

**MARCO DE TRABAJO PARA LA CAPTURA Y ESPECIFICACIÓN DE LA
FIABILIDAD DE UN PRODUCTO SOFTWARE**



CARLOS ANDRÉS ORDOÑEZ RENGIFO

OMAR RICARDO BONILLA HERNÁNDEZ

Universidad del Cauca

Facultad de Ingeniería Electrónica y Telecomunicaciones

Popayán

2022

**MARCO DE TRABAJO PARA LA CAPTURA Y ESPECIFICACIÓN DE LA
FIABILIDAD DE UN PRODUCTO SOFTWARE**

CARLOS ANDRÉS ORDOÑEZ RENGIFO

OMAR RICARDO BONILLA HERNÁNDEZ

Trabajo de grado presentado a la Facultad de Ingeniería

Electrónica y Telecomunicaciones de la

Universidad del Cauca para obtención del

Título de:

Ingeniero de sistemas

Director:

PhD. Sandra Lorena Buitrón Ruíz.

Codirector:

PhD. Francisco José Pino Correa.

Popayán

2022

TABLA DE CONTENIDO

Capítulo 1 – Introducción.....	1
1.1 Planteamiento del problema y justificación	1
1.2 Objetivos	3
Objetivo general	3
Objetivos específicos.....	3
1.3 Estrategia de investigación	3
Ciclo Conceptual: Análisis conceptual.	3
Ciclo metodológico: Definición del marco de trabajo	3
Ciclo de Evaluación: Evaluación del marco de trabajo.....	4
Ciclo de documentación y socialización	4
1.4 Organización del documento	4
Capítulo 2 – Marco teórico y estado del arte.	6
2.1 Marco teórico	6
Calidad de producto software.....	6
Características de calidad del producto software	7
Fiabilidad.....	7
ISO 33000-4	8
2.2 Estado del arte.....	8
2.2.1 Protocolo de revisión de la información	8
2.2.1.1 Búsqueda de información primaria.....	8
2.2.1.2 Búsqueda de información relevante	9
2.2.2 Análisis a partir de la información seleccionada.....	10
2.2.2.1 Propuestas sobre asegurar la fiabilidad de productos software.	10

2.2.2.2 Propuestas sobre definiciones de fiabilidad y sus sub características.	12
2.2.3 Aporte.....	13
Capítulo 3 – Propuesta para el marco de trabajo para la captura y especificación de requisitos de la fiabilidad.....	14
3.1 Definición propuesta taxonómica de fiabilidad.....	14
3.1.1 Análisis de las sub-características de fiabilidad de acuerdo con la literatura	15
3.1.2 Análisis de nuevos aspectos relacionados a Fiabilidad de acuerdo con la literatura consultada.	20
3.1.3 Definición taxonómica para las sub-características de fiabilidad.....	24
3.1.3.1 Madurez.....	26
3.1.3.2 Capacidad de recuperación.....	30
3.1.3.3 Disponibilidad	34
3.1.3.4 Tolerancia a fallos	38
3.1.4 Análisis de las definiciones de fiabilidad de acuerdo con la literatura	41
3.1.5 Definición taxonómica para la característica de fiabilidad.....	47
3.1.6 Extensión de la definición de fiabilidad.....	48
3.2 Proceso de referencia para captura y especificación de requisitos de fiabilidad.....	51
3.2.1 Definición de artefactos para captura y especificación de requisitos de fiabilidad	51
3.2.1.1 Presentación de la propuesta	52
3.2.1.2 Guía para la aplicación del proceso de referencia para capturar y especificar requisitos de fiabilidad	52
3.2.1.3 Preguntas para la especificación de requisitos de fiabilidad	53
3.2.1.4 Formato de modelado de proceso de negocio adaptado.....	56
3.2.1.5 Formato de prototipado para RNF	57
3.2.1.6 Familia iconográfica de fiabilidad extendida	58
3.2.2 Definición del proceso de referencia	60

3.2.2.1 Definición la actividad para contextualizar sobre la taxonomía fiabilidad.....	61
3.2.2.2 Definición del subproceso para capturar requisitos de fiabilidad	63
3.2.2.3 Definición del subproceso para especificar requisitos de fiabilidad	64
3.2.2.4 Definición de la actividad de analizar requisitos de fiabilidad	65
3.2.2.5 Definición de roles a intervenir en proceso de referencia	61
Capítulo 4 – Evaluación de marco de trabajo para capturar y especificar de requisitos de fiabilidad.	67
4.1 Diseño del estudio de caso.....	67
4.2. Intervención	70
4.3 Resultados.....	73
4.3.1 Resultado del uso de la plantilla “Proceso de negocio originación del proyecto Solucredit”	73
4.3.2 Resultado del uso de la plantilla “Prototipos de RNF proceso de negocio originación del proyecto Solucredit”	74
4.3.3 Resultado del uso de la plantilla “Proceso de negocio autorización de proyecto Solucredit”	77
4.3.4 Resultado del uso de la plantilla “Prototipos de RNF proceso de negocio autorización del proyecto Solucredit”	78
5.3.5 Resultado encuesta analista de requisitos	81
5.3.6 Resultado encuesta cliente	82
4.4 Análisis de resultado.....	84
4.4.1 Análisis de resultados desde la perspectiva de requisitos no funcionales obtenidos y especificados	84
4.4.2 Análisis de resultados desde la perspectiva del uso de las propiedades definidas en la taxonomía.....	86
4.4.3 Análisis de resultados desde la perspectiva de medidas definidas en el estudio de caso	87

4.4.4 Respuestas a las preguntas de investigación.....	88
4.4.5 Puntos de mejora.....	90
5.4.6 Análisis de validez de la propuesta.....	90
5.4.7 Limitaciones.....	91
Capítulo 5 – Conclusiones, lecciones aprendidas y trabajos futuros.	92
5.1 Conclusiones.....	92
5.2 Lecciones aprendidas.....	93
5.3 Trabajos futuros.....	94
Referencias	95

LISTA DE TABLAS

Tabla 1. Definiciones de sub-características de fiabilidad.	15
Tabla 2. Criterios de clasificación para definiciones y sub-características de fiabilidad.	17
Tabla 3. Comparativa sub-características de fiabilidad (ISO/IEC 25010 Vs. Literatura consultada).	18
Tabla 4. Resultados comparativa sub-características de fiabilidad.	20
Tabla 5. Nuevos aspectos relacionados a la fiabilidad	21
Tabla 6. Criterios de clasificación para nuevos aspectos de fiabilidad.	22
Tabla 7. Comparativa sub-características de fiabilidad Vs. Nuevas sub-características de fiabilidad	23
Tabla 8. Definiciones de fiabilidad de acuerdo con la literatura.....	41
Tabla 9. Comparativo definiciones de fiabilidad	44
Tabla 10. Resultados comparativa definiciones de fiabilidad	45
Tabla 11. Preguntas para capturar y especificar requisitos de fiabilidad	54
Tabla 12. Roles proceso de referencia.....	61
Tabla 13. Preguntas de investigación estudio de caso.....	67
Tabla 14. Sujetos de investigación.....	69
Tabla 15. Métricas estudio de caso	69
Tabla 16. Actividades para aplicación de estudio de caso	70
Tabla 17. Unidad de análisis para estudio de caso	71
Tabla 18. Escala medición para evaluar aspectos	81
Tabla 19. Respuesta analista de requisitos sobre aspectos.....	81
Tabla 20. Escala de interpretación	82
Tabla 21. Respuesta a pregunta 2 cliente	82
Tabla 22. Respuesta cliente sobre aspectos.....	83
Tabla 23. Respuesta a pregunta 2 cliente	83

LISTA DE ILUSTRACIONES

Ilustración 1. Marco de trabajo para la captura y especificación de requisitos de fiabilidad	14
Ilustración 2. Jerarquía taxonomía de fiabilidad.....	25
Ilustración 3. Descomposición semántica de madurez.	27
Ilustración 4. Descomposición semántica de estabilidad	27
Ilustración 5. Descomposición semántica de robustez	27
Ilustración 6. Descomposición semántica de consistencia	27
Ilustración 7. Esquema de elementos relacionados de madurez.....	28
Ilustración 8. Taxonomía de Madurez	30
Ilustración 9. Descomposición semántica de capacidad de recuperación	30
Ilustración 10. <i>Descomposición semántica de resiliencia</i>	31
Ilustración 11. Esquema de elementos relacionados de capacidad de recuperación.....	32
Ilustración 12. Taxonomía de Capacidad de Recuperación.....	34
Ilustración 13. Descomposición semántica de disponibilidad	35
Ilustración 14. Descomposición semántica Longevidad	35
Ilustración 15. Esquema de elementos relacionados de Disponibilidad	36
Ilustración 16. Taxonomía de Disponibilidad	37
Ilustración 17. Descomposición semántica tolerancia a fallos.....	39
Ilustración 18. Descomposición semántica cobertura	39
Ilustración 19. Esquema de elementos relacionados a Tolerancia a fallos.....	40
Ilustración 20. Taxonomía de Tolerancia a Fallos	41
Ilustración 21. Fiabilidad de acuerdo con ISO/IEC 25010.....	47
Ilustración 22. Taxonomía característica de fiabilidad	48
Ilustración 23. Guía para la aplicación del Proceso de referencia para capturar y especificar requisitos de fiabilidad	52
Ilustración 24. Formato de modelado de proceso de negocio	56
Ilustración 25. Formato de modelado de proceso de negocio adaptado	57
Ilustración 26. Formato de prototipado para RNF	58
Ilustración 27. Familia Iconográfica de fiabilidad	59
Ilustración 28. Familia iconográfica de fiabilidad extendida.....	60
Ilustración 29. Subproceso para capturar requisitos de fiabilidad	63
Ilustración 30. Subproceso para especificar requisitos de fiabilidad	64
Ilustración 31. Proceso de referencia para captura y especificación de requisitos de fiabilidad	66

Ilustración 32. Evidencia ejecución estudio de caso	72
Ilustración 33. Evidencia ejecución estudio de caso	72
Ilustración 34. Proceso de negocio originación - Formato de modelado de proceso de negocio adaptado	74
Ilustración 35. Resultado proceso de negocio originación - Formato de prototipado para RNF	75
Ilustración 36. Resultado proceso de negocio originación - Formato de prototipado para RNF	75
Ilustración 37. Resultado proceso de negocio originación - Formato de prototipado para RNF	76
Ilustración 38. Resultado proceso de negocio originación - Formato de prototipado para RNF	76
Ilustración 39. Resultado proceso de negocio originación - Formato de prototipado para RNF	77
Ilustración 40. Proceso de negocio autorización - Formato de modelado de proceso de negocio adaptado	78
Ilustración 41. Resultado proceso de negocio autorización - Formato de prototipado para RNF.....	79
Ilustración 42. Resultado proceso de negocio autorización - Formato de prototipado para RNF.....	79
Ilustración 43. Resultado proceso de negocio autorización - Formato de prototipado para RNF.....	80
Ilustración 44. Resultado proceso de negocio autorización - Formato de prototipado para RNF.....	80
Ilustración 45. Relación entre preguntas planteadas vs. Numero de requisitos capturados	85
Ilustración 46. Cantidad de requisitos por sub-característica.....	86
Ilustración 47. Cantidad de requerimientos por Propiedad	87

LISTAS DE ECUACIONES

Ecuación 1. Ecuación de disponibilidad 34

Capítulo 1 – Introducción

En la actualidad, las empresas dedicadas al desarrollo de software han empezado a darle mayor importancia a la forma en que se realiza el proceso de construcción de estos productos software. Dentro de los aspectos que se vienen considerando en esta reflexión empresarial se encuentran los elementos que pertenecen a la calidad del producto software, la cual está determinada, a través de modelos de calidad, por características y sub-características de calidad. En este capítulo se presenta el planteamiento del problema específico para este contexto, centrado en la característica de fiabilidad. Así mismo, se describen los argumentos tenidos en cuenta para la realización del presente trabajo, incluyendo la pregunta de investigación.

Además, se describen los objetivos planteados para el presente trabajo, su estrategia de investigación con sus actividades, la solución propuesta para cada capítulo y la organización del actual documento.

1.1 Planteamiento del problema y justificación

Los requisitos funcionales(RF) son la descripción técnica de qué servicios serán soportados u ofrecidos por el sistema; mientras que, los requisitos no funcionales (RNF) describen el comportamiento, aspecto o acciones de ejecución que realiza el sistema, pero que no están relacionadas como una función ofrecida al usuario para interactuar directamente, tales como: concurrencia de mil usuarios por segundo, cifrado de la información, mecanismos de recuperación ante fallas o facilidad de navegación [1]. Los RNF se desprenden de las características de calidad del producto software, las cuales son según la ISO/IEC 25010, las siguientes: eficiencia de desempeño, compatibilidad, usabilidad, fiabilidad, seguridad, mantenibilidad y portabilidad [2].

Entre todas las características de calidad mencionadas anteriormente, la fiabilidad ocupa un lugar relevante para trabajar [3], porque se busca tener software maduro que mantenga su funcionamiento y disponibilidad frente a fallas, ya que pueden causar pérdidas financieras y poner en riesgo la vida de las personas, principalmente en sistemas críticos [3-5]. La fiabilidad se define entonces, como la capacidad de un sistema o componente para desempeñar las funciones especificadas, cuando se usa bajo unas condiciones y periodo de tiempo determinados, a nivel de disponibilidad, madurez, tolerancia a fallos y capacidad de recuperación [2].

De otro lado, la característica de fiabilidad se considera fundamental para otras características, tales como la mantenibilidad y seguridad, quienes se relacionan estrechamente con esta característica [6]. Los RNF relacionados con la disponibilidad y tolerancia a fallos, afectan la forma en que los usuarios finales perciben al producto software, otros usuarios como gerentes de proyecto y líderes técnicos advierten que su ausencia hace propenso que los costos de mantenimiento sean más grandes de lo esperado; además afecta la productividad de los empleados que utilizan el producto como herramienta para agilizar sus labores en las empresas [3]. Adicionalmente, es importante asegurarse que el producto software tenga los mecanismos apropiados para protegerse frente ataques o modificaciones no autorizadas que coloquen en riesgo su disponibilidad, ya que no tener mecanismos que implementen la seguridad apropiadamente afecta directamente la disponibilidad, como una de las sub-características principales de la fiabilidad [7].

La fiabilidad es una capacidad esencial de los sistemas software, ya que en la actualidad está integrada en una amplia variedad de sistemas críticos como redes, dispositivos médicos y aviones, que se ejecutan en entornos donde los eventos ocurren al azar y, por lo tanto, son propensos a fallas del sistema [6]. Pese a que en la literatura se evidencia la importancia de esta característica, también se observa que no está definida de manera detallada; por lo tanto su definición y las sub-características que la componen tienden a ser ambiguas y poco claras [8]. Del mismo modo se evidencia que no existen un método para la captura y especificación de esta característica en proyectos, lo cual permitiría asegurar que es considerada desde etapas tempranas de desarrollo de software.

En este sentido, teniendo en cuenta la relevancia planteada frente a la característica de fiabilidad en los productos de software, es pertinente trabajar en estructurar, profundizar y ampliar la definición de esta característica, elementos conceptuales (taxonomía de fiabilidad) y metodológicos (proceso de referencia) relacionados con la misma, a través de un marco de trabajo¹ que apoye la captura y especificación de RNF de fiabilidad desde etapas tempranas del desarrollo de software. Por lo tanto, la pregunta de investigación de este trabajo de grado es ¿Cómo apoyar la captura y especificación de requisitos asociados a la fiabilidad del producto software?

¹ Marco de trabajo: se puede definir como una red de conceptos interrelacionados que juntos proporcionan una comprensión integral de un fenómeno [9] Y. Jabareen, "Building a conceptual framework: philosophy, definitions, and procedure," *International journal of qualitative methods*, vol. 8, pp. 49-62, 2009..

1.2 Objetivos

Objetivo general

Proponer un marco de trabajo que establezca elementos conceptuales y metodológicos que apoyen la captura y especificación de requisitos asociados a la fiabilidad del producto software.

Objetivos específicos

- Identificar aspectos de calidad que describan la fiabilidad de un producto software, por medio de una revisión de la literatura.
- Definir una estructura conceptual de la fiabilidad en productos software que permita clasificar los aspectos previamente identificados y sus relaciones.
- Diseñar un proceso de referencia que apoye la captura y especificación en etapas tempranas de requisitos de fiabilidad en productos software.
- Evaluar de manera preliminar el marco de trabajo utilizando un método de investigación que conlleve su aplicación al contexto real.

1.3 Estrategia de investigación

Para llevar a cabo el presente trabajo de investigación se ejecutaron las actividades presentadas en la metodología de investigación-acción multi-ciclo con bifurcación [10], comprendidas en los siguientes ciclos de investigación: conceptual, metodológico, de evaluación y de documentación.

Ciclo Conceptual: Análisis conceptual.

- **Estudio de la literatura sobre fiabilidad del producto software:** se reconocieron desde la literatura los diferentes aspectos que hacen referencia a la fiabilidad entendido como atributo de calidad en productos software.
- **Identificación componentes de un proceso de referencia:** fueron revisadas y analizadas las características que conforman un proceso de referencia.

Ciclo metodológico: Definición del marco de trabajo

- **Identificación de aspectos de un marco de trabajo:** Se identificaron las características que pertenecen a un marco de trabajo y analizar la forma de armonizarlos con los elementos hallados en el ciclo conceptual.

- **Definición del marco de trabajo:** Se diseñó y estructuró el marco de trabajo de manera que se integren los elementos: (a) estructura conceptual de la fiabilidad y (b) proceso de referencia para la captura y especificación de requisitos de fiabilidad de un producto software.

Ciclo de Evaluación: Evaluación del marco de trabajo

Se realizó la evaluación del marco de trabajo mediante un estudio de caso compuesto por 5 pasos, basados en [11]:

- **Diseño del estudio de caso:** Se definen los objetivos y se planifica el estudio de caso.
- **Preparación para recolección de datos:** Se definen los procedimientos y protocolos para la recolección de datos.
- **Recolección de evidencias:** Ejecución del estudio de caso y recolección de datos.
- **Análisis de colección de datos:** Se analizan los datos recolectados durante la ejecución.
- **Reporte:** Se comunican los resultados del estudio de caso.

Ciclo de documentación y socialización

Se realizó de manera transversal al desarrollo de trabajo de grado con el objetivo principal de organizar toda la documentación obtenida en los ciclos preliminares dándoles una estructura dentro de la monografía, de tal manera que se den las conclusiones del trabajo para posteriormente lograr publicar los resultados a través de un artículo.

- **Elaboración de la monografía:** redactar la monografía.
- **Elaborar un artículo para la divulgación de los resultados:** elaborar artículo para publicar los resultados.
- **Divulgar y sustentar los resultados:** divulgar los resultados del trabajo de grado y sustentarlos.

1.4 Organización del documento

En esta sección se describen cómo se encuentra organizada la monografía y los objetivos de los capítulos que la componen.

Capítulo 2: Presenta el estado del arte relacionado al tema de investigación de la característica de fiabilidad, producto de los resultados obtenidos a partir de la revisión de la literatura. Además, presenta el marco teórico relacionado a calidad de producto software, ISO/IEC 25010, fiabilidad e ISO 33000-4.

Capítulo 3: Presenta la propuesta para el marco de trabajo para la captura y especificación de requisitos de la fiabilidad, compuesta por un elemento conceptual (taxonomía de fiabilidad) y un elemento metodológico (proceso de referencia).

Capítulo 4: Presenta el diseño y los resultados de un estudio de caso, a través del cual fue evaluado de forma preliminar el marco de trabajo para la captura y especificación de requisitos de fiabilidad.

Capítulo 5: Presenta las conclusiones, lecciones aprendidas y trabajos futuros obtenidos durante el planteamiento y aplicación del marco de trabajo para la captura y especificación de requisitos de fiabilidad.

Capítulo 2 – Marco teórico y estado del arte

En este capítulo se presenta el marco teórico el cual está compuesto por los conceptos relacionados con el tema de estudio; así como el estado del arte en el cual se describe el protocolo de revisión de la literatura, la selección de la información y el análisis realizado sobre esta información. Por último, se especifica el aporte propuesto en el trabajo.

2.1 Marco teórico

Requisitos: Necesidad o expectativa establecida, generalmente implícita u obligatoria [12].

Tipos de requisitos: Los requisitos se dividen en dos tipos: (i) Requisito funcional: Es una especificación formal de un requerimiento enfocado en las funcionalidades del proceso de negocio[1]. (ii) Requisito no funcional: Es un atributo de, o una restricción del sistema. Un requisito de calidad específico es un requisito que pertenece a un problema de calidad que no sea la calidad de cumplimiento de un requisito funcional, una restricción es un requisito que restringe el espacio de la solución más allá de lo necesario para cumplir con la calidad funcional, de rendimiento y requerimientos de calidad específica [13].

Requisito no funcional de fiabilidad: Es una representación formal que establece el comportamiento correcto de las funcionalidades que debe soportar el sistema, producto o componente, frente a situaciones o condiciones específicas, en periodos de tiempos determinados [2].

Calidad de producto software

Las organizaciones no solo se interesan por la calidad de los procesos implementados en el desarrollo de software, sino también por la calidad de los productos que se realizan y adquieren, debido a que estos pueden afectar negativamente factores clave que impactan directamente el éxito de un proyecto[14]. Por tal motivo, actualmente se cuenta con la familia de normas ISO/IEC 25000 sustituyendo a las anteriores ISO/IEC 9126 e ISO/IEC 14598, las cuales tienen como objetivo la creación de un marco de trabajo común para evaluar la calidad del producto software [15]. Esta familia se encuentra compuesta por seis divisiones, una de ellas llamada modelo de calidad ISO/IEC 2501n en la cual se encuentra la norma ISO/IEC 25010 en donde se define la calidad del

producto software, como el grado en que dicho producto satisface los requisitos de sus usuarios, implementando un conjunto de características de calidad que se pueden evaluar en un producto software[2].

Características de calidad del producto software

Para las características de los productos software existen diferentes modelos con el objetivo de realizar su evaluación, como el desarrollado por McCall en 1977, Evan y Marciniak en 1987, Deusth y Willis en 1988, los dos anteriores son modelos alternativos al de McCall, basados igualmente en la identificación de factores. En cualquier caso, todos los modelos de calidad buscan definir las características, para medir si el software está cumpliendo cuantitativamente todos los criterios[16]. Los requisitos de desempeño o también llamados requisitos no funcionales describen las condiciones bajo las cuales una solución debe seguir siendo eficaz o las cualidades que debe tener la solución[17].

Fiabilidad

Según la ISO/IEC 25010 la fiabilidad se define como la capacidad de un sistema o componente para desempeñar las funciones especificadas, cuando se usa bajo unas condiciones y periodo de tiempo determinados[2]. Esta característica se divide a su vez en las siguientes características:

- **Madurez.** Capacidad del sistema o producto que satisface las necesidades de fiabilidad en condiciones normales de operación.
- **Disponibilidad.** Capacidad del sistema o producto operativo y accesible para su uso cuando se requiere.
- **Tolerancia a fallos.** Capacidad del sistema o producto para operar según lo previsto en presencia de fallos hardware o software.
- **Capacidad de recuperación.** Capacidad del sistema o producto software para recuperar los datos afectados y restablecer el estado deseado del sistema en caso de interrupción o fallo.

Del mismo modo la fiabilidad depende del número de defectos reportados, su densidad y su tasa de llegada[18]. muchas veces definida matemáticamente como la probabilidad de una operación libre de errores[17] relacionándose entre el tiempo de prueba y el número acumulativo de fallas detectadas por la prueba[19], garantizando así que las aplicaciones nunca fallen en las

circunstancias más exigentes[20]. También la fiabilidad y la tolerancia a fallos es un requisito clave, especialmente en aplicaciones críticas y en la seguridad de estas[6, 21].

ISO 33000-4

La finalidad de esta norma es establecer los requerimientos para la construcción y revisión de modelos de referencia de procesos, modelos de evaluación, y modelos de madurez[16], esto quiere decir un conjunto de procesos que colectivamente puedan alcanzar su propósito[22]. La norma establece que los procesos deben tener una única descripción e identificación, además de cumplir con las siguientes reglas:

- a) Un proceso debe ser descrito en términos de su propósito y salidas de proceso.
- b) En cualquier descripción del proceso el conjunto de salidas del proceso debe ser el necesario y suficiente para alcanzar el propósito de este.
- c) Las descripciones de procesos no deberían contener o implicar aspectos de características de calidad del proceso, más allá de los más bajos niveles de su escala de medición esperada.

2.2 Estado del arte

A continuación, se presenta el estado del arte del tema de investigación, el protocolo de revisión de la información, protocolo de selección de la información, el análisis realizado a partir de la información seleccionada y el aporte propuesto en el trabajo.

2.2.1 Protocolo de revisión de la información

En la siguiente sección se describirá el proceso que se siguió para la revisión de la literatura, este se realizó con el propósito de definir el estado del arte actual de la literatura frente a las propuestas del atributo de calidad conocido como fiabilidad, y sus respectivas características como: tolerancia a fallos, madurez, recuperabilidad y disponibilidad.

2.2.1.1 Búsqueda de información primaria

Para la definición del estado del arte de la fiabilidad como atributo de calidad en productos software, se hizo uso de un proceso para la revisión de la literatura, este proceso se dividió en cuatro etapas según [23]. Estas etapas son:

- Definición de las cadenas de búsqueda.

- Ejecución de cadenas de búsqueda.
- Filtración de resultados por criterios de inclusión.
- Filtración de resultados por criterios de exclusión.

En la primera etapa se establecieron dos cadenas de búsqueda, las cuales relacionaban palabras claves como la característica en cuestión y las sub-características relacionados, estas cadenas son:

- KEY (“reliability” AND “software product” AND “quality”) AND (LIMIT-TO (SUBJAREA, "COMP") OR LIMIT-TO (SUBJAREA, "ENGI"))
- TITLE-ABS-KEY ("reliability" AND "software product" AND ("maturity" OR "availability" OR "fault tolerance" OR "recoverability")) AND (LIMIT-TO (SUBJAREA,"COMP") OR LIMIT-TO (SUBJAREA,"ENGI"))

2.2.1.2 Búsqueda de información relevante

En la segunda etapa, se procedió a ejecutar las cadenas de búsqueda establecidas, obteniendo un volumen de resultados de 96 artículos encontrados para la primera cadena y 65 artículos encontrados para la segunda cadena.

