP, los participantes revisaron cuantas historias de usuario se han construido en alguno de los formatos presentados en la descripción de la Tabla 5.30, ya que estos son los formatos de carácter independiente que han sido considerados en la métrica, es decir que no dependen de un Sistema Operativo o plataforma específica. De esta forma, se evidenció que ninguna de las historias de usuario analizadas fue construida en un formato independiente, es decir; No implementada (NI).

Tabla 6.32. Resultados de aplicación de la métrica DIP.

Resultados de la métrica DIP: Documentación independiente de plataformas específicas			
Pregunta	Valor	Medición	Escala
TAFI: ¿Cuántos artefactos están contruidos en un formato independiente?	0	0%	NI
TAS: ¿Cuántos artefactos componen el sistema o han sido analizados?	5		
Porcentaje de artefactos independientes de un sistema operativo o de plataformas específicas		0%	NI

Acrónimos: TAFI: Artefactos independientes, TAS: Artefactos del sistema o analizados, Medición: Aplicación de la métrica.

De manera similar, los participantes revisaron cuantas historias de usuario requieren de una plataforma específica para ser gestionadas, evidenciando que todas las historias de usuario fueron documentadas en tablas de Microsoft Office Excel 2019, es decir; Totalmente Implementada (TI).

Tabla 6.33. Resultados de aplicación de la métrica DCE.

Resultados de la métrica DCE: Documentación que requiere configuraciones especiales			
Pregunta	Valor	Medición	Escala
TAD: ¿Cuántos artefactos requieren de una configuración especial o dependen de un sistema operativo en particular?	5	100%	TI
TAS: ¿Cuántos artefactos componen el sistema o han sido analizados?	5		
Porcentaje de artefactos dependientes de una configuración especial o de un sistema operativo		100%	TI

Acrónimos: TAD: Artefactos dependientes, TAS: Artefactos del sistema o analizados, Medición: Aplicación de la métrica.

6.3.1.2.12. Medición de la característica de integridad

Como se describió en la sección 5.1.4.4.12, página 79 del presente documento, esta característica asocia las métricas: PID (Percepción de Integridad de la Documentación) y PAAS (Porcentaje de artefactos con un archivo de seguimiento). Para la métrica PID, los participantes evaluaron las historias de usuario respondiendo la encuesta asociada donde las respuestas de cada pregunta tienen una ponderación de 1 a 5 como se explica en la Tabla 5.32. De esta forma, cada participante aplicó la encuesta a las historias de usuario seleccionadas registrando las ponderaciones en las columnas P54 a P57 en la Tabla 6.34, y el resultado de la medición en la columna *medición*. De esta forma, los participantes perciben que las historias de usuario analizadas tienen un nivel de integridad del **32%**, es decir; Con Implementación Básica (IB). Para la métrica PAAS, se les preguntó a los participantes cuantas de las historias de usuario analizadas tiene asociado un archivo de seguimiento de los cambios que ha sufrido, registrando los valores en la Tabla 6.35. De esta manera, se evidenció que

ninguna de las historias de usuario analizadas cuenta con un archivo que registre un historial de cambios que se hayan realizado, es decir; No Implementada (NI).

Tabla 6.34. Resultados de aplicación de la métrica PID.

Resultados de la métrica PID: Percepción de integridad de la documentación							
Participante	H.U	P54	P55	P56	P57	Medición	Escala
PEC1	HU-009	1	3	2	1	35%	IB
PEC2	HU-004	1	1	1	1	20%	IB
PEC3	HU-006	1	3	2	1	35%	IB
PEC4	HU-005	1	2	1	1	25%	IB
PEC5	HU-008	2	2	3	2	45%	MI
Nivel de integridad percibido						32%	IB

Acrónimos: H.U: Historia de Usuario evaluada, P: Pregunta de la encuesta, Medición: Aplicación de la métrica.

Tabla 6.35. Resultados de aplicación de la métrica PAAS.

Resultados de la métrica PAAS: Porcentaje de artefactos con un archivo de seguimiento			
Pregunta	Valor	Medición	Escala
TArS: ¿Cuántos archivos de seguimiento existen?	0	0%	NI
TAS: ¿Cuántos artefactos componen el sistema o han sido analizados?	5		
Porcentaje de artefactos sobre los que se lleva un registro de cambios		0%	NI

Acrónimos: TArS: Archivos de seguimiento, TAS: Artefactos del sistema o analizados, Medición: Aplicación de la métrica.

6.3.1.2.13. Medición de la característica de actualidad

Como se describió en la sección 5.1.4.4.13, página 81 del presente documento, esta característica asocia las métricas: PACD (Percepción de Actualidad de la Documentación), PARev (Porcentaje de Artefactos Revisados) y PIRU (Problemas de Inconsistencia Reportados por los Usuarios). Para la métrica PACD, los participantes evaluaron las historias de usuario respondiendo la encuesta asociada donde las respuestas de cada pregunta tienen una ponderación de 1 a 5 como se explica en la Tabla 5.34. Así, cada participante aplicó la encuesta a las historias de usuario seleccionadas registrando las ponderaciones en las columnas P58 a P61 en la Tabla 6.36, y el resultado de la medición en la columna *medición*. De esta forma, los participantes perciben que las historias de usuario analizadas tienen un nivel de actualidad del **52%**, es decir; Medianamente Implementada (MI).

Tabla 6.36. Resultados de aplicación de la métrica PACD.

Resultados de la métrica PACD: Percepción de Actualidad de la Documentación							
Participante	H.U	P58	P59	P60	P61	Medición	Escala
PEC1	HU-009	4	5	2	1	60%	AI
PEC2	HU-004	1	5	3	1	50%	MI
PEC3	HU-006	2	5	2	1	50%	MI
PEC4	HU-005	1	5	3	1	50%	MI
PEC5	HU-008	2	3	3	2	50%	MI
Nivel de actualidad percibido						52%	MI

Acrónimos: H.U: Historia de Usuario evaluada, P: Pregunta de la encuesta, Medición: Aplicación de la métrica.

Para la métrica PARev, se les preguntó a los participantes cuantas de las historias de usuario analizadas corresponden realmente a la versión más actual del sistema que describen. De esta manera, se evidenció que ninguna historia de usuario analizada corresponde a la última versión del sistema que describe, es decir; No implementada (NI).

Tabla 6.37. Resultados de aplicación de la métrica PARev.

Resultados de la métrica PARev: Porcentaje de artefactos revisados			
Pregunta	Valor	Medición	Escala
TAC: ¿Cuántos artefactos coinciden con la versión del sistema?	0	0%	NI
TAS: ¿Cuántos artefactos componen el sistema o han sido analizados?	5		
Porcentaje de artefactos que coinciden con la versión actual del sistema		0%	NI

Acrónimos: TAC: Archivos que coinciden con la versión del sistema, TAS: Artefactos del sistema o analizados, Medición: Aplicación de la métrica.

Para la métrica PIRU, los participantes respondieron dos preguntas sobre los problemas reportados por los usuarios relacionados con la documentación de cada historia de usuario con respecto a todos los problemas reportados con el sistema. De esta forma, se evidenció que el **27.34%** de los problemas reportados por los usuarios están relacionados con la documentación de las historias de usuario, es decir; Con Implementación Básica (IB).

Tabla 6.38. Resultados de aplicación de la métrica PIRU.

Resultados de la métrica PIRU: Problemas de inconsistencia reportados por los usuarios					
Participante	H.U	PIR: ¿Cuántos problemas relacionados con la documentación se han reportado?	TPR: ¿Cuántos problemas en general han sido reportados sobre la funcionalidad?	Medición	Escala
PEC1	HU-009	2	5	40%	MI
PEC2	HU-004	3	10	30%	IB
PEC3	HU-006	3	10	30%	IB
PEC4	HU-005	1	15	6.7%	NI
PEC5	HU-008	3	10	30%	IB
Promedio de problemas de inconsistencia en la documentación reportados por los usuarios				27.34%	IB

Acrónimos: **H.U:** Historia de Usuario evaluada, **TAE:** Artefactos analizados, **TAM:** Artefactos con marcas de confidencialidad, **Medición:** Aplicación de la métrica.

6.3.1.2.14. Medición de la característica de trazabilidad

Como se describió en la sección 5.1.4.4.14, página 82 del presente documento, esta característica asocia las métricas: CDT (Cobertura de Trazabilidad), NDD (Nivel de Dependencia entre los Documentos) y NAA (Nivel de Asociación entre artefactos). Para la métrica CDT, los participantes determinaron que no utilizan una matriz de trazabilidad o similares, por lo tanto, la totalidad de los requisitos documentados en las historias de usuario analizadas no se encuentran relacionados entre sí. De esta forma, a pesar de que esta métrica tenga un 100% no es un valor positivo, debido a que esto indica la cantidad de documentación que no puede ser rastreada. Por lo tanto, se puede usar la categoría *No implementada* para representar el nivel de implementación de la característica.

Tabla 6.39. Resultados de aplicación de la métrica CDT.

Resultados de la métrica CDT: Cobertura de trazabilidad		
Pregunta	Valor	Medición
RSR: ¿Cuántos requisitos no se relacionan con otros en la matriz de trazabilidad o similares?	5	100%
NTR: ¿Cuántos requisitos componen el sistema o han sido analizados?	5	
Porcentaje de requisitos que no pueden ser rastreados		100%

Acrónimos: **RSR:** Requisitos relacionados en la matriz, **NTR:** Requisitos del sistema, **Medición:** Aplicación de la métrica.

Por otro lado, para la métrica NDD se utiliza el grafo construido para la métrica IAD presentado en la Tabla 6.12. Con esta información, se evidencia que todos los artefactos manejan el mismo nivel de dependencia ya que el resultado de la métrica es **0**.

Tabla 6.40. Resultados de aplicación de la métrica NDD.

Resultados métrica NDD: Nivel de dependencia entre los documentos			
Pregunta	Valor	Medición	Escala
GS: ¿Cuál es la suma de los grados de salida del grafo?	0	0	NI
PAS: ¿Cuál es la suma de los pesos de las aristas de salida?	0		
GE: ¿Cuál es la suma de los grados de entrada del grafo?	0		
PAE: ¿Cuál es la suma de los pesos de las aristas de entrada?	0		
N: ¿Cuántos nodos tiene el grafo?	5		
Nivel de dependencia: todos los documentos manejan el mismo nivel de dependencia			NI

Acrónimos: **H.U:** Historia de Usuario evaluada, **Medición:** Aplicación de la métrica.

Finalmente, la métrica NAA también utiliza el grafo construido para la métrica IAD presentado en la Tabla 6.12. Con esta información, se puede evidenciar que las historias de usuario analizadas no presentan ningún tipo de asociación entre sí, es decir; No implementada (NI).

Tabla 6.41. Resultados de aplicación de la métrica NAA.

Resultados métrica NAA: Nivel de asociación entre los documentos			
Pregunta	Valor	Medición	Escala
GS: ¿Cuál es la suma de los grados de salida del grafo?	0	0%	NI
GE: ¿Cuál es la suma de los grados de entrada del grafo	0		
n: ¿Cuántos nodos tiene el grafo?	5		
El nivel de asociación entre los documentos es del 0%			NI

Acrónimos: H.U: Historia de Usuario evaluada, **Medición:** Aplicación de la métrica.

6.3.1.2.15. Medición de la característica de expansibilidad

Como se describió en la sección 5.1.4.4.15, página 84 del presente documento, esta característica asocia las métricas: PED (Percepción de Expansibilidad de la Documentación) y PUP (Porcentaje de Uso de Plantillas). Para la métrica PED, los participantes evaluaron las historias de usuario respondiendo la encuesta asociada donde las respuestas de cada pregunta tienen una ponderación de 1 a 5 como se explica en la Tabla 5.40. De esta forma, cada participante aplicó la encuesta a las historias de usuario seleccionadas registrando las ponderaciones en las columnas *P16* a *P20* y *P62* en la Tabla 6.42, y el resultado de la medición en la columna *medición*. De esta forma, los participantes perciben que las historias de usuario analizadas tienen una capacidad de expansión del **72.5%**, es decir; Altamente Implementada (AI).

Tabla 6.42. Resultados de aplicación de la métrica PED.

Resultados de la métrica PED: Percepción de expansibilidad de la documentación									
Participante	H.U	P16	P17	P18	P19	P20	P62	Medición	Escala
PEC1	HU-009	4	3	4	4	5	4	80%	TI
PEC2	HU-004	5	4	3	4	4	4	80%	TI
PEC3	HU-006	2	3	3	3	3	3	56.66%	MI
PEC4	HU-005	2	4	4	4	3	3	70%	AI
PEC5	HU-008	4	4	4	3	4	4	76%	AI
Nivel de expansión percibido								72.5%	AI

Acrónimos: H.U: Historia de Usuario evaluada, P: Pregunta de la encuesta, **Medición:** Aplicación de la métrica.

Para la métrica PUP, los participantes utilizaron el valor obtenido en la métrica UPD presentados en la Tabla 6.9 para cada historia de usuario analizada. De esta forma se puede apreciar como las historias de usuario analizadas han utilizado un **76.4%** de la plantilla definida por la organización para documentarlas, es decir; Altamente Implementada (AI).

Tabla 6.43. Resultados de aplicación de la métrica PUP.

Resultados de la métrica PUP: Porcentaje de uso de plantillas					
Participante	H.U	UPD: Utilizar el valor de la métrica UPD de la característica de uniformidad	n: ¿Cuántos artefactos componen el sistema o fueron analizados?	Medición	Escala
PEC1	HU-009	50	1	76.4	AI
PEC2	HU-004	83	1		
PEC3	HU-006	83	1		
PEC4	HU-005	83	1		
PEC5	HU-008	83	1		
Promedio de uso de las plantillas de documentación en la empresa				76.4%	AI

Acrónimos: H.U: Historia de Usuario evaluada, UPD: Valor de la métrica UPD, **Medición:** Aplicación de la métrica.

6.3.1.2.16. Medición de la característica de validez

Como se describió en la sección 5.1.4.4.16, página 85 del presente documento, esta característica asocia las métricas: PAE (Porcentaje de artefactos aprobados por la empresa) y ARA (Artefactos Revisados y Aprobados). Para la métrica PAE, se les preguntó a los participantes cuantas de las historias analizadas han sido aprobadas por la organización,

dando como resultado que el **100%** las historias fueron aprobadas, es decir; Totalmente Implementada (TI).

Tabla 6.44. Resultados de aplicación de la métrica PAE.

Resultados de la métrica PAE: Porcentaje de artefactos aprobados por la empresa			
Pregunta	Valor	Medición	Escala
TADp: ¿Cuántos artefactos han sido aprobados por la organización?	5	100%	TI
TAS: ¿Cuántos artefactos componen el sistema o han sido analizados?	5		
Porcentaje de la documentación que ha sido aprobada por la empresa.		100%	TI

Acrónimos: **TADp**: Documentos aprobados, **TAS**: Documentos analizados, **Medición**: Aplicación de la métrica.

Como se describe en la Tabla 5.43, la métrica ARA contempla el uso de un archivo donde se registre el historial de revisiones y aprobaciones de las historias de usuario que se analizaron. Asimismo, para que se registre de manera completa dicho historial, la métrica sugiere que el archivo esté compuesto por siete secciones que contienen toda la información necesaria para el seguimiento y el control de las diferentes revisiones y aprobaciones por las que pasa una historia de usuario antes de implementación. En ese sentido, cada participante verificó cuantas de las secciones propuestas por la métrica coincidían con el historial de revisiones que maneja la empresa para cada historia de usuario analizada registrando los valores en la Tabla 6.45. De esta manera, se evidenció que todas las historias de usuario analizadas en el estudio de caso coincidían solo con dos secciones: la fecha de revisión y el nombre del revisor. Por lo tanto, se aprecia como el historial de revisiones está completo solo en un **28.57%**, es decir; Con Implementación Básica (IB).

Tabla 6.45. Resultados de aplicación de la métrica ARA.

Resultados de la métrica ARA: Porcentaje de artefactos revisados					
Participante	H.U	SNC: ¿Cuántas secciones del historial de revisiones coinciden con las secciones propuestas en la métrica?	n: ¿Cuántos artefactos componen el sistema o fueron analizados?	Medición	Escala
PEC1	HU-009	2	1	28.57%	IB
PEC2	HU-004	2	1		
PEC3	HU-006	2	1		
PEC4	HU-005	2	1		
PEC5	HU-008	2	1		
Promedio de completitud del historial de revisiones con base en la propuesta de la métrica				28.57%	IB

Acrónimos: **H.U**: Historia de Usuario evaluada, **SNC**: Secciones que coinciden con la métrica, **Medición**: Aplicación de la métrica.

6.3.1.2.17. Medición de la característica de verificabilidad

Como se describió en la sección 5.1.4.4.17, página 86 del presente documento, esta característica asocia la métrica DHR (Documentos con un Historial de Revisiones), a través de la cual los participantes verificaron cuantas de las historias de usuario analizadas tenían asociado un documento formal de revisiones. De esta manera se evidenció que ninguna de las historias de usuario analizadas contaba con este documento.

Tabla 6.46. Resultados de aplicación de la métrica DHR.

Resultados de la métrica DHR: Porcentaje de documentos un historial de comentarios o revisiones		
Pregunta	Valor	Medición
DAH: ¿Cuántos artefactos tienen asociado un documento de historial de revisiones?	0	0%
n: ¿Cuántos artefactos componen el sistema o han sido analizados?	5	
Porcentaje de la documentación con un historial de revisiones.		0%

Acrónimos: **DAH**: Documentos con historial de revisiones, **Medición**: Aplicación de la métrica.

6.3. Fase 4: Análisis de resultados

En esta fase, se realiza el análisis de toda la información recolectada en el estudio de caso. A continuación, se presentan los detalles de cada actividad en esta fase:

6.4.1. Análisis cualitativo

Utilizando los resultados presentados anteriormente en la Tabla 6.7, en la página 93, se puede determinar que la documentación estudiada ha logrado una calificación de **46.1%** luego de aplicar el modelo de métricas, es decir; Medianamente Implementada (MI), lo cual indica que aun presenta un nivel de deuda del **53.9%**.

