

Framework de gestión de requisitos no funcionales basado en Double Diamond: estudio de caso en una institución pública de educación superior

Non-Functional Requirements Management Framework Based on the Double Diamond: A Case Study in a Public Higher Education Institution

ZORAIDA YANET VIDAL MELGAREJO^{1*}  | LUISA ANGÉLICA OREJUELA GUERRERO DE MOSQUERA² 

Afiliación:

¹Escuela de Ingeniería de Sistemas, Universidad Nacional de Trujillo, La libertad, Perú

²Escuela de Ingeniería Industrial, Universidad Nacional de Trujillo, La libertad, Perú

Información del artículo:

Recibido: 10/09/2026

Aceptado: 02/10/2026

Publicado: 12/10/2026

Autor de correspondencia: E-mail: *zvidal@unitru.edu.pe

Resumen

La gestión de requisitos no funcionales (RNF) resulta crítica para asegurar atributos de calidad en los sistemas de información universitarios. El objetivo fue diseñar, implementar y evaluar el *framework Double Diamond–Non-Functional Requirements* (DD-NFR) en una institución pública de educación superior. DD-NFR se implementó como una aplicación web modular que integra la elicitación de múltiples fuentes, el análisis de conflictos entre requisitos funcionales y no funcionales, la priorización mediante el proceso analítico jerárquico (AHP) y la generación de especificaciones técnicas. El *framework* se evaluó mediante un estudio de caso en una universidad pública de Lima, con 15 participantes vinculados al desarrollo y la gestión de software y tres sesiones de 4 h. La completitud de los RNF pasó de 38,0 % a 82,0 %; los conflictos RF-RNF identificados y confirmados, de 4 a 27; el tiempo promedio de gestión, de 48,3 a 25,6 min por RNF; las especificaciones completas, de 12,0 % a 91,0 %; y la satisfacción, de 2,8 a 4,3 sobre 5. En el caso evaluado, DD-NFR se asoció con mayor cobertura documental, mejor trazabilidad y una formalización más consistente de los RNF, sin que estos resultados impliquen una generalización causal a otras instituciones.

Palabras clave: calidad del software; ingeniería de requisitos; priorización multicriterio; trazabilidad de requisitos.

Abstract

The management of non-functional requirements (NFRs) is critical for ensuring quality attributes in university information systems. This study aimed to design, implement, and evaluate the Double Diamond–Non-Functional Requirements (DD-NFR) framework in a public higher education institution. DD-NFR was implemented as a modular web application that integrates multi-source elicitation, conflict analysis between functional and non-functional requirements, prioritization using the Analytic Hierarchy Process (AHP), and technical specification generation. The framework was evaluated through a case study at a public university in Lima, involving 15 participants engaged in software development and management across three 4-hour sessions. NFR completeness increased from 38.0 % to 82.0 %; identified and confirmed FR-NFR conflicts increased from 4 to 27; average management time decreased from 48.3 to 25.6 min per NFR; complete specifications increased from 12.0 % to 91.0 %; and satisfaction increased from 2.8 to 4.3 out of 5. In the evaluated case, DD-NFR was associated with greater documentary coverage, improved traceability, and more consistent formalization of NFRs, without implying a causal generalization of these results to other institutions.

Keywords: software quality; requirements engineering; multicriteria prioritization; requirements traceability.

1. Introducción

Los sistemas de información sostienen procesos académicos, administrativos y estratégicos en las instituciones públicas de educación superior. En estos entornos, el cumplimiento funcional no resulta suficiente, ya que el software también debe satisfacer atributos de calidad como seguridad, rendimiento, disponibilidad, usabilidad, mantenibilidad, interoperabilidad, trazabilidad y cumplimiento normativo. Estas condiciones se expresan mediante requisitos no funcionales (RNF), los cuales establecen cómo debe comportarse el sistema y bajo qué criterios debe operar para ser aceptable, confiable y sostenible, además de orientar las decisiones arquitectónicas y operativas. Cuando los RNF se incorporan de manera tardía, ambigua o incompleta, pueden trasladar problemas hacia el diseño, las pruebas y el mantenimiento (Chung et al., 2000; Dongmo, 2024; Gondal et al., 2020).

La gestión de los RNF presenta dificultades relacionadas con la comprensión compartida entre los actores, su documentación y su articulación con los requisitos funcionales. Werner et al. (2020) muestran que la falta de entendimiento común afecta la coordinación. Por su parte, Eckhardt et al. (2016) evidencian que numerosos RNF describen comportamientos verificables, lo que dificulta establecer una separación rígida entre requisitos funcionales y no funcionales. Asimismo, Behutiye et al. (2017) y Robiolo et al. (2019) documentan problemas de registro de RNF y su relación con la deuda técnica, particularmente en contextos ágiles donde las restricciones de calidad pueden quedar insuficientemente especificadas.

La formulación de los RNF también requiere que sus condiciones puedan ser observadas y verificadas. Expresiones como “el sistema debe ser rápido”, “debe ser seguro” o “debe estar siempre disponible” no establecen por sí mismas umbrales, métricas ni mecanismos de prueba. Por ello, su gestión requiere relacionar cada requisito con su fuente, categoría, requisito funcional afectado y criterio técnico de verificación. La trazabilidad permite conservar estas relaciones y facilitar el seguimiento de los requisitos a través de los artefactos del desarrollo (Yoo et al., 2024; Santos

et al., 2024). En este sentido, la documentación no debe limitarse a identificar términos asociados con atributos de calidad, sino mantener una cadena de información desde la fuente documental hasta la especificación final (Behutiye et al., 2017; Robiolo et al., 2019).

La priorización constituye una actividad relevante cuando existen restricciones de tiempo y recursos, ya que permite ordenar los RNF según diferentes criterios de importancia. Entre las técnicas disponibles, el Analytic Hierarchy Process (AHP) permite establecer prioridades mediante comparaciones por pares y comprobar la coherencia de los juicios mediante la razón de consistencia (Saaty, 1980; Bukhsh et al., 2020). En la gestión de RNF, esta actividad puede involucrar dependencias, conflictos y diferentes criterios de importancia, por lo que debe articularse con su identificación y relación con los demás requisitos, en lugar de tratarse como una actividad aislada.

La automatización de la ingeniería de requisitos ha evolucionado hacia técnicas de aprendizaje automático y procesamiento de lenguaje natural. Li y Nong (2022) desarrollaron una aproximación para la clasificación automática de RNF, y Rosado da Cruz y Cruz (2025) identifican aplicaciones de inteligencia artificial en actividades como elicitación, clasificación, priorización, especificación y trazabilidad. Sin embargo, la automatización no elimina la participación del analista, debido a que la interpretación de un RNF depende del contexto institucional y de las relaciones que mantiene con otros requisitos. En DD-NFR, la detección automática funciona como mecanismo de apoyo, de modo que el analista puede confirmar, corregir, descartar o incorporar manualmente los requisitos identificados. Yamamoto (2024) muestra, además, que los RNF pueden emerger del análisis de excepciones funcionales. De Martino y Palomba (2025), por su parte, evidencian que su clasificación y gestión continúan planteando desafíos incluso en sistemas apoyados por aprendizaje automático.