En la tercera etapa, se procedió a aplicar criterios de inclusión para filtrar los artículos obtenidos por medio de las cadenas de búsqueda. Estos criterios de inclusión son:

- El título o las palabras claves deberán incluir la característica de fiabilidad o alguna de sus sub-características.
- El resumen del artículo deberá incluir información relevante que indique la presencia de información de la característica de fiabilidad en su contenido.

Después de aplicar los criterios de inclusión se obtuvo 37 artículos relevantes para la primera cadena y 12 artículos relevantes para la segunda cadena. Quedando en total 49 artículos relevantes.

En la cuarta etapa, se aplicó un criterio de exclusión para los 49 artículos relevantes. El criterio de exclusión es:

- No se tendrán en cuenta los artículos que, en la introducción, contexto, propuesta, análisis de resultados o conclusiones no incluya definiciones, detalles o relaciones de valor sobre la característica de fiabilidad o alguna de sus sub-características.

Después de aplicar el criterio de exclusión se obtuvieron 21 artículos primarios, los cuales serán la base que tomaremos para realizar nuestro estudio entorno a la característica de fiabilidad en productos software.

2.2.2 Análisis a partir de la información seleccionada

2.2.2.1 Propuestas sobre asegurar la fiabilidad de productos software.

Realizando un estudio de los artículos primarios obtenidos, se encontró que 33,34% de las propuestas abordadas tienen como enfoque predecir la fiabilidad de productos software durante la etapa de desarrollo. La predicción de la fiabilidad comprende un punto de interés para algunas organizaciones, las cuales buscan calcular el grado de fiabilidad que tienen sus productos software; para realizar este cálculo de predicción se han generado diferentes modelos y métricas.

En primer lugar tenemos los Modelos de Crecimiento de Fiabilidad del Software (Software Reliability Growth Models (SRGM)), estos modelos se dividen en dos tipos: el primero predice la fiabilidad a partir del diseño de parámetros como líneas de código; el segundo predice la fiabilidad basándose en datos de defectos recolectados en determinados rangos de tiempo [3]. Pese a que se han desarrollado aproximadamente 200 modelos desde principios de la década de 1970, la mayoría de las modelos SRGM no pueden mantener el mismo grado de fiabilidad entre versiones, ya que agregar nuevas funcionalidades puede significar un aumento relevante en las fallas del sistema y por lo tanto afectar la fiabilidad del mismo [24]. Además, cabe resaltar que los modelos propuestos de predecir la fiabilidad en entornos de producción son escasos, aun sabiendo sobre la creciente demanda de nuestra sociedad frente a los diferentes sistemas impulsados por software [8].

En segundo lugar tenemos métricas que hacen uso de datos como: la cantidad de defectos encontrados, defectos reparados y el grado de cobertura del código [25]. Una métrica recurrente en los estudios abordados es usada para determinar la disponibilidad de los productos software, esta métrica se define por la siguiente fórmula: $\text{Disponibilidad} = \text{MTTF} / \text{MTTF} + \text{MTTR}$. Donde MTTF es el tiempo medio para medir el tiempo de falla del software, MTBF es el tiempo promedio

entre el tiempo de falla del sistema y cuándo ocurrirá la próxima vez y MTTR es el tiempo medio requerido para reparar la falla ocurrida en el sistema [26].

Por otro lado, el 38,09% de las propuestas aborda mecanismos para asegurar que la fiabilidad está presente desde la etapa de diseño del producto software, esto se debe a que una de las principales causas de la ausencia de esta característica son las malas prácticas durante el diseño de la arquitectura del software, evidenciando así, la falta de investigación en la fiabilidad en etapas tempranas del desarrollo. Desde esta perspectiva se proponen métodos que establecen parámetros para que la arquitectura del producto software aplique conceptos que permitan tener una menor tasa de incidencia a fallos a fin de garantizar la supervivencia del sistema, y de esta forma asegurar una menor afectación en la estabilidad del software y mantener su disponibilidad [27].

Adicionalmente, según la literatura las medidas relacionadas a la fiabilidad son beneficiosas para el software en producción, porque no solo rastrean el rendimiento a lo largo del tiempo, sino que también pueden ser requeridas para garantizar que el software cumpla o supere los requisitos de rendimiento [17]. A pesar de la existencia de un gran número de medidas de software, la mayoría de ellas no tienen éxito (Kokol y Brest, 1997) debido a una serie de debilidades: por ejemplo, las medidas de software suelen definirse de manera informal (Alikacem y Sahraoui, 2006). , incompleta y / o inexacta (Dawson y Nolan, 2003) y, por lo tanto, tales medidas no producen la información requerida [28].

Pese a que los modelos y métricas anteriormente expuestos han demostrado resultados confiables en torno a asegurar la fiabilidad de productos software, cabe resaltar que están enfocados a ser utilizados mayormente durante las etapas de desarrollo y producción [25]. Se ha observado en algunos estudios que desplazar la importancia de verificar la fiabilidad del producto únicamente a las pruebas no es suficiente, menos en una industria donde actualmente las actividades de pruebas son afectadas por recorte en sus tiempos, como consecuencia de las exigencias de entregar productos que agreguen valor al cliente en el menor tiempo posible aunque se sacrifiquen aspectos de calidad [25].

Lo anterior evidencia la falta de investigación en las etapas de elicitación y especificación de requisitos de fiabilidad, que aporten así acciones concretas que aseguren productos que aumenten

la confianza del cliente al momento de su uso y también disminuya la incidencia en comportamientos no esperados.

Tomando en cuenta que existe una necesidad sobre crear mecanismos para elicitar y especificar requisitos de fiabilidad, el primer paso a dar es ampliar el concepto de fiabilidad, por lo tanto, se hizo un sondeo en la literatura consultada sobre las definiciones de fiabilidad y las sub-características que la componen, de las cuales se hablarán en la siguiente sección.

2.2.2.2 Propuestas sobre definiciones de fiabilidad y sus sub características.

En el 23,8% de las propuestas se aborda la relación de la fiabilidad con otras características de calidad de software como seguridad y mantenibilidad, compartiendo regularmente la definición dada por la ISO/IEC 25010. Un estudio explicaba la relación que existe entre la característica de disponibilidad y el atributo de seguridad, ya que implementar mecanismos de seguridad que protejan al sistema de futuros ataques, ayuda certeramente a mantener la disponibilidad del producto [7]. Además, se encontró que la mantenibilidad va relacionada con la sub-característica de longevidad, debido a que el software es más fácil de mantener, se modifica fácilmente y, por lo tanto, es más probable que se mantenga en lugar de desecharlo[17].

Para determinar qué volumen de propuestas incluían las sub-características de fiabilidad especificadas en la ISO/IEC 25010 o algunas sub-características nuevas, se realizó un análisis de las propuestas revisadas encontrando lo siguiente: La fiabilidad como atributo de calidad en productos software se ha definido principalmente en torno a la sub-característica de tolerancia a fallos, teniendo que el 85.7% de la literatura consultada habla sobre esta sub-característica. Frente a otras sub-características, se encontró que la sub-característica de disponibilidad se abordó en el 47,6%; la sub-característica de recuperabilidad en un 9,5% y la sub-característica de madurez en un 9,5%. En una menor medida encontramos que el 4,84% de las propuestas abordan de forma directa la definición de la fiabilidad como característica de calidad de software. Además, se encontró que el 23,8% de la literatura consultada incluye sub-características no presentes en la definición de fiabilidad para la ISO/IEC 25010, como, por ejemplo: resiliencia, consistencia, estabilidad, robustez y longevidad [17, 29], las cuales ayudan a esclarecer y ampliar la visión en torno a los aspectos que definen la fiabilidad en productos software.

Ampliando los puntos anteriores, se encontró que la sub-característica de madurez se encuentra relacionada con nuevos términos como consistencia, estabilidad, robustez y longevidad; así mismo se encontró que la sub-característica de recuperabilidad se define como parte de la disponibilidad de un producto software[8].

La sub-característica de tolerancia a fallos se encuentra fuertemente apoyada con otras características de calidad como la modularidad [21], adicionalmente la tolerancia a fallos busca asegurar la fiabilidad implementando técnicas de detección de errores para operar y funcionar de forma esperada. Mientras la sub-característica de disponibilidad tiene como finalidad estar accesible y operativo en el tiempo determinado.

Se puede afirmar que la fiabilidad es fundamental, especialmente en aplicaciones críticas debido a las configuraciones cambiantes en el tiempo de ejecución por ejemplo la realización de líneas de productos de software dinámico (DSPL). DSPL constituye un área de investigación emergente pero prometedora. Sin embargo, asegurar la fiabilidad en los DSPL sigue siendo insuficientemente explorado [6].

2.2.3 Aporte

Con lo anterior, se evidencia que, aunque existen estudios en torno a asegurar la fiabilidad en productos software, se hace necesario proponer un marco de trabajo que considere un análisis conceptual con respecto a elementos involucrados con la característica de fiabilidad, y un proceso de referencia que considere definiciones, elementos conceptuales, entre otros que apoye la captura y especificación de esta característica, dado que este tipo de aportes metodológicos no ha sido identificado en la revisión de la literatura realizada.

Por lo tanto, nuestro aporte va dirigido a:

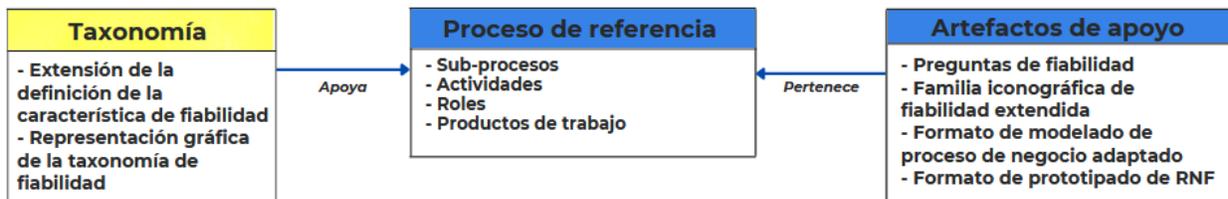
- Fortalecer la definición de fiabilidad como atributo de calidad en productos software.
- Fortalecer las características relacionadas con la fiabilidad como atributo de calidad en productos software.
- Proponer un marco de trabajo que permita identificar los aspectos que afectan la fiabilidad, y apoye la captura y especificación de requisitos asociados a la fiabilidad del producto software.

Capítulo 3 – Propuesta para el marco de trabajo para la captura y especificación de requisitos de la fiabilidad

En este capítulo se presenta la propuesta del marco de trabajo para la captura y especificación de requisitos de la fiabilidad, el cual está compuesto por dos elementos uno conceptual y uno metodológico. Al elemento conceptual se le ha denominado taxonomía de fiabilidad, el cual busca extender el concepto de la característica de fiabilidad a partir de la información de la literatura y de la norma ISO/IEC 25010. Esta taxonomía apoya al elemento metodológico del marco de trabajo al cual se le ha denominado proceso de referencia para la captura y especificación de requisitos de fiabilidad.

Esta relación entre la taxonomía y el proceso de referencia se puede ver en la siguiente ilustración:

Ilustración 1. Marco de trabajo para la captura y especificación de requisitos de fiabilidad



La propuesta se plantea como una alternativa de solución en un contexto de desarrollo de software en el cual se cuenta con unas especificaciones funcionales (historias de usuario) junto con sus prototipos para poder ser aplicada.

A continuación, se describen estos elementos conceptuales y metodológicos del marco de trabajo.

3.1 Definición propuesta taxonómica de fiabilidad

En esta sección se presenta las actividades realizadas para lograr una definición conceptual extendida acerca de la característica de fiabilidad. Las actividades que permitieron obtener como resultado una taxonomía y una definición ampliada de fiabilidad fueron: (i) Análisis de las sub-características de fiabilidad de acuerdo con la literatura (ii) Extensión de la definición para cada sub-características de fiabilidad (iii) Análisis de las definiciones de fiabilidad de acuerdo con la literatura (iv) Extensión de la definición de fiabilidad (v) Propuesta taxonómica del atributo de fiabilidad.

3.1.1 Análisis de las sub-características de fiabilidad de acuerdo con la literatura

Como primer paso, se realizó un proceso de análisis de los 21 artículos primarios obtenidos en el estado del arte después de realizar la revisión de la literatura, la cual permitió extraer propuestas sobre las sub-características de fiabilidad establecidas por las ISO/IEC 25010, como lo son tolerancia a fallos, disponibilidad, madurez y capacidad de recuperación. Dicha información se puede contemplar en la *Tabla 1*.

Tabla 1. Definiciones de sub-características de fiabilidad.

Artículo/Norma	Sub-característica	Definición
ISO/IEC 25010 [2]	<i>Madurez</i>	Capacidad del sistema para satisfacer las necesidades de fiabilidad en condiciones normales.
	<i>Tolerancia a fallos</i>	Capacidad del sistema o componente para operar según lo previsto en presencia de fallos hardware o software.
	<i>Disponibilidad</i>	Capacidad del sistema o componente de estar operativo y accesible para su uso cuando se requiere.
	<i>Capacidad de recuperación</i>	Capacidad del producto software para recuperar los datos directamente afectados y reestablecer el estado deseado del sistema en caso de interrupción o fallo.
The development and reliability analysis environment of fault-tolerance multiversion software [21]	<i>Tolerancia a fallos</i>	Es una condición necesaria para asegurar la fiabilidad y seguridad de los sistemas técnicos.
	<i>Modularidad</i>	Es la simplificación del diseño de un programa por medio de su división en módulos/partes y la definición de las interrelaciones de estos componentes. Un sistema distribuido apropiadamente por módulos tiene como ventajas la facilidad de modificar, testear y encontrar errores en el software. Aun así, posee desventajas, como aumento en el tiempo de ejecución de la aplicación, mayor ocupación de memoria y la complejidad de las conexiones.

The comprehensive study on software reliability [26] Software Fault Tolerance Forestalls Crashes: To Err Is Human; To Forgive Is Fault Tolerant [20]	Disponibilidad	Se define como <i>MTTF</i> (el tiempo medio de la falla del sistema) dividido la suma de <i>MTBF</i> (tiempo promedio entre el momento de la falla y la próxima vez que ocurre la falla) y <i>MTTR</i> (tiempo medio requerido para reparar la falla).
Application of diversity method for reliability of cloud computing [29]	Capacidad de recuperación	Es aquel (sistema) que detecta las fallas mediante pruebas de aceptación y se recupera mediante el remplazo de una versión libre de fallas, por medio de la diversidad.
Reliability prediction for fault-tolerant software architectures [30]	Tolerancia a fallos	Los mecanismos cubren las fallas en los sistemas software y previenen que provoquen fallas. (Se amplía en los mecanismos de tolerancia a fallos).
Comparing reliability and security: Concepts, requirements, and techniques [7]	Tolerancia a fallos	El objetivo de la tolerancia a fallos es prevenir fallas del sistema en presencia de errores. La tolerancia a fallos se logra mediante la implementación de técnicas de detección y recuperación de errores en el sistema. Aprueba de fallas significa que el sistema falla en un modo que deja los procesos y componentes del sistema en una condición segura. Por lo tanto, para que un sistema sea a prueba de fallas, se requiere una cobertura de detección de fallas casi perfecta para los mecanismos de seguridad.
Software Fault Tolerance Forestalls Crashes: To Err Is Human; To Forgive Is Fault Tolerant [20]	Tolerancia a fallos	El software tolerante a fallas puede ser medido en términos de la disponibilidad, que es una función de la fiabilidad. Un entorno tolerante a fallas que incorpore mantenimiento preventivo a bordo es fundamental para maximizar la fiabilidad de una nave espacial en una misión en el espacio exterior.
SAS® Data Analytic Development: Dimensions of Software Quality [17]	Disponibilidad	Se mide como el periodo de tiempo que el software es funcional.
Software reliability modeling based on ISO/IEC SQuaRE [8]	Madurez	Grado en el que un sistema satisface las necesidades de fiabilidad en condiciones normales de funcionamiento, esto quiere decir que cumple con las expectativas del usuario. La madurez está definida por la suma entre Estabilidad y Robustez.

		La fiabilidad es, por lo tanto, una determinada función de Disponibilidad y Madurez en la que la madurez resume la corrección del comportamiento que es consistente con las expectativas de las partes interesadas.
	Disponibilidad	Grado en el que un sistema, producto o componente es operativo y accesible cuando se requiere para su uso.
	Tolerancia a fallos	Grado en el que un sistema, producto o componente funciona según lo previsto a pesar de la presencia de fallas de hardware o software.
	Capacidad de recuperación	Grado en el que, en caso de una interrupción o falla, un producto o sistema puede recuperar los datos directamente afectados y restablecer el estado deseado del sistema.
IEEE 1633 - IEEE Recommended Practice on Software Reliability [31]	Tolerancia a fallos	El atributo de supervivencia de un sistema que le permite brindar el servicio requerido después de que las fallas se han manifestado por sí mismas dentro del sistema.

Después de obtener la información relacionada a las definiciones de las sub-características, se propuso la generación de una tabla de clasificación, con el fin de conocer el grado de relación que tienen las definiciones de las sub-características expuestas en la tabla anterior con las definiciones otorgadas por la ISO/IEC 25010. Dicha información se encontrará en la **Tabla 2**.

Tabla 2. *Criterios de clasificación para definiciones y sub-características de fiabilidad.*

Valor	Criterio
Completamente relacionado	La definición por comparar debe compartir todos o la mayoría de los aspectos establecidos en la definición dada por la ISO/IEC 25010.
Ampliamente relacionado	La definición por comparar debe compartir al menos la mitad de los aspectos mencionados en la definición dada por la ISO/IEC 25010.
Parcialmente relacionado	La definición por comparar debe compartir pocos o ningún aspecto con la definición dada por la ISO/IEC 25010.

No relacionado	No se encuentra definición en la literatura para comparar.
----------------	--

Posteriormente, para la realización de la comparativa se propuso la **Tabla 3**, la cual presenta en su primera columna todos los trabajos relacionados que contienen definiciones sobre las sub-características de fiabilidad; mientras la segunda columna se divide entre las sub-características de madurez, disponibilidad tolerancia a fallos y capacidad de recuperación. La forma utilizada para diligenciar los resultados de dicha tabla fue tomar la definición de la sub-característica encontrada en el trabajo relacionado, y compararla con la definición de la ISO/IEC 25010 utilizando la **Tabla 2** de clasificación. Al terminar dicho análisis, se procedió a colocar en la intersección del artículo y la sub-característica evaluada, el resultado obtenido de acuerdo con los criterios de clasificación. Después de realizar dicho proceso para cada una de las definiciones consignadas en la **Tabla 1**, se obtuvo la información que se puede observar en la **Tabla 3**.

Tabla 3. Comparativa sub-características de fiabilidad (ISO/IEC 25010 Vs. Literatura consultada).

Definiciones en la literatura	Definición que contiene sub-características ISO/IEC 25010			
	<i>Madurez</i>	<i>Disponibilidad</i>	<i>Tolerancia a fallos</i>	<i>Capacidad de recuperación</i>
[21]	<i>No relacionado</i>	<i>No relacionado</i>	<i>Parcialmente relacionado</i>	<i>No relacionado</i>
[24]	<i>No relacionado</i>	<i>No relacionado</i>	<i>Ampliamente relacionado</i>	<i>No relacionado</i>
[26]	<i>No relacionado</i>	<i>Parcialmente relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>
[29]	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>	<i>Ampliamente relacionado</i>
[3]	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>
[32]	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>
[1]	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>

[7]	<i>No relacionado</i>	<i>Completamente relacionado</i>	<i>Ampliamente relacionado</i>	<i>No relacionado</i>
[20]	<i>No relacionado</i>	<i>Parcialmente relacionado</i>	<i>Ampliamente relacionado</i>	<i>No relacionado</i>
[25]	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>
[33]	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>
[18]	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>
[17]	<i>No relacionado</i>	<i>Ampliamente relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>
[8]	<i>Ampliamente relacionado</i>	<i>Completamente relacionado</i>	<i>Completamente relacionado</i>	<i>Completamente relacionado</i>
[28]	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>
[4]	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>
[27]	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>
[24]	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>
[19]	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>
[6]	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>
[31]	<i>No relacionado</i>	<i>No relacionado</i>	<i>Ampliamente relacionado</i>	<i>No relacionado</i>

Con el fin de tener una visión más clara sobre los resultados obtenidos de la comparativa realizada en la **Tabla 3**, se agruparon los criterios establecidos en la tabla de clasificación y se calculó el porcentaje de concurrencia para cada una de las sub-características evaluadas. La información se puede observar en la **Tabla 4**.

Tabla 4. Resultados comparativa sub-características de fiabilidad.

	Madurez	Disponibilidad	Tolerancia a fallos	Capacidad de recuperación
Completamente relacionado	0%	14,29%	14,29%	4,76%
Ampliamente relacionado	4,77%	4,76%	19,04%	4,76%
Parcialmente relacionado	0%	4,76%	4,77%	0%
No relacionado	95,23%	76,19%	61,90%	90,48%

De acuerdo con la tabla anterior, se observó que los autores están trabajando mayormente con la sub-característica de tolerancias a fallos, seguidamente disponibilidad, en tercer lugar, la sub-característica de capacidad de recuperación y por último en menos medida están trabajando con la sub-característica de madurez. Es importante recalcar que tolerancia a fallos es la sub-característica que más porcentaje de definiciones tiene en una clasificación diferente a completamente relacionado, lo cual hace que tenga ventaja para encontrar definiciones que otorguen nuevos aspectos que puedan nutrir su definición. Aunque disponibilidad, capacidad de recuperación y madurez tienen un porcentaje menor en clasificaciones diferente a completamente relacionado, se espera que las pocas definiciones que comprenden la clasificación de amplia y parcialmente relacionado permitan realizar una adecuada extensión de sus definiciones.

3.1.2 Análisis de nuevos aspectos relacionados a Fiabilidad de acuerdo con la literatura consultada.

Si bien en la sección anterior nos enfocamos en las definiciones de sub-características de fiabilidad encontradas en la literatura consultada, también se realizó una profundización en otros aspectos relacionados a la fiabilidad que no tienen una relación directa con las sub-características de fiabilidad establecidas en la ISO/IEC 25010, pero que son abordados por los autores consultados. Dichos nuevos aspectos se encontrarán en la **Tabla 5** con sus respectivas definiciones.

Tabla 5. Nuevos aspectos relacionados a la fiabilidad

Artículo	Aspecto	Definición
The development and reliability analysis environment of fault-tolerance multi-version software [21]	Modularidad	Es la simplificación del diseño de un programa por medio de su división en módulos/partes y la definición de las interrelaciones de estos componentes. Un sistema distribuido apropiadamente por módulos tiene como ventajas la facilidad de modificar, testear y encontrar errores en el software. Aun así, posee desventajas, como aumento en el tiempo de ejecución de la aplicación, mayor ocupación de memoria y la complejidad de las conexiones.
Application of diversity method for reliability of cloud computing [29]	Resiliencia	Capacidad para recuperarse o adaptarse fácilmente a la falla o al cambio. Es la capacidad de un sistema para responder a las fallas de forma transparente y automática. Que otorguen la capacidad de reconfigurarse a sí mismos y soporten su propia evolución.
Comparing reliability and security: Concepts, requirements, and techniques [7]	Confianza	La capacidad de evitar fallas en el servicio que son más frecuentes y graves de lo que es aceptable; es decir, las fallas que ocurren están dentro de los límites especificados por los requisitos. La confianza está definida por los atributos de fiabilidad, mantenibilidad, disponibilidad, integridad y confidencialidad.
	Cobertura	La cobertura se define como una medida de la capacidad del sistema para realizar la detección de fallas, la localización de fallas, la contención de fallas y / o la recuperación de fallas. Una definición más específica o estrecha de cobertura es como una medida de la capacidad del sistema para recuperarse automáticamente dentro de la duración diseñada después de la ocurrencia de una falla, tolerando así la falla.
SAS® Data Analytic Development: Dimensions of Software Quality [17]	Consistencia	El comportamiento inconsistente puede comprometer o eliminar el valor del producto, el software debe mantener la coherencia funcional en el sentido de que se produzca la salida o solución correcta cada vez.
	Longevidad	Tiempo de vida útil del software (años), dado que en este tiempo se deben percatar los cambios futuros que exigirán una respuesta eficiente y permanente del software tanto a nivel de procesamiento de

		datos (incremento de datos) como de ajuste al alcance de manera oportuna (mejoras, evoluciones, eliminaciones).
Software reliability modeling based on ISO/IEC SQuaRE [8]	Estabilidad	Grado en el cual el sistema no requiere cambios o correcciones.
	Robustez	Grado en el que un sistema o componente puede funcionar correctamente en presencia de entradas no validas o un entorno estresante.

Al tener los nuevos aspectos relacionados a la fiabilidad consignados en la tabla anterior, se hizo necesario realizar un análisis más profundo de cada definición, para así precisar a cuál de las sub-características de fiabilidad establecidas por la norma se encontraban relacionados cada uno de estos nuevos aspectos. Para dicho objetivo se propuso la siguiente tabla de clasificación, la cual nos ayudó a clasificar el grado de relación que tienen estos aspectos, con las sub-características establecidas por la norma. A continuación, se pueden observar los criterios de clasificación en la **Tabla 6**.

Tabla 6. *Criterios de clasificación para nuevos aspectos de fiabilidad.*

Valor	Criterio
Completamente relacionado	El nuevo aspecto tiene una relación estrecha en su definición con la sub-característica a comparar propuesta por la ISO/IEC 25010.
Ampliamente relacionado	El nuevo aspecto tiene una relación en al menos la mitad de los aspectos de su definición con la sub-característica a comparar propuesta por la ISO/IEC 25010.
Parcialmente relacionado	El nuevo aspecto tiene una relación con algunos aspectos en su definición con la sub-característica a comparar propuesta por la ISO/IEC 25010.
No relacionado	El nuevo aspecto no tiene ninguna relación con la sub-característica a comparar propuesta por la ISO/IEC 25010.

Continuando con el proceso, se propuso la **Tabla 7**, la cual en su primera columna lista cada uno de los nuevos aspectos encontrados en la literatura consultada, mientras de la segunda a la quinta columna se listarán las sub-características de fiabilidad establecidas por la ISO/IEC 25010. El

proceso constó en tomar cada definición de los nuevos aspectos, y compararlo con las definiciones de sub-características de la norma utilizando la **Tabla 6** de clasificación, esto con el fin de saber el grado de relación entre las definiciones. Los resultados obtenidos durante el análisis realizado para cada aspecto se pueden encontrar en la **Tabla 7**.