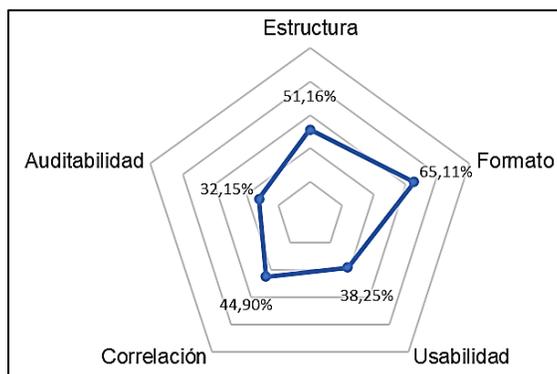


Figura 6.3. Valoración de cada dimensión de la documentación.

En la Figura 6.3 se resume gráficamente la calificación alcanzada por cada dimensión, donde la mejor calificación se logró en la dimensión de *Formato* con 65.11%, es decir; Altamente Implementado (AI). Como lo muestra la Tabla 6.47 en la página 93, esto se debe a que la documentación estudiada tiene un alto nivel de claridad (80%), pero tiene niveles medios de legibilidad (52.7%) y concisión (62.65%). Esto se vio reflejado en aspectos como: falta de comprensión en el propósito de la documentación y su público objetivo y falta de elementos de apoyo como ejemplos o imágenes para explicar detalles del contexto.

La dimensión de *Estructura* logró una calificación del 51.16%, es decir; Medianamente Implementado (MI), debido, principalmente, a la falta de modularidad de la documentación ya que no fue posible identificar como están relacionados los artefactos que componen la documentación del sistema. Esto afectaría gravemente la comprensión del sistema y dificultaría entender cómo puede impactar un cambio en futuro a las funcionalidades que ya se encuentren desarrolladas. Asimismo, al momento de realizar el estudio de caso, fue posible evidenciar que la empresa depende en gran medida de la experiencia de su equipo de desarrollo para entender cómo se relacionan las diferentes funcionales del sistema, pero esto no se puede evidenciar en la documentación.

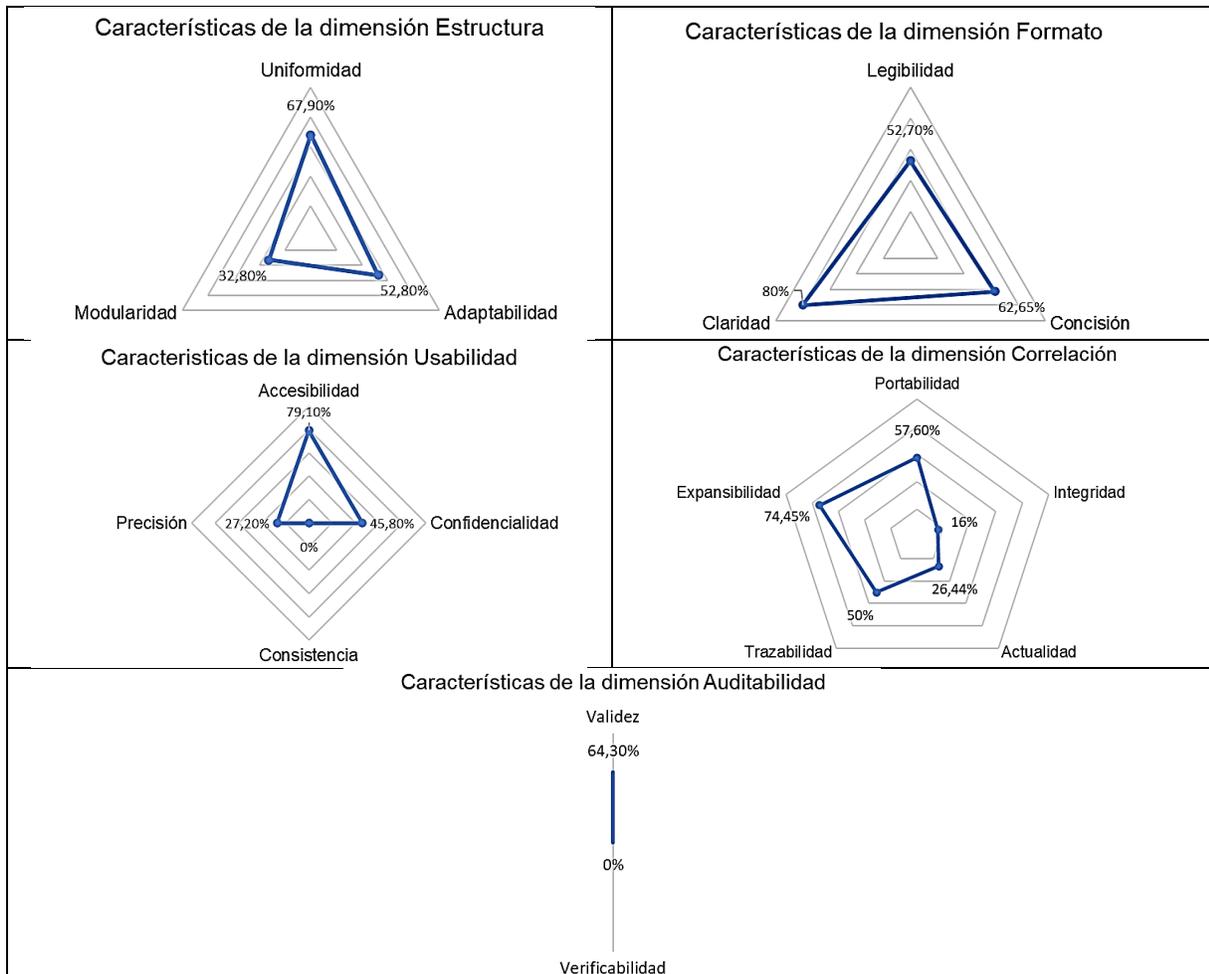
La dimensión de *Correlación* logró una calificación de 44.90%, es decir; Medianamente Implementado (MI), debido a que existe un desfase muy grave entre las funcionalidades reales del sistema y lo que se describe en la documentación, por lo tanto, la documentación presenta bajos niveles de integridad con respecto al sistema que describe, al igual que un alto nivel de desactualización. Adicionalmente, debido a los problemas de modularidad en la documentación, no es posible realizar una trazabilidad adecuada sobre las funcionalidades del sistema, con lo cual, se incrementa la dependencia de los miembros del equipo que mejor conozcan el sistema.

La dimensión de *Usabilidad* logró una calificación de 38.25%, es decir; Con Implementación Básica (IB), como consecuencia de la falta de modularidad, trazabilidad e integridad de la documentación analizada, lo que llevó a una falta total de consistencia. Adicionalmente, la documentación presentó un alto nivel de discrepancia entre su contenido y lo que realmente hace el sistema, lo cual es consecuente con los problemas de correlación. Por otro lado, fue posible evidenciar la falta de instrucciones precisas que permitan resolver dudas o problemas que se puedan presentar al momento de usar la documentación, lo cual lleva a otro inconveniente relacionado con la falta de ejemplos o casos prácticos para complementar el contenido demasiado especializado o complejo de entender.

La dimensión de *Auditabilidad* logró la calificación más baja con 32.15%, es decir; Con Implementación Básica (IB), debido a que la documentación estudiada no cuenta con registros o información adicional que evidencie un esfuerzo de revisión y aprobación que pudiera ser verificado por agentes externos o internos a la empresa en procesos de auditoria o similares. De manera indirecta, esta baja calificación es producto de la falta de integridad y la alta desactualización evidenciadas en la documentación.

Otros detalles de la calificación lograda en las dimensiones y sus características se pueden consultar en la Tabla 6.47, donde se presenta un resumen gráfico de dicha resultados.

Tabla 6.47. Resumen gráfico de los resultados de la medición por dimensión.



Para complementar el análisis, en la Tabla 6.48, se presenta un consolidado del nivel de implementación de cada característica evidenciado en la documentación utilizada en el estudio de caso. Se puede observar que solo la característica de *claridad* se encuentra *Totalmente implementada* con base en la escala de implementación descrita en la Tabla 6.6. Por otra parte, las características de *Uniformidad*, *Concisión*, *Accesibilidad*, *Expansibilidad* y *Validez* se encuentran *Altamente implementadas*. Las características de *Adaptabilidad*, *Legibilidad*, *Confidencialidad*, *Portabilidad* y *Trazabilidad* se encuentran *Medianamente implementadas*. Las características de *Modularidad*, *Precisión*, *Integridad* y *Actualidad* se encuentran *Implementadas de manera básica*. Y finalmente, las características de *Consistencia* y *Verificabilidad* no se encuentran implementadas en la documentación de la empresa.

Tabla 6.48. Clasificación de los resultados por característica.

No.	Dimensión	Característica	Promedio por característica	Clasificación	Acrónimo
1	Estructura	Uniformidad	67.9%	Altamente implementada	AI
		Adaptabilidad	52.8%	Medianamente implementada	MI
		Modularidad	32.8%	Con implementación básica	IB
2	Formato	Legibilidad	52.7%	Medianamente implementada	MI
		Concisión	62.65%	Altamente implementada	AI
		Claridad	80%	Totalmente implementada	TI
3	Usabilidad	Accesibilidad	79.1%	Altamente implementada	AI
		Confidencialidad	45.8%	Medianamente implementada	MI
		Consistencia	0%	No implementada	NI
		Precisión	27.2%	Con implementación básica	IB
4	Correlación	Portabilidad	57.6%	Medianamente implementada	MI
		Integridad	16%	Con implementación básica	IB
		Actualidad	26.44%	Con implementación básica	IB
		Trazabilidad	50%	Medianamente implementada	MI
		Expansibilidad	74.45%	Altamente implementada	AI
5	Auditabilidad	Validez	64.3%	Altamente implementada	AI
		Verificabilidad	0%	No implementada	NI

Al final del estudio de caso se realizó una breve discusión entre los participantes y el grupo investigador, donde fue posible identificar información adicional e importante para el estudio. A continuación, se listan las conclusiones más importantes de la discusión:

- El equipo de desarrollo realiza absolutamente todo, documentar, codificar, probar, corregir, entre otros, entonces se encuentran sobrecargados y con el tiempo muy reducido para realizar tareas de documentación y desarrollo.
- Se requiere de un equipo de personas dedicado exclusivamente a crear y actualizar la documentación de un sistema para liberar la sobrecarga que sufren los desarrolladores al realizar tareas diferentes a las de codificación o similares.
- Debido a los tiempos tan cortos que se establecen en cada proyecto por utilizar metodologías ágiles, resulta muy difícil documentar, y se opta por no documentar o hacerlo luego de que finalice la construcción del proyecto.
- Es muy importante mejorar la forma en cómo se realizan reuniones de seguimiento, ya que demoran tres horas aproximadamente, aunque se realicen semanalmente.
- Es importante que las personas que asuman la tarea de documentar el software, reciban capacitaciones regularmente por parte de la empresa, especialmente sobre: técnicas o tecnologías para ese propósito, esto con el objetivo de mejorar la documentación constantemente.
- Documentar no es difícil, siempre y cuando el desarrollo de software siga un cronograma claro para que la documentación se integre de manera satisfactoria.
- La empresa desea utilizar metodologías ágiles, pero no lo hace bien, ya que no se respeta la filosofía ágil, y menos al momento de documentar. Esto se ve reflejado en que el equipo

no ha recibido la capacitación necesaria al respecto y se delega la responsabilidad a quien de forma autónoma o empírica tiene más conocimientos.

- La documentación es crucial para el desarrollo de sistemas software, en especial para sistemas que son muy grandes, porque facilita la comprensión de las funcionalidades y ayuda a que la integración de nuevo personal sea más productiva.
- Cuando un sistema software no tiene documentación o es deficiente, los equipos de desarrollo se enfrentan al gran desafío de entender cómo funciona, para lo cual se ven obligados a experimentar y probar hipótesis de funcionamiento.
- La empresa maneja un portafolio bastante grande de proyectos que serán gestionados a través de metodologías ágiles, sin embargo, la mala adopción y uso de este tipo de metodologías lleva a la empresa y a los equipos de desarrollo a una cultura de la inmediatez, ignorando el futuro de esos sistemas en su evolución o mantenimiento.

Finalmente, tomando como referencia la información de la Tabla 5.2 de los 15 riesgos identificados para la documentación, se presenta en la Tabla 6.49 el análisis de la probabilidad de ocurrencia de cada uno de estos riesgos con base en el cálculo del promedio de la calificación obtenida por las características que tienen asociadas.

Tabla 6.49. Probabilidad de ocurrencia de los riesgos para la documentación de la empresa.

Riesgo	Características de la documentación asociadas	Probabilidad
R1	Uniformidad, Claridad, Legibilidad, Integridad, Precisión, Modularidad	53.90%
R2	Legibilidad, Conciso, Claridad, Consistencia, Verificabilidad	60.93%
R3	Uniformidad, Claridad, Consistencia, Actualidad, Trazabilidad, Validez, Verificabilidad	58.76%
R4	Legibilidad, Conciso, Claridad, Accesibilidad, Confidencialidad, Verificabilidad	46.62%
R5	Legibilidad, Conciso, Claridad, Accesibilidad, Confidencialidad, Integridad, Validez, Verificabilidad	49.93%
R6	Uniformidad, Claridad, Accesibilidad, Confidencialidad, Consistencia, Portabilidad	44.93%
R7	Uniformidad, Claridad, Accesibilidad, Confidencialidad, Precisión, Portabilidad, Verificabilidad	48.91%
R8	Claridad, Accesibilidad, Confidencialidad, Consistencia	48.77%
R9	Accesibilidad, Confidencialidad, Consistencia, Integridad, Verificabilidad	71.82%
R10	Modularidad, Accesibilidad, Confidencialidad, Integridad, Verificabilidad	65.26%
R11	Uniformidad, Claridad, Accesibilidad, Confidencialidad, Consistencia, Precisión, Verificabilidad	57.14%
R12	Legibilidad, Conciso, Accesibilidad, Confidencialidad, Consistencia	51.95%
R13	Uniformidad, Modularidad, Legibilidad, Conciso, Accesibilidad, Confidencialidad, Precisión, Integridad, Actualidad, Trazabilidad, Validez, Verificabilidad	56.25%
R14	Modularidad, Accesibilidad, Confidencialidad, Consistencia, Precisión, Actualidad, Trazabilidad, Validez, Verificabilidad	63.81%
R15	Uniformidad, Modularidad, Legibilidad, Conciso, Claridad, Accesibilidad, Consistencia, Precisión	49.70%

De esta forma, se puede apreciar cómo la mayoría de riesgos tienen una probabilidad de ocurrencia cercana al 50%, lo que sugiere una incertidumbre equitativa sobre si el riesgo se puede materializar. Sin embargo, aquellos riesgos que superan una probabilidad del 50% sugieren que en algún momento pueden ocurrir. En ambos casos, es necesario tomar medidas proactivas sobre la documentación que crea la empresa para gestionar los riesgos, y de esta manera mitigar sus posibles impactos, donde evaluaciones con modelos como el presentado en este documento puede ayudar a esos propósitos.

6.4.2. Análisis de la encuesta

Al final del estudio de caso, los participantes respondieron de manera anónima la encuesta de evaluación del estudio con el propósito recopilar información valiosa relacionada con las perspectivas de cada participante luego de utilizar la propuesta para medir la deuda de la documentación en el desarrollo de software ágil. Específicamente, la encuesta se dividió en seis secciones: (i) se consigna información sobre la edad, profesión, cargo en la organización y años de experiencia; (ii) se evalúa la claridad en la descripción de los componentes del modelo (dimensiones, características, riesgos y métricas); (iii) se evalúa la aplicabilidad del

modelo y su capacidad para identificar la deuda de la documentación; (iv) se evalúa la idoneidad de las características de la documentación que propone el modelo; (v) se evalúa la completitud del modelo para capturar y medir todos los aspectos relevantes de la deuda de la documentación de software; y (vi) preguntas abiertas para conocer otros aspectos que pueden mejorar el modelo. En la Tabla 6.50 se presenta la encuesta del estudio de caso.

Tabla 6.50. Cuestionario de evaluación del estudio de caso con escala Likert.