A partir de la revisión realizada, se identificó una brecha de integración. Los estudios disponibles suelen abordar de manera independiente actividades como la documentación, clasificación, trazabilidad,



priorización o especificación de RNF. Dentro del alcance revisado, no se identificó una solución operacionalizada que integrara en un mismo flujo la elicitación multi-fuente, la identificación y relación de RNF con requisitos funcionales, el análisis de conflictos RF-RNF, la priorización mediante AHP y la generación de especificaciones técnicas verificables para el contexto de sistemas universitarios públicos. Esta brecha evidencia la necesidad de una solución que articule estas actividades dentro de un proceso trazable, en lugar de limitarse a una etapa específica de la ingeniería de requisitos.

Para responder a esta necesidad, se desarrolló DD-NFR, un framework de gestión de requisitos no funcionales basado en Double Diamond. Este modelo organiza el flujo de trabajo mediante las fases de descubrir, definir, desarrollar y entregar, y Design Science Research orientó la construcción y evaluación del artefacto (Design Council, 2025; Hehn & Mendez, 2022; Peffers et al., 2007). Esta combinación permitió estructurar la exploración de fuentes, la depuración de RNF y conflictos, la priorización y la producción de especificaciones verificables, incorporando mecanismos de automatización como apoyo al trabajo del analista. En consecuencia, el objetivo general fue diseñar, implementar y evaluar un framework de gestión de requisitos no funcionales basado en Double Diamond mediante un estudio de caso en una institución pública de educación superior. Como objetivos específicos se plantearon: diseñar la arquitectura del framework; implementar sus módulos como una aplicación web; evaluar su desempeño mediante indicadores de completitud, conflictos, tiempo de gestión, especificaciones técnicas, satisfacción de los usuarios y consistencia de la priorización; y contrastar sus funcionalidades con enfoques alternativos documentados en la literatura.

2. Metodología

2.1. Diseño del estudio de caso

Se adoptó un estudio de caso único para examinar el funcionamiento de DD-NFR en su contexto real de uso. El caso correspondió a la Oficina de Tecnologías de la Información de una

universidad pública de Lima y se desarrolló durante marzo de 2025, en el marco de un proyecto institucional de sistema de gestión académica. El diseño permitió observar el proceso de gestión de RNF antes y durante la aplicación del framework, manteniendo el foco en el artefacto, sus módulos y los indicadores definidos para su evaluación. Las unidades de observación fueron los RNF gestionados, los conflictos RF-RNF confirmados, las especificaciones elaboradas, los resultados del AHP y las valoraciones de los 15 participantes. El desarrollo del artefacto se articuló con las fases de Double Diamond y la lógica de Design Science Research, con el propósito de vincular el problema, el diseño, la implementación, la demostración y la evaluación contextual del framework.

2.2. Indicadores y criterios de evaluación

La evaluación del framework se realizó mediante seis indicadores observables: completitud de RNF identificados, conflictos RF-RNF confirmados, tiempo promedio de gestión por requisito, especificaciones técnicas completas, satisfacción de los usuarios y razón de consistencia del proceso AHP. Estos indicadores permitieron valorar la cobertura, la capacidad de análisis, la eficiencia operativa, la formalización documental, la aceptación de los usuarios y la coherencia de la priorización, como se muestra en la Tabla 1. La población de estudio estuvo conformada por los actores institucionales vinculados con la identificación, análisis, priorización, documentación o validación de requisitos de software en instituciones públicas de educación superior. Entre estos actores se consideraron analistas de sistemas, responsables de tecnologías de la información, especialistas funcionales, usuarios clave y personal relacionado con el desarrollo o gestión de sistemas institucionales. Como fuentes de información complementarias, se utilizaron documentos normativos, catálogos de requisitos, especificaciones funcionales, formatos técnicos y registros de proyectos institucionales.

La muestra estuvo integrada por 15 participantes vinculados con actividades de análisis, desarrollo, validación, gestión o uso de sistemas institucionales. Se aplicó un muestreo

no probabilístico intencional, considerando criterios de pertinencia técnica y funcional. Se incluyó a quienes contaban con experiencia en proyectos de software institucional, conocimiento del proceso evaluado, responsabilidad funcional o técnica y disponibilidad para participar en las sesiones de validación. Se excluyó a las personas que no tenían relación directa con la gestión de requisitos o los sistemas institucionales.

2.3. Instrumentos y calidad de la medición

Se emplearon una ficha de registro de indicadores y un cuestionario de satisfacción con escala Likert de cinco puntos (1 = totalmente en desacuerdo, 2 = en desacuerdo, 3 = ni de acuerdo ni en desacuerdo, 4 = de acuerdo y 5 = totalmente de acuerdo). La ficha registró por sesión los documentos procesados, candidatos detectados, RNF confirmados e ingresados manualmente, conflictos, tiempos de gestión, especificaciones generadas y razón de consistencia AHP. Incluyó fecha y hora, participantes, proyecto evaluado, número de documentos, candidatos detectados automáticamente, RNF confirmados, RNF ingresados manualmente, conflictos sugeridos, conflictos confirmados y falsos positivos, tiempo total, RNF priorizados, razón de consistencia (CR), especificaciones completadas y observaciones. Esta estructura permitió reconstruir cada sesión y distinguir las sugerencias automáticas de las decisiones confirmadas por los analistas.

La validez de contenido de los instrumentos fue revisada por especialistas con experiencia en ingeniería de requisitos, desarrollo de software y gestión de tecnologías de la información, considerando los criterios de claridad, pertinencia, coherencia y suficiencia. El cuestionario estuvo conformado por 10 ítems: (1) El framework DD-NFR facilita la identificación de requisitos no funcionales; (2) el módulo de elicitación multi-fuente permite revisar documentos institucionales de manera ordenada; (3) la clasificación de requisitos no funcionales por categorías resulta clara y útil; (4) la matriz de conflictos ayuda a identificar tensiones entre requisitos funcionales y no funcionales; (5) el framework permite mejorar la trazabilidad de los requisitos no funcionales; (6)

el módulo de priorización AHP ayuda a ordenar los requisitos según su importancia institucional; (7) el proceso de comparación por pares del método AHP resulta comprensible para el usuario; (8) las especificaciones técnicas generadas por el framework son claras y completas; (9) los reportes generados en PDF o Excel son útiles para documentar el proceso de requisitos; y (10) el framework DD-NFR puede aplicarse en proyectos reales de software institucional. Durante la aplicación se mantuvo la revisión humana en cada etapa. Los participantes confirmaron, corrigieron o descartaron candidatos; los analistas y usuarios clave revisaron los conflictos; y la coherencia de la priorización se verificó mediante la razón de consistencia de Saaty (1980).