Tabla 7. Comparativa sub-características de fiabilidad Vs. Nuevas sub-características de fiabilidad

Aspectos relacionados a fiabilidad	Sub-características de fiabilidad			
	Madurez	Tolerancia a fallos	Disponibilidad	Capacidad de recuperación
Estabilidad	<i>Completamente relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>
Robustez	<i>Completamente relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>
Resiliencia	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>	<i>Completamente relacionado</i>
Cobertura	<i>No relacionado</i>	<i>Completamente relacionado</i>	<i>No relacionado</i>	<i>Parcialmente relacionado</i>
Longevidad	<i>No relacionado</i>	<i>No relacionado</i>	<i>Ampliamente relacionado</i>	<i>No relacionado</i>
Consistencia	<i>Ampliamente relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>	<i>No relacionado</i>

Se determinó con la tabla anterior que de acuerdo con la literatura los nuevos aspectos relacionados a fiabilidad comparten algunas similitudes con las sub-características de fiabilidad. Se encontró que la sub-característica de madurez se encuentra completa y ampliamente relacionada con los aspectos de estabilidad, robustez y consistencia, lo cual evidencia que, aunque fue la sub-característica con menor porcentaje de trabajos relacionados, dicho porcentaje trae consigo definiciones claves para la extensión de la sub-característica. La sub-características de capacidad

de recuperación, al igual que la sub-característica de madurez, pese a su poca incidencia en los trabajos consultados tuvo aspectos que lo relacionaban, como son cobertura y resiliencia. Por el contrario, aunque tolerancia a fallos tenía un alto porcentaje de trabajos relacionados, en esta ocasión se evidenció que solo tiene un aspecto completamente relacionado llamado cobertura. Y por último se determinó que la sub-característica de disponibilidad tiene un aspecto ampliamente relacionado llamado longevidad, dicho aspecto supone una ventana de oportunidad para ampliar la sub-característica con mayor precisión.

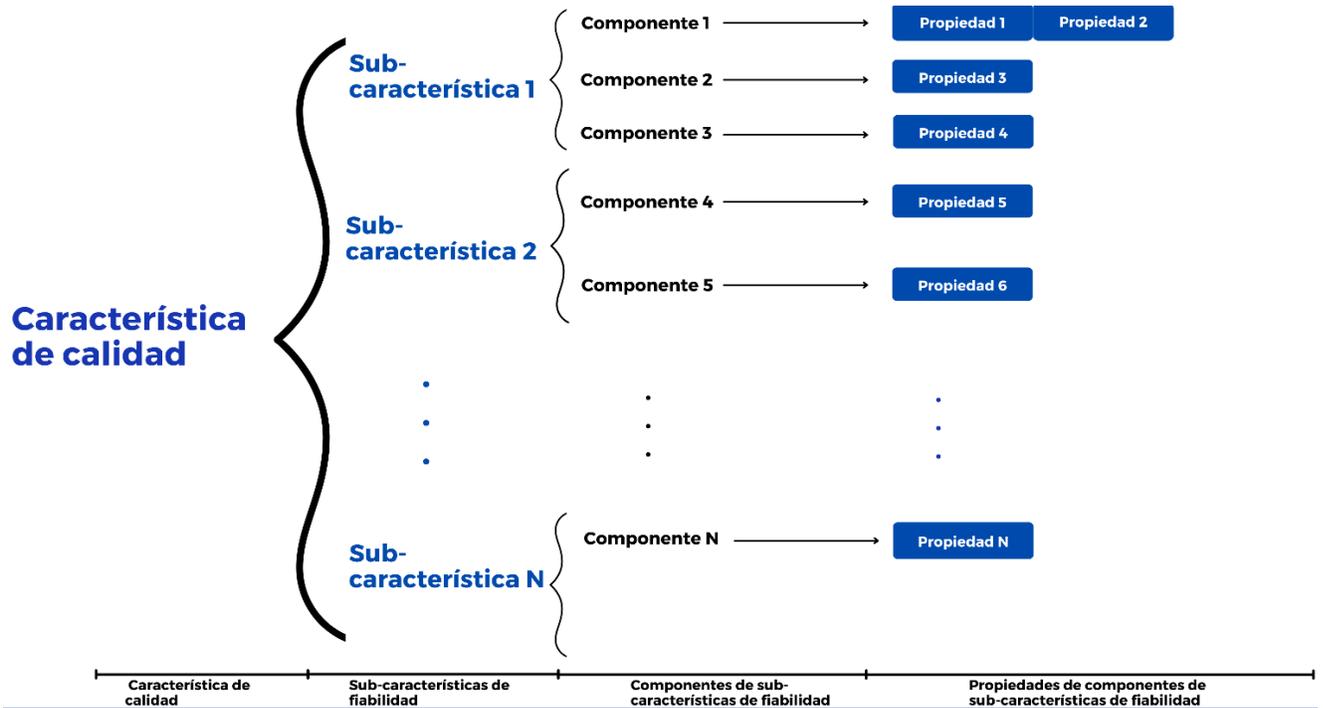
3.1.3 Definición taxonómica para las sub-características de fiabilidad

Para la definición de la taxonomía de fiabilidad, se estableció que estaría compuesta por 4 niveles, los cuales son:

- **Característica de calidad:** Es el primer nivel establecido por la ISO/IEC 25010, donde se ubica la característica de fiabilidad.
- **Sub-características de fiabilidad:** Es el segundo nivel establecido por la ISO/IEC 25010, donde se ubican las sub-características relacionadas a la fiabilidad, como: madurez, capacidad de recuperación, disponibilidad y tolerancia a fallas.
- **Componentes de sub-características de fiabilidad:** Es el tercer nivel establecido en la presente propuesta, donde se ubican los elementos que componen a la sub-característica de fiabilidad.
- **Propiedades de componentes de sub-características de fiabilidad:** Es el cuarto nivel establecido en la presente propuesta, donde se ubican los elementos que diferencian al componente de otros.

La organización de los niveles anteriormente descritos que componen la propuesta taxonómica de fiabilidad se puede observar en la *Ilustración 2*.

Ilustración 2. Jerarquía taxonomía de fiabilidad



A partir de los resultados obtenidos en la sección 3.1.1 y 3.1.2, frente a las sub-características, nuevos aspectos de fiabilidad y las relaciones entre estos, se procedió a realizar un análisis de las definiciones encontradas, esto con el fin de construir y proponer taxonomías para cada una de ellas que sirvan como insumo para la construcción de la propuesta taxonómica de la característica de fiabilidad. Para dicho objetivo se realizó el siguiente proceso:

- **Paso 1:** Se listarán las definiciones de la sub-característica.
- **Paso 2:** Se realizará una descomposición semántica de las definiciones relacionadas a la sub-característica, con el fin de identificar sus componentes.
- **Paso 3:** Se listarán las definiciones de los nuevos aspectos relacionados a cada sub-característica.
- **Paso 4:** Se realizará una descomposición semántica de las definiciones relacionadas a los nuevos aspectos.
- **Paso 5:** Se definen los elementos relacionados obtenidos del análisis realizado a cada sub-características y nuevos aspectos relacionados a esta.

Con la información obtenida en los pasos anteriores, se continua con el proceso para proponer la taxonomía de cada sub-características.

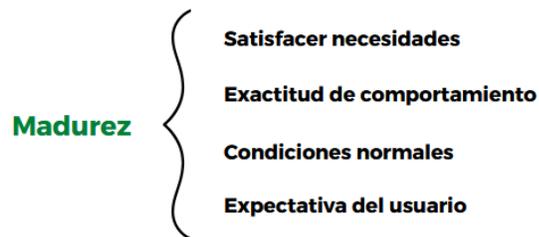
- **Paso 6:** En el nivel de sub-característica de fiabilidad, se ubicarán las sub-características mencionadas en la norma ISO/IEC 25010.
- **Paso 7:** En el nivel de componentes de sub-características de fiabilidad, solo se ubicarán elementos que agrupen dos o más propiedades que definan a la sub-característica. Además, se aplicará el criterio 1.
 - i. **Criterio 1:** En el nivel de propiedades de componentes de sub-características de fiabilidad, solo se ubicarán elementos que ayuden a diferenciar al componente de los demás componentes.
- **Paso 8:** Se creará un nuevo componente que agrupe elementos con un propósito en común, el cual se subdividirá en los elementos mencionados que pasarán a ser propiedades de este.
- **Paso 9:** Se comparan los elementos del segundo nivel de la estructura conceptual, para determinar si es un componente o propiedad de un componente. Para dicho objetivo, se aplicará el criterio 2.
 - i. **Criterio 2:** Los elementos del esquema conceptual, que no se adapten a la categorización de componente o propiedad, y además en su definición sean más grandes que la sub-característica en mención, no serán tomados en cuenta para la construcción de la taxonomía.

3.1.3.1 Madurez

Al realizar el análisis de madurez, se encontró que la mayoría de los trabajos consultados no ofrecían definiciones sobre esta sub-característica, teniendo que la única definición encontrada comparte similitud con la norma, pero introduce nuevos conceptos y relaciones.

De acuerdo con [8], la madurez es el grado en el que un sistema satisface las necesidades de fiabilidad en condiciones normales de funcionamiento, esto quiere decir que cumple con las expectativas del usuario. De igual manera, la madurez resume la exactitud del comportamiento que es consistente con las expectativas de las partes interesadas [8]. Para conocer más a detalle los componentes de la madurez, se realizó una descomposición semántica de la definición otorgada por la norma y la obtenida de la literatura, el resultado se puede ver en la **Ilustración 3**.

Ilustración 3. *Descomposición semántica de madurez.*



Entre los nuevos aspectos relacionados a la sub-característica de madurez tenemos estabilidad y robustez, la primera se define como el grado en el que un sistema o componente no requieren cambios o correcciones; mientras la segunda se define como el grado en el que un sistema o componente puede funcionar correctamente en presencia de entradas no válidas o en un entorno estresante [8]. A continuación, se pueden encontrar en la **Ilustración 4** la descomposición semántica de estabilidad, y en la **Ilustración 5** la descomposición semántica de robustez.

Ilustración 4. *Descomposición semántica de estabilidad*

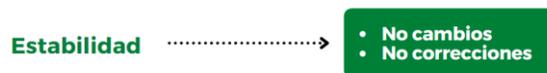


Ilustración 5. *Descomposición semántica de robustez*



En ese mismo sentido, otro nuevo aspecto relacionado a la madurez es la consistencia, la cual establece que la variabilidad es la antítesis de la fiabilidad, además el comportamiento inconsistente puede comprometer o eliminar el valor del producto, por lo tanto el software debe mantener la coherencia funcional en el sentido de que se produzca la salida o solución correcta cada vez que el software está en funcionamiento [17]. A continuación, se pueden encontrar el resultado de la descomposición semántica realizada a este nuevo aspecto en la **Ilustración 6**.

Ilustración 6. *Descomposición semántica de consistencia*



Como se puede notar, la invariabilidad y el comportamiento consistente son conceptos que aparecen implícitamente en las definiciones de madurez consultadas, por lo cual se deduce que en efecto la consistencia es un nuevo aspecto de la madurez para tener en consideración, tal como la estabilidad y robustez.

Teniendo en cuenta la descomposición semántica de las definiciones de madurez y las definiciones de los nuevos elementos relacionados con esta sub-característica, se propuso el siguiente esquema de elementos relacionados a la sub-característica de madurez, el cual se puede ver en la **Ilustración 7**.

Ilustración 7. Esquema de elementos relacionados de madurez



Tomando como base el esquema de elementos relacionados de madurez, se procedió a realizar la creación de la taxonomía de esta sub-característica.

Para definir los componentes de la sub-característica de madurez, se tomaron aquellos elementos relacionados que estuvieran compuestos por otros elementos. Por un lado, la estabilidad y robustez son citadas por [8] como elementos derivados de madurez, por lo tanto, se mantendrá su jerarquización. Debido a lo anterior, tanto estabilidad como robustez fueron ubicados en el nivel de componentes de madurez. En la misma vía, se observó que el elemento de consistencia presenta agrupación de varios elementos, esta particularidad hace que cumplan con el criterio 2, lo cual los ubica en el mismo nivel de robustez y estabilidad.

Frente a las propiedades que componen la estabilidad, se tomó los elementos de no cambios y no correcciones, y se transformaron en los elementos de sin cambios en la estructura y sin cambios en

el comportamiento, lo cual siguiendo con el criterio 1, estas propiedades logran el objetivo de diferenciar al componente de otros.

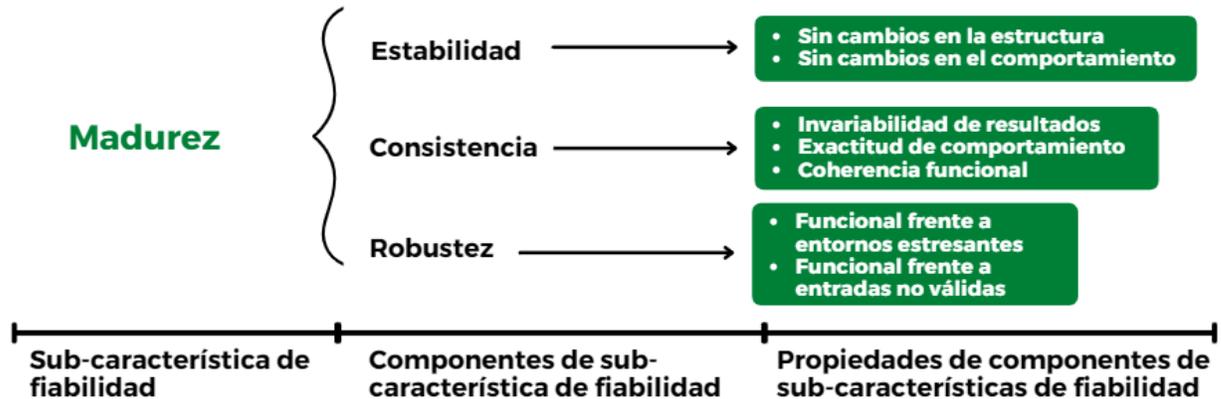
Frente a las propiedades de robustez, se tomaron los elementos de entorno estresante y entradas no válidas, y se adaptaron a los elementos de funcional frente a entornos estresantes y funcional frente a entradas no válidas. Estas propiedades ayudan a diferenciar de forma explícita al componente de robustez.

Frente a las propiedades de consistencia, se pudo notar que los elementos de invariabilidad, comportamiento consistente y salidas correctas comparten una definición parecida entre sí, por lo tanto, se unificaron como un nuevo elemento llamado invariabilidad de resultados ubicada en el nivel de propiedades de componentes de madurez, la cual se mantuvo relacionada al componente de consistencia. El elemento de exactitud de comportamiento tiene una definición muy específica y se alinea con el componente de consistencia, por lo pasó a ser una propiedad de este componente dentro de la taxonomía. Mientras el elemento de coherencia funcional se mantendrá relacionado a la consistencia como una propiedad.

Siguiendo con el análisis del esquema conceptual, se determinó que los elementos de satisfacer necesidades, condiciones normales, funcionamiento correcto, eliminar el valor del producto y expectativas del usuario son términos muy amplios en su enfoque, por lo cual, no es posible ubicarlos en ninguno de los niveles establecidos dentro de la taxonomía, cumpliendo con criterio 2 no fueran tomados en cuenta para ser parte de la taxonomía de madurez.

Aplicando los criterios establecidos en el esquema conceptual de madurez, y analizando detenidamente cada una de las decisiones tomadas, se presenta en la **Ilustración 8** la taxonomía de la sub-característica de madurez.

Ilustración 8. Taxonomía de Madurez



3.1.3.2 Capacidad de recuperación

Al realizar el análisis de capacidad de recuperación, se encontró que la mayoría de los trabajos consultados no ofrecían definiciones sobre esta sub-característica, teniendo que la única definición encontrada solo compartía algunas similitudes con la ofrecida por la norma.

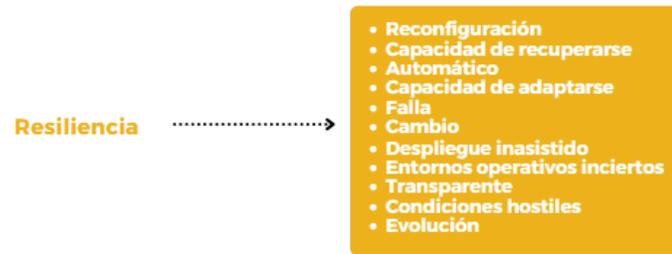
En [29] se define esta sub-característica como la capacidad de un sistema para detectar las fallas mediante pruebas de aceptación y recuperarse mediante el reemplazo de una versión libre de fallas. Con el fin de conocer los componentes de esta sub-característica, se procedió a realizar una descomposición semántica de las definiciones dadas por la norma y la literatura consultada, el resultado se puede ver en la *Ilustración 9*.

Ilustración 9. Descomposición semántica de capacidad de recuperación



En ese mismo sentido, el trabajo trae a colación un nuevo término relacionado a esta sub-característica, llamado resiliencia, la cual está definida como la habilidad para recuperarse o adaptarse fácilmente a la falla o al cambio. Es la capacidad de un sistema para responder a las fallas de forma transparente y automática. Que otorguen la capacidad de reconfigurarse a sí mismo y soporten su propia evolución [29]. A continuación, en la *Ilustración 10* se pueden encontrar el resultado de la descomposición semántica realizada a este nuevo aspecto.

Ilustración 10. Descomposición semántica de resiliencia



Otro de los nuevos aspectos relacionados a capacidad de recuperación, de acuerdo al análisis realizado en el sección 3.1.2, es la cobertura, la cual se define como la capacidad del sistema para realizar detección, localización, contención y/o recuperación de fallos [7]. Aunque en su definición este aspecto tiene en cuenta la capacidad del sistema de recuperarse ante la ocurrencia de una falla, se encuentra más enfocado a la sub-característica de tolerancia a fallos, por lo tanto, este aspecto no será tomado en cuenta para la definición de la taxonomía de capacidad de recuperación, pero si para la definición de la taxonomía de tolerancia a fallos.

Siguiendo con el proceso, como se puede observar en la descomposición semántica anterior, la resiliencia y la capacidad de recuperación comparten mucha similitud entre sus definiciones, además analizando el trabajo [27] se pudo constatar que el autor establece implícitamente los términos como sinónimos. Por lo anterior, se tomó la decisión de que la resiliencia en lugar de ser tratada como un aspecto perteneciente a la sub-característica de capacidad de recuperación o como una sub-característica diferente, se trató como sinónimo de esta última y las definiciones obtenidas de ambos conceptos se usaron para extender la definición presentada en la ISO/IEC 25010 en torno a la sub-característica en mención.

A continuación, en la **Ilustración 11** se puede observar el esquema de elementos relacionados, compuesto por los elementos obtenidos en la descomposición semántica de la sub-característica de capacidad de recuperación y los nuevos aspectos relacionados a esta.

Ilustración 11. Esquema de elementos relacionados de capacidad de recuperación



Por consiguiente, tomando como base el esquema de elementos relacionados de capacidad de recuperación, se procedió a realizar la creación de la propuesta taxonómica de esta sub-característica.

Como se puede apreciar, el esquema de elementos relacionados solo está compuesto por dos niveles, por lo cual los criterios se aplicarán únicamente a los elementos del segundo nivel. Iniciando con el análisis, el elemento de capacidad de adaptarse es definido por [29] como sinónimo de resiliencia y capacidad de recuperación, por lo tanto, no será incluido en la taxonomía.

Por otra parte, el elemento de restablecer estado deseado se enfoca en la capacidad del sistema para regresar a un estado estable que pueda brindar al usuario el funcionamiento esperado del sistema, este elemento es amplio y puede relacionarse con otros elementos como automático, transparencia y reconfiguración. En el caso del elemento de automático, se espera que dicha capacidad de restablecimiento se realice de manera automática por parte del sistema. En ese mismo sentido, se espera que este proceso sea claro para el usuario, de esta forma no inducir a confusiones, lo cual se aproxima al elemento de transparencia. Además, en caso de que el sistema se enfrente a situaciones no contempladas, tenga la capacidad de reconfigurarse así mismo con el objetivo de regresar al estado deseado. Por lo tanto, se tomó la decisión que restablecer estado deseado sea un componente compuesto por las propiedades de restablecimiento automático y restablecimiento transparente.

Al continuar con el análisis, se estableció que el elemento de recuperar datos afectados es muy importante para un sistema que tenga la sub-característica de capacidad de recuperación, esto

debido a que un sistema no solo debe recuperarse en términos de funcionalidad, sino también en términos de los datos, ya que estos son vitales para el funcionamiento completo del sistema. Recuperar datos afectados se refiere a que después de un fallo o algún evento que ponga en peligro los datos, el sistema tenga la capacidad de recuperarlos en la mayor parte posible, asegurando la persistencia, almacenamiento y procesamiento de estos. Dicho lo anterior, el elemento de recuperar datos afectados será un componente que se extenderá con las propiedades de asegurar persistencia de los datos y asegurar almacenamiento de los datos.

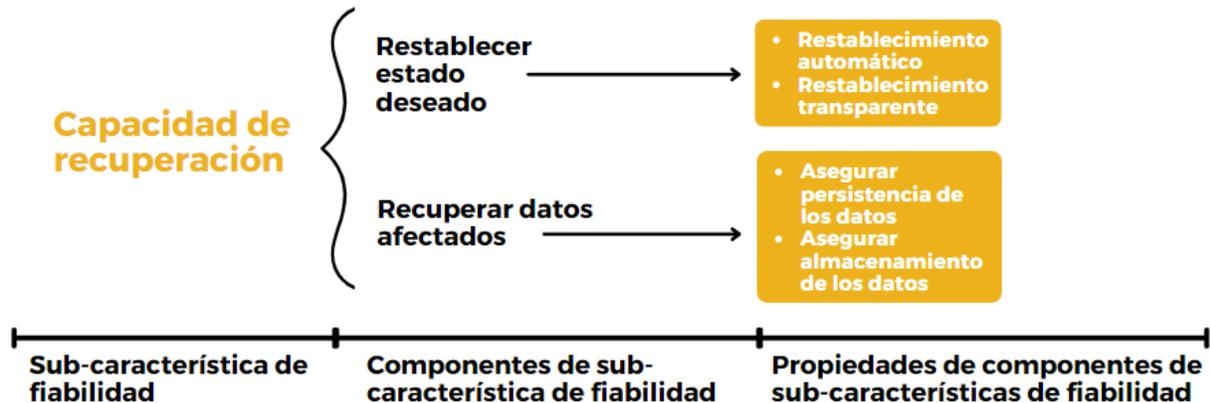
Los elementos de entornos operativos inciertos y condiciones hostiles, no se tomaron en cuenta, debido a que se enfocan más a la sub-característica de madurez, por lo tanto, no se incluyeron en la taxonomía de capacidad de recuperación.

Los elementos de detectar fallos y reemplazo de versión sin falla, aunque son importantes para esta sub-característica, no se tomaron en consideración ya que el primero es un elemento clave en la sub-característica de tolerancia a fallos, mientras el segundo es más una técnica para asegurar la recuperación del sistema que una capacidad en si de este.

Frente a los elementos que no se incluyeron en la taxonomía tenemos por una parte a evolución y cambio, los cuales cumpliendo con el criterio 2, no se tomaron en cuenta debido a que sus definiciones son más amplias que la sub-característica de capacidad de recuperación.

Aplicando los criterios establecidos en el esquema de elementos relacionados a capacidad de recuperación, y analizando detenidamente cada una de las decisiones tomadas, se presenta en la ***Ilustración 12*** la taxonomía de la sub-característica de capacidad de recuperación.

Ilustración 12. Taxonomía de Capacidad de Recuperación



3.1.3.3 Disponibilidad

Al realizar el análisis de la sub-característica de disponibilidad, se encontraron cinco trabajos que la abordan, dos de ellos se remiten a la definición dada por la ISO/IEC 25010, mientras los restantes otorgan fórmulas y otros conceptos que pueden ayudar en el mayor entendimiento de esta.

La primera definición encontrada en [17], define que la disponibilidad se mide como el periodo de tiempo que el software es funcional. Por otro lado, la segunda definición está dada por los trabajos [20, 26] que citan la *ecuación 1* como una forma de calcular la disponibilidad de los sistemas basándose en medidas obtenidas de los fallos de software y su respectivo tiempo de reparación.

Ecuación 1. Ecuación de disponibilidad

$$Disponibilidad = \frac{MTTF}{MTTF + MTTR}$$

Donde *MTTF* se define como el tiempo medio de la falla del sistema y *MTTR* como el tiempo medio requerido para reparar la falla. Esta ecuación busca medir el tiempo promedio en el cual un sistema se mantiene disponible en presencia de fallos [20, 26]. Aunque se comprende que las medidas MTTF y MTTR tienen una relación importante con la disponibilidad, su definición no está dentro de lo establecido en la norma ISO/IEC 25010. Por un lado, MTTF se enmarca en los fallos del sistema encontrados durante la etapa de pruebas, mientras MTTR se establece más como una medida obtenida de la capacidad para reparar la falla. Además, según [19], MTTF y MTTR son medidas que ayudan a definir la tolerancia a fallos en función de la disponibilidad, por lo tanto, se decidió no tomarlas en cuenta para el análisis de esta última sub-característica.

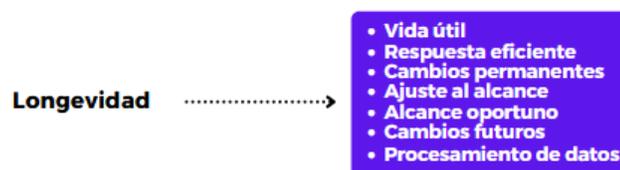
En [8], se establece que la tolerancia a fallos y la capacidad de recuperación son propiedades de la disponibilidad. La anterior afirmación es entendible, ya que un sistema tolerante a fallos aumenta evidentemente su capacidad de estar disponible en presencia de estas, y de igual forma, un sistema con la capacidad de recuperarse permite aumentar significativamente la disponibilidad del sistema al tener la habilidad de restablecerse en presencia de entornos estresantes o inciertos, lo cual pone en riesgo la disponibilidad de muchos sistemas. Aun así, nuestro trabajo se encuentra en el marco de lo establecido por la norma ISO/IEC 25010, la cual define que la tolerancia a fallos y capacidad de recuperación son pares de la disponibilidad, y aunque unos pueden influir en otros, se mantendrá su independencia. Con el fin de conocer los componentes de esta sub-característica, se procede a realizar una descomposición semántica de las definiciones dadas por la norma y la literatura consultada, el resultado se puede ver en la **Ilustración 13**.