No.	Aspecto a evaluar	Id	Pregunta	Respuestas				
				TA	DA	I	ED	TD
1	Claridad	P1	¿Considera que las dimensiones de la documentación desarrolladas en el modelo propuesto son claras?	1	4	0	0	0
		P2	¿Considera que las características de la documentación desarrolladas en el modelo propuesto son claras?	1	4	0	0	0
		P3	¿Considera que los riesgos que pueden ocasionar deuda en la documentación de software desarrollados en el modelo propuesto son claros?	1	4	0	0	0
		P4	¿Considera que las métricas definidas en el modelo propuesto son claras?	1	4	0	0	0
2	Aplicabilidad	P5	¿Considera que las métricas definidas en el modelo propuesto tienen suficiente rigor matemático?	1	2	2	0	0
		P6	De acuerdo con su experiencia: ¿Considera que la forma como se evalúan las características de la documentación de software es adecuada y permite identificar aspectos de la deuda de la documentación?	2	3	0	0	0
3	Idoneidad	P7	¿Considera que las métricas propuestas para medir la característica de uniformidad de la documentación son adecuadas?	3	1	1	0	0
		P8	¿Considera que las métricas propuestas para medir la característica de adaptabilidad de la documentación son adecuadas?	1	3	1	0	0
		P9	¿Considera que las métricas propuestas para medir la característica de modularidad de la documentación son adecuadas?	1	4	0	0	0
		P10	¿Considera que las métricas propuestas para medir la característica de legibilidad de la documentación son adecuadas?	2	3	0	0	0
		P11	¿Considera que las métricas propuestas para medir la característica de concisión de la documentación son adecuadas?	3	1	1	0	0
		P12	¿Considera que las métricas propuestas para medir la característica de claridad de la documentación son adecuadas?	3	2	0	0	0
		P13	¿Considera que las métricas propuestas para medir la característica de accesibilidad de la documentación son adecuadas?	2	3	0	0	0
		P14	¿Considera que las métricas propuestas para medir la característica de confidencialidad de la documentación son adecuadas?	2	2	1	0	0
		P15	¿Considera que las métricas propuestas para medir la característica de consistencia de la documentación son adecuadas?	2	2	1	0	0
		P16	¿Considera que las métricas propuestas para medir la característica de precisión de la documentación son adecuadas?	2	2	1	0	0
		P17	¿Considera que las métricas propuestas para medir la característica de portabilidad de la documentación son adecuadas?	3	2	0	0	0
		P18	¿Considera que las métricas propuestas para medir la característica de integridad de la documentación son adecuadas?	3	2	0	0	0
		P19	¿Considera que las métricas propuestas para medir la característica de actualidad de la documentación son adecuadas?	3	1	1	0	0
P20	¿Considera que las métricas propuestas para medir la característica de trazabilidad de la documentación son adecuadas?	2	2	1	0	0		
P21	¿Considera que las métricas propuestas para medir la característica de expansibilidad de la documentación son adecuadas?	3	1	1	0	0		
P22	¿Considera que las métricas propuestas para medir la característica de validez de la documentación son adecuadas?	1	2	2	0	0		
P23	¿Considera que las métricas propuestas para medir la característica de verificabilidad de la documentación son adecuadas?	1	3	1	0	0		
4	Completitud	P24	¿Considera que las métricas propuestas en el modelo son suficientes para garantizar una evaluación completa de la deuda de la documentación de software?	3	2	0	0	0
TOTAL				47	59	14	0	0

Como se puede apreciar en las figuras 6.4 y 6.5, el modelo tuvo una muy buena aceptación por parte de los participantes del estudio de caso, pero aún es un modelo que requiere ajustes para mejorar su precisión, descripción y automatización. En ese sentido, la Tabla 6.51 presenta las respuestas de las preguntas abiertas para identificar otros aspectos que complementan la evaluación del modelo propuesto.

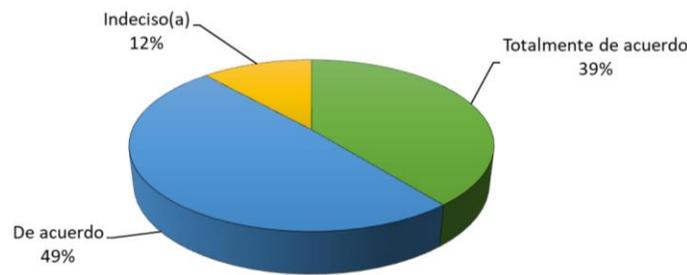


Figura 6.4. Distribución porcentual de las respuestas de la encuesta.

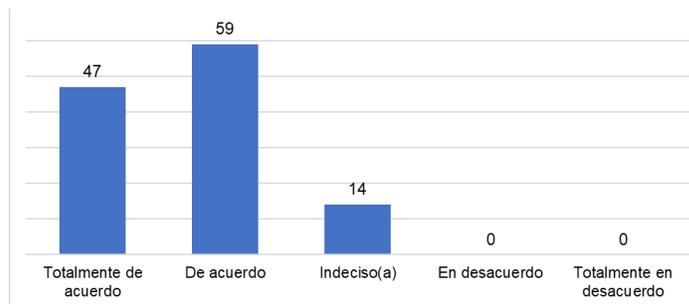


Figura 6.5. Distribución de las respuestas de la encuesta.

Tabla 6.51. Preguntas abiertas de la encuesta.

Id	Pregunta	Respuestas
P25	¿Cree que se ha(n) omitido alguna(s) dimensión(es) de la documentación de software que usted considere importante(s) incluir en el modelo?	<ul style="list-style-type: none"> Se podría evaluar que tan interactiva es la documentación.
P26	¿Cree que se ha(n) omitido alguna(s) característica(s) de la documentación de software que usted considere importante(s) incluir en el modelo?	<ul style="list-style-type: none"> No hay comentarios
P27	¿Considera que alguna de las dimensiones o características no es fundamental o deba ser trabajada de otra forma?	<ul style="list-style-type: none"> Considero que la característica de concisión es muy subjetiva y muy difícil de medir.
P28	De acuerdo con su experiencia ¿Considera que el modelo de métricas propuesto puede ser aplicado para medir la deuda de la documentación en proyectos de software ágil?	<ul style="list-style-type: none"> Considero que le falta más trabajo para cumplir con su cometido, pero es una muy buena iniciativa.
P29	¿Con base en los resultados obtenidos luego de aplicar el modelo de métricas, considera que le ha permitido identificar oportunidades de mejora asociados a la deuda de la documentación?	<ul style="list-style-type: none"> Si, definitivamente.
P30	¿Tienes sugerencias o comentarios que para mejorar o corregir el modelo presentado que no hayamos abordado en otras preguntas? En caso negativo solo copiar No	<ul style="list-style-type: none"> Pienso que para el futuro se puede pensar en automatizar el proceso de medición para que no sea tan manual. Brindar recomendaciones de tener un equipo de documentación encargado en las diferentes entidades, para así quitar carga a los desarrolladores

6.4.3. Análisis del esfuerzo y el costo

En la Tabla 6.52 se presenta el esfuerzo aplicado en el estudio de caso tomando como base las horas necesarias para ejecutarlo en la empresa. Para ello, se documentaron las horas empleadas por el equipo investigador y los recursos humanos de la empresa durante las etapas del estudio de caso: (i) Socialización de la propuesta, (ii) Planificación, (iii) Preparación previa, (iv) Aclaración de dudas, (v) Ejecución y (vi) Análisis de los resultados.

Tabla 6.52. Esfuerzo en horas aplicados.

No.	Etapas	Esfuerzo individual (horas)	Esfuerzo total (horas)
1	Socialización de la propuesta	0.5	2.5
2	Planificación	0.5	2.5
3	Preparación previa	2	10
4	Aclaración de dudas	0.5	2.5

5	Ejecución	3	15
6	Análisis de los resultados	6	6
Total		12.5	38.5

En la Tabla 6.52 se muestra el esfuerzo realizado por cada participante del estudio de caso, dando como resultado un esfuerzo individual de 12.5 horas y grupal de 38.5 horas. De acuerdo a la Ley 2101 de 2021 en Colombia, un mes de trabajo está compuesto por 188 horas, lo que representa 47 horas de trabajo a la semana y aproximadamente 7.8 horas laborales diarias. En ese sentido, cada participante aplicó un estimado de 12.5 horas de su jornada laboral durante el estudio de caso, siendo necesario que cada participante ajustara una parte de su horario para participar del estudio de caso en cada etapa, principalmente durante la ejecución.

Tomando como base el esfuerzo en horas, se realizó una conversión de los resultados obtenidos a costos por cada participante, de acuerdo con los valores promedios por hora en dólares americanos (USD) descritos en Pay Scale [80]. De esta forma, se establece un valor de 42.8 dólares por hora para el rol de Desarrollador, por lo tanto, el costo de aplicar el modelo de métricas es de 535 dólares por persona y 2675 dólares para un grupo de 5 personas. Ahora bien, teniendo en cuenta el salario promedio que un desarrollador de software puede ganar en Colombia, el costo de una hora de trabajo es de 19.231 (COP), por lo tanto, el uso del modelo de métricas es de \$240.387 (COP) por persona y de \$1'201.935 (COP). En ambos casos, representan un costo relativamente bajo en comparación con otros gastos que asumen las empresas en aspectos como: capacitaciones, certificaciones y auditorías internas.

6.4.4. Análisis de la validez

A continuación, se describe el análisis de la validez del constructo, la validez externa y de confiabilidad del modelo respecto a los resultados obtenidos en el estudio de caso.

6.4.4.1. Validez del constructo

Con el objetivo de conservar la validez del constructo y garantizar que los resultados obtenidos representen de manera objetiva aquello que se pretende evaluar en concordancia con las preguntas de investigación, se llevó a cabo una reunión inicial en la cual se presentó: el contexto del proyecto de investigación, la solución propuesta, el plan de trabajo para el estudio de caso y se solucionaron inquietudes o ambigüedades de los participantes. Además, se presentó a los participantes la documentación necesaria para entender el modelo, apoyar su evaluación y facilitar la resolución de dudas.

6.4.4.2. Validez externa

A través del estudio de caso realizado, es posible generalizar el modelo y aplicarlo en otros escenarios para probar su potencial en empresas de la misma industria que enfrentan desafíos similares en la gestión de la documentación de software. Además, puede aplicarse a proyectos de diferentes tamaños y complejidades para reforzar su validez externa al sugerir que no se limita a situaciones particulares. Por lo tanto, la propuesta se presenta como una herramienta potencialmente valiosa y generalizable para abordar la deuda en la documentación de software en diversos contextos industriales y organizativos.

6.4.4.3. Confiabilidad

Con el objetivo de reducir el sesgo y garantizar la independencia entre los resultados obtenidos y los investigadores que participaron en el estudio de caso, se realizaron las siguientes actividades: (i) definición de los criterios para la selección del sujeto de estudio, (ii) definición

del procedimiento de campo y de recolección de datos, (iii) elaboración de documentación detallada del modelo de métricas para ser usada en cada estudio de caso, y (iv) soporte permanente al evaluador y participantes del estudio de caso durante la fase de ejecución.

6.4.5. Análisis de las limitaciones

A continuación, se presentan las limitaciones identificadas en la ejecución del estudio de caso:

- El estudio de caso se realizó en una sola empresa desarrolladora de software, razón por la cual no es posible generalizar los resultados. Por lo tanto, es necesario replicar el estudio de caso en otras empresas software con características diferentes y con grupos interdisciplinarios con el objetivo de extender la naturaleza de los resultados.
- Existe cierto grado de sesgo asociado a los participantes luego de aplicar el instrumento de evaluación. Adicionalmente, la interpretación de los resultados está sujeta a un grado de subjetividad por parte del equipo investigador.
- Debido a que la etapa de ejecución tuvo una duración de tres horas, es posible que se haya generado un agotamiento progresivo que haya afectado el juicio crítico de los participantes frente a la evaluación de la propuesta.
- En el estudio de caso se evaluaron cinco historias de usuario para determinar su nivel de deuda, sin embargo, a pesar de que se obtuvo información muy valiosa, no es representativa de todas las historias de usuario ni de toda la documentación del proyecto. Con esto se puede concluir que el modelo de métricas no es específico para la documentación generada en entornos ágiles sino genérica.

Finalmente, se presenta una respuesta a las preguntas de investigación planteadas para el estudio de caso en la Tabla 6.2:

- *¿El modelo de métricas propuesto les permite a las empresas de software identificar el nivel de deuda en la documentación de sus proyectos de software ágiles?*

El modelo de métricas propuesto proporciona a las empresas de software una herramienta que les puede ayudar a identificar el nivel de deuda en la documentación de sus proyectos de software ágiles. Este enfoque no solo ofrece una visión integral de la calidad de la documentación, sino que también facilita la detección temprana de posibles áreas de mejora. Creemos que implementar este modelo permitirá a las empresas optimizar sus procesos, mejorar la transparencia y garantizar un desarrollo de software más eficiente y sostenible en el largo plazo. Estamos emocionados por los beneficios que esta metodología puede aportar a la gestión de proyectos ágiles y esperamos que sea de gran utilidad para la industria.

- *¿El modelo de métricas propuesto permite identificar oportunidades de mejora en la documentación de software de los proyectos ágiles?*

Creemos que el modelo de métricas propuesto ofrece una perspectiva valiosa para identificar oportunidades de mejora en la documentación de software en proyectos ágiles. Aunque reconocemos que ninguna metodología es perfecta, este enfoque ha demostrado ser efectivo al resaltar aspectos de la documentación donde se podrían realizar ajustes beneficiosos. Es fundamental reconocer que ninguna metodología es infalible y, en este sentido, el modelo propuesto no constituye una excepción. En cambio, se erige como una propuesta que, con humildad, busca ofrecer una visión constructiva para la mejora continua. En este sentido, se contempla la posibilidad de adaptar la metodología a las particularidades de cada proyecto, reconociendo la unicidad de los mismos y la necesidad de ajustes específicos.

- *¿El esfuerzo de aplicar el modelo de métricas propuesto es adecuado para medir la deuda de la documentación en proyectos de software ágiles?*

Creemos que el esfuerzo depende de cada empresa, sin embargo, consideramos que la aplicación del modelo de métricas propuesto se percibe como adecuada y razonable para la evaluación de la deuda en la documentación de proyectos de software ágil en relación con los beneficios potenciales que puede aportar. Es importante señalar que, aunque confiamos en la utilidad del modelo, reconocemos que aún requiere ajustes producto de la retroalimentación para llegar a ser una solución útil y adaptable a las necesidades de la industria.

Capítulo 7: Conclusiones y trabajos futuros

En este capítulo se presentan los aspectos más importantes de acuerdo con los resultados obtenidos después de realizar el trabajo de investigación. Inicialmente, se presenta un análisis del grado de cumplimiento de cada uno de los objetivos de investigación, luego se presentan las conclusiones y se finaliza con los trabajos futuros.

7.1. Análisis de la pregunta, hipótesis y objetivos de investigación

A continuación, se presenta en análisis de cómo se logró dar solución a los objetivos de investigación planteados y sus capítulos asociados.

7.1.1. Respuesta a la pregunta e hipótesis de investigación

La pregunta de investigación planteada fue: **¿Es posible reducir la deuda de la documentación generada durante el desarrollo ágil de software a través de la evaluación de la calidad de la documentación desde el punto de vista de la transparencia?**

A partir de esta pregunta se plantearon dos hipótesis:

- **H1:** Mediante la evaluación de la calidad de la documentación desde el punto de vista de la transparencia se reduce la deuda de la documentación en los proyectos de desarrollo ágil de software.
- **H0:** Mediante la evaluación de la calidad de la documentación desde el punto de vista de la transparencia NO se reduce la deuda de la documentación en los proyectos de desarrollo ágil de software.

De acuerdo a los resultados obtenidos con esta investigación, podemos afirmar que la reducción de la deuda de la documentación en el desarrollo ágil de software se presenta como una posibilidad tangible mediante la evaluación centrada en la calidad desde la óptica de la transparencia. Al adoptar este enfoque, se logra no solo identificar las áreas susceptibles de acumular deuda documental, sino también; mejorar la comprensión global de la documentación existente. De esta manera, la transparencia en la calidad de la documentación implica no solo la claridad en la presentación de información, sino también la coherencia, la actualización constante y la accesibilidad para todos los miembros del equipo. Este enfoque no solo facilitaría una colaboración más efectiva, sino que también nutre una cultura organizativa basada en la confianza y la toma de decisiones informada. Asimismo, al considerar atributos de transparencia en la documentación de software, se establece una base sólida para implementar medidas correctivas oportunas, mitigando así la acumulación de deuda y fortaleciendo la integridad y utilidad de la documentación en el desarrollo ágil de software.

7.1.2. Respuesta al Objetivo general

Diseñar un modelo de métricas para medir la deuda de la documentación que se genera durante la ejecución de proyectos de desarrollo ágil de software mediante la definición de mecanismos de evaluación que integren características de transparencia sobre los atributos de la documentación en este tipo de proyectos.

El objetivo general ha sido cumplido mediante el modelo descrito en el Capítulo 5 en el cual se formaliza el conjunto de aspectos a considerar para evaluar de manera cuantitativa el nivel de deuda en la documentación de software de los proyectos ágiles siguiendo el formalismo propuesto por el enfoque GQM.

7.1.3. Respuesta a los Objetivos específicos (OE)

OE 1. Establecer los atributos fundamentales a tenerse en cuenta para caracterizar la deuda de la documentación en proyectos de desarrollo ágil de software a partir de un mapeo sistemático de la literatura.

Este objetivo fue posible cumplirlo a través de tres productos. El primero, la realización de una revisión sistemática de la literatura para identificar las iniciativas científicas más relevantes sobre deuda de la documentación en el desarrollo ágil de software y así recopilar toda la información posible sobre definiciones, propuestas, validaciones, procesos y métodos de investigación en dicha temática. Además, fue posible recolectar información valiosa en esta temática como: definiciones conceptuales; causas, efectos, impactos y limitaciones; técnicas y métodos de evaluación, herramientas tecnológicas; procesos de validación en la industria; y metodologías de documentación. Por otra parte, fue posible observar que los hallazgos presentaban un alto nivel de ambigüedad y desorganización, por lo que surgió el segundo producto, el cual fue una caracterización, de donde surgieron entre otras cosas, un conjunto de 17 características o atributos de calidad de la documentación de software clasificados en 5 dimensiones de análisis. Finalmente, gracias a esta caracterización fue posible crear el tercer producto, el cual consistió en proponer una versión preliminar de una ontología de la documentación con la cual se intenta dar el primer paso hacia la organización del conocimiento en esta temática ya que no fue posible evidenciar algo similar. Todo lo mencionado se describe en los Capítulos 2, 3 y 4, respectivamente.

OE 2. Construir un modelo de métricas asociadas a un subconjunto de las características identificadas en el OE1 a partir del enfoque Goal-Question-Metric (GQM) para apoyar la medición de la deuda en la documentación.

Este objetivo fue posible cumplirlo a partir de la caracterización descrita en el Capítulo 3, la cual permitió la construcción de un modelo compuesto por 39 métricas relacionadas con las características de la documentación mediante el enfoque Goal-Question-Metric (GQM), el cual garantizó que las métricas fueran construidas siguiendo un proceso ordenado y estructurado. Adicionalmente, se definió un conjunto de 15 riesgos para la documentación de software a partir de las características del modelo. El proceso llevado a cabo y el modelo de métricas propuesto se presentan en el Capítulo 5.

OE 3. Evaluar el modelo propuesto a través de su aplicación en un estudio de caso en un proyecto de desarrollo ágil de software para medir el nivel de deuda en su documentación.