2.4. Desarrollo e implementación del framework DD-NFR

El framework DD-NFR se implementó como una aplicación web modular. El frontend se desarrolló con Streamlit y el backend con Python. La lógica de negocio utilizó pandas y NumPy; el almacenamiento se realizó en PostgreSQL, y las utilidades incluyeron procesamiento básico de lenguaje natural, generación de PDF y exportación a Excel. El flujo conserva la trazabilidad entre las fuentes documentales, los RNF identificados, los requisitos funcionales, los conflictos, las priorizaciones y las especificaciones, como se muestra en la Figura 1. La arquitectura evidencia que el framework no opera como un clasificador aislado, sino como un flujo integrado, en el que cada módulo produce información verificable que se almacena en PostgreSQL y sirve de entrada al módulo siguiente. La correspondencia entre las fases de Double Diamond y los módulos implementados se resume en la Tabla 2. El módulo de elicitación utilizó expresiones regulares y un diccionario de palabras clave organizado en cinco categorías: seguridad, rendimiento, disponibilidad, usabilidad y normativo. Los candidatos fueron revisados por los analistas antes de ser confirmados, de modo que la automatización funcionó como apoyo y no como sustitución del juicio experto. Las categorías y palabras clave empleadas durante la elicitación se detallan en la Tabla 3.



Tabla 1

Indicadores y criterios de evaluación del framework DD-NFR

Indicador	Definición operativa	Unidad o criterio
Complejidad de RNF	Proporción de RNF identificados respecto del total de RNF validados.	Porcentaje (%)
Conflictos RF-RNF	Número de conflictos entre requisitos funcionales y no funcionales confirmados por los analistas.	Frecuencia (n)
Tiempo de gestión	Tiempo promedio requerido para gestionar cada RNF.	min por RNF
Especificaciones completas	Proporción de RNF priorizados que cuentan con todos los campos de la ficha técnica.	Porcentaje (%)
Satisfacción	Valoración promedio de utilidad, facilidad de uso, claridad, trazabilidad y aplicabilidad.	Escala Likert de 1 a 5
Consistencia AHP	Coherencia de los juicios emitidos en las comparaciones por pares.	CR < 0,10

Figura 1

Arquitectura del framework DD-NFR y sus componentes principales

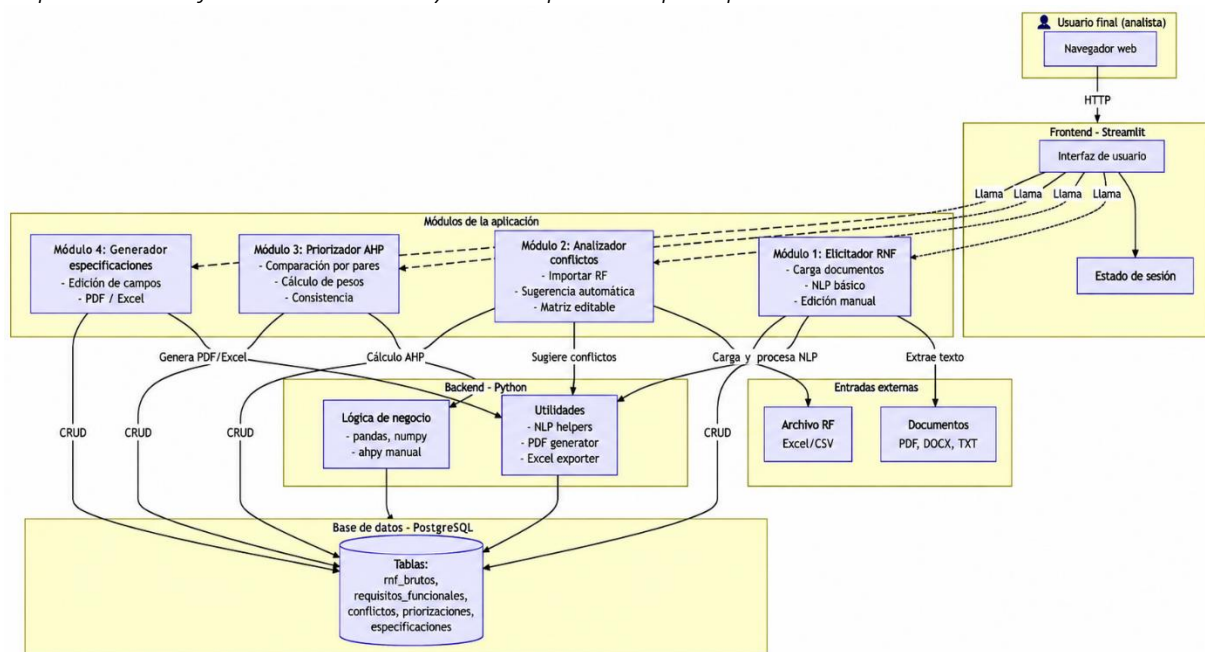


Tabla 2

Correspondencia entre las fases de Double Diamond y los módulos de DD-NFR

Fase	Definición operativa	Función
Descubrir	Módulo 1: Elicitador RNF	Carga documentos, aplica reglas de NLP básico, sugiere categorías y admite edición manual.
Definir	Módulo 2: Analizador de conflictos	Importa requisitos funcionales, sugiere conflictos y permite validarlos en una matriz editable.
Desarrollar	Módulo 3: Priorizador AHP	Ejecuta comparaciones por pares, calcula pesos, ranking y consistencia.
Entregar	Módulo 4: Generador de especificaciones	Completa campos técnicos y genera reportes PDF y Excel.

Tabla 3

Diccionario de palabras clave implementado en el módulo de elicitación

Categoría	Palabras clave incluidas	Norma o fuente de referencia
Seguridad	Cifrado, autenticación, autorización, confidencialidad, integridad, auditoría	ISO/IEC 27001; NTP-ISO/IEC 17799
Rendimiento	Tiempo de respuesta, latencia, throughput, carga, concurrente, rápido	ISO/IEC 25010
Disponibilidad	Uptime, disponibilidad, caída, failover, alta disponibilidad	ISO/IEC 25010; SLA institucional
Usabilidad	Accesibilidad, WCAG, interfaz, clics, intuitivo, experiencia de usuario	WCAG 2.1; ISO 9241-11
Normativo	SUNEDU, ley, directiva, cumplimiento, norma, estándar	Ley Universitaria N.º 30220; directivas SUNEDU

El módulo de conflictos relacionó requisitos funcionales y RNF mediante una matriz que registró el tipo de conflicto, la severidad, la solución y el responsable. El módulo AHP utilizó matrices de comparación por pares y calculó el vector de prioridades mediante la media geométrica de las filas. Posteriormente, obtuvo $\lambda_{\max}$, el índice de consistencia y la razón de consistencia. Las sesiones se limitaron a un máximo de 10 RNF para mantener las comparaciones cognitivamente manejables.

El generador de especificaciones transformó los RNF priorizados en fichas técnicas que incluyeron identificador, categoría, descripción, criterio de aceptación SMART, métrica, herramienta de verificación, prioridad AHP, responsable y observaciones. La información podía exportarse a PDF y Excel.

2.5. Procedimiento de evaluación y análisis de datos

El procedimiento comprendió ocho etapas: revisión del problema y de la literatura; definición de requisitos del artefacto; implementación del elicitor multi-fuente; implementación del analizador de conflictos; desarrollo del priorizador AHP; construcción del generador de especificaciones; evaluación mediante estudio de caso; y organización e interpretación de los datos. La evaluación se realizó en tres sesiones de 4 h cada una. Antes de la intervención se recuperaron indicadores del procedimiento convencional. En cada sesión se registraron los documentos procesados, candidatos detectados, RNF confirmados e ingresados manualmente, conflictos sugeridos, confirmados y descartados,

tiempo, RNF priorizados, CR, especificaciones completadas y observaciones. Al cierre se aplicó el cuestionario de satisfacción.