Ilustración 13. Descomposición semántica de disponibilidad



Por otro lado, teniendo en cuenta el análisis realizado a los nuevos aspectos encontrados en la literatura, se concluyó que existe un nuevo aspecto que comparte alguna relación con la disponibilidad, este se conoce como longevidad. En [17] se define a la longevidad como el tiempo de vida útil del software (medido en semanas, meses o años), dado que en este tiempo se deben precatar los cambios futuros que exigirán una respuesta eficiente y permanente del software tanto a nivel de procesamiento de datos (o incremento de datos) como de ajuste al alcance de manera oportuna (mejoras, evoluciones o eliminaciones) [17]. Con el fin de conocer los componentes de este nuevo concepto, se procede a realizar una descomposición semántica, el resultado se puede ver en la **Ilustración 14**.

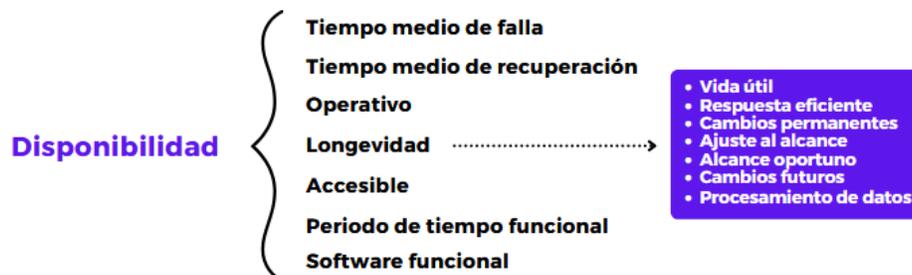
Ilustración 14. Descomposición semántica Longevidad



Después de tener las descomposiciones semánticas de disponibilidad y longevidad, se inició una comparativa de los elementos que componen ambas definiciones. Se pudo notar que la vida útil mencionada en la longevidad estaba estrechamente relacionada con los elementos encontrados en las definiciones de disponibilidad, como: periodo de tiempo funcional, software funcional, operativo y accesible. Lo anterior, debido a que se puede decir que la vida útil de un sistema está vinculada con la disponibilidad de un sistema, ya que esta define cual es el tiempo en el cual el sistema se va a mantener operativo a través del tiempo, ya sea en semanas, meses o años, esto incluye implícitamente la necesidad que el software se mantenga operativo, accesible y funcional en un periodo de tiempo específico [8, 17].

Habiendo determinado la relación que existe entre longevidad y disponibilidad, se propuso el siguiente esquema de elementos relacionados, compuesto por los elementos obtenidos en la descomposición semántica de la sub-característica de disponibilidad y los nuevos aspectos relacionados a esta, el cual se puede ver en la *Ilustración 15*.

Ilustración 15. Esquema de elementos relacionados de Disponibilidad



Después de obtener el anterior esquema de elementos relacionados de disponibilidad, se procedió a generar una propuesta taxonómica de la sub-característica de disponibilidad.

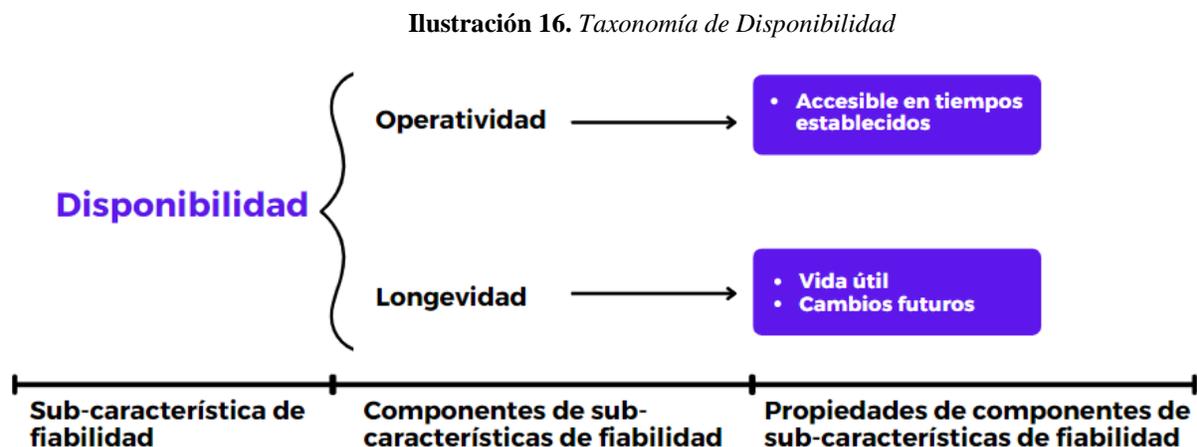
Iniciando con el análisis, el elemento de accesible se refiere a que los usuarios puedan acceder al sistema, y por consiguiente a las funcionalidades que ofrece. En este mismo sentido, la disponibilidad se mira relacionada a que el sistema se muestre funcional en todos sus aspectos, lo cual se relaciona con el elemento de software funcional. Por otro lado, existe un elemento que marca la pauta dentro de la disponibilidad, ya que se enfoca en los intervalos relacionado al tiempo en los que un sistema puede otorgarles acceso a las funcionalidades al usuario final, este elemento es periodo de tiempo funcional.

Siguiendo con el enfoque anterior, se analizó el elemento de operativo, el cual se entendió como la capacidad de un sistema de mantenerse funcional y accesible en un rango de tiempo determinado, lo cual evidentemente se relacionó con los primeros tres elementos descritos. De acuerdo con lo anterior, se tomó la decisión de cambiar la descripción del elemento de operativo por operatividad con el fin de relacionarlo a una capacidad del sistema, el cual esta deriva a la propiedad de accesible en tiempos establecidos.

El elemento de longevidad se subdivide en varios elementos, por lo tanto, se mantendrá en el nivel de componentes de sub-características de fiabilidad. Ahora, continuando con el análisis, se tuvo en cuenta que los elementos relacionados a longevidad son muy concisos en su enfoque y ayudan a describir a detalle de que se trata el componente de longevidad, por dicha razón se determinó que los elementos de vida útil y cambios futuros son propiedades del nuevo componente.

Frente a los elementos del esquema conceptual como respuesta eficiente, cambios permanentes, ajuste al alcance, alcance oportuno y procesamiento de datos, se determinó que no son conceptos que ayuden a diferenciar la longevidad, por lo tanto, no se tomaron en cuenta para la taxonomía.

Aplicando los criterios establecidos en el esquema de elementos relacionados a disponibilidad, y analizando detenidamente cada una de las decisiones tomadas, se presenta la taxonomía de la sub-característica de disponibilidad en la *Ilustración 16*.



3.1.3.4 Tolerancia a fallos

Al realizar el análisis de la sub-característica de tolerancia a fallos, se encontraron 7 trabajos que ofrecían una definición sobre ésta. Para su estudio, se descartaron las definiciones completamente relacionadas con la definición dada por la ISO/IEC 25010, ya que no aportan información adicional para su consideración; por otro lado, se tomaron en cuenta los trabajos que fueron clasificados con relación amplia y parcial relacionada frente a la tolerancia a fallos, ya que aportan nuevos conceptos para tener en cuenta para la extensión y taxonomía.

En [21] se define que la tolerancia a fallos es una condición necesaria para asegurar la fiabilidad y seguridad de los sistemas técnicos. Por otro lado, [7] establece que el objetivo de esta sub-característica es prevenir fallos del sistema en presencia de errores, además que ésta se logra mediante la implementación de técnicas de detección y recuperación de fallos en el sistema. Que un sistema sea a prueba de fallos significa que el sistema falla en un modo que deja los procesos y componentes del sistema en una condición segura. Por lo tanto, para que un sistema sea a prueba de fallos, se requiere una cobertura de detección de fallos casi perfecta para los mecanismos de seguridad. Por otra parte, [24] establece que ésta se puede lograr implementando mecanismos que cubren los defectos en los sistemas software, los cuales previenen que provoquen fallas. En ese mismo sentido, [27] establece que la tolerancia a fallos es el atributo de supervivencia de un sistema que le permite brindar el servicio requerido después de que los fallos se han manifestado por sí mismos dentro del sistema. Por último, [20] menciona que un entorno tolerante a fallos es aquel que incorpore mantenimiento preventivo, el cual es fundamental para maximizar la fiabilidad de sistemas críticos como naves utilizadas en misiones de la NASA.

Cuando se realizó el análisis de la sub-característica de disponibilidad, se mencionó que ésta podría ser calculada con la *Ecuación 1*. Aún así, no se tomó en cuenta dicha ecuación para la expansión de ésta sub-característica, ya que estaba enfocada en medidas obtenidas de los fallos de un sistema, por lo tanto en este caso tendría más sentido considerarla como insumo para la expansión de la sub-característica de tolerancia a fallos. Con el fin de detallar conceptualmente esta sub-característica, se procedió a realizar una descomposición semántica de las definiciones dadas por la norma y la literatura consultada, el resultado se puede ver en la *Ilustración 17*.

Ilustración 17. Descomposición semántica tolerancia a fallos



Entre los nuevos aspectos encontrados en la literatura consultada, se consideró únicamente a cobertura debido a que tiene una clasificación de nivel completamente relacionado con tolerancia a fallos. En [7] se determina que la cobertura se define como una medida de la capacidad del sistema para realizar la detección, localización, contención y/o la recuperación de fallos. Una definición más específica o estrecha de cobertura es: una medida de la capacidad del sistema para recuperarse automáticamente dentro de la duración diseñada después de la ocurrencia de un fallo, tolerando así dicho fallo. Teniendo en cuenta la definición de cobertura, se procedió a realizar una descomposición semántica para analizar los términos que nos ayudó a expandir la definición de la sub-característica en mención, el resultado se puede ver en la **Ilustración 18**.

Ilustración 18. Descomposición semántica cobertura



Como se pudo notar, la mayoría de términos relacionados a cobertura tienen una relación evidente con los términos extraídos de tolerancia a fallos, por lo tanto se pudo deducir que la definición de cobertura puede ser utilizada directamente en la expansión de la subcaracterística de tolerancia a fallos. Dicho lo anterior, se procedió a proponer el esquema de elementos relacionados de la sub-característica de tolerancia a fallos, la cual se presenta en la **Ilustración 19**.

Ilustración 19. *Esquema de elementos relacionados a Tolerancia a fallos*



Teniendo el esquema de elementos relacionados anterior, se procedió a realizar un análisis que permitió crear una propuesta taxonómica de la sub-característica de tolerancia a fallos. Dicho análisis se basó únicamente en el segundo nivel del esquema de elementos relacionados, debido a que el esquema solo se encuentra compuesto por dos niveles.

Iniciando con el análisis, se determinó que, entre los elementos presentados en el esquema de elementos relacionados, se encontraron dos elementos que condensan de una forma muy clara el comportamiento de un sistema tolerante a fallos, estos elementos son detectar fallos y contener fallos.

El elemento de detectar fallos es crucial para un sistema con la capacidad de tolerar fallos, debido a que es indispensable que el sistema pueda prepararse para detectar todos los fallos del sistema o al menos la mayoría de estos. La detección de los fallos se mira relacionada con otros elementos del esquema, los cuales son: prevenir fallos y localizar fallos.

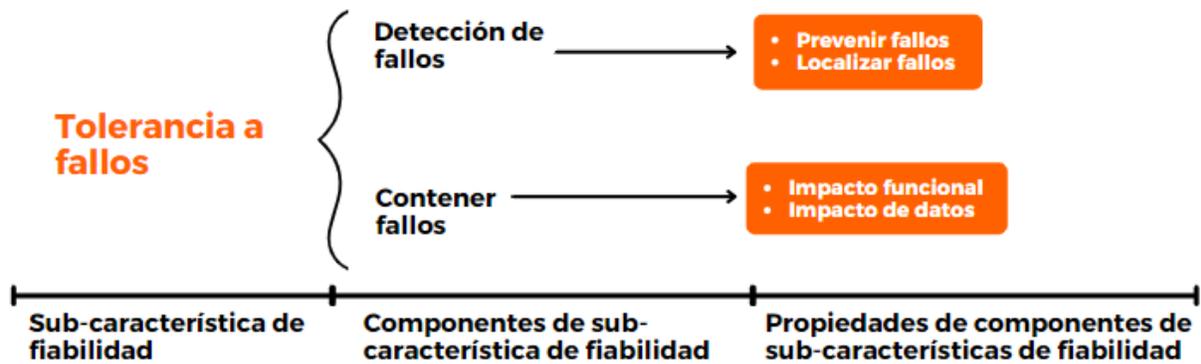
Es importante que el sistema pueda ubicar en que parte se encuentran los fallos para poder brindar un soporte más oportuno y eficaz, esto último se relaciona con el elemento de localizar fallos. La detección oportuna de los fallos puede incluso lograr que el sistema pueda ejecutar mecanismos para prevenirlos o al menos que no afecten de una forma tan drástica. Dicho lo anterior, se tomó la decisión que el elemento de detectar fallos sea un componente compuesto por las propiedades de prevenir fallos y localizar fallos.

El elemento de contener fallos es vital para un sistema, puesto que después de detectar y localizar fallos, el sistema debe estar en la capacidad de ejecutar mecanismos que le permitan seguir operando, esto con el objetivo de brindarle el servicio permanentemente al usuario. La contención

del fallo se mide a nivel impacto funcional y de datos que pueda provocar el fallo, por lo tanto, se tomó la decisión que contener fallos fuera un componente que derive las propiedades de impacto funcional e impacto de datos.

Aplicando los criterios establecidos en el esquema de elementos relacionados a tolerancia a fallos, y analizando detenidamente cada una de las decisiones tomadas, se presenta a continuación la taxonomía de la sub-característica de tolerancia a fallos en la *Ilustración 20*.

Ilustración 20. Taxonomía de Tolerancia a Fallos



3.1.4 Análisis de las definiciones de fiabilidad de acuerdo con la literatura

Como primer paso, tal como se realizó para las sub-características en la sección 3.1.1, se llevó a cabo un proceso de análisis de la literatura consultada, la cual permitió extraer propuestas sobre la definición de fiabilidad como característica de calidad. Dicha información se puede observar en la **Tabla 8**.

Tabla 8. Definiciones de fiabilidad de acuerdo con la literatura

Artículo	Definición de fiabilidad
The development and reliability analysis environment of fault-tolerance multiversion software [21]	La fiabilidad de software se puede lograr con un cuidadoso diseño de arquitectura e identificación temprana de errores en los componentes que tienen grandes efectos en la fiabilidad del sistema. La fiabilidad del software depende de las partes que lo componen y las relaciones entre ellos.
The comprehensive study on software reliability [26]	La fiabilidad del software es la posibilidad de que el software no cause ninguna falla por un número de veces específicas.

Application of diversity method for reliability of cloud computing [29]	La fiabilidad está definida por la tasa de fallos o la probabilidad de falla de un software, esto sirve para definir si es apto para pasar a producción.
Predicting incident reports for IT help desk [3]	La fiabilidad es cuando un software mantiene sus servicios disponibles de forma continua. La fiabilidad del software es cuando se logra tener la mínima cantidad de errores que afecten la disponibilidad de sí mismo.
The comparative study for ENHPP software reliability growth model based on mixture coverage function [32]	La fiabilidad se define como el software que tenga el menor número de fallas encontradas.
A model for analyzing architectural attributes [1]	La fiabilidad es la capacidad de un sistema software en mantener el tiempo de operación libre de probabilidad de falla.
Comparing reliability and security: Concepts, requirements, and techniques [7]	La fiabilidad se define en la medida en que un sistema tiene un alto nivel de cobertura frente a fallas. Los requisitos de confiabilidad tienden a ser simples declaraciones cuantitativas de la disponibilidad del servicio esperada para un sistema o aplicación.
Software Fault Tolerance Forestalls Crashes: To Err Is Human; To Forgive Is Fault Tolerant [20]	La fiabilidad es garantizar que las aplicaciones nunca se "rompan"(fallen), incluso en las circunstancias más exigentes. “La confiabilidad es una propiedad integral, que abarca la seguridad, incluida la confidencialidad, integridad y disponibilidad, corrección, confiabilidad, privacidad, seguridad y supervivencia. No es suficiente abordar solo algunas de estas diversas dimensiones, ni es suficiente simplemente ensamblar componentes que sean en sí mismos confiables. Integrar los componentes y comprender cómo interactúan las dimensiones de confiabilidad es un desafío central en la construcción de un Sistema de Información en Red confiable”.
Predicting Release Reliability [25]	La fiabilidad es la incidencia de errores encontrados en el software en etapas de pruebas y producción.

Longitudinal evaluation of software quality metrics in open-source applications [33]	La capacidad de un sistema o componente para desempeñar las funciones especificadas, cuando se usa bajo unas condiciones y periodo de tiempo determinados.
Software metrics for policy-driven software development life cycle automation [18]	La fiabilidad se define como el grado de confianza para que el software funcione de la manera esperada.
SAS® Data Analytic Development: Dimensions of Software Quality [17]	La fiabilidad es una medida de desempeño contra fallas, a menudo es definida matemáticamente como la probabilidad de funcionamiento sin errores.
Software reliability modeling based on ISO/IEC SQuARE [8]	La fiabilidad es el grado en el que un sistema, producto o componente realiza funciones específicas en condiciones ideales durante un período de tiempo determinado.
A generic model for the specification of software reliability requirements and measurement of their functional size [28]	La fiabilidad se define como la capacidad del producto de software para mantener un nivel específico de rendimiento cuando se usa en condiciones específicas.
Software reliability prediction for open source software adoption systems based on early lifecycle measurements [4]	La fiabilidad del software se define como la probabilidad de que el software funcione sin fallas durante un período de tiempo específico en un entorno específico.
Using NHPP model for ERP software module reliability growth [27]	La fiabilidad del software se define como la probabilidad de operación sin fallas de un programa de computadora por un tiempo específico en un ambiente determinado.
Development of software reliability growth model incorporating enhancement of features and related release policy [24]	La fiabilidad del software es la probabilidad de que un determinado software funcione correctamente en un entorno determinado durante un período de tiempo específico.
A new analysis concept in application of software reliability growth models [19]	La fiabilidad del software significa la relación entre el tiempo de prueba y el número acumulativo de fallas detectadas por la prueba.

Building reliable and maintainable Dynamic Software Product Lines: An investigation in the Body Sensor Network domain [6]	La fiabilidad se caracteriza como la capacidad del sistema de software para brindar servicios en los que se puede confiar justificadamente.
IEEE 1633 - IEEE Recommended Practice on Software Reliability [31]	<p>En IEEE 1633 se establecen dos definiciones para la fiabilidad, las cuales son:</p> <p>a. La probabilidad de que el software no cause falla de un sistema durante un tiempo específico bajo condiciones específicas.</p> <p>b. La capacidad de un programa para realizar una función requerida bajo condiciones establecidas durante un período de tiempo establecido.</p> <p>NOTA: Para la definición (a), la probabilidad es una función de las entradas y el uso del sistema, así como una función de la existencia de fallas en el software. Las entradas al sistema determinan si se encuentran fallas existentes, si las hay.</p>

Por consiguiente, se procedió a realizar la comparación entre las definiciones obtenidas de la literatura consultada, y la definición base ofrecida por la norma, con el fin de conocer el grado de similitud entre las definiciones. Para esto se propuso el uso de la **Tabla 2**, utilizada en la comparación realizada para las sub-características de fiabilidad, ya que los criterios ahí mencionados funcionan para la comparación que se espera hacer en esta sección. Comparando las definiciones de fiabilidad tomando como base la tabla de clasificación, se obtuvo el siguiente resultado que se puede apreciar en la **Tabla 9**.

Tabla 9. Comparativo definiciones de fiabilidad

Definiciones en la literatura	Definición Fiabilidad ISO/IEC 25010
[21]	<i>No relacionado</i>
[26]	<i>Completamente relacionado</i>
[29]	<i>Moderado</i>
[3]	<i>Completamente relacionado</i>

[32]	<i>Completamente relacionado</i>
[1]	<i>Completamente relacionado</i>
[7]	<i>Completamente relacionado</i>
[20]	<i>Ampliamente relacionado</i>
[25]	<i>Completamente relacionado</i>
[33]	<i>Completamente relacionado</i>
[18]	<i>Completamente relacionado</i>
[17]	<i>Completamente relacionado</i>
[8]	<i>Completamente relacionado</i>
[28]	<i>Completamente relacionado</i>
[4]	<i>Completamente relacionado</i>
[27]	<i>Completamente relacionado</i>
[24]	<i>Completamente relacionado</i>
[19]	<i>No relacionado</i>
[6]	<i>Ampliamente relacionado</i>
[31]	<i>Completamente relacionado</i>

Después de tener la clasificación de las definiciones, se realiza el siguiente consolidado de resultados obtenidos, con el propósito de saber el porcentaje de definiciones y el grado de similitud que comparten con la definición de la norma. El consolidado se puede observar en la **Tabla 10**.

Tabla 10. Resultados comparativa definiciones de fiabilidad

Descripción	Escala
Completamente relacionado	75%
Ampliamente relacionado	15%

Parcialmente relacionado	10%
No relacionado	0%

Como producto del análisis realizado a las definiciones de fiabilidad ofrecidas por la literatura consultada, se procede a aplicar un proceso que nos permita proponer una extensión de la definición de fiabilidad tomada de la norma, la cual estará acompañada de una taxonomía que permitirá identificar con mayor claridad los componentes de este atributo.

Después de clasificar las definiciones de fiabilidad obtenidas, se considera que solo se tomarán en cuenta las definiciones cuya clasificación sea completa y parcialmente relacionada, ya que estas contienen nuevos conceptos que pueden aportar a la extensión de esta característica de calidad.

En primera instancia, [20] establece que la fiabilidad garantiza que las aplicaciones nunca fallen, incluso en las circunstancias más exigentes. Además, define que este atributo es una propiedad integral, que abarca la seguridad, incluida la confidencialidad, integridad, disponibilidad, exactitud, privacidad, seguridad y supervivencia.

Por otro lado, [29] establece que la fiabilidad está definida por la tasa de fallos o la probabilidad de fallo de un software, esto sirve para definir si es apto para pasar a producción. Mientras [6] menciona que este atributo se caracteriza como la capacidad del sistema de software para brindar servicios en los que se puede confiar justificadamente [6].

Un poco más alejada de la norma, [21] establece que la fiabilidad de software se puede lograr con un cuidadoso diseño de arquitectura e identificación temprana de errores en los componentes que tienen grandes efectos en la fiabilidad del sistema. La fiabilidad del software depende de las partes que lo componen y las relaciones entre ellos.

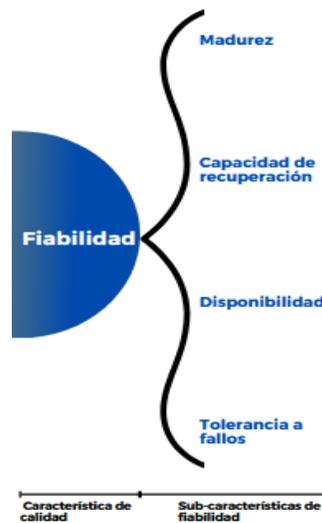
Por último, en la ISO/IEC 25010 se establece que la fiabilidad es el grado en el que un sistema, producto o componente lleva a cabo las funciones especificadas, cuando se usa bajo las condiciones y un periodo de tiempo determinado [2].

Después de tener las definiciones anteriores, se procedió a extender la definición de fiabilidad otorgada por la ISO/IEC 25010 con respecto a las definiciones encontradas en los trabajos relacionados. Dicha extensión de la definición se presentó en la sección 3.1.6.

3.1.5 Definición taxonómica para la característica de fiabilidad

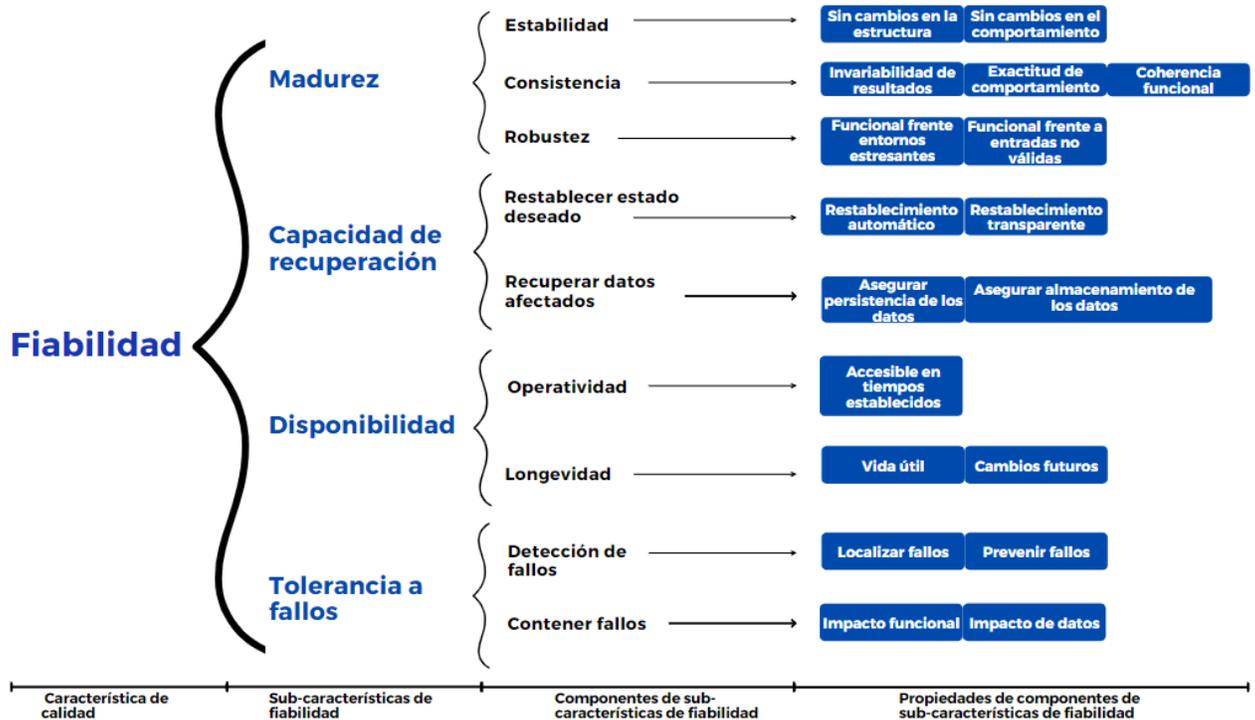
Para construir la propuesta taxonómica de la característica de fiabilidad se tuvo en cuenta que se mantendría la jerarquización establecida por la norma 25010 ISO/IEC, la cual establece que la fiabilidad se extiende en las sub-características de madurez, capacidad de recuperación, disponibilidad y tolerancia a fallos. Dicha jerarquía se puede observar en la **Ilustración 21**.