Este objetivo se cumplió mediante la aplicación del modelo de métricas en una empresa de software regional a través de un estudio de caso, el cual permitió determinar la efectividad del modelo en la medición del nivel de deuda en la documentación de software, además de su claridad, aplicabilidad, idoneidad y completitud. Gracias a la ejecución del estudio de caso, fue posible medir de manera cuantitativa, el nivel de deuda en la documentación de la empresa estudio de caso, indicar factores de riesgo que llevan a su existencia y comprender como este tipo de deuda puede influir en el corto, mediano y largo plazo sobre el desarrollo de sus proyectos. Esta información puede ser consultada en el Capítulo 6.

7.2. Conclusiones

A lo largo de esta investigación, fue posible explorar el fenómeno de la deuda de la documentación en el desarrollo ágil de software; iniciando con una revisión sistemática de la literatura, pasando por la caracterización de esta temática, luego la formulación de un modelo de métricas y finalizando con la aplicación del modelo en un estudio de caso. En ese sentido, la convergencia de todas estas etapas proporciona una visión integral de los desafíos y oportunidades que aún persisten en esta área tan crítica para el desarrollo de software. Por lo tanto, en esta sección se presentan las conclusiones más importantes de esta investigación.

- La revisión sistemática de la literatura sobre deuda de la documentación en el desarrollo ágil de software, permitió identificar una correlación sustancial entre las deficiencias de la documentación y la acumulación de deuda técnica en proyectos software, especialmente en proyectos ágiles. Esto indica firmemente que la falta de documentación clara y actualizada persiste como un desafío crítico para la mantenibilidad del software en contextos ágiles, llevando a los equipos a enfrentar dificultades para compaginar la agilidad con la creación y mantenimiento de la documentación. A causa de esto, es importante que las empresas de software sean conscientes que la falta de compromiso para crear documentación de calidad puede contribuir a la acumulación de deuda, subrayando la necesidad de una cultura organizativa que valore la documentación como un componente integral del ciclo de vida del software.
- La evidencia obtenida en esta investigación apunta a que la deuda de la documentación incide negativamente en la colaboración y comunicación dentro de los equipos ágiles. Por lo tanto, la carencia de documentación clara y accesible puede propiciar malentendidos entre los miembros del equipo y obstaculizar la transferencia efectiva de conocimiento. En consecuencia, es necesario que las empresas replanteen sus estrategias de documentación para alinearse de manera efectiva con los principios ágiles, lo cual representaría un paso significativo para reducir la deuda y ser altamente productivos en el desarrollo de sus productos software.
- Los resultados de esta investigación invitan a la reflexión sobre las consideraciones éticas intrínsecas a la creación, mantenimiento y utilización de la documentación en el desarrollo ágil de software. De esta forma, aspectos como la transparencia y la claridad se convierten en principios éticos fundamentales que afectan la toma de decisiones informada y la rendición de cuentas dentro de los equipos ágiles. Por lo tanto, la información documentada debe reflejar de manera precisa el estado del software que se está desarrollando, permitiendo a todos los miembros del equipo comprender las decisiones tomadas y asumir la responsabilidad colectiva por el progreso y los desafíos. Adicionalmente, la manipulación o tergiversación de la documentación no solo compromete la calidad del producto, sino que también plantea cuestiones éticas relativas a la confianza y la honestidad del equipo.

- El modelo de métricas propuesto demostró ser eficaz en la medición de la deuda de la documentación mediante indicadores específicos que permitieron una evaluación más detallada y cuantificable de la calidad de la documentación, ofreciendo una visión más clara de las áreas que requieren atención inmediata. Por lo tanto, es una propuesta que permite una evaluación granular y adaptativa del nivel de deuda en la documentación de software ya que tiene en cuenta atributos de calidad y métodos de medición objetivos, los cuales ayudan a establecer un diagnóstico beneficioso para el software a mediano y largo plazo.
- A pesar que el estudio de caso fue realizado siguiendo un protocolo y una estructura definida, este cuenta con varias limitantes: (i) los resultados obtenidos no se pueden generalizar debido a la naturaleza del estudio, por lo cual es necesario extender su alcance mediante la aplicación de estudios de caso adicionales en otras empresas con diferentes características, y (ii) la existencia de un sesgo tácito que resulta de los métodos llevados a cabo para la evaluación y el análisis de los resultados generados por el equipo investigador.
- Los resultados de esta investigación han resaltado la importancia de considerar la deuda en la documentación como algo significativo en el desarrollo ágil de software que requiere de mucha atención de parte de los equipos ágiles. Aunque estas metodologías promuevan la eficiencia y la adaptabilidad, también pueden conducir a una acumulación gradual de deuda en la documentación de un sistema software si no se gestiona adecuadamente. Por otro lado, este problema se puede agravar si se integran otros enfoques como DevOps debido a que tienden a aumentar aún más la velocidad con la cual se entrega valor al cliente y por consiguiente a exacerbar los problemas de insuficiencia y desactualización de la documentación. Sin embargo, también observamos que una administración correcta de este tipo de deuda puede mejorar la eficiencia del desarrollo de software y la calidad del producto final. Por lo tanto, es esencial que los equipos de desarrollo ágil adopten estrategias para gestionar adecuadamente este tipo de deuda según las necesidades de la empresa, especialmente si se integran con otros enfoques como DevOps.

7.3. Trabajos futuros

A continuación, se presentan las brechas que pueden ser abordadas como trabajo futuro en nuevas investigaciones:

- **Actualizar la revisión sistemática:** La revisión sistemática fue realizada hasta junio de 2023, sin embargo, al momento de la construcción de este documento pueden haber surgido otras investigaciones relevantes para esta temática. Por lo cual se sugiere actualizar la revisión sistemática para incluir estas nuevas propuestas incluyendo nuevas condiciones de búsqueda en la cadena que puedan especializar aún más los resultados.
- **Validar teóricamente la ontología de la documentación:** Uno de los resultados de esta investigación fue la creación de una versión preliminar de la ontología de la deuda de la documentación como una iniciativa para estructurar el conocimiento en esta temática. Sin embargo, es necesario revisar la propuesta para ajustarla con nuevos conceptos y relaciones para crear una nueva versión que pueda ser validada de manera teórica.
- **Validar la ontología con expertos:** Relacionado con el anterior trabajo futuro, una vez se haya construido una nueva versión de la ontología, se sugiere que sea sometida a evaluación por parte de expertos en documentación de software o similares, con el propósito que la ontología cumpla también con las expectativas de la industria.

- **Diseñar un proceso de documentación:** Con toda la información recolectada en esta investigación, se puede considerar la creación de un proceso de documentación que pueda ser integrado en los procesos de desarrollo de software en las empresas, en especial para aquellas que construyen software utilizando metodologías ágiles.
- **Proponer iniciativas para automatizar la documentación de software:** Dada la necesidad de las empresas por reducir la carga de sus equipos de desarrollo, se sugiere proponer técnicas que automaticen la creación de la documentación a medida que avance el desarrollo de software.
- **Ejecutar otros estudios de caso:** Se sugiere realizar otros estudios de caso en empresas software con diferentes características para perfeccionar la propuesta y mejorar su efectividad para medir el nivel de deuda de la documentación.

Bibliografía

- [1] E. Aghajani *et al.*, «Software Documentation Issues Unveiled», *Proceedings - International Conference on Software Engineering*, vol. May, pp. 1199-1210, 2019, doi: 10.1109/ICSE.2019.00122.
- [2] G. Maturro, F. Raschetti, y C. Fontán, «A systematic mapping study on soft skills in software engineering», *Journal of Universal Computer Science*, vol. 25, n.º 1, pp. 16-41, 2019, doi: 10.3217/jucs-025-01-0016.
- [3] Y. Shmerlin, I. Hadar, D. Kliger, y H. Makabee, «To Document or Not to Document? An Exploratory Study on Developers' Motivation to Document Code», *Lecture Notes in Business Information*, vol. 215, pp. 100-106, 2015, doi: 10.1007/978-3-319-19243-7_10.
- [4] N. Rios, R. O. Spínola, M. Mendonça, y C. Seaman, «The practitioners' point of view on the concept of technical debt and its causes and consequences: a design for a global family of industrial surveys and its first results from Brazil», *Empirical Software Engineering*, vol. 25, n.º 5, pp. 3216-3287, 2020, doi: 10.1007/s10664-020-09832-9.
- [5] P. W. McBurney *et al.*, «Towards prioritizing documentation effort», *IEEE Transactions on Software Engineering*, vol. 44, n.º 9, pp. 897-913, 2017, doi: 10.1109/TSE.2017.2716950.
- [6] S. Shylesh, «A Study of Software Development Life Cycle Process Models», *SSRN Electronic Journal*, 2017, doi: 10.2139/ssrn.2988291.
- [7] R. Shaydulin y J. Sybrandt, «To Agile, or not to Agile: A Comparison of Software Development Methodologies», abr. 2017, Accedido: abr. 06, 2023. [En línea]. Disponible en: <https://arxiv.org/abs/1704.07469v1>.
- [8] A. K. . Z. Islam y D. A. Ferworn, «A Comparison between Agile and Traditional Software Development Methodologies», *Global Journal of Computer Science and Technology*, jul. 15, 2020. <https://bit.ly/3KaIXIE> (accedido oct. 22, 2022).
- [9] S. Al-Saqqqa, S. Sawalha, y H. Abdelnabi, «Agile software development: Methodologies and trends», *International Journal of Interactive Mobile Technologies*, vol. 14, n.º 11, pp. 246-270, 2020, doi: 10.3991/ijim.v14i11.13269.
- [10] M. Fowler, «Technical Debt Quadrant», 2009. <https://bit.ly/2QIEvbl> (accedido oct. 17, 2022).
- [11] J. P. Zumba y C. A. L. Arreaga, «Evolución de las metodologías y modelos utilizados en el desarrollo de software.», *INNOVA Research Journal*, oct. 05, 2018. <https://bit.ly/3KGFNNn> (accedido abr. 07, 2023).
- [12] A. Martini, T. Besker, y J. Bosch, «Process debt: A first exploration», *Proceedings - Asia-Pacific Software Engineering Conference, APSEC*, pp. 316-325, 2020, doi: 10.1109/APSEC51365.2020.00040.
- [13] N. Rios *et al.*, «Hearing the Voice of Software Practitioners on Causes, Effects, and Practices to Deal with Documentation Debt», *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, n.º March, pp. 55-70, 2020, doi: 10.1007/978-3-030-44429-7_4.
- [14] W. N. Behutiye, P. Rodríguez, M. Oivo, y A. Tosun, «Analyzing the concept of technical debt in the context of agile software development: A systematic literature review», *Information and Software Technology*, vol. 82, pp. 139-158, 2017, doi:

- 10.1016/j.infsof.2016.10.004.
- [15] A. Martini, V. Stray, y N. B. Moe, «Technical-, Social- and Process Debt in Large-Scale Agile: An Exploratory Case-Study», en *Lecture Notes in Business Information Processing*, vol. 364, Springer International Publishing, 2019, pp. 112-119.
- [16] Z. Li, P. Avgeriou, y P. Liang, «A systematic mapping study on technical debt and its management», *Journal of Systems and Software*, vol. 101, pp. 193-220, 2015, doi: 10.1016/j.jss.2014.12.027.
- [17] N. S. R. Alves, T. S. Mendes, M. G. De Mendonça, R. O. Spinola, F. Shull, y C. Seaman, «Identification and management of technical debt: A systematic mapping study», *Information and Software Technology*, vol. 70, pp. 100-121, 2016, doi: 10.1016/j.infsof.2015.10.008.
- [18] M. Shafiq y U. sman Waheed, «Documentation in Agile Development A Comparative Analysis», en *2018 IEEE 21st International Multi-Topic Conference (INMIC)*, nov. 2018, pp. 1-8, doi: 10.1109/INMIC.2018.8595625.
- [19] S. Voigt, J. Von Garrel, J. Müller, y D. Wirth, «A Study of Documentation in Agile Software Projects», *International Symposium on Empirical Software Engineering and Measurement*, vol. 08-09-Sept, pp. 2-7, 2016, doi: 10.1145/2961111.2962616.
- [20] C. Cappelli, P. Engiel, R. M. de Araujo, y J. C. S. do P. Leite, «Managing Transparency Guided by a Maturity Model», *3rd Global Conference on Transparency Research HEC PARIS*, n.º Chung 2000, p. 17, 2013, [En línea]. Disponible en: <https://t.ly/M2BT>.
- [21] J. Holvitie *et al.*, «Technical debt and agile software development practices and processes: An industry practitioner survey», *Information and Software Technology*, vol. 96, pp. 141-160, 2018, doi: 10.1016/j.infsof.2017.11.015.
- [22] J. Bogner, R. Verdecchia, y I. Gerostathopoulos, «Characterizing Technical Debt and Antipatterns in AI-Based Systems: A Systematic Mapping Study», *Proceedings - 2021 IEEE/ACM International Conference on Technical Debt, TechDebt 2021*, 2021, [En línea]. Disponible en: <http://arxiv.org/abs/2103.09783>.
- [23] J. Y. Khan, M. Tawkat Islam Khondaker, G. Uddin, y A. Iqbal, «Automatic Detection of Five API Documentation Smells: Practitioners' Perspectives», *Proceedings - 2021 IEEE International Conference on Software Analysis, Evolution and Reengineering, SANER 2021*, pp. 318-329, 2021, doi: 10.1109/SANER50967.2021.00037.
- [24] L. T. Portela y G. Borrego, «Scrumconix: Agile and documented method to AGSD», *Proceedings - 11th IEEE International Conference on Global Software Engineering, ICGSE 2016*, pp. 195-196, 2016, doi: 10.1109/ICGSE.2016.39.
- [25] J. Conklin, «A Taxonomy for Learning , Teaching , and Assessing : A Revision of Bloom 's Taxonomy of Educational Objectives», *Educational Horizons*, vol. 83, n.º 3, pp. 154-159, 2016.
- [26] P. Checkland y S. Holwell, «Action research: Its Nature and Validity», *System Practice and Action Research*, pp. 3-17, 2003, doi: 10.1007/978-0-387-36060-7_1.
- [27] H. G. Ridder, «The theory contribution of case study research designs», *Business Research*, vol. 10, n.º 2, pp. 281-305, 2017, doi: 10.1007/s40685-017-0045-z.
- [28] J. C. S. do P. Leite y C. Cappelli, «Software Transparency», *Business & Information Systems Engineering*, vol. 2, n.º 3, pp. 127-139, 2010, doi: 10.1007/s12599-010-0102-z.
- [29] J. Zhi, V. Garousi-Yusifolu, B. Sun, G. Garousi, S. Shahnewaz, y G. Ruhe, «Cost, benefits and quality of software development documentation: A systematic mapping», *Journal of Systems and Software*, vol. 99, pp. 175-198, 2015, doi: 10.1016/j.jss.2014.09.042.
- [30] V. S. Chomal y J. Saini, «Significance of Software Documentation in Software Development Process», *International Journal of Engineering Innovation & Research*, vol. 3, n.º 4, pp. 410-416, 2014.

- [31] D. Pajk, M. Indihar-Stemberger, y A. Kovacic, «Reference model design: An approach and its application», IEEE, 2012. doi: 10.2498/iti.2012.0419.
- [32] V. R. Basili y G. Caldiera, «The Goal Question Metric Paradigm», *Encyclopedia of Software Engineering - 2 Volume Set*, vol. 2, pp. 528-532, 2000, [En línea]. Disponible en: <https://go.umd.edu/3ol7AtO>.
- [33] A. Javier, «ISACA Glossary of Terms», 2015. Accedido: jun. 28, 2022. [En línea]. Disponible en: <https://bit.ly/3A4ILWN>.
- [34] A. R. Mondragón Pérez, «¿Qué son los indicadores?», *Revista de información y análisis*, n.º 19, pp. 52-58, 2002.
- [35] T. R. Gruber, «Toward principles design ontologies used for knowledge sharing», *International Journal of Human-Computer Studies*, vol. 43, n.º 5-6, pp. 908-928, 1995.
- [36] F. Ruiz y J. R. Hilera, «Using ontologies in software engineering and technology», *Ontologies for Software Engineering and Software Technology*, pp. 49-102, 2006, doi: 10.1007/3-540-34518-3_2.
- [37] J.-C. Narváez-Narváez, C.-J. Pardo-Calvache, y C.-E. Orozco-Garcés, «Deuda de la documentación en el desarrollo ágil de software: mapeo sistemático de la literatura», *Revista Científica*, vol. 46, n.º 1, pp. 107-121, ene. 2023, doi: 10.14483/23448350.19670.
- [38] K. Petersen, S. Vakkalanka, y L. Kuzniarz, «Guidelines for conducting systematic mapping studies in software engineering: An update», *Information and Software Technology*, vol. 64, pp. 1-18, 2015, doi: 10.1016/j.infsof.2015.03.007.
- [39] B. Kitchenham y S. M. Charters, «Guidelines for performing Systematic Literature Reviews in Software Engineering», ene. 2007. <https://bit.ly/3L8EOFW> (accedido jun. 18, 2022).
- [40] R. Wieringa, N. Maiden, N. Mead, y C. Rolland, «Requirements engineering paper classification and evaluation criteria: A proposal and a discussion», *Requirements Engineering*, vol. 11, n.º 1, pp. 102-107, 2006, doi: 10.1007/s00766-005-0021-6.
- [41] T. Dybå y T. Dingsøyr, «Strength of Evidence in Systematic Reviews in Software Engineering», *ESEM'08: Proceedings of the 2008 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, n.º 7465, pp. 178-187, 2008, doi: 10.1145/1414004.1414034.
- [42] L. Yang *et al.*, «Quality Assessment in Systematic Literature Reviews: A Software Engineering Perspective», *Information and Software Technology*, vol. 130, n.º March 2019, p. 106397, 2021, doi: 10.1016/j.infsof.2020.106397.
- [43] M. Ivarsson y T. Gorschek, «A method for evaluating rigor and industrial relevance of technology evaluations», *Empirical Software Engineering*, vol. 16, n.º 3, pp. 365-395, 2011, doi: 10.1007/s10664-010-9146-4.
- [44] S. Jalali y C. Wohlin, «Systematic Literature Studies: Database Searches vs. Backward Snowballing», *International Symposium on Empirical Software Engineering and Measurement*, p. 29, 2012, doi: 10.1145/2372251.2372257.
- [45] I. Sommerville, «Software Documentation», *Software Review*, vol. 1, n.º 2, pp. 165-167, 2001.
- [46] W. Ding, P. Liang, A. Tang, y H. Van Vliet, «Knowledge-based approaches in software documentation: A systematic literature review», *Information and Software Technology*, vol. 56, n.º 6, pp. 545-567, 2014, doi: 10.1016/j.infsof.2014.01.008.
- [47] M. Priestley y M. H. Utt, «Unified process for software and documentation development», *IEEE International Professional Communication Conference*, pp. 221-238, 2000.
- [48] N. J. Haneef, «Software Documentation and Readability: A Proposed Process Improvement», *ACM SIGSOFT Software Engineering Notes*, vol. 23, n.º 3, pp. 75-77, 1998.
- [49] C. Treude, J. Middleton, y T. Atapattu, «Beyond accuracy: Assessing software