El análisis fue descriptivo y comparativo. Se calcularon frecuencias, porcentajes, promedios, diferencias en puntos porcentuales y variaciones relativas. Para completitud, tiempo, especificaciones, satisfacción y CR, el valor posterior correspondió al promedio de las tres sesiones; para los conflictos se utilizó el total acumulado confirmado al cierre de la tercera sesión. El análisis cualitativo se centró en patrones recurrentes de utilidad, trazabilidad, claridad, curva de aprendizaje y aplicabilidad.

La construcción del artefacto combinó el modelo Double Diamond con la lógica de Design Science Research. Design Council (2025) organiza el proceso en descubrir, definir, desarrollar y entregar, mientras que Peffers et al. (2007) estructuran la investigación de diseño mediante identificación del problema, definición de objetivos, diseño y desarrollo, demostración, evaluación y comunicación. La fase descubrir permitió reunir documentos institucionales y necesidades de los actores; definir concentró la depuración de RNF y la identificación de conflictos; desarrollar incorporó la comparación por pares y el cálculo AHP; y entregar transformó los requisitos priorizados en especificaciones técnicas. Esta correspondencia permitió convertir el modelo de diseño en una secuencia operativa implementada mediante módulos.

Muntean y Militaru (2022) señalan que Design Science Research resulta apropiado cuando se construyen soluciones tecnológicas para problemas prácticos y, al mismo tiempo, se



genera conocimiento evaluable. Kroop (2025) advierte que la existencia del artefacto no basta para sustentar su utilidad y adecuación contextual; es necesario reunir evidencia sobre utilidad, coherencia y aplicabilidad. Por ello, la evaluación de DD-NFR incluyó medidas relacionadas con cobertura, conflictos detectados, eficiencia temporal, completitud documental, satisfacción y consistencia de la priorización. Estas medidas permiten observar tanto el resultado técnico como la experiencia de los participantes durante el uso del framework.

El módulo AHP organizó la decisión mediante comparaciones por pares. Saaty (1980) establece que los juicios deben representarse en una matriz recíproca y que la coherencia puede evaluarse mediante el índice y la razón de consistencia. En la investigación se consideró aceptable una razón de consistencia inferior a 0,10. El procedimiento calculó el vector de prioridades mediante la media geométrica de las filas, normalizó los pesos y obtuvo el valor máximo característico aproximado para determinar el índice de consistencia. Esta verificación es importante porque una jerarquización numérica puede parecer precisa aun cuando los juicios de los participantes sean contradictorios.

Bukhsh et al. (2020) identifican AHP como una de las técnicas más utilizadas y precisas para priorizar requisitos, aunque advierten limitaciones de escalabilidad cuando aumenta el número de elementos. Akbar et al. (2021) muestran su utilidad para jerarquizar barreras de ingeniería de requisitos en problemas complejos. Jamasb et al. (2025) confirman que la priorización continúa orientándose hacia marcos automatizados o semiautomatizados. En DD-NFR se mitigó la carga de comparación mediante conjuntos previamente clasificados y criterios institucionales explícitos, evitando aplicar matrices excesivamente grandes. La herramienta no reemplazó el juicio experto, sino que lo estructuró, registró y sometió a una comprobación de consistencia.

2.6. Reproducibilidad y tratamiento de la información

La reproducibilidad se fortaleció mediante la conservación de las entradas, los resultados

intermedios y las salidas de cada módulo. Los RNF brutos se almacenaron con identificador, descripción, fuente, fecha, categoría preliminar y fase de Double Diamond. Los conflictos conservaron el requisito funcional asociado, los RNF afectados, tipo, severidad, solución y responsable. Las priorizaciones registraron peso global, posición y fecha, mientras que las especificaciones se almacenaron en formato estructurado. Esta organización permitió reconstruir el recorrido de un requisito desde su detección hasta su documentación técnica final. El estudio trabajó con documentos y participantes vinculados a procesos institucionales de software. La información se utilizó exclusivamente para evaluar el framework y se reportó de manera agregada, sin atribuir respuestas ni resultados individuales. La identidad de la institución y de los participantes se mantuvo reservada, omitiendo identificadores personales e institucionales.

3. Resultados

3.1. Funcionamiento de los módulos

El módulo 1 permitió procesar documentos PDF, DOCX y TXT, extraer texto y detectar candidatos a RNF según el diccionario implementado. Cada candidato conservó información sobre descripción, categoría preliminar, fuente y fase de Double Diamond. La revisión humana permitió confirmar, editar o descartar los resultados, así como incorporar manualmente RNF no detectados. El módulo 2 generó una matriz de conflictos entre requisitos funcionales y no funcionales. Los analistas pudieron registrar la severidad, describir la incompatibilidad, proponer una solución y asignar un responsable. Esta funcionalidad hizo explícitas relaciones que, antes del framework, permanecían dispersas o no documentadas.

El módulo 3 produjo rankings de prioridad y verificó la coherencia de los juicios. En el ejemplo documentado, el requisito de disponibilidad mínima de 99,5 % obtuvo el mayor peso global (0,312), seguido del almacenamiento seguro de contraseñas (0,278), el tiempo máximo de respuesta de 3 s (0,191), el cumplimiento de WCAG 2.1 nivel AA (0,143) y los formatos

aceptados por SUNEDU (0,076). La razón de consistencia fue 0,07, como se observa en la Tabla 4. El módulo 4 generó especificaciones técnicas estructuradas para los RNF priorizados. La incorporación de criterios de aceptación, métricas y mecanismos de verificación permitió transformar enunciados generales de calidad en condiciones más precisas y utilizables por analistas, desarrolladores y responsables de control de calidad.

La visualización conjunta de la Figura 2 muestra que el incremento de la completitud y de las especificaciones técnicas ocurrió paralelamente a la reducción del tiempo de gestión. Esta relación descriptiva respalda la utilidad operativa del flujo integrado, aunque no constituye por sí sola una demostración causal ni reemplaza una evaluación inferencial con observaciones individuales.

3.2. Resultados de la evaluación

La completitud alcanzó 76,0 % en la primera sesión, 82,0 % en la segunda y 88,0 % en la tercera. El valor postintervención consolidado fue de 82,0 %, obtenido mediante el promedio de las tres sesiones. Los conflictos RF-RNF confirmados aumentaron de manera acumulativa hasta alcanzar 27 casos al finalizar la tercera sesión. El tiempo promedio por RNF disminuyó de 29,4 min en la primera sesión a 25,6 min en la segunda y 21,8 min en la tercera; el valor consolidado postintervención fue de 25,6 min. Las especificaciones técnicas completas alcanzaron 84,0 %, 91,0 % y 98,0 %, respectivamente. La satisfacción evolucionó de 4,1 a 4,3 y 4,5 puntos, con un promedio consolidado de 4,3 sobre 5. Las razones de consistencia fueron 0,08, 0,06 y 0,06, respectivamente, todas inferiores al límite de 0,10.