Ilustración 21. *Fiabilidad de acuerdo con ISO/IEC 25010*



Teniendo en cuenta lo anterior, y contando con las taxonomías individuales de las sub-características de fiabilidad, se procedió a construir la propuesta taxonómica de la característica de fiabilidad, la cual se podrá observar en la **Ilustración 22**.

Ilustración 22. Taxonomía característica de fiabilidad



3.1.6 Extensión de la definición de fiabilidad

Después de realizar el proceso que permitió construir la propuesta taxonómica de la característica de fiabilidad, fue importante extender las definiciones de la característica, las sub-características y los componentes, con el fin de tener mayor claridad sobre los elementos que componen la taxonomía presentada en la sección 3.1.5.

Para dicho propósito, se decidió tomar como base las definiciones de la norma ISO/IEC 25010 y la literatura consultada para construir definiciones más completas que nos ayuden a identificar cada uno de los elementos en un sistema durante el proceso de levantamiento y especificación de requisitos de fiabilidad. A continuación, se encuentran las definiciones extendidas de la característica de fiabilidad, además de sus sub-características y componentes.

Fiabilidad: Es el grado en el que un sistema, producto o componente lleva a cabo las funciones especificadas *con exactitud*, cuando se usa bajo las condiciones o circunstancias más exigentes, en un periodo de tiempo determinado, *para brindar servicios integrales en los que se puede confiar justificadamente* [2, 6, 20]. Esta característica se subdivide a su vez en:

- **Madurez:** Es el grado en el que un sistema, producto o componente satisface las necesidades *de confianza* bajo condiciones normales y/o anormales de operación, además de mantener un comportamiento consistente con lo esperado, esto quiere decir que cumple con las expectativas de los interesados [2, 8]. Esta sub-característica se divide a su vez de los siguientes componentes:
 - **Estabilidad:** Es el grado en el que un sistema, producto o componente no requiere cambios o correcciones en su estructura o comportamiento [8]. *Esta propiedad es de vital importancia en escenarios donde los cambios puedan introducir nuevos defectos en el sistema, los cuales pueden degradar la calidad del producto software y en ese mismo sentido, la confianza del cliente.*
 - **Robustez:** Es el grado en el que un sistema, producto o componente puede funcionar correctamente en presencia de entradas no válidas o en un entorno estresante [8]. *Esta propiedad define el comportamiento del sistema frente a condiciones de ejecución anormales, esto quiere decir que el sistema se mantenga en un estado funcional a pesar de enfrentarse a dichas condiciones.*
 - **Consistencia:** Es el grado en que un sistema, producto o componente no presenta variabilidad en sus resultados, los cuales pueden comprometer o eliminar su valor, manteniendo así la coherencia funcional en el sentido de que se produzcan salidas o soluciones de forma correcta [17].
- **Capacidad de recuperación:** Es el grado en que un sistema, producto o componente en caso de interrupción, fallo o cambio, puede recuperar los datos directamente afectados y restablecer el estado deseado del sistema de forma automática y transparente. De igual forma, se espera que tenga la habilidad de reconfigurar y adaptarse a sí mismo en entornos hostiles o inciertos, soportando así su propio crecimiento y evolución [2, 29]. Esta sub-característica se divide a su vez de los siguientes componentes:

- **Restablecer estado deseado:** Es el grado en que un sistema, producto o componente restaura su funcionamiento esperado de forma automática y transparente para el usuario. Esto se puede lograr por medio de mecanismos de reconfiguración automática que tenga a su disposición.
- **Recuperar datos afectados:** *Es el grado en que un sistema, producto o componente recupera los datos almacenados después de un eventual fallo, con el fin de asegurar la persistencia de estos y por consiguiente su procesamiento.*
- **Disponibilidad:** Es el grado en que un sistema, producto o componente es funcional, operativo y accesible cuando se requiere su uso, esto quiere decir en periodos de tiempo específicos como horas, días, semanas, meses o años [2, 17]. Esta sub-característica se divide a su vez de los siguientes componentes:
 - **Operatividad:** *Es el grado en que un sistema, producto o componente se mantiene funcional en un periodo de tiempo específico. Lo anterior, con el objetivo de lograr que sea accesible para el usuario en los tiempos establecidos.*
 - **Longevidad:** *Es el grado en que un sistema, producto o componente mantiene un periodo tiempo de vida útil del software medido en semanas, meses o años, dado que en este tiempo se deben prever los cambios futuros que exigirán una respuesta eficiente y permanente del software tanto a nivel de procesamiento e incremento de datos, como también el ajuste al alcance de manera oportuna ya sea mejoras, correcciones, evoluciones o eliminaciones en el sistema, ya que estas pueden afectar significativamente la vida útil prevista para el software desde su etapa de especificación [17].*
- **Tolerancia a fallos:** Es el grado en que un sistema, producto o componente, puede localizar, contener, detectar y prevenir fallos, hardware o software, que afecten la operatividad de este según lo previsto, con el objetivo de asegurar su fiabilidad y seguridad

[2, 7, 20]. Esto se puede lograr mediante la implementación de técnicas de detección, contención y prevención de fallos que permitan al sistema seguir ofreciendo los servicios que se requieran, así garantizando su supervivencia [7, 27]. Esta sub-característica se divide a su vez de los siguientes componentes:

- **Detección de fallos:** *Es el grado en que un sistema, producto o componente, tiene la capacidad de detectar y localizar fallos que coloquen en riesgo su operación. El objetivo de detectar fallos es que el sistema tenga la capacidad de contenerlos y cubrir el mayor porcentaje posible, e incluso prevenirlos.*
- **Contener fallos:** *Es el grado en que un sistema, producto o componente, tiene la capacidad de contener los fallos detectados, con el objetivo de fallar en condiciones seguras que disminuyan el impacto del fallo a nivel funcional y de datos. Aumentando de esta forma las probabilidades de supervivencia de este.*

3.2 Proceso de referencia para captura y especificación de requisitos de fiabilidad

En esta sección se describe el proceso de referencia para la captura y especificación de requisitos de fiabilidad, el cual incluye los subprocesos, actividades, artefactos que lo componen y una guía para su aplicación. Este proceso utiliza la taxonomía y las definiciones ampliadas de la fiabilidad, presentadas en la sección anterior.

3.2.1 Definición de artefactos para captura y especificación de requisitos de fiabilidad

Para el proceso de referencia de captura y especificación de requisitos de fiabilidad, se utilizaron artefactos, formatos y plantillas, los cuales apoyarán las actividades de contextualización y ejecución de este. Estos artefactos son los siguientes:

- Presentación marco de trabajo para la captura y especificación de la fiabilidad de un producto software (*Anexo A*)
- Guía para la aplicación del proceso de referencia para capturar y especificar requisitos de fiabilidad (*Anexo B*).
- Preguntas para la captura y especificación de requisitos de fiabilidad (*Tabla 11*).
- Proceso de referencia para capturar y especificar requisitos de fiabilidad (*Ilustración 31*).

- Formato de modelado de proceso de negocio adaptado (*Anexo C*)
- Formato de prototipado para RNF (*Anexo D*).
- Familia iconográfica de fiabilidad extendida (*Ilustración 27*).

3.2.1.1 Presentación de la propuesta

El artefacto denominado presentación de la propuesta, está compuesto por dos partes. La primera parte se trata de una presentación gráfica, donde se explica la propuesta taxonómica de fiabilidad detallando los niveles, elementos y definiciones que la componen. Mientras la segunda parte, se enfoca en explicar el proceso de referencia, las actividades y artefactos que se usan. Este artefacto se encontrará en el Anexo A, el cual servirá para realizar la capacitación a la persona encargada de dirigir la ejecución del proceso de referencia para capturar y especificar requisitos de fiabilidad.

3.2.1.2 Guía para la aplicación del proceso de referencia para capturar y especificar requisitos de fiabilidad

Para la aplicación del proceso de referencia se adapta la guía llamada “*Guía para la aplicación del componente RNF-REP*” tomada de [34], para que sea aplicada en los proyectos de desarrollo de software.

La guía adaptada solo cuenta con dos fases: Fase uno – Identificar proceso de negocio, interfaces y normativas; Fase dos – Especificar requisitos de fiabilidad. Las fases fueron adaptadas a las actividades especificadas en el proceso de referencia. La guía adaptada se encuentra en la *Ilustración 23* y *Anexo B*.

Ilustración 23. Guía para la aplicación del Proceso de referencia para capturar y especificar requisitos de fiabilidad

Check List de Ejecución del Componente de Representación de Requisitos No Funcionales				
Nombre del proyecto:		Número de analistas:	Número de interesados participantes:	
FASE UNO - IDENTIFICAR PROCESO DE NEGOCIO, INTERFACES Y NORMATIVAS				
No.	Tarea	Check	Grado de facilidad de aplicación del ítem (A-M-B)	Observaciones de la ejecución
1	Identificar el conjunto de procesos de negocio involucrados en el proyecto (considerar el concepto de dominio de aplicación) En la plantilla denominada "Formato de modelado de proceso de negocio adaptado", realizar por cada proceso identificado:			
2				
2.1	Registrar el nombre del proceso de negocio			
2.2	Identificar y marcar si hay o no interfaces con sistemas externos			
2.3	Registrar el nombre del sistema que actuará como emisor en la interfaz			
2.4	Registrar el nombre del sistema que actuará como receptor en la interfaz			
2.5	Identificar y marcar el tipo de interfaz (Automática-Semiamatemática-Manual)			
2.6	Identificar y registrar la información/datos que sirven de entrada en la interfaz			
2.7	Identificar y registrar la información/datos que sirven de salida en la interfaz			
2.8	Identificar y marcar si hay o no normativas, políticas que afecten al proceso de negocio en análisis			
2.9	Registrar el nombre de la norma identificada			
2.10	Seleccionar y registrar los ítems regulatorios que deberán ser considerados en el producto software			
2.11	Asociar los identificadores de los prototipos funcionales relacionados con el proceso de negocio			
FASE DOS - ESPECIFICAR REQUISITOS DE FIABILIDAD				
No.	Tarea	Check	Grado de facilidad de aplicación del ítem (A-M-B)	Observaciones de la ejecución
21	Para cada prototipo de los procesos de negocio identificados en la Fase Uno:			

3.2.1.3 Preguntas para la especificación de requisitos de fiabilidad

Las preguntas que apoyan la captura y especificación de requisitos de fiabilidad se plantearon tomando como base la propuesta taxonómica de fiabilidad presentada en la sección 3.1 de este documento, debido a que esta estructura permite definir de forma más concreta unos componentes y a la vez estos componentes se desglosan en propiedades para cada sub-característica de la fiabilidad.

Para la definición de las preguntas de fiabilidad, se tomó la decisión de derivarlas desde las propiedades de cada componente de las sub-características de fiabilidad, eso con el objetivo de obtener preguntas atómicas que se enfocarán en capturar requisitos de fiabilidad de una manera más precisa. Para esto, se tomaron las propiedades y se consultaron las definiciones de las palabras que lo componen en la RAE, para darle un enfoque determinado a la pregunta. En la **Tabla 11** se muestra la sub-característica, componente y propiedad a la cual se encuentra relacionada cada pregunta, esto para facilitar la trazabilidad del requisito que se obtenga.

Las preguntas podrán ser resueltas por el cliente y/o por el analista de requisitos que son el equipo mínimo que requieren las organizaciones para su aplicación.

Tabla 11. Preguntas para capturar y especificar requisitos de fiabilidad

Listado preguntas para captura y especificación de requisitos de fiabilidad				
Sub-característica	Componente	Propiedad	ID Pregunta	Pregunta
Madurez	<i>Estabilidad</i>	<i>Sin cambios en el comportamiento</i>	1	¿Cuáles de las funcionalidades y/o datos del prototipo podrían ser datos o funciones configurables/parametrizables?
		<i>Sin cambios en la estructura</i>	2	¿Cuál componente del prototipo ha requerido correcciones en implementaciones anteriores?
		<i>Sin cambios en el comportamiento</i>	3	¿Cuáles de los datos del prototipo requieren estabilidad para lograr la funcionalidad esperada?
	<i>Robustez</i>	<i>Funcional frente a entradas no validas</i>	4	¿Cuál alerta debería producir el sistema al ingresar un valor no válido en uno de los campos del prototipo con un tipo de dato específico?
		<i>Funcionales frente entornos estresantes</i>	5	¿Cuál de las funciones del prototipo podrían presentar alteración del comportamiento normal?
		<i>Funcionales frente entornos estresantes</i>	6	En caso de que se presenten recursos insuficientes para ejecutar la funcionalidad ¿Cómo se asegurará que el sistema siga siendo funcional? (Ejemplo: Se ofrece una versión más liviana de la aplicación, se informa el tiempo de ejecución promedio, etc.)
	<i>Consistencia</i>	<i>Coherencia funcional</i>	7	En caso de que los procesos de cálculo automático den resultados incorrectos ¿Cómo considera que se deban evidenciar dichos errores?
		<i>Exactitud de comportamiento</i>	8	¿En cuantos decimales espera que el sistema produzca los resultados de operaciones numéricas?
		<i>Invariabilidad de resultados</i>	9	¿Cuándo se lograría la consistencia en los resultados de la funcionalidad?
Capacidad de recuperación	<i>Restablecer estado deseado</i>	<i>Restablecimiento transparente</i>	10	Frente a un fallo ¿Cuál considera que es la forma más transparente de comunicarle al usuario que el sistema se encuentra recuperándose? (Ejemplo: Mensaje de espera.)
		<i>Restablecimiento Transparente</i>	11	Luego de una perdida de conexión y recuperar la conexión ¿El sistema debe recargar la funcionalidad de manera automática, o informar al usuario que debe hacerlo de manera manual? (Ejemplo: Mensaje de que el usuario envíe de nuevo la petición, el sistema informa que la conexión se restableció y se está intentando de nuevo enviar la petición)

		Restablecimiento Transparente	12	Después de un eventual fallo ¿Cómo espera que el sistema le notifique que el fallo fue superado y el sistema se encuentra disponible para su uso?
		Restablecimiento automático	13	¿Cuál de las funcionalidades debería tener la capacidad de reconfigurarse automáticamente frente un fallo?
	Recuperar datos afectados	Asegurar persistencia de los datos	14	En caso de fallo durante el diligenciamiento y carga de información ¿Al ingresar nuevamente espera que la información consignada se muestre de nuevo?
		Asegurar almacenamiento de los datos	15	En escenarios de recuperación parcial (por ejemplo, cuando se recuperan solo una parte de los datos) de los datos ¿cómo la funcionalidad debería informar que faltan datos por diligenciar o cargar?
Disponibilidad	Operatividad	Accesible en tiempos establecidos	16	¿Cómo se debe informar que no es posible acceder a la funcionalidad en horarios no establecidos? (Ejemplo: Mensaje de alerta de acceso restringido)
		Accesible en tiempos establecidos	17	¿Qué situaciones específicas requerirían un periodo de tiempo funcional especial?
		Accesible en tiempos establecidos	18	¿Cuándo tiempo debe estar disponible el sistema?
	Longevidad	Vida útil	19	¿Cuáles evoluciones y de que tipo se pueden realizar en la funcionalidad?
		Cambios futuros	20	¿Cuáles serían las funcionalidades de este prototipo que pueden ir evolucionando a través del tiempo?
	Tolerancia a fallos	Detección de fallos	Localizar fallos	21
Localizar fallos			22	En caso de fallo ¿Se debería mostrar el nombre del componente/funcionalidad donde se localiza el fallo?
Prevenir fallos			23	¿Cuál funcionalidad podría ser más susceptible a presentar fallos en un futuro?
Contener fallos		Impacto funcional	24	¿Cuál funcionalidad podría generar el mayor impacto funcional en caso de un fallo?
		Impacto de datos	25	Al momento de un fallo, ¿Cuáles datos involucrados en el prototipado son más susceptibles a generar un mayor impacto?
		Impacto funcional	26	¿Cuál funcionalidad debe controlar la ejecución del proceso aun cuando el servicio falle? (Ejemplo: La funcionalidad permite seguir editando el documento sin conexión a internet.)

3.2.1.4 Formato de modelado de proceso de negocio adaptado

El segundo artefacto llamado “Formato de modelado de proceso de negocio adaptado” se basó en la propuesta dada por [34], la cual se compone de: proceso de negocio, interfaces, tipos de interfaz, entradas, salidas, normas, puntos normativos, y representación de RNF. A continuación, se puede observar la plantilla original en la *Ilustración 24*.

Ilustración 24. Formato de modelado de proceso de negocio

Proceso de negocio:			
Interfaces (SI/NO)		Tipo Interfaz:	
Emisor	Receptor	A	SA
			M
Entradas:		Salidas:	
Normas (SI/NO)		Puntos normativos:	
Nombre:			
 <input type="checkbox"/>	 <input type="checkbox"/>	 <input type="checkbox"/>	 <input type="checkbox"/>
Mantenibilidad	Usabilidad	Fiabilidad	Seguridad
 <input type="checkbox"/>	 <input type="checkbox"/>	 <input type="checkbox"/>	
Portabilidad	Compatibilidad	Eficiencia de desempeño	

PROTOTIPOS ASOCIADOS		

Teniendo en cuenta que nuestro trabajo está enmarcado en requisitos de fiabilidad, se tomó la decisión de adaptar el formato de modelado de proceso de negocio a la característica de fiabilidad, para esto se decidió suprimir la sección de características de calidad presentadas en [34].

A continuación, en la **Ilustración 25**, se presenta el formato de modelado de proceso de negocio para RNF de fiabilidad.

Ilustración 25. Formato de modelado de proceso de negocio adaptado

Proceso de negocio:		
Interfaces (SI/NO)		Tipo Interfaz:
Emisor	Receptor	A SA M
Entradas:	Salidas:	
Normas (SI/NO)	Puntos normativos:	
Nombre:		

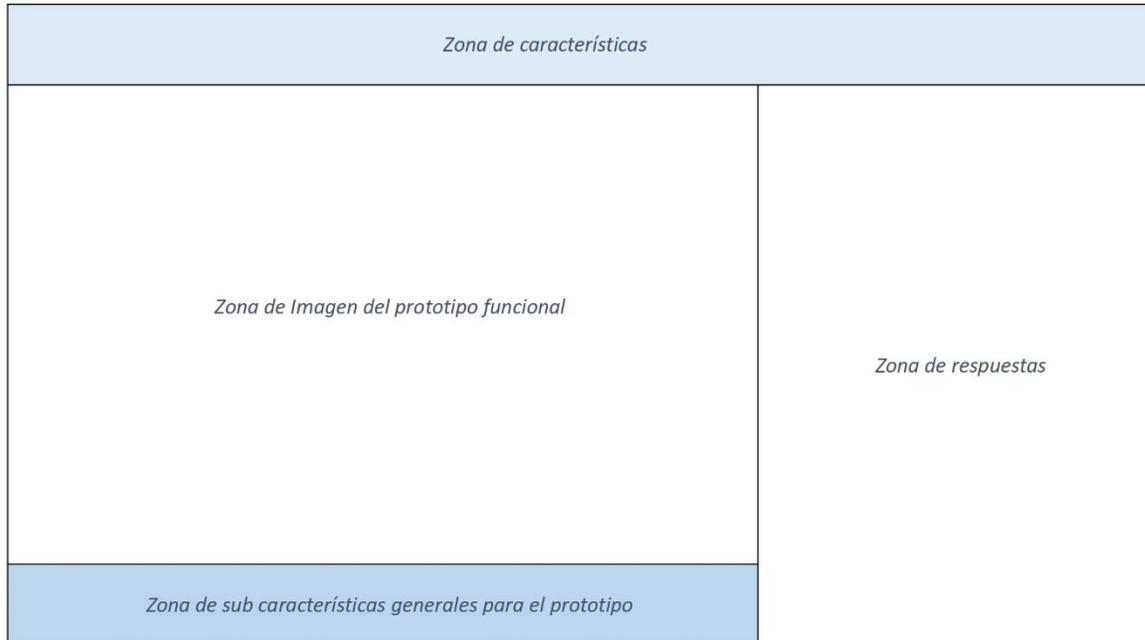
PROTOTIPOS ASOCIADOS		

4.2.1.5 Formato de prototipado para RNF

El tercer artefacto para utilizar es formato de prototipado para RNF tomado de [34], el cual se compone del prototipo, la característica de calidad y las sub-características asociadas a esta. Para nuestro caso en particular, en la parte superior solo existirá el icono de fiabilidad y en la parte inferior las sub-características de fiabilidad, mientras al costado se consignarán las preguntas

desglosadas a nivel de componentes de sub-características de fiabilidad. El formato de prototipado para RNF se puede observar en la *Ilustración 26*.

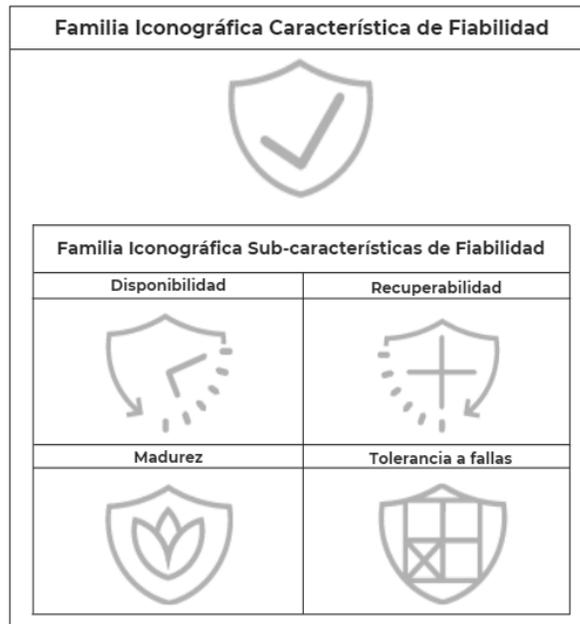
Ilustración 26. Formato de prototipado para RNF



3.2.1.6 Familia iconográfica de fiabilidad extendida

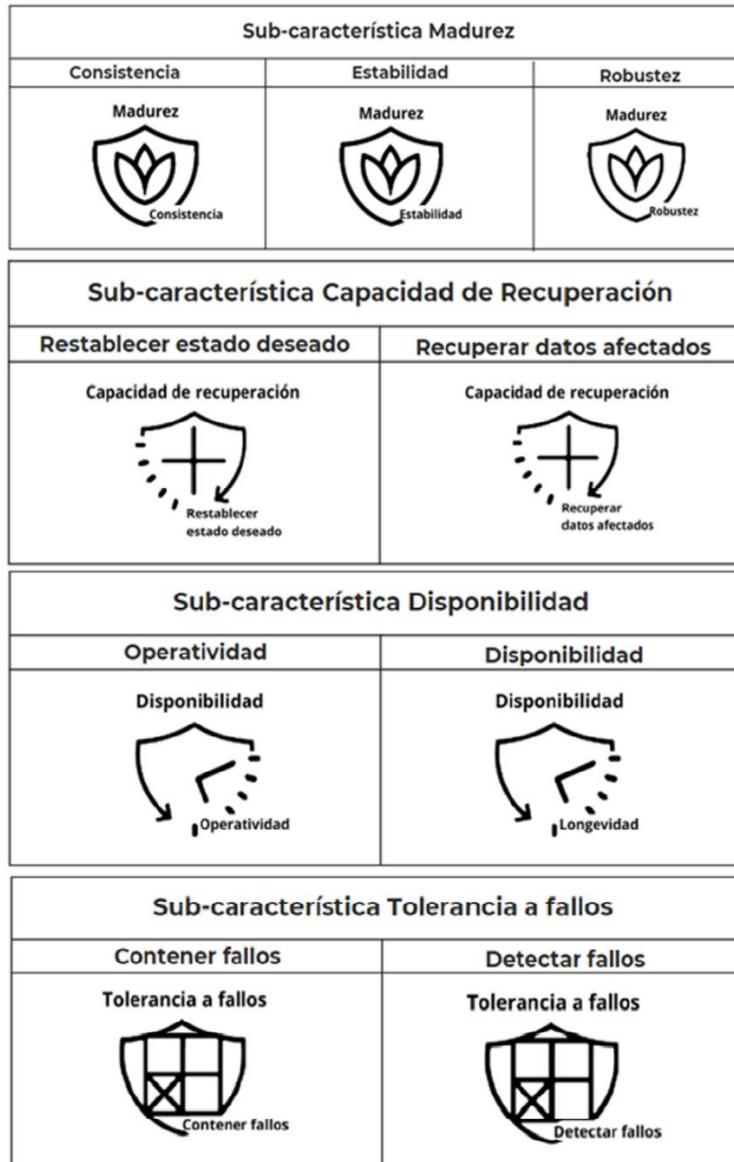
La familia iconográfica de fiabilidad es una propuesta de [34], la cual es usada por el autor para representar de una forma gráfica la fiabilidad y las sub-características que se derivan de esta. La propuesta original se puede observar en la *Ilustración 27*.

Ilustración 27. Familia Iconográfica de fiabilidad



Se pensó en extender la familia iconográfica de fiabilidad presentada por [34], teniendo en cuenta la propuesta taxonómica de fiabilidad presentada en la sección 3.1, con el objetivo de definir iconos a nivel de componentes de sub-características de fiabilidad, los cuales permitieran identificar de una forma más precisa el requisito de fiabilidad que se especificará a nivel de prototipo en el “Formato de prototipado para RNF”. Para esto, se decidió tomar el icono relacionado a la sub-característica, añadirle el nombre de la sub-característica en la parte superior y añadirle el nombre del componente al costado inferior derecho. A continuación, en la **Ilustración 28**, se presenta la familia iconográfica de fiabilidad extendida.

Ilustración 28. Familia iconográfica de fiabilidad extendida



3.2.2 Definición del proceso de referencia

El proceso de referencia, realizado en notación BPMN en la herramienta Bizagi Modeler, se describe en términos de propósito, cuenta con una identificación de proceso y además incluye información como: roles, subprocessos, actividades, entradas, salidas y productos de trabajo [22].