- documentation quality», *ESEC/FSE 2020 - Proceedings of the 28th ACM Joint Meeting European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pp. 1509-1512, 2020, doi: 10.1145/3368089.3417045.
- [50] D. V. Koznov, D. V. Luciv, y G. A. Chernishev, «Duplicate management in software documentation maintenance», *CEUR Workshop Proceedings*, vol. 1989, pp. 195-201, 2017.
- [51] M. Linares-Vásquez, B. Li, C. Vendome, y D. Poshyvanyk, «Documenting database usages and schema constraints in database-centric applications», *ISSTA 2016 - Proceedings of the 25th International Symposium on Software Testing and Analysis*, pp. 270-281, 2016, doi: 10.1145/2931037.2931072.
- [52] J. Bayer y D. Muthig, «A view-based approach for improving software documentation practices», *Proceedings of the International Symposium and Workshop on Engineering of Computer Based Systems*, pp. 269-278, 2006, doi: 10.1109/ECBS.2006.18.
- [53] M. Zanoni, F. Perin, F. A. Fontana, y G. Viscusi, «Pattern detection for conceptual schema recovery in data-intensive systems», *Journal of Software: Evolution and Process*, vol. 26, n.º 12, pp. 1172-1192, 2014, doi: 10.1002/smr.
- [54] J. D. Arthur y K. T. Stevens, «Assessing the adequacy of documentation through document quality indicators», *Conference on Software Maintenance*, pp. 40-49, 1989, doi: 10.1109/icsm.1989.65192.
- [55] R. Plosch, A. Dautovic, y M. Saft, «The value of software documentation quality», *Proceedings - International Conference on Quality Software*, pp. 333-342, 2014, doi: 10.1109/QSIC.2014.22.
- [56] F. F. Correia, H. S. Ferreira, A. Aguiar, y N. Flores, «Patterns for consistent software documentation», *ACM International Conference Proceeding Series*, 2010, doi: 10.1145/1943226.1943241.
- [57] M. Kajko-Mattsson, «A Survey of Documentation Practice within Corrective Maintenance», *Empirical Software Engineering*, vol. 10, n.º 1, pp. 31-55, 2004, doi: 10.1023/b:lida.0000048322.42751.ca.
- [58] A. Forward y T. C. Lethbridge, «The Relevance of Software Documentation, Tools and Technologies: A Survey», en *Proceedings of the 2002 ACM Symposium on Document Engineering*, 2002, pp. 26-33, doi: doi.org/10.1145/585058.585065.
- [59] D. Poshyvanyk y a Marcus, «Using Traceability Links to Assess and Maintain the Quality of Software Documentation», *ACM Intl Symp on Grand Challenges in Traceability GCTTEFSE*, vol. 28, pp. 27-30, 2007, [En línea]. Disponible en: <http://www.cs.wm.edu/~denys/pubs/PosMar.QualityDocs.TEFSE2007.pdf>.
- [60] S. Nagano, Y. Ichikawa, y T. Kobayashi, «Recovering traceability links between code and documentation for enterprise project artifacts», *Proceedings - International Computer Software and Applications Conference*, pp. 11-18, 2012, doi: 10.1109/COMPSAC.2012.10.
- [61] D. Fensel *et al.*, «Ontology-based knowledge management», *Computer*, vol. 35, n.º 11, pp. 56-59, 2002, doi: 10.1109/MC.2002.1046975.
- [62] M. Fernández-López, A. Gómez-Pérez, y N. Juristo, «METHONTOLOGY: From Ontological Art Towards Ontological Engineering», *Proceedings of the Ontological Engineering AAAI-97 Spring Symposium Series | AAAI-97 Spring Symposium Series | 24-26 March 1997 | Stanford University, EEUU*, 1997, Accedido: abr. 22, 2023. [En línea]. Disponible en: bit.ly/46vJVbR.
- [63] C. Tautz y C. G. Von Wangenheim, «REFSENO: A Representation Formalism for Software Engineering Ontologies», 1998. Accedido: abr. 25, 2023. [En línea]. Disponible en: bit.ly/44t22gH.
- [64] D. B. Jayasuriya y I. Perera, «Ontology Based Software Design Documentation for Design Reasoning», *MERCon 2019 - Proceedings, 5th International Multidisciplinary*

- Moratuwa Engineering Research Conference*, pp. 710-715, 2019, doi: 10.1109/MERCon.2019.8818813.
- [65] N. S. R. Alves, L. F. Ribeiro, V. Caires, T. S. Mendes, y R. O. Spínola, «Towards an ontology of terms on technical debt», *Proceedings - 2014 6th IEEE International Workshop on Managing Technical Debt, MTD 2014*, n.º March 2020, pp. 1-7, 2014, doi: 10.1109/MTD.2014.9.
- [66] F. García *et al.*, «Effective use of ontologies in software measurement», *Knowledge Engineering Review*, vol. 24, n.º 1, pp. 23-40, 2009, doi: 10.1017/S0269888909000125.
- [67] C. J. Pardo-Calvache, F. O. García-Rubio, M. Piattini-Velthuis, F. J. Pino-Correa, y M. T. Baldassarre, «A reference ontology for harmonizing processreference models», *Revista Facultad de Ingenieria*, vol. 1, n.º 73, pp. 29-42, 2014.
- [68] F. García *et al.*, «Towards a consistent terminology for software measurement», *Information and Software Technology*, vol. 48, n.º 8, pp. 631-644, 2006, doi: 10.1016/j.infsof.2005.07.001.
- [69] G. Cugola y C. Ghezzi, «Software processes: a retrospective and a path to the future», *Software Process: Improvement and Practice*, vol. 4, n.º 3, pp. 101-123, 1998, doi: 10.1002/(sici)1099-1670(199809)4:3<101::aid-spip103>3.0.co;2-k.
- [70] International Organization for Standardization ISO, «ISO/IEC 12207 - Software life cycle processes», 2017.
- [71] International Organization for Standardization ISO, «Norma Técnica Colombiana NTC-ISO 31000», 2011.
- [72] H. F. Soares, N. S. R. Alves, T. S. Mendes, M. Mendonca, y R. O. Spinola, «Investigating the Link between User Stories and Documentation Debt on Software Projects», *Proceedings - 12th International Conference on Information Technology: New Generations, ITNG 2015*, pp. 385-390, 2015, doi: 10.1109/ITNG.2015.68.
- [73] C. J. Satish y M. Anand, «Software documentation management issues and practices: A survey», *Indian Journal of Science and Technology*, vol. 9, n.º 20, 2016, doi: 10.17485/ijst/2016/v9i20/86869.
- [74] R. Almeida, I. Percheiro, C. Pardo, y M. M. da Silva, «An ontology-based model for ITIL process assessment using TIPA for ITIL», *Communications in Computer and Information Science*, vol. 918, pp. 104-118, 2018, doi: 10.1007/978-3-030-00623-5_8.
- [75] IEEE, «ISO/IEC/IEEE 15939: Systems and software engineering — Measurement process», 2017. <https://bit.ly/469uLZg> (accedido sep. 28, 2023).
- [76] E. Aghajani *et al.*, «Software Documentation Issues Unveiled», *Proceedings - International Conference on Software Engineering*, vol. 2019-May, pp. 1199-1210, 2019, doi: 10.1109/ICSE.2019.00122.
- [77] R. Flesch, «A new readability yardstick», *Journal of Applied Psychology*, vol. 32, n.º 3, pp. 221-233, 1948, doi: 10.1037/h0057532.
- [78] G. Duraisamy y R. Atan, «Requirement traceability matrix through documentation for SCRUM methodology», *Journal of Theoretical and Applied Information Technology*, vol. 52, n.º 2, pp. 154-159, 2013.
- [79] P. Runeson y M. Höst, «Guidelines for conducting and reporting case study research in software engineering», *Empirical Software Engineering*, vol. 14, n.º 2, pp. 131-164, 2009, doi: 10.1007/s10664-008-9102-8.
- [80] «Payscale - Salary Comparison, Salary Survey, Search Wages». <https://www.payscale.com/> (accedido nov. 08, 2023).

Anexos

Anexo A: Informe técnico del modelo de métricas

MODELO DE MÉTRICAS PARA APOYAR LA EVALUACIÓN DE LA DEUDA DE LA DOCUMENTACIÓN EN
PROYECTOS DE SOFTWARE ÁGIL



JUAN CARLOS NARVÁEZ NARVÁEZ

Reporte técnico

Director:
César Jesús Pardo Calvache, Ph.D. MSc.

Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Sistemas
Grupo de Investigación y Desarrollo en Tecnologías de la Información - GTI
Línea de investigación en Ingeniería de Software
Popayán, noviembre de 2023

1. Introducción

En el contexto del desarrollo de software, la documentación hace referencia a los artefactos, físicos, digitales y web, que permiten describir las características, funcionalidades y detalles de uso del software; así como los procesos de desarrollo, mantenimiento y gestión que permitieron su construcción. Es por esto que la documentación tiene una importancia estratégica relevante para las organizaciones ya que permite: comprender el sistema de manera holística, transferir conocimientos, realizar eficientemente las tareas de mantenimiento o actualización, mejorar la calidad del sistema, brindar soporte a los usuarios finales, entre otros. Sin embargo, cuando la documentación no tiene la calidad suficiente ocurre un fenómeno bastante nocivo para software conocido como *deuda técnica de la documentación*, el cual se relaciona con la documentación inadecuada, desactualizada, insuficiente, incomprensible o inaccesible. Además, este inconveniente suele agravarse en proyectos ágiles donde se busca la entrega de software funcional en ciclos cortos de tiempo y la colaboración cercana entre los miembros del equipo, pero se suele pasar por alto la documentación o se minimiza su importancia. Por lo tanto, es de vital importancia conocer el nivel de deuda en la documentación de software para mejorarla e impactar positivamente en la eficiencia, calidad y escalabilidad del sistema.

En ese sentido, en este reporte técnico se presenta la primera versión del modelo de métricas diseñado para medir la deuda de la documentación en el desarrollo ágil de software, el cual está asociado a un conjunto de diecisiete características de la documentación que se encuentran agrupadas en cinco dimensiones de análisis. En la Figura 1 se presentan los componentes del modelo de métricas compuesto por: (i) 5 dimensiones de análisis de la documentación; (ii) 17 características de la documentación; y (iii) 39 métricas asociadas a dichas características. Adicionalmente, el modelo propone la definición de quince riesgos que pueden ocasionar deuda en la documentación, los cuales fueron obtenidos a partir de un conjunto de causas y efectos descritos en: <https://bit.ly/3WBY5mz>, <https://bit.ly/3WZGYdU>.

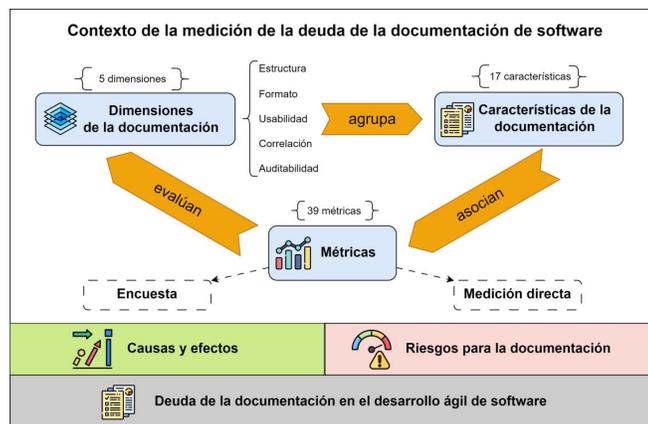


Figura 1. Estructura del modelo de métricas propuesto.

2. Propósito y alcance

El modelo de métricas permite medir la deuda de la documentación en proyectos de software ágil y establece un conjunto de indicadores capaces de valorizar los riesgos relacionados con los atributos más críticos e importantes de la documentación de software. Los problemas más típicos de la documentación son: la incompletitud, ambigüedad, ejemplos inexplicados, obsolescencia, incoherencia, incorrección, hinchazón, fragmentación, exceso de información estructural e información enmarañada; estos problemas pueden llevar a que la documentación no sea confiable, sea inadecuada, incompleta, inexistente, carecer de trazabilidad, desactualizada y carente de integridad. En consecuencia, pueden verse comprometidos aspectos como: la eficiencia, la calidad y la sostenibilidad del software. El modelo propuesto permitiría evaluar algunas características de la documentación de software para identificar sus falencias y diseñar estrategias para construir una documentación sólida que beneficie a todos los aspectos del ciclo de vida del software, desde el diseño y la implementación hasta el mantenimiento y la adopción por parte de los usuarios.

3. Limitaciones del modelo

El modelo propuesto presenta algunas limitaciones que deben tenerse en cuenta en el momento de aplicarlo e interpretar los resultados que se obtengan a partir de él. Específicamente, el modelo propuesto podría no ser capaz de capturar o representar adecuadamente la subjetividad inherente en la evaluación de la deuda en la documentación, ya que diferentes individuos que lo utilicen pueden poseer percepciones distintas sobre lo que constituye una documentación de calidad o sobre el nivel de deuda que podría ser tolerable. En ese sentido, otra limitación del modelo se relaciona con la dificultad de medir con precisión la capacidad de la documentación para ser comprendida por los usuarios, ya que es un proceso extremadamente subjetivo que depende en gran medida de la experiencia y habilidades del lector. Asimismo, es posible que para el modelo resulte difícil medir la precisión técnica de la documentación debido a que puede contener errores técnicos o inexactitudes. Finalmente, a pesar de que el modelo intenta trabajar con valores cuantitativos en cantidades exactas como la cantidad de documentos con marcas de confidencialidad, la cantidad de secciones de un documento que coincide con una plantilla, entre otros; no tiene en cuenta la medición específica de aspectos con una alta carga de subjetividad como: la calidad de la documentación o su utilidad real para la organización.

4. Dimensiones y características de la documentación de software

Otro componente del modelo de métricas corresponde a las 5 dimensiones de análisis definidas para estudiar la documentación de software, las cuales se describen a continuación:

Estructura

Esta dimensión corresponde a la forma como está distribuido el contenido de un artefacto donde cada parte debe cumplir con una función específica y correlacionarse con las demás partes para darle sentido. Por lo tanto, en esta dimensión se pueden considerar elementos como: capítulos, subcapítulos, índice, listas de ilustraciones, listas de tablas, introducción, instrucciones de uso, posibles errores y su solución, glosario, artefactos relacionados, navegación y conceptos importantes.

Formato

Esta dimensión corresponde a la presentación visual del contenido, así como el lenguaje y la gramática utilizados para facilitar la comprensión de la información. En ese sentido, los elementos que hacen parte de esta dimensión son: estilo de redacción, cantidad de páginas o vistas, uso de diagramas o ejemplos, disposición espacial, complejidad técnica, ortografía y gramática.

Usabilidad

La usabilidad indica la facilidad con la que un lector puede extraer toda la información necesaria para cumplir con sus objetivos inmediatos. En ese sentido, esta dimensión permite estudiar el uso efectivo de la información cumpliendo con requisitos que influyen en su accesibilidad a través del esquema de navegación en los artefactos. Por lo tanto, esta dimensión tiene en cuenta la estructura del artefacto, su formato, la facilidad para acceder y utilizar su información, la efectividad de su navegación y los mecanismos para ser preservado.

Correlación

Esta dimensión se refiere a la correspondencia que debe existir entre el sistema y la documentación que describe sus funcionalidades, características técnicas, los procesos que se siguieron y en general todo aquello que permitió su implementación, utilización y soporte. De esta forma, se abarcan los atributos de la documentación enfocados en mantener actualizada la información cada vez que ocurre un cambio en el sistema; manteniendo su correspondencia con la última versión para evitar confusiones, inconvenientes o sobre esfuerzos en el futuro.

Auditabilidad

Esta dimensión se refiere a los atributos de la documentación que permiten su inspección y verificación sistemáticas con respecto al comportamiento del software, al tipo de información que se registra y a los criterios establecidos para su revisión. Con esto se busca que la documentación apoye procesos de auditoría internos o externos describiendo el historial de las acciones y decisiones tomadas para la construcción del software.

5. Características de la documentación

Se han definido 17 características o atributos de la documentación de software clasificadas en las 5 dimensiones mencionadas anteriormente, las cuales se presentan en la Tabla 1.

Tabla 1. Características o atributos de la documentación de software.