Los valores consolidados se calcularon mediante la media aritmética de las tres sesiones para completitud, tiempo, especificaciones, satisfacción y CR; para los conflictos se empleó el total acumulado confirmado al cierre de la tercera sesión. En consecuencia, los resultados deben interpretarse considerando tanto la progresión por sesión como los valores

consolidados, y no como estimaciones poblacionales. La comparación de los indicadores pre y postintervención se presenta en la Tabla 5. La completitud se incrementó en 44 puntos porcentuales y la proporción de especificaciones completas en 79 puntos porcentuales. El tiempo promedio disminuyó 22,7 min por requisito. Los conflictos identificados pasaron de 4 a 27, lo que refleja una mayor capacidad para hacer visibles las tensiones RF-RNF, no un incremento de problemas generados por el framework.

3.3. Percepción de los participantes

Los participantes valoraron favorablemente la organización del proceso, la trazabilidad entre la fuente, la categoría, el conflicto, la prioridad y la especificación, así como la posibilidad de generar entregables en PDF y Excel. El módulo de elicitación fue considerado útil para reducir omisiones, mientras que la matriz de conflictos facilitó la discusión entre los actores técnicos y funcionales. Asimismo, el generador de especificaciones fue identificado como uno de los componentes con mayor valor práctico. La principal dificultad percibida fue la curva de aprendizaje del módulo AHP, especialmente por las comparaciones por pares. Los participantes sugirieron incorporar una interfaz más guiada, ejemplos y mecanismos de apoyo para facilitar la formulación de juicios.

El incremento de la completitud desde 38,0 % hasta 82,0 % representa una ganancia absoluta de 44 puntos porcentuales y una mejora relativa aproximada de 115,8 % respecto del valor inicial. Este resultado sugiere que el procedimiento asistido permitió recuperar una proporción considerablemente mayor de criterios de calidad que la práctica previa. No obstante, la completitud no alcanzó 100 %. Alrededor del 18 % de los RNF debió incorporarse manualmente, lo cual confirma que el diccionario de palabras clave funciona como mecanismo de apoyo y no como sustituto del análisis experto. La revisión humana fue necesaria para interpretar expresiones implícitas, términos dependientes del dominio y requisitos formulados sin vocabulario técnico explícito.



Tabla 4

Ranking de requisitos no funcionales generado mediante AHP

Ranking	Requisito no funcional	Categoría	Peso global
1	El sistema debe garantizar disponibilidad mínima del 99,5 % en horario hábil	Disponibilidad	0,312
2	Las contraseñas deben almacenarse con cifrado bcrypt o equivalente	Seguridad	0,278
3	El tiempo de respuesta para consultas de matrícula no debe exceder 3 s	Rendimiento	0,191
4	El sistema debe cumplir con los estándares WCAG 2.1 nivel AA	Usabilidad	0,143
5	Los reportes deben generarse en formatos aceptados por SUNEDU	Normativo	0,076

Figura 1

Evolución de la completitud de RNF, especificaciones técnicas y tiempo promedio según sesión

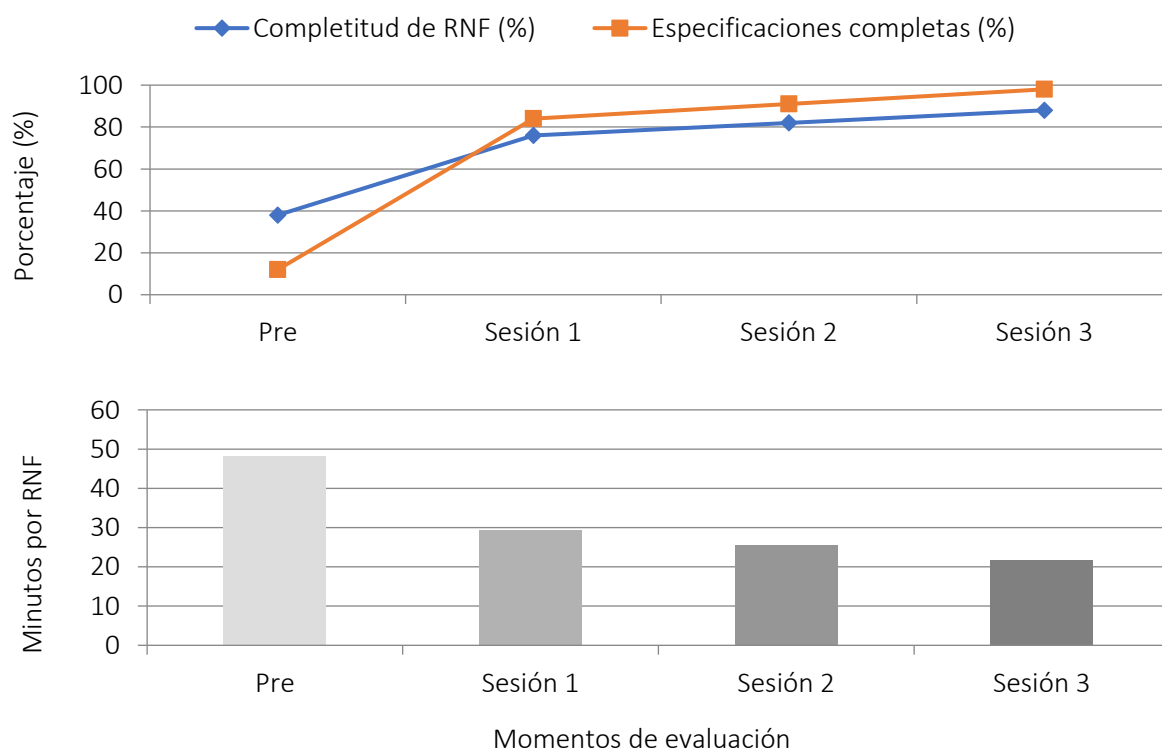


Tabla 5

Comparación de indicadores antes y después de la intervención

Indicador	Pre intervención	Post intervención	Variación (%)
Completitud de RNF identificados (%)	38,0	82,0	+115,8
Conflictos RF-RNF detectados (N)	4	27	+575,0
Tiempo de gestión por RNF (min)	48,3	25,6	-47,0
Especificaciones técnicas completas (%)	12,0	91,0	+658,3
Satisfacción de usuarios (1-5)	2,8	4,3	+53,6
Razón de consistencia AHP	No aplica	0,07	Aceptable(<0,10)

Tabla 6

Comparación del framework DD-NFR con enfoques alternativos

Dimensión	DD-NFR	Literatura		
		Che Embi y Keong (2022)	Santos et al. (2024)	Chung et al. (2000)
Elicitación multi-fuente	Sí: PDF, DOCX, TXT y registro manual	Parcial	No es su eje	Conceptual
Análisis de conflictos	Matriz RF-RNF validada	No explícito	Parcial	Relaciones entre softgoals
Priorización	AHP y razón de consistencia	No central	No central	Razonamiento cualitativo
Especificación técnica	Fichas y reportes PDF/Excel	Documentación general	Escenarios verificables	Modelado conceptual
Trazabilidad	Fuente-RNF-conflicto-prioridad-especificación	Parcial	Parcial	Parcial
Soporte para IPES públicas	Específico	No específico	No específico	No específico
Implementación web	Sí	No	No necesariamente	No

Los conflictos confirmados aumentaron de 4 a 27. Este cambio no debe interpretarse como un deterioro de los requisitos, sino como una mayor capacidad para visibilizar incompatibilidades previamente no documentadas. La matriz permitió asociar cada conflicto con severidad, propuesta de solución y responsable, convirtiendo hallazgos dispersos en elementos gestionables. La identificación temprana es relevante porque un RNF puede restringir una funcionalidad, elevar su costo o exigir decisiones arquitectónicas específicas. En el contexto evaluado, el resultado es compatible con una mayor capacidad para hacer explícitas las incompatibilidades del proceso de requisitos.