3.2.2.1 Definición de roles a intervenir en proceso de referencia

De acuerdo con ISO/IEC/IEEE 29148, antes de iniciar con un proceso de captura de requisitos, es importante definir los involucrados que van a intervenir en el proceso, ya sea por su rol, posición o influencia en el proyecto [15]. Por dicha razón se proponen dos roles (analista de requisitos y cliente) que pueden aportar a la captura y especificación de requisitos de fiabilidad, basados en su experiencia y ejercicio en el rol.

El proceso de captura de requisitos es ejecutado principalmente por un rol de analista de requisitos, la persona encargada tiene el dominio sobre el producto software, es quien está encargado de capturar y especificar los requisitos tanto funcionales como no funcionales, encontrando la diferencia entre lo que dice el cliente y lo que en realidad necesita.

El cliente sin duda es el punto central en el proceso de captura de requisitos, es quien determina si se está cumpliendo sus expectativas, debido a que conoce de cerca el proceso de negocio.

Por lo anterior, los roles que se tomaron en cuenta para el proceso de captura de requisitos de fiabilidad son: analista de requisitos y cliente. Para cada uno de los roles a considerar se añadió sus respectivas competencias. A continuación, en la **Tabla 12**, se presentan los roles que se tomaron en cuenta.

Tabla 12. Roles proceso de referencia

ID	Rol	Competencias
1	Analista de requisitos (AR)	Es el responsable de convertir los requerimientos del cliente en requisitos que sirvan como insumo principal para el diseño e implementación del producto software. Lo anterior lo hacen por medio de técnicas de captura, especificación y diseño de requisitos.
2	Cliente (C)	Es la persona que tiene conocimiento del proceso de negocio. Tiene la capacidad de establecer las reglas de negocio que se deben cumplir, como el flujo del proceso por el cual se debe regir el comportamiento del sistema. De igual forma, brinda información relevante a los analistas del producto pueden definir de la forma más detalla los requisitos. Por último, toma las decisiones finales referentes al producto desarrollado.

3.2.2.2 Definición la actividad para contextualizar sobre la taxonomía fiabilidad.

Partiendo que el proceso de referencia tiene como objetivo capturar y especificar requisitos de fiabilidad, y en el capítulo anterior se presentó una propuesta taxonómica de la característica de

fiabilidad, se consideró importante incluir una actividad de contextualización como primer paso para iniciar con el proceso de referencia.

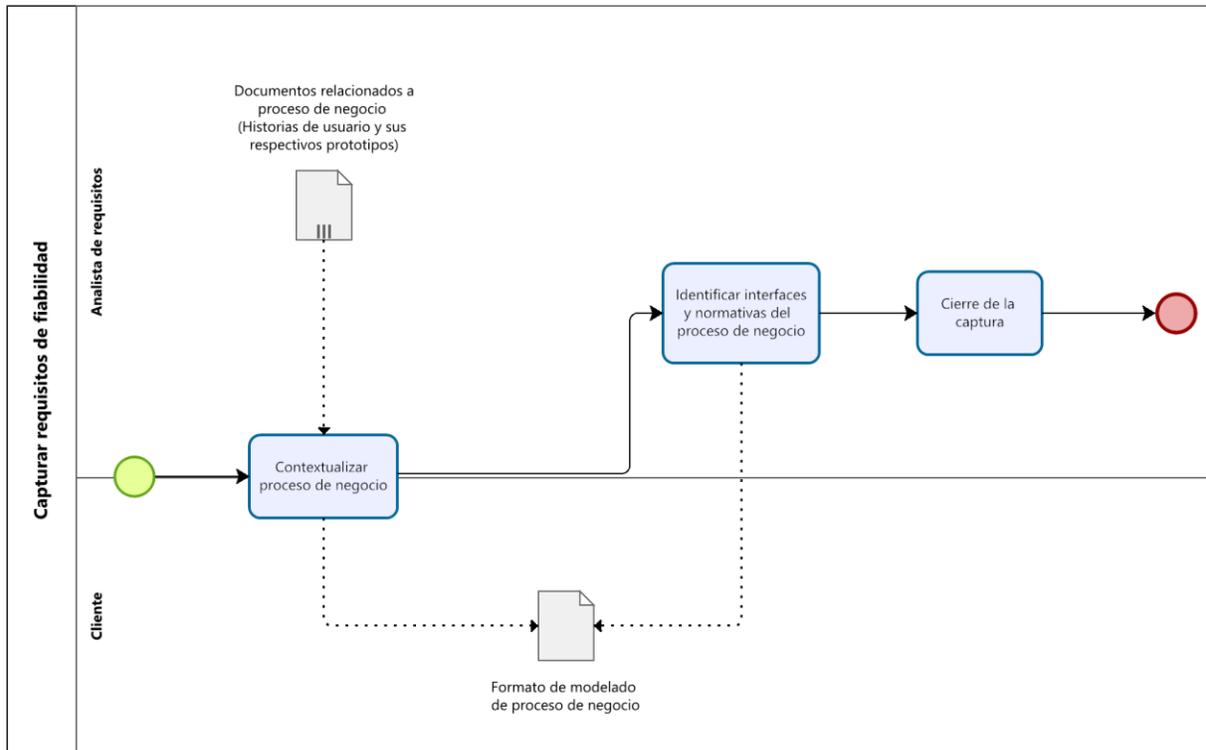
La actividad de contextualización tiene con objetivo socializar la propuesta taxonómica para que el equipo de interesados del proyecto conozca la característica de fiabilidad con las sub-características, componentes y propiedades que la componen, lo cual les permitirá tener conocimientos más amplios y específicos sobre la fiabilidad, y ayudará significativamente en el proceso de captura y especificación.

Esta actividad parte del supuesto que los interesados a intervenir en el proceso de referencia no conocen más allá de las sub-características de fiabilidad, ya que la definición de los niveles de componentes de sub-características de fiabilidad y propiedades de componentes de sub-características de fiabilidad es un producto de este trabajo.

- **Entradas:** Taxonómica de fiabilidad (*Ver secciones 3.1.5 Definición taxonómica para la característica de fiabilidad y 3.1.6 Extensión de la definición de fiabilidad*)

3.2.2.3 Definición del subproceso para capturar requisitos de fiabilidad

Ilustración 29. Subproceso para capturar requisitos de fiabilidad



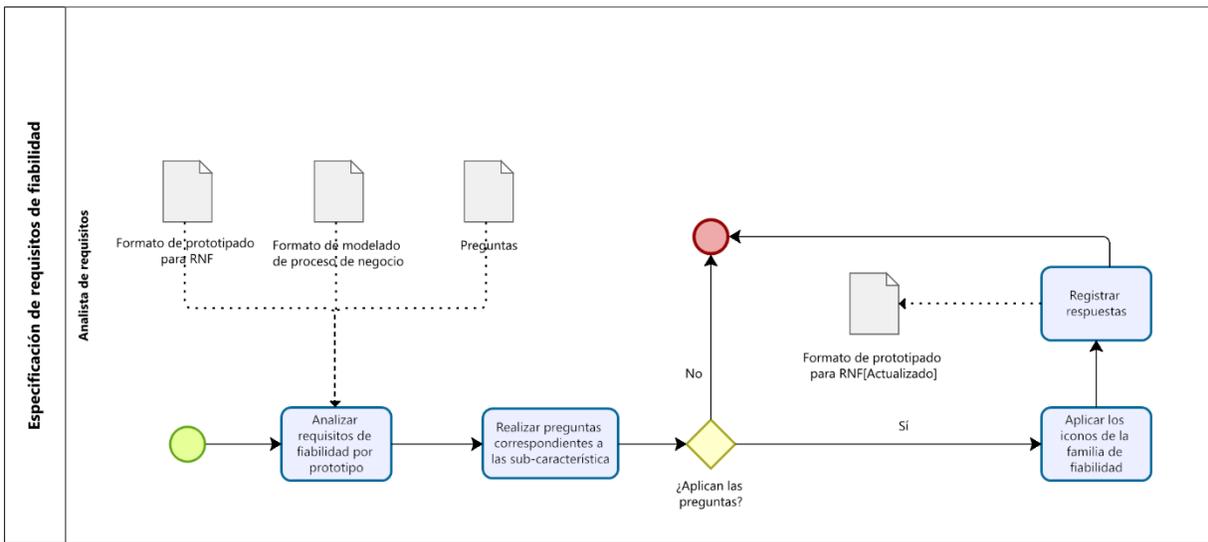
El subproceso de captura de requisitos de fiabilidad, **Ilustración 29**, tiene como objetivo identificar el proceso de negocio y las sub-características de fiabilidad asociadas a cada funcionalidad. Las actividades relacionadas a este subproceso son las siguientes:

- **Contextualizar sobre proceso de negocio:** Esta actividad tiene como objetivo estudiar el proceso de negocio donde se va a aplicar el proceso de referencia, esto por medio de la revisión de artefactos de especificación de requisitos como historias de usuario y sus respectivos prototipos. Adicionalmente, durante esta actividad se diligenciará el formato de modelado de proceso de negocio, el cual fue tomado del trabajo y fue adaptado para la captura de requisitos de fiabilidad. En dicho formato se capturará información relevante como nombre de la funcionalidad, interfaces relacionadas, normativas que intervienen y las sub-características de fiabilidad que se identificaron en la funcionalidad.
 - **Entradas:** Documentos relacionados a proceso de negocio.

- **Salidas:** Formato de modelado de proceso de negocio.
- **Cierre de la captura:** Esta actividad tiene como objetivo realizar el cierre del proceso de captura de requisitos de fiabilidad, se socializará a los interesados involucrados en el proceso las actividades que se realizarán después de finalizar la captura.

3.2.2.4 Definición del subproceso para especificar requisitos de fiabilidad

Ilustración 30. Subproceso para especificar requisitos de fiabilidad



El objetivo de este subproceso, *Ilustración 30*, es tomar los requerimientos de fiabilidad obtenidos por medio del subproceso de captura de requisitos, y especificarlos por medio de actividades colaborativas con los interesados.

- **Analizar requisitos de fiabilidad por prototipo:** Esta actividad tiene como objetivo analizar las funcionalidades del proceso de negocio especificadas en el formato de modelado de proceso de negocio diligenciado durante el subproceso de captura. En este punto se prepara un formato de prototipado para RNF de fiabilidad para cada prototipo relacionado a cada funcionalidad identificada durante las actividades de captura. Además, se determinarán un conjunto de preguntas asociadas a las propiedades del componente de la sub-característica que aplica para la funcionalidad.

- **Entradas:** Formato de prototipado para RNF de fiabilidad, formato de modelado de proceso de negocio, listado preguntas para captura y especificación de requisitos de fiabilidad.
- **Realizar preguntas correspondientes a las sub-características de fiabilidad:** Esta actividad tiene como objetivo realizar las preguntas definidas en la actividad anterior para cada prototipo/funcionalidad. Las preguntas se realizarán a cada interesado dependiendo el dominio que tiene sobre atributos de fiabilidad específicos, partiendo de la experiencia y formación de su rol. Si la pregunta aplica se procede a pasar a la actividad de aplicar los iconos de la familia de fiabilidad, de lo contrario se da por terminado el subproceso de especificación de requisitos de fiabilidad para el prototipo actual.
- **Aplicar los iconos de la familia de fiabilidad:** Esta actividad tiene como objetivo aplicar los iconos de la familia de fiabilidad, partiendo que las preguntas sobre los componentes de sub-características de fiabilidad aplican para especificar el requisito de fiabilidad.
- **Registrar respuestas:** Esta actividad tiene como objetivo registrar las respuestas dadas por cada interesado sobre la pregunta realizada, y consignarla en el formato de prototipado para RNF de fiabilidad para cada prototipo/funcionalidad.
 - **Salidas:** Formato de prototipado para RNF.

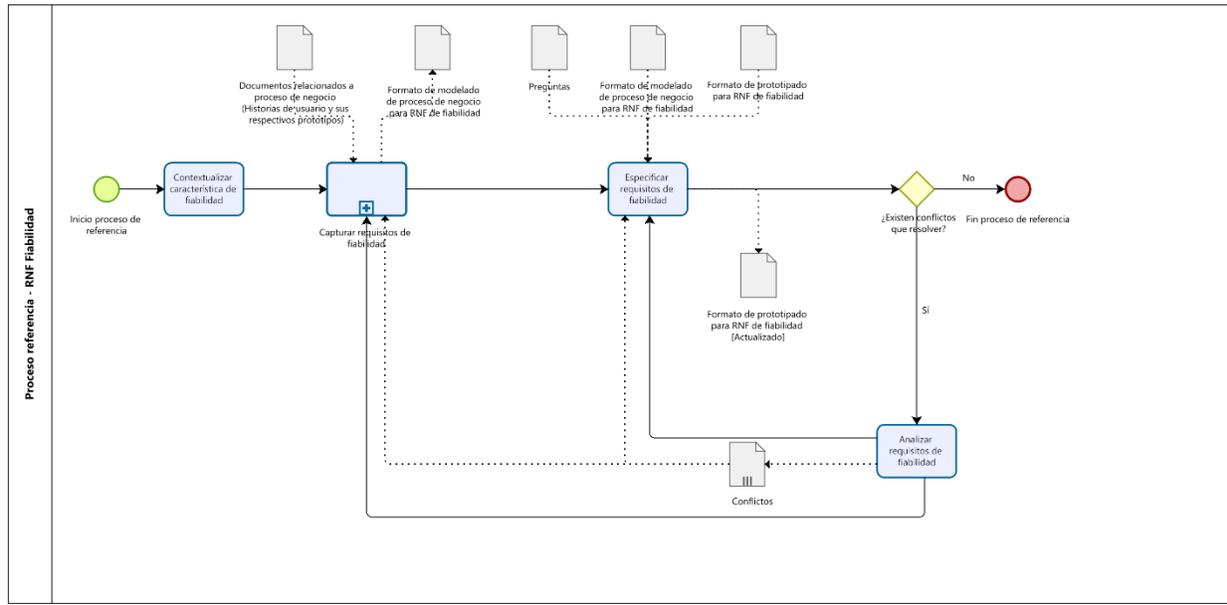
3.2.2.5 Definición de la actividad de analizar requisitos de fiabilidad

Esta actividad tiene como objetivo analizar, validar y verificar los requisitos de fiabilidad especificados durante el proceso, con el fin de encontrar defectos o conflictos en el requisito. Estos serán solucionados por medio de reuniones con los interesados, y podrán terminar refinándose en la etapa de captura o especificación del requisito de fiabilidad.

- **Entradas:** Conflictos (Resueltos).

A continuación, en la **Ilustración 31**, se presenta el proceso de referencia para capturar y especificar requisitos de fiabilidad, compuesto por los subprocesos y actividades descritos anteriormente.

Ilustración 31. Proceso de referencia para captura y especificación de requisitos de fiabilidad



Capítulo 4 – Evaluación de marco de trabajo para capturar y especificar de requisitos de fiabilidad.

En este capítulo se describe la evaluación preliminar del marco de trabajo para capturar y especificar requisitos de fiabilidad, dicha evaluación fue realizada por medio de un estudio de caso, el cual permitió comprobar la utilidad del proceso de referencia aplicado a una empresa de desarrollo de software en un contexto real.

Para realizar la evaluación, se hizo uso del protocolo de estudio de caso simple propuesto por [35], debido a que se aplicará el marco de trabajo para captura y especificación de requisitos de fiabilidad en un proyecto perteneciente a una organización.

4.1 Diseño del estudio de caso

El objeto del estudio de caso es el marco de trabajo para la captura y especificación de requisitos de fiabilidad, el cual se encuentra compuesto por un elemento conceptual (taxonomía de la fiabilidad) y un elemento metodológico (proceso de referencia), el cual se aplicará en un proyecto en el contexto real con el objetivo de evaluar la utilidad del marco de trabajo. Para el estudio de caso se definió una pregunta de investigación principal y dos preguntas adicionales, las cuales se pueden observar en la Tabla 13.

Tabla 13. Preguntas de investigación estudio de caso

Tipo	Enfocada en	Preguntas
Principal	Marco de trabajo para capturar y especificar requisitos de fiabilidad	¿El marco de trabajo propuesto es útil ² para capturar y especificar requisitos de fiabilidad?
Adicional	Los elementos del proceso de referencia para capturar y especificar requisitos de fiabilidad	¿Los artefactos que componen el proceso de referencia son comprensibles al ser aplicados?

² Utilidad: Que puede servir y aprovechar en alguna línea. (Real Academia de la Lengua Española. <https://dle.rae.es>). El proceso de referencia será evaluado en términos de la utilidad para capturar y especificar requisitos de fiabilidad.

Adicional	Los interesados y su entendimiento de la característica de fiabilidad	¿El proceso de referencia permite a los interesados identificar y solicitar requisitos de fiabilidad?
-----------	---	---

En cuanto al criterio para seleccionar la empresa, se decidió buscar una empresa en el sector local que tuviera mínimo 5 años de experiencia en la industria del desarrollo software y hubiera manejado con anterioridad temas relacionados a ISO/IEC 25010. Además, tuviera proyectos software con interés en capturar y especificar requisitos de fiabilidad, dada la importancia de estos requisitos en los procesos de negocio. La empresa Namtrik Development es una microempresa colombiana con sede en Popayán - Cauca, y con operación desde el año 2016, que cuenta con 3 personas en la base de la administración y que realizan subcontratación de acuerdo con las oportunidades de negocio. Esta empresa se dedica al desarrollo de software, especializada en aplicaciones Web, móviles, financieras y desktop.

En las conversaciones que se sostuvieron con la empresa, se expresó el interés en capturar y especificar requisitos de fiabilidad, debido a que hasta la actualidad solo han especificado requisitos relacionados a las características de calidad como seguridad y usabilidad. Además, consideraron que los requisitos de fiabilidad son muy importantes para algunos de sus productos software, y que estos deben tener alta tolerancia al fallo y capacidad de recuperación. Lo anterior, evidencia la falta de un proceso importante para la organización y que ayudaría de una forma muy amplia a la calidad de sus productos software.

En cuanto a los sujetos de investigación, se definió que el equipo de investigación estaría compuesto por: (i) dos investigadores, expertos en la característica de fiabilidad y el proceso de referencia, los cuales capacitarán al analista de requisitos en la taxonomía de fiabilidad y los artefactos involucrados en el proceso de referencia para su aplicación. (ii) un equipo técnico, compuesto por un analista de requisitos y un líder técnico, ver **Tabla 14**, para realizar la aplicación del marco de trabajo; estos realizarán las actividades de captura y especificación de los requisitos de fiabilidad haciendo uso de los artefactos de “Formato de modelado de proceso de negocio adaptado” y “Formato de prototipado para RNF”. Debido a que Namtrik Development es una microempresa, se pudo evidenciar que los diferentes roles técnicos que se definen dentro del ciclo de desarrollo del software suelen ser ejercidos por pocas personas lo que muestra que se requiere

una capacidad de trabajo multidisciplinaria, por lo tanto, para efecto de la aplicación del estudio de caso el arquitecto de software que asigna la empresa tendrá el rol de analista de requisitos, mientras el líder técnico tendrá el rol de cliente.

Tabla 14. *Sujetos de investigación*

Rol	Responsabilidad
Analista de requisitos	Liderar la ejecución del proceso de referencia haciendo uso de los artefactos definidos.
Cliente	Colaborar en la ejecución del proceso de referencia dando la información respectiva a los procesos de negocio implicados en el proyecto, y dando respuesta a las consultas que realice el analista de requisitos.
Investigador	Acompañar y apoyar en dudas de aplicación de la propuesta.

Para analizar la utilidad del marco de trabajo para la captura y especificación de requisitos de fiabilidad, se definieron medidas cuantitativas que permitieran evaluar los resultados obtenidos en el estudio de caso. A continuación, se presentan las medidas utilizadas en la **Tabla 15**.

Tabla 15. *Métricas estudio de caso*

Preguntas	Medidas
¿Los artefactos que componen el proceso de referencia son comprensibles al ser aplicados?	Nivel de claridad de los artefactos del proceso de referencia.
¿El proceso de referencia permite a los interesados identificar y especificar requisitos de fiabilidad?	Nivel de practicidad para la captura de requisitos de fiabilidad.

Las métricas cuantitativas serán tomadas a través de las encuestas que se realizarán al analista de requisitos y el cliente, de acuerdo con las plantillas y formatos diseñados en este protocolo de evaluación preliminar. Dichas encuestas se pueden encontrar en el **Anexo E** y **Anexo F**.

La empresa expresó su interés de aplicar el marco de trabajo en uno de los proyectos que se encuentra actualmente en producción, llamado Solucredit, el cual se basa en la integración de

procesos de crédito para el mercado de microfinanzas. Solucredit tiene un concepto de plataforma que unifica los procesos de crédito de las organizaciones dándole acceso a los aliados que usan su método de crédito, pero sirve a la vez como una pasarela de crédito que permite ejecutar procesos de otorgamiento del crédito. El proyecto Solucredit integra tres sistemas llamados: Software de Firma electrónica y digital, Software Consulta de datos públicos y Software de Scoring Crediticio, los cuales están interconectados; para la evaluación del marco de trabajo se tomó el sistema de firma digital como *unidad de análisis*.

4.2. Intervención

Para la intervención del estudio de caso, se definió como lugar de ejecución las instalaciones de la empresa Namtrik Development, y se establecieron distintas actividades para realizar la aplicación de acuerdo con las disposiciones de la empresa. Las actividades en mención se pueden observar en la **Tabla 16**.

Tabla 16. *Actividades para aplicación de estudio de caso*

Actividad	Fechas de ejecución	Objetivo	Responsables
Presentación de la propuesta	17 de mayo de 2022	-Dar a conocer la propuesta del marco de trabajo junto con los objetivos académicos que se buscan de la aplicación de esta. -Explicar la guía y los artefactos que componen la propuesta.	-Investigadores
Contextualización del concepto de fiabilidad	17 de mayo de 2022	-Reforzar los conocimientos específicos frente al concepto de fiabilidad para aportar una mayor fluidez en la ejecución de la propuesta.	-Investigadores
Ejecución del proceso de referencia	17 de mayo de 2022	-Realizar cada uno de los pasos de la guía dentro del	-Analista de requisitos. -Cliente.

		contexto del proyecto Solucredit.	-Investigadores (observadores)
Cierre de la experiencia	17 de mayo de 2022	-Recibir y consolidar los resultados de la experiencia para un posterior análisis.	-Investigadores (observadores).

La *unidad de análisis* escogida para la intervención del estudio de caso es el sistema integrado Firma digital que se encuentra dentro de Solucredit. A continuación, en la **Tabla 17**, se presenta el proyecto, la empresa, el nombre del proyecto y la descripción del proyecto.

Tabla 17. *Unidad de análisis para estudio de caso*

ID Proyecto	Empresa	Nombre del proyecto	Descripción del proyecto
Proyecto 1 (P1)	Namtrik Development	Solucredit – Firma digital	Generación y autorización de solicitudes de créditos de bajo monto en Solucredit.

A continuación, se describirá la ejecución del estudio de caso para validar la utilidad del marco de trabajo para la captura y especificación requisitos de fiabilidad.

En la intervención del estudio de caso, los investigadores realizaron una capacitación al equipo de arquitecto de software y líder técnico sobre la taxonomía de fiabilidad y los artefactos relacionados al proceso de referencia, como formatos y plantillas. Además, se explicó las actividades que la suceden a la contextualización, ya que sería el analista de requisitos y el cliente quienes las realizarían. La evidencia de la intervención realizada en las instalaciones de Namtrik se pueden ver en la **Ilustración 32** e **Ilustración 33**.

Ilustración 32. Evidencia ejecución estudio de caso



Ilustración 33. Evidencia ejecución estudio de caso



Preliminarmente la empresa seleccionó los procesos de negocio originación y autorización para aplicar el marco de trabajo, y se tuvo a disposición los requisitos funcionales (historias de usuario y prototipos) necesarios para iniciar con la aplicación.

El arquitecto de software ejerciendo el rol de analista de requisitos, inició la ejecución de sus actividades en la fase uno - identificar proceso de negocio, interfaces y normativas. En dicha fase, el arquitecto tomó la guía de para la aplicación del proceso de referencia, analizando detenidamente para tener claridad en las actividades que debía ejecutar. El arquitecto empieza a diligenciar el formato de modelado de proceso de negocio adaptado en colaboración con el líder técnico, donde especifican el nombre de proceso de negocio, el tipo de interfaz, entradas de la interfaz, salidas de la interfaz, normas, normativas y los prototipos relacionados.

Al tener el proceso de negocio capturado, el arquitecto de software tomó el formato de prototipado para RNF, colocó el prototipo asociado al proceso de negocio en la plantilla y el icono de fiabilidad en la sección de característica de calidad. El arquitecto tomó el listado de preguntas para captura y especificación de requisitos de fiabilidad, empezó a formular cada pregunta al líder técnico, si la

preguntaba aplicaba se consignaba la respuesta en la sección de respuestas y se ubicaba el icono relacionado al componente de la pregunta en la parte del prototipo donde aplicaba; si la pregunta no aplicaba, la respuesta no se diligenciaba en el formato.

Al finalizar la captura y especificación de requisitos de fiabilidad, se aplicaron encuestas al arquitecto de software como analista de requisitos, y al líder técnico como cliente, los cuales nos permitirán responder las preguntas de investigación del estudio de caso.

4.3 Resultados

A continuación, se muestran los resultados obtenidos de la aplicación del marco de trabajo para la captura y especificación de requisitos de fiabilidad.

- Información general del proyecto Solucredit:
- Número de arquitectos software: 1
- Número de líderes técnicos: 1
- Número de interesados del negocio participantes en el proyecto: 2
- Número de procesos de negocio: 2
- Número de prototipos elaborados y trabajados en el proyecto: 2

4.3.1 Resultado del uso de la plantilla “Proceso de negocio originación del proyecto Solucredit”

Los resultados del uso de la plantilla de modelado de proceso de negocio adaptado para el proceso de negocio de originación se puede ver en el *Anexo G e Ilustración 34*.