Dimensión	Estructura	Formato	Usabilidad	Correlación	Auditabilidad
Características de la documentación	Uniformidad Adaptabilidad Modularidad	Legibilidad Conciso Claridad	Accesibilidad Confidencialidad Consistencia Precisión	Portabilidad Integridad Actualidad Trazabilidad Expansibilidad	Validez Verificabilidad

Uniformidad (Uniformity)

Esta característica hace referencia a que la información del artefacto sea coherente y no existan conflictos. Para esto, se debe detectar, analizar y corregir las contradicciones y la duplicidad de información en la documentación. En consecuencia, se encuentra fuertemente relacionada con la característica de *consistencia*, ya que su ausencia puede confundir a los usuarios del artefacto y obligarlos a sobre esforzarse para comprender su contenido. También se relaciona con la *integridad* ya que si el artefacto no contiene la información necesaria sobre el sistema los profesionales y usuarios finales no podrían realizar sus tareas adecuadamente. Asimismo, se relaciona con la característica de *actualidad* por que facilita que la información permanezca correcta y completa después de actualizar el sistema y permite la reutilización de la documentación en futuros desarrollos. La uniformidad también facilita que la documentación pueda revisarse adecuadamente durante el control de calidad, corregir rápidamente los errores y reestructurar los artefactos. Para esto, se pueden considerar cinco tipos de uniformidad:

- **Uniformidad en el formato:** Se refiere a adecuar el estilo de redacción y utilizar correctamente las reglas gramaticales para mantener la unanimidad de la información de cada artefacto de acuerdo a su propósito.
- **Uniformidad en la estructura:** Se refiere a estandarizar la presentación de la información en el artefacto mediante la numeración de páginas, encabezados, notas al pie, figuras, tablas, esquema de navegación, entre otros, teniendo en cuenta el propósito del artefacto y sus posibles usuarios.
- **Uniformidad en el almacenamiento de la información:** Se refiere a estandarizar los mecanismos de almacenamiento y preservación de todos los artefactos teniendo en cuenta las características particulares de cada uno.
- **Uniformidad en el acceso a la información:** Se refiere a estandarizar los mecanismos de acceso a la información de acuerdo al nivel de acceso otorgado a cada interesado en la documentación.
- **Uniformidad en la actualización de la documentación:** Se refiere a estandarizar los mecanismos que permiten actualizar la documentación, los cuales deben mantener una concordancia con las políticas de actualización del sistema que describe.

Adaptabilidad (Customizable)

Esta característica ha sido poco abordada en investigación y corresponde a la capacidad que tiene el documento para evolucionar y mantenerse relevante a medida que cambian las necesidades y los requisitos del proyecto. Beneficia en gran medida a la portabilidad de la documentación, sin embargo, se debe gestionar con precaución ya que en la práctica no es totalmente viable personalizar los artefactos ante cada nueva necesidad debido al elevado costo de planificar, crear y mantener esta nueva documentación. En su lugar, se puede crear modelos comunes a partir de los diferentes puntos de vista de todos los interesados que posteriormente sirvan para crear diferentes tipos de artefactos. Estos modelos también se utilizarían como una plantilla para añadir los aspectos específicos de cada interesado y así adaptar mejor la documentación.

Modularidad (Modularity)

Esta característica permite fragmentar un artefacto en partes o secciones cohesivas capaces de cumplir con un objetivo específico, interactuar entre sí y complementarse para presentar, estructurar y explicar el contenido del artefacto. La modularidad le permite a los usuarios de la documentación conocer ciertos detalles del sistema sin necesidad de estudiar toda la documentación de principio a fin. Para lograrlo, es necesario que cada sección sea lo más independiente posible, ya sea dentro de un artefacto o entre artefactos, y mantener solamente las relaciones necesarias. De esta forma se podrían modificar integrar o eliminar secciones con el menor impacto posible sobre toda la documentación.

Legibilidad (Readability)

Esta característica se relaciona con la facilidad de lectura de un artefacto, lo cual convierte a la legibilidad en un atributo subjetivo que depende de la percepción de los usuarios. En ese sentido, la información abstracta, demasiado técnica o sobre cargada, y errores tipográficos son aspectos en la documentación que puede afectar su legibilidad. Por otro lado, es importante aclarar la diferencia entre legibilidad y comprensibilidad; puntualmente, la legibilidad mide el esfuerzo de los usuarios para acceder a la información contenida en el artefacto, mientras que la comprensibilidad mide la complejidad de dicha información. La legibilidad puede estudiarse desde cuatro perspectivas: visual, espacial, de alineación y lingüística, así como la coherencia textual y la cantidad de conceptos utilizados.

Conciso (Conciseness)

Esta característica hace referencia a la capacidad de la documentación para utilizar la información estrictamente necesaria para explicar los detalles del sistema sin perder la exactitud en su contenido. Está fuertemente relacionada con la legibilidad debido a que el contenido del artefacto debe ser lo suficientemente detallado para describir correctamente las funcionalidades del sistema sin incluir secciones inútiles que ocasionen discrepancias o lo hagan pesado, incoherente e incomprensible.

Claridad (Clarity)

Esta característica se relaciona con la capacidad del contenido para ser comprensible e inteligible sin que genere dudas o incertidumbres. La claridad depende en gran medida de utilizar un lenguaje apropiado y un estilo de redacción simple y eficiente, que permita a los usuarios encontrar rápidamente la información que requieren sin necesidad de releer el contenido. Asimismo, la claridad debe tener en cuenta la estructura, el lenguaje visual y el diseño del contenido para que el mensaje sea transmitido de manera efectiva a los usuarios, de acuerdo a conocimientos técnicos y capacidades cognitivas. Por lo tanto, la claridad es un atributo subjetivo que depende también de la percepción de quien utiliza el artefacto, con lo cual, también se relaciona con la legibilidad de la información. Sin embargo, es posible identificar la falta de claridad en aspectos tan específicos como ejemplos de código poco claros, nombres de métodos confusos, títulos confusos, suposición de conocimientos previos, uso excesivo de siglas, entre otros.

Accesibilidad (Accessibility)

Esta característica se refiere a la capacidad de la documentación para ser localizada y operativa cuando sea necesario para recuperar, presentar, leer e interpretar su contenido. La accesibilidad es una de las características sobre la cual se ha discutido bastante junto con la integridad y la consistencia debido a que considera la disponibilidad de la información, el nivel de ocultamiento que se maneja para cada usuario y la facilidad de localización. Sin embargo, esta característica va más allá de solo permitir el acceso a la información, ya que involucra entender el contexto organizacional, empresarial y técnico sobre el cual se va a utilizar la documentación para comprender claramente en qué momento se debe crear, consultar, modificar o eliminar un artefacto.

Confidencialidad (Privacy)

Tomando como base la norma ISO 27001 se seguridad de la información, la confidencialidad es una propiedad de la documentación para estar disponible solo para aquellas personas, entidades o procesos autorizados. Es una característica clave, junto con la disponibilidad e integridad, para mantener segura la información. Para que la documentación cuente con esta característica, es necesario clasificar los artefactos de acuerdo a su nivel de accesibilidad: público (abierto a todo el público), de uso interno (exclusivo para el equipo de trabajo), restringido (accesible de acuerdo a las políticas de la empresa) y confidencial (accesible solo con autorización). Debe ser una característica claramente visible en los artefactos, por ejemplo, utilizando etiquetas de color que indiquen su clasificación.

Consistencia (Consistency)

Hace referencia a la capacidad de la documentación para evitar discrepancias potenciales o reales entre el contenido y el funcionamiento real del sistema. La consistencia puede estudiarse desde diferentes perspectivas, por ejemplo, en la formalidad del artefacto, la semántica, contenido visual, organización del contenido y en el formato. La consistencia se apoya en la uniformidad e integridad del artefacto, ya que si la información es inconsistente y contradictoria causará confusión y sobreesfuerzos para comprenderla. La consistencia también involucra la gestión adecuada de las relaciones entre artefactos debido a que la información podría estar duplicada o dispersa en varios artefactos. Sin embargo, no se suele mantener correctamente la consistencia y trazabilidad de la documentación debido, generalmente, a los tiempos de entrega del software.

El resultado de esto es una documentación envejecida y que fracasa en su función fundamental de comunicar. De esta forma, la trazabilidad se convierte en un indicio del grado de consistencia en la descripción de los distintos aspectos de un sistema.

Precisión (Accuracy)

Esta característica se relaciona con la capacidad del artefacto para transmitir la información de manera clara, exacta y sin ambigüedades, reflejando el estado real del sistema que describe. Por el contrario, cuando la documentación es imprecisa o demasiado abstracta puede ser mucho peor que si no existiera y esto se debe a que puede impedir o confundir a los usuarios cuando realizan sus actividades. Por lo tanto, la documentación debe ser lo más precisa posible al describir las características y funcionalidades del software para que sea altamente efectiva durante la comunicación entre todas las partes interesadas.

Ahora bien, según Arthur y Stevens en, la falta de precisión en la documentación se pueda dar por diferentes razones, pero se destacan dos; la primera, por cometer errores en la traducción de los requisitos del sistema a instrucciones de diseño; y la segunda, por una mala interpretación de las especificaciones de diseño o la actualización del sistema sin actualizar la documentación respectiva. En cualquiera de los dos casos, el resultado es que toda la documentación, o al menos una parte, será incorrecta e imprecisa con respeto al funcionamiento real del sistema.

Portabilidad (Portability)

Según el diccionario de la Real Academia Española, la portabilidad es la cualidad de un objeto para que sea movable y fácil de transportar, mientras que la ISO 25010 la define como la capacidad de un producto para ser transferido de un entorno a otro de manera efectiva y eficiente. Teniendo en cuenta estas definiciones en el ámbito de la documentación, la portabilidad es una característica de la documentación que permite transferirla de un entorno tecnológico, operacional o de uso a otro de forma eficiente y efectiva. Para lograrlo, la documentación debe ser: (i) adaptable a diferentes entornos hardware, software u operacionales, (ii) fácil de disponer en un determinado entorno, y (iii) fácil de reemplazar ante un fallo o inconveniente. Actualmente, existen diferentes tipos de artefactos (física, digita, páginas web, wikis y aplicaciones móviles), con lo cual se pueden usar estándares como PDF (Portable Document Format), el cual es reconocido por la Organización Internacional para la Estandarización (ISO) como un formato que permite presentar e intercambiar documentos de manera fiable sin depender del software, hardware o sistema operativo, con lo cual se logra que la documentación sea totalmente independiente e interoperable.

Integridad (Completeness)

Esta característica describe hasta qué punto la información de la documentación permanece correcta y completa después de un cambio en el sistema y mantiene su capacidad de apoyar las tareas de desarrollo y mantenimiento. La integridad, es una característica que asegura que la documentación contenga la información suficiente para describir el sistema, de tal manera que los usuarios puedan recuperar la información necesaria para realizar sus tareas. Por el contrario, la falta de integridad ocasiona que la documentación no cumpla con su propósito y sea inútil para los usuarios.

Actualidad (Up-to-dateness)

Es una característica que describe hasta qué punto la documentación se mantiene actualizada a lo largo del ciclo de vida del software. Es una característica que depende de una buena trazabilidad de los artefactos para que el sistema y su documentación evolucionen al mismo tiempo manteniendo una correspondencia. En ese sentido, la desactualización de la documentación ocurre cuando su elaboración no está sincronizada con el desarrollo del sistema. Por lo tanto, es muy importante mantener una documentación actualizada, de lo contrario, se describirían versiones anteriores del sistema, la información sería incorrecta y puede inducir al error a los usuarios.

Trazabilidad (Traceability)

Esta característica describe la evolución del artefacto con el propósito de identificar donde se hizo el cambio, en qué momento, quién lo realizó y la razón del cambio. De esta forma, la trazabilidad ayuda a rastrear la modificación de un artefacto y mantener su consistencia con el sistema. Sin embargo, a pesar de su importancia, es un aspecto muy descuidado en la documentación del software que requiere de mucha atención, en especial, en la documentación técnica debido a la correspondencia que debe mantener con el software. Por otro lado, la trazabilidad permite entender la relación entre dos o más artefactos y el nivel de dependencia entre ambos. Por lo tanto, es una característica que permite justificar y verificar todos los cambios que ocurren en el software, además de ser muy útil en procesos técnicos de ingeniería inversa, mantenimiento y pruebas de software.

Expansibilidad (Expandability)

Esta característica hace referencia a la capacidad agregar nueva información en un artefacto de documentación o modificar la que ya existe, por ejemplo, aumentando la extensión de su contenido, integrando otros tipos de contenido, cambiando el volumen de los artefactos o el alcance de los mismos, entre otros. Es una característica que se relaciona con la evolución y el mantenimiento del sistema ya que son etapas del ciclo de vida del software donde pueden surgir nuevas funcionalidades o actualizar las que ya existen. En ambos casos, se debe documentar cualquier tipo de expansión y verificar la integridad y la consistencia entre la documentación y el software. Por lo tanto, cada artefacto debe permitir la integración de nueva información sin afectar a las demás características.

Validez (Validity)

Esta característica acredita la autenticidad, veracidad, legalidad y fiabilidad de la información contenida en un artefacto. La validez se logra cuando un artefacto cuenta con dos elementos: (1) la aceptación y firma de las partes involucradas en su creación y (2) un sello de autenticidad por parte de la organización. De esta forma es posible formalizar los acuerdos entre las partes interesadas, verificar el cumplimiento de los compromisos adquiridos y certificar la pertinencia y propiedad de la información. Es una característica poco estudiada, pero es muy importante porque permite evaluar si el software y su documentación cumplen con las expectativas de los usuarios.

Verificabilidad (Correctness)

Esta característica permite corroborar si la documentación proporciona información exacta conforme a hechos reales sin entrar en conflictos con otros artefactos. De esta forma, si un artefacto presenta información incorrecta o que no haya sido verificada puede traer graves consecuencias no previstas, como, por ejemplo, inducir al error luego de utilizar la documentación, reproducir código fuente erróneamente o seguir instrucciones equivocadas de instalación y despliegue. Por lo tanto, es una característica que se relaciona con la exactitud del contenido y ayuda a mejorar su consistencia. Asimismo, se apoya en la legibilidad y estructuración del contenido ya que una buena gestión de cada uno aumentará la probabilidad de realizar una verificación efectiva.

6. Riesgos para la documentación

Además de las dimensiones y características de la documentación mencionadas anteriormente, se han definido un conjunto de 15 riesgos que pueden llevar a crear e incrementar la deuda técnica en la documentación del software. Para esta definición, se ha establecido el formato presentado en la Tabla 2, y en la Tabla 3 se muestra la definición de cada riesgo y las características con las que se relaciona.

Tabla 2. Formato para la definición de riesgos.

Para (quien o quienes se ven afectados por la ocurrencia del riesgo) + **existe el riesgo** (especificación del riesgo) + **debido a** (una o varias causas del riesgo) + **esto puede** (las consecuencias de la materialización del riesgo)

Tabla 3. Listado de riesgos propuesto para el modelo de métricas.

Id	Descripción	Características asociadas
R1	Para los equipos de desarrollo y de mantenimiento, existe el riesgo de no comprender correctamente el código fuente del sistema en el futuro, esto, debido a que no tiene comentarios que respalden la explicación de su intención y la documentación técnica es escasa, incomprensible, desactualizada o nula. Esto puede afectar la mantenibilidad y modificabilidad del sistema, el trabajo en equipo, la transferencia de conocimientos a nuevo personal, y la resolución de problemas.	Uniformidad, Claridad, Legibilidad, Integridad, Precisión, Modularidad
R2	Para los equipos de desarrollo, los administradores y usuarios finales, existe el riesgo que el sistema sea altamente vulnerable, esto, debido a que su documentación no describe correctamente su arquitectura, diseño, funcionalidades, interfaces y restricciones. Esto puede afectar la comprensión de riesgos y amenazas, dificultar la implementación o mejora de prácticas de seguridad, y aumentar la complejidad en la gestión y atención de incidentes.	Legibilidad, Conciso, Claridad, Consistencia, Verificabilidad
R3	Para la empresa existe el riesgo que el sistema sea muy difícil de mantener en el futuro debido a que su documentación no permite comprender adecuadamente su funcionamiento, su arquitectura, su diseño y las decisiones que se tomaron durante su construcción. Esto puede afectar negativamente la refactorización del código fuente, disminuye la coherencia en el desarrollo del sistema, dificulta la trazabilidad de los cambios e incrementa el esfuerzo necesario para identificar, comprender y solucionar los incidentes.	Uniformidad, Claridad, Consistencia, Actualidad, Trazabilidad, Validez, Verificabilidad
R4	Para la empresa existe el riesgo que el sistema sea de baja calidad, esto, debido a que su documentación no ofrece información clara y precisa que garantice la comprensión de su diseño, su arquitectura y su funcionamiento. Esto puede dificultar el control de calidad del sistema, su escalabilidad y mantenibilidad, la transferencia de conocimiento a nuevo personal, y aumentar la dependencia del personal que participó en su desarrollo inicialmente. Por otro lado, el sistema puede causar insatisfacción en los interesados y hacer que la empresa deteriore sus relaciones comerciales perdiendo competitividad en el mercado.	Legibilidad, Conciso, Claridad, Accesibilidad, Confidencialidad, Verificabilidad
R5	Para la empresa existe el riesgo que se generen reprocesos durante el desarrollo del sistema, esto, debido a que su documentación no es clara, está incompleta, es inconsistente o está desactualizada. Esto puede generar malos entendidos en la interpretación de los requisitos, se incrementa la probabilidad de cometer errores, incluso omitir información importante durante el diseño o la implementación. Esto obliga a realizar revisiones y reprocesos para corregir errores o incorporar información faltante, lo cual puede retrasar las entregas.	Legibilidad, Conciso, Claridad, Accesibilidad, Confidencialidad, Integridad, Validez, Verificabilidad
R6	Para la empresa existe el riesgo de acumular progresivamente niveles inmanejables de deuda técnica, esto, debido a la poca claridad en la documentación, la ausencia de buenas prácticas y estándares, falta de detalles sobre las dependencias y riesgos, y documentación inexistente o desactualizada del diseño y la arquitectura. Esto puede llevar a decisiones apresuradas o incorrectas durante el desarrollo, lo que resulta en soluciones subóptimas y de baja calidad.	Uniformidad, Claridad, Accesibilidad, Confidencialidad, Consistencia, Portabilidad
R7	Para la empresa existe el riesgo que los procesos de desarrollo y de calidad del sistema tengan un alto grado de informalidad, esto, debido a que la documentación no registra claramente los lineamientos, las pautas y los estándares que se deben seguir en cada etapa. Esto puede llevar a una falta de uniformidad y coherencia en la ejecución, seguimiento y control de los procesos, debido a la ambigüedad en la definición de los roles y responsabilidades. Por otro lado, se incrementa la probabilidad de perder conocimiento valioso debido a que los procesos se ejecutan con base en la intuición en lugar de seguir una guía documentada y registrar las lecciones aprendidas en cada paso.	Uniformidad, Claridad, Accesibilidad, Confidencialidad, Precisión, Portabilidad, Verificabilidad
R8	Para la empresa existe el riesgo de no mejorar sus procesos debido a que la documentación no registra adecuadamente como se deben ejecutar para que sean eficientes, consistentes y mejoren la calidad de las operaciones. Esto puede afectar la visibilidad de los procesos, el análisis y la medición de su desempeño, incrementa la incertidumbre, y limita la capacidad de mejora continua al no existir una base sólida de buenas prácticas y lecciones aprendidas.	Claridad, Accesibilidad, Confidencialidad, Consistencia
R9	Para los equipos encargados de implementar el sistema, existe el riesgo que el sistema no se pueda implementar correctamente en su ambiente, esto, debido a que la documentación no cuenta con la información precisa y completa para comprender su funcionamiento y cómo se debe instalar, configurar y utilizar correctamente. Esto puede afectar la planificación de la implementación, cometer errores u omisiones de etapas o actividades durante el proceso y dificulta la transferencia de conocimiento o el funcionamiento del sistema luego de su implementación.	Accesibilidad, Confidencialidad, Consistencia, Integridad, Verificabilidad
R10	Para la empresa existe el riesgo de perder el conocimiento aprendido por los miembros del equipo de desarrollo, esto, debido a documentación deficiente, incompleta o inexistente donde se registre información de aspectos relevantes como detalles y decisiones de diseño, funcionalidades, configuración,	Modularidad, Accesibilidad,