El tiempo promedio disminuyó de 48,3 a 25,6 min por requisito, equivalente a una reducción de 22,7 min y 47,0 %. Esta reducción observada se relaciona con la centralización del flujo, la reutilización de estructuras y la generación automática de salidas. El indicador no significa que todo RNF pueda gestionarse en 25,6 min en cualquier proyecto, sino que expresa el promedio obtenido bajo las condiciones del caso. La reducción coexistió con un aumento de la completitud y de la documentación, por lo que no se produjo a costa de omitir actividades, sino mediante la reorganización del proceso.

Las especificaciones completas pasaron de 12,0 % a 91,0 %, con una ganancia absoluta de

79 puntos porcentuales. Este fue el cambio proporcional más amplio de los indicadores porcentuales. La plantilla obligó a completar campos relacionados con descripción, criterio de aceptación, métrica, herramienta de verificación, responsable y trazabilidad. En el caso analizado, el framework redujo la dependencia de descripciones abiertas y produjo entregables más próximos a la implementación y a las pruebas. La satisfacción aumentó de 2,8 a 4,3 sobre 5, lo que sugiere una valoración favorable de utilidad, claridad y facilidad de uso.

La razón de consistencia consolidada de 0,07 se mantuvo por debajo del umbral de 0,10, lo que indica una coherencia aceptable en las comparaciones AHP realizadas en las sesiones. La curva de aprendizaje observada en las comparaciones por pares sugiere la conveniencia de interfaces más guiadas cuando participan usuarios con poca experiencia en AHP.

3.4. Estabilidad operativa entre sesiones

La evaluación en tres sesiones permitió observar la evolución del uso. La completitud avanzó de 76,0 % a 82,0 % y 88,0 %, mientras que las especificaciones completas aumentaron de 84,0 % a 91,0 % y 98,0 %. La satisfacción progresó de 4,1 a 4,3 y 4,5. Esta tendencia ascendente es compatible con un proceso de familiarización de los participantes y ajuste operativo de la



herramienta. Al mismo tiempo, aconseja interpretar el promedio consolidado junto con los valores de cada sesión, ya que parte del rendimiento puede estar asociado al aprendizaje de uso.

Los resultados de las sesiones no evidenciaron una pérdida de consistencia en AHP. Por el contrario, la razón de consistencia se mantuvo dentro del límite aceptable. La arquitectura modular también permitió que los participantes verificaran los productos parciales antes de continuar con el proceso, incluidos los candidatos a RNF, la matriz de conflictos, el ranking y las especificaciones. Este control por etapas contribuyó a reducir la posibilidad de que un error inicial se propagara sin revisión hasta el documento final.

4. Discusión

Los resultados muestran que DD-NFR contribuyó a organizar actividades que antes se realizaban de forma fragmentada. La mejora de la completitud y de las especificaciones técnicas es coherente con Behutiye et al. (2017) y Che Embi y Keong (2022), quienes señalan que la documentación de RNF continúa siendo problemática, sobre todo cuando se privilegia la entrega funcional y no se definen criterios transversales de calidad. El incremento de conflictos confirmados debe interpretarse como una mejora de visibilidad y análisis, no como un deterioro del proceso. Eckhardt et al. (2016) cuestionan la separación estricta entre requisitos funcionales y no funcionales debido a que muchos RNF expresan comportamientos observables. En DD-NFR, la matriz de conflictos operacionaliza esta relación mediante el registro de incompatibilidades, severidad, soluciones y responsables.

La trazabilidad obtenida coincide con la necesidad descrita por Yoo et al. (2024) de conservar vínculos entre requisitos y artefactos posteriores. El framework extiende esta relación al conectar la fuente documental, el RNF, los conflictos, el peso AHP y la especificación técnica. Este flujo resulta especialmente relevante en organizaciones públicas sujetas a auditorías, normativas y continuidad operativa. El módulo AHP produjo una razón de consistencia

consolidada de 0,07. Este valor cumple el criterio de Saaty (1980) y respalda la coherencia de las comparaciones realizadas. El hallazgo también es consistente con Bukhsh et al. (2020), quienes reconocen AHP como una técnica precisa y frecuente para priorizar requisitos. Sin embargo, la curva de aprendizaje y el límite de hasta 10 RNF por sesión confirman la advertencia de escalabilidad señalada en dicha revisión.

La reducción del tiempo promedio de 48,3 a 25,6 min sugiere una ganancia operativa asociada a la centralización de tareas, la reutilización de categorías, el apoyo automático en la elicitación y la generación estructurada de entregables. No obstante, al tratarse de un estudio de caso único y no haberse aplicado una prueba inferencial, esta diferencia debe interpretarse como evidencia contextual y no como una estimación causal generalizable. En comparación con otros enfoques, Che Embi y Keong (2022) se concentran en la documentación de RNF en metodologías ágiles, Santos et al. (2024) plantean escenarios BDD para hacer verificables las condiciones de calidad y Chung et al. (2000) ofrecen una base conceptual mediante softgoals. DD-NFR integra estos intereses dentro de una aplicación web e incorpora además elicitación multi-fuente, conflictos RF-RNF, priorización AHP y generación de documentos técnicos. La comparación de estas funcionalidades se presenta en la Tabla 6.

4.1. Contraste con la literatura y alcance del aporte

Los hallazgos coinciden con Dongmo (2024) y Gondal et al. (2020), quienes plantean que los RNF deben incorporarse tempranamente y mantenerse vinculados al proceso de desarrollo. La mejora de la completitud observada respalda la utilidad de un mecanismo sistemático de búsqueda y registro. Sin embargo, el porcentaje de incorporación manual confirma lo señalado por Werner et al. (2020), quienes sostienen que la comprensión compartida de los RNF depende de la interacción entre actores y no puede reducirse a una operación automática. Por esta razón, DD-NFR debe entenderse como un sistema de apoyo a la decisión y documentación, no como un reemplazo del analista.

El aumento de especificaciones completas guarda relación con Santos et al. (2024), quienes destacan la necesidad de expresar las condiciones de calidad de forma observable y verificable. También responde a la problemática descrita por Robiolo et al. (2019), Behutiye et al. (2017) y Che Embi y Keong (2022) respecto de la documentación insuficiente de RNF. El aporte del módulo generador reside en convertir un requisito priorizado en una ficha con información técnica mínima para su implementación y prueba. Esto fortalece la trazabilidad, pero no garantiza por sí mismo que el sistema final cumpla el requisito. La verificación debe continuar durante el desarrollo, las pruebas y la operación.

La integración con Double Diamond aporta una lógica centrada en exploración, definición, desarrollo y entrega. Hehn y Mendez (2022) sostienen que Design Thinking puede complementar la ingeniería de requisitos cuando se busca comprender necesidades y construir soluciones centradas en las personas. Fatima et al. (2024) también resaltan el valor de la participación en actividades de requisitos. En el framework, esta orientación se materializó en la revisión de fuentes, la confirmación de candidatos, la discusión de conflictos y la comparación por pares. Por tanto, Double Diamond no se utilizó únicamente como denominación del modelo, sino como estructura para ordenar la participación y los productos de cada fase.