Ilustración 34. Proceso de negocio originación - Formato de modelado de proceso de negocio adaptado

Proceso de negocio: ORIGINACIÓN		
Interfaces (SI/NO)		Tipo Interfaz:
Emisor	Receptor	A <input checked="" type="checkbox"/> M
SOLUCREDIT	API GATEWAY	
Entradas: -Valor del credito. -Cuota inicial. -Número de cuotas. -Número de cedula. -API Key.		Salidas: -Valor de cuota con interes. -Valor de cuota sin interes. -Valor fianza. -Valor de interes.
Normas (SI/NO) SI Nombre: -Tasa de usura del banco de la republica.		Puntos normativos: N/A

PROTOTIPOS ASOCIADOS		
P2		

4.3.2 Resultado del uso de la plantilla “Prototipos de RNF proceso de negocio originación del proyecto Solucredit”

Las *ilustraciones 35, 36, 37, 38, 39* corresponden al proceso de negocio de originación, teniendo en cuenta la longitud de las preguntas y las respuestas, no fue posible tener solo un formato de prototipado de RNF para cada prototipo. El formato de prototipado de RNF diligenciado se puede encontrar completo en el *Anexo H*.

Ilustración 35. Resultado proceso de negocio originación - Formato de prototipado para RNF

Top Sales

Bajo Monto - Credibike

Crédito de montos de hasta T000.000

<input type="text" value="Valor solicitado*"/>	<input type="text" value="Cuota inicial*"/>
<input type="text" value="Número de cuotas*"/>	<input type="text" value="Valor estimado de cuota*"/>
<input type="text" value="Tipo de identificación*"/>	<input type="text" value="Número de identificación*"/>
<input type="text" value="Cédula de ciudadanía"/>	
<input type="text" value="Fecha de expedición*"/>	
<input type="text" value="Primer nombre*"/>	<input type="text" value="Segundo nombre"/>
<input type="text" value="Primer apellido*"/>	<input type="text" value="Segundo apellido"/>
<input type="text" value="Fecha de nacimiento*"/>	<input type="text" value="Correo electrónico*"/>
<input type="text" value="Número de teléfono*"/>	

Calcular cuota

Volver
Siguiente

Madurez

1. ¿Cuáles de las funcionalidades y/o datos del prototipo podrían ser datos o funciones configurables/parametrizables?
RTA: *API KEY podría parametrizable.*

2. ¿Cuál componente del prototipo ha requerido correcciones en implementaciones anteriores?
RTA: *Calcular cuota, anteriormente el numero de identificación no era un parámetro del API GATEWAY.*

3. ¿Cuáles de los datos del prototipo requieren estabilidad para lograr la funcionalidad esperada?
RTA: *Calcular cuota, porque recoge los datos más importantes y da el valor de la cuota calculada.*

4. ¿Cuál alerta debería producir el sistema al ingresar un valor no válido en uno de los campos del prototipo con un tipo de dato específico?
RTA: *Especificar y marcar en rojo el campo mal diligenciado.*

5. ¿Cuál de las funciones del prototipo podrían presentar alteración del comportamiento normal?
RTA: *Los campos de calendario, porque tienen incompatibilidad con los navegadores regularmente. Calcular cuota, el formulario no puede continuar*

Ilustración 36. Resultado proceso de negocio originación - Formato de prototipado para RNF

Top Sales

Bajo Monto - Credibike

Crédito de montos de hasta T000.000

<input type="text" value="Valor solicitado*"/>	<input type="text" value="Cuota inicial*"/>
<input type="text" value="Número de cuotas*"/>	<input type="text" value="Valor estimado de cuota*"/>
<input type="text" value="Tipo de identificación*"/>	<input type="text" value="Número de identificación*"/>
<input type="text" value="Cédula de ciudadanía"/>	
<input type="text" value="Fecha de expedición*"/>	
<input type="text" value="Primer nombre*"/>	<input type="text" value="Segundo nombre"/>
<input type="text" value="Primer apellido*"/>	<input type="text" value="Segundo apellido"/>
<input type="text" value="Fecha de nacimiento*"/>	<input type="text" value="Correo electrónico*"/>
<input type="text" value="Número de teléfono*"/>	

Calcular cuota

Volver
Siguiente

Madurez

6. En caso de que se presenten recursos insuficientes para ejecutar la funcionalidad ¿Cómo se asegurará que el sistema siga siendo funcional? (Ejemplo: Se ofrece una versión más liviana de la aplicación, se informa el tiempo de ejecución promedio, etc.)
RTA: *Debería mostrar una alerta de que el tiempo de ejecución ha excedido, y reintentar.*

7. En caso de que los procesos de cálculo automático den resultados incorrectos ¿Cómo considera que se deban evidenciar dichos errores?
RTA: *Almacenar la respuesta para posterior auditoría.*

8. ¿En cuantos decimales espera que el sistema produzca los resultados de operaciones numéricas?
RTA: *No manejar decimales, lo aproxima al valor entero más próximo.*

9. ¿Cuándo los resultados de la funcionalidad van a ser para el negocio consistente?
RTA: *Cuando obtenemos los valores esperados.*

75

Ilustración 37. Resultado proceso de negocio originación - Formato de prototipado para RNF

Bajo Monto - Credibike

Crédito de montos de hasta 1000.000

Valor solicitado*	Cuota inicial*
Número de cuotas*	Valor estimado de cuota*
Tipo de identificación*	Número de identificación*
Cédula de ciudadanía	
Fecha de expedición*	Calcular cuota
da/mm/aaaa	
Primer nombre*	Segundo nombre
Primer apellido*	Segundo apellido
Fecha de nacimiento*	Correo electrónico*
01/05/2022	
Número de teléfono*	

Capacidad de recuperación

Restablecer estado deseado

Capacidad de recuperación

Volver **Siguiente**

Capacidad de recuperación

10. Frente a un fallo ¿Cuál considera que es la forma más transparente de comunicarle al usuario que el sistema se encuentra recuperándose? (Ejemplo: Mensaje de espera.)
RTA: *Mostrar un mensaje y una barra de progreso.*

11. Luego de una pérdida de conexión y recuperar la conexión ¿El sistema debe recargar la funcionalidad de manera automática, o informar al usuario que debe hacerlo de manera manual? (Ejemplo: Mensaje de que el usuario envíe de nuevo la petición, el sistema informa que la conexión se restableció y se está intentando de nuevo enviar la petición)
RTA: *Debería hacerlo automática.*

12. Después de un eventual fallo ¿Cómo espera que el sistema le notifique que el fallo fue superado y el sistema se encuentra disponible para su uso?
RTA: *Informa al usuario que la conectividad a sido restablecida.*

14. En caso de fallo durante el diligenciamiento y carga de información ¿Al ingresar nuevamente espera que la información consignada se muestre de nuevo?
RTA: *El formulario recarga la información diligenciada hasta el momento.*

15. En escenarios de recuperación parcial (por ejemplo, cuando se recuperan solo una parte de los datos) de los datos ¿cómo la funcionalidad debería informar que faltan datos por diligenciar o cargar?
RTA: *Al presionar el botón siguiente debe informar los campos obligatorios, marcándolos en rojo.*

Ilustración 38. Resultado proceso de negocio originación - Formato de prototipado para RNF

Bajo Monto - Credibike

Crédito de montos de hasta 1000.000

Valor solicitado*	Cuota inicial*
Número de cuotas*	Valor estimado de cuota*
Tipo de identificación*	Número de identificación*
Cédula de ciudadanía	
Fecha de expedición*	Calcular cuota
da/mm/aaaa	
Primer nombre*	Segundo nombre
Primer apellido*	Segundo apellido
Fecha de nacimiento*	Correo electrónico*
01/05/2022	
Número de teléfono*	

Volver

Siguiente

Disponibilidad

18. ¿Cuándo tiempo debe estar disponible el sistema?
RTA: *24/7*

19. ¿Cuáles evoluciones y de que tipo se pueden realizar en las funcionalidad?
RTA: *Agregar más campos al formulario.*

20. ¿Cuáles serían las funcionalidades de este prototipo que pueden ir evolucionando a través del tiempo?
RTA: *Calcular cuota y el formulario.*

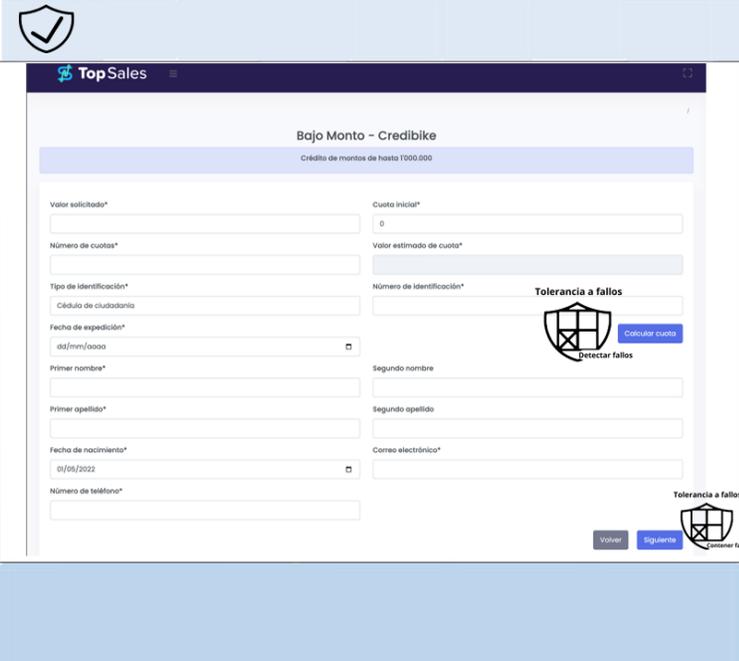
Disponibilidad

Operatividad

Disponibilidad

Longevidad

Ilustración 39. Resultado proceso de negocio originación - Formato de prototipado para RNF



The screenshot shows a web form for 'Bajo Monto - Credibike' with a sub-header 'Crédito de montas de hasta 1000.000'. The form contains several input fields: 'Valor solicitado*', 'Número de cuotas*', 'Tipo de identificación*' (with a dropdown for 'Cédula de ciudadanía'), 'Fecha de expedición*' (with a date picker), 'Primer nombre*', 'Primer apellido*', 'Fecha de nacimiento*' (with a date picker), 'Número de teléfono*', 'Cuota inicial*' (with a value of 0), 'Valor estimado de cuotas*', 'Número de identificación*', 'Segundo nombre', 'Segundo apellido', and 'Correo electrónico*'. A 'Tolerancia a fallos' section is highlighted with a red box and contains a 'Detectar fallos' button. At the bottom right, there are 'Volver', 'Siguiente', and 'Continuar' buttons.

Tolerancia a fallos

21. ¿Cómo se debe informar en el prototipo que se ha detectado una falla?
RTA: *Mostrando un mensaje de alerta.*

22. En caso de fallo ¿Se debería mostrar el nombre del componente/funcionalidad donde se localiza el fallo?
RTA: *Si, ejemplo: ocurrió un error calculando la cuota, por favor intente de nuevo.*

23. ¿Cual funcionalidad podría ser más susceptible a presentar fallos en un futuro?
RTA: *Calcular cuota.*

24. ¿Cuál funcionalidad podría generar el mayor impacto funcional en caso de un fallo?
RTA: *Calcular cuota.*

25. Al momento de un fallo, ¿Cuáles datos involucrados en el prototipado son más susceptibles a generar un mayor impacto?
RTA: *Todos los datos obligatorios.*

26. ¿Cual funcionalidad debe controlar la ejecución del proceso aun cuando el servicio falle? (Ejemplo: La funcionalidad permite seguir editando el documento sin conexión a internet.)
RTA: *Inhabilita el botón de continuar en el formulario.*

4.3.3 Resultado del uso de la plantilla “Proceso de negocio autorización de proyecto Solucredit”

Los resultados del uso de la plantilla de modelado de proceso de negocio adaptado para el proceso de negocio de autorización se pueden ver en el *Anexo I* e *Ilustración 40*.

Ilustración 40. Proceso de negocio autorización - Formato de modelado de proceso de negocio adaptado

Proceso de negocio: AUTORIZACIÓN	
Interfaces (SI/NO)	
Emisor	Receptor
SOLUCREDIT	RUBRICA
Tipos Interfaz:	
A	<input checked="" type="checkbox"/>
M	
Entradas: -Tipo de identificación. -Número de identificación. -Número telefónico. -Correo electrónico. -API Key. -ID Servicio. -ID Plantilla.	Salidas: -ID Sobre.
Normas (SI/NO) SI Nombre: -Ley 1266 de 2008. -Ley 527 de 1999 -Decreto 2364 de 2012.	Puntos normativos: -Toda la norma.

↓

PROTOTIPOS ASOCIADOS		
P3		

4.3.4 Resultado del uso de la plantilla “Prototipos de RNF proceso de negocio autorización del proyecto Solucredit”

Las *ilustraciones 41, 42, 43 y 44* corresponden al proceso de negocio de autorización, teniendo en cuenta que en el formato de prototipado no era posible ingresar todas las preguntas. En el Anexo H se podrá encontrar el formato de prototipado de RNF diligenciado.

Ilustración 41. Resultado proceso de negocio autorización - Formato de prototipado para RNF

	
	<p>Madurez</p> <p>1. ¿Cuáles de las funcionalidades y/o datos del prototipo podrían ser datos o funciones configurables/parametrizables? RTA: <i>API KEY, ID SERVICIO, ID PLANTILLA</i></p> <p>3. ¿Cuáles de los datos del prototipo requieren estabilidad para lograr la funcionalidad esperada? RTA: <i>Botón Iniciar proceso.</i></p> <p>4. ¿Cuál alerta debería producir el sistema al ingresar un valor no válido en uno de los campos del prototipo con un tipo de dato específico? RTA: <i>Especificar y marcar en rojo el campo mal diligenciado.</i></p> <p>5. ¿Cuál de las funciones del prototipo podrían presentar alteración del comportamiento normal? RTA: <i>Botón Iniciar proceso.</i></p> <p>6. En caso de que se presenten recursos insuficientes para ejecutar la funcionalidad ¿Cómo se asegurará que el sistema siga siendo funcional? (Ejemplo: Se ofrece una versión más liviana de la aplicación, se informa el tiempo de ejecución promedio, etc.) RTA: <i>Debería mostrar una alerta de que el tiempo de ejecución ha excedido, y reintentar.</i></p> <p>9. ¿Cuándo los resultados de la funcionalidad van a ser para el negocio consistente? RTA: <i>Obtener URL, ID del sobre.</i></p>
	

Ilustración 42. Resultado proceso de negocio autorización - Formato de prototipado para RNF

	
	<p>Capacidad de recuperación</p> <p>10. Frente a un fallo ¿Cuál considera que es la forma más transparente de comunicarle al usuario que el sistema se encuentra recuperándose? (Ejemplo: Mensaje de espera.) RTA: <i>Mostrar un mensaje y una barra de progreso.</i></p> <p>11. Luego de una pérdida de conexión y recuperar la conexión ¿El sistema debe recargar la funcionalidad de manera automática, o informar al usuario que debe hacerlo de manera manual? (Ejemplo: Mensaje de que el usuario envíe de nuevo la petición, el sistema informa que la conexión se restableció y se está intentando de nuevo enviar la petición) RTA: <i>Debería hacerlo automática.</i></p> <p>12. Después de un eventual fallo ¿Cómo espera que el sistema le notifique que el fallo fue superado y el sistema se encuentra disponible para su uso? RTA: <i>Informa al usuario que la conectividad a sido restablecida.</i></p> <p>15. En escenarios de recuperación parcial (por ejemplo, cuando se recuperan solo una parte de los datos) de los datos ¿cómo la funcionalidad debería informar que faltan datos por diligenciar o cargar? RTA: <i>Se debe informar que hay parámetros no configurados.</i></p>
<p>Capacidad de recuperación</p> 	<p>Capacidad de recuperación</p> 

Ilustración 43. Resultado proceso de negocio autorización - Formato de prototipado para RNF

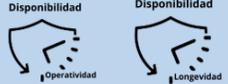
	
	<p>Disponibilidad</p> <p>18. ¿Cuándo tiempo debe estar disponible el sistema? RTA: 24/7</p> <p>19. ¿Cuáles evoluciones y de que tipo se pueden realizar en las funcionalidad? RTA: <i>Que la interfaz solicite más parámetros.</i></p> <p>20. ¿Cuáles serían las funcionalidades de este prototipo que pueden ir evolucionando a través del tiempo? RTA: <i>Los parámetros de la intertz</i></p>
	

Ilustración 44. Resultado proceso de negocio autorización - Formato de prototipado para RNF

	
	<p>Tolerancia a fallos</p> <p>21. ¿Cómo se debe informar en el prototipo que se ha detectado una falla? RTA: <i>Mostrando un mensaje de alerta.</i></p> <p>22. En caso de fallo ¿Se debería mostrar el nombre del componente/funcionalidad donde se localiza el fallo? RTA: <i>Si, ejemplo: ocurrió un error iniciando el proceso, inténtalo de nuevo.</i></p> <p>23. ¿Cual funcionalidad podría ser más susceptible a presentar fallos en un futuro? RTA: <i>Iniciar proceso.</i></p> <p>24. ¿Cuál funcionalidad podría generar el mayor impacto funcional en caso de un fallo? RTA: <i>Iniciar proceso.</i></p> <p>25. Al momento de un fallo, ¿Cuáles datos involucrados en el prototipado son más susceptibles a generar un mayor impacto? RTA: <i>Iniciar proceso.</i></p> <p>26. ¿Cual funcionalidad debe controlar la ejecución del proceso aun cuando el servicio falle? (Ejemplo: La funcionalidad permite seguir editando el documento sin conexión a internet.) RTA: <i>Inhabilita el botón de iniciar proceso.</i></p>

5.3.5 Resultado encuesta analista de requisitos

Los resultados de la encuesta realizada al analista de requisitos se pueden ver en el *Anexo J*.

Objetivo de la encuesta: Obtener información cuantitativa para validar la utilidad del proceso de referencia de la fiabilidad.

Escala de calificación para evaluar aspectos:

Tabla 18. Escala medición para evaluar aspectos

Calificación	Interpretación
1	El aspecto evaluado no se cumple en el artefacto
2	El aspecto evaluado se cumple escasamente en el artefacto
3	El aspecto evaluado se cumple parcialmente en el artefacto
4	El aspecto evaluado se cumple ampliamente en el artefacto
5	El aspecto evaluado se cumple completamente en el artefacto

Por favor califique los siguientes aspectos del proceso de referencia:

1. Grado de comprensión de los componentes.

Tabla 19. Respuesta analista de requisitos sobre aspectos

Aspectos sobre los artefactos de apoyo usados por el analista de requisitos	Calificación
La taxonomía de fiabilidad utilizada fue entendible.	5
La guía utilizada para la aplicación del proceso de referencia es clara.	5
Las preguntas utilizadas fueron suficientes para la especificación de requisitos de fiabilidad.	4
El formato de modelado de proceso de negocio adaptado es comprensible.	5
El formato de prototipado para RNF es entendible	4

2. De acuerdo con los siguientes rangos, marque con una X, cuál es su nivel de conocimiento sobre fiabilidad antes y después de usar el proceso de referencia para la fiabilidad.

Tabla 20. Escala de interpretación

Calificación	Interpretación
0% - 30%	Sin conocimiento
31% - 50%	Poco conocimiento
51% - 70%	Moderado conocimiento
71% - 100%	Mucho conocimiento

Tabla 21. Respuesta a pregunta 2 cliente

Antes	0% – 30%	31% - 50%	51%-70%	71%-100%
Después	0% – 30%	31% - 50%	51%-70%	71%-100%

3. En el momento de la aplicación del proceso de referencia ¿Que tan practico fue identificar y especificar los requisitos de fiabilidad?: En un rango de 1 a 5.

1 2 3 4 5

Nivel practico para identificar y especificar los requisitos de fiabilidad				X	
--	--	--	--	---	--

Explicación: Me hubiese gustado encontrar más ejemplos en los componentes y así poder comprender mejor.

5.3.6 Resultado encuesta cliente

Los resultados de la encuesta realizada al cliente se pueden ver en el Anexo K.

Objetivo de la encuesta: Obtener información cuantitativa para validar la utilidad del proceso de referencia de la fiabilidad.

Por favor califique los siguientes aspectos del proceso de referencia:

1. Grado de comprensión de los componentes.

Tabla 22. *Respuesta cliente sobre aspectos*

Aspectos sobre los artefactos de apoyo usados por el analista de requisitos	Calificación
Las preguntas utilizadas fueron suficientes para la especificación de requisitos de fiabilidad.	4
El formato de modelado de proceso de negocio adaptado es comprensible.	5
El formato de prototipado para RNF es entendible	4

2. De acuerdo con los siguientes rangos, marque con una X, cuál es su nivel de conocimiento sobre fiabilidad antes y después de usar el proceso de referencia para la fiabilidad.

Tabla 23. *Respuesta a pregunta 2 cliente*

Antes	0% – 30%	31% - 50%	51%-70%	71%-100%
Después	0% – 30%	31% - 50%	51%-70%	71%-100%

3. En el momento de la aplicación del proceso de referencia ¿Que tan practico fue identificar y especificar los requisitos de fiabilidad?: En un rango de 1 a 5.

1 2 3 4 5

Nivel practico para identificar y especificar los requisitos de fiabilidad			X		
--	--	--	---	--	--

Explicación: Me hubiese gustado encontrar más ejemplos en los componentes y así poder comprender mejor.

Adicional a esta información recolectada, los investigadores observaron el comportamiento de los involucrados en la intervención, entre las cuales se logró registrar las siguientes:

- Frente al proceso de referencia, al momento de diligenciar el formato de modelado de proceso de negocio adaptado, el cliente presentó dudas sobre los tipos de

interfaces que se colocaban como opción (automática, semiautomática, manual), refiriéndose durante el proceso que le parecería que se especificarán con ejemplos estos tipos de interfaces para lograr un mayor entendimiento.

- El cliente pudo responder con certeza las normas legales que regían el proceso de negocio, pero cuando pasó a diligenciar los puntos normativos presentó dudas, debido a que consideraba que la norma era muy extensa y no podía establecer cuales puntos normativos en específicos aplicaban al proceso de negocio, ya que bajo su conocimiento se aplicaba toda la norma.
- Sobre el diligenciamiento del Formato de prototipado para RNF, se pudo observar que el formato fue diligenciado de forma ordenada y sin mucho contratiempo por parte del analista de requisitos y el cliente, lo cual se le atribuye a la atención que puso el analista durante la capacitación realizada por los investigadores y lo funcional que pudo ser la guía propuesta para la aplicación del proceso de referencia.

4.4 Análisis de resultado

En esta sección se presenta el análisis de los resultados obtenidos de la evaluación del marco de trabajo para la captura y especificación de requisitos de fiabilidad a través del estudio de caso desde tres perspectivas: (i) Requisitos no funcionales obtenidos y especificados, (ii) Uso de las propiedades definidas en la taxonomía, (iii) Medidas definidas en el estudio de caso

4.4.1 Análisis de resultados desde la perspectiva de requisitos no funcionales obtenidos y especificados

La aplicación del marco de trabajo para la captura y especificación de requisitos de fiabilidad obtuvo un total de 23 requisitos de fiabilidad para el proceso de originación y un total de 19 requisitos de fiabilidad para el proceso de autorización. Estos requisitos se consideran especificados a través del instrumento Formato de prototipado para RNF, que se puede encontrar en el *Anexo H*.

La sub-característica de fiabilidad con mayor cantidad de requisitos capturados y especificados fue madurez, con un total de 15 requisitos, esta diferencia se debe a que la madurez cuenta con más componentes (3) que las otras sub-características de fiabilidad, las cuales cuentan con 2

componentes cada una; esto, hace que la madurez tenga más probabilidad de tener un mayor número de propiedades.

La sub-característica de tolerancia a fallos tuvo un total de 12 requisitos capturados y especificados, aunque esta sub-característica tiene solo 4 propiedades se lograron derivar 6 preguntas, las cuales tuvieron un porcentaje de respuesta del 100%, permitiendo que se ubicara como la segunda sub-característica con más requisitos obtenidos y especificados. De igual forma, la característica de capacidad de recuperación tiene 4 propiedades, derivando 6 preguntas de estas, siendo así la tercer sub-característica con más requisitos pese a que en ambos procesos de negocio no se contestaron la totalidad de las preguntas, teniendo un total de 9 requisitos capturados y especificados. Por otro lado, la sub-característica con menor porcentaje de requisitos capturados y especificados es disponibilidad, lo cual se pudo dar porque esta sub-característica solo tiene 5 preguntas relacionadas, de las cuales solo se contestaron 3, ya que el cliente determinó que 2 de ellas no aplicaban para el proceso de negocio y prototipo. La relación entre propiedades definidas en la taxonomía, preguntas relacionadas a las propiedades, se puede observar en la **Ilustración 45**.

Del mismo modo, a partir de estos resultados se logra consolidar la distribución de cantidad de requisitos por cada sub-característica, la cual se puede observar en la **Ilustración 46**.

Ilustración 45. Relación entre preguntas planteadas vs. Numero de requisitos capturados

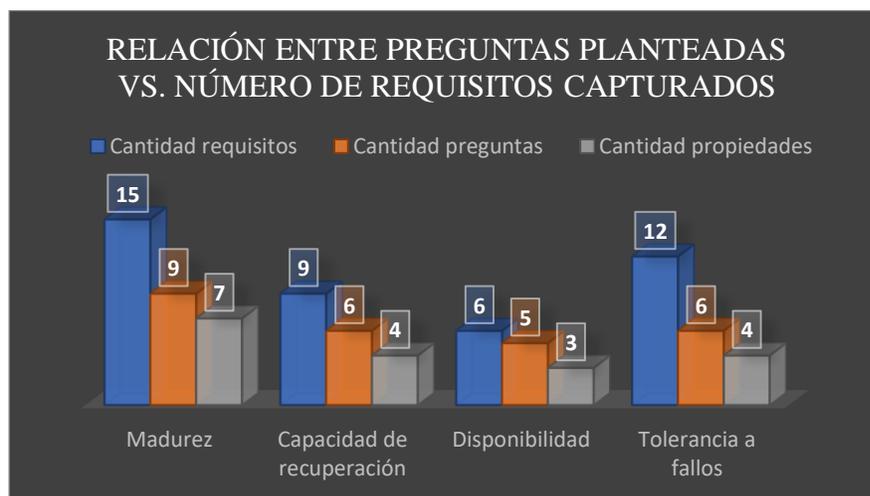
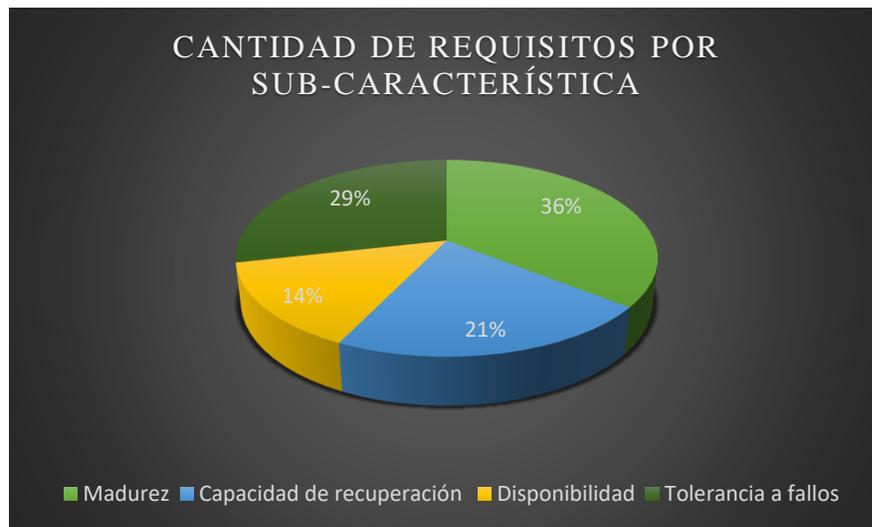


Ilustración 46. Cantidad de requisitos por sub-característica.