	integraciones, gestión de requisitos, soluciones a problemas y lecciones aprendidas. Esto puede incrementar la dependencia del conocimiento tácito de ciertos miembros del equipo, el esfuerzo para adquirir nuevo conocimiento y se disminuye la productividad del equipo.	Confidencialidad, Integridad, Verificabilidad
R11	Para el equipo de desarrollo existe el riesgo de una mala implementación de los requisitos del sistema, esto, debido a que la documentación los describe con ambigüedades. Esto puede ocasionar que los requisitos sean mal interpretados, inconsistentes o contradictorios entre sí causando confusión durante el diseño y el desarrollo del sistema. Por otro lado, aumenta la probabilidad de omitir detalles importantes en las funcionalidades clave, restricciones, reglas de negocio o requisitos no funcionales, sobrecargar al equipo y reducir su productividad.	Uniformidad, Claridad, Accesibilidad, Confidencialidad, Consistencia, Precisión, Verificabilidad
R12	Para la empresa existe el riesgo de no integrar eficazmente a nuevo personal al equipo de desarrollo, esto, debido a que la documentación del sistema no permite comprender adecuadamente el sistema, su estado actual, su funcionamiento, los requisitos que debe satisfacer, su diseño y su arquitectura. Esto puede prolongar la curva de aprendizaje para comprender los procesos, las políticas y los procedimientos, así como retrasar la adaptación del nuevo personal al contexto del sistema para contribuir plenamente al cumplimiento de los objetivos del equipo.	Legibilidad, Conciso, Confidencialidad, Consistencia
R13	Para el equipo de control de calidad existe el riesgo que sea muy difícil determinar qué aspectos del sistema deben ser probados, esto, debido a que la documentación presenta información ambigua o incompleta de los requisitos y no explica claramente cuáles son los criterios de éxito bajo los cuales se consideran cumplidos o hechos. Esto puede llevar al diseño de casos de prueba poco efectivos que no cubran todos los escenarios y datos de prueba relevantes, gestionar de forma inadecuada los defectos del sistema, perder la trazabilidad para determinar que partes del sistema se han probado y que casos de prueba certifiquen el cumplimiento o no de los requisitos.	Uniformidad, Modularidad, Legibilidad, Conciso, Accesibilidad, Confidencialidad, Precisión, Integridad, Actualidad, Trazabilidad, Validez, Verificabilidad
R14	Para la empresa existe el riesgo de perder la capacidad de realizar un seguimiento adecuado de la evolución y los cambios realizados a lo largo del tiempo en uno o varios artefactos de documentación debido a la deficiencia o ausencia de un control de versiones en la información de cada artefacto. Esto puede afectar la gestión del historial de versiones, dificultar la comprensión del contexto o la necesidad de las modificaciones, generar confusión y pérdida de información.	Modularidad, Accesibilidad, Confidencialidad, Consistencia, Precisión, Actualidad, Trazabilidad, Validez, Verificabilidad
R15	Para el equipo de desarrollo existe el riesgo que se generen problemas de comunicación debido a que la documentación es ambigua, poco clara o desactualizada en la descripción del sistema, de sus funcionalidades y sus condiciones de uso. Esto puede llevar a interpretaciones erradas, malentendidos, y confusiones en las comunicaciones internas y externas, afectando de igual forma la colaboración y la resolución de problemas con base en información obsoleta.	Uniformidad, Modularidad, Legibilidad, Conciso, Claridad, Accesibilidad, Consistencia, Precisión

7. Descripción del modelo de métrica

El modelo consta de 39 métricas relacionadas con las 17 características de la documentación presentadas anteriormente. A continuación, se presentan los detalles de las 39 métricas construidas utilizando la plantilla presentada en la Tabla 4.

Tabla 4. Plantilla para la descripción de métricas.

Nombre de la métrica	
Identificador	En este campo describe el identificador único de la métrica.
Propósito	En este campo describe el propósito de la métrica.
Unidad	En este campo expone la unidad que soporta la métrica.
Escala	En este campo expone la escala de valores que utiliza la métrica.
Ecuación	En este campo se expresa la expresión matemática de la métrica.
Variables	En este campo se listan las variables utilizadas en la ecuación.
Observaciones	En este campo se describen los aspectos que se deben tener en cuenta para la métrica.
Ejemplo de utilización	En este campo se describe un ejemplo de utilización de métrica.

A continuación, se presenta un extracto de las métricas desarrolladas para entender su estructura.

Métricas para la característica de uniformidad

Para medir la uniformidad de la documentación de software se han definido dos métricas. Inicialmente, la métrica PUD (Percepción de Uniformidad de la Documentación) es descrita en la Tabla 5, y permite establecer el promedio de una encuesta compuesta por diez preguntas y cinco opciones de respuestas ponderadas de 1 a 5 con las cuales se evalúa la percepción que tienen los usuarios sobre la uniformidad de la documentación en relación con su formato, estructura, almacenamiento, accesibilidad y actualización.

Tabla 5. Métrica de percepción de uniformidad.

Indicador de percepción de uniformidad de la documentación				
Identificador	PUD			
Propósito	Determinar la percepción de uniformidad en la documentación por parte de los usuarios de dicha documentación con base en la siguiente encuesta: https://bit.ly/46nwOcj .			
Unidad	Cantidad entera positiva			
Escala	[20, 100]			
Ecuación	$PUD = 2 * (P1 + P2 + P3 + P4 + P5 + P6 + P7 + P8 + P9 + P10)$			
Variables	<ul style="list-style-type: none"> • P1, P2, ..., P10: Hace referencia al valor entre 1 y 5 que se le puede asignar a las preguntas 1 a 10 según las posibles respuestas en la encuesta. • PUD: Indicador de la percepción general de uniformidad en la documentación. 			
Observaciones	Las opciones de respuesta de la encuesta tienen la siguiente ponderación:			
	Totalmente en desacuerdo	Nunca	Muy insatisfecho	1
	En desacuerdo	Casi nunca	Insatisfecho	2
	Neutral	Algunas veces	Neutral	3

	De acuerdo	Frecuentemente	Satisfecho	4
	Totalmente de acuerdo	Siempre	Muy satisfecho	5
Ejemplo de utilización	<p>Para aplicar la métrica PUD sobre un artefacto de documentación, se debe responder la encuesta que tiene asociada seleccionando una de las opciones de respuesta disponibles en cada pregunta las cuales tienen un puntaje entre 1 y 5 como se describe en las <i>observaciones</i>. Suponiendo que las respuestas a las preguntas P1 a P10 de la encuesta hayan sido las siguientes:</p> <p>P1: Neutral P2: En desacuerdo P3: Totalmente de acuerdo</p> <p>P4: Nunca P5: Frecuentemente P6: Siempre</p> <p>P7: Algunas veces P8: Muy satisfecho P9: Satisfecho P10: Muy satisfecho</p> <p>Entonces se reemplazarán los siguientes valores en la ecuación:</p> $PUD = 2 * (3 + 1 + 5 + 1 + 4 + 5 + 3 + 5 + 4 + 5) = 2 * (36) = 72$ <p>Por lo tanto, se tiene una percepción del 72% de uniformidad en el artefacto.</p>			

Por otro lado, la métrica UPD (Utilización de las Plantillas de Documentación) es descrita en la Tabla 6, y permite determinar el nivel de correspondencia entre el artefacto de documentación que se está analizando y la plantilla que haya definido la organización para dicho artefacto. Esta correspondencia se determina a partir del número de secciones y los títulos de cada sección que coinciden entre el artefacto y su plantilla.

Tabla 6. Métrica para medir el nivel de fidelidad con la plantilla.

Indicador de utilización de las plantillas de documentación	
Identificador	UPD
Propósito	Determinar el nivel de fidelidad del artefacto con la plantilla que haya diseñado la organización comparando la cantidad de secciones que coinciden entre el documento y la plantilla definida para ese documento.
Unidad	Porcentaje (%)
Escala	[0, 100]
Ecuación	$UPD = \left(\frac{SC}{SP}\right) * 100$
Variables	<ul style="list-style-type: none"> • <i>SC</i>: Cantidad de secciones del documento que coinciden con la plantilla. • <i>SP</i>: Total de secciones de la plantilla.
Observaciones	Para esta métrica, si la organización no ha definido una plantilla para documentar el artefacto que se está analizando, entonces el valor de UPD será igual a cero.
Ejemplo de utilización	<p>Para aplicar la métrica UPD se debe contar la cantidad de secciones que coinciden entre el artefacto de documentación y la plantilla que haya definido la organización para construirlo. Sin embargo, si la organización no ha definido una plantilla, entonces UPD = 0.</p> <p>De esta forma, si la plantilla consta de 6 secciones (SP) y artefacto coincide con 4 secciones (SC), entonces:</p> $UPD = \left(\frac{4}{6}\right) * 100 = 0,667 * 100 = 66.7\%$ <p>Por lo tanto, se tiene que el artefacto coincide en un 66.7% con respecto a su plantilla.</p>

Anexo B: Formulario de recolección de datos

Historia de Usuario: _____

Métricas de uniformidad:

1	Métrica	Preguntas										Aplicación de la métrica
	PUD	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	

2	Métrica	Preguntas			Resp.	Aplicación de la métrica
	UPD	SP: ¿Cuántas secciones tiene la plantilla definida por la organización?				
		SC: ¿Cuántas secciones del artefacto coinciden con la plantilla?				

Métricas de adaptabilidad:

3	Métrica	Preguntas					Aplicación de la métrica
	PAD	P11	P12	P13	P14	P15	

Métricas de modularidad:

4	Métrica	Preguntas					Aplicación de la métrica
	PMD	P16	P17	P18	P19	P20	

5	Métrica	Preguntas			Resp.	Aplicación de la métrica
	IAD	TNG: ¿Cuántos documentos se utilizaron en el grafo?				
		PA: ¿Cuál es la suma de los pesos de todas las aristas del grafo?				
		n: ¿Cuántas aristas tiene el grafo?				

Métricas de legibilidad:

6	Métrica	Preguntas				Aplicación de la métrica
	PLD	P21	P22	P23	P24	

7	Métrica	Preguntas		Resp.	Aplicación de la métrica
	IFKE	NTP: ¿Cuántas palabras tiene el artefacto?			
		NTO: ¿Cuántas oraciones tiene el artefacto?			
		NTS: ¿Cuántas sílabas tiene el artefacto?			

Métricas de concisión:

8	Métrica	Preguntas						Aplicación de la métrica
	PCD	P25	P26	P27	P28	P29	P30	

Métricas de claridad:

9	Métrica	Preguntas				Aplicación de la métrica
	PCLD	P31	P32	P33	P34	

10	Métrica	Preguntas		Resp.	Aplicación de la métrica
	PPO	TPC: ¿Cuántas palabras tiene el texto?			
		TOC: ¿Cuántas oraciones tiene el texto?			

11	Métrica	Preguntas			Resp.	Aplicación de la métrica
	PFO	TFC: ¿Cuántas frases tiene el texto?				

		TOC: ¿Cuántas oraciones tiene el texto?		
--	--	---	--	--

Métricas de accesibilidad:

12	Métrica	Preguntas					Aplicación de la métrica
	PACD	P35	P36	P37	P38	P39	

13	Métrica	Preguntas		Resp.	Aplicación de la métrica
	SAS	SAR: ¿Cuántas solicitudes de acceso se han realizado?	SAA: ¿Cuántas solicitudes de acceso han sido aprobadas?		

Métricas de confidencialidad:

14	Métrica	Preguntas				Aplicación de la métrica
	PConD	P41	P42	P43	P44	

15	Métrica	Preguntas		Resp.	Aplicación de la métrica
	AMC	TAE: ¿Cuántos artefactos fueron analizados?	TAM: ¿Cuántos artefactos tienen marcas o etiquetas de confidencialidad?		

16	Métrica	Preguntas		Resp.	Aplicación de la métrica
	DNA	MCNA: ¿Cuántos miembros del equipo tienen claro su nivel de acceso a la documentación?	ME: ¿Cuántos miembros tiene el equipo?		

17	Métrica	Preguntas		Resp.	Aplicación de la métrica
	DLA	TDP: ¿Cuántos documentos son de carácter público?	TDSN: ¿Cuántos documentos no tienen definido un nivel de acceso?		
		TDS: ¿Cuántos documentos componen el sistema?			

18	Métrica	Preguntas		Resp.	Aplicación de la métrica
	DEC	TDC: ¿Cuántos documentos tienen el máximo nivel de confidencialidad?	TDS: ¿Cuántos documentos componen el sistema?		

19	Métrica	Preguntas		Resp.	Aplicación de la métrica
	CIL	MCI: ¿Cuántos miembros del equipo conoce las implicaciones legales?	ME: ¿Cuántos miembros tiene el equipo?		

Métricas de consistencia (utilizar el grafo de la métrica IAD):

20	Métrica	Preguntas		Resp.	Aplicación de la métrica
	PRI	TRI: ¿Cuántas relaciones del grafo son incorrectas o innecesarias?	TRG: ¿Cuántas relaciones o aristas tiene el grafo?		

21	Métrica	Preguntas		Resp.	Aplicación de la métrica
	DRD	TAG: ¿Cuántas aristas tiene el grafo sin tener en cuenta su ponderación?	TNG: ¿Cuántos nodos tiene el grafo?		

Métricas de precisión:

22	Métrica	Preguntas					Aplicación de la métrica
	PPD	P45	P46	P47	P48	P49	

23	Métrica	Preguntas		Resp.	Aplicación de la métrica
	RMT	NRC: ¿Cuántos requisitos se han cubierto en la matriz de trazabilidad o similares?	NTR: ¿Cuántos requisitos componen el sistema?		

Métricas de portabilidad:

24	Métrica	Preguntas				Aplicación de la métrica
	PPRD	P50	P51	P52	P53	

25	Métrica	Preguntas	Resp.	Aplicación de la métrica
	DIP	TAFI: ¿Cuántos artefactos están construidos en un formato independiente? TAS: ¿Cuántos artefactos componen el sistema?		

26	Métrica	Preguntas	Resp.	Aplicación de la métrica
	DCE	TAD: ¿Cuántos artefactos requieren de una configuración especial o dependen de un S.O. en particular? TAS: ¿Cuántos artefactos componen el sistema?		

Métricas de integridad:

27	Métrica	Preguntas				Resp.	Aplicación de la métrica
	PID	P54	P55	P56	P57		

28	Métrica	Preguntas	Resp.	Aplicación de la métrica
	PAAS	TARs: ¿Cuántos archivos de seguimiento existen? TAS: ¿Cuántos artefactos componen el sistema?		

Métricas de actualidad:

29	Métrica	Preguntas				Resp.	Aplicación de la métrica
	PACD	P58	P59	P60	P61		

30	Métrica	Preguntas	Resp.	Aplicación de la métrica
	PARev	TAC: ¿Cuántos artefactos coinciden con la versión del sistema? TAS: ¿Cuántos artefactos componen el sistema?		

31	Métrica	Preguntas	Resp.	Aplicación de la métrica
	PIRU	PIR: ¿Cuántos problemas relacionados con la documentación han sido reportados? TPR: ¿Cuántos problemas en general han sido reportados?		

Métricas de trazabilidad:

32	Métrica	Preguntas	Resp.	Aplicación de la métrica
	CDT	RSR: ¿Cuántos requisitos no se relacionan con otros elementos en la matriz o similares? NTR: ¿Cuántos requisitos componen el sistema?		

33	Métrica	Preguntas	Resp.	Aplicación de la métrica	
	NDD	GS: ¿Cuál es la suma de los grados de salida del grafo?			
		PAS: ¿Cuál es la suma de los pesos de las aristas de salida?			
		GE: ¿Cuál es la suma de los grados de entrada del grafo?			
		PAE: ¿Cuál es la suma de los pesos de las aristas de entrada?			
n: ¿Cuántos nodos tiene el grafo?					

34	Métrica	Preguntas	Resp.	Aplicación de la métrica	
	NAA	GS: ¿Cuál es el grado de salida del grafo?			
		GE: ¿Cuál es grado de entrada del grafo?			
		n: ¿Cuántos nodos tiene el grafo?			

Métricas de expansibilidad:

35	Métrica	Preguntas					Resp.	Aplicación de la métrica
	PED	P16	P17	P18	P19	P20		

36	Métrica	Preguntas	Resp.	Aplicación de la métrica
	PUP	UPD: Para este valor utilizar la métrica 2 (UPD) de la característica de uniformidad n: ¿Cuántos artefactos componen el sistema?		