En relación con la priorización, los resultados son coherentes con Bukhsh et al. (2020), Akbar et al. (2021) y Felfernig et al. (2018), quienes muestran que el valor, impacto y utilidad para los interesados deben formar parte de la decisión. La razón de consistencia permitió distinguir entre un ranking numérico y un ranking sustentado en juicios razonablemente coherentes. La principal precaución es la escalabilidad. Si se comparan demasiados RNF, el número de pares crece rápidamente. Por ello, la clasificación previa y la selección de conjuntos manejables son condiciones necesarias para conservar la viabilidad del método.

Patrício et al. (2025) proponen integrar la ingeniería de requisitos con prácticas posteriores del ciclo de vida, evitando que los requisitos

permanezcan aislados en documentos iniciales. DD-NFR avanza en esa dirección al producir especificaciones técnicas con métricas y mecanismos de verificación. Asimismo, Yoo et al. (2024) señalan que la trazabilidad debe vincular requisitos con artefactos técnicos. Aunque el framework no implementa todavía una integración completa con repositorios de código, pruebas automatizadas o herramientas DevOps, su modelo de datos establece una base para futuras conexiones.

4.2. Implicancias prácticas para instituciones públicas de educación superior

La aplicación del framework puede contribuir a estandarizar actividades que suelen depender de documentos dispersos, hojas de cálculo o experiencia individual. En una institución pública, la formalización de los RNF puede facilitar la continuidad de proyectos ante cambios de personal, mejorar la comunicación entre oficinas usuarias y equipos técnicos y proporcionar evidencia para revisiones de calidad o auditorías. La trazabilidad desde la fuente hasta la especificación permite justificar por qué se incorporó un requisito, qué proceso afecta, quién debe atenderlo y cómo se verificará. Esto resulta particularmente relevante cuando los sistemas deben cumplir políticas institucionales, disposiciones de seguridad, niveles de servicio y restricciones presupuestarias.

La utilidad práctica también depende de que el framework se incorpore a los procedimientos existentes y no se utilice como una herramienta aislada. Se recomienda definir responsables para confirmar candidatos, revisar conflictos, participar en comparaciones AHP y aprobar especificaciones. Asimismo, los criterios de priorización deben acordarse antes de evaluar los requisitos, con el fin de evitar que el ranking refleje preferencias circunstanciales. El resultado del AHP debe interpretarse como apoyo estructurado a la decisión y no como una sustitución automática de la deliberación técnica e institucional.

Desde la perspectiva de gestión, los indicadores utilizados ofrecen una base para monitorear la madurez del proceso. La completitud permite observar la cobertura de identificación, los conflictos revelan la capacidad



de anticipar incompatibilidades, el tiempo promedio aproxima la eficiencia, las especificaciones completas reflejan el nivel de formalización y la satisfacción recoge la percepción de los usuarios. El seguimiento periódico de estos indicadores podría ayudar a identificar retrocesos, comparar proyectos y orientar mejoras del framework. No obstante, cada institución debería adaptar las metas y los umbrales a su tamaño, criticidad de los sistemas y disponibilidad de recursos.

4.3. Limitaciones del estudio

Una limitación específica del módulo de elicitación es su dependencia de expresiones regulares y diccionarios. Durante el estudio de caso, aproximadamente el 18 % de los RNF finalmente incluidos debieron registrarse manualmente porque no fueron detectados automáticamente. Por tanto, el motor puede generar falsos negativos ante vocabulario técnico no previsto. Este resultado respalda la necesidad de combinar automatización y revisión humana.

Otra limitación corresponde al alcance de la evaluación, que se realizó en una sola institución y con 15 participantes seleccionados intencionalmente. La muestra resulta informativa para el estudio de caso, pero no permite inferir que los mismos resultados se reproduzcan en otras universidades. Asimismo, no se aplicó una prueba inferencial pre-post ni un diseño experimental de control, por lo que las diferencias observadas no establecen causalidad. Además, el cuestionario de satisfacción se interpretó de manera descriptiva y no se estimó consistencia interna debido al tamaño muestral. Estas condiciones limitan el alcance de los resultados y deben considerarse al interpretar la evidencia obtenida.

4.4. Líneas futuras de investigación

Las restricciones identificadas justifican evaluar DD-NFR en múltiples instituciones, proyectos y equipos con diferentes niveles de experiencia. También resulta pertinente incorporar técnicas de procesamiento de lenguaje natural para mejorar la identificación de RNF sin eliminar la confirmación humana. Asimismo, estudios posteriores podrán emplear diseños

comparativos y muestras mayores para examinar la transferibilidad del framework a otros escenarios institucionales. También será pertinente ampliar su integración con repositorios, pruebas automatizadas y herramientas DevOps, de modo que pueda fortalecerse la continuidad de la trazabilidad a lo largo del ciclo de vida del software.

5. Conclusiones

El framework DD-NFR integró en un flujo único la elicitación multi-fuente, el análisis de conflictos entre requisitos funcionales y no funcionales, la priorización mediante AHP y la generación de especificaciones técnicas. En el caso estudiado, la correspondencia entre estos módulos y las fases descubrir, definir, desarrollar y entregar permitió operacionalizar el modelo Double Diamond dentro de un proceso trazable de gestión de RNF. La completitud de RNF pasó de 38,0 % a 82,0 %, las especificaciones técnicas completas aumentaron de 12,0 % a 91,0 % y el tiempo promedio de gestión disminuyó de 48,3 a 25,6 min por RNF. Asimismo, se confirmaron 27 conflictos RF-RNF y la satisfacción promedio alcanzó 4,3 sobre 5. Estos resultados describen el comportamiento del framework en las condiciones del caso estudiado.

La razón de consistencia consolidada de 0,07 evidenció que los juicios empleados en la priorización se mantuvieron dentro del límite aceptable. Las comparaciones por pares requirieron un proceso de familiarización, especialmente entre los participantes con menor experiencia en AHP. El principal aporte del framework fue establecer la trazabilidad entre las fuentes documentales, los RNF identificados, los conflictos, las prioridades y las especificaciones técnicas. Esta articulación permitió organizar el proceso de gestión de RNF mediante módulos relacionados y productos verificables en cada etapa.

Contribución de los autores

Z. Y. Vidal: Conceptualización, curación de datos, análisis formal, adquisición de fondos, investigación, metodología, administración del proyecto, recursos, software, validación,

visualización, redacción del borrador original, revisión y edición. L. A. Orejuela: Investigación, metodología, supervisión, validación, redacción del borrador original, revisión y edición.

Conflictos de interés

Los autores manifiestan no tener conflictos de interés.