4.4.2 Análisis de resultados desde la perspectiva del uso de las propiedades definidas en la taxonomía

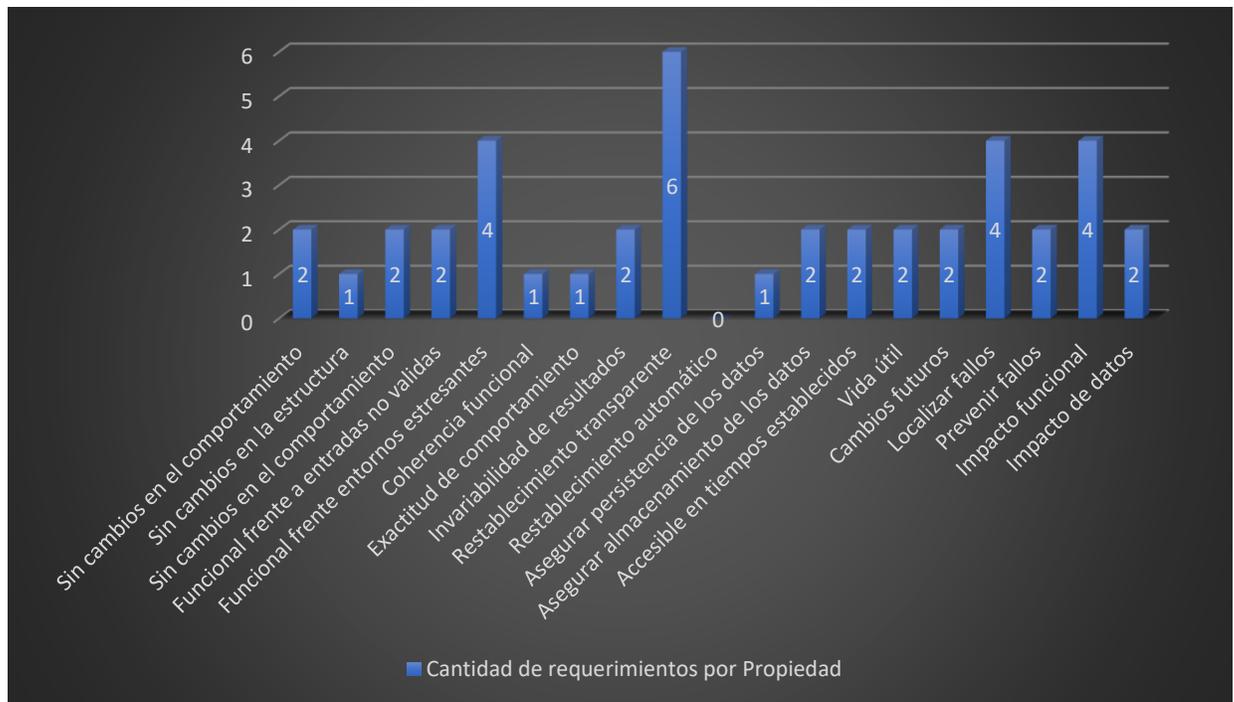
A nivel de propiedades de componentes de sub-características de fiabilidad, se encontró que restablecimiento automático, el cual contaba con solo una pregunta (¿Cuál de las funcionalidades debería tener la capacidad de reconfigurarse automáticamente frente un fallo?), no logró tener ningún requisito especificado para los dos procesos de negocio donde se aplicó el proceso de referencia. El cliente consultó sobre un ejemplo de reconfiguración automática para poder aplicarla a los procesos de negocio y finalmente indicó que no aplicaba para ningún escenario. Por lo anterior, se interpretó que una mejora importante para el proceso de referencia pudiera ser acompañar a las preguntas con ejemplos, de tal forma que el cliente pueda entender de una manera más clara en qué contexto del proceso de negocio se puede aplicar esta propiedad.

Se encontró casos como el de la propiedad de accesible en tiempos establecidos perteneciente a la sub-característica de disponibilidad, a pesar de ser una de las propiedades con mayor número de preguntas derivadas con un total de 3 preguntas, durante la aplicación del proceso de referencia solo se contestó 1 de ellas. En el caso anterior, se pudo observar cómo alguna respuesta a una pregunta puede invalidar otras preguntas dependiendo del contexto donde se aplique el marco de trabajo.

La propiedad de restablecimiento transparente obtuvo respuesta a todas las preguntas derivadas de ella, lo cual supone que las preguntas fueron claras para el cliente y fueron fáciles de aplicar en los

procesos de negocio para obtener respuestas de valor que sirvan para la etapa de diseño de software. Haciendo un contraste entre la *Ilustración 45* y la *Ilustración 47*, se puede observar que la mayoría de las propiedades cumplen con una cantidad moderada de respuestas que van de 2 a 4 requisitos, por lo que se puede presenciar que el marco de trabajo para la captura y especificación de requisitos de fiabilidad fue útil para el proyecto.

Ilustración 47. Cantidad de requerimientos por Propiedad



4.4.3 Análisis de resultados desde la perspectiva de medidas definidas en el estudio de caso

Con el objetivo de comprobar los aspectos medibles establecidos (nivel de claridad, nivel de practicidad) en el estudio de caso, se describen los puntos analizados a partir de la información obtenida a través de los artefactos utilizados durante la intervención:

Análisis de resultados de las encuestas a analista de requisitos y cliente

A partir de la información obtenida a través de las encuestas realizadas al analista de requisitos y al cliente del proyecto se pudo evidenciar, y de manera unánime, que:

1. Frente a la suficiencia de las preguntas utilizadas para la especificación de requisitos de fiabilidad, este artefacto cumple ampliamente.

2. Frente a la comprensibilidad del formato de modelado de proceso de negocio adaptado el artefacto cumple completamente.
3. Frente a qué tan entendible es el formato de prototipado para RNF, el artefacto cumple ampliamente.

De otro lado, en las dos preguntas adicionales que se le realizó al analista de requisitos, para la evaluación preliminar de la taxonomía y la guía para la aplicación del proceso de referencia, estos elementos de la propuesta cumplen completamente.

En cuanto al nivel de conocimiento de fiabilidad antes de la aplicación de la propuesta, el analista indico tener “Poco conocimiento” (31%-50%) y el cliente indico tener un nivel “Sin conocimiento” (0%-30%). En cuanto al nivel de conocimiento de fiabilidad después de la aplicación de la propuesta, el analista y el cliente coincidieron en tener “Moderado conocimiento” (51%-70%). Se pudo lograr un aumento importante en el entendimiento de la característica de fiabilidad para los involucrados en el proyecto a través de la taxonomía y el proceso de referencia que componen el marco de trabajo.

Frente al nivel de practicidad del proceso de referencia, el analista indico que este proceso cumple ampliamente con el objetivo de capturar y especificar requisitos de fiabilidad. Por el contrario, el cliente indico que este proceso cumple parcialmente con este objetivo, explicando que para lograr una mayor practicidad del proceso de referencia se sugieren incluir más ejemplos en las preguntas.

4.4.4 Respuestas a las preguntas de investigación

Después de analizar los resultados obtenidos de la evaluación del marco de trabajo para capturar y especificar requisitos de fiabilidad, se procede a responder las preguntas de investigación planteadas en el estudio de caso.

Pregunta 2: ¿Los artefactos que componen el proceso de referencia son comprensibles al ser aplicados?

Respuesta: Los artefactos que componen el proceso de referencia para capturar y especificar requisitos de fiabilidad, son altamente comprensibles. Este grado, se sustenta en las respuestas dadas por analista y el cliente frente a concluir que el formato de modelado de proceso de negocio adaptado fue completamente comprensible, mientras que el formato de prototipado para RNF fue

ampliamente entendible. De igual forma, el analista de requisitos concluyó que la taxonomía de fiabilidad y la guía para la aplicación del proceso de referencia fueron completamente comprensibles. Por último, frente al conocimiento adquirido sobre la fiabilidad después de aplicar el proceso de referencia para capturar y especificar requisitos de fiabilidad, tanto el analista de requisitos como el cliente concluyeron que habían adquirido un conocimiento moderado de la característica en comparación a los conocimientos que tenían de esta antes de la aplicación de la propuesta.

Pregunta 3: ¿El proceso de referencia permite a los interesados identificar y solicitar requisitos de fiabilidad?

Respuesta: El proceso de referencia permitió de manera adecuada identificar y solicitar requisitos de fiabilidad. Este grado, se sustenta en las respuestas dadas por el analista de requisitos y el cliente, quienes concluyeron que las preguntas utilizadas durante la etapa de especificación de requisitos de fiabilidad fueron ampliamente suficientes. De la igual manera, el analista de requisitos concluyó que el proceso de referencia a nivel de practicidad cumple ampliamente, en comparación con el cliente que concluyó que cumple parcialmente. Lo anterior se puede interpretar como la suficiencia del proceso de referencia para cumplir el objetivo de identificar y solicitar requisitos de fiabilidad, aun así, los involucrados consideraron que el proceso se podría mejorar a nivel del artefacto de Formato de modelado de proceso de negocio adaptado, incluyendo ejemplos de los tipos de interfaces, así mismo como mejoras en las preguntas de fiabilidad de manera que sirvieran como soporte o guía para la especificación de los requisitos, esto último aportando que la estructura de las preguntas serían más entendibles si se aplicaran ejemplos concretos a cada una de ellas de tal forma que pudieran aterrizar los conceptos técnicos de la pregunta a un entorno más palpable a nivel del producto software.

A partir de las respuestas a las preguntas adicionales, se concluye que el marco de trabajo para la captura y especificación de requisitos de fiabilidad es útil para los proyectos de desarrollo de software. Los resultados de la evaluación preliminar sobre los elementos que constituyen el marco de trabajo alcanzaron un nivel de cumple completamente con el objetivo de la evaluación. Esta utilidad se puede observar en la cantidad de requisitos de fiabilidad capturados y especificados al aplicar el marco de trabajo, donde se obtuvieron un total de 42 requisitos entre los dos procesos de negocio donde se aplicó la propuesta.

4.4.5 Puntos de mejora

- Al evaluar la claridad de las preguntas, el cliente expresó que cada pregunta debería tener un conjunto de ejemplos que le permitieran identificar con mayor precisión el enfoque que debe darle a la respuesta.
- Durante el diligenciamiento del formato de proceso de negocio adaptado, el cliente expresó que desearía mirar ejemplos sobre los tipos de interfaces que se encuentran expresados en la plantilla como automática, semiautomática y manual, de esta forma poder identificar a que tipo de interfaz pertenece las interfaces el proceso de negocio.

Estos dos puntos de mejora serán considerados en trabajos futuros de la propuesta, debido a que consideramos importante su inclusión.

4.4.6 Análisis de validez de la propuesta

El marco de trabajo para capturar y especificar requisitos de fiabilidad supuso diferentes riesgos internos y externos, los cuales debieron ser gestionados de la siguiente forma.

El marco de trabajo para capturar y especificar requisitos de fiabilidad pretendía ofrecer un proceso útil para las organizaciones de desarrollo software, el cual le permitiera identificar, capturar y especificar requisitos de fiabilidad de una forma práctica. Para esto, se buscó dar la validez interna, asegurando lo anterior con el desarrollo de una propuesta taxonómica de fiabilidad, que estuviera compuesta por niveles y definiciones que permitieran a los involucrados comprender conceptos de fiabilidad a un nivel más bajo, y les permitiera lograr una captura y especificación de requisitos de fiabilidad más precisa y útil. Esta taxonomía fue socializada a los participantes del proyecto.

De igual manera, en cuanto a la validez externa, se consideraron aspectos externos que podrían afectar la aplicación del marco de trabajo; se tomó la decisión de que la organización escogida tuviera experiencia mínima de 5 años desarrollando productos software y tuviera algo de conocimiento sobre la ISO/IEC 25010, además se planteó un proceso de referencia enmarcado dentro de las actividades comunes de captura y especificación de requisitos reconocidos por la industria del software, incluyendo los artefactos que apoyaran de forma familiar la aplicación del marco de trabajo.

El estudio de caso buscaba medir la utilidad del marco de trabajo para capturar y especificar requisitos de fiabilidad, haciendo uso de la propuesta taxonómica de fiabilidad, proceso de

referencia, artefactos de captura y artefactos de especificación. Los artefactos de medición definidos permitieron capturar datos en términos de claridad y practicidad para su posterior análisis.

4.4.7 Limitaciones

Las limitaciones encontradas durante la aplicación del marco de trabajo para capturar y especificar requisitos de fiabilidad fueron las siguientes: (i) Se presentaron limitaciones de tiempo con el equipo técnico de Namtrik debido a que solo disponían de pocas horas para aplicar el proceso de referencia de fiabilidad al proyecto Solucredit. (ii) La aplicación del marco de trabajo para la captura y especificación de requisitos de fiabilidad solo se realizó con un proyecto, por lo tanto, los resultados no se pueden generalizar para la industria del software. (iii) El poco conocimiento sobre la temática de la propuesta por parte de la empresa y sus participantes, no permitió desarrollar las actividades con la fluidez esperada.

Capítulo 5 – Conclusiones, lecciones aprendidas y trabajos futuros

En este capítulo se presentan las conclusiones describiendo los puntos más importantes, así como las lecciones aprendidas y por ultimo los trabajos futuros a realizar para el trabajo.

5.1 Conclusiones

Para este trabajo de investigación se definió como objetivo proponer un marco de trabajo que utilice elementos conceptuales y metodológicos que apoyen la captura y especificación de requisitos asociados a la fiabilidad del producto software; para ello, se realizó una revisión bibliográfica del tema objeto de estudio, de la cual se pudo concluir que la información asociada al tema de fiabilidad ha sido poco trabajada, ya que los registros encontrados son escasos. A continuación, se presentan los puntos clave que permitieron concluir el trabajo de investigación.

- Pese a la poca información que existe sobre la característica de fiabilidad, se logró extender la definición de la característica en términos taxonómicos y semánticos, permitiendo proponer dos niveles (componentes y propiedades) sobre la jerarquización establecida por la ISO/IEC 25010, además de extender la definición de fiabilidad y los elementos que la componen. De dicha forma, se alcanzó un entendimiento más detallado de la característica de fiabilidad.
- Utilizar la taxonomía de fiabilidad propuesta en este trabajo de investigación como insumo para la derivación de preguntas utilizadas en la etapa especificación de requisitos de fiabilidad, permitió especificar una cantidad de requisitos de fiabilidad importante durante la ejecución del estudio de caso. Este resultado demostró que la propuesta taxonómica siendo un producto conceptual, pudo ser aplicado de manera útil en el proceso de referencia que incluye la propuesta y obtener de esta manera resultados satisfactorios.
- Aunque la madurez fue una de las sub-características con menor información obtenida durante el proceso conceptual realizado al momento de la definición taxonómica, aun así, se logró el mayor número de propiedades y componentes en comparación con las otras sub características.
- Realizar una evaluación del marco de trabajo para capturar y especificar requisitos de fiabilidad por medio de un estudio de caso aplicado a una empresa de desarrollo de

software, permitió comprobar la utilidad del marco en términos de la claridad y practicidad de la taxonomía de fiabilidad y el proceso de referencia.

- El uso de artefactos como la taxonomía, prototipos, preguntas, formato RNF facilita que la captura de requisitos no funcionales sea más fácil para el cliente, ya que le permite aterrizar los conceptos de fiabilidad de una forma más práctica y guiada.
- Aplicar el estudio de caso permitió encontrar puntos de mejora al marco de trabajo, los cuales se esperan ser abarcados en trabajos futuros, con el objetivo de mejorarlo, y de esta manera poder obtener un mayor volumen de resultados de su aplicación en contextos reales.
- Se pudo concluir que la construcción de propuestas de investigación conceptuales y metodológicas podrían ayudar a las empresas desarrolladoras de software en la mejora de sus procesos de captura y especificación de requisitos.

5.2 Lecciones aprendidas

Durante la realización de este trabajo de investigación se lograron las siguientes lecciones aprendidas.

- La investigación realizada para proponer la taxonomía de fiabilidad permitió ampliar nuestro conocimiento sobre esta característica de calidad y todos los elementos que la componen, además de entender su influencia en otras características de calidad del software.
- Utilizar el producto del trabajo conceptual realizado para extender la característica de fiabilidad con el proceso de referencia para la captura y especificación de requisitos, permitió mostrar cómo los procesos de investigación basados en la literatura pueden aplicarse de forma útil en entornos reales.
- Se considera viable la aplicación de esta metodología para el desarrollo de otras características que hacen parte de la ISO/IEC 25010, lo que contribuirá a capturar y especificar requisitos que aporten valor al producto software y satisfagan las necesidades del cliente.
- Se considera que, para la aplicación de los productos de investigación en entornos reales, se debe tener plena disponibilidad de los involucrados, de tal forma que los procesos se realicen de forma consciente y tranquila, para obtener resultados más concretos y fiables.

- Como investigadores fue necesario durante toda la ejecución del estudio de caso tener una alta claridad sobre los elementos construidos, para lograr una ejecución adecuada a la evaluación de la propuesta.

5.3 Trabajos futuros

En el trabajo de investigación se determinaron algunos puntos que pueden ser tomados en cuenta para trabajos futuros, como lo son:

- De acuerdo con la sugerencia expresada por los interesados, se acompañará a todas las preguntas con un ejemplo para que estas sean más fáciles de aplicar en sus futuros proyectos.
- Tomando en cuenta que la validación preliminar fue realizada solo en una empresa, se espera aplicar la propuesta en un conjunto más amplio de empresas, para continuar validando la utilidad del marco de trabajo para la captura y especificación de requisitos de fiabilidad.

Referencias

- [1] I. Derbel, L. L. Jilani, and A. Mili, "A model for analyzing architectural attributes," in *ACS/IEEE International Conference on Computer Systems and Applications-AICCSA 2010*, 2010, pp. 1-7.
- [2] ISO/IEC, "ISO/IEC 25010: 2011 Systems and software engineering--Systems and software Quality Requirements and Evaluation (SQuaRE)--System and software quality models," ed: CH: ISO Geneva, 2011.
- [3] A. Andrews and J. Lucente, "Predicting incident reports for IT help desk," in *44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2014*, 2014, pp. 678-683.
- [4] W. Lee, J. K. Lee, and J. Baik, "Software reliability prediction for open source software adoption systems based on early lifecycle measurements," in *35th Annual IEEE International Computer Software and Applications Conference, COMPSAC 2011*, Munich, 2011, pp. 366-371.
- [5] H. B. Duygulu and O. Tosun, "An algorithm for software reliability growth model selection," in *Informatics 2008 and Data Mining 2008, MCCSIS'08 - IADIS Multi Conference on Computer Science and Information Systems*, Amsterdam, 2008, pp. 115-122.
- [6] L. Pessoa, P. Fernandes, T. Castro, V. Alves, G. N. Rodrigues, and H. Carvalho, "Building reliable and maintainable dynamic software product lines: An investigation in the body sensor network domain," *Information and Software Technology*, vol. 86, pp. 54-70, 2017.
- [7] R. S. Hanmer, D. T. McBride, and V. B. Mendiratta, "Comparing reliability and security: Concepts, requirements, and techniques," *Bell Labs Technical Journal*, vol. 12, pp. 65-78, 2007.
- [8] F. Febrero, C. Calero, and M. Á. Moraga, "Software reliability modeling based on ISO/IEC SQuaRE," *Information and Software Technology*, vol. 70, pp. 18-29, 2016.
- [9] Y. Jabareen, "Building a conceptual framework: philosophy, definitions, and procedure," *International journal of qualitative methods*, vol. 8, pp. 49-62, 2009.
- [10] F. J. Pino, M. Piattini, and G. Horta Travassos, "Managing and developing distributed research projects in software engineering by means of action-research," *Revista Facultad de Ingeniería Universidad de Antioquia*, pp. 61-74, 2013.
- [11] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empirical software engineering*, vol. 14, pp. 6-7, 2009.
- [12] I. ISO, "ISO 9001:2015 Quality management systems — Requirements," vol. 5, ed, 2015, p. 29.
- [13] M. Glinz, "On non-functional requirements," in *15th IEEE international requirements engineering conference (RE 2007)*, 2007, pp. 21-26.
- [14] I. ISO, "ISO 25000:2005. (2005). ISO/IEC 25000:2005. Available: <https://iso25000.com/index.php/normas-iso-25000?limit=4&limitstart=0>
- [15] I. ISO, IEEE 29148, "Systems and software engineering—Life cycle processes—Requirements engineering," ed: International Organization for Standardization, 2018.
- [16] F. J. P. Félix O. García Rubio, Ignacio García Rodríguez de Guzmán, Mario G. Piattini Velthuis, *Calidad de sistemas de información. 4ª edición ampliada y actualizada*, 2018, p. 696.
- [17] T. M. Hughes, "SAS® Data Analytic Development: Dimensions of Software Quality," *SAS® Data Analytic Development: Dimensions of Software Quality*, p. 37, 2016.
- [18] L. Borodaev, A. Telea, R. Groenboom, and R. Smedinga, "Software metrics for policy-driven software development life cycle automation," in *11th IEEE International Conference on Software Testing, Verification and Validation Workshops, ICSTW 2018*, 2018, pp. 169-174.

- [19] T. Fujiwara and S. Yamada, "A new analysis concept in application of software reliability growth models," in *15th ISSAT International Conference on Reliability and Quality in Design*, San Francisco, CA, 2009, pp. 270-274.
- [20] L. Bernstein, "Software Fault Tolerance Forestalls Crashes: To Err Is Human; To Forgive Is Fault Tolerant," in *Advances in Computers* vol. 58, ed, 2003, pp. 239-286.
- [21] I. V. Kovalev, D. I. Kovalev, V. S. Chefonov, N. A. Testodvov, A. A. Koltyshev, and A. G. Krivogornitsyn, "The development and reliability analysis environment of fault-tolerance multiversion software," in *2nd International Scientific Conference on Advanced Technologies in Aerospace, Mechanical and Automation Engineering, MIST: Aerospace 2019*, 2020.
- [22] I. I. J. S. S. a. s. engineering, "ISO/IEC 33004:2015 Information technology — Process assessment — Requirements for process reference, process assessment and maturity models," in *ISO*, ed, 2015, p. 9.
- [23] S. o. C. S. a. M. Software Engineering Group, Keele University, Keele, Staffs,UK, "Guidelines for performing Systematic Literature Reviews in Software Engineering," vol. Version 2.3, p. 6, 2007.
- [24] P. K. Kapur, R. B. Garg, U. Chanda, and A. Tandon, "Development of software reliability growth model incorporating enhancement of features and related release policy," *International Journal of Systems Assurance Engineering and Management*, vol. 1, pp. 52-58, 2010.
- [25] P. Rotella and S. Chulani, "Predicting release reliability," in *2017 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, 2017, pp. 39-46.
- [26] I. Tariq, T. B. Maqsood, B. Hayat, K. Hameed, M. Nasir, and M. Jahangir, "The comprehensive study on software reliability," in *2018 International Conference on Computing, Mathematics and Engineering Technologies, iCoMET 2018*, 2018, pp. 1-7.
- [27] F. Urem and Z. Mikulić, "Using NHPP model for ERP software module reliability growth," in *33rd International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2010*, Opatija, 2010, pp. 414-417.
- [28] K. Meridji, K. T. Al-Sarayreh, and A. Al-Khasawneh, "A generic model for the specification of software reliability requirements and measurement of their functional size," *International Journal of Information Quality*, vol. 3, pp. 139-163, 2013.
- [29] S. Boranbayev, A. Boranbayev, S. Altayev, and Y. Seitkulov, "Application of diversity method for reliability of cloud computing," in *2014 IEEE 8th International Conference on Application of Information and Communication Technologies (AICT)*, 2014, pp. 1-5.
- [30] F. Brosch, B. Buhnova, H. Koziolk, and R. Reussner, "Reliability prediction for fault-tolerant software architectures," in *7th Int. ACM SIGSOFT Conf. on the Quality of Software Archit., QoSA 2011 and the 2nd ACM SIGSOFT Int. Symp. on Archit. Critical Syst., ISARCS 2011, Part of the 2011 Federated Events on Component-Based Software Eng. and Software Archit., CompArch'11*, Boulder, CO, 2011, pp. 75-84.
- [31] "IEEE Recommended Practice on Software Reliability," *IEEE Std 1633-2016 (Revision of IEEE Std 1633-2008)*, pp. 1-261, 2017.
- [32] H. C. Kim and H. K. Park, "The comparative study for ENHPP software reliability growth model based on mixture coverage function," in *International Conference on Grid and Distributed Computing, GDC 2011, Held as Part of the 3rd International Mega-Conference on Future-Generation Information Technology, FGIT 2011* vol. 261 CCIS, ed. Jeju Island, 2011, pp. 187-194.
- [33] A. J. Molnar, A. Neamțu, and S. Motogna, "Longitudinal evaluation of software quality metrics in open-source applications," in *14th International Conference on Evaluation of Novel Approaches to Software Engineering, ENASE 2019*, 2019, pp. 80-91.

- [34] S. L. B. Ruiz, "Framework Para La Elicitación De Requisitos No Funcionales Usando Su Representación," Universidad del cauca, 2021.
- [35] R. K. Yin, "Case study research: Design and methods (applied social research methods)," *Case study research design and methods*, vol. 34, 2008.