Métricas de validez:

37	Métrica	Preguntas	Resp.	Aplicación de la métrica
----	---------	-----------	-------	--------------------------

	PAE	TDAp: ¿Cuántos artefactos han sido aprobados por la organización?		
		TAS: ¿Cuántos artefactos componen el sistema?		

	Métrica	Preguntas	Resp.	Aplicación de la métrica
38	ARA	SNC: ¿Cuántas secciones del historial de revisiones coinciden con las secciones propuestas en la métrica?		
		n: ¿Cuántos artefactos componen el sistema?		

Métricas de verificabilidad:

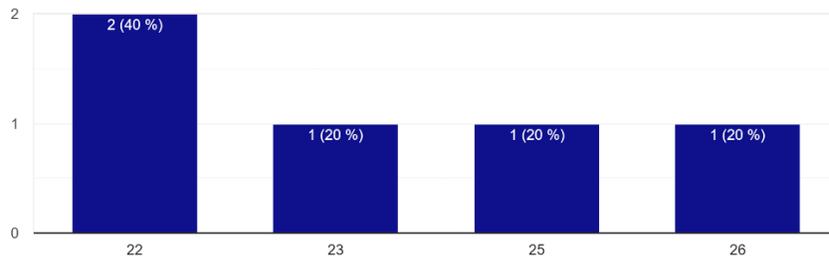
	Métrica	Preguntas	Resp.	Aplicación de la métrica
39	DHR	DAH: ¿Cuántos artefactos tienen asociado un documento de historial de revisiones?		
		n: ¿Cuántos artefactos componen el sistema?		

Anexo C: Encuesta de evaluación

Caracterización de los participantes:

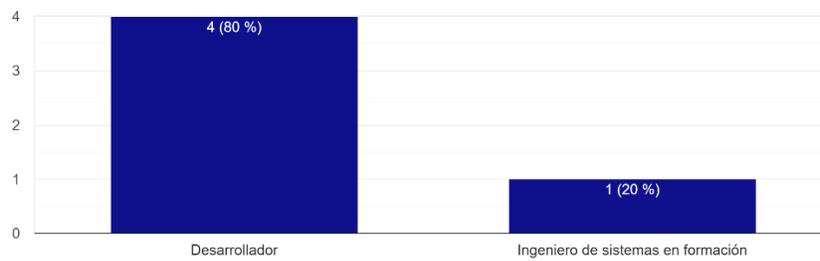
Edad

5 respuestas



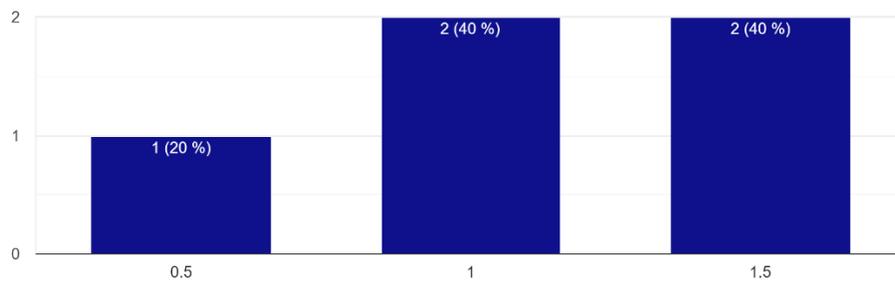
Cargo en la organización

5 respuestas



Años de experiencia

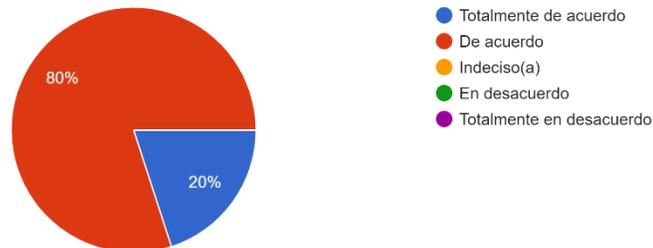
5 respuestas



Evaluación de la claridad de la propuesta

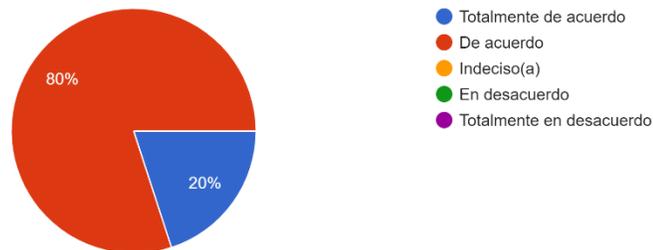
P1: ¿Considera que las dimensiones de la documentación desarrolladas en el modelo propuesto son claras?

5 respuestas



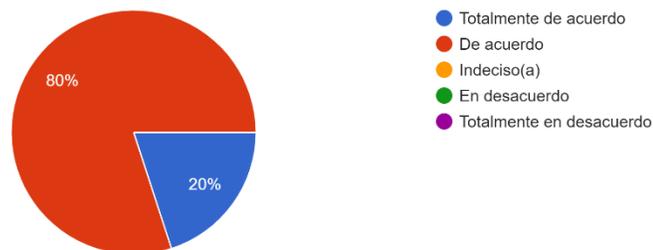
P2: ¿Considera que las características de la documentación desarrolladas en el modelo propuesto son claras?

5 respuestas



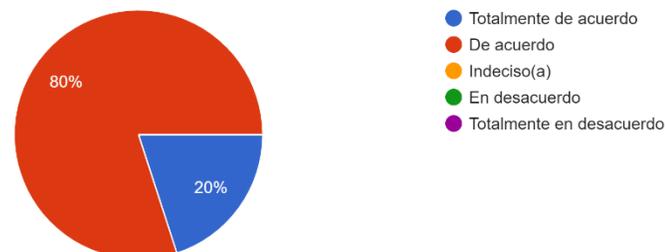
P3: ¿Considera que los riesgos que pueden ocasionar deuda en la documentación de software desarrollados en el modelo propuesto son claros?

5 respuestas



P4: ¿Considera que las métricas definidas en el modelo propuesto son claras?

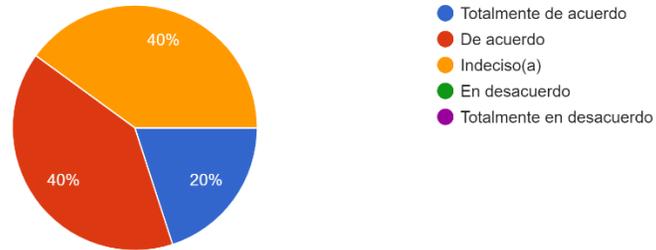
5 respuestas



Evaluación de la aplicabilidad de la propuesta

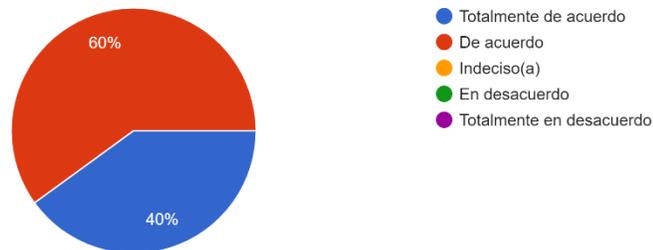
P5: ¿Considera que las métricas definidas en el modelo propuesto tienen suficiente rigor matemático?

5 respuestas



P6: De acuerdo con su experiencia: ¿Considera que la forma como se evalúan las características de la documentación de software es adecuada y permite identificar aspectos de la deuda de la documentación?

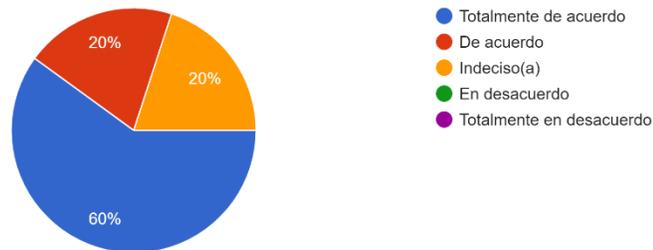
5 respuestas



Evaluación de la idoneidad de la propuesta

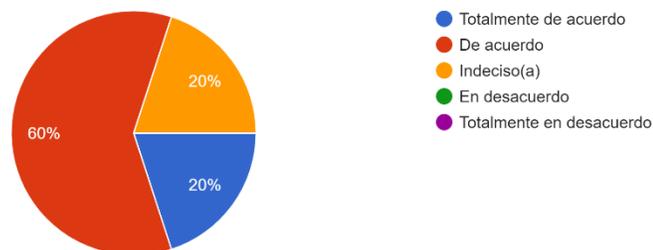
P7: ¿Considera que las métricas propuestas para medir la característica de uniformidad de la documentación son adecuadas?

5 respuestas



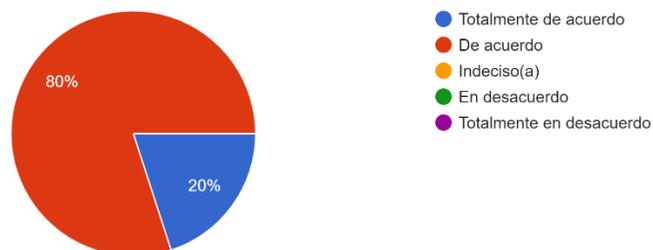
P8: ¿Considera que las métricas propuestas para medir la característica de adaptabilidad de la documentación son adecuadas?

5 respuestas



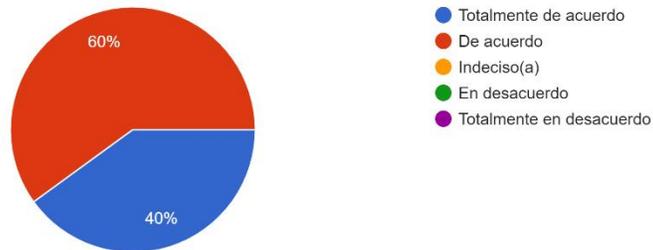
P9: ¿Considera que las métricas propuestas para medir la característica de modularidad de la documentación son adecuadas?

5 respuestas



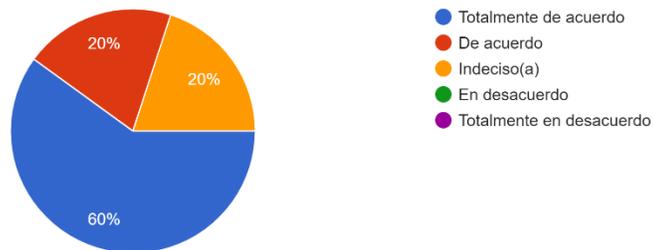
P10: ¿Considera que las métricas propuestas para medir la característica de legibilidad de la documentación son adecuadas?

5 respuestas



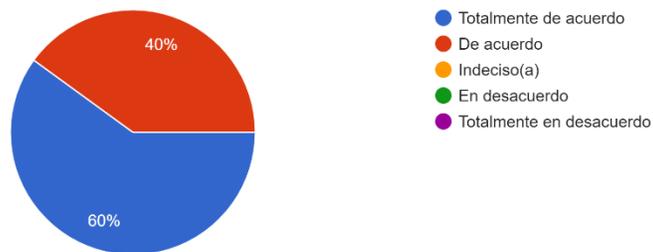
P11: ¿Considera que las métricas propuestas para medir la característica de concisión de la documentación son adecuadas?

5 respuestas



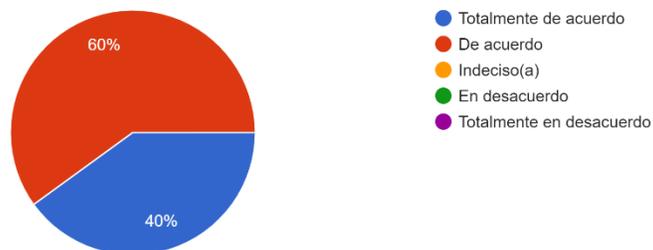
P12: ¿Considera que las métricas propuestas para medir la característica de claridad de la documentación son adecuadas?

5 respuestas



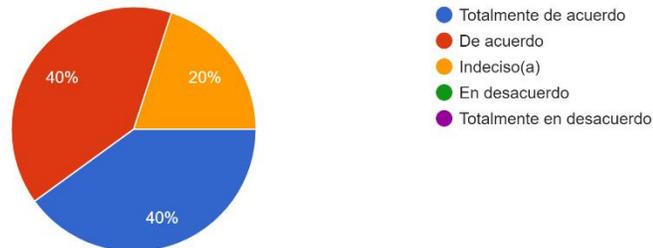
P13: ¿Considera que las métricas propuestas para medir la característica de accesibilidad de la documentación son adecuadas?

5 respuestas



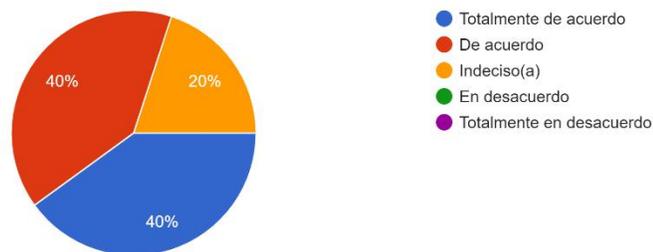
P14: ¿Considera que las métricas propuestas para medir la característica de confidencialidad de la documentación son adecuadas?

5 respuestas



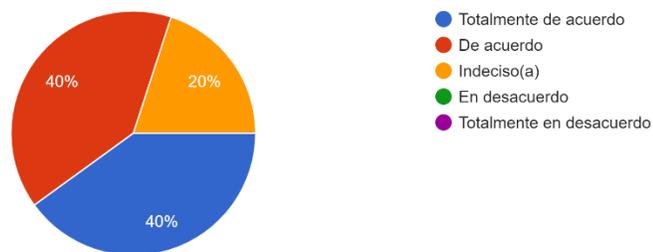
P15: ¿Considera que las métricas propuestas para medir la característica de consistencia de la documentación son adecuadas?

5 respuestas



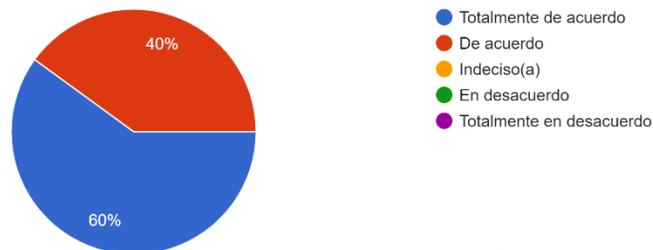
P16: ¿Considera que las métricas propuestas para medir la característica de precisión de la documentación son adecuadas?

5 respuestas



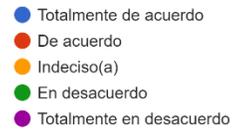
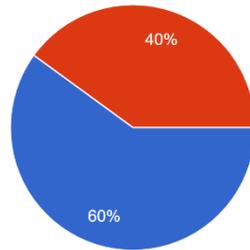
P17: ¿Considera que las métricas propuestas para medir la característica de portabilidad de la documentación son adecuadas?

5 respuestas



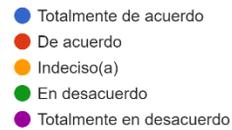
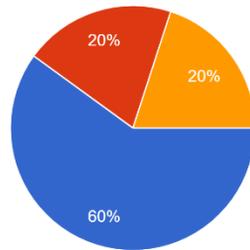
P18: ¿Considera que las métricas propuestas para medir la característica de integridad de la documentación son adecuadas?

5 respuestas



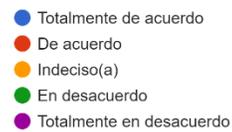
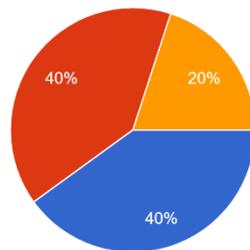
P19: ¿Considera que las métricas propuestas para medir la característica de actualidad de la documentación son adecuadas?

5 respuestas



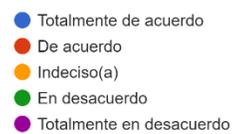
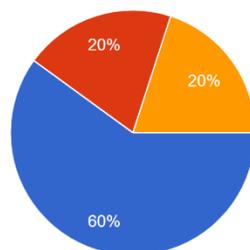
P20: ¿Considera que las métricas propuestas para medir la característica de trazabilidad de la documentación son adecuadas?

5 respuestas



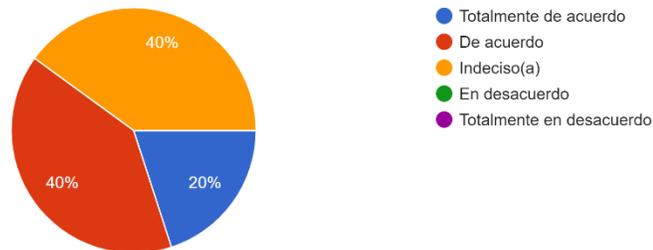
P21: ¿Considera que las métricas propuestas para medir la característica de expansibilidad de la documentación son adecuadas?

5 respuestas



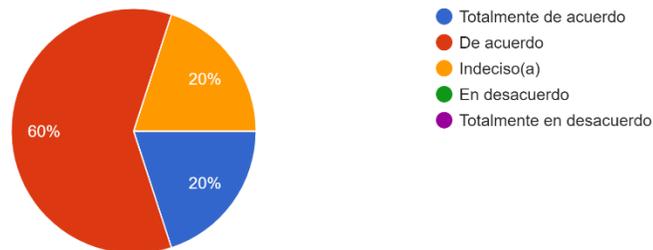
P22: ¿Considera que las métricas propuestas para medir la característica de validez de la documentación son adecuadas?

5 respuestas



P23: ¿Considera que las métricas propuestas para medir la característica de verificabilidad de la documentación son adecuadas?

5 respuestas



P24: ¿Considera que las métricas propuestas en el modelo son suficientes para garantizar una evaluación completa de la deuda de la documentación de software?

5 respuestas