6. Referencias bibliográficas

- Achimugu, P., Selamat, A., Ibrahim, R., y Mahrin, M. N. (2014). A systematic literature review of software requirements prioritization research. *Information and Software Technology*, 56(6), 568–585. <https://doi.org/10.1016/j.infsof.2014.02.001>
- Akbar, M. A., Naveed, W., Mahmood, S., Rafi, S., Alsanad, A., Abdul-Aziz Alsanad, A., Gumaei, A., y Alothaim, A. (2021). Prioritization of global software requirements' engineering barriers: An analytical hierarchy process. *IET Software*, 15(4), 277–291. <https://doi.org/10.1049/sfw2.12022>
- Behutiye, W., Karhapää, P., Costal, D., Oivo, M., y Franch, X. (2017). Non-functional requirements documentation in agile software development: Challenges and solution proposal. En M. Felderer, D. Méndez Fernández, B. Turhan, M. Kalinowski, F. Sarro, y D. Winkler (Eds.), *Product-Focused Software Process Improvement* (Lecture Notes in Computer Science, Vol. 10611, pp. 315–341). Springer. https://doi.org/10.1007/978-3-319-69926-4_41
- Bukhsh, F. A., Bukhsh, Z. A., y Daneva, M. (2020). A systematic literature review on requirement prioritization techniques and their empirical evaluation. *Computer Standards & Interfaces*, 69(103389), 103389. <https://doi.org/10.1016/j.csi.2019.103389>
- Che Embi, Z., y Keong, S. L. M. (2022). A systematic review on non-functional requirements documentation in agile methodology. *Journal of Informatics and Web Engineering*, 1(2), 19. <https://doi.org/10.33093/jiwe.2022.1.2.2>
- Chung, L., Nixon, B., Yu, E., y Mylopoulos, J. (2012). *Non-Functional Requirements in Software Engineering*. Springer. <https://doi.org/10.1007/978-1-4615-5269-7>
- De Martino, V., y Palomba, F. (2025). Classification and challenges of non-functional requirements in ML-enabled systems: A systematic literature review. *Information and Software Technology*, 181(107678), 107678. <https://doi.org/10.1016/j.infsof.2025.107678>
- Design Council. (2025). *The Double Diamond*. <https://www.designcouncil.org.uk/resources/the-double-diamond/>
- Dongmo, C. (2024). A review of non-functional requirements analysis throughout the SDLC. *Computers*, 13(12), 308. <https://doi.org/10.3390/computers13120308>
- Eckhardt, J., Vogelsang, A., y Méndez Fernández, D. (2016). Are “non-functional” requirements really non-functional? An investigation of non-functional requirements in practice. En *Proceedings of the 38th International Conference on Software Engineering (ICSE '16)* (pp. 832–842). ACM. <https://doi.org/10.1145/2884781.2884788>
- Fatima, A., Shaheen, A., Ahmed, S., Fazal, B., Ahmad, F., Liew, T. W., y Ahmed, Z. (2024). Exploring the use of gamification in human-centered agile-based requirements engineering. *Frontiers in Computer Science*, 6. <https://doi.org/10.3389/fcomp.2024.1442081>
- Felfernig, A., Stettinger, M., Atas, M., Samer, R., Nerlich, J., Scholz, S., Tiisonen, J., y Raatikainen, M. (2018). Towards utility-based prioritization of requirements in open source environments. En *2018 IEEE*



- 26th International Requirements Engineering Conference (RE) (pp. 406–411). IEEE. <https://doi.org/10.1109/RE.2018.00-17>
- Gondal, M. A., Qureshi, N. A., Mukhtar, H., y Ahmed, H. F. (2020). An engineering approach to integrate non-functional requirements (NFR) to achieve high quality software process. En *Proceedings of the 22nd International Conference on Enterprise Information Systems (ICEIS 2020)* (Vol. 2, pp. 377–384). SciTePress. <https://doi.org/10.5220/0009568503770384>
- Hehn, J., y Mendez, D. (2022). Combining design thinking and software requirements engineering to create human-centered software-intensive systems. En J. Hehn, D. Mendez, W. Brenner, y M. Broy (Eds.), *Design Thinking for Software Engineering* (pp. 11–60). Springer. https://doi.org/10.1007/978-3-030-90594-1_2
- Jamasb, B., Khayami, S. R., Akbari, R., y Taheri, R. (2025). An automated framework for prioritizing software requirements. *Electronics*, 14(6), 1220. <https://doi.org/10.3390/electronics14061220>
- Kroop, S. (2025). Artifact validity in Design Science Research (DSR): A comparative analysis of three influential frameworks. En S. Chatterjee, J. vom Brocke, y R. Anderson (Eds.), *Local Solutions for Global Challenges. DESRIST 2025* (Lecture Notes in Computer Science, Vol. 15703, pp. 199–215). Springer. https://doi.org/10.1007/978-3-031-93976-1_13
- Li, B., y Nong, X. (2022). Automatically classifying non-functional requirements using deep neural network. *Pattern Recognition*, 132(108948), 108948. <https://doi.org/10.1016/j.patcog.2022.108948>
- Muntean, M., y Militaru, F. D. (2022). Design science research framework for performance analysis using machine learning techniques. *Electronics*, 11(16), 2504. <https://doi.org/10.3390/electronics11162504>
- Patrício, L., Varela, L., y Silveira, Z. (2025). Framework for integrating Requirements Engineering and DevOps practices in Robotic Process Automation with a focus on optimizing human–computer interaction. *Applied Sciences (Basel, Switzerland)*, 15(7), 3485. <https://doi.org/10.3390/app15073485>
- Peffer, K., Tuunanen, T., Rothenberger, M. A., y Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3), 45–77. <https://doi.org/10.2753/mis0742-1222240302>
- Robiolo, G., Scott, E., Matalonga, S., y Felderer, M. (2019). Technical debt and waste in non-functional requirements documentation: An exploratory study. En X. Franch, T. Männistö, y S. Martínez-Fernández (Eds.), *Product-Focused Software Process Improvement* (Lecture Notes in Computer Science, Vol. 11915, pp. 220–235). Springer. https://doi.org/10.1007/978-3-030-35333-9_16
- Rosado da Cruz, A. M., y Cruz, E. F. (2025). Machine Learning techniques for requirements engineering: A comprehensive literature review. *Software*, 4(3), 14. <https://doi.org/10.3390/software4030014>
- Saaty, T. L. (1980). *The analytic hierarchy process: Planning, priority setting, resource allocation*. McGraw-Hill. <https://archive.org/details/analytichierarch0000saat>
- Santos, S., Pimentel, T., Rocha, F. G., y Soares, M. S. (2024). Using Behavior-Driven Development (BDD) for non-functional requirements. *Software*, 3(3), 271–283.

<https://doi.org/10.3390/software3030014>

Werner, C., Li, Z. S., Ernst, N., y Damian, D. (2020). The lack of shared understanding of non-functional requirements in continuous software engineering: Accidental or essential?. En *2020 IEEE 28th International Requirements Engineering Conference (RE)* (pp. 90–101). IEEE. <https://doi.org/10.1109/RE48521.2020.00021>

Yamamoto, S. (2024). Non-functional requirements through exceptions. *Procedia Computer Science*, 246, 333–342. <https://doi.org/10.1016/j.procs.2024.09.412>

Yoo, I., Park, H., Lee, S.-W., y Ryu, K.-Y. (2024). Building traceability between functional requirements and component architecture elements in embedded software using structured features. *Applied Sciences (Basel, Switzerland)*, 14(23), 10796. <https://doi.org/10.3390/app142310796